UNIVERSITY OF SOUTHAMPTON

Time Persistent Feature Detection via Phase Congruency

by

Peter J. Myerscough

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the
Faculty of Engineering, Science and Mathematics
School of Electronics and Computer Science

January 2006

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING, SCIENCE AND MATHEMATICS
SCHOOL OF ELECTRONICS AND COMPUTER SCIENCE

Doctor of Philosophy

by Peter J. Myerscough

The use of feature detection is a standard process within the computer vision community for simplifying a complex image to a more manageable representation. Feature detection is typically applied to individual images, even if they are a part of a more extensive image sequence. In this thesis we present new methods for feature detection via phase congruency, applied to image sequences. This work shows the improvements that can be gained from taking an image in its context. The first section of work focuses upon extending a previous feature detector, phase congruency, to operate on an image sequence. This new technique shows improvements in the robustness of the feature detector under increasing levels of noise. It also improves feature orientation description allowing for the component velocity of a feature to be evaluated. After further evaluation however this method produced undesirable results for fast moving features. In response to this, a novel method for evaluating phase congruency has been developed. The new method is achieved by modelling the filtering process used to derive phase congruency by measuring the standard deviation of the normalised energy response. Accordingly, the new method is termed statistical phase congruency. This new approach is implemented first for 2-D images, showing improvements over the initial image-based phase congruency technique. Furthermore, it is extended to detect time persistent features in image sequences whilst also providing improved results for detecting fast moving features. It is intended that the results of this work will provide a basis for detecting time persistent features under noisy conditions. The final portion of this thesis gives some conclusions and adds some direction for future work on these ideas.

Contents

List of Figures

List of Tables

Acknowledgements

This research has been funded by the EPSRC, with additional support from European Research Office of the US Army, Contract No.N68171-01-C-9002. Thanks also go to Ohio State University for some of the images and image sequences used in this thesis. Thanks go to Professor Mark Nixon for his guidance and thought on all areas of this project, as well as reminding me that I'm not completing 'a life, the universe and everything' Ph.D . Appreciations also go to Mike Grant and Oliver Blacklock for their discussions and wranglings during this research and to Lee Middleton and Matt for proof reading. Much thanks go to Ruth who has put up with me as I've struggled through this last year. Finally thanks go to God for whom I would not have started a Ph.D without.

Chapter 1

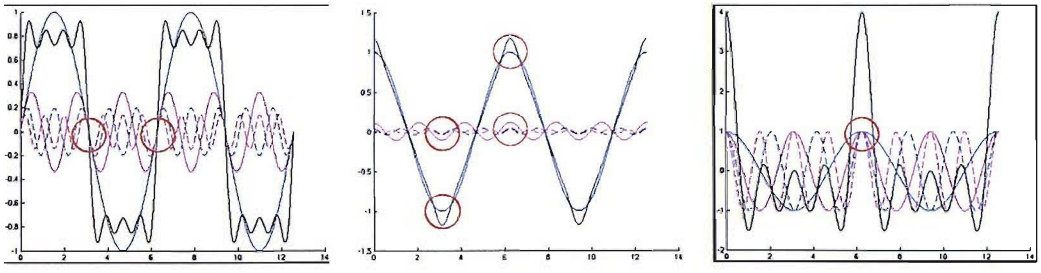
Context and Contributions

1.1 Feature detection

Feature detection has been an area of research within computer vision for a number of decades. The aim of feature detection is to reduce a complex image to a line drawing or similarly abstracted image. Initial operators[24] were simple kernels convolved with the image in question, such as the Prewitt, Roberts or Sobel operators. Since then detectors have been designed for different environments and different 'ideal' features. Canny[9] first stated three key design criteria of non-spurious response, single edge response, and correct location. Other important approaches with alternative bases include Spacek[30], Petrou[27], and Marr-Hildreth[18]. Each of these took a different view on what an important feature was, for example selecting 'step' features, or 'ramps', or alternative differentiation paradigms. Other approaches have used cues from human and biological vision. These include using statistical methods to 'learn' what is perceived as a feature from a human perspective [15]. Further cues have also been gained from work by Morrone[22] showing that humans respond to points of high underlying phase congruency. This enables explanation of work by Kovesi [16], which detects features based upon measuring this phase congruency. Examples of phase congruency are shown in figure 1.1.

These detectors differ in cost and performance, and have extensions to modify their behaviour. A common extension is that of sub-pixel feature detection, where the 'feature' is said to be positioned off the normal pixel grid [29]. Some operators are more robust, at the cost of computational effort. It is for these reasons that work continues in this area to allow the available choices when approaching such diverse problems as facial gesture recognition, automotive tracking or medical imaging to be broad enough for an acceptable solution to be found.

With the increase in computing power, it is becoming possible to process images in sequences. Much of the work within this thesis is particularly concerned with detecting



(a) Square wave (b) Triangle wave (c) Impulse wave

FIGURE 1.1: Three example waveforms showing high phase congruency highlighted by circles. Figures 1.1(a) to 1.1(c) show the component frequencies that the waveforms can be built up from. In the highlighted regions it is possible to see the component frequencies align and become phase congruent, even though their amplitudes are very different. In figure 1.1(a), an example of high phase congruency can be seen at the edge of a step function, and in figure 1.1(b) at the peak and trough of a triangle wave. Figure 1.1(c) also shows the occurrence of high phase congruency at the centre of an impulse function. In this way, features can be derived from phase congruent signals.

features within a sequence, both from a previously established single image feature detector[16], and also a detector described within this thesis.

Since we are concerned with extracting information from image sequences, the area of optical flow or motion estimation has also been instructive within the work. Optical flow estimation was first formulated by Horn and Schunck [12]. The initial ideas of estimating image motion as a low-level operator have been extended to use various bases of operation. Optical flow techniques can be classified as gradient based operators (or first and higher order differential methods) [13, 25], correlation based approaches [5], and frequency based approaches [10, 11]. A number of reviews of optical flow operators have been conducted [4, 20] comparing these and other techniques on synthetic and real image sequences. It has been noted that the different types of operator have different properties, and broadly speaking frequency and gradient-based methods are more tuned to smaller velocities (< 2 pixels per frame), whereas correlation based methods are better suited to larger velocities and do not have the same apparent sub-pixel velocity resolution. Both of these problems can be circumvented to an extent with the use of image decomposition [2], or possibly image expansion/interpolation. These techniques are the vital first step in describing moving objects and phenomena, but they also give insight when developing a moving feature detector.

Problems have been discovered in attempting to extract optical flow from an image sequence. Notably, the aperture problem, formulated by Marr [19], states that when looking at an image, or part of an image, if it has structure about only one axis, then full 2D velocity estimation is ill-defined. This will particularly affect any effort to detect full 2D velocity estimates of a feature, but should encourage component image velocity[11]

estimates to be considered acceptable results.

1.2 Time persistent feature detection

Although computer vision has been analysing image sequences for many years, the most commonly taken approach to detecting features that persist over time is to apply a feature detector upon individual frames, and then apply some method of tracking. This approach typically uses spatial correlation to assist in the detection of features, but does not take advantage of temporal correlation, i.e. the time persistency of features, until after the feature detection is finished. This temporal correlation could enhance the process of feature detection itself, rather than just being used to filter out features that do not persist over time.

There have been few attempts at detecting time persistent features. The closest to this thesis is that of Mulet-Parada[26], who developed a moving feature detector for use in echocardiography. This was based initially upon the work of Kovesi, and the concept of phase congruency, but removed the use of multiple scales and used the log-Gabor framework to measure asymmetry as a measure of 'edge' strength that was robust to noise. This is in contrast to this thesis that maintains the multi-scale approach and detects features independently of their symmetry.

Techniques for detecting motion boundaries, such as Mitiche[1], Spoerri[31] or Lui[17], are closely related to time persistent feature detection. Time persistent feature detection however, concerns an ability to detect features within an object that move with the same velocity as its surrounding region.

Other work exist in detecting features or surfaces in 3D, such as Pudney's surface detection via local energy and ridge tracking[7, 28], and Monga's recursive filtering technique in 3D [21]. There have also been discussions as to the most appropriate method for extending the Sobel operator[14, 6, 8] into 3D. Each of these detectors ignores the problems associated with the object's movement and not forming smooth surfaces in the 2D+T domain, an aspect this thesis overcomes.

There are a number of compelling aspects to detecting time persistent features. Firstly it is expected that a time persistent feature detector will be more robust to noise, extending the applications of computer vision into more difficult environments. Secondly, it also provides additional information with regards the velocity of a feature by describing a feature's spatio-temporal orientation. Detecting time persistent features may also help in the detection of optical flow by providing the motion boundaries within an image sequence. All these reasons mean that a time persistent feature detector could be a very useful commodity within the computer vision community and beyond.

1.3 Thesis overview

To summarise this thesis, we begin with background information into two previously published techniques. We describe the phase congruency operator, a robust feature detector, in section 2.1, and a moving feature detector in section 2.2. The contributions associated with this work are presented in three sections, firstly the temporal phase congruency operator, a new extension in time to phase congruency, is presented in chapter 3. Chapter 4 describes a novel approach to measuring phase congruency that specifically attempts to avoid some of the pitfalls of the earlier approach. This is then extended into 2D+T and shown to be useful as a time persistent feature detector in chapter 5. We finish with some final conclusions as well as some ideas for future work in chapter 6. Appendix A also includes work into using time persistent feature detection to help reduce blur in optical flow, which provided motivation for this work.

1.4 Contributions

A number of papers have been written in the course of this research: these are listed below. The first is included in Appendix A as part of the motivation for the thesis as a whole, but is not an integral part of the thesis itself.

P.J. Myerscough, M.S. Nixon, and J.N. Carter. Guiding optical flow estimation. In R. Harvey and J.A. Bangham, editors, *British Machine Vision Conference*, pages 679–688. British Machine Vision Association, 2003.

P.J. Myerscough and M.S. Nixon. Temporal phase congruency. In Proceedings *South-West Symposium on Image Analysis and Interpretation*, 2004.

P.J. Myerscough and M.S. Nixon. Measuring temporal phase congruency. In Proceedings *Symposium on Spatiotemporal Image Processing*. British Machine Vision Association, 2004.

P.J. Myerscough and M.S. Nixon. Estimating the phase congruency of localised frequencies. In Proceedings*IEEE International Conference on Image Processing*, 2004.

Chapter 2

Background

The material provided in this chapter gives not only a technical background to this thesis, but also provided some of the motivation for the thesis itself. Section 2.1 describes phase congruency, a robust feature detector, and our extensions to it. Section 2.2 describes the work of Mulet-Parada, who began to extend phase congruency into 2D+T, but in focusing on an application specific feature type removed much of the framework of the original phase congruency technique. It is included here as a motivation and an introduction into time persistent feature extraction.

2.1 Phase Congruency

Phase congruency is a robust feature detector. It detects not only step and line responses, but also a broader set of features [3]. Its robustness is found in its ability to detect features in images with high levels of noise. Its other attributes include a high degree of invariance to lighting variation within images and a normalised measure of feature strength. This thesis extends the phase congruency technique to work with image sequences.

The technique's main premise is that points of interest or features within an image can be highlighted by finding points in an image that have a high degree of phase congruency. The technique combines the measurement of phase congruency with a number of checks and balances. The first is that any feature must be composed of a spread of frequencies, this prevents pure sine waves from being detected as features. The second is that images typically contain noise, and it is helpful to be able to estimate the level of that noise so as to compensate for its effect.

Phase congruency is measured as a 1D phenomenon that occurs within 2D images. The approach of applying 1D feature detection is similar to the Sobel operator, in that there is a primary feature detector that is convolved with an orthogonal spreading function. In

phase congruency the primary method of feature detection is the relationship between a series of quadrature filters, see equation 2.1, that are convolved with a number of different orthogonal Gaussian spreading functions, see equation 2.2. This produces filters with different central wavelengths and orientations of interest. They are then grouped into sets of filters by orientation and used to generate a measure of frequency spread and a noise estimate per orientation. The combination of phase congruency and these other two measures proves to be a robust feature detector that is invariant to lighting variation across an image.

The following sections describe an initial naive measure of phase congruency (section 2.1.1), how to measure frequency spread (section 2.1.3), a method for estimating noise in images (section 2.1.4), and with a final section on how they are all combined to provide phase congruency as presented by Kovesi.

2.1.1 A Naive Measure of Phase Congruency

Phase congruency can naively be measured by convolving a set of filters with an image, and calculating the difference between the absolute of the sum of the filter responses and the sum of the absolute responses to the filters. The filters used are described in equation 2.3. This equation uses two filter types, one a log-Gabor filter with 'M' different central frequencies, $lg(\cdots)$, and the other a Gaussian filter with 'L' different orientations, $Ga(\cdots)$. These act along two orthogonal axes.

$$lg(\omega, \omega_m) = \begin{cases} \frac{1}{\sqrt{2\pi\sigma_\beta^2}} e^{\frac{-(\log(\omega/\omega_m))^2}{2(\log(\beta))^2}} & \omega \neq 0 \\ 0 & \omega = 0 \end{cases} \tag{2.1}$$

$$Ga(\theta, \theta_l) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(\theta-\theta_l)^2}{2\sigma_s^2}} \tag{2.2}$$

$$lg2D(\theta, \omega, \theta_l, \omega_m) = Ga(\theta, \theta_l) lg(\omega, \omega_m) \tag{2.3}$$

where θ and ω are the spatial angle and frequency axis, θ_l is the filter's angle of focus spatially and ω_m is the centre frequency of the filter. σ_s controls the spread of the filter about the spatial angle axis. β is a constant describing the bandwidth of the filter. The filters must be constructed in the Fourier domain because the log-Gabor function has a singularity at $\omega = 0$ because $log0$ is undefined. It is also worth noting that the filters in the form presented in equation 2.3 are a combination of a cosine and a sine based log-Gabor filter, and the response from each is separated into the real and imaginary parts of the response.

To measure phase congruency, PC, about a single orientation, θ_l, using these filters one could use equations 2.5 and 2.4.

$$R_m(x,y) = \mathbb{F}^{-1}\left[lg2D(\theta,\omega,\theta_l,\omega_m)\right](x,y) * image(x,y) \tag{2.4}$$

$$PC(x,y) = \frac{|\sum_{m=1}^{M} R_m(x,y)|}{\sum_{m=1}^{M} |R_m(x,y)| + \varepsilon} \tag{2.5}$$

where R_m is the response to the convolution of the image with the mth filter of M different central frequencies and ε is used to prevent divisions by zero. The processing of different spatial orientations separately to calculate feature strength suggests that if the correct spatio-temporal filters are constructed measuring temporal phase congruency in a spatio-temporal framework would be possible.

2.1.2 An Improved Phase Congruency Measuring Function

Although equation 2.5 is able to distinguish between points of high and low phase congruency, differences are very small between degrees of phase congruency, when the level of phase congruency is high. Kovesi therefore proposed an improved function that used the cross and dot products between the cosine and sine based responses and the mean phase angle. This is presented in equation 2.6

$$iPC_l(x,y) = \frac{\sum_{m=1}^{M} A_m(x,y)\left(\cos\left(\Delta\phi_m\left(x,y\right)\right) - \left|\sin\left(\Delta\phi_m\left(x,y\right)\right)\right|\right)}{\sum_{m=1}^{M} A_m(x,y) + \varepsilon} \tag{2.6}$$

$$\Delta\phi_m\left(x,y\right) = \phi_m\left(x,y\right) - \bar{\phi}\left(x,y\right) \tag{2.7}$$

where $A_m(x,y)$ is the magnitude of the response of the complex pair of log-Gabor filters and is equivalent to $|R_m(x,y)|$. $\Delta\phi_m$ is the difference between the phase of the mth filter, $\phi_m(x,y)$, and the mean phase, $\overline{\phi}(x,y)$. This improved measure of phase congruency provides a more linear variation in iPC_l with respect to changes in the underlying phase congruency in an image. This is not though the final form for measuring phase congruency that Kovesi presents, but it is the basis. There are additional factors of a measure of frequency spread and an estimate of noise in the image.

2.1.3 Measuring the Frequency Spread

Although a useful method for measuring phase congruency is shown in eqn. 2.5 it also detects high degrees of phase congruency when only one frequency is present in an image, for example a sine wave. Since this is an undesirable response, Kovesi developed a method to measure the frequency spread. Measuring the frequency spread uses two functions. The first provides an estimate of the frequency spread itself, and the second weights that measure to provide a more useful value. The measure of frequency spread is again calculated on a per orientation basis, with different filters varying by their wavelength. Here $w_m(x, y)$ is calculated as

$$w_m(x, y) = \frac{\sum_{m=1}^{M} |R_m(x, y)|}{(M|R_{max}(x, y)| + \varepsilon)} \tag{2.8}$$

where $|R_m(x, y)|$ is the absolute of the response to filter m and $|R_{max}(x, y)|$ is the maximum absolute response for all the filters from all M central frequencies. ε is a small constant used for avoiding division by zero which ensures that if the amplitude at a pixel becomes too small it is masked out. This is then mapped through a sigmoid function to produce $W_m(x, y)$ the measure of frequency spread.

$$W_m(x, y) = \frac{1}{1 + e^{(c - w_m(x,y))g}} \tag{2.9}$$

where c and g control the mapping of $w_m(x, y)$ to $W_m(x, y)$. $W_m(x, y)$ is then used to calculate the more robust implementation of phase congruency in section 2.1.5. Before this calculation is possible, an estimate of the noise within the image is calculated.

2.1.4 Calculating an Estimate of Noise in an Image

An estimate of the noise in an image is needed to reduce the amount of spurious responses that are produced by phase congruency. The estimate allows phase congruency to detect features that occur above the estimated noise's energy level whilst still not using energy specific feature detection. To provide this estimate of noise within an image, T, an orientation specific noise threshold is calculated. This is based upon a number of assumptions. These are that the noise is additive, that its power spectrum is even across the image, and that features occur infrequently within an image. Kovesi uses the estimated noise level in an image by subtracting it from the estimates of phase congruency.

To estimate the noise level, Kovesi considers the noise to be zero mean additive Gaussian noise. The mean magnitude of the noise is therefore a Rayleigh distribution described

in equation 2.10

$$X(x) = \frac{x}{\sigma_G^2} e^{\frac{-x^2}{2\sigma_G^2}} \tag{2.10}$$

where x is the magnitude of the noise and σ_G^2 is the variance of the Gaussian. The mean of the Rayleigh distribution and therefore the mean level of noise, μ_X, and the variance of the noise, σ_X^2, is defined in equations 2.11 and 2.12.

$$\mu_X = \sigma_G \sqrt{\frac{\pi}{2}} \tag{2.11}$$

$$\sigma_X^2 = \frac{4-\pi}{2} \sigma_G^2 \tag{2.12}$$

where both equations depend upon σ_G^2, the variance of the zero mean additive Gaussian noise, which is an unknown quantity. To estimate σ_G^2 Kovesi considers the expected value of the energy taken from the norm of the cosine and sine based filter pairs. The expected value of the energy squared is equivalent to twice σ_G^2. The expected value can then be calculated from the filters and filter responses used in the rest of the technique.

$$\mathbb{E}(E^2) = 2|\hat{g}|^2 \mathbb{E}\left(\sum_m N_m^2\right) + 4|\hat{g}|^2 \mathbb{E}\left(\sum_{i<j} N_i N_j\right) \tag{2.13}$$

where $\mathbb{E}(E^2)$ is the expected value of the squared energy responding to noise from all the filters used per orientation in phase congruency. N_i is the ith cosine or sine based filter. In the generation of a noise estimate the filters are not used in complex pairs, but as real valued filters represented here by N_i. Finally $|\hat{g}|$ is the amplitude of the noise spectrum. This is estimated from the previous assumption that an image contains infrequent features, therefore the smallest filter used in phase congruency will mainly be responding to background noise.

$$|\hat{g}|^2 \backsimeq \frac{-median(A_0^2)/ln(.5)}{\mathbb{E}(lg2D_0^2)} \tag{2.14}$$

where A_0 is the amplitude response, or energy of the smallest central frequency filter. The median value from all the responses in A_0 from an image is used. This helps to avoid the outliers generated by the actual features in the image. $lg2D_0$ is the smallest sine and cosine filter. The resolution of $|\hat{g}|^2$ allows for the full calculation the mean noise energy, μ_X, and its variance, σ_X. These two quantities are then used to define a value

T, see equation 2.15, which is subtracted from the absolute sum of responses.

$$T = \mu_X + k\sigma_X \tag{2.15}$$

where σ_X is multiplied by k to allow for different percentages of the noise to be subtracted, a value of between 2 and 3 allows for 60-90% of the noise to be removed, but also adds a slight bias to the resulting phase congruency estimate.

2.1.5 Calculating a Robust Measure of Phase Congruency

The phase congruency, PC, at each orientation, PC_l, is calculated from the sum of the log-Gabor filter responses, R_i.

$$PC_l(x,y) = \frac{\sum_{m=1}^{M} \lfloor A_m(x,y) pc_m(x,y) - T \rfloor W_m(x,y)}{\sum_{m=1}^{M} A_m(x,y) + \varepsilon} \tag{2.16}$$

$$pc_m(x,y) = \cos\left(\Delta\phi_m\left(x,y\right)\right) - \left|\sin\left(\Delta\phi_m\left(x,y\right)\right)\right| \tag{2.17}$$

$$\Delta\phi_m\left(x,y\right) = \phi_m\left(x,y\right) - \bar{\phi}\left(x,y\right) \tag{2.18}$$

where $\phi_m(x,y)$ is the phase at point (x,y) for central frequency m and $\bar{\phi}(x,y)$ is the mean phase across all filters at that point. The use of $\lfloor\ \rfloor$ denoted that if the quantity is negative it is set to zero. The sum is thresholded by the noise level estimate, T, and scaled by the measure of the spread of the energy mapped through a sigmoid function, $W_m(x,y)$. This is then divided by the total energy at the chosen orientation to produce a measure of phase congruency, PC_l, for that orientation. The choice of central frequencies depends on the size of the features of interest in the images undergoing feature detection. The difference between central frequencies is typically a factor of two and gives a relatively good coverage of the frequency domain. The smallest scale is typically three times that of a feature's width, so in images where features of interest are a pixel wide then the smallest scale is 3 pixels. Repeating and summing of the results for the 'L' orientations gives the improved phase congruency measure for the image, $PC(x,y)$.

$$PC(x,y) = \sum_{l}^{L} PC_l(x,y) \tag{2.19}$$

This summing could be changed to a different function that combines the different orientation responses in a non-linear manner.

2.2 Feature Asymmetry Information Rich Algorithm

FAIR, an intensity invariant feature detector, has been developed by Miguel Mulet-Parada and is described in [23, 26]. The technique took some of its inspiration from the work of Kovesi presented in section 2.1, but explicitly focused upon the application of feature extraction within the field of echocardiography. The resulting algorithm was formulated for single frames and sequences of echocardiographic images. The technique is designed to be resistant to speckle noise and is a phase-based technique allowing it to work well in the highly variable lighting conditions found in echocardiographic images.

The technique has four stages that are applied to echocardiographic data generating a rich feature map. The initial stage is a series of log-Gabor filterings selecting different orientations of data. This is followed by the calculation of an energy independent measure of asymmetry. The last two stages threshold the data based upon the measured asymmetry, and extract additional information from the filtering process to describe the detected feature's orientation and whether it is a rising or falling edge. This is in contrast to phase congruency and the work presented in this thesis, which maintains its ability to detect a broad range of features.

2.2.1 Log-Gabor filtering

The first stage of FAIR is the convolving of a series of log-Gabor filters with the image data. The filters are constructed using polar co-ordinates in the Fourier domain. The radial component acting along the frequency axis is a log-Gabor filter similar to that described in equation 2.1, with β fixed at 0.55. This gives the filter a fixed bandwidth of 2 octaves. The FAIR algorithm uses filters in just one scale of operation that can be tuned to get better results. The guidelines given based on the data sets available to Mulet-Parada suggested the focus, ω_m, of any filter to be equivalent to 32-56 pixels. The FAIR algorithm is designed to be used in equipment operated by a medical practitioner so this parameter would be tunable by the operator.

Convolved with the log-Gabor filter is a spreading function, $spr(\cdots)$. This allows image data to be filtered based upon orientation. Mulet-Parada argues that this particular spreading function allowed for more efficient interpolation of a feature's orientation.

$$lg(\omega, \omega_m) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma_\beta} e^{\frac{-(\log(\omega/\omega_m))^2}{2(\log(.55))^2}} & \omega \neq 0 \\ 0 & \omega = 0 \end{cases} \tag{2.20}$$

$$spr(\theta, \theta_l) = \left(1 - \left(\frac{2\,(\theta - \theta_l)}{\pi}\right)^2\right) \tag{2.21}$$

$$lg2D_{FAIR}(\theta, \omega, \theta_l, \omega_m) = spr(\theta, \theta_l) lg(\omega, \omega_m) \tag{2.22}$$

where ω is the frequency axis and ω_m describes the focus of the filter. θ represents the spatial angular axis with θ_l changing value to select filters of different orientations. This filter can be extended into the 2D+T domain by calculating the angular difference, $\theta - \theta_l$, in 2D+T space using a dot product.

These filters are typically multiplied with a Fourier transformed version of the image data. After inverse Fourier transformation the log-Gabor's even and odd components are represented in the real and imaginary parts of the complex result. This allows for a single inverse transform to calculate results for a filter pair.

The number and arrangement of orientations for the filters is described in section 2.2.3.

2.2.2 Measuring Asymmetry

The detection of points in an image that have a high asymmetry is closely related to the detection of step edges. Parada uses the measurement of asymmetry in an image or image sequence to detect the edge of the heart tissue in echocardiographic images. The first stage of this measurement process involves the use of log-Gabor filters as described in section 2.2.1. Each filter is produces complex results with the real part containing the response to an even log-Gabor filter, $e(X)$, and the odd part contained in the imaginary part of the response, $o(X)$, where X are the axes; spatial or spatio-temporal.

Equation 2.23 is used to calculate the asymmetry of a point in an image, X. This equation was shown in Parada's thesis to have a linearly varying response to the dominant phase of a feature. A step edge has a phase that is 90° different to an impulse, and for equation 2.23 is represented by a maximal response. The equation could also be interpreted in a more interesting manner. Since Parada is interested only in step edges, convolving the image with a step template would give a good level of step detection. If we then consider the Hilbert transform of the step we have an impulse like function, we can then improve our signal to noise ratio by penalising all the image points that match the Hilbert transform of our desired feature. This evaluation would give $|o(X)| - |e(X)|$, but with the inclusion of the denominator in eqn 2.23 the results can be normalised to be between 0 and 1, removing illumance variation in the results.

$$asym(X) = \frac{|o(X)| - |e(X)|}{\sqrt{o(X)^2 + e(X)^2}} \tag{2.23}$$

This robust calculation is then applied to the response of each filter pair generating a measure of asymmetry for each orientation about the chosen single central frequency of the filters. Since the response of this function to the phase of a feature is linear, Parada suggests a fixed thresholding based upon the desired variation from a pure step edge's response. This value was placed at .9, allowing for a phase variation of ±5°.

2.2.3 Filter Orientation Selection

In selecting a set of filter orientations two criteria were considered important to the development of FAIR. The first was to use as few as possible to allow for fast processing of image data. The second was that the responses from the filters should allow for a more accurate degree of orientation to be assigned to a detected feature. Therefore, it is important that the set of filters give a unique response to each different orientation of feature. In [26] it is shown that $2^{n-1}+1$ orientations are needed for unique responses from a set of filters, where n is the number of dimensions. In 2D this means 3 filters spread 30 degrees apart, in 3D (or 2D+T) this means 5 filters, but if 6 are used then they can be evenly spaced on the unit sphere, see figure 2.1.

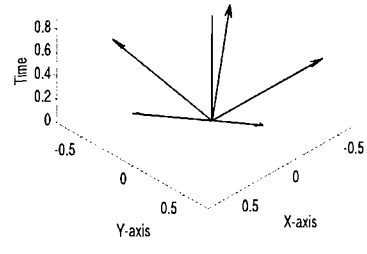


FIGURE 2.1: The focus of the 6 filter orientations used in the 2D+T FAIR algorithm.

It is known that the angular spreading function controls the filter response with respect to feature orientation. It follows then that if the angular spreading function is fitted to the data the orientation of the feature can be estimated. To simplify this fitting, a polynomial spreading function was chosen, see equation 2.21. For the 2D FAIR algorithm, the fitting then becomes a polynomial fit calculable via the pseudo-inverse of a Vandermonde matrix. In addition since all the spreading functions are known these matrices can be pre-computed. The nature of this calculation makes it affordable to do on each feature pixel detected within an image. This technique produced very positive results when testing. The mean error on a noiseless step was 1.18°, with a standard deviation 1.17°. This decreased if the number of orientations was increased, although for many applications this level of accuracy is sufficient, bearing in mind errors that can be attributed to discretisation of a feature. For the 2D+T FAIR algorithm, a different fitting method was used. It was commented that there are polynomial fitting algorithms available to fit a surface to 2D[1] data, but they are non-linear and were deemed too costly in computational time for the chosen application. Parada therefore proposed the use of an ellipsoid fitting function for the FAIR 2DT implementation.

[1]Although FAIR 2DT works in the 2DT domain orientations are only 2D.

2.3 Conclusions

Both the techniques presented in this chapter strongly support the work of this thesis. Phase congruency provides a robust intensity invariant feature detector. In its current formulation however, it can not be applied to image sequences. This prevents it from detecting features that persist over time, and lacks the benefits for robust feature detection that temporal correlation gives. It also does not provide the spatio-temporal orientation or velocity of a feature, stopping higher level computer vision techniques from using this information. The FAIR algorithm provides insight into extending filters into the spatio-temporal domain, but is limited to detecting features about a single scale. It is also only able to detect step edge features rather than the wider range of features of the techniques presented in this thesis.

Chapter 3

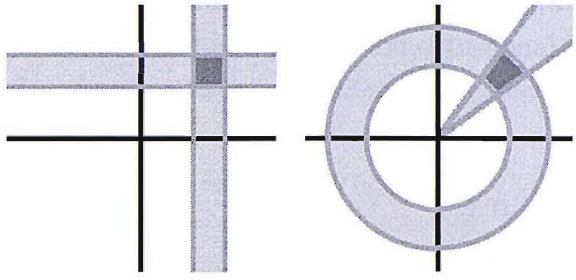
Temporal Phase Congruency

3.1 Introduction

In this chapter we describe the extension of phase congruency into the spatio-temporal domain. Creating this new temporal phase congruency operator has two main advantages. The first advantage is found in the orientation at which phase congruency can be detected at a particular pixel. This now describes not only its spatial, but also its temporal orientation. This is the same as describing its velocity. Therefore all features extracted with this extended method also have their velocity extracted.

Secondly the technique should be more robust to noise. This gain in robustness is justified by examining the neighbourhood of a pixel. In one-dimensional space a useful feature pixel exists in 'isolation'. In two-dimensions useful features are typically surrounded by supporting similar features allowing spatial techniques to require or benefit from this neighbourhood support. This improves the signal-to-noise ratio when processing an image. Therefore, when considering a point in 2D+T space, the supporting responses of a point's neighbours, in both spatial and temporal directions, increase the robustness of the detection process against noise.

This chapter presents a new extension of phase congruency into the 2D+T domain. Section 3.2 describes the new technique, including extending all aspects of phase congruency into 2D+T and a method for estimating a feature's primary spatio-temporal orientation. Results from applying temporal phase congruency to a series of test image sequences are shown in section 3.3, including sequences highlighting the technique's response to occlusion and noise, and examining its ability to measure feature velocity. The conclusions in section 3.4 cover the benefits and limitations of temporal phase congruency, and give further motivation for the work in chapters 4 and 5.



(a) example in a 2D Cartesian co-ordinate system being used to select a region using two orthogonal functions

(b) example in a 2D polar co-ordinate system being used to select regions using two orthogonal functions

FIGURE 3.1: Comparison of filter construction methods using Cartesian and polar co-ordinate systems to define a region in a 2D frequency space.

3.2 Method

We now extend phase congruency to use inter-frame data to enable estimation of time persistent features with resilience to noise. Kovesi's technique determined features in a two-dimensional image and used filters that were constructed from a one-dimensional signal, the log-Gabor function. This is convolved with an orthogonal spreading function, in this instance the Gaussian function. An additional spreading function (orthogonal to the two original functions) is used to create a three-dimensional (2D+T) filter to enable the detection of time persistent features.

3.2.1 Constructing filters under different co-ordinate systems

In our extension from the spatial(2D) to the spatio-temporal(2D+T) domain, we describe each stage of the phase congruency operator. The initial stage of the technique is the filtering of an image sequence. We construct filters using a polar co-ordinate system in the frequency domain continuing on from the work of Kovesi. These filters are separable about each polar axis. This approach is akin to successively applying different filters until a region is selected from the overall domain. In figure 3.1 the shape of the filter in a Cartesian system shows that it is roughly equivalent to a square, whereas using polar coordinates as in 3.1(b) the segment is an arc.

Using a polar co-ordinate system in the frequency domain we extend this from a 2D system with axes of (ω,θ) to 2D+T with axes of (ω,θ,ψ). In each case ω represents the frequency axis, θ represents the angle of that frequency with respect to the x-axis on the x-y plane, and ψ represents the angle with respect to the time axis. We then take three orthogonal functions to select particular regions of the 2D+T frequency domain. This

extends the arc shape in the 2D domain to a patch from the surface of a sphere, like a time zone of the earth in the 2D+T domain. The first orthogonal function, applied along the ω-axis, is a log-Gabor function as in eqn. 2.3 and repeated here in equation 3.1.

$$lg(\omega, \omega_n) = \begin{cases} e^{\frac{-(\log(\omega/\omega_n))^2}{2(\log(\beta))^2}} & \omega \neq 0 \\ 0 & \omega = 0 \end{cases} \tag{3.1}$$

where ω is frequency, ω_n is the tuning frequency of the filter, and β controls the spread of the filter. We then use two Gaussian functions as our other orthogonal functions along the θ-axis and the ψ-axis to generate the new filter, $lg2DT_n$.

$$lg2DT_n(\omega, \theta, \psi, \omega_n, \theta_n, \psi_n) = \frac{1}{\sqrt{2\pi\sigma_\theta^2}\sqrt{2\pi\sigma_\psi^2}} e^{\frac{-(\theta-\theta_n)^2}{2\sigma_\theta^2}} e^{\frac{-(\psi-\psi_n)^2}{2\sigma_\psi^2}} lg(\omega, \omega_n) \tag{3.2}$$

where ω represents the spatio-temporal frequency, θ represents the spatial angle of that frequency and ψ represents the temporal angle. θ_n and ψ_n are the angles the filters are focused upon, and σ_θ and σ_ψ control the spatial and temporal angular spread of the filters. Generating an even spread of these filters involves tiling the filters in a two dimensional orientation space. This allows for a wide range of tilings to be used and still generate an relatively even covering of the space.

An alternative method of interpreting spatial and temporal orientation would be to use a dot product between a filter's angular focus (θ_n, ψ_n) and the remainder of the Fourier space. This would then generate a uniform shape wherever the filter was focused in spatio-temporal angular space. This spreading function is described using Cartesian co-ordinates in equation 3.3 and combined with the log-Gabor function to give an alternative filter construction of $lgdot2DT_n$.

$$\begin{aligned} \delta\theta(\omega_x, \omega_y, \omega_t) &= cos^{-1}\left(\frac{\omega_x\omega_{x_n} + \omega_y\omega_{y_n} + \omega_t\omega_{t_n}}{\sqrt{\omega_x^2 + \omega_y^2 + \omega_t^2}\sqrt{\omega_{x_n}^2 + \omega_{y_n}^2 + \omega_{t_n}^2}}\right) \\ lgdot2DT_n(\omega_x, \omega_y, \omega_t) &= \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(\delta\theta)^2}{2\sigma^2}} lg(\omega, \omega_n) \end{aligned} \tag{3.3}$$

where ω_x, ω_y and ω_t represent the frequency domain equivalent of the x-axis, y-axis and t-axis. The vector $(\omega_{x_n}, \omega_{y_n}, \omega_{t_n})$ represents the filter's spatio-temporal angle of focus, and ω_n describes the frequency of the filter along the ω-axis. A single σ controls the spread of the filter in the angular axes. Arranging filters using this technique to achieve an even covering of 2D+T space is a packing problem in three dimensions. This limits the even arrangements of the filters in 2D+T space.

Using the polar separable construction allows for filters to be spread evenly about the spatial and temporal angle axes. If the filters were constructed using the dot product type spreading function then the number of arrangements for evenly distributed filters is limited to 4, 6 or 10, of which only 6 or 10 are valid if the orientation of a feature is to be extracted[26]. We have therefore chosen to construct 2D+T filters using equation 3.2 rather than equation 3.3 because it allows for any number of filters to be evenly positioned. This in turn allows for primary feature orientation to be extracted with five or more filters. Additional filter orientations should allow for secondary feature orientation to be extracted. Secondary orientations could occur at junctions of features that are not intrinsically 1D in nature. Increasing the number of orientations should also increase the robustness of the technique, if we assume that a feature's energy occupies a confined section of frequency space, and noise is broadband in nature. Filters that cover a smaller part of the frequency domain will therefore have higher signal to noise ratios.

Each filter is then multiplied in the frequency domain with a Fourier transformed version of the image sequence, and the result is inverse Fourier transformed to obtain the signal domain results. Previously stated in 2.1 the log-Gabor filters are required to be convolved in pairs, one a cosine based filter and the other a sine based filter. The above filter is in fact a combination of both these filters and when inverse Fourier transformation occurs the sine based results are present in the imaginary portion of the results and the cosine based results are in the real portion of the results. This removes the need to perform an inverse Fourier transform for each filter.

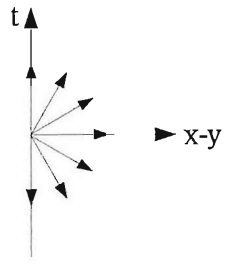
3.2.2 Selecting the filter orientations

There are a number of different schemes for selecting the arrangement of filters. The initial arrangement was to have each filter separated by 30 degrees spatially and temporally. This allows two schemes to be proposed : one appears to exhibit 'singularities' [1] with filters centred about $\psi = 0$ and $\psi = \pi$, fig. 3.2(b), but not extending to negative values or values greater than π, The other scheme has the temporal angles offset by 15 degrees from the 'singularities' in fig. 3.2(d). In each case the spatial orientation is kept the same, so both of these filter distributions have been used in the work.

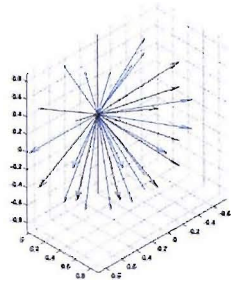
3.2.3 Extending the helper functions into 2D+T

After each filtering operation the resultant filtered image sequence is essentially a 1D signal orientated at the angle of the filter used. In the 2D domain equations 2.8, 2.9, and 2.19 all process 1D signals orientated in a 2D 'block'. Extending these equations to

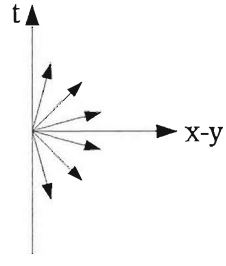
[1] These 'singularities' are similar to the north and south poles of the earth. On reaching the north pole it is impossible to travel further north, Similarly, the filters do not spread beyond t=0 where a similar 'singularity' point lies.



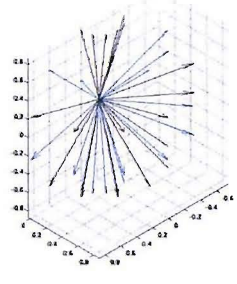
(a) Temporal axes filter orientations with 'singularities'



(b) 42 filter orientations with 'singularities'



(c) Temporal axes filter orientations without 'singularities' point



(d) 36 filter orientations without 'singularities' point

FIGURE 3.2: This figure shows the differences between two sets of orientations of filters in 2D+T space, where each is defined as being 30 degrees apart in both spatial and temporal axes. (The different colours are used to help distinguish the different vectors in the figures.)

operate using a 2D+T 'block' is a matter of substituting their axes of operation from (x, y) to (x, y, t).

3.2.4 Feature velocity estimation

In the process of detecting features in 2D+T space, we can also gather useful information on the spatio-temporal orientation of the feature or velocity. The first approach to estimating the velocity of a feature is to find the orientation of the filter that produced the largest temporal phase congruency. This will give one of 36 (or 42) different velocities. This is a very poor estimate of the true velocity of a feature. To improve the estimate we also consider the responses from orientations around the strongest response. These responses can be used to interpolate a more precise localised velocity estimate, for each feature point.

Also since the features we detect are not extracted from a 1D signal, a point may have more than one strong orientation, i.e. it may be a 2D feature like a corner or junction between lines or a more complex 2D+T feature. In the first instance however, it is important that the response of all the filters to a 1D feature about various spatial and temporal angles is investigated.

If we were calibrating the original technique by Kovesi (that operates upon single images), then we could examine just one orientation, and gather the response of that orientation to features of different orientation allowing us to generate a mapping from different orientation responses to a known orientation or to generate a function for approximation or interpolation. This differs from temporal phase congruency in that it operates upon image sequences, which can suffer from the effects of temporal aliasing, as features move. Four sets of image sequences were used to calibrate the new temporal phase congruency operator. The first set was a series of image sequences containing a single anti-aliased step edge in each image. In each sequence the edges moved at different speeds (temporal angle) and had different spatial orientations. The spatial angles varied from 0 to 90 degrees in 5 degree increments. The temporal angles ranged from 0 to 90 degrees. For the first set of image sequences the temporal angles were converted to velocities and the feature position was calculated and an anti-aliased step edge was generated at that position. In testing these spatio-temporal angles against all orientations in the temporal phase congruency operator the other quadrants can be mapped out. To contrast this, a second set of image sequences was generated where the temporal angles were anti-aliased. In both of these test cases the anti-aliased step was positioned between pixels as shown in figure 3.3(a), a further two sets were generated with the feature point centred on a pixel, see figure 3.3(b). This slightly changes the profile of the feature, possibly towards a 'ramp' like feature, except that surrounding pixels display a feature width of just one pixel.

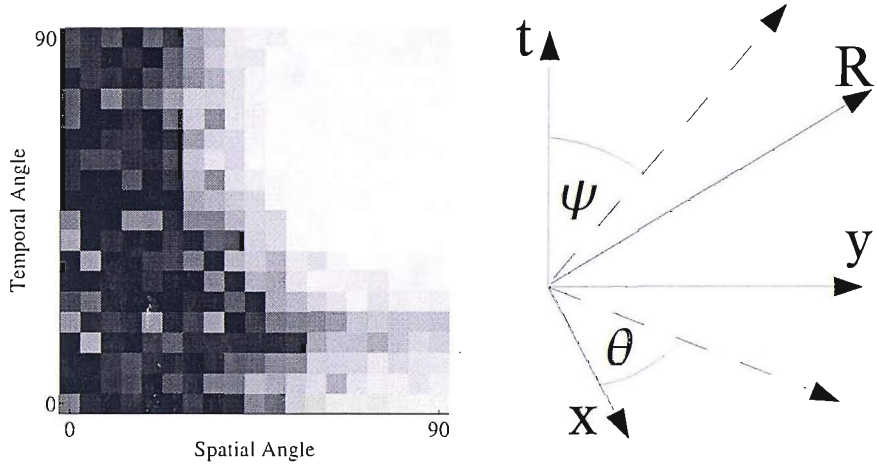
(a) A step edge feature centred 'between' pixel co-ordinates

(b) A step edge feature centred on integer pixel co-ordinates

FIGURE 3.3: These figures show step edges centre 'between' pixels and in the middle of a pixel.

Applying non-maximal suppression[24] to an output image is not useful to understand the response of temporal phase congruency to a feature at different orientations because of its reliance upon edge orientation data. Therefore in the case of the feature being

centred between pixels, the mean value of the four surrounding pixels was collected. In the case of the feature being centred on an integer pixel value that value was collected. These results were then arranged into images with the pixel position being the angle of the edge in the image sequence, and its strength being the value extracted from after feature detection at the centre pixel(s). The images were 19x19 pixels in size, with the horizontal axis describing a 90 degree variation in the spatial angle of the feature, and the vertical axis describing a 90 degree temporal variation. Images can then be generated combining the responses from the separate responses of the different filter orientations, and the different feature types. An example image is shown in figure 3.4.



(a) Shows all orientation

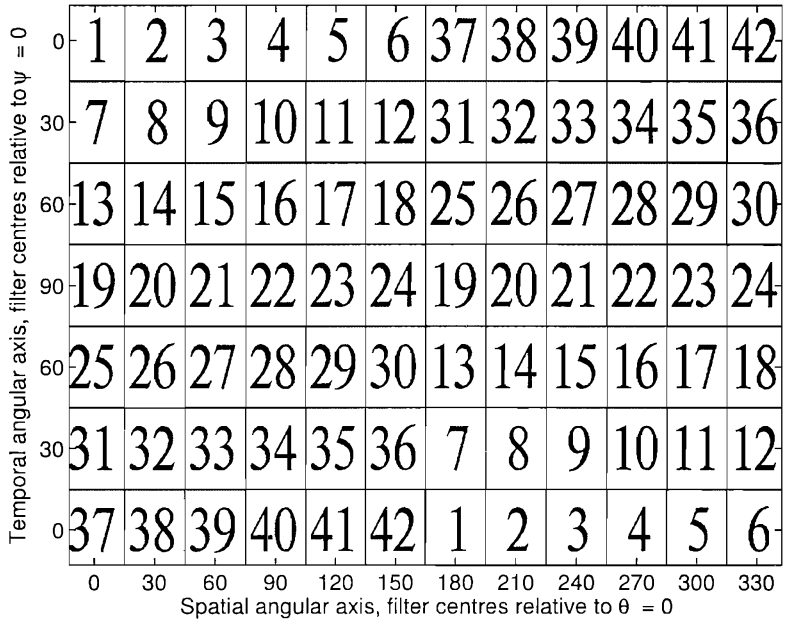
(b) Phasor description of angles

FIGURE 3.4: The responses from applying temporal phase congruency to an image sequence containing just one feature. The pixel intensities correspond to the feature strength detected at a single orientation with white being a strong response and black a weak one. The pixels are arranged such that pixels nearer the top of the graph are results from 'faster' features. The pixels also vary from left to right as the angle of the test feature is rotated through 90 degrees. A phasor plot in figure 3.4(b) shows the relationship between the response, R, and the spatial angle, θ, the temporal angle, ψ, and the x,y and t axes.

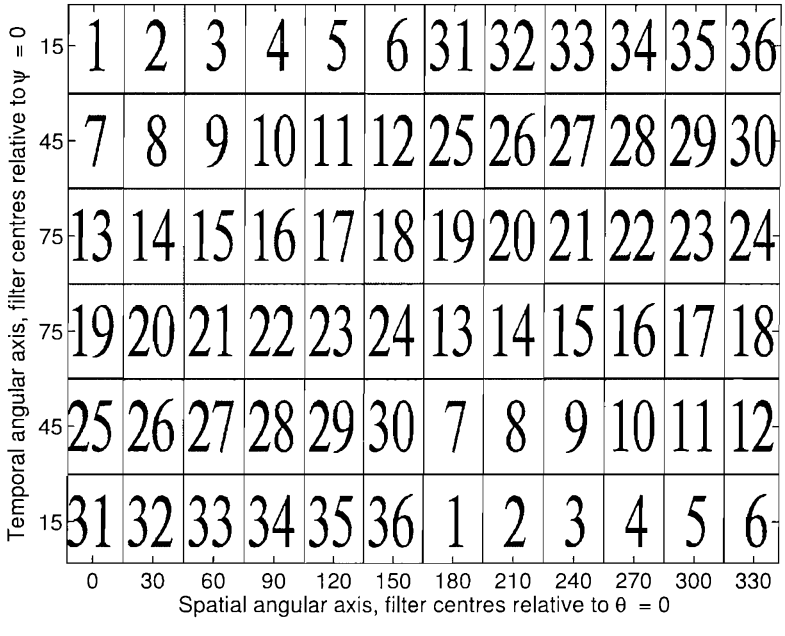
In the example image in 3.4(a) the strongest responses in white are in the top right of the image. This signifies that the orientation of the filter is centred about a spatial angle of 90 degrees and a temporal angle of 90 degrees. From these results we can tell that this orientation is tuned to primarily detect stationary or very slow moving features. There are also some stronger responses in the bottom right hand corner of the image, these are associated with aliasing in the test sequence causing this filter to also detect a stronger response away from its central focus.

Sets of these images are presented in tables of figures containing 36 or 42 images, see figures 3.6(b)-3.8(b). The images are each results from filters tuned to different central

frequencies. They are then arranged such that the filters tuned to the smallest temporal angle are positioned on the top row. Each successive row contains results from filters with a central frequency that is temporally 30 degrees more than the previous row. The middle row(s) show the slowest orientations with further rows 'increasing' in speed. The central frequency can also be altered in the spatial angle and so the central frequency of the filters used to calculate an image is incremented by 30 degrees between each horizontal image, with the final one wrapping around to the first. It is due to this wrapping around that the images are repeated in the table of images to show the relationships between the orientations more completely on paper. The two variations of the orientations are presented as in figure 3.5(a) for the singularity, and in figure 3.5(b) for the non-singularity arrangement.



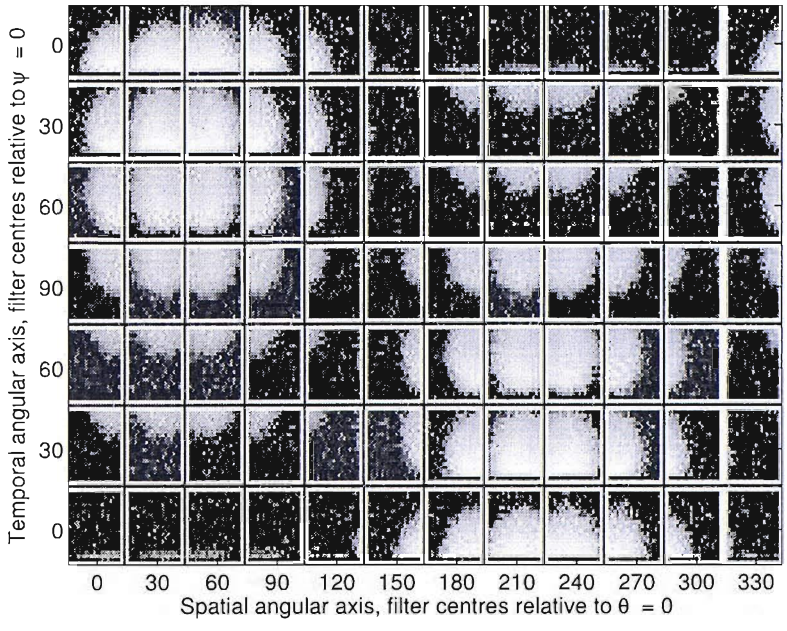
(a) singularity layout



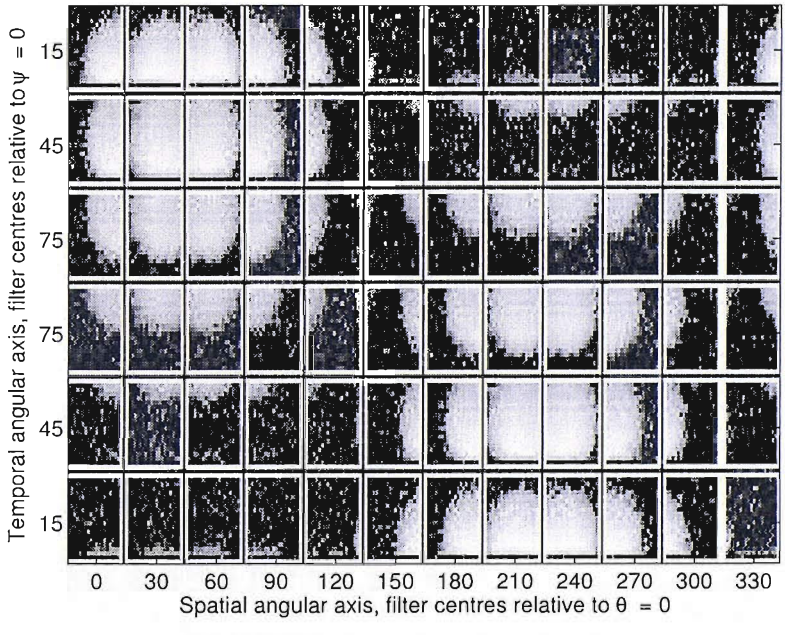
(b) non-singularity layout

FIGURE 3.5: These plots show the layout of the responses of temporal phase congruency to features at different spatial and temporal angles using the singularity fig. 3.5(a) and non-singularity fig. 3.5(b) arrangement of filters. The images vary in spatial angle by column, with 30 degrees between each column of images. The images vary in temporal angle by row, with the fastest at the top and each lower row reducing by 30 degrees.

Temporal phase congruency uses Gaussian spreading functions about the spatial and temporal angle axes when generating the filters. In a noise free signal the filters will respond significantly to energy away from its angle of focus because the Gaussians never reach zero. To prevent this a small amount of additive Gaussian noise (zero mean, sigma equal to 10% of the size of the step) was added to the test sequences. The results show an understandable structure, and we see that the response of the temporal phase congruency operator is as expected, in that the shape of the responses appears Gaussian.

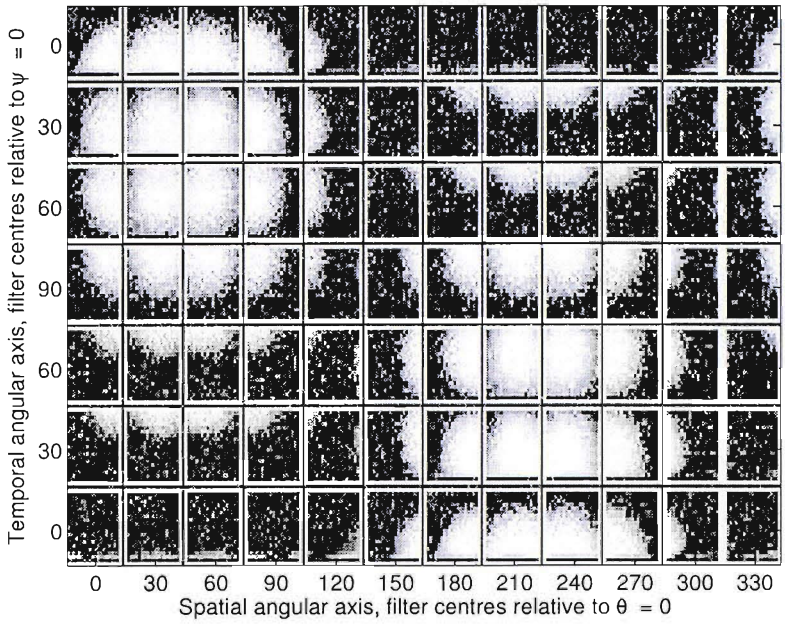


(a) results from filters using the singularity layout

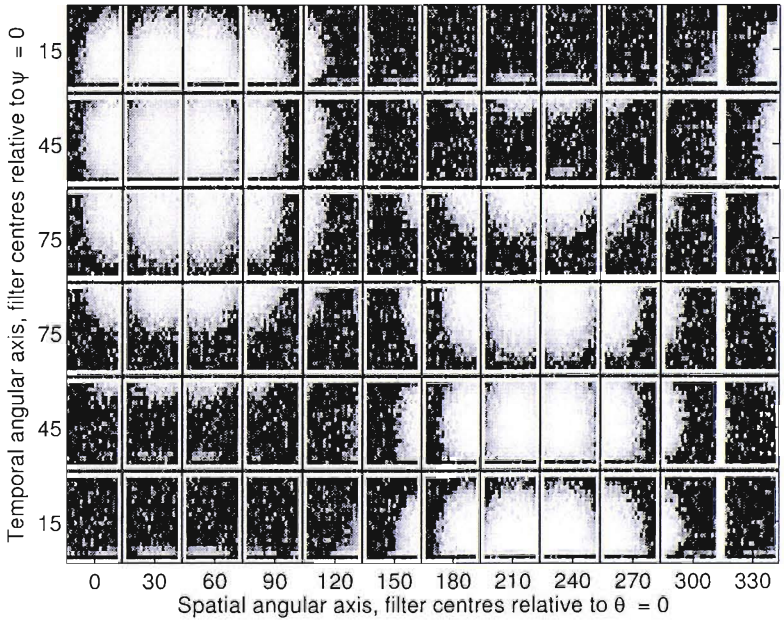


(b) results from filters using the non-singularity layout

FIGURE 3.6: These plots show the response of temporal phase congruency to features at different spatial and temporal angles. The images used were anti-aliased temporally, and the features were centred between pixels. Each of the images was also subjected to zero mean additive Gaussian noise with a sigma of 10% of the difference between the high and the low of the step.



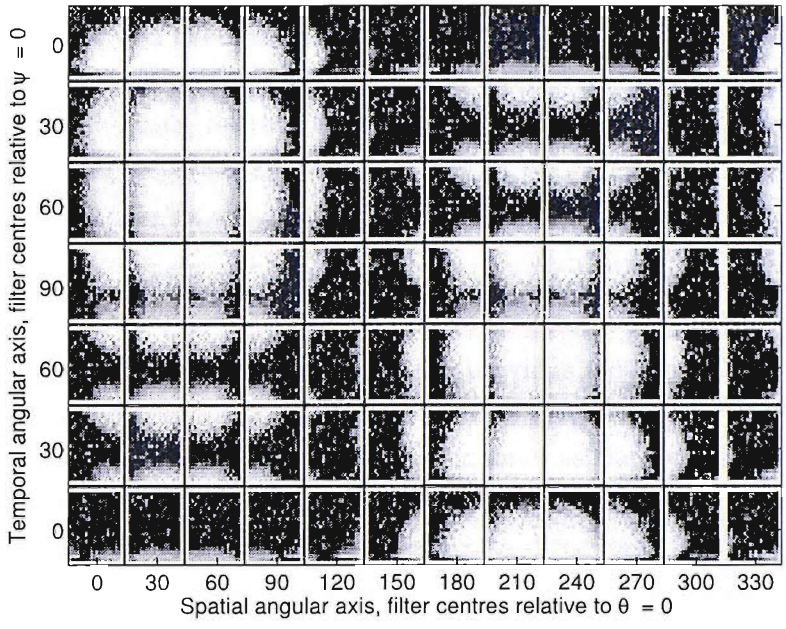
(a) results from filters using the singularity layout



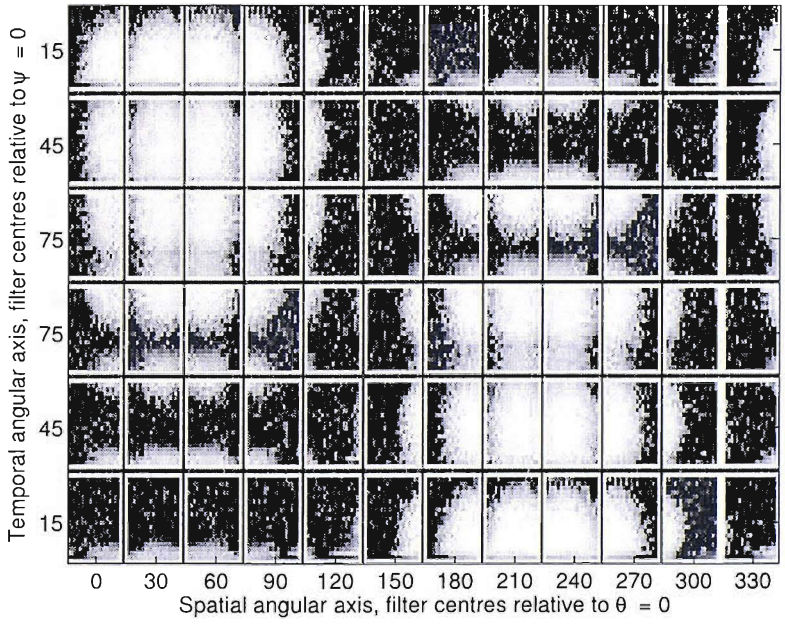
(b) results from filters using the non-singularity layout

FIGURE 3.7: These plots show further responses of temporal phase congruency similar to that of figure 3.6(b). The difference between the two figures is that the test features for this figure were centred in the middle of a pixel.

From figures 3.6(a), 3.6(b), 3.7(a) and 3.7(b) there is an obvious structure to the response of temporal phase congruency to features at different orientations. These figures contain results on features anti-aliased in the temporal axis with each sub-plot showing a 'window' of how a single orientation of temporal phase congruency responds to features at different orientations. The combined effect of grouping the plots suggests that the individual orientations detect features at a strength that varies relative to the angle between the filter and the feature, in a Gaussian like manner. It is also worth noting that the different edge types show little difference and so we only present results for the features centred in the middle of a pixel. The results for aliased image sequences show a more complicated behaviour.



(a) results from filters using the singularity layout



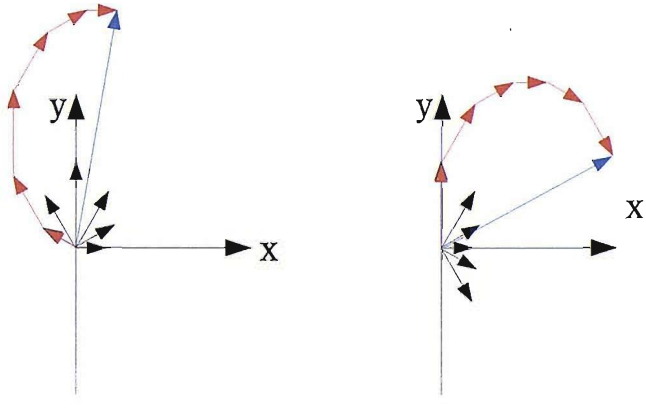
(b) results from filters using the non-singularity layout

FIGURE 3.8: These plots show the response of temporal phase congruency to features at different spatial and temporal angles. The images used contained no temporal anti-aliasing, and the features were centred in the middle of a pixel. Each of the images was also subjected to zero mean additive Gaussian noise with a sigma of 10% of the difference between the high and the low of the step.

Figures 3.8(a) and 3.8(b) show similar structure to figures 3.6(a)-3.7(b), but they also have erroneous responses due to temporal aliasing. This can be seen in the lower parts of the plots particularly in the second and third columns in the rows where $\psi = 45$ or 75. These slower channels are detecting high phase congruency at 0°, due to aliasing, and 90°, due to the features, on the temporal axis. This effect though does not affect the spatial variation in responses.

3.2.5 Orientation Estimation

From the experimental evidence presented in the previous section and from the design of the technique, feature responses vary with respect to the difference between the initial filter's angle or focus and the feature in a similar manner as the spreading function. We have therefore developed a geometrically inspired method which treats the orientation responses as amplitudes and the orientations as vector directions. The weighted vectors were then summed together. The mean angle of the responses is then taken as the estimate of the feature's primary orientation. Since the orientations of the filters all occupy one half of the possible spatio-temporal angles, the resulting sums would all be biased towards the centre of this region. To avoid this the weighted basis vectors are summed a number of times with different centres. Example sums are shown graphically in figure 3.9



(a) correctly biased orientations

(b) wrongly biased orientations

FIGURE 3.9: Two ways of summing feature strengths from different angles. Black vectors represent the responses at the described orientation with longer vectors depicting greater response. The red vectors show the summing of the vectors with the resultant vector in blue. The correctly biased plot, figure 3.9(a), has a greater resultant vector than figure 3.9(b).

The sum that produces the greatest response has the least amount of destructive summation, and hence should be the best estimate. Mathematically this is described in

equation 3.4.

$$est_s = \sum_{l=1}^{k} tpc_l e_l + \sum_{l=k+1}^{L} tpc_l(-e_l) \tag{3.4}$$

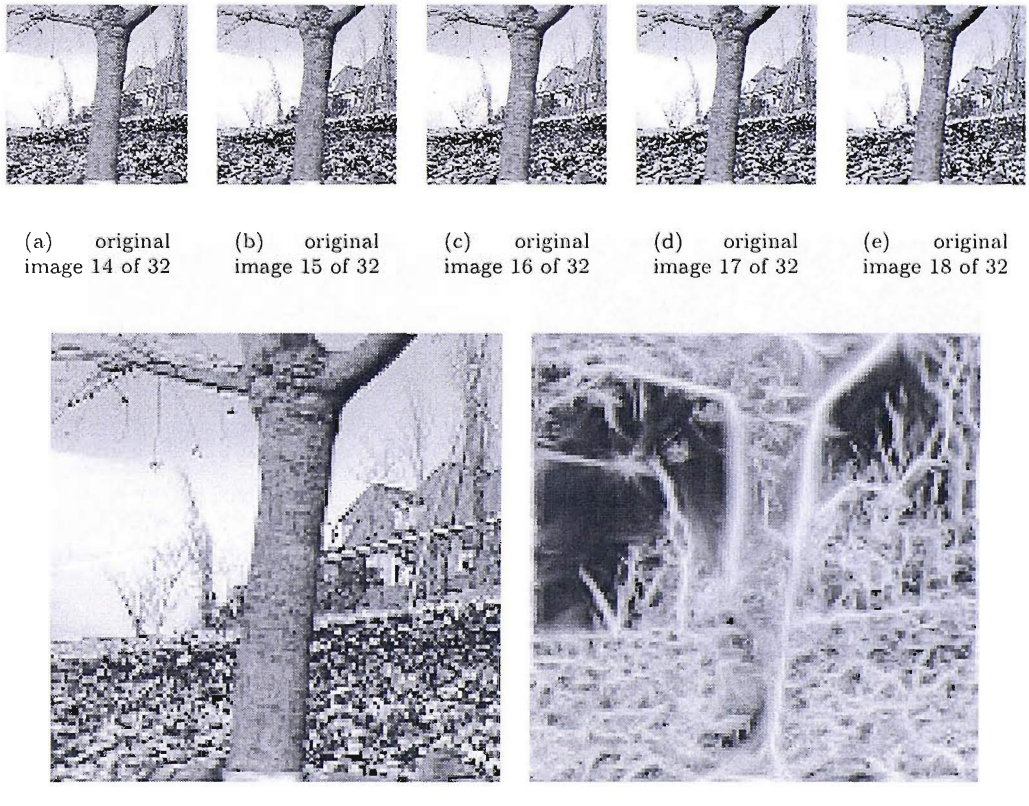
where tpc_l is the measure of temporal phase congruency measured using the lth filter. e_l is a basis vector that has the same orientation as the lth filter. All the basis vectors range between 0 and π radians, with the negative basis vectors ranging from $-\pi$ to 0 radians. Whilst varying k between 1 and N the result, est_s, that has the greatest value is the best approximation to a feature's orientation. For this calculation the feature strengths, tpc_l, are sorted so that the lth filter's angle is between that of the $(l-1)$th and $(l+1)$th filter's angle.

This calculation is repeated separately for the temporal axis. In both instances the method produces acceptable results relatively quickly.

3.3 Results

The results for temporal phase congruency are presented in six sections examining different performance factors. The first section gives some examples of the output of the temporal phase congruency operator. The second section gives results for the effects of noise on the new technique and compares it to the original. The following section examines how the technique responds to occlusion. Section 3.3.4 contains results for tests on the accuracy of temporal phase congruency to estimate the orientation of a feature along with some example results of the orientation estimates. The fifth section shows how a sequence of images can be filtered according to velocity information to select, for example, static objects or quickly moving objects. The final section examines in more detail some of the problems that temporal phase congruency suffers, describing some of the limitations of phase congruency.

3.3.1 Example results



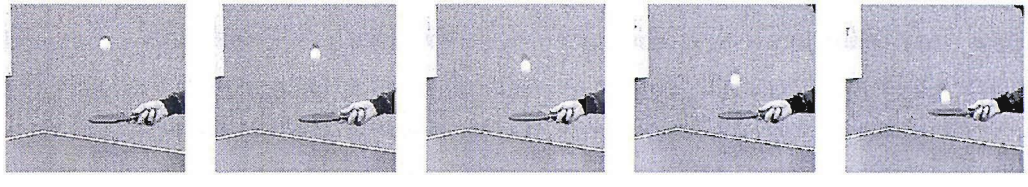
(a) original image 14 of 32 (b) original image 15 of 32 (c) original image 16 of 32 (d) original image 17 of 32 (e) original image 18 of 32

(f) original image 16 of 32 (g) temporal phase congruency output

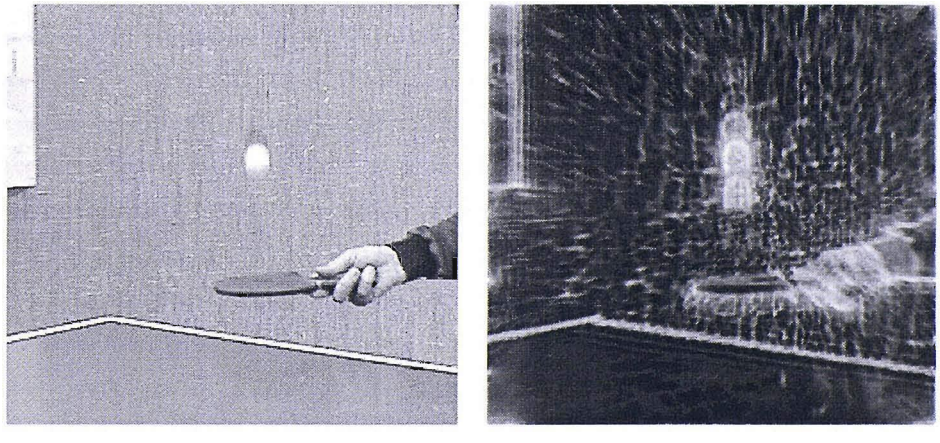
FIGURE 3.10: Results from the phase congruency operator on a tree sequence. Figure 3.10(g) is generated by taking the maximum response from each of the different orientations.

Figure 3.10 shows the results of applying temporal phase congruency to a sequence of

a camera moving parallel to a tree in front of a row of houses. The five central frames of the sequence are shown in figures 3.10(a) to 3.10(e) with the frame 16 shown again in figure 3.10(f) alongside figure 3.10(g) showing results of applying temporal phase congruency to frame 16. This sequence was chosen because it is commonly used in optical flow analysis due partially to the different layers of movement of the tree, the flower bed, and the houses in the background. These results, shown in figure 3.10, show the roofs of the houses in the background have been detected. The tree trunk, which moves relative to the camera, displays some minor blurring, but considering the high degree of variation in texture along its right side the detected strength of the edge remains quite constant. The lower left side of the tree trunk is not so strongly detected, but in the original image the boundary is very ill defined and is more perceivable by a change in texture rather than a bounding line. It is important to note that the edge data that is extracted with this technique also contains velocity information as well as the normal spatial information the original phase congruency technique extracted.



(a) original image 14 of 32 (b) original image 15 of 32 (c) original image 16 of 32 (d) original image 17 of 32 (e) original image 18 of 32



(f) original image 16 of 32 (g) temporal phase congruency output

FIGURE 3.11: Results from the phase congruency operator on a ping-pong ball sequence.

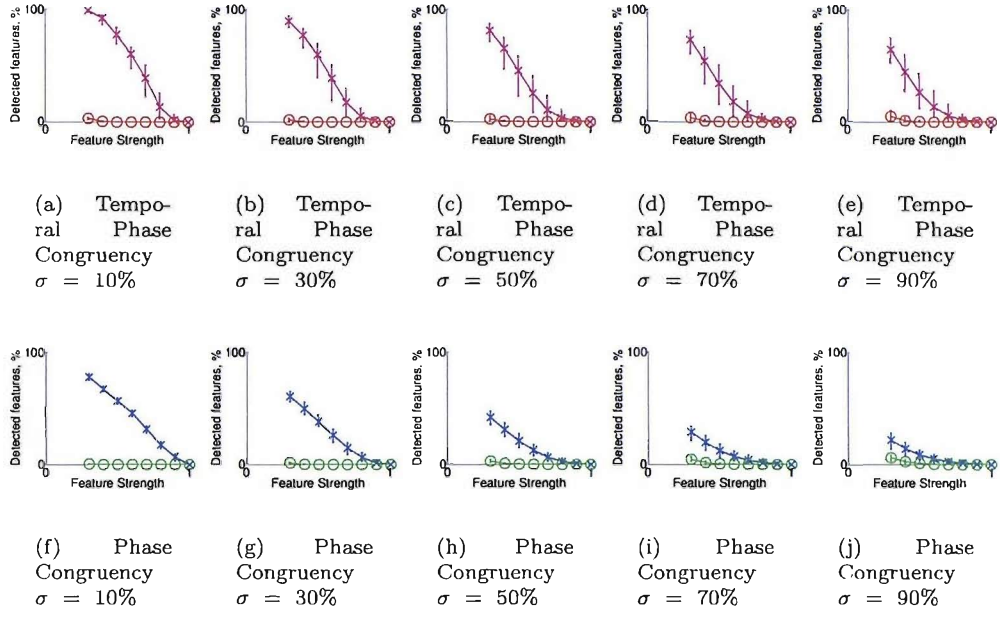
A sequence of images of a person bouncing a ping-pong ball is often used in the evaluation of techniques for optical flow calculation; some selected images from this sequence are shown in figures 3.11(a)-3.11(e) Figure 3.11(f) is taken from the middle of a sequence of images of a person bouncing a ping-pong ball on a table tennis bat. In the

processed image, figure 3.11(g), the technique has highlighted strongly the edge of the table. In the sequence the table is not moving significantly as such, it is persistent as is the background. The background also appears 'noisy', this is due to the texture of the background which appears hashed with thin lines across it at many angles also being strengthened by their appearance in many frames. The ball is the fastest moving object within this sequence and shows 'echoes' from the previous and next frames appearing above and below its current position. These echoes are responses to the presence of the edge of the ball in the surrounding frames. This is an artifact of the current implementation. The echoes should be removable using the velocity information of the features as they are the positions of the ball in the surrounding frames, this could then leave the output clean with just one copy of the ball.

3.3.2 Noise Testing

The results shown in figures 3.10 and 3.11 show successful extraction of moving and stationary time-persistent features. In addition to this, a series of noise tests was performed to gain a deeper insight in to the new technique. Two types of noise were used in increasing amounts with a synthetic image sequence of a slowly moving circle. The first noise used to test temporal phase congruency was additive Gaussian noise, where a zero-mean Gaussian distributed random number was added to each pixel. The second noise type was salt and pepper noise, where the 'noisy' pixels were randomly selected and then 50% were set to black and 50% to white, again using a random number generator. We used additive Gaussian noise because it is commonly used in noise testing and since the central limit theorem suggests that other forms of noise can in amalgamation be considered to be Gaussian distributed. The second type of noise was chosen because an impulse affects the whole spectrum and so applying salt and pepper noise adds a certain degree of broadband noise to a signal affecting the whole of the frequency space. In addition to this it could also be viewed as removing information from the images, as individual pixels are masked and replaced with a new random pixel. The resulting feature maps are then thresholded at different levels and compared with ground truth, where acceptable positions for a feature are compared with the actual feature map. Ground truth is defined here with all the pixels bordering a different colour pixel being considered a valid edge pixel when viewing a noise free version of the image. Decreasing the threshold increases the number of pixels detected as significant and so as the threshold is lowered the number of pixels with where features are detected is cumulative. These cumulative results are plotted as a percentage of features that are both positively and falsely detected as in figures 3.12 and 3.13. We also applied the same tests to Kovesi's image based phase congruency[16] to provide a comparison.

Results in figure 3.12 show that temporal phase congruency is significantly more robust when attempting to detect features in a noisy environment than single image based



(a) Temporal Phase Congruency σ = 10%
(b) Temporal Phase Congruency σ = 30%
(c) Temporal Phase Congruency σ = 50%
(d) Temporal Phase Congruency σ = 70%
(e) Temporal Phase Congruency σ = 90%

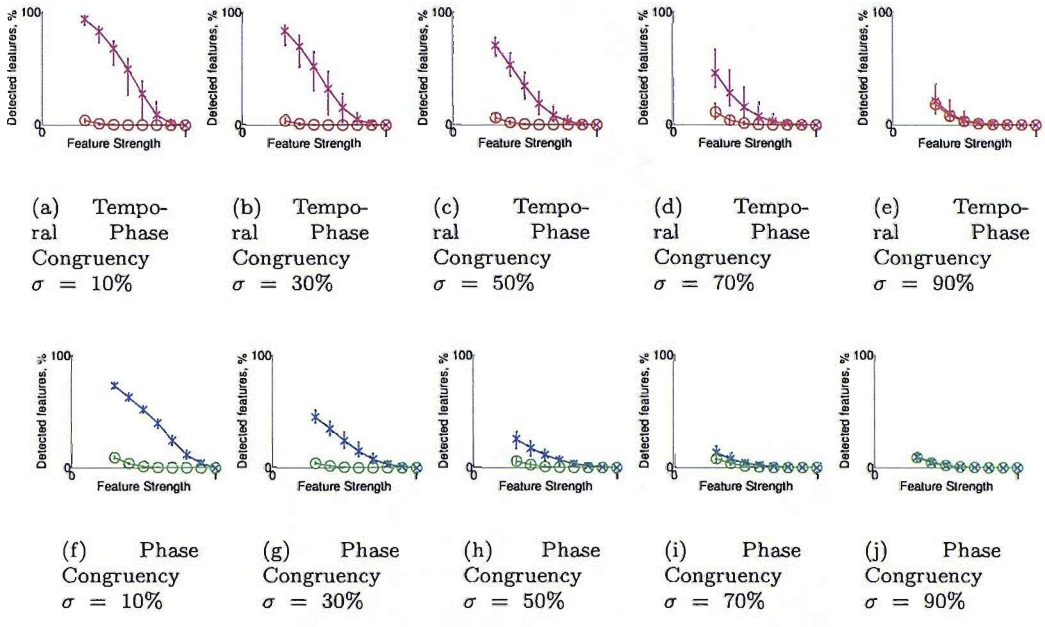
(f) Phase Congruency σ = 10%
(g) Phase Congruency σ = 30%
(h) Phase Congruency σ = 50%
(i) Phase Congruency σ = 70%
(j) Phase Congruency σ = 90%

FIGURE 3.12: Results from testing a simple white disc moving on a black background with increasing amounts of Gaussian noise added. Crossed magenta or blue lines represent true features detected, circled red or green lines represent false features detected. Error bars show the range of results due to the test being run fifty times.

phase congruency. At moderate levels of noise, σ = 50% of the difference between black and white in the image, temporal phase congruency detects about 90% of the features whereas image based phase congruency detects less than half the target features. Similarly, where the noise level is increased to σ = 90% the detection rate of image based phase congruency is very low, whereas temporal phase congruency is detecting more than 50% of the target features. It is interesting to also note that the shapes of the graphs appear very similar between phase congruency and temporal phase congruency.

The results for the salt and pepper noise tests in figure 3.13 also show that the temporal phase congruency measure is detecting more feature pixels of the circle as it moves through the sequence, and with a higher measure of feature strength or significance. Again at σ = 50% almost 80% of the features are detected succesfully whereas image based phase congruency has fallen to detecting around 30% of the target features. It also shows the limits of the technique, which is informative. The results are still very good considering the final two graphs in figures 3.13(d) and 3.13(e) have 70% and 90% of the image pixels removed and replaced with a random black or white pixels. These results are consistent for both types of additive noise across the whole range of noise levels with the new technique showing better detection rates and lower false detection results.

A final set of examples shown in figure 3.14 show the massive improvement in detection that can be achieved using temporal phase congruency when compared to image based



(a) Temporal Phase Congruency σ = 10%
(b) Temporal Phase Congruency σ = 30%
(c) Temporal Phase Congruency σ = 50%
(d) Temporal Phase Congruency σ = 70%
(e) Temporal Phase Congruency σ = 90%

(f) Phase Congruency σ = 10%
(g) Phase Congruency σ = 30%
(h) Phase Congruency σ = 50%
(i) Phase Congruency σ = 70%
(j) Phase Congruency σ = 90%

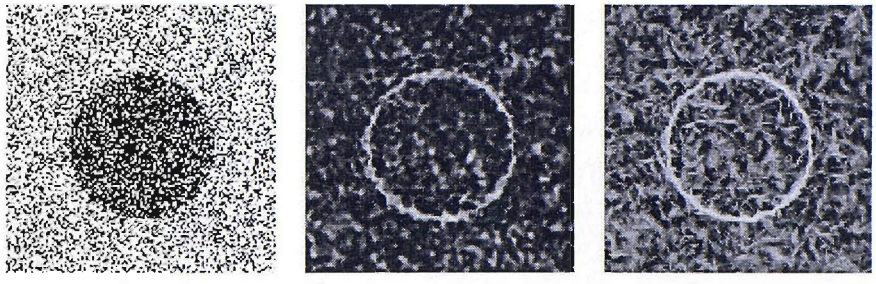
FIGURE 3.13: Results from testing a simple white disc moving on a black background with increasing amounts of salt and pepper noise. Crossed magenta or blue lines represent true features detected, circled red or green lines represent false features detected. Error bars show the range of results due to the test being run fifty times.

phase congruency. Both techniques detect features about a number of orientations, and they are displayed to show the differences in contrast that the methods for combining the results to a single image have. In figure 3.14(b) the circle is detected, but the edges are somewhat erratic in strength. The noise however seems to have far less significance than compared to figure 3.14(c) where the boundary of the circle is consistent, but the noise has a greater significance. The summed variant is also much more variable, and is theoretically bounded between 0 and L where L is the number of orientations, where as the maximum value of each orientation is fixed at 1 so figure 3.14(c) is bounded between 0 and 1. More research could be done into methods for robustly combining multiple orientations into a single feature map, or higher level techniques could be implemented to take advantage of the separation of features by orientation.

3.3.3 Occlusion

Within many environments objects moving in a scene become occluded, thus it is important to test the response of the technique against occlusion. The test sequence again was a moving circle and the occlusion was rectangular in shape, causing part of the moving circle to disappear.

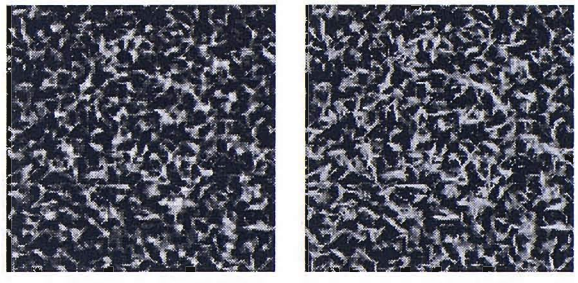
The expected behaviour of the technique would be that the occluded region would have some degree of blurring of feature information from the moving circle echoed from the



(a) Central frame from the input sequence

(b) Temporal phase congruency results (orientations summed)

(c) Temporal phase congruency results (max. of orientations)



(d) Image based phase congruency results for central frame (orientations summed)

(e) Image based phase congruency results for central frame (max. of orientations)

FIGURE 3.14: Results from a sequence of a moving circle with zero mean Gaussian additive noise (sigma = 90% of difference between black and white). Temporal phase congruency used a 32 frame sequence.

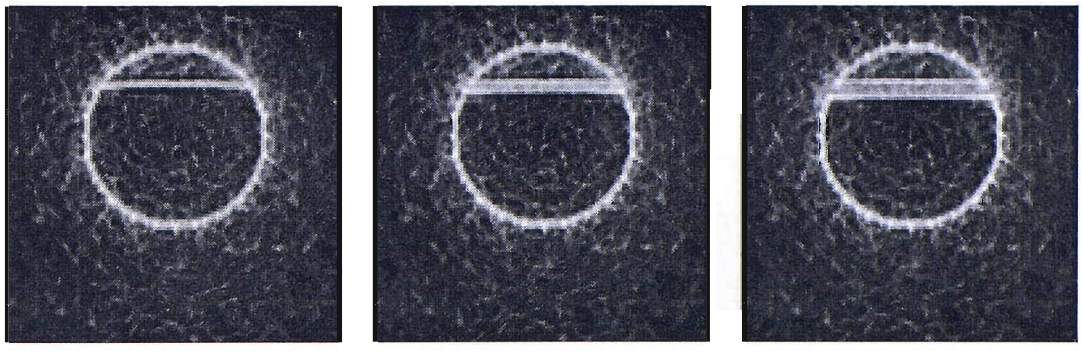
surrounding time and space neighbourhood to where the circle ought to be without the occlusion there. The results though show much less blurring than expected, and are very promising.

In figure 3.15 the circle is very clearly detected in all three cases. The circle has also maintained a uniform shape, with the lines of occlusion being shown in the middle of each circle. They all though show some amount of blurring between the edges of the occlusion, but this sort of blurring can also be seen around the edge of the circle and should be removable via non-maximal suppression.

Broader occlusions are shown in figure 3.16. These are much clearer as the distance between features is greater.

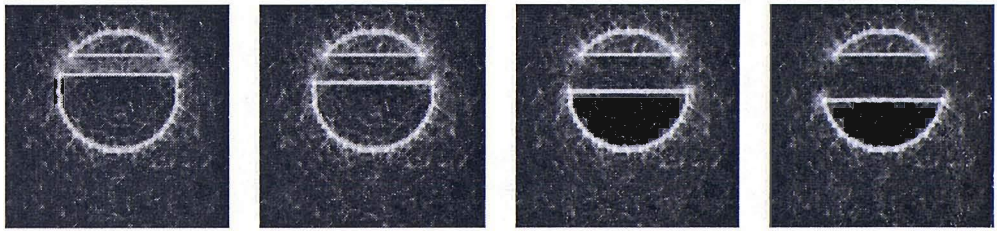
3.3.4 Feature Velocity Testing

Since temporal phase congruency is a time persistent feature detector, it can detect a feature's orientation through space-time. It is important therefore to present examples



(a) 1 pixel wide (b) 3 pixels wide (c) 5 pixels wide

FIGURE 3.15: Results from the temporal phase congruency operator on sequences with a fixed position bar occluding the image sequence.

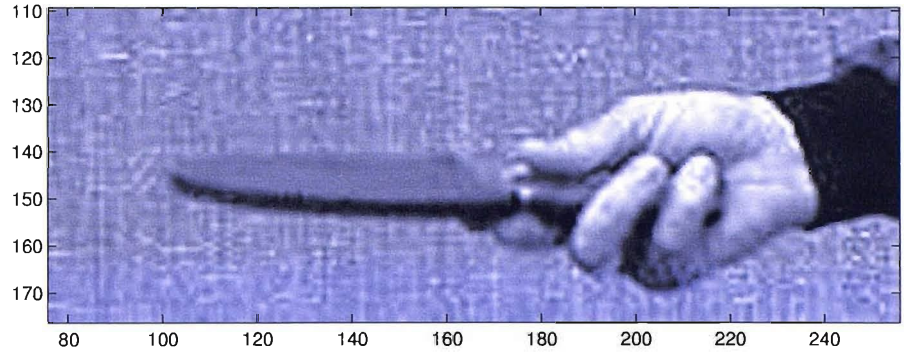


(a) 10 pixels wide (b) 15 pixels wide (c) 20 pixels wide (d) 25 pixels wide

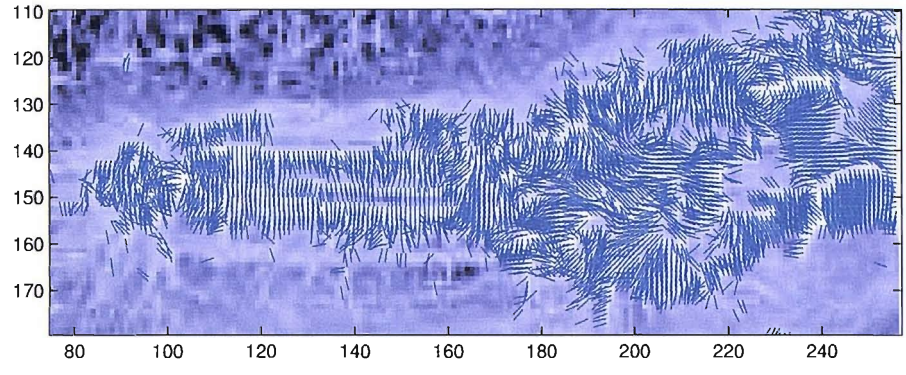
FIGURE 3.16: Results from the temporal phase congruency operator on sequences with a fixed position bar occluding the image sequence.

results and test the accuracy of the orientation information of any detected features.

Figure 3.17 shows an example of the normal orientation data that can be extracted from temporal phase congruency. Presentation and evaluation of this data is difficult and so a quantitative approach testing against a simple test sequence of a shrinking and expanding circle in 2D+T space was conducted. (This could also be considered a sphere in a 3D space and was constructed as such.) The test sequence had zero mean Gaussian noise (with a sigma of 10% of the difference between the circle and the background) added. This was to avoid the anomalies discussed in section 3.2.4. The test sequence was then passed through the temporal phase congruency feature detector, and then the orientation of all the significant feature points was established using the estimation method in section 3.2.4. The orientations are compared to values calculated when generating the test data and distances from this 'ground truth' is calculated for all features over .7 in strength. The results are then presented and compared to similar results for a 3D Sobel operator[8] that has been applied to the sequence with additive noise.



(a) Original image



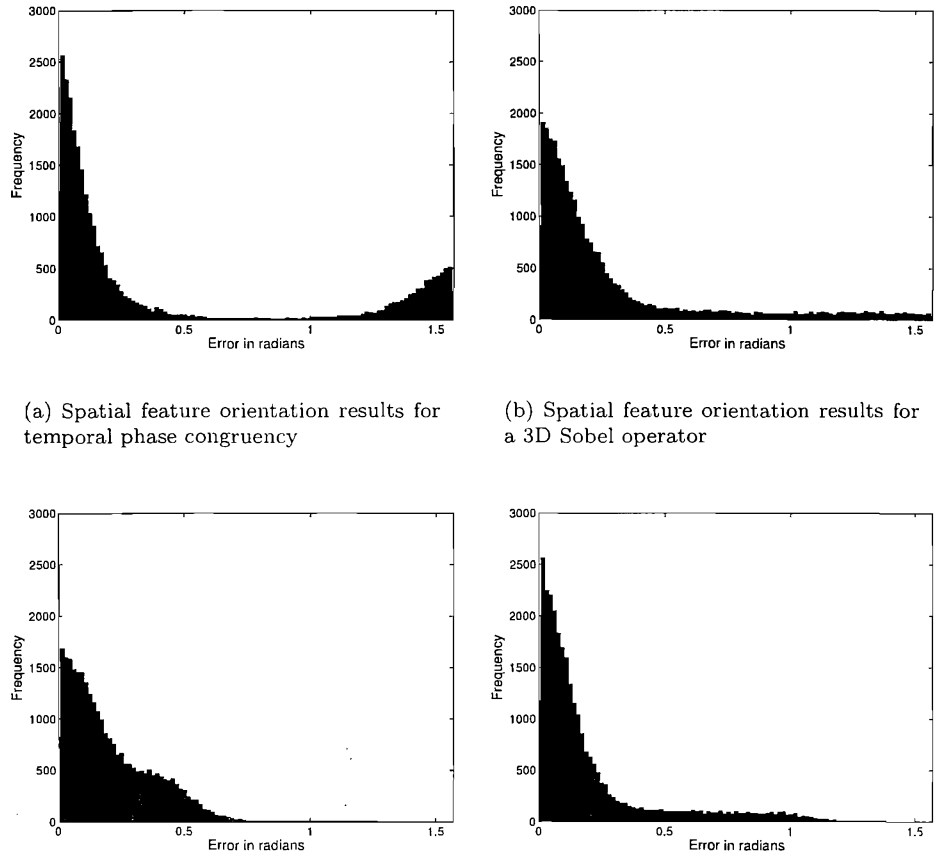
(b) Example results with orientation vectors overlaid

FIGURE 3.17: Example results of orientation information for features extracted from temporal phase congruency of a hand and table tennis racket.

From the separate histograms in figure 3.18 temporal phase congruency is able to detect feature orientation to a similar degree that a Sobel operator may. We do however see that the two operators fail in different ways. Temporal phase congruency, in figure 3.18(c), seems to have two narrow distributions centred correctly about zero radians error and incorrectly about $\pi/2$ radians error. The estimates generated by the Sobel operator however do not have this second erroneous peak and most of the estimates have less than .3 radians error, as shown in figure 3.18(d). Similarly about the temporal axis both detectors have different behaviours. Temporal phase congruency has a wider spread of absolute errors in figure 3.18(a) which is not so tightly distributed as Sobel in figure 3.18(b).

3.3.5 Velocity Thresholding

Since the new technique extracts the velocity of a feature, it is possible to threshold based upon that velocity. This could be useful for selecting features of interest based



(a) Spatial feature orientation results for temporal phase congruency

(b) Spatial feature orientation results for a 3D Sobel operator

(c) Temporal feature orientation results for temporal phase congruency

(d) Temporal feature orientation results for a 3D Sobel operator

FIGURE 3.18: These plots show the ability of the temporal phase congruency operator to detect the orientation of a wide variety of feature orientations

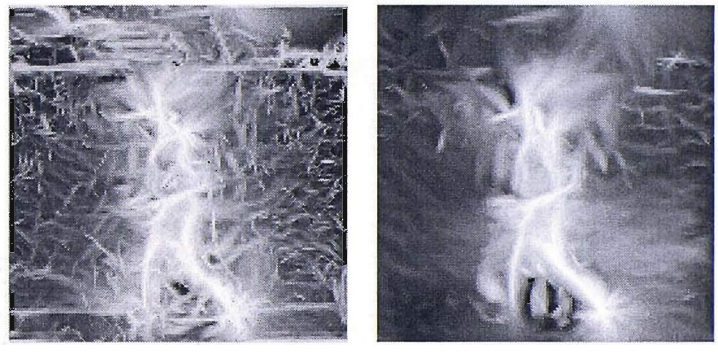
upon their velocity in an image sequence, such as detecting objects only moving in a particular direction, or estimating the background feature map. This can be a post-processing stage, or the technique can be altered so that it only looks for features at a particular velocity. Altering the technique should reduce the time taken to process a sequence, but the additional orientations at 'unwanted' velocities could be used in noise detections as features are detected in orientations that are similar to their true orientation, whereas typically noise is not. Example frames from a sequence of a person walking are shown in figure 3.19.

In figure 3.19 the results show a strong progression from detecting the stationary background features such as the ceiling tiles and the floor, to progressively detecting more of the moving features in the scene. It is interesting to note that the slower moving orientations are detecting high phase congruency at the boundaries of the legs, and as the algorithm focuses upon increased speed the highest phase congruency is found in the centre of the legs giving an almost skeletal feature detection. This 'skeletal' detection is

(a) all velocities

(b) near zero velocities selected

(c) slow velocities selected



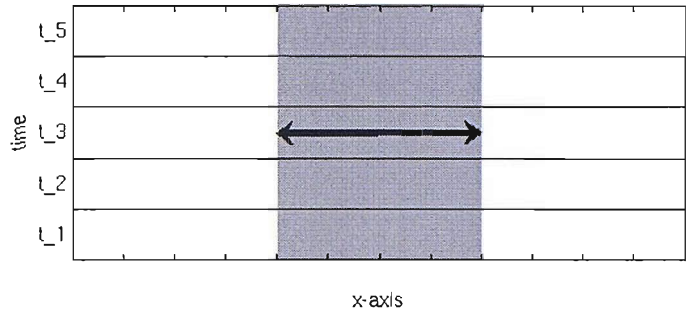
(d) medium velocities selected

(e) fast velocities selected

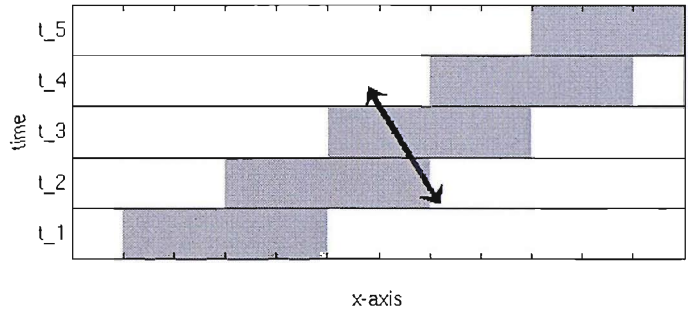
FIGURE 3.19: Results from the temporal phase congruency operator on sequences with ranges of velocities selected.

explained by looking at a simplified 1D+T example.

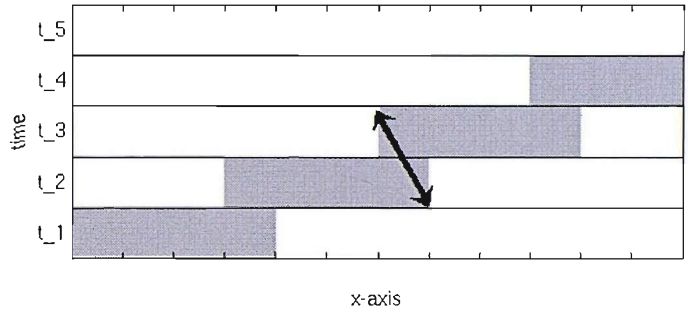
Figure 3.20 shows three examples of a grey object on a white background in 1D+T. If we examine the width of the objects we see in figure 3.20(a) the object is stationary and its minimum width is the spatial width of the object. In figure 3.20(b) we see that even slow movement can reduce the minimum width of the object in space-time. In this figure if the object is viewed at an angle the width of the object is 'reduced'. Finally in figure 3.20(c) the width of the object becomes nearly 1, thus any 1D+T feature detector oriented at an appropriate angle would detect an impulse and not a rising then failing edge of a pulse. This reduction to an impulse changes the number of features this object has from 2 to 1, an operation that the legs of the person in figure 3.19 also undergo. The detection of such an object like this should provide important information, as it firstly implies that there is an object of a determinable width centred at the peak feature point. The bounds of the object should also be linked to the velocity of the feature detected, as the transition from a pulse to an impulse is directly related to the objects width and velocity.



(a) A stationary 1D object over time



(b) A slow moving 1D object over time



(c) A fast moving 1D object over time

FIGURE 3.20: These three figures illustrate that an object's width in space-time, becomes smaller when moving.This illustrates why the legs in figure 3.19 are detected as a single feature.

Re-examining figure 3.19, it is also interesting to trace the front foot through the different velocities as it is clearly marked in figure 3.19(b), but in figures 3.19(d) and 3.19(e) it is not present. The most noticeable aspects of these images though are the multiple edges of the back and back leg found in figure 3.19(b). This shows that the feature detector is possibly suffering the effects of aliasing within the image sequence as the signal is insufficiently sampled to capture smooth motion between frames. Essentially what is being exposed is the response of the feature detector to features present in the surrounding frames. These results do suggest that the implementation allows for a

concept of velocity scale which emphasises differences from image base phase congruency and optical flow: it is possible to target features according to proportionate movement which is not possible with these techniques, except by analysis of results.

3.3.6 Problem results

This technique has shown an increased measure of robustness when compared to Kovesi's[16] image based technique. It is able to extract a feature's spatio-temporal angle giving new and important information for further stages in any computer vision system. However, during the testing of this new technique a number of images generated problematic results.

Initial results show that faster movement can cause 'echoes' due to temporal aliasing, but further investigation shows an interesting and very undesirable response. To understand the cause of these 'echoes' the input data was simplified to a time varying 1D signal (1D+T). Figure 3.21 shows a 1D+T signal where the signal is initially uniform, then an impulse 'moves' along the signal at 4 pixels per sample. If viewed using a spatial framework, it could be thought of as a 'textured' line at a slight angle or an angled impulse train. In a more localised framework, a viewer may only see single impulses and no significant 2D feature.

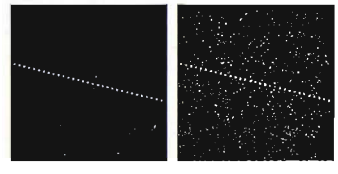
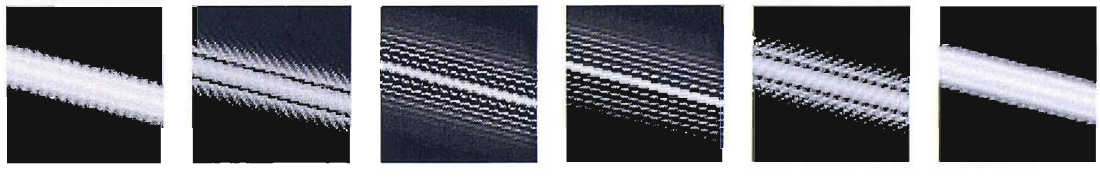


FIGURE 3.21: This image can be thought of as a 1D+T signal representing a moving impulse, or an image with a 'loosely connected line'.

The responses to this input image from the 2D feature detector developed by Kovesi are presented on a per orientation basis. Here we use 6 orientations with 30 degrees between each.

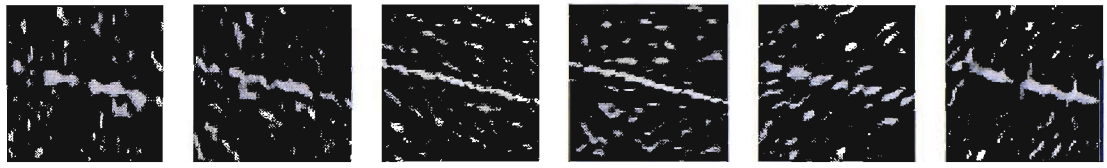


(a) 0° (b) 30° (c) 60° (d) 90° (e) 120° (f) 150°

FIGURE 3.22: These show the results from applying phase congruency upon the 1D+T moving dot signal in figure 3.3.6

Since we became aware of temporal phase congruency's problems with synthetically 'perfect' signals in section 3.2.4. We repeated this test with a similar measure of additive Gaussian noise. These results are shown in the following figure 3.23.

These results now show a more acceptable 'line' across the signal. There are many noisy 'features' detected, but this is to be expected in this noisy signal. Adding noise seems



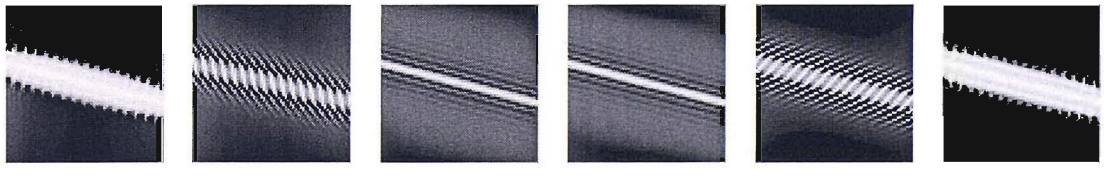
(a) 0° (b) 30° (c) 60° (d) 90° (e) 120° (f) 150°

FIGURE 3.23: These show the results from applying phase congruency upon the 1D+T moving dot signal from figure 3.3.6.

to reduce or remove these errors. Further tests upon a similarly 'moving' step feature display problems with both the clean and the noisy image. The source images are shown in figure 3.24, with the results from phase congruency in figure 3.25 for the clean signal and in figure 3.26 for the noisy signal..

(a) A moving step (b) A moving step with noise

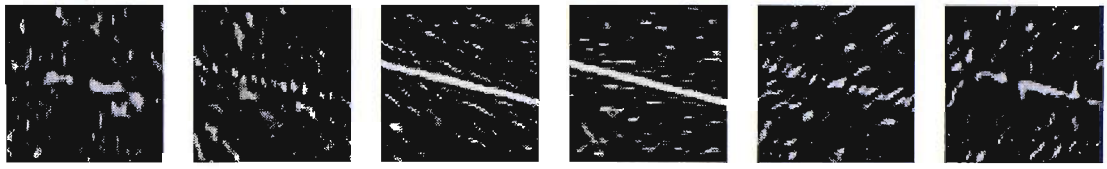
FIGURE 3.24: These two figures of a 1D+T moving step function are used to highlight some of the problems phase congruency has. The noise added to figure 3.24(b) is 10% additive Gaussian noise.



(a) 0° (b) 30° (c) 60° (d) 90° (e) 120° (f) 150°

FIGURE 3.25: These figures show the results from applying phase congruency upon the 1D+T moving step signal from figure 3.24(a).

Both sets of results in figure 3.25 and figure 3.26 show problem results. Results in figure 3.25 show a strong central line detected, which is acceptable, but there are lines running parallel that would remain even after non-maximal suppression. The same lines are still present in figure 3.26 even if they are less consistent. From the results, it appears prudent to consider an alternative method of measuring phase congruency.



(a) 0° (b) 30° (c) 60° (d) 90° (e) 120° (f) 150°

FIGURE 3.26: These figures show the results from applying phase congruency upon the 1D+T moving step signal from figure 3.24(b).

3.4 Conclusions

This chapter has successfully shown a new extension to phase congruency to allow the processing of time persistent features. This allows for computer vision algorithms that deal with image sequences to take advantage of an image's presence in a sequence and not just as an isolated signal.

The new technique shows good improvements in the robustness of the operator within noisy environments. This should allow progression of computer vision in more difficult imaging environments where noise is common and features persist in time.

Temporal phase congruency also provides new and useful spatio-temporal orientation information that should assist in providing a richer set of feature properties to higher level computer vision algorithms. For example, this spatio-temporal orientation information could be used to supplement optical flow results.

There has also been some useful analysis upon the limits of the technique which has provided motivation for further development in the measurement of phase congruency within signals. This is addressed in the following chapters which overcome these limitations.

Chapter 4

Statistical Phase Congruency

From the previous chapter we have seen that although phase congruency can be extended to operate in the spatio-temporal domain there are a number of inputs that can cause multiple false high points of phase congruency to be detected in images and image sequences. This chapter looks at a new technique to measure phase congruency that avoids this behaviour, but is developed within this chapter to operate on single images and extended in chapter 5 to operate on images sequences.

The chapter begins in section 4.1 by modelling phase congruency using a localised model of filtering. Using this model a new method for the calculation of phase congruency is developed that uses families of filters in section 4.2. The development of these filter families is continued in section 4.3 where they are evaluated in terms of their ability to measure phase congruency and also the spatial extent. Section 4.4 adds the use of the mean energy to prevent phase congruency from being detected in parts of an image that give filter responses that are dominated by noise. This final step of processing allows for the new feature detector to be tested and results produced. This is the subject of section 4.6 where example images and their statistical phase congruency feature detected results are presented. This section also examines the technique's response to noise in section 4.6.2, whilst the final section compares the Kovesi's method for calculating phase congruency with statistical phase congruency on the images that produced erroneous results. This shows a strong improvement for statistical phase congruency over Kovesi's method and points to a more appropriate feature detector for extending into 2D+T. Further conclusions are presented in section 4.7.

4.1 Modelling localised phase congruency

Previously Gabor filters have been used to measure the phase of a signal for a particular frequency band. This phase information can be thought of as the mean phase for all the

frequencies contained within the selected band. Up until this point phase congruency has been an inter-band measure and does not allow the evaluation of the degree of phase congruency within the bands themselves. Analysing the intra-band phase congruency could remove the erroneous results found in section 3.3.6 because although the inter-band phase is congruent, the intra-band phase is expected to contain incongruencies that will be detected and reduce or remove the erroneous results.

This chapter therefore develops a technique that can measure the structure of these frequency bands, without attempting to measure more finely the frequencies, or the localisation of those frequencies. To achieve this measurement of the structure between the different frequencies we first model the process of (Gabor) filtering a 1D signal, as in eqn. 4.1.

$$\begin{aligned} G_c &= \sum_{n=1}^{m} a_n \alpha_n \cos(\theta_n) \\ G_s &= \sum_{n=1}^{m} b_n \alpha_n \sin(\theta_n) \end{aligned} \tag{4.1}$$

where a_n and b_n are the even and odd filter coefficients, α_n is the nth frequency's amplitude of m frequencies, and θ_n is the nth frequency's phase. G_c and G_s are the cosine and sine Gabor filter responses respectively. In equation 4.1 the frequencies are local frequencies, that are intractable due to the uncertainty principle, i.e. they are localised and separated. The coefficients are also intractable, but are intuitively assumed to be similar to those of the Gabor filter's Fourier domain representation. In other words, although the Fourier transform of a Gabor filter is Gaussian shaped, this is just a representation of the filter in a global sense. Locally it is impossible to know the exact coefficients of the filter. Similarly when we model any data, (α_n, θ_n), it is local frequency data and not a Fourier transform of a whole signal, but rather the local spectrum. Without this, detecting phase congruency of Fourier components is possible from a Fourier transform, and would produce a global measure of phase congruency without localisation.

Consider that if two frequencies in the sum are π radians out of phase then they will combine destructively. Unless all the frequencies are perfectly in phase, the sum of the absolute energy of the local frequencies will always be greater than the sum of the energy of the frequencies. Mathematically, to solve equation 4.1 using linear equations to find all the θ_n's, we would need to repeat the filtering process many times with different coefficients. This is impractical and would also resolve the amplitude of each individual localised frequency, information we are not interested in. Instead, we note the relationship between the phase congruency of the frequencies and the sum of energy. It is known that phase congruent frequencies, when summed, give greater responses than

random phased frequencies that will destructively superpose.

4.2 Measuring phase congruency

To provide the new measure of phase congruency, we consider adding different phase shifts to the Gabor filter's coefficients. This will affect the sum of energy from the filters due to varying measures of constructive and destructive summation of local energy. When comparing between a normal Gabor filter and a phase shifted Gabor filter, perfectly congruent frequencies would be expected to reduce in the energy of response, and random phase frequencies remain of approximately the same energy. Such a filtering step is illustrated in eqn. 4.2.

$$\begin{aligned} G_c &= \sum_{n=1}^{m} a_n \cos(\psi_n)\alpha_n \cos(\theta_n) \\ G_s &= \sum_{n=1}^{m} b_n \sin(\psi_n)\alpha_n \sin(\theta_n) \end{aligned} \tag{4.2}$$

where ψ_n represent the different phase angles of the Gabor filter's coefficients. (The values of these differing angles is discussed later.) Since the exact distribution of energy amongst the frequencies will be unknown, a single phase shift may not significantly affect the response to the filters. It is necessary then to repeat the filtering L times, forcing different amounts of phase shift to occur between different frequencies. In the repetition it becomes apparent that the variation of the phase shifted filters should describe the phase congruency sufficiently without the linear phase Gabor filter, which could be thought of as just one of the phase 'shifted' filters.

Since the energy of the frequencies will be mixed between the cosine and sine-based Gabor filter responses, we use the combined energy of the filter pairs to extract a measure of the effect of the phase shifting, as in equation 4.3.

$$E_l = \sqrt{G_{c_l}^2 + G_{s_l}^2} \tag{4.3}$$

where G_{c_l} and G_{s_l} are the lth cosine and sine based phase shifted Gabor filter responses. Having calculated the energy for each filter pair we normalise by dividing by the mean energy, $\overline{E}$. This is shown in equation 4.4.

$$\hat{E}_l = \frac{E_l}{\overline{E}} \tag{4.4}$$

where $\hat{E}_l$ is the normalised measure of energy the lth filter pair generated. The normalisation of the filter energies is necessary as localised frequencies with the same degree of phase (in)congruency will have a standard deviation that is dependent upon the energy of the localised frequencies. It is however desirable that we only measure phase congruency, and not the measure of local energy. This normalisation of E_l therefore removes the energy dependency, but retains the variation between the different filter results. We then calculate the standard deviation of each of the energies after normalisation by the mean energy giving ρ_E.

$$\rho_E = \frac{1}{L}\sum_{l=1}^{L}(\hat{E}_l - 1)^2 \tag{4.5}$$

where $\hat{E}_l$ is the normalised measure of energy for the lth filter pair of L filter pairs, and 1 is the 'mean' due to E_l being normalised.

To test whether ρ_E describes the phase congruency of frequencies we use the model of filtering. We use sets of complex numbers as our frequency data. These sets will allow us to test whether a series of phase shifts can be used to detect phase congruency through the measure ρ_E, and also whether our model is valid. Each set of synthetic frequencies has random amplitude and mean phase, with increasing measures of zero mean Gaussian noise perturbing the phase of the frequencies. If ρ_E is to be a measure of phase congruency it should be correlated to the change in noise levels, or to the reduction in phase congruency.

Before testing, the phase shifts applied to the Gabor filters need to be defined. If we look at randomly shifting each frequency element in the Gabor filter, the spatial extent of the filter would extend significantly possibly to the entire signal width. Therefore the phase shifts applied to the Gabor filter should affect the spatial extent as little as possible. There should also be different amounts of shift applied between frequencies. Initially we explored the use of linear phase shifts ($\psi_n = 2\pi n/m$) which shifted the filters in the time domain, but it was important to investigate other functions to see if any improvement could be made on this set of phase shifts.

4.3 Improving the phase shift functions

Although the concept for measuring levels of phase congruency amongst the separate frequencies has been discussed the appropriate functions and parameters for those functions needed to be evaluated. Section 4.2 also assumed that a Gabor filter might be the appropriate choice for the basis filter. There are in fact at least 3 filter types that have been useful in phase analysis in this work. These are a standard Gabor filter, a log-Gabor filter and a log-Gabor filter convolved with a low pass filter. These will affect

the shape in the frequency domain of our filters. The log-Gabor filters can also be varied in bandwidth. The percentage bandwidth of a Gabor filter however cannot exceed 80% of an octave before the DC component starts to becomes significant. It is desirable to have a low level of DC sensitivity in any filter used so that any following technique will not be unduly affected by an image's lighting variation. The Gabor filter is thus limited to such a bandwidth during these tests.

Additionally the central frequency of each filter can be altered. This will be important when considering the scale of a feature and also when considerations of spatial extent are examined. Smaller central wavelengths will be expected to produce smaller filters spatially.

Once a basis function is chosen there are a number of possibly conflicting requirements for the 'optimal' solution to our problem of measuring phase congruency in this way. The possible requirements could be listed in such a way:

Any filter (families) used should have:

fixed gain A *fixed gain* for each frequency component in each of the different filters is important so that they can be compared.

phase angle invariance Phase congruency can occur at any phase angle therefore the filter families need to be invariant to the actual phase of a feature.

correlation between congruency and ρ_E It is the *correlation* that enables the *detection of phase congruency* whether it is a negative or positive correlation.

limited spatial extent To detect local features and to avoid features affecting one another a *limited spatial extent* is important.

4.3.1 Using sine based phase shifts

The approach taken was to use a family of sinusoids. The different attributes of the sine waves were changed to attempt to find a better set of functions to use as phase shifts. These functions apply different phase shifts to the individual frequency components. We can define the lth phase shift function, ps_{l_1}, in a family of L filters as in equation 4.6.

$$ps_{l_1}(\omega) = \cos^{m_l}(\omega n + \lambda_l) + i.\sin^{m_l}(\omega n + \lambda_l) \tag{4.6}$$

where m_l alters the shape of the filter without affecting the periodicity of the phase shift, ω alters the number of oscillations within the phase shift and λ_l alters the starting phase of the phase shifted filters. m was varied between .25 and 4. This equation allows

for sets of L filters to be tested. An alternative set of sine-waves was also tested, these used the absolute value of the cosine and sine functions, as in equation 4.7.

$$ps_{l_2}(\omega) = |\cos^{m_l}(\omega n + \lambda_l)| + i.|\sin^{m_l}(\omega n + \lambda_l)| \tag{4.7}$$

The phase shifts, λ_n, were defined in two ways. The first provided linear phase shifts between different filters, as shown in equation 4.8. The second provided an increasing set of phase shifts between filters as in equation 4.9.

$$\lambda_{l_1} = rng * l/L \tag{4.8}$$
$$\lambda_{l_2} = rng * l^2/L^2 \tag{4.9}$$

where rng is the range the phase shifts are varied over. In all the following testing rng was set to either π or 2π.

A search of the parameter space was then conducted which allowed for variation in all the described attributes, with the aim of determining how each variable could affect each of our design criteria. (It is acknowledged at this stage that any number of function families could be chosen instead of the sinusoids.)

Initial testing had shown that functions that behaved better in the model also behaved better on real signals. Using the model allowed for the mean phase angle for phase congruency detection to be altered easily, as well as allowing for the extent of congruency to be altered in a measured manner. It is at this point the concept of phase congruency must be scrutinised. The underlying question that must be asked is: Does a useful measure of phase congruency include dependence upon energy? For example, consider a frequency spectrum consisting of 5 frequencies where 3 are congruent and very low amplitude, and the fourth and fifth frequency are orders of magnitude greater and one of them is congruent with the 3 very low amplitude frequencies whilst the other is π radians out of phase. An energy dependent model would state that the frequencies were not congruent, as the sum of the frequency components is significantly less than the sum of the magnitudes, but an energy independent measure would state that there is phase congruency of the order of 60-80% since 4 frequencies are congruent and a fifth is not. A more meaningful measure may be an energy dependent model, particularly for feature detection. Kovesi also resolved to measure phase congruency in this energy dependent manner.

Having established the type of phase congruency we wished to measure, applying particular degrees of phase congruency would have been unnecessarily costly. Therefore we apply a constrained, but random amount of noise to the localised frequencies, and then

measure the actual phase congruency that this new spectrum possesses. This allows the response from the technique and the various functions under test to be reviewed against a wide and varying range of phase congruency.

During testing 10000 such randomised spectrum were produced to test each filter family. The tests produce a distribution of results. In figure 4.1 we see an example of one of the sets of results.

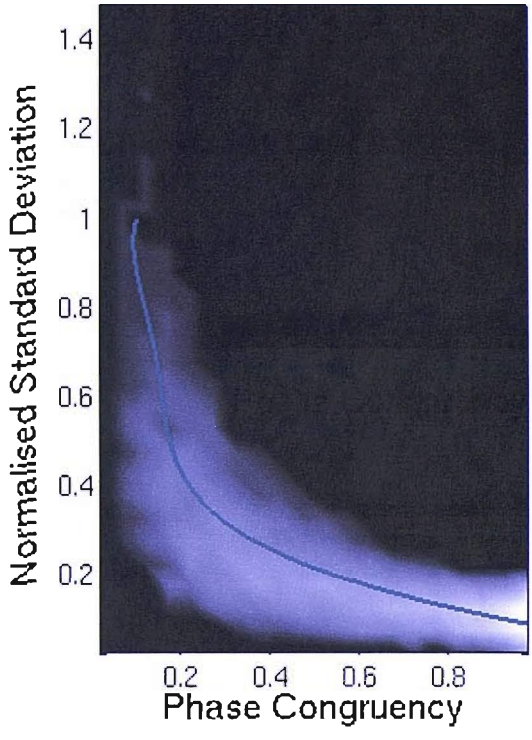


FIGURE 4.1: An example set of 10,000 measurements of phase congruency versus the measured normalised standard deviation, ρ_E from one of the sine functions. A line of best fit is also overlaid upon the distribution.

As can be seen from figure 4.1 this particular function does display a correlation between phase congruency and the normalised standard deviation with smaller normalised standard deviations correlating to higher degrees of phase congruency and vice versa. This result is somewhat unexpected, as it had been previously reasoned that more phase congruent signals would have a higher variance in response to different phase shifted filters. It does not show a unique mapping between phase congruency and normalised standard deviation, but more of a distribution of results that changes for different measures of phase congruency. To use the normalised standard deviation we fit a line to the middle of the distribution that allows the mapping from the normalised standard deviation to the phase congruency. When comparing the different sets of functions the key measurement is the spread of the distributions from the line used in the mapping. To measure the spread of the data we chose the normalised residual generated when fitting a polynomial to each set of data.

We note that the distribution becomes significantly wider from about 0.3 phase congruency and the spread is significantly tighter about 1.0 phase congruency. To gain an increased insight into how the error from the line of fit varies as we decrease the degree of phase congruency, we apply the fitting process to different ranges of the data always starting from a phase congruency of 1. The results from this analysis are plotted in figure 4.2. Although they do not show the individual results clearly the figure does show the variation and the general shape of the normalised residual error against the ranges of phase congruency.

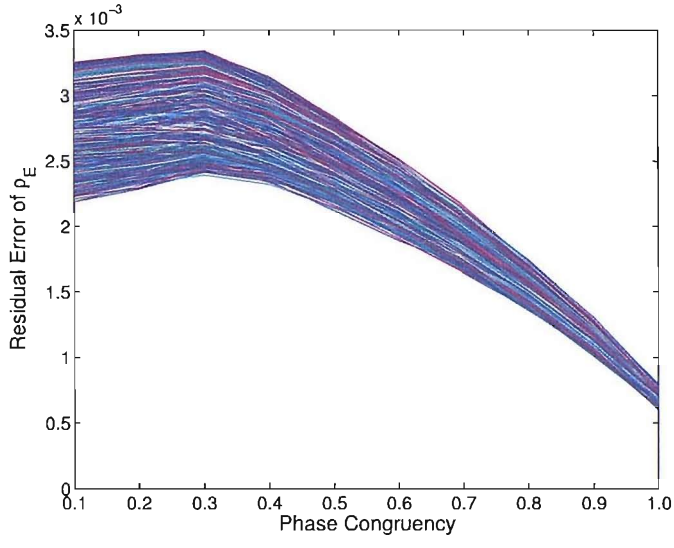


FIGURE 4.2: The different residual errors as calculated from different ranges of the test data for all the test functions used.

From this plot we chose to take the 0.3 phase congruency values as the point at which to differentiate between function performance. These points are firstly the maximum in figure 4.2 for many of the functions, but also signify reasonable points at which any phase congruency results could be thresholded. It was also important that any function chosen should operate well over the range between .3 and 1.0. This would enable a broader range of feature strengths to be extracted and for the resulting operator to be useful for images with different levels of phase congruent features.

Using these results we can compare the effects of the various parameters that have been altered in our phase shift function. This though only represents the ability of our filters to distinguish phase congruency, but it is also important that our filters have a limited spatial extent. This is the focus of the following sections.

4.3.2 Filter construction in the Fourier domain

In the previous sections we have modelled and developed filters that measure local phase congruency. These filters have been modelled in the local frequency domain and

have been optimised to measure the local frequency's degree of phase congruency. It is however impossible to construct a filter in the local frequency domain as frequency separation and localisation are inversely related. We have therefore considered that if a Gabor filter is transformed into the Fourier domain and the phase shifts applied to the filter in the same manner as has been applied to the local frequency domain then we assume that the filters will measure global phase congruency. If, however, the filters have a spatial extent that is just a small percentage of the real domain when inverse Fourier transformed then the phase congruency calculations will be considered to be calculated locally. It is with this perspective that we must therefore measure spatial extent. It is also important that we formalise how the filters are constructed.

To develop and test the different phase shifting filters we have stated that the local coefficients of a Gabor filter are the similar to the coefficients of the Gabor filter's Fourier domain coefficients. This allowed us to test the ability of different phase shifting functions to measure local phase congruency. We now define the filters in terms that are realisable and so we reverse the notion and use the local filter coefficients to describe the Fourier domain coefficients of our filters. At this stage the filters will have a measurable spatial extent that will begin to define whether they are measuring local frequency information. The filters are thus defined as in equation 4.10 using polar coordinates.

$$\begin{aligned} spc2d_n &= G(\omega, \omega_n)\ Gaussian(\theta, \theta_n)\ ps_n(\omega) \\ Gaussian(\theta, \theta_n) &= e^{\frac{-(\theta-\theta_n)^2}{2\sigma_\theta^2}} \end{aligned} \tag{4.10}$$

where $spc2d_n$ is the nth filter in a family. G is the base Gabor filter and could be a log-Gabor filter or a Gabor filter, either of which will act along the ω-axis and have their central frequency at ω_n. *Gaussian* is the spreading function that we use about the θ-axis with the focal angle of the filter specified by θ_n (σ_θ is usually less than the angular distance between filters). ps_n is the phase shift function that also acts along the ω-axis and it is results between filters with the same first two elements that are used to generate the normalised standard deviation, ρ_E, on a per orientation and scale basis. We now consider the spatial extent of our filters.

4.3.3 Measuring the spatial extent of the filters

The spatial extent of our filters is important for a number of reasons. Firstly it is assumed that a smaller spatial extent of our filters should support better localisation of results. Secondly it allows for features to be detected closer together without interference. It is therefore important that the chosen filters be 'small'.

Measuring the spatial extent of the chosen filters could be approached analytically,

except one of the base functions that we use, the log-Gabor filter, has a singularity in the Fourier domain, so this is not possible. Kovesi proposed two methods for evaluating the spatial extent of the log-Gabor filter so as to understand the behaviour with varying bandwidths. The first method was to measure the number of terms needed to represent a percentage of the filter's total area. The second method measured the distance from the centre of the filter to the term that was a small percentage of the peak value in that filter. We re-use these methods to measure the spatial extent of our phase-shifted filters. These we normalise against standard Gabor filters so as to provide a guide rather than a series of magical numbers. This leaves our measures of spatial extents in percentages of the normal Gabor or log-Gabor filter widths as appropriate.

So as to provide further information we considered this for a range of percentages. For the area measure we evaluated the width of the filter needed to represent from 95-99% of the filter's area in 1% increments. For the peak measure we examined the smallest width that contains all values smaller than 1%-5% in 1% increments of the greatest value in the filter. Since these filters are complex this operation is done on the magnitude of the filters, not their real and imaginary parts. In refining the choice of spatial extent measure it was argued that if our filters were cut off at the extent described then it would be important not to introduce too great an error. With this in mind an error of $\pm 1\%$ was acceptible and so the area measure containing 99% of a filter's area was chosen.

4.3.4 Interpreting the measures of spatial extent

Having established a means to evaluate an individual filter's spatial extent, it is now important to consider that all the filters are not generated to produce isolated results. More precisely the statistical phase congruency measure uses the relationship *between* filters, therefore the spatial extent of a set of filters is the actual subject of interest.

It could be argued that the spatial extent of such a family of filters is the greatest spatial extent of the individual filters, since there will be a change in response from the technique as the larger filter becomes affected by a 'nearby' feature. This effect though is balanced by the other filters in a family, and point towards a more important aspect being that of the smallest filter. Although the larger filters will hopefully align about a significant feature, a smaller filter may only do so when it is covering the feature of interest. This could be thought of as a coarse-to-fine view of how such a family of filters may work.

Essentially the desired mathematics would be an analytical function that describes the effects of a single term in one filter upon the resulting response from the overall technique. It may then be possible to design functions that can balance this analytical basis of spatial extent against a similar analytical basis of phase congruency measurement. It is

suspected though that these two analyses will produce a tension that may not be solved or are inversely proportional similar to Gabor's understanding of frequency analysis and temporal localisation, but it is hoped that since the degree of frequency domain information that is being 'extracted' or measured is less than a Fourier or even Gabor like transform, that the degree of spatial localisation should be increased.

In lieu of this, the use of mean spatial extents of a filter family has been used to analyse the filters. An alternative might be the measurement of the spatial extent of a filter constructed from the sum of the filters, but even then since the relationship between filter responses is non-linear through the use of standard deviation, that may provide no better information.

4.3.5 Finding good solutions

In the previous sections measures for the ability of the filter families to distinguish phase congruency and measures of spatial extent have been established. The filter families were then tested to establish their ability to distinguish phase congruency and their measure of mean spatial extent. From these results those families that had less than or equal to a 100% of a Gabor filter's width were selected. From this selection the five families with the lowest residual error were chosen. This process was then repeated for with different relative widths of Gabor filter ranging from 125% to 400%. The results for this process yielded only 7 filter families that are listed in table 4.1

#	Base Filter	β	ω_n	L	λ_l	rng	m_l	Eq.n used
1	Gabor	—	3	2-9	$\pi l/L$	4π	1	4.7
2	log-Gabor	0.55	8	2-9	$\pi l/L$	4π	3	4.7
3	log-Gabor	0.74	3	2-9	$\pi l/L$	4π	1	4.7
4	log-Gabor	0.41	3	2-9	$2\pi l^2/L^2$	2π	.25	4.7
5	log-Gabor + low pass	0.41	8	2-9	$2\pi l^2/L^2$	π	2	4.6
6	log-Gabor + low pass	0.41	17	2-9	$2\pi l^2/L^2$	4π	3	4.7
7	Gabor + low pass	—	3	2-9	$2\pi l^2/L^2$	π	.25	4.7

TABLE 4.1: The best 7 sets of parameters that have a limited spatial extent and a good ability to distinguish phase congruency

The results from this table reveal a number of interesting facts. It appears that the number of different filters used per family can range from 2 to 9 and the resultant distinguishability of statistical phase congruency remains the same. The different mean spatial extents generated from using different numbers of phase shifts also appears to be stable. It is also interesting to see that equation 4.7 is favoured over equation 4.6. Finally it is important to remember that these values have been selected from tests run entirely upon synthetic data.

4.4 Energy Masking

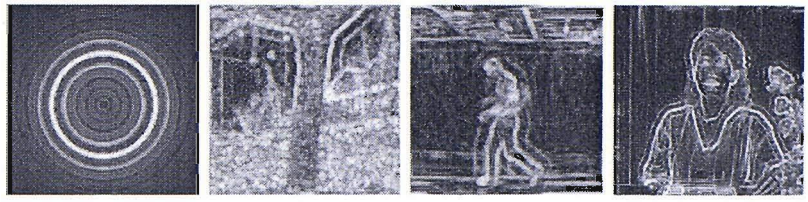
In section 4.3.5 a number of families of parameters have been selected for use in defining the phase shift function used in statistical phase congruency. This allows phase congruency to be detected in an image. Though there are parts of an image, that may have near zero energy in the selected band of the base filter. The responses from these filters and the following measure ρ_E will be therefore be based upon noise or numerical effects. To mask these points in an image the measured phase congruency is multiplied by the mean energy. This is an effective solution and enables feature detection of valid features.

4.5 Testing

It is important to evaluate the parameters selected in section 4.3.5 against real data. The results of testing are shown in figure 4.4 and show the variation between the different families. In each case the number of filters used is 2. The four different images, shown in fig. 4.3, are useful because they have different properties varying in features present and levels of illumination.

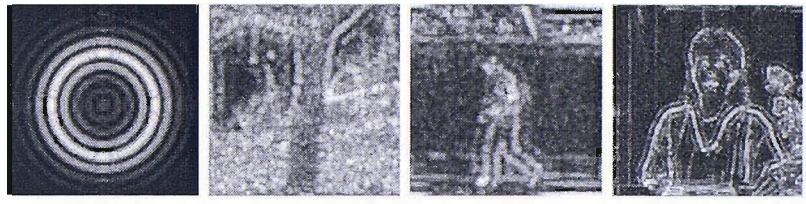
FIGURE 4.3: Four test images of a circle, a tree and some houses, a person walking, and a woman sat behind a desk.

The first column of results are from a solid black circle on a white background. This is a synthetic image useful for showing the response that each filter family has to step edge features. Since our technique relies on detecting phase congruency it is to be expected that 'ringing' occurs where an additional phantom feature is detected away from the edge due to the underlying phase of the sine-waves re-aligning, but to a lesser degree. It is also to be expected that this ringing is present in a synthetic image without noise, but should reduce or disappear in real images where other features and noise disturb local frequency structure. In figure 4.4 we see that most of the families show ringing in response to the circle, but families 1-3 and 6 also show ringing around the edge of tree, walking person and the woman sat at the desk. This leaves families 4, 5 and 7 as useful options. We then consider the problem cases of 'faster' moving features from section 3.3.6, and show the results for a 'fast' moving impulse and step with noise.

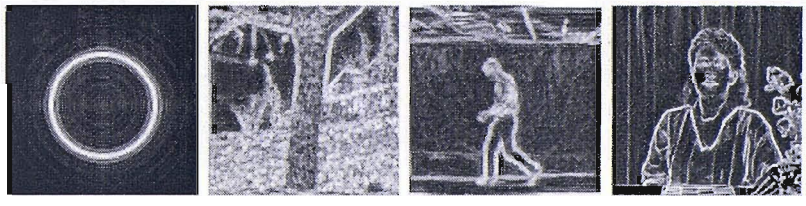


(a) using family 1

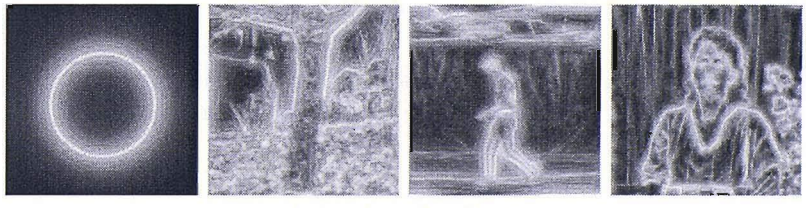
(b) using family 2



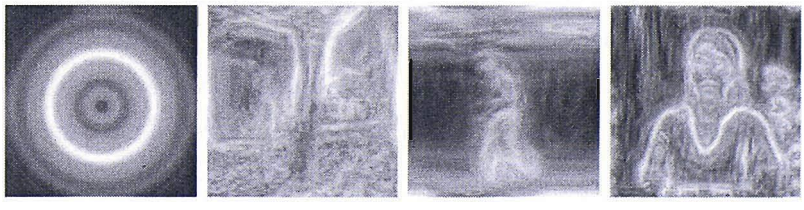
(c) using family 3



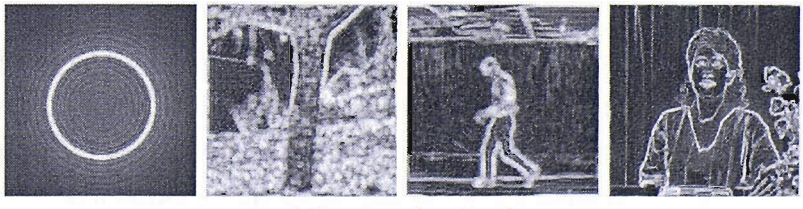
(d) using family 4



(e) using family 5



(f) using family 6



(g) using family 7

FIGURE 4.4: Results from applying statistical phase congruency to the test images of fig. 4.3

4.6 Results

This section presents the results of statistical phase congruency. Section 4.6.1 shows some example results and compares them to standard image based phase congruency. This is followed by some results of applying noise to synthetic images in section 4.6.2. Finally in section 4.6.3 the response of statistical phase congruency is evaluated against two 1D+T signals that caused phase congruency to produce erroneous response.

4.6.1 Example results

This section reviews some of the results that are generated using statistical phase congruency on images. To ease comparison we have just presented results for family 4. Families 4, 5, 6, and 7 are considered more in the formal testing in sections 4.6.2 and 4.6.3.

(a) original image of a woman behind a desk

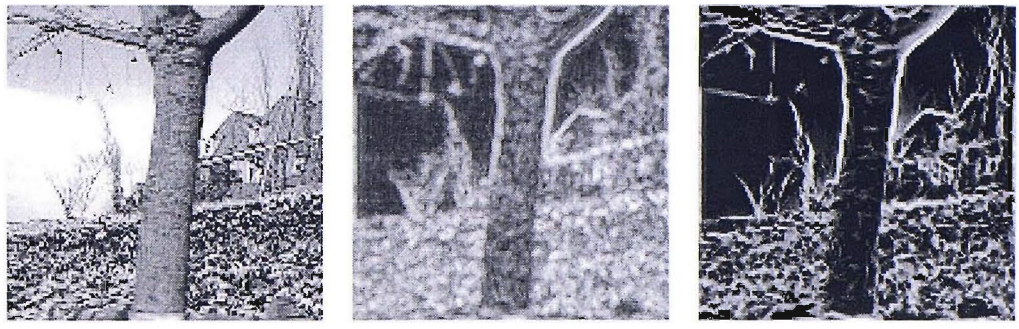
(b) results from statistical phase congruency feature detection

(c) results from phase congruency feature detection

FIGURE 4.5: A woman behind a desk and the results from feature detection using statistical phase congruency and phase congruency.

In figure 4.5 we see that statistical phase congruency is performing very well and detects the overall outline of the woman well. In addition the edges of the stripes on the woman's shirt are well detected even though the greyscale difference is small. Statistical phase congruency compares well with phase congruency in its detection of the features of the flowers to the right of the woman. In figure 4.5(b) we see the second flower down has details of the petals whereas figure 4.5(c) does not, but has generally sharper responses. Also the features detected in the background vary both giving acceptable responses when detecting the folds.

Figure 4.6 shows again a comparison between statistical phase congruency and phase congruency. This particular image contains a lot of noisy textures in the flower beds, on the tree trunk and the thin trees to the extreme right and on the left. Statistical phase congruency does well at detecting these thin trees and some of the finer branches coming



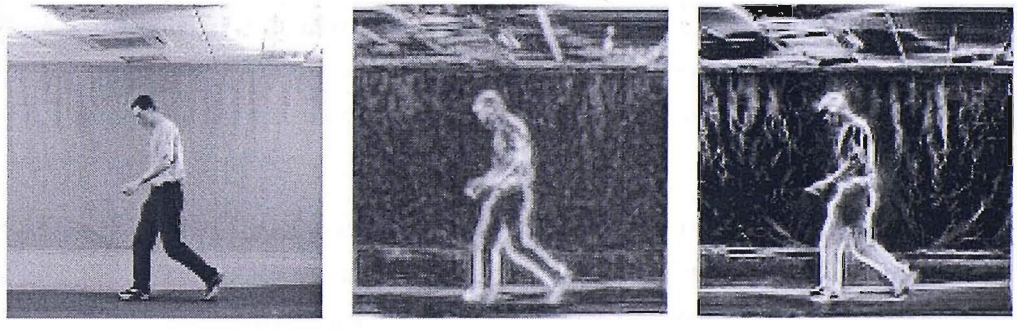
(a) original image of a tree behind a desk

(b) results from statistical phase congruency feature detection

(c) results from phase congruency feature detection

FIGURE 4.6: Results from feature detection using statistical phase congruency and phase congruency. The original image contains a tree in the foreground and houses in the background.

from the main branches on the left. In comparison to phase congruency, which does not detect these features, it should also be noted that it is again slightly more blurred or has wider feature responses.



(a) original image of a walking inside behind a desk

(b) results from statistical phase congruency feature detection

(c) results from phase congruency feature detection

FIGURE 4.7: A person walking inside and the results from feature detection using statistical phase congruency and phase congruency.

Figure 4.7 shows the strongest improvement from phase congruency to statistical phase congruency. In figure 4.7(c) we see that the centre of the leg has been detected as a feature, whereas statistical phase congruency manages to detect just the edges of the leg as expected. The ceiling also shows finer details being detected using statistical phase congruency particularly around the vent. The torso in this image proves to be the most difficult to detect feature as the grey-scale differences between foreground and background are small. Phase congruency possibly produces better results for the shoulder whereas statistical phase congruency produces better results around the hand

giving it a plausible outline. In fact, there is more of the arm's structure in statistical phase congruency compared with image based phase congruency.

4.6.2 Noise testing

In section 3.3.2 temporal phase congruency was tested using two different noise models. This section also examines the response of statistical phase congruency to noise. First zero mean additive Gaussian noise is applied to a synthetic test image, then salt and pepper noise is applied. The chosen image is that of a black circle on a white background.

Figure 4.8 shows clearly the differences between the results generated from families 4 and 5, and that statistical phase congruency provides a framework for feature detection and dependent upon the parameter choices can have different properties. Family 4 has generated possibly more visually pleasing results in Section 4.6.1, but it is clear in figure 4.8 that this set of parameters is not resilient to higher levels of Gaussian noise since the circle disappears in the 50% noise case. Family 5 does at this point show that statistical phase congruency can be used in noisy environments to detect features with performance comparable to that of phase congruency. This difference in noise resilience could be because of the smaller central wavelength of family 4.

Figure 4.9 shows that salt and pepper noise affects both statistical phase congruency and phase congruency significantly in their ability to detect features. Family 4 is again very poor and phase congruency appears slightly better than family 5, but neither detect the circle under 70% noise.

Further testing shows that family 6 has the best performance at detecting features under these noisy conditions. Previously (in figure 4.4) family 6 produced blurred output, but was able to select the larger scale features such as the tree's trunk, and the outline of the woman. This highlights that this family operates at a larger scale due to its central wavelength being 17 pixels instead of the 3 and 8 of the other families. This makes family 6 more appropriate when pixel based noise is added to an image, and this can be seen from the results in figure 4.10 where there is considerably more structure in the response to 90% Gaussian noise than for phase congruency. This is also the case for the salt and pepper noise tests where there is good structure in the 50% noise case and residual structure at 70%.

(a) The test image with 10% Gaussian noise

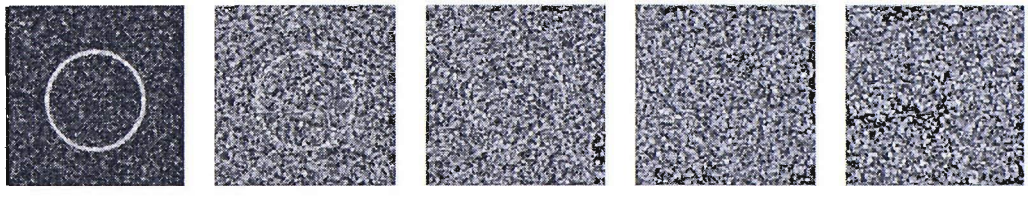
(b) The test image with 30% Gaussian noise

(c) The test image with 50% Gaussian noise

(d) The test image with 70% Gaussian noise

(e) The test image with 90% Gaussian noise

Test images



(f) Results with 10% Gaussian noise

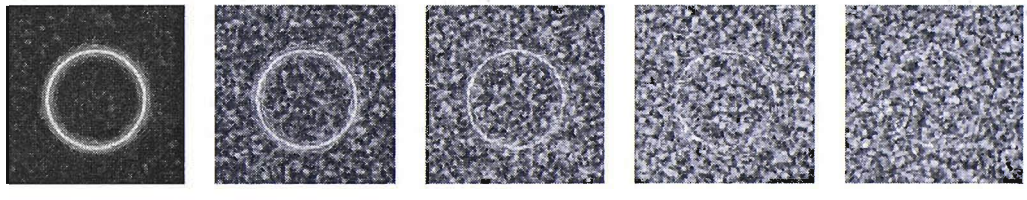
(g) Results with 30% Gaussian noise

(h) Results with 50% Gaussian noise

(i) Results with 70% Gaussian noise

(j) Results with 90% Gaussian noise

Family 4



(k) Results with 10% Gaussian noise

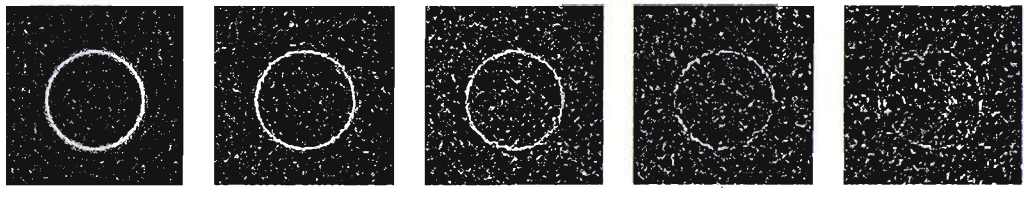
(l) Results with 30% Gaussian noise

(m) Results with 50% Gaussian noise

(n) Results with 70% Gaussian noise

(o) Results with 90% Gaussian noise

Family 5



(p) Results with 10% Gaussian noise

(q) Results with 30% Gaussian noise

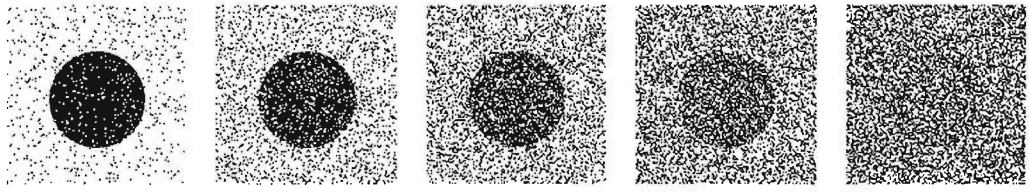
(r) Results with 50% Gaussian noise

(s) Results with 70% Gaussian noise

(t) Results with 90% Gaussian noise

Phase congruency

FIGURE 4.8: Responses of statistical phase congruency to a test image with increasing levels of zero mean additive Gaussian noise with comparative results generated using phase congruency.



(a) The test image with 10% salt and pepper noise

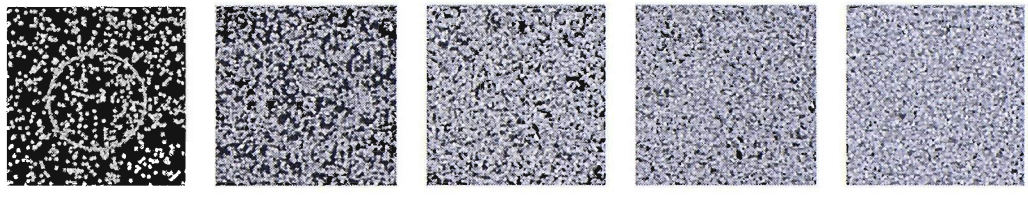
(b) The test image with 30% salt and pepper noise

(c) The test image with 50% salt and pepper noise

(d) The test image with 70% salt and pepper noise

(e) The test image with 90% salt and pepper noise

Test images



(f) Results with 10% salt and pepper noise

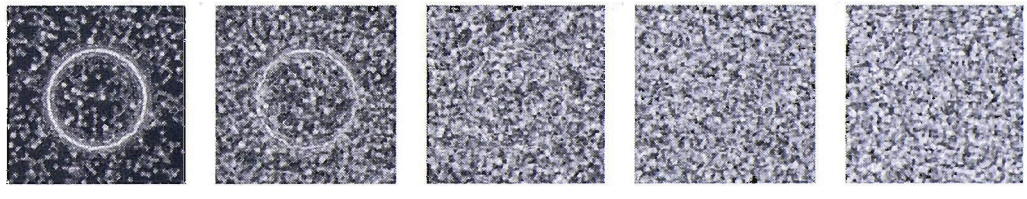
(g) Results with 30% salt and pepper noise

(h) Results with 50% salt and pepper noise

(i) Results with 70% salt and pepper noise

(j) Results with 90% salt and pepper noise

Family 4



(k) Results with 10% salt and pepper noise

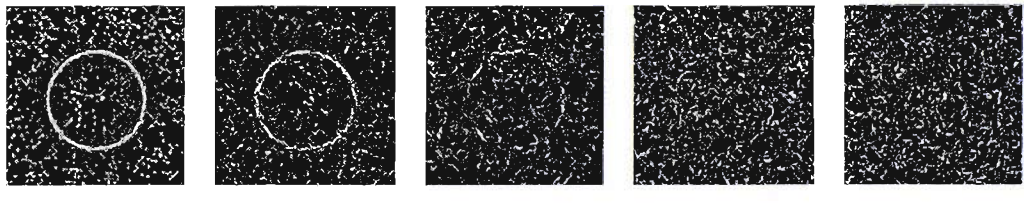
(l) Results with 30% salt and pepper noise

(m) Results with 50% salt and pepper noise

(n) Results with 70% salt and pepper noise

(o) Results with 90% salt and pepper noise

Family 5



(p) Results with 10% salt and pepper noise

(q) Results with 30% salt and pepper noise

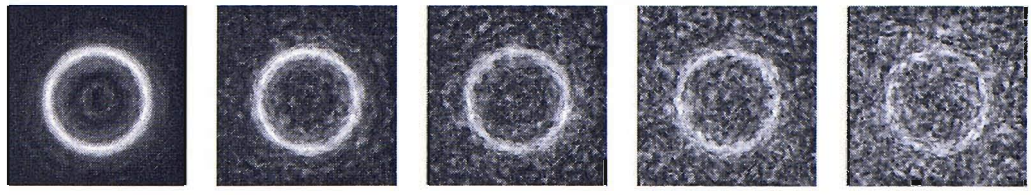
(r) Results with 50% salt and pepper noise

(s) Results with 70% salt and pepper noise

(t) Results with 90% salt and pepper noise

Phase congruency

FIGURE 4.9: Response of statistical phase congruency to a test image with increasing levels of salt and pepper noise with comparative results generated using phase congruency.



(a) Results with 10% Gaussian noise

(b) Results with 30% Gaussian noise

(c) Results with 50% Gaussian noise

(d) Results with 70% Gaussian noise

(e) Results with 90% Gaussian noise

Family 6

(f) Results with 10% Gaussian noise

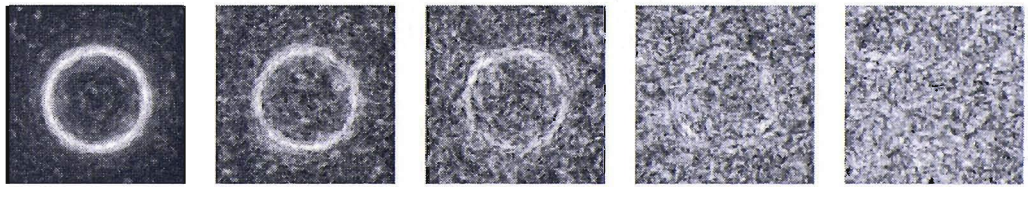
(g) Results with 30% Gaussian noise

(h) Results with 50% Gaussian noise

(i) Results with 70% Gaussian noise

(j) Results with 90% Gaussian noise

Phase congruency



(k) Results with 10% salt and pepper noise

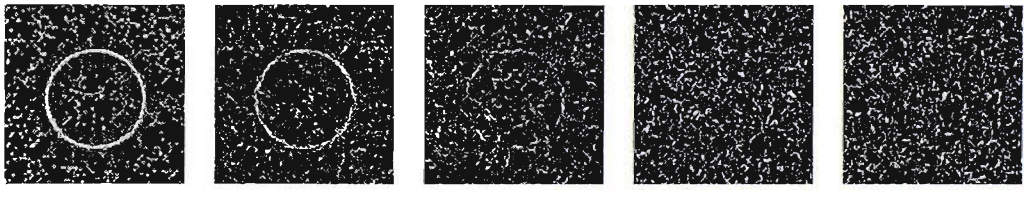
(l) Results with 30% salt and pepper noise

(m) Results with 50% salt and pepper noise

(n) Results with 70% salt and pepper noise

(o) Results with 90% salt and pepper noise

Family 6



(p) Results with 10% salt and pepper noise

(q) Results with 30% salt and pepper noise

(r) Results with 50% salt and pepper noise

(s) Results with 70% salt and pepper noise

(t) Results with 90% salt and pepper noise

Phase congruency

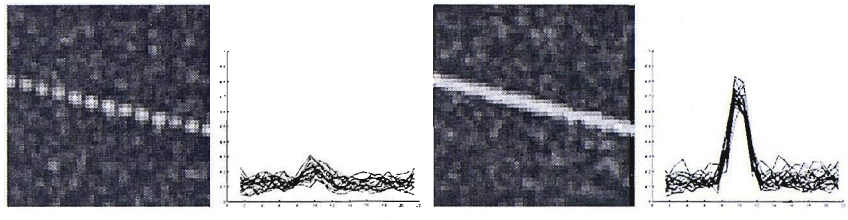
FIGURE 4.10: Statistical phase congruency tuned to detect features even in extremely noisy conditions.

4.6.3 Examining the effects of 'discontinuous' lines

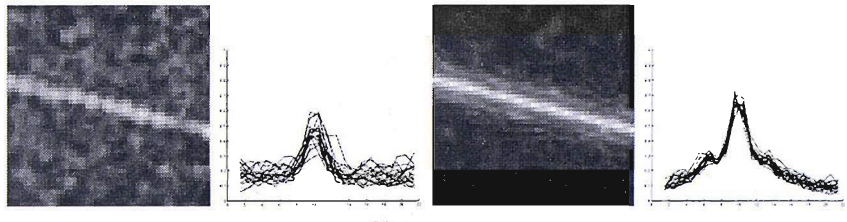
One of the main motivations behind developing this technique for measuring phase congruency is the negative results from feature detection applied to a plain image with a line of dots traversing it, for example in fig. 4.6.3. The technique by Kovesi shows some odd behaviour, as do some results for moving features, both of which were discussed in Section 3.3.6.

Figure 4.11 shows results for statistical phase congruency and phase congruency to two synthetic images. The results are present as complete images showing the overall response, and alongside each image cross sections have been presented. The cross sections are taken from columns between the impulses and between the transitions in the step edge case. They are overlaid and each cross section is aligned such that the centre of the line is in the same place. Each of the resulting images show a strong response to the feature in the image. The cross section slices show strong feature detection even in the case of the impulses where no feature is actually present due to temporal support. The noise in both images produces stronger responses from the phase congruency operator than any of the statistical phase congruency families. It is also noticeable in the cross section of the moving step edge that phase congruency produces echoes to the sides of the main response. It is these echoes that statistical phase congruency was designed to reduce or remove. Removing these echoes is an important process when considering extending a feature detector into 2D+T, where edges and features can move in such 'disconnected' manners.

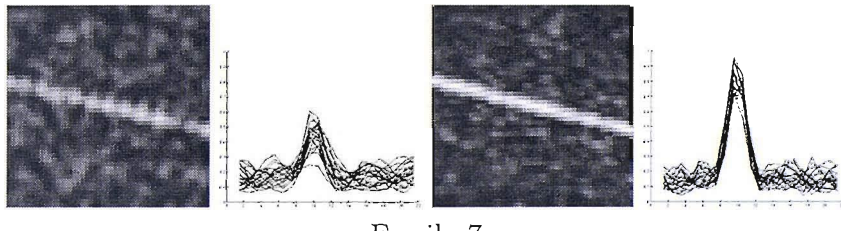
A zero mean Gaussian noised impulse and step used for testing.



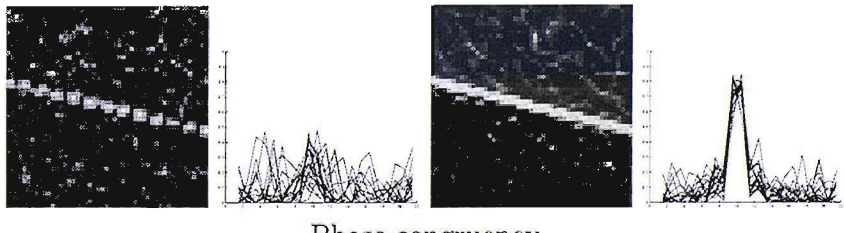
Family 4



Family 5



Family 7



Phase congruency

FIGURE 4.11: Different responses to a 'fast moving' impulse and step from statistical phase congruency (family 4, 5 and 7) and phase congruency. Both in image form and plots of the cross sections from the columns between the points of 'movement'.

4.7 Conclusions

In this chapter a new method for measuring phase congruency has been presented. To do this the filtering process has been modelled. The model has been used to analyse different families of filters with respect to their ability to measure phase congruency. The spatial extent of each filter has also been measured and a mean value taken of each family. With both the spatial extent and a filter family's ability to measure phase congruency seven families were chosen. From these seven, three families showed good results against a series of test images.

Noise testing of the new statistical phase congruency produced positive results with improvements over the previous method of measuring phase congruency. This is very encouraging, and in addition some problematic results found using phase congruency in chapter 3 have also been removed. In overcoming these problems we have designed a feature detector that should be more suitable for extending into 2D+T. This is the subject of the next chapter.

Chapter 5

Temporal Statistical Phase Congruency

In chapter 3 we extended phase congruency into the temporal domain. Further investigation into phase congruency in section 3.3.6 showed some interesting and undesirable results. These results provoked the development of statistical phase congruency in chapter 4. This new method for feature detection proved to produce good results on test images, but also produced acceptable results when tested against an image of an inclined line of impulses, or if viewed using a 1D+T framework, a fast moving impulse, something phase congruency fails to do. The ability of the technique to correctly detect a fast moving impulse in 1D+T, suggests that this method for measuring phase congruency should be more suited to temporal extension for detecting time persistent features.

Since statistical phase congruency is fundamentally a 1D metric, section 5.1.1 describes the changes needed to transform the filters used in 2D in chapter 4 into the 2D+T domain. The extension of statistical phase congruency into the temporal domain allows for features to be detected at multiple spatial and temporal angles. In section 5.3 interpolation methods previously described in section 3.2.4 are applied to results and allow the dominant velocity and spatial orientation of a feature to be established. Results from applying temporal statistical phase congruency are presented in section 5.5. These begin with example images in section 5.5.1, with results from different noise tests in section 5.5.2. The technique's response to occlusion is examined in section 5.5.3 and section 5.5.4 evaluates the accuracy of the velocity interpolation aspect of temporal statistical phase congruency. The final set of results in section 5.5.5 show the effectiveness of temporal statistical phase congruency when used in velocity thresholding. The last section of this chapter gives some conclusions to the technique and suggests further possible areas of research.

5.1 Introduction

In extending statistical phase congruency to image sequences a number of additional stages are necessary. Although we rely heavily upon the underlying ability of statistical phase congruency to detect features, we also attempt to detect these features at different spatio-temporal angles in an image sequences. In section 5.1.1 of this chapter, we describe how the filters are extended into 2D+T. The use of these filters to measure temporal statistical phase congruency is described in section 5.2. Section 5.3 suggests the use of a velocity interpolation algorithm, and a noise reduction method is described in section 5.4.

5.1.1 Filter construction

The filters used in the temporal extension to statistical phase congruency are constructed using polar coordinates. If we consider the 2D filters described again in equation 5.1 we see that there are three constituent parts. Each filter is defined by a Gabor function, $G(\cdots)$, either log-Gabor or normal Gabor function with or without low-pass filtering. This function selects the overall band of the frequency spectrum the filter is operating in. The second part is a Gaussian function, $Gaussian(\cdots)$, that defines the spatial angle of focus that the filter will operate on. The third part is the phase shift function, $ps_n(\cdots)$. This function is defined more completely in equations 4.6 and 4.7, and it is the relationship between filters with different ps_n that are used to estimate phase congruency. It is therefore worth noting that the first two parts of the filter, $spc2d_n$, select the focus of the filter.

$$spc2d_n = G(\omega, \omega_n)\ Gaussian(\theta, \theta_n)\ ps_n(\omega) \tag{5.1}$$

$$Gaussian(\theta, \theta_n) = e^{\frac{-(\theta-\theta_n)^2}{2\sigma_\theta^2}} \tag{5.2}$$

where ω is the spatial frequency and θ is the spatial angle. ω_n is the central frequency of the Gabor filter, and θ_n is the centre angle of the filter. σ_θ controls the spread of the filter about the angular axis.

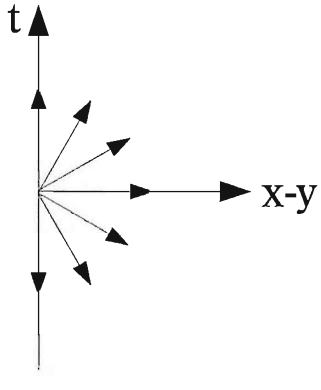
To extend these filters into the 2D+T domain we add a further Gaussian function that operates about the temporal angular axis, ψ. This then gives us equation 5.3 composed of four distinct functions. The first three select the frequency band or scale of operation (ω_n), the spatial angular focus (θ_n) and the temporal angular focus (ψ_n). This allows phase congruency to be detected within this section of frequency space using filters with

different phase shift functions, ps_n. These phase shift functions are inherited from the 2D version and all seven families will be evaluated.

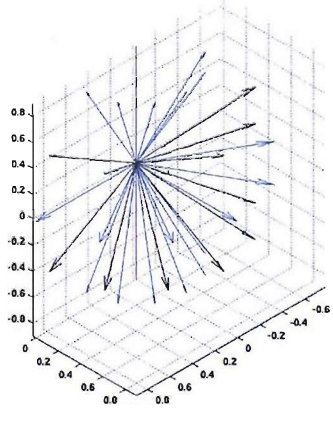
$$spc2dt_n = G(\omega, \omega_n)\ Gaussian(\theta, \theta_n)\ Gaussian(\psi, \psi_n)\ ps_n(\omega) \quad (5.3)$$

$$Gaussian(\psi, \psi_n) = e^{\frac{-(\psi-\psi_n)^2}{2\sigma_\psi^2}} \quad (5.4)$$

The number of filters used in estimating phase congruency using statistical phase congruency is a combination of the number of different spatial and temporal angles selected, and the number of different phase shift functions are chosen. In developing temporal phase congruency we used two different arrangements of filters in 2D+T space. We chose here to use just the 'singularity' arrangement in developing temporal statistical phase congruency. This arrangement sets some of the filters to be aligned to zero speed, which would make them particularly sensitive to stationary features which could be useful. Either arrangement is suitable, but choosing one reduces the number of tests needing to be completed as there are already 7 families of phase shift function to test with. A diagram of the 'singularity' arrangement of spatio-temporal foci is shown in figure 5.1



(a) Temporal axes filter orientations with 'singularities'



(b) filter orientations with 'singularities'

FIGURE 5.1: This diagram shows the foci of the filters used in temporal statistical phase congruency.

5.1.2 Frequency band and central wavelength selection

In chapter 4 we examined a method for measuring phase congruency using multiple phase altered filters centred about a single section of the frequency space. The application of multiple phase shifted filters has allowed the detection of features within images of differing feature scales based upon the bandwidth of the filter selected, and its central frequency. Re-applying this concept to the spatio-temporal domain, however could produce interesting results. If we consider that the spatio-temporal frequency is a mixture of the spatial and the temporal, then fixing the spatio-temporal frequency and altering the temporal angle will alter the portion of the frequency that pertains to a feature's spatial aspects. The alteration is actually a reduction in the amount of high frequency spatial frequencies the filter will capture, reducing the weighting given to smaller features. This would cause a problem for filters with smaller bandwidths. Therefore, it may be important to use filters that cover the desired bandwidth with leeway given to the exact measure. This argument does not take into consideration the effects of aliasing upon what a given bandwidth may contain when smaller features move 'quickly'.

5.2 Calculating temporal statistical phase congruency

The calculation of temporal statistical phase congruency uses the results of filters with the same central wavelength, spatial angular focus and temporal angular focus, but different phase shift function to estimate the degree of phase congruency at a pixel in an image sequence. The result is an estimate of feature strength considering the filter's angle of focus. This allows a single point to have a high estimate about one spatio-temporal angle and low estimates at others. It also allows for features to be detected with multiple high degrees of phase congruency that may signify a corner or other type of junction.

We consider that R_n is the nth complex response of a filter ($spc2dt_n$) to an image sequence, I. This is shown in eqn. 5.5 where the total energy from the filter's is E_n.

$$R_n = I * spc2dt_n \tag{5.5}$$

$$E_n = |R_n| \tag{5.6}$$

This allows m filter responses all with the same central wavelength and spatio-temporal focus, yet different phase shift function to be combined to give the normalised standard deviation, ρ.

$$\overline{E} = \frac{1}{m}\sum_{n=1}^{m} E_n \tag{5.7}$$

$$\hat{E_n} = \frac{E_n}{\overline{E}} \tag{5.8}$$

$$\rho = \frac{1}{m}\sum_{n=1}^{m}(\hat{E_n} - 1)^2 \tag{5.9}$$

where $\overline{E}$ is the mean energy for this set of filters and $\hat{E_n}$ is the normalised energy of the nth response. ρ then provides the basis for our measure of feature strength for the same orientation as the filters used to produce R_n. ρ has proved to be correlated to phase congruency, but inversely so. This means that ρ is mapped using a polynomial line of best fit, $\mathcal{P}\{\}$, to an estimate of phase congruency. Also ρ measures phase congruency, but it does not account for points in an image where the energy at a point is so small that the results are dominated by noise. To mask out these erroneous results the mean energy, $\overline{E}$, is used as a mask, this is similar to Kovesi's implementation[16]. The final estimate of phase congruency, *spc*, is given in equation 5.10 and is calculated for each set of filter orientations, (θ_n, ψ_n).

$$spc_{\theta_n,\psi_n} = \mathcal{P}\{\rho\}\overline{E} \tag{5.10}$$

where the mapping $\mathcal{P}\{\}$ is created by fitting a line to the distribution generated when modelling the response of ρ to differing levels of phase congruency, see section 4.3.1.

5.3 Feature Velocity Interpolation

Similar to temporal phase congruency, temporal statistical phase congruency also has the ability to detect features at different orientations in the spatio-temporal domain. The technique does this by using multiple filters with specific spatio-temporal orientations and detecting a pixel's strength with respect to that particular orientation. In combining these results from the different filters we gain a finer estimate of the feature's true spatio-temporal orientation.

The technique developed in section 3.2.4 is applicable again for estimating the primary orientation of a feature in 2D+T space and results are presented in section 5.5.4.

5.4 Reducing the effects of noise

The original 2D image based statistical phase congruency technique has no method for measuring or countering the effect of noise. The new temporal statistical phase congruency method however is able to estimate noise levels at each pixel. Using the assumption that a single pixel will not have significant structure about all orientations allows an estimate to be derived for the noise level at that pixel. The estimate is taken as the smallest response from one of the orientations as this is likely to be entirely based upon the noise that pixel is subject to. Therefore to counter the effects of noise we subtract the smallest feature strength from all orientations as in equation 5.11 to generate the noise reduced estimate of statistical phase congruency, $nrspc$.

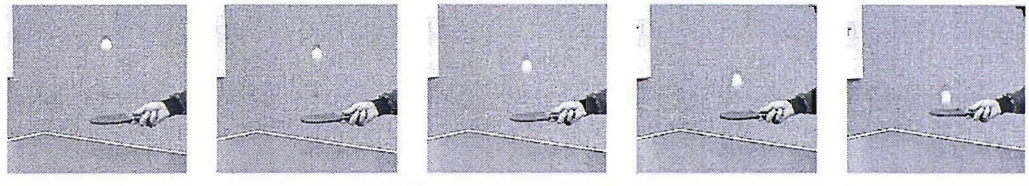
$$nrspc_{\theta_n,\psi_n} = spc_{\theta_n,\psi_n} - min(spc) \tag{5.11}$$

where $min(spc)$ is the minimum response of all spc_{θ_n,ψ_n}.

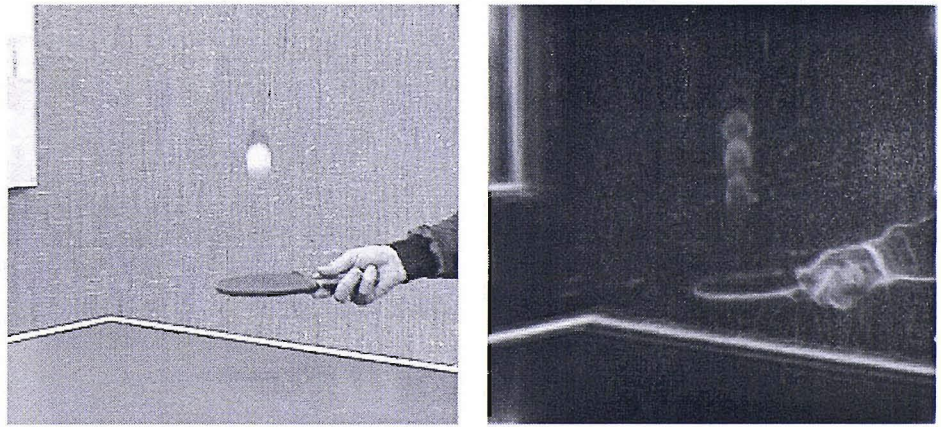
5.5 Results

This section shows the results of temporal statistical phase congruency on a number of different image types. Initially we show some example images and image sequences in section 5.5.1. Section 5.5.2 continues with more rigorous testing, and examines the effects of zero mean additive Gaussian noise and salt and pepper noise on the technique. Then repeating previous tests from section 3.3.3 we test the response to simple occluding of objects in section 3.3.3 and look at the ability of the technique to separate out different velocity features, in section 5.5.3.

5.5.1 Example results



(a) original image 14 of 32 (b) original image 15 of 32 (c) original image 16 of 32 (d) original image 17 of 32 (e) original image 18 of 32



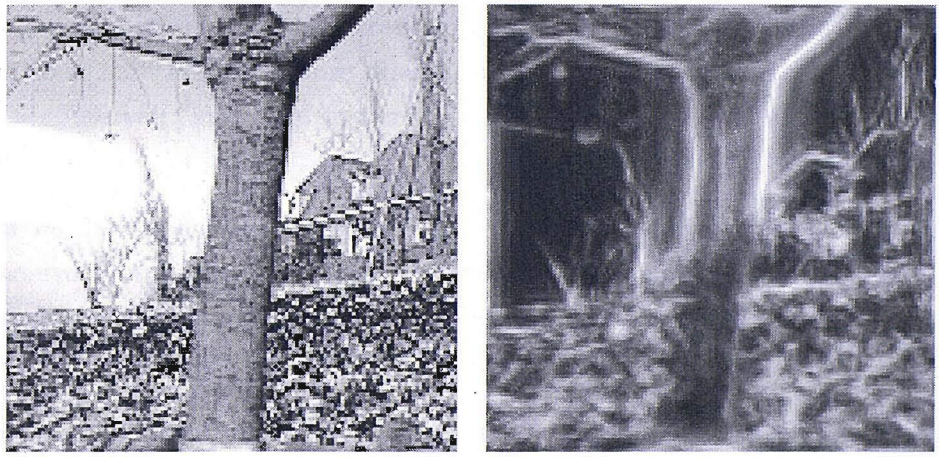
(f) original image 16 of 32 (g) temporal statistical phase congruency output

FIGURE 5.2: Results from the temporal statistical phase congruency operator on a ping-pong ball sequence.

Figure 5.2 shows the response of temporal statistical phase congruency to a range of movements. The images presented in figures 5.2(a)-5.2(e) show the ping pong ball moves relatively quickly and the hand, table and background are stationary. We see clearly that the technique strongly highlights the window frame and the table's edge. These are both strong features that persist in time through the whole sequence. The hand also

persists, but in the sequence of 32 frames used in this test sequence it moves around. the technique is still able to give good feature positioning, and pick out the details of the edges between the fingers and the detail on the sleeve on the arm. The ball however is a much faster moving object, and it is detected at its strongest in the middle position, but there are echoes of the frames either side. This was also the case in temporal phase congruency, and a suitable non-maximal suppression algorithm should be able to remove the duplicates with the added velocity of the features also being known. The side of the window frame does however have a slight surrounding echo that would not be removed by non-maximal suppression, but could be removed if the image was correctly thresholded.

(a) original image 14 of 32 (b) original image 15 of 32 (c) original image 16 of 32 (d) original image 17 of 32 (e) original image 18 of 32



(f) original image 16 of 32 (g) temporal statistical phase congruency output

FIGURE 5.3: Results from the temporal statistical phase congruency operator on a tree sequence.

Figure 5.3 shows the result of applying temporal statistical phase congruency to an image sequence of a tree in front of some houses. Five images from the sequence are shown in figures 5.3(a)-5.3(e) these are taken from the sequence of 32 images used in processing. The central frame is repeated in figure 5.3(f) with the time persistent feature detected image in figure 5.3(g). The feature detection process strongly highlights the outline of the tree particularly where the edge of the tree becomes more distinct in the upper half of the image. The roof line is also distinctly marked with the main branches of the trees to the left of the main trunk being highlighted too. There is also some detail

within the main trunk near the top of the image that is also strongly highlighted, both bark detail and a line of symmetry in the middle of the trunk which is erroneous. This is probably caused by wrongly detecting the trunk of the tree as a broad line feature. In addition to this we also note that the time-persistent feature detection also provides velocity information for all the features. This should help in combining the objects in the image as the tree trunk will have a different velocity to the features it intersects with such as the roof line and the top of the flower bed.

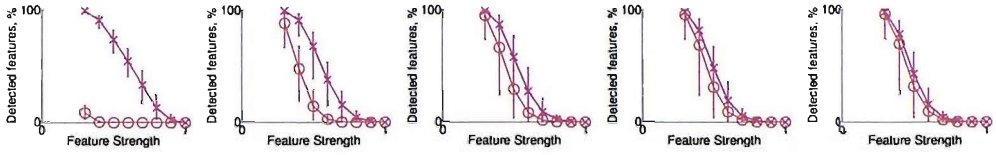
5.5.2 Noise Testing

We repeat the same noise tests applied to temporal phase congruency to temporal statistical phase congruency. We use salt and pepper noise and additive Gaussian noise in testing, the first because of its effect on the whole spectrum and the second because other noise, at the limit, can be reduced to it. The tests consisted of detecting the edge of a circle moving in an image sequence. The region where valid feature pixels should be detected is a ring 2 pixels wide in each image and is a small percentage of the total image sequence. In comparing the number of false pixels detected and the number of true feature detections we have used percentages of each region where features are detected. There is no accounting for feature orientation information in this test. The thresholds were at set intervals, measured in percentages of the peak value, of 10% between 30% and 100%.

The first results are for the sequence with zero mean Gaussian noise added. The noise level is controlled by varying σ, the spread of the Gaussian noise. This spread is measured as a percentage of the greyscale difference between the circle and the background. The results presented are for the central frame of the sequences.

We have included the results from 4 families in figure 5.4 and compared them with the results from temporal phase congruency. The four different families show different degrees of resilience to noise. This is not surprising as they have different central wavelengths of their base filters. Family 6 produces the best results, and this is to be expected as the base filter for this family has a wavelength of 17. The other families all produce good results for $\sigma = 10\%$, and family 5 produces acceptable differences at $\sigma = 30\%$. Temporal phase congruency also shows some very good results in comparison, with an interesting difference between it and family 6. Family 6 detects the edge of the circle fully, even at the greatest noise level, but also begins to erroneously detect parts of the background as feature points. Temporal phase congruency detects less of the valid feature region, but also less of the background is picked out as noise.

The second test is run using increasing salt and pepper noise. This noise affects an increasing number of pixels in the image, and randomly sets 50% of those selected pixels to black and the rest to white. These tests show that temporal statistical phase



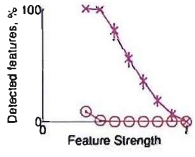
(a) Family 4 σ = 10%

(b) Family 4 σ = 30%

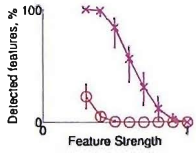
(c) Family 4 σ = 50%

(d) Family 4 σ = 70%

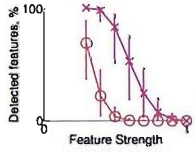
(e) Family 4 σ = 90%



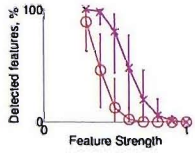
(f) Family 5 σ = 10%



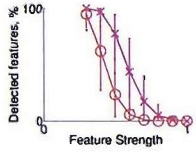
(g) Family 5 σ = 30%



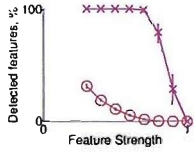
(h) Family 5 σ = 50%



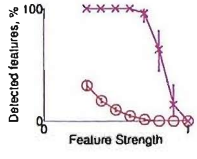
(i) Family 5 σ = 70%



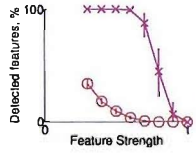
(j) Family 5 σ = 90%



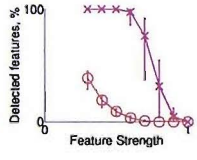
(k) Family 6 σ = 10%



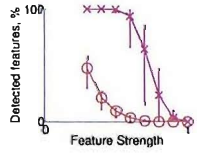
(l) Family 6 σ = 30%



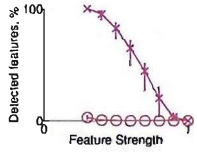
(m) Family 6 σ = 50%



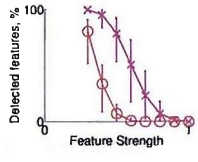
(n) Family 6 σ = 70%



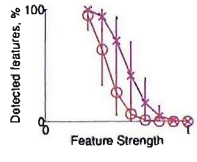
(o) Family 6 σ = 90%



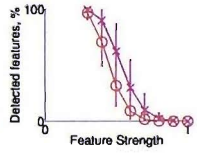
(p) Family 7 σ = 10%



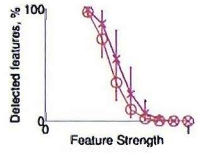
(q) Family 7 σ = 30%



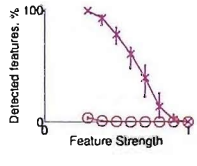
(r) Family 7 σ = 50%



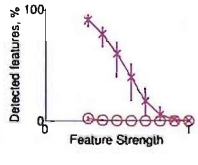
(s) Family 7 σ = 70%



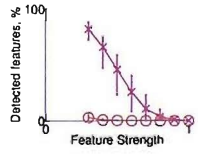
(t) Family 7 σ = 90%



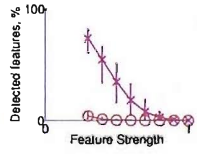
(u) Temporal Phase Congruency σ = 10%



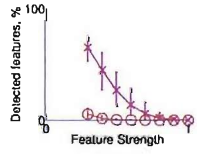
(v) Temporal Phase Congruency σ = 30%



(w) Temporal Phase Congruency σ = 50%



(x) Temporal Phase Congruency σ = 70%



(y) Temporal Phase Congruency σ = 90%

FIGURE 5.4: Results from testing a simple white disc moving on a black background with Gaussian noise added. The plots show the percentage of each region that contains detected features. The magenta crossed line shows the percentage true positives and the red ringed line shows the percentage of false positives. A range of results are presented due to the test being run fifty times.

congruency produces better results if using family 6.

Although both sets of results show varying degrees of success at detecting the correct edge pixels they also highlight the differences between the techniques. Families 4, 5, and 7 all have relatively small central wavelengths and the success of temporal phase congruency and family 6 highlights the fact that these tests are testing a technique's ability to distinguish between two regions. If the circle was a line feature rather that a step feature it would quickly disappear amongst the noise.

5.5.3 Occlusion

In this section we present results from applying temporal statistical phase congruency to image sequences that contain simple occlusion. All the sequences are of a circle moving vertically, with a fixed region of the circle being occluded. Testing against occlusion is important as it is a common occurrence in moving object analysis. Results presented here are for family 4.

In figure 5.6 the circle is very clearly detected in all the images, and the edges are well defined at the point of occlusion. This is a positive result as it shows that even though the temporal aspect of the new operator enhances its resilience to noise, it does not blur the important boundaries between objects.

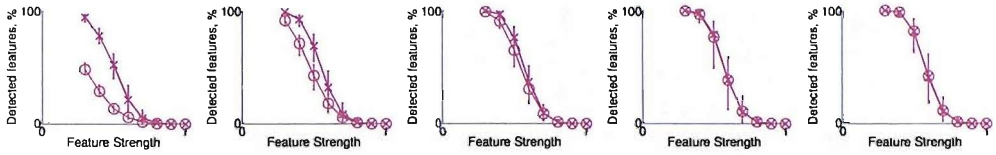
Broader occlusions are shown in figure 5.7. These are much clearer as the distance between features is greater.

5.5.4 Feature Velocity Testing

Since temporal phase congruency is a time persistent feature detector. It can detect a feature's orientation through space-time. It is important therefore to present example results and test the accuracy of the orientation information of any detected features.

Figure 5.8 shows an example of the normal orientation data that can be extracted from temporal phase congruency. From the image it looks like the normals to the features appear consistent with the underlying image data. A more quantitative set of results is also calculated from a synthetic image sequence. The sequence is the same as that used in section 3.3.4 which if defined in 3D space would be the same as a solid black sphere inside a solid white cube. This allows the testing of different spatial and temporal angles with a known 3D orientations, or known 2D+T velocities. A slight amount of zero mean additive Gaussian noise was also added.

The test sequence was passed through the temporal phase congruency feature detector, and then the orientation of all the significant feature points was established using the estimation method described in section 3.2.4.



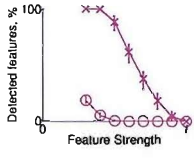
(a) Family 4 σ = 10%

(b) Family 4 σ = 30%

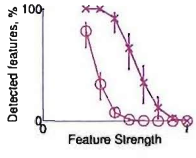
(c) Family 4 σ = 50%

(d) Family 4 σ = 70%

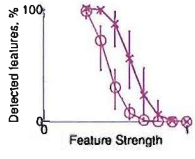
(e) Family 4 σ = 90%



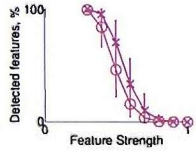
(f) Family 5 σ = 10%



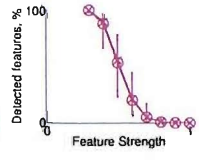
(g) Family 5 σ = 30%



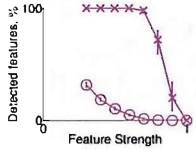
(h) Family 5 σ = 50%



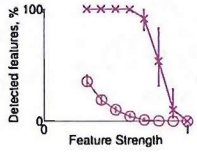
(i) Family 5 σ = 70%



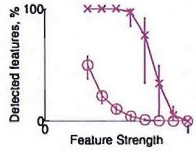
(j) Family 5 σ = 90%



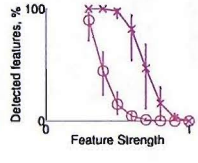
(k) Family 6 σ = 10%



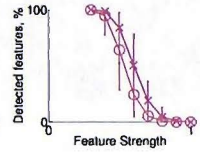
(l) Family 6 σ = 30%



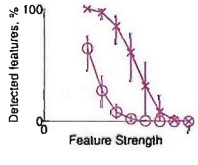
(m) Family 6 σ = 50%



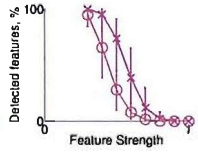
(n) Family 6 σ = 70%



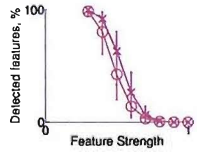
(o) Family 6 σ = 90%



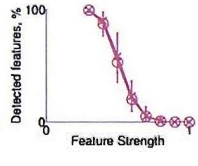
(p) Family 7 σ = 10%



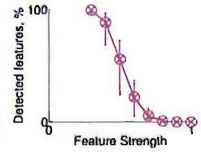
(q) Family 7 σ = 30%



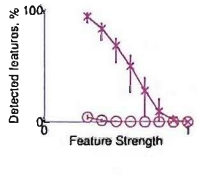
(r) Family 7 σ = 50%



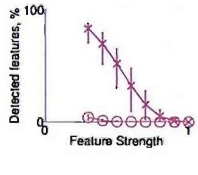
(s) Family 7 σ = 70%



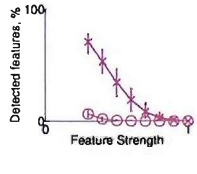
(t) Family 7 σ = 90%



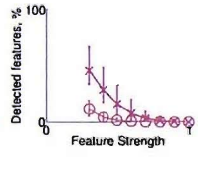
(u) Temporal Phase Congruency σ = 10%



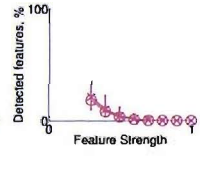
(v) Temporal Phase Congruency σ = 30%



(w) Temporal Phase Congruency σ = 50%

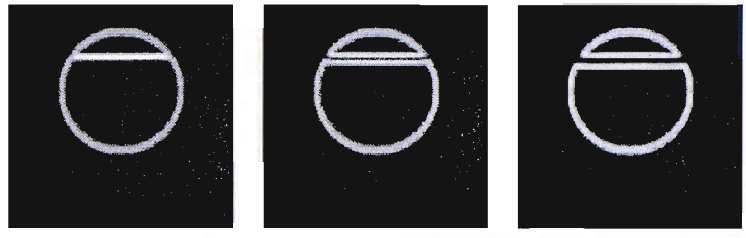


(x) Temporal Phase Congruency σ = 70%



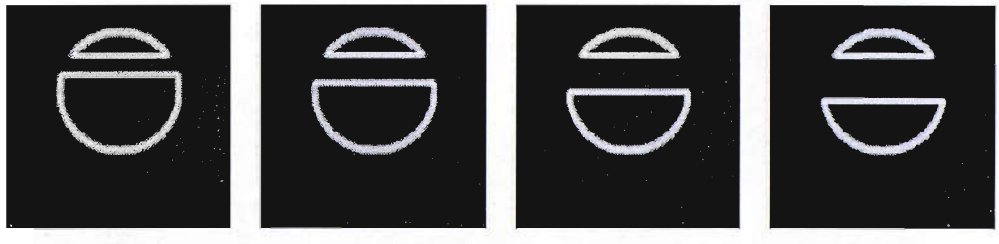
(y) Temporal Phase Congruency σ = 90%

FIGURE 5.5: Results from testing a simple white disc moving on a black background with Gaussian noise added. The plots show the percentage of each region that contains detected features. The magenta crossed line shows the percentage true positives and the red ringed line shows the percentage of false positives. Error bars show the range of results due to the test being run fifty times.



(a) 1 pixel wide (b) 3 pixels wide (c) 5 pixels wide

FIGURE 5.6: Results from the temporal statistical phase congruency operator on sequences of a circle moving vertically with a fixed position bar of differing heights occluding the circle.



(a) 10 pixels wide (b) 15 pixels wide (c) 20 pixels wide (d) 25 pixels wide

FIGURE 5.7: Results from the temporal statistical phase congruency operator on sequences with a fixed position bar of differing heights occluding the image sequence.

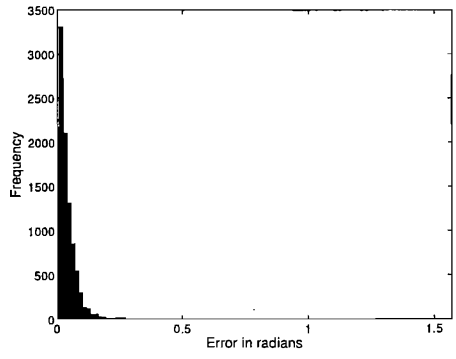
The orientations are compared to values calculated when generating the test data and distances from this 'ground truth' is calculated for all features over .7 in strength. The results are then presented and compared to similar results for a 3D Sobel operator[8] that has been applied to the same sequence.

From the separate histograms in figure 5.9 temporal statistical phase congruency is able to detect feature orientation better than the Sobel operator. The spatial accuracy appears to be better than the temporal accuracy for the temporal statistical phase congruency operator, but both are an improvement upon the Sobel operator's estimate. It is important to note that temporal statistical phase congruency does of course inherit illumination invariant properties which are not a tenet in traditional edge detection. As such, though the improvement here does not appear to be considerable, there remains significant advantage in feature detection by temporal statistical phase congruency with improved (secondary) description capabilities.

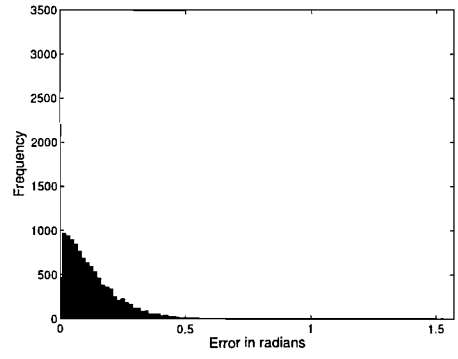
FIGURE 5.8: Example results of orientation normal information for features extracted from temporal phase congruency of a woman's face.

5.5.5 Velocity Thresholding

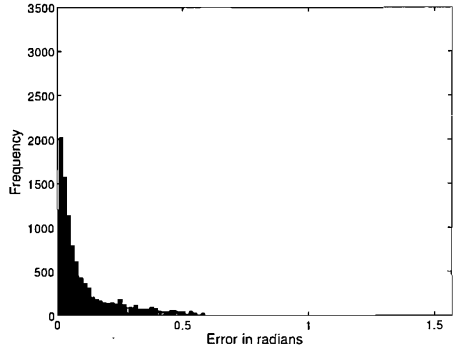
Since temporal statistical phase congruency extracts the velocity of a feature, it is possible to threshold based upon velocity. This can be a post-processing stage, or the technique can be altered so that it only looks for features of a particular velocity. Altering the process should reduce the time the technique takes to process a sequence, but the additional orientations at 'unwanted' velocities could be used in noise detections as features are detected in orientations that are similar to their true orientation, whereas noise is not necessarily detected. Example frames from a sequence of a person walking are shown in figure 5.10.



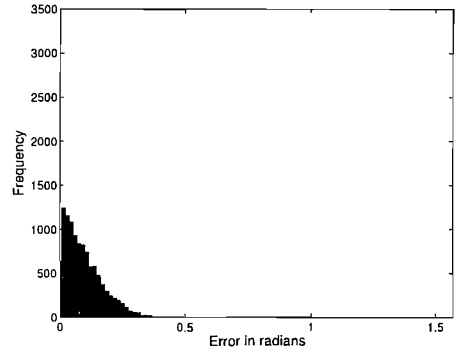
(a) Spatial feature orientation results for temporal statistical phase congruency



(b) Spatial feature orientation results for a 3D Sobel operator



(c) Temporal feature orientation results for temporal statistical phase congruency



(d) Temporal feature orientation results for a 3D Sobel operator

FIGURE 5.9: These plots show the ability of the temporal statistical phase congruency operator to detect the orientation of a wide variety of feature orientations.

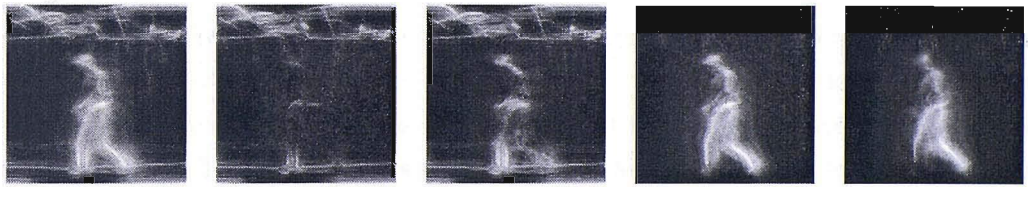
Figure 5.10 show the results from selecting different velocity ranges from the results of applying temporal statistical phase congruency to an image sequence of a person walking. The different velocity ranges are very effective at showing the aspect of the image that is moving. In the results from each of the different techniques we see the front foot is very slow moving, and so is present in the near zero and slow velocity results, but not present in the two faster moving velocity ranges. The same is true of the ceiling which does not move at all. It is also interesting to see the front thigh and the whole of the back leg are detected at the higher speeds in each of the families.

(a) original image 14 of 32 (b) original image 15 of 32 (c) original image 16 of 32 (d) original image 17 of 32 (e) original image 18 of 32

Test images

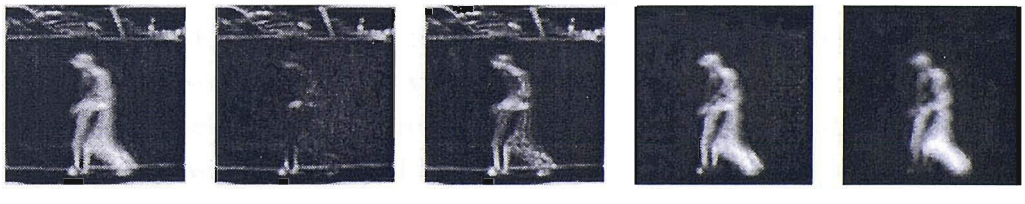
(f) all velocities (g) near zero velocities selected (h) slow velocities selected (i) medium velocities selected (j) fast velocities selected

Family 4



(k) all velocities (l) near zero velocities selected (m) slow velocities selected (n) medium velocities selected (o) fast velocities selected

Family 5



(p) all velocities (q) near zero velocities selected (r) slow velocities selected (s) medium velocities selected (t) fast velocities selected

Family 7

FIGURE 5.10: Results from the temporal statistical phase congruency operator on sequences with ranges of velocities selected. All results are taken from a sequence of 32 frames of which the 16th frame is shown.

5.6 Conclusions

In this chapter we have taken the statistical phase congruency operator and extended it to work in the spatio-temporal domain. This allows the detection of time persistent features. The extension of statistical phase congruency into the 2D+T domain has increased the information that is generated about feature points, particularly the orientation of a feature is now extended into time giving the velocity information.

The extension into time has also given an increased degree of robustness and resilience to noise as shown in section 5.5.2, and has not affected the technique's ability to detect the edges of moving objects undergoing occlusion.

It is also apparent that it is possible to use the technique's ability to detect the velocity of a feature. This was strongly highlighted by thresholding an image sequence and being able to select the different features based purely on their temporal angle of detection. This new dimension of data that is available if a feature is detected using a time persistent feature detector shows to be a useful element to the information that can be extracted from an image sequence, especially with the other benefits gained.

The different parameters that can be used to define the filters used in temporal statistical phase congruency have been shown to give different benefits. Three particular families seemed to produce results that detected the finer features, but due to their more localised operation were less resilient to noise. A fourth family which detected larger scale features did show a strong resilience to noise, equal to and at times an improvement on temporal phase congruency. The use of filters that were specifically chosen to have a small spatial extent could have effects on the overall technique's ability to withstand noise. Further research could look into the properties of filter families that were not as compact as those chosen and may produce improvements particularly under noisy conditions.

Another method for improving the technique's ability to accurately detect time persistent features would be to combine the results of the different families either by simply multiplying the results together or via some other method. The variations between the filters could also be different. Currently two filters are used $\pi/2$ radians apart, if they were perfect sine waves they would be orthogonal. It would therefore be interesting to combine the results from applying filters which are orthogonal in another manner i.e. via increasing numbers of oscillation. Investigation into results from different filters in this manner have not been carried out due to time constraints.

In addition to extending statistical phase congruency, a new and quick method for noise suppression has been used in temporal statistical phase congruency that was not used in temporal phase congruency. The application of this form of noise suppression along side the current noise suppression method already used could increase the robustness of that technique, or replace it if a greater processing speed was required.

Chapter 6

Conclusions

In this thesis, time persistent feature detection using phase congruency has shown to be useful and deserving of future research. The two techniques described for time-persistent feature detection have shown marked advantages over image based feature detection. Time persistent feature detection allows for features to be detected in higher levels of noise. It takes advantage of temporal correlation and will hopefully allow computer vision to operate in more difficult imaging environments.

Time persistent feature detection also gives new information of a feature's velocity within an image sequence. Previous techniques that grouped features together to give lines or shapes could only rely on a feature's spatial orientation. The new velocity information should also allow features that move together to be grouped together, and allow for lines and shapes to be separated in time and space more accurately and robustly.

Temporal phase congruency has been shown to be a useful method for detecting time persistent features. Chapter 3 described the extension of phase congruency, an image based technique, to temporal phase congruency, a new image sequence based technique. This new idea allowed previously high levels of robustness to noise to be exceeded. It also allowed a global intensity invariant time persistent feature detector to be implemented. This is good because not only is the technique useful if there is a variation in illumination across an image, but also should the illumination change over time the technique should not be affected. The results proved also that the technique is able to extract useful features. The response of the technique to features at different spatial and temporal angles also highlighted the problems of temporal aliasing, and highlighted some of the limitations of phase congruency. The results from the different orientations were also combined in a new way to estimate the primary orientation of a feature in 2D+T.

One of the sections in chapter 3 highlighted some unusual results generated from phase congruency and also temporal phase congruency. This lead to the work of chapter 4 which considered a new way of measuring phase congruency. This method was initially

developed for images and estimated the phase structure at a point using the variation in response from different filters. A process of modelling local frequencies was developed and allowed for different filter designs to be evaluated. Seven families of filters were selected and tested on a number of different images. Three families were then highlighted and results presented. These showed improvements over phase congruency, in particular the new technique avoided problems associated with detecting a shallow angled line. These results were exciting and provided a strong foundation for extending the technique to detect time persistent features.

The final chapter of work within this thesis, chapter 5, described temporal statistical phase congruency. It contributes a novel, and robust feature detector to the field of computer vision. The technique builds upon the work of chapter 4 and applies a temporal extension using the work and inspiration from chapter 3. This new method for measuring spatio-temporal phase congruency detects features more robustly than temporal phase congruency and also estimates of feature angle are better. The different families used in temporal statistical phase congruency show the technique's ability to be adapted to different requirements.

6.1 Future work

There are a number of opportunities for further study into the research presented in this thesis. The first is shown in appendix A. This paper presented at British Machine Vision Conference 2003 described the use of temporal phase congruency for providing good estimates of phase congruency about boundaries of moving objects. This work with the further development of temporal statistical phase congruency and the ability to estimate more accurately spatio-temporal angle could be developed to provide more robust optical flow information, particularly at motion boundaries.

This work has piloted new techniques in an area of emergent interest. Naturally, it is likely that tuning could improve performance further. For both spatio-temporal techniques this tuning could involve a different method for the construction of 2D+T filters or the number and arrangement of the filters in 2D+T space. Temporal phase congruency also has a more computationally costly method for the estimation of noise than temporal statistical phase congruency, changing this noise estimation method could decrease the computational expense of the technique.

Statistical phase congruency relies upon the choice of filter family to robustly detect features. During development families were selected based upon a measure of their spatial extent. After noise testing it was shown that filters with a high wavelength and therefore spatial extent were able to detect features with higher noise levels than filter families with smaller spatial extents. Further research could evaluate additional families with larger spatial extents, but similar or better abilities to measure phase congruency.

To add to this the development of statistical phase congruency only investigated filter families constructed from sine-waves and feature strengths were calculated with filters with the same phase shape, but different positions on the frequency axis. Future research could consider mixing sine-waves with different frequencies in the same family, and also other phase varying functions. This sort of an extension to this thesis could also consider combining results from the current families in some way to further enhance the robustness of the technique. An alternative method for applying the phase shift functions about the θ-axis and/or the ψ-axis could also be considered. Currently the technique estimates the structure along the ω-axis and yet there is no reason why phase shifting needs to be along such an axis if a feature exists in 2D+T rather than being a 1D feature oriented in 2D+T space. Investigations into filters constructed with such variations could also have interesting spatial properties as well.

In a more fundamental regard, the central premise underlying this thesis has been that it is possible to detect time persistent features by processing an image sequence as a whole. As yet there has been no consideration of fundamental properties of moving feature analysis. In this regard, the basic nature of analysis could be refined further.

Despite this, we have certainly demonstrated that it is possible to detect moving and time persistent features at different scales of velocity and in respect of practical image attributes including illumination, noise and occlusion. In this regard we look forward to further developments in this new area of spatio-temporal image analysis.

Bibliography

[1] A. Mitiche, R. Feghali, and A. Mansouri. Motion tracking as spatio-temporal motion boundary detection. *Robotics and Autonomous Systems*, 43:39–50, 2003.

[2] P. Anandan. A computational framework and an algorithm for the measurement of visual motion. *International Journal of Computer Vision*, 2:283–310, 1987.

[3] Y. K. Aw, R. Owens, and J. Ross. A catalog of 1-D features in natural images. *CVGIP: Graphical Models and Image Processing*, 56(2):173–181, March 1994.

[4] J.L. Barron, D.J. Fleet, and S.S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.

[5] H. Bulthoff, J. Little, and T. Poggio. A parallel algorithm for real-time computation of optical-flow. *Nature*, 337(6207):549–553, February 1989.

[6] C. Pudney News group posting on 3D sobel. News group : sci.image.processing, March 1995.

[7] C. Pudney, M. Robins, B. Robbins, and P. Kovesi. Surface detection in 3D confocal microscope images via local engergy and ridge tracing. *Journal of Computer-Assisted Microscpy*, 8(1):5–20, 1996.

[8] C. Ronse. News group posting on 3D sobel. sci.image.processing, Nov. 1995.

[9] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[10] M.M. Del Viva and M.C. Morrone. Motion analysis by feature tracking. *Vision Research*, 38(22):3633–3653, November 1998.

[11] D.J. Fleet and A.D. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5(1):77–104, 1990.

[12] B.K.P. Horn and B.G. Schnuck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[13] B.K.P. Horn and B.G. Schunck. Determining optical flow : a retrospective. *Artificial Intelligence*, 59(1-2):81–87, 1993.

[14] I. Sobel. An isotropic 3x3x3 volume gradient operator. *http://www-users.itlabs.umn.edu/classes/Fall-2002/csci8980-2/doc/sobel.ps*

[15] S. Konishi, A. Yuille, and J. Cohghlan. A statistical approach to multi-scale edge detection. *Image and Vision Computing*, 21:37–48, 2003.

[16] P. Kovesi. Image features from phase congruency. *Videre : Journal of Computer Vision Research*, 1(3):1–27, 1999.

[17] H. Lui, T.Hong, M. Herman, and R. Chellappa. Motion-model-based boundary extraction and a realtime implementation. *Computer Vision and Image Understanding*, 70(1):87–100, april 1998.

[18] D. Marr and E. Hildreth. Theory of edge detection. *Proceedings of the Royal Society of London*, 207:187–217, 1980.

[19] D. Marr and S. Ullmann. Directional selectivity and its use in early visual processing. *Proceedings of the Royal Society of London B*, pages 151–180, 1981.

[20] B. McCane, B. Galvin, and K. Novins. On the evaluation of optical flow algorithms. In *International Conference on Control, Automation, Robotics and Vision*, 1998.

[21] Olivier Monga, Rachid Deriche, and Jean-Marie Rocchisani. 3d edge detection using recursive filtering: Application to scanner images. *CVGIP: Image Understanding*, 53(1):76–87, 1991.

[22] M.C. Morrone and D.C. Burr. Feature detection in human vision: A phase-dependent energy model. *Proceedings of the Royal Society of London. Series B, Biological Sciences*, 235(1280):221–245, December 1988.

[23] M. Mulet-Parada and J.A. Noble. 2D+T acoustic boundary detection in echocardiography. *Medical Image Analysis*, 4:21–30, 2000.

[24] Mark S. Nixon and Alberto S. Aguado. *'Feature Extraction & Image Processing'*. Newnes, 2002.

[25] M. Otte and H.H. Nagel. Estimation of optical-flow based on higher-order spatiotemporal derivatives in interlaced and noninterlaced image sequences. *Artificial Intelligence*, 78(1-2):5–43, October 1995.

[26] M. Mulet-Parada. *Intensity independent feature extraction and tracking in echocardiographic sequences*. PhD thesis, University of Oxford, 2000.

[27] M. Petrou and J. Kittler. Optimal edge detectors for ramp edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(5):483–491, May 1991.

[28] C. Pudney, P. Kovesi, and D. Stretch. Surface detection using local energy for 3D confocal microscope images. *Image Analysis*, 1999.

[29] P.I. Rockett. The accuracy of sub-pixel localisation in the Canny edge detector. *British Machine Vision Conference*, 1999.

[30] L.A. Spacek. Edge detection and motion detection. *Image and Vision Computing*, 4(1):43–56, 1986.

[31] A. Spoerri and S. Ullman. The early detection of motion boundaries. *International Conference on Computer Vision*, pages 209–218, 1987.

Appendix A

Related Work: Guiding Optical Flow Estimation, British Machine Vision Conference, 2003

Guiding Optical Flow Estimation

Peter J. Myerscough, Mark S. Nixon, and John N. Carter
Department of Electronics and Computer Science,
University of Southampton,
Southampton, SO17 1BJ, UK
[pjm01r|msn|jnc]@ecs.soton.ac.uk

Abstract

We show how optical flow estimates can be combined with boundary estimation to improve estimates of motion. The improvement is associated with blending of estimates from complementary bases of operation. The paper combines a phase-based method for optical flow with a time extended version of the phase congruency operator. By evaluation on synthetic and real image sequences, the combination of the two techniques is shown to improve motion estimation with particular advantages at motion boundaries, regions which have posed considerable difficulty for previous motion estimation techniques. The advantage is derived using the moving feature information in an extended phase congruency operator to constrain correct data in the optical flow field.

1 Introduction

Motion blur is a real problem for optical flow calculation. Most optical flow techniques use operations on groups of pixels to calculate the optical flow at a particular point. To justify this, an assumption of a single motion field, (or a smoothly varying motion field) model is used. If, however, within a group of pixels there is a motion boundary, or multiple motions violate this assumption, then the resulting field will be biased or erroneous. We show that it is possible to use motion boundary information to separate motion fields and reduce or remove blurring across such boundaries.

Knowing that both optical flow and motion boundary estimation can be computationally expensive, two techniques that are phase-based have been chosen as a basis for this work. The first is phase-based optical flow developed by Fleet and Jepson [4]. It provides dense flow fields and sub-pixel accuracy. In a review by Barron [2] it was shown to produce good results in comparison to a number of other techniques. The second is a robust feature detector which uses phase congruency, developed by Kovesi [6]. This technique is designed for image feature detection, but has been extended here to detect features that persist over time.

1.1 Phase-based Optical Flow

Fleet and Jepson[4] propose that the flow of the phase values of an image sequence's component frequencies is synonymous with the optical flow of the sequence. The first step of the technique is the convolution of a series of Gabor filters, $N(x, y, t, \omega_{x_i}, \omega_{y_i}, \omega_{t_i})$,

(with zero DC response), with the image sequence, $I(x,y,t)$ as in eqn. 1 to obtain filtered images , $R_i(x,y,t)$, as

$$R_i(x,y,t) = I(x,y,t) * N(x,y,t,\omega_{x_i},\omega_{y_i},\omega_{t_i}) \tag{1}$$

where

$$N(x,y,t,\omega_x,\omega_y,\omega_t) = G(x,y,t)(e^{-i(\omega_x x+\omega_y y+\omega_t t)} - e^{-b^2/2}) \tag{2}$$

$$G(x,y,t) = \left(\frac{1}{2\pi\sigma}\right)^{3/2} e^{\frac{-((x-x_0)^2+(y-y_0)^2+(t-t_0)^2)}{2\sigma^2}} \tag{3}$$

$$b = \frac{2^\beta+1}{2^\beta-1} \tag{4}$$

and β is a measure of the bandwidth the filter has one σ from its centre. There are 22 sets of constants $\omega_{x_i}, \omega_{y_i}$, and ω_{t_i} which orientate the filtering at six 30° intervals with $\omega_{t_i} = 0$, ten 36° intervals with $\omega_{t_i} = 1/\sqrt{3}$, and six 60° intervals with $\omega_{t_i} = \sqrt{3}$. The Gaussian envelopes are centred at (x_0,y_0,t_0). This produces a set of complex responses for frequencies at various orientations in the sequence containing phase and amplitude information. The differentials of these responses about each axis $[\nabla R_x(x,y,t), \nabla R_y(x,y,t)$, $\nabla R_t(x,y,t)]$ are then calculated using a 5-point complex central differencing kernel. These are then combined to produce an estimate of the phase gradient about each axes, $\nabla\phi_x(x,y,t)$, $\nabla\phi_y(x,y,t)$ and $\nabla\phi_t(x,y,t)$, using the identity in equation 5.

$$\frac{\delta\log(R(x,y,t))}{\delta x} = \frac{R^*(x,y,t)\nabla R_x(x,y,t)}{|R(x,y,t)|^2} = \frac{\nabla\rho_x(x,y,t)}{\rho(x,y,t)} + i.\nabla\phi_x(x,y,t) \tag{5}$$

where $R^*(x,y,t)$ is the complex conjugate of the response, $R(x,y,t)$. This in turn enables the component velocity, $\tilde{V}(x,y,t)$, to be calculated as in equation 6.

$$\tilde{V}(x,y,t) = \left(\frac{-(\nabla\phi_x(x,y,t)\nabla\phi_t(x,y,t)}{\nabla\phi_x(x,y,t)^2+\nabla\phi_y(x,y,t)^2}, \frac{-(\nabla\phi_y(x,y,t)\nabla\phi_t(x,y,t)}{\nabla\phi_x(x,y,t)^2+\nabla\phi_y(x,y,t)^2}\right) \tag{6}$$

The results are also thresholded dependent upon conditions that eliminate phase singularities [5] and points where the response to the Gabor filter is too small and possibly dominated by noise. A final step of applying a least squares operation on a local neighbourhood of component velocities produces full 2D velocity estimates.

1.2 Phase Congruency

Phase congruency is a robust feature detector. It detects not only step and line responses, but also a broader set of features [1]. Its attributes include a high degree of invariance to lighting variation within images. This paper extends the phase congruency technique to work with image sequences. The technique's first step is to convolve the image with a set of log-Gabor filters at 'l' different orientations and 'm' different scales. Log-Gabor filters are chosen because they have zero DC response, and in cosine and sine based pairs they have a quasi-quadrature relationship. At each orientation a measure of the spread of energy amongst the different scales, $w_n(x,y)$, is calculated.

$$w_n(x,y) = \frac{A_n(x,y)}{m(A_{max}(x,y)+\varepsilon)} \tag{7}$$

where $A_n(x,y)$ is the amplitude of the response to the quadrature pair of filters at scale n and $A_{max}(x,y)$ is the maximum amplitude response for the set of log-Gabor filters at all orientations. ε is a small constant avoiding division by zero which ensures that if the amplitude at a pixel becomes too small it is masked out. This is then mapped through a sigmoid function to produce $W_n(x,y)$

$$W_n(x,y) = \frac{1}{1+e^{(c-w_n(x,y))g}} \tag{8}$$

where c and g control the mapping of $w_n(x,y)$ to $W_n(x,y)$. Also an estimate of the level of noise at the different scales, T, is calculated[6]. Then phase congruency, PC, at each orientation, PC_l, is calculated from the vector sum of the log-Gabor filter responses, R_i.

$$PC_l(x,y) = \frac{\sum_n^m \left\lfloor A_n(x,y)\left(\cos(\Delta\phi_n(x,y)) - |\sin(\Delta\phi_n(x,y))|\right) - T \right\rfloor W_n(x,y)}{\sum_n^m A_n(x,y) + \varepsilon} \tag{9}$$

$$\Delta\phi_n(x,y) = \phi_n(x,y) - \bar{\phi}(x,y) \tag{10}$$

where $\phi_n(x,y)$ is the phase at point (x,y) for scale n and $\bar{\phi}(x,y)$ is the mean phase across all scales at that point. The use of $\lfloor\ \rfloor$ depict that if the quantity between is negative it is set to zero. The vector sum is thresholded by the noise level estimate, T, and scaled by the measure of the spread of the energy mapped through a sigmoid function, $W_n(x,y)$.

This is then divided by the total energy at the chosen orientation to produce a measure of phase congruency, PC_l, for that orientation. Repeating and summing of the results for the 'l' orientations gives the phase congruency measure for the image, $PC(x,y)$.

$$PC(x,y) = \sum_l^l PC_l(x,y) \tag{11}$$

2 Method

2.1 Temporal Phase Congruency

We now extend phase congruency to use inter-frame data to enable estimation of moving features with resilience to noise. The original technique looked for features in a two-dimensional image and used filters that were built from a one-dimensional signal, the log Gabor function. This was convolved with an orthogonal spreading function, in this instance the Gaussian function. An additional spreading function (orthogonal to the two original functions) can be used to create a three-dimensional (2D+T) filter to enable the detection of moving features. The measures for the estimation of noise, and energy spread are also extensible to image sequences. The original log Gabor function is

$$lg(\omega,\omega_i) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-\log(\omega/\omega_i)}{2\log(\sigma/\omega_i)}} & \omega \neq 0 \\ 0 & \omega = 0 \end{cases} \tag{12}$$

where ω is frequency, and ω_i is the tuning frequency of the filter. σ controls the spread of the filter.

This filter is convolved in the time domain (multiplied in the frequency domain) as in equation 13 for 2D filtering and as in equation 14 for 2D+T filtering. The filters are based upon a polar co-ordinate method for making log-Gabor filters, with the two orthogonal Gaussian spreading functions operating in the angular axes, and the log-Gabor filter about the radius or magnitude of frequency axis.

$$lg2D_i(\omega,\theta,\omega_i,\theta_i) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-(\theta-\theta_i)^2}{2\sigma^2}} lg(\omega,\omega_i) \quad (13)$$

$$lg2D+T_i(\omega,\theta,\psi,\omega_i,\theta_i,\psi_i) = \frac{1}{2\pi\sigma} e^{\frac{-(\theta-\theta_i)^2}{2\sigma^2}} e^{\frac{-(\psi-\psi_i)^2}{2\sigma^2}} lg(\omega,\omega_i) \quad (14)$$

where ω represents the spatial or spatio-temporal frequency, θ represents the spatial angle of that frequency and ψ represents the temporal angle in frequency space. θ_i and ψ_i are the angles the filters are focused upon, and again σ controls the spread of the filters.

This extension to phase congruency has two main advantages. The first advantage is found in the orientation at which phase congruency is detected at a particular pixel. This describes not only its spatial, but also its temporal orientation. This is the same as describing its velocity. Therefore all features extracted with the extended method have this additional attribute already defined.

Secondly the technique should be more robust to noise. This gain in robustness is justified by examining the feature that the filters respond to. In the one-dimensional case the filters are responding at a point. In the two-dimensional case the filters are responding at a point, which if part of a feature will likely be surrounded by valid feature points in a line on either side, that by themselves would cause a minor response to the filter due to the Gaussian spreading function. This improves the signal-to-noise ratio when processing an image. Therefore when considering a point in 2D+T space, the supporting responses of a point's neighbours in both spatial and temporal directions should increase the robustness.

2.2 Guiding Optical Flow Estimation

Optical flow operators suffer from motion blurring since at a boundary the estimates for motion can become mixed between one moving object and another. This is because optical flow operators typically use neighbourhood operations to compute velocity estimates or in filtering stages. Both of these occur in Fleet's technique. An example of motion blurring can be seen in figure 1 where it is possible to see that the estimates for motion in the image blur across the boundary of the circle onto the stationary (smoothly varying) background. With a moving feature detector, it should be possible to define where the motion boundary is. With this information it is then possible to erode the motion field back towards the motion boundary, reducing errors in the motion field produced.

The erosion process uses the current velocity estimates, v, the original velocity estimates, v_{orig}, and the phase congruency measures, pc to produce the new estimate, v' as in equation 15.

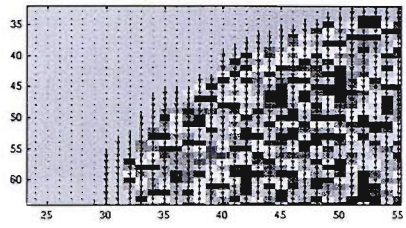


Figure 1: This figure shows part of a frame from a sequence of a moving circle with the motion vectors superimposed on each pixel

$$v'(x,y) = \begin{cases} \text{incorrect} & [|v(x,y) - v(a,b)| > \lambda_1 \wedge v_{orig}(x,y) > v_{orig}(a,b)]_{a,b \in \mathbb{R}} \wedge \\ & [pc(c,d) < \lambda_2]_{c,d \in \mathbb{P}} \\ \text{correct} & \text{in all other cases} \end{cases} \quad (15)$$

Co-ordinates (x,y) are those of the current point being considered, (a,b) are the nearest points to (x,y) in direction of the motion at that point. Points (c,d) are the points 'north', 'south', 'east' and 'west' of the current. λ_1 controls variation in the velocities. Previous values that have been deemed incorrect are always 'different' from another velocity estimate. λ_2 controls how significant a feature needs to be before it stops the erosion process. In our studies, phase congruency values greater than 0.33 are significant. Testing the original velocities means that the erosion having started from a motion boundary only creeps in one direction, that of the faster moving region. This is prescribed because faster moving regions should have a larger motion blur. Future work needs to examine more complex motion boundaries to ensure this is a valid and useful assumption.

3 Results

3.1 Temporal Phase Congruency

The new temporal phase congruency has been tested against the original phase congruency technique on a synthetic sequence of moving circles. The first test has been using a simple visual comparison. Both techniques extracted the edges of this simple image sequence very well. To gain a deeper insight, salt and pepper noise was added to the sequence in increments of 10%. At 50% salt and pepper noise, half of the pixels are set arbitrarily to black or white. Examples of the middle frame of the sequence with different noise levels are shown in figure 2.

The resulting 'feature' maps are then passed through a velocity Hough transform[7], which is a robust moving circle detector. In the results shown in figure 3 only the highest point in the accumulator for that sequence was a correct identification of the circle's velocity, and position. Anything different was considered a fail, a harsh judgement, but illustration enough of the performance possible here.

The result shows that the thresholded variant of the new temporal phase congruency operator improves results. This is because the number of sequences for which a correct

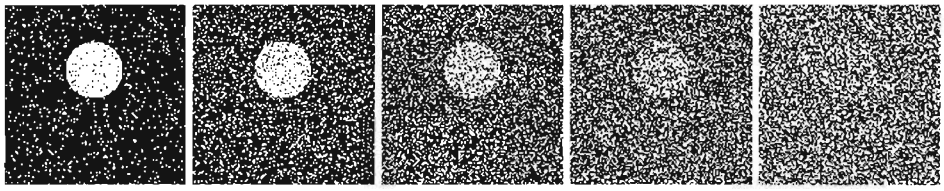


Figure 2: An example frame with 10%, 30%, 50%, 70% and 90% salt and pepper noise.

result obtained is more for the new technique and all except one exceed that of the original version. The lower results at 60% noise for the temporal phase congruency method when compared to the image based method could be attributed to too small a test set, but merits further investigation.

3.2 Guided Optical Flow

To test the new guided optical flow two sequences were used. The first was of a generated disc with a fixed random texture moving on a linearly varying background or 'slope'. The second was from the Southampton Gait Database [8], and involved a person walking on a green background. This sequence was processed three times using the separate red, green and blue channels, with the final flow fields being assimilated to produce a more dense flow field, than if either a grayscale sequence or a single colour channel was used. The densities of the flow fields even after combining the three channels were still too low. This is because Fleet's technique can only detect motion up to 2 pixels per frame without sub-sampling the images. Accordingly, another optical flow technique by Bulthoff [3] was used to buttress the density of the optical flow estimates. Differences in the density of results can be seen in figure 4.

It was assumed that within the circumference of the circle and the person were the only pixels that should contain any movement. In this way the results for this test were in four categories:

- Correct results, non-zero velocity estimates only within the 'shapes'.
- False zero velocity estimates where velocity estimates should be higher than zero
- False non-zero velocity estimates where background estimates should be shown
- Unclassified results, pixels for which the velocity is indeterminable by the optical flow techniques, or is eroded.

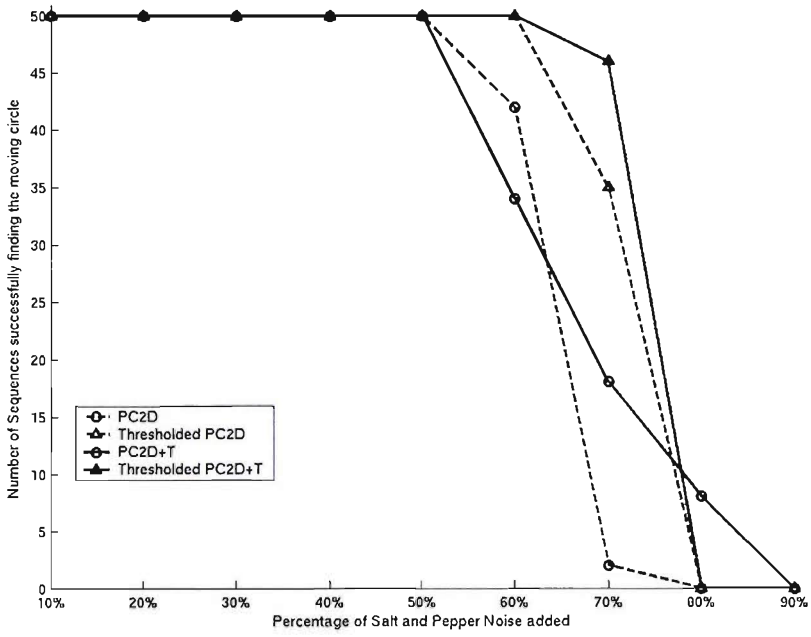
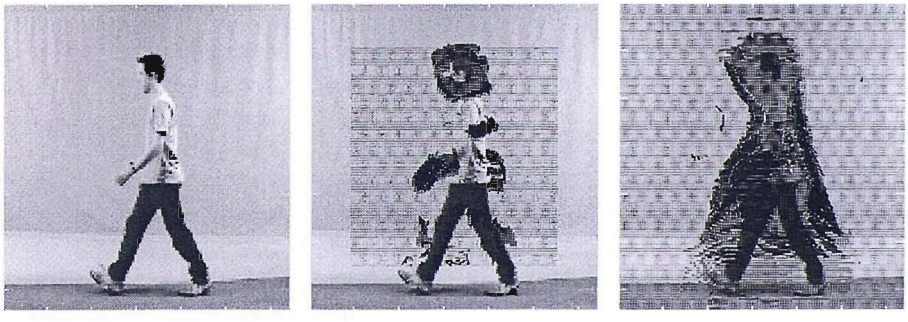


Figure 3: Graph comparing the phase congruency and temporal phase congruency.



(a) Original Image (b) Fleet(phase) (c) Bulthoff(correlation)

Figure 4: Flow field estimation for a walking person.

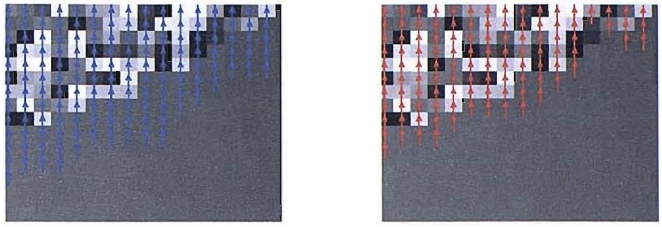
Iteration	Total Correct		False Zero		False Positive		Unclassified	
No.	No.	Percent	No.	Percent	No.	Percent	No.	Percent
0	224660	94.80%	186	0.08%	12146	5.13%	0	0.00%
1	224659	94.80%	186	0.08%	9700	4.09%	2447	1.03%
2	224583	94.76%	186	0.08%	8088	3.41%	4135	1.74%
3	224409	94.69%	186	0.08%	7440	3.14%	4957	2.09%
4	224274	94.63%	186	0.08%	7260	3.06%	5272	2.22%
5	224168	94.59%	186	0.08%	7213	3.04%	5425	2.29%
6	224132	94.57%	186	0.08%	7180	3.03%	5494	2.32%
7	224114	94.57%	186	0.08%	7157	3.02%	5535	2.34%
8	224100	94.56%	186	0.08%	7134	3.01%	5572	2.35%
9	224087	94.55%	186	0.08%	7111	3.00%	5608	2.37%
10	224081	94.55%	186	0.08%	7088	2.99%	5637	2.38%
11	224080	94.55%	186	0.08%	7076	2.99%	5650	2.38%

Table 1: Results from a sequence of images with a textured circle moving on a smoothly varying background

Iteration	Total Correct		False Zero		False Positive		Unclassified	
No.	No.	Percent	No.	Percent	No.	Percent	No.	Percent
0	9139	55.78%	3261	19.90%	2283	13.93%	1701	10.38%
1	8945	54.60%	3223	19.67%	1976	12.06%	2240	13.67%
2	8816	53.81%	3195	19.50%	1784	10.89%	2589	15.80%
3	8737	53.33%	3164	19.31%	1668	10.18%	2815	17.18%
4	8670	52.92%	3138	19.15%	1585	9.67%	2991	18.26%
5	8614	52.58%	3116	19.02%	1530	9.34%	3124	19.07%
6	8560	52.25%	3103	18.94%	1479	9.03%	3242	19.79%
7	8516	51.98%	3095	18.89%	1439	8.78%	3334	20.35%
8	8481	51.76%	3082	18.81%	1404	8.57%	3417	20.86%
9	8454	51.60%	3067	18.72%	1380	8.42%	3483	21.26%
10	8431	51.46%	3055	18.65%	1364	8.33%	3534	21.57%
11	8416	51.37%	3041	18.56%	1349	8.23%	3578	21.84%

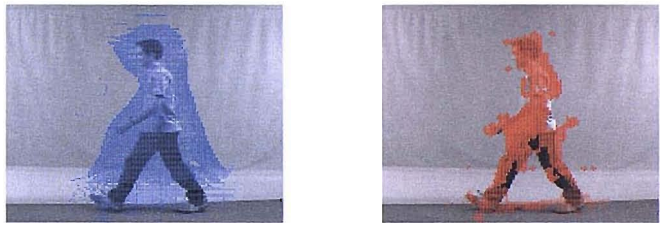
Table 2: Results from the central frame of the walking person sequence.

In both table 1 and table 2 the errors produced by the initial optical flow techniques are reclassified as 'unclassified'. This removes false confidences in the original data. The number of reclassifications is higher in the first few iterations, but the process stabilises and areas of blur are reduced to phase congruency boundaries.



(a) Flow Superimposed (b) After Erosion

Figure 5: Segments from the moving circle sequence with flow fields superimposed



(a) Flow Superimposed (b) After Erosion

Figure 6: Flow fields for the central frame of the walking person sequence.

Both figure 5 and figure 6 show that there is motion blur after the original optical flow techniques. After twelve iterations the flow field in figure 5 has stopped receding and stabilised closer to the circle's boundary. Figure 6 shows the Bulthoff optical flow operator's broad flow fields can be guided in their reduction. In this instance the erosion has eroded some valid flow vectors, but results in table 2 show the invalid vectors are more greatly reduced.

4 Conclusions

Results from a moving feature extraction technique can be used to guide selection of correct optical flow estimates thus improving the quality of motion extraction. The tests shown are currently single objects moving on a stationary background and reclassification of velocity vectors removes erroneous vectors. Future work should include multiple objects passing behind and in front of each other, as well as more complex motion junctions.

In developing this combination of motion detection, an enhanced form of the phase congruency operator has been developed. This shows improvements over the original operator in noisy conditions, although further work to remove some anomalies may be necessary. It also provides velocity information for the moving features detected. Inclusion of this motion information in the combined algorithm should also be a future work.

Preliminary studies on real image data were hampered by the sparsity of flow estimates, in part due to the large motions in the test sequences. Currently fast motion causes problems in obtaining sufficiently dense optical flow fields. This may be over come by pyramid decomposition of the image sequence, along with a method for recombining multiple scales of velocity estimates. This and other aspects merit future investigation.

Acknowledgements

This research has be funded by the EPSRC, with additional support from European Research Office of the US Army, Contract No.N68171-01-C-9002.

References

[1] Y. K. Aw, R. Owens, and J. Ross. A catalog of 1-d features in natural images. *CVGIP: Graphical Models and Image Processing*, 56(2):173–181, March 1994.

[2] J.L. Barron, D.J. Fleet, and S.S. Beauchemin. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.

[3] H. Bulthoff, J. Little, and T. Poggio. A parallel algorithm for real-time computation of optical-flow. *Nature*, 337(6207):549–553, February 1989.

[4] D.J. Fleet and A.D. Jepson. Computation of component image velocity from local phase information. *International Journal of Computer Vision*, 5(1):77–104, 1990.

[5] D.J. Fleet and A.D. Jepson. Stability of phase information. *IEEE Trans. PAMI*, 15(12):1253–1268, December 1993.

[6] P Kovesi. Image features from phase congruency. *Videre : Journal of Computer Vision Research*, 1(3):1–27, 1999.

[7] J. M. Nash, J. N. Carter, and M. S. Nixon. Dynamic feature extraction via the velocity hough transform. *Pattern Recognition Letters*, 18:1035–1047, 1997.

[8] J.D. Shutler, M. G. Grant, M. S. Nixon, and J. N. Carter. On a large sequence-based human gait database. In *Proc. RASC*, December 2002.