Extracting Differential Invariants of Motion Directly From Optical Flow

Shih Ching Fu and Peter Kovesi
School of Computer Science & Software Engineering
The University of Western Australia
35 Stirling Highway, Crawley, W.A. 6009, Australia.
{scfu, pk}@csse.uwa.edu.au

ABSTRACT

The structure-from-motion problem is traditionally addressed by establishing point correspondences and then applying classic geometry techniques. However, depending on the application, perhaps only partial scene reconstruction is necessary. In such cases, computing the first-order differential invariants of image motion, namely divergence, curl, and deformation, can directly provide information about scene structure, while avoiding complex projective geometry. Even though divergence, curl, and deformation have been shown to be useful for partial scene reconstruction, little work has been done to extract these quantities from image sequences. In this paper we propose a way to extract the differential invariants of image motion from an optical flow field using a bank of filters. The output of these filters can later be used for the recovery of surface normals and time-to-contact.

Keywords: Optical flow, motion fields, differential invariants, structure-from-motion.

1. INTRODUCTION

Computer vision is concerned with inferring three-dimensional (3D) scene information from two-dimensional (2D) images. The human visual system is adept at discerning quantities such as depth and motion, allowing us to interact with our environment without needing to come directly in contact with it [12]. Of particular interest is the role that visual motion plays in gathering information about our surroundings. Even in the absence of stereo, a monocular observer is still able to determine scene structure through deliberate camera movements, known as *active vision* [6]. The focus of this paper is on the analysis of image motion or *optical flow*, and in particular the extraction of the first-order differential invariants of image velocity using correlative filtering.

Optical flow is the approximation of a scene's 2D motion field and is typically derived from an image sequence. Two-dimensional image motion fields comprise the projections of 3D object velocities onto a 2D image plane. These velocities may be due to viewer movement (*ego-motion*), movement of individual scene objects, or a combination of both. Optical flow is only an approximation of the image motion field since assumptions need to be made about the lighting and texturing of surfaces. Assumptions of static light sources and adequate scene texture can result in the presence of optical flow in places of zero motion and vice versa [12].

13th School of Computer Science & Software Engineering Research Conference, Yanchep, Western Australia, 20th–21st September 2004.

1.1 Determining Optical Flow

There has been much work done in the area of optical flow computation [3]. However, most of the recently published literature in this field focuses on improving the speed, robustness, and accuracy of existing optical flow algorithms rather than developing new approaches [4, 7, 2, 18]. Below is a brief outline of the main techniques used in optical flow determination.

1.1.1 Gradient based methods

Gradient methods such as that by Sobey and Srinivasan [16, 17] compute image velocities by calculating the spatial and temporal derivatives of image intensities. Typically they involve finding the solution of an overdetermined system of linear equations where one constraint is the *optical flow constraint* as defined in Equation 1.

$$I_x u + I_y v + I_t = 0 \tag{1}$$

I is image intensity where subscripts denote partial derivatives, and u and v are the x and y components of the optical flow. The optical flow constraint equation is derived from the assumption that for a point on a 3D surface, the image intensity in a small neighbourhood around that point does not vary with time. Energy-based methods can be shown to use similar constraints as gradient (or differential) methods but transformed into the Fourier domain [3].

1.1.2 Point or Region Correspondence methods

Correspondence or matching methods require regions or features to be tracked through a sequence of images. Such methods are useful when accurate differentiation of the image intensities is impractical due to noise. The recovery of the motion field is therefore similar to solving the correspondence problem where point trajectories are interpreted as instantaneous velocity vectors. From this information, scene reconstruction can be treated as a classical projective geometry problem. Like many other optical flow methods,

matching methods need to assume rigid body motion, and encounter problems when the scene contains several moving objects or occlusions [5].

1.1.3 Phase based methods

A further approach to optical flow computation is to use a phase representation as done by Fleet [11]. Most algorithms, by assuming that image intensity is not time varying, approximate image motion as pure image translation. Fleet argues that the dynamics of image phase contours are better approximations to the motion field. It is proposed that a phase based approach need not assume pure image translation and performs well under image contrast variations and geometric deformation due to perspective.

1.2 Differential Invariants of Image Motion

Koenderink and Van Doorn [13, 14] have shown how the distortion of an object's 2D image can be decomposed into components of divergence, curl, and deformation. A geometric summary of these is provided in Fig 1. These quantities are known as first-order *differential invariants* of image motion fields because their values are independent of both the choice of coordinate system and any viewer rotations about the projection centre.

For example, consider a stationary scene being viewed by a moving camera and $\vec{v}(x, y)$ is the image motion field with (u, v) being a motion vector associated with every (x, y) in the image. For a small field of view we can approximate the motion field with an affine transformation:

$$\begin{bmatrix} u \\ v \end{bmatrix} \approx \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} + \begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{2}$$

where (u_0, v_0) defines the translation of the centroid of the object's image. The velocity gradient tensor (u_x, u_y, v_x, v_y), comprising first order partial derivatives, can be decomposed as follows [9]:

$$\begin{aligned} \begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix} &= \frac{\operatorname{div} \vec{v}}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ &+ \frac{\operatorname{curl} \vec{v}}{2} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \\ &+ \frac{\operatorname{def} \vec{v}}{2} \begin{bmatrix} \cos 2\mu & \sin 2\mu \\ \sin 2\mu & -\cos 2\mu \end{bmatrix} \end{aligned} \tag{3}$$

Thus we can see how the image velocity field can be deconstructed in divergence, curl, and deformation components.

1.2.1 Scene structure from divergence, curl, and deformation

The above-mentioned properties are directly related to 3D scene structure and ego-motion, and can be deduced from their affect on scene geometry. In particular, divergence relates to isotropic expansion or dilatation; curl relates to rotation or vorticity; and deformation relates to symmetric and anti-symmetric shearing. Through these relationships, these three quantities can be used to derive information about surface orientations and time-to-contact. More specifically, the *slant*, σ, and *tilt*, τ, of the surfaces in our image can be found [9].

$$\operatorname{curl} \vec{v} = -|\vec{\sigma}||\vec{V}_{\text{ang}}| \sin \tau + \Omega_{\text{rad}} \tag{4}$$

$$\operatorname{div} \vec{v} = |\vec{\sigma}||\vec{V}_{\text{ang}}| \cos \tau - \frac{2|\vec{V}_{\text{rad}}|}{|\vec{R}|} \tag{5}$$

$$\operatorname{def}^{+} \vec{v} = -|\vec{\sigma}||\vec{V}_{\text{ang}}| \cos \tau \tag{6}$$

$$\operatorname{def}^{-} \vec{v} = -|\vec{\sigma}||\vec{V}_{\text{ang}}| \sin \tau \tag{7}$$

where

$\vec{R}$	is the 3D object position relative to viewing point.
$\vec{V}_{\text{ang}}$	is the object's velocity perpendicular to the viewing direction divided by its distance from viewer.
$\vec{V}_{\text{rad}}$	is the object's velocity along the viewing direction.
$\vec{v}$	is the angular velocity of the image point.
Ω	is the angular velocity of the object about an axis through the viewing point.
Ω_{rad}	is the component of Ω about the viewing direction $\vec{R}$.
$\vec{\sigma}$	is the slant of the surface defined as the vector $\Delta\vec{R}/\vec{R}$.
τ	is the tilt of the surface.

1.2.2 Dilatation

Dilatation is interpreted geometrically as isotropic expansion about a focus of divergence, or conversely, isotropic contraction about a focus of convergence. It is typically due to the relative motion of objects toward or away from the camera. The relationship between the divergence at a point in the vector field and the orientation of the tangent plane at that point in 3D space is given by Equation 5. Notice that two terms constitute this equation. The first is proportional to the magnitude of the deformation at that point and the second term is related to the time-to-impact of the surface.

Time-to-contact, t_c, is a useful quantity for obstacle avoidance problems and can be easily calculated for scenes containing pure divergence or convergence since the first term of Equation 5 is zero.

$$t_c = \frac{|\vec{R}|}{|\vec{V}_{\text{rad}}|} = -\frac{2}{\operatorname{div} \vec{v}} \tag{8}$$

Equation 8 can be used to calculate depth maps of a scene except that distances will be measured in temporal units rather than spatial units.

1.2.3 Vorticity

Vorticity corresponds with 2D rotation of the object's image as shown by the top left illustration in Figure 1. It is expressed as a curl component in the motion field. Similar to divergence, the relationship between curl and scene structure comprises two terms. The first term in Equation 4 is

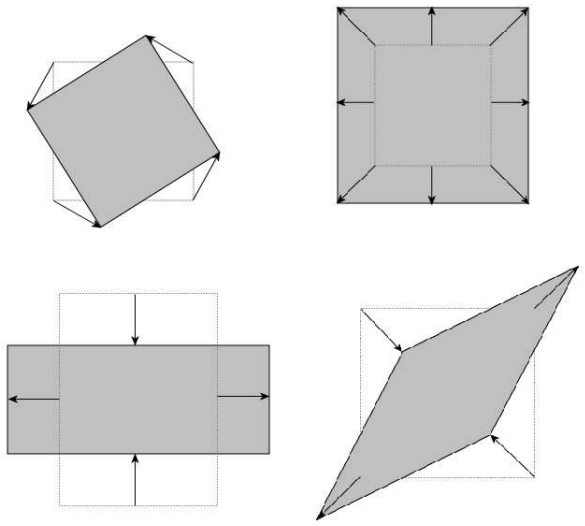

Figure 1: Koenderink and van Doorn showed that an image velocity field can be decomposed into components of curl, divergence, and deformation. From left to right, top to bottom: curl (vorticity), divergence (dilatation), and deformation (shearing about two different axes). The first two and the magnitude of the last are independent of coordinate system. The choice of axis for the deformation components means only its magnitude is a differential invariant, but any deformation can be expressed as a combination of these two components.

dependent on the apparent deformation of the object and the second term is the angular velocity about the viewing direction. It is important to note that the curl component of a vector field is orthogonal to its divergence response.

1.2.4 Shearing

Deformation or shearing, while affecting edge orientations, does not change the apparent area of a closed contour. Deformation fields are characterised by an axis of maximum extension and another perpendicular axis that describes the direction of maximum contraction. Therefore two components, a symmetric component and an anti-symmetric component, are needed to sufficiently define the shear distortion of an image shape. The two components of deformation are illustrated in the bottom two diagrams of Figure 1. Equations 6 and 7 show that the tilt of a surface can be found by the arctangent of the ratio of the symmetric and anti-symmetric deformation components. Note that only the magnitude of the deformation is a differential invariant of motion.

2. EXTRACTION OF DIFFERENTIAL INVARIANTS

Cipolla and Blake [8, 10] have done work on how to derive surface orientation and time-to-contact from divergence and deformation information. They use B-spline snakes to track the change in the apparent area of scene objects to approximate the divergence and deformation of the motion field. However, this method proves problematic in the absence of trackable features. Work has also been done by Nelson and Aloimonos [15] on the use of divergence for obstacle avoidance, deriving it mathematically but needing to use many images over time to produce good results. The time-to-crash detector implemented by Ancona [1] utilizes optical flow directly rather than post-processing for divergence.

The remainder of this paper discusses the possibility of using a correlative filtering method to extract divergence, curl, and deformation from dense motion fields. It is assumed that existing optical flow algorithms are suitably accurate and robust to provide good input fields for filtering.

2.1 Filter Design

Similar to how image features can be detected by using tuned filters, it is proposed that a vector field can be processed for dilatation, vorticity, and shearing. Below we present a preliminary design of a bank of filters for recovering divergence, curl, and deformation from optical flow fields. Examples of such filters are shown in Figures 2 to 5.

Each filter is a simple vector field depicting divergence, curl, or deformation as appropriate. The magnitude of the individual vectors is proportional to its distance from the centre of the filter. The vector magnitudes are then modulated by a 2D Gaussian envelope to reduce the effect that a sharp discontinuity at the mask boundaries might have on the filter response. The suitability of other shapes for this envelope are yet to be investigated, although intuitively it seems that Gaussian smoothing is appropriate.

The direction of the vectors in the mask depends on the mask type. Divergence masks have vectors which point radially outward from the centre as seen in Figure 2, curl masks have tangential vectors as depicted in Figure 3, and the vectors in the deformation masks lie between the axes of maximal expansion and contraction. For deformation, the two cases necessary to capture distortion in all directions are distinguished by the tilt of their axis of maximum extension, μ, as illustrated in Figures 4 and 5. Expressions for the vector field can be derived from Equations 2 and 3 by assuming that image centroid translation, (u_0, v_0) is zero.

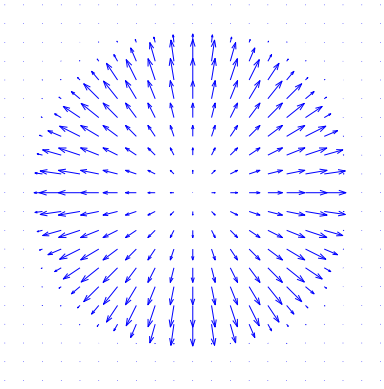

Figure 2: An example divergence detector mask of size 21×21 pixels, and radius of 9 pixels.

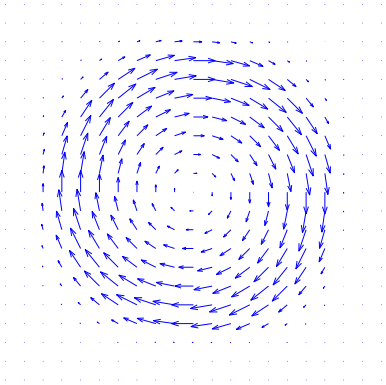

Figure 3: An example curl detector mask of size 21×21 pixels, and radius of 9 pixels.

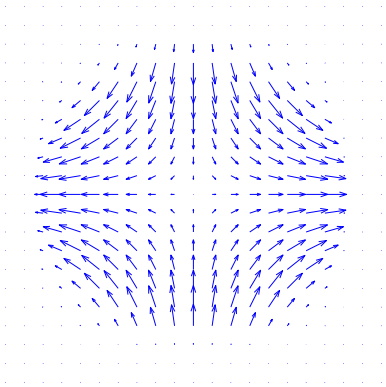

Figure 4: An example deformation detector mask of size 21×21 pixels, radius of 9 pixels, and axis of maximum extension tilted at $\mu = 0°$.

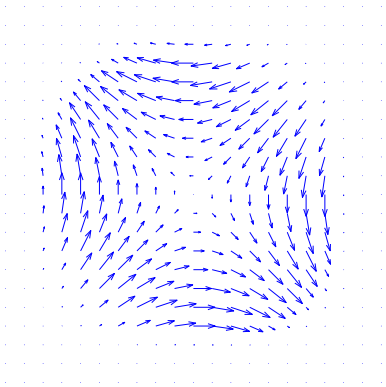

Figure 5: An example deformation detector mask of size 21×21 pixels, radius of 9 pixels, and axis of maximum extension tilted at $\mu = 45°$.

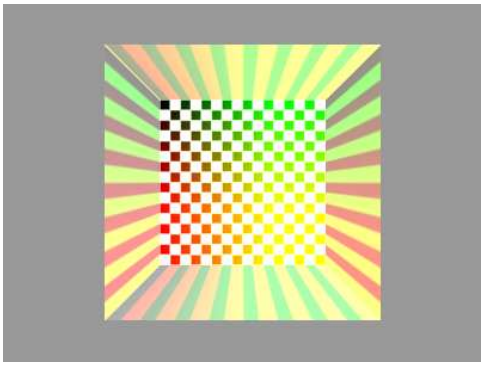

Figure 6: The synthetic scene viewed from a camera at $(0, 0, 0)$ in 3D space. It consists of an open ended cube with a patterned internal surface. The 2D image has dimensions 400×300 pixels, with the cube centred at $(0, 0, -100)$

3. EXPERIMENTAL RESULTS

This section outlines some of the results of our filtering technique when used on synthetic motion fields. The purpose of these experiments is to verify that the filter designs perform as expected over synthetic data. This should be done before applying the filters to real images that might contain only partial information or are noisy.

The environment in this case comprises a hollow textured cube with one of its sides removed revealing its internal surfaces, and a textureless background at 'infinity'. The cube has dimensions of $50 \times 50 \times 50$ pixels and its centre at $(0, 0, -100)$ in the 3D space. Figure 6 shows a view of the cube from the initial camera position. The camera is centred at $(0, 0, 0)$, has a field of view (FOV) of $60°$, and an orientation of $(0, -180, 0)$ corresponding to pitch, yaw, and roll respectively. The texture on the inside of the cube was intended for optical flow determination, but for the following examples the ground truth motion field is used. Note that all the filter outputs in the following sections are in grey-scale: white areas denote maximally positive values and black areas denote maximally negative ones.

3.1 Zooming Camera

Figure 7 shows the motion field caused by forward camera translation, or zoom, superimposed onto the new object image. In this case, the orientation of the camera does not change with the camera axis remaining perpendicular to the back wall of the cube.

3.1.1 Divergence

The 400×300 vector field of Figure 7 is convolved with a filter mask similar to that of Figure 2, but with radius $= 40$, to produce the response depicted in Figure 8. It is clear that the divergence caused by the approaching cube is detected by the filtering. Of particular interest is the different magnitudes of divergence detected on the inner side walls of the cube compared with the back wall. The bright white of the side walls indicates there is greater divergence there than on the back wall. This result is expected since the side walls are closer to the camera and hence have greater

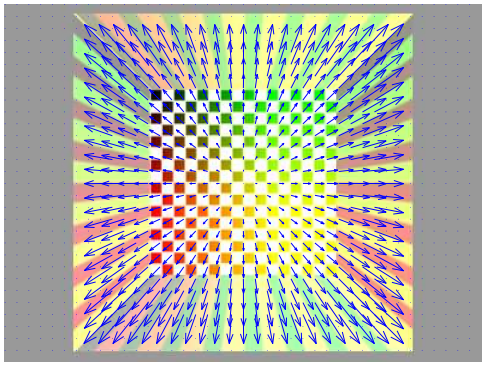

Figure 7: The camera has zoomed in toward the original cube depicted in Figure 6. There has only been forward motion from $(0, 0, 0)$ to $(0, 0, -14)$. That is, no camera rotation or lateral translation.

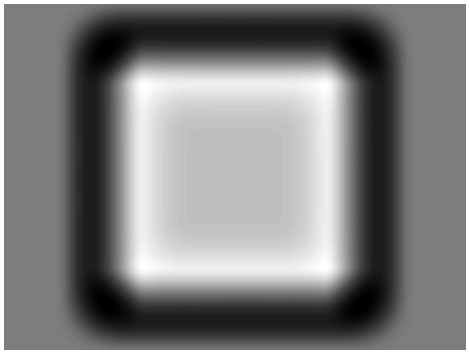

Figure 8: The output of correlating the zooming camera motion field depicted in Figure 7, with a divergence detector similar to that of Figure 2 except with a radius of 40 pixels. Notice that the vertical and horizontal walls of the cube responded more strongly than the back wall.

apparent velocity.

3.1.2 Curl

The response of the curl filter, as depicted in Figure 9, is mostly devoid of any features. This is expected since the pure dilatation in this particular motion field contains no rotational component; divergence and curl are orthogonal to one another. However, there are some responses near the corners of our cube image. These are false positives and investigation needs to be done into differentiating useful responses from extraneous ones.

3.1.3 Deformation

The motion of the camera results in a shear distortion of the apparent shape of the cube's side walls in addition to its isotropic dilatation. The projections of the 3D points close to the camera experience more shearing effect than those far away. We observe this distortion being detected by the deformation masks in Figures 10 and 11. The stretching of the horizontal sides upward and downward is detected

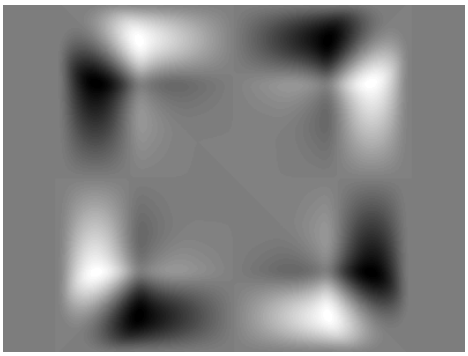

Figure 9: In contrast to Figure 8, there is little detection of curl in the zooming camera motion field. There are some edge effects but no obvious features. This is as expected since the curl and divergence masks are orthogonal to one another.

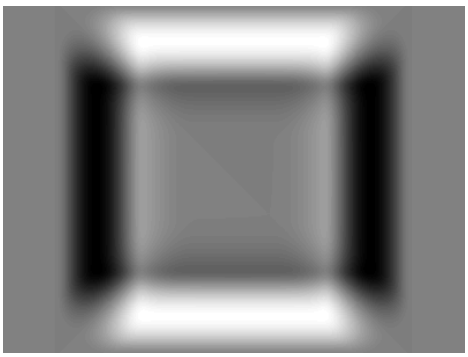

Figure 10: The output of the deformation detector on the diverging vector field with an axis of maximal expansion at 0°. Notice that the walls of the cube, except the back wall, return some correlation to the deformation mask reflecting the slight deformation on those faces.

by the symmetric mask, while the stretching of the corners is picked up by the anti-symmetric mask. The response of the deformation filtering provides us information about the surface orientations in the 3D scene since vertical and horizontal shearing will be detected by a deformation mask with corresponding axes of expansion and contraction.

3.2 Rolling Camera

To test the curl detector a rolling camera motion was introduced to the scene of Figure 6. The camera, from an initial orientation of $(0, -180, 0)$ was rotated by 10°, to $(0, -180, 10)$. The resultant object image and motion vector field is shown in Figure 12. Notice that the rotation is in a clockwise direction, the same as the curl basis vector field shown in Figure 3.

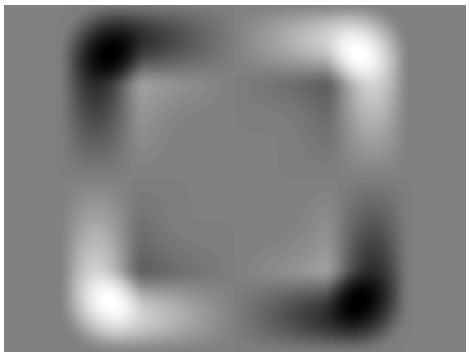

Figure 11: The output of the deformation detector on the diverging vector field with an axis of maximal expansion at $45°$.

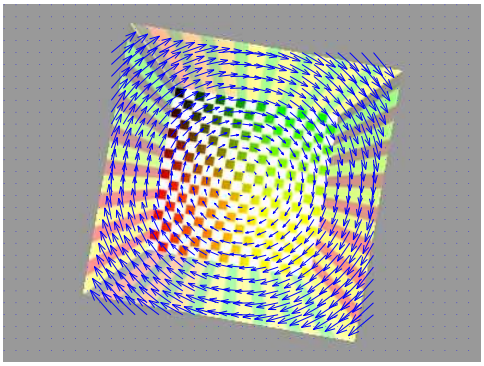

Figure 12: The motion field as a result of the change in camera orientation by $10°$ clockwise from Figure 6.

3.2.1 *Divergence*

Similar to the response of the curl mask over the diverging motion field, there is little correlation between the divergence mask and the rotating vector field of Figure 12. The subsequent filter output in Figure 13 shows few features, although there are still false responses similar to those in Figure 9.

3.2.2 *Curl*

The rotation of the camera affects all parts of the cube image equally and is illustrated by the uniform response of the curl filter over the entire cube's image pixels (Figure 14). The dark negative band surrounding the bright white response is a consequence of the correlation procedure where the convolution mask is only partially over the region of interest. Removal of these dark bands is desirable and might be achieved by pre-processing or segmentation of the motion field.

3.2.3 *Deformation*

As stated earlier, deformation describes the distortion of image shape along an axis of expansion and an axis of contraction perpendicular to the first axis. However, the apparent area of the object surface remains unchanged. For

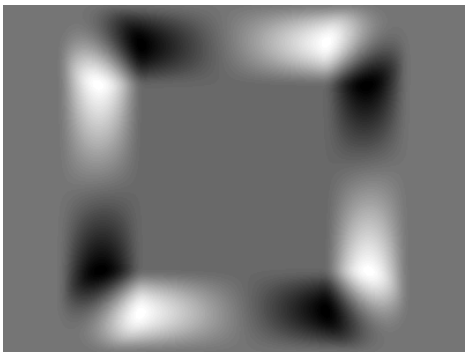

Figure 13: Similar to the output in Figure 9, there is no significant correlation between the divergence mask and the rolling camera motion field.

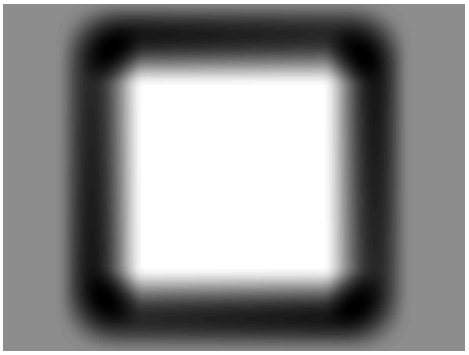

Figure 14: The vorticity caused by the change in camera orientation is detected strongly by our filters. Notice that due to the absence of texture in the background, there is no motion field outside the image dimensions of the cube.

the motion field produced by rotational ego-motion, the deformation masks have produced outputs similar to those for the zooming camera. A comparison of Figure 10 and 16, and Figure 11 and 15 suggests that there is a relationship between the magnitude of deformation and the other two differential invariants of motion. This relationship perhaps comes from the similarity of terms in Equations 5 to 7.

3.3 Translation and Noise

Given that optical flow data is rarely noise free and that objects are rarely moving directly along the camera axis, the promising results of the previous two sections are exaggerated outcomes. The motion field we analyse in this section includes some Gaussian noise and lateral camera translation.

This new optical flow field is generated by adding filter masks similar to those in Figures 2 to 5 to an otherwise featureless field and introducing normally distributed noise and tangential translation. The resultant field is depicted in Figure 18. The original uncorrupted vector field, shown in Figure 17, has four synthetic features, a focus of expansion, a centre of rotation, and two deformation fields all with radius = 40. The convolution masks used for the filtering are

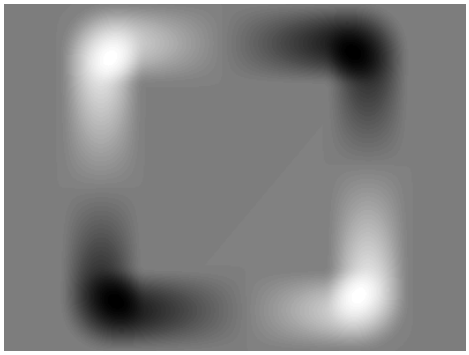

Figure 15: Using the same deformation mask as in Figure 10, and ignoring edge effects, there is little useful correlation with the rotating motion field.

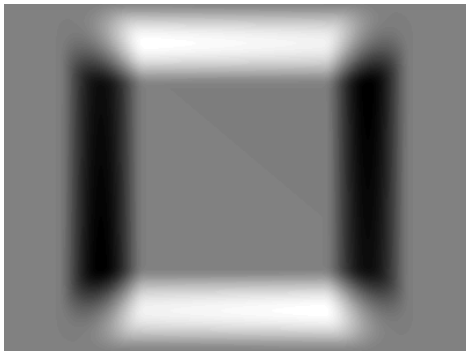

Figure 16: Using the same deformation mask as in Figure 11, the filter response is very similar to that of Figure 10. Further investigation needs to be done into why this is so.

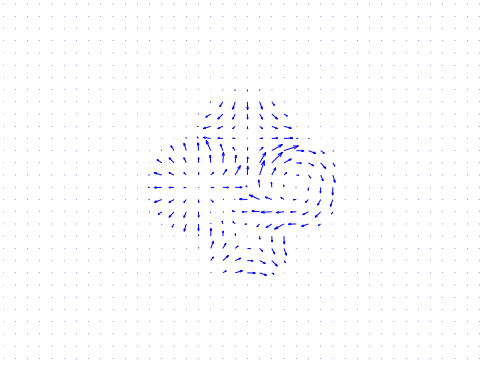

Figure 17: The above 400×300 vector field contains four synthetic features: a focus of expansion at $(160, 150)$, a centre of rotation at $(240, 150)$, and two regions of deformation located at $(200, 110)$ and $(200, 190)$.

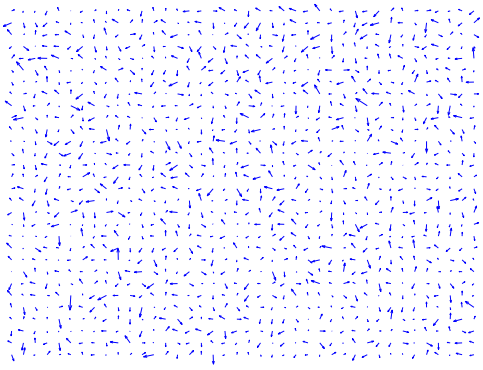

Figure 18: The above is the same vector field as in Figure 17 but corrupted by Gaussian noise and superposed with a global translation component (top right to bottom left). An astute observer can still resolve the four features.

of similar size.

The filter responses in Figures 19 to 22 demonstrate that the superimposed noise and global translation had only a limited effect on the feature extraction. Possibly, the effect of global translation is transparent to the filtering process since the heavy correlation of the filter mask in some regions is offset by the heavy negative correlation in other regions. The robustness in the face of a noisy vector field as in Figure 18, is a promising result given that existing optical flow algorithms do not return perfect motion fields. Further investigation needs to be done into quantifying the robustness of this filtering method for invariants extraction in general.

4. CONCLUSION

Although there has been much work done in optical flow determination, as well as research into the usefulness of the geometrical properties of optical flow (such as divergence, curl, and deformation), there is little work done on how to link the two together. Apart from Cipolla's closed-curved

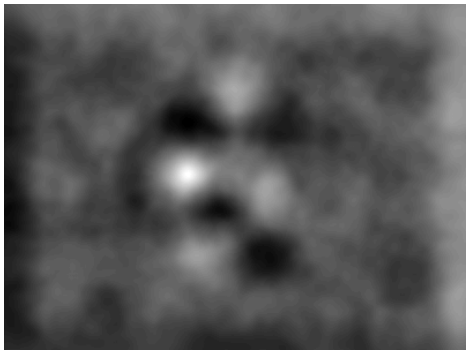

Figure 19: The high response (white shading) of the divergence mask at $(160, 150)$**, reveals the location the focus of expansion quite clearly.**

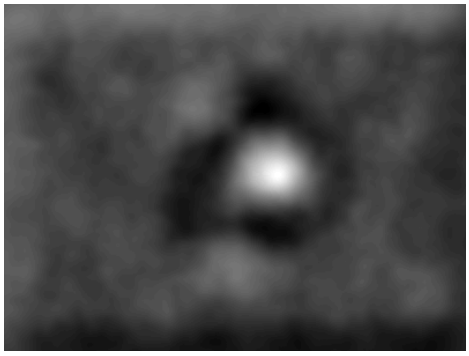

Figure 20: The response of the curl mask reveals the centre of rotation at $(240, 150)$**.**

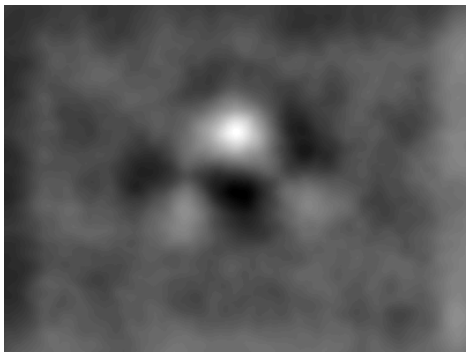

Figure 21: The deformation detector shows the deformation field at $(200, 110)$**.**

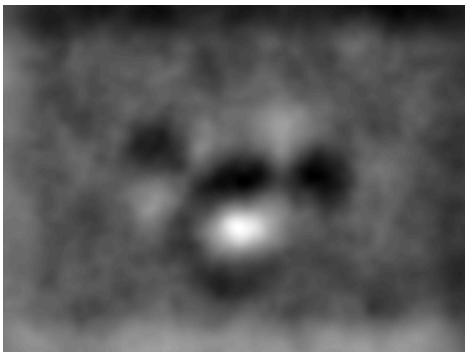

Figure 22: The anti-symmetric deformation detector shows the deformation field at $(200, 190)$**.**

tracking method [10], there are no other well-documented methods for deriving the differential invariants of image velocity. Presented in this paper is a new way to determine the differential invariants of optical flow fields which uses input optical flow data from existing algorithms. It is hypothesized that a simple filter correlation technique similar to that of signal deconstruction can be used for this purpose. The preliminary experimentation presented here using small images and filters has produced promising results, particularly with respect to noise robustness, but many extensions remain to be investigated.

Ultimately, rather than using the traditional methodology of deducing optical flow, reconstructing the entire 3D scene, and interpreting this resultant scene for useful information; intensive image processing is saved by deriving useful information directly from the image motion field through the extraction of differential invariants.

5. FURTHER CONSIDERATIONS

As mentioned throughout this paper, there are several extensions that can be made to our method. In particular, it is desirable to design a bank of filters such that each filter type (divergence, curl, or deformation) is orthogonal to every other type. This would allow a vector field to be truly broken down into its constituent components.

Further investigation needs to be done into the effects of noise on the filter responses. Perhaps a different envelope shape or image segmentation will be necessary to improve the accuracy of filter outputs.

The main goal is to extract quantitatively accurate values for divergence, curl, and deformation such that slant and tilt values for scene points can be calculated and used for partial 3D scene reconstruction.

6. ACKNOWLEDGEMENTS

This work is supported by an Australian Postgraduate Award as well as an *ad hoc* scholarship from the School of Computer Science & Software Engineering at the University of Western Australia. Thanks go to my supervisor Peter Kovesi for his help and supervision of this work.

7. REFERENCES

[1] Nicola Ancona and Tomaso Poggio. Optical flow from 1D correlation: Application to a simple time-to-crash detector. In *Proceedings of the Fourth International Conference on Computer Vision*, May 1993.

[2] Alireza Bab-Hadiashar and David Suter. Robust optic flow computation. *International Journal of Computer Vision*, 29(1):59–77, August 1998.

[3] J. L. Barron, S. S. Beauchemin, and D. J. Fleet. On optical flow. In I. Plander, editor, *6th International Conference on Artificial Intelligence and Information Control Systems of Robots (AIICSR)*, pages 3–14, Bratislava, Slovakia, September 1994. World Scientific.

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

[5] S. S. Beauchemin and J. L. Barron. The computation of optical flow. *ACM Computing Surveys*, 27(3):433–467, 1995.

[6] Steven S. Beauchemin, Ruzena Bajcsy, and John L. Barron. Recent advances in motion understanding. In Reinhard Klette, Georgy Gimel'farb, and Ramakrishna Kakarda, editors, *International Conference on Image and Vision Computing (IVCNZ98)*, pages 29–37, The University of Auckland, November 1998.

[7] Ted Camus. Real-time quantized optical flow. *Real-Time Imaging*, 3:71–86, 1997.

[8] Roberto Cipolla. *Active Visual Inference of Surface Shape*, volume 1016 of *Lecture Notes in Computer Science*. Springer-Verlag, Heidelberg, Germany, 1995.

[9] Roberto Cipolla and Andrew Blake. Surface orientation and time to contact from image divergence and deformation. *Lecture Notes in Computer Science*, 588:187–202, 1992.

[10] Roberto Cipolla and Andrew Blake. Image divergence and deformation from closed curves. *International Journal of Robotics Research*, 16(1):77–96, 1997.

[11] David J. Fleet. *Measurement of Image Velocity*. Kluwer Academic, 1992.

[12] Berthold Klaus Paul Horn. *Robot Vision*. MIT Electrical Engineering and Computer Science Series. MIT Press, 1986.

[13] J. J. Koenderink and A. J. van Doorn. Invariant properties of the motion parallax field due to the movement of rigid bodies relative to an observer. *Optica Acta*, 22(9):773–791, January 1975.

[14] J. J. Koenderink and A. J. van Doorn. How an ambulant observer can construct a model of the environment from the geometrical structure of the visual inflow. *Kybernetik*, pages 224–247, 1978.

[15] Randal C. Nelson and John (Yiannis) Aloimonos. Obstacle avoidance using flow field divergence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(10):1102–1106, October 1989.

[16] P. Sobey and M. V. Srinivasan. Measurement of optical flow by a generalized gradient scheme. *Optical Society of America*, 8(9):1488–1498, September 1991.

[17] P. J. Sobey, M. G. Nagle, Y. V. Venkatesh, and M. V. Srinivasan. Measurement of complex optical flow with use of an augmented generalized gradient scheme. *Optical Society of America*, 11(11):2787–2798, November 1994.

[18] Changming Sun. Fast optical flow using 3D shortest path techniques. *Image and Vision Computing*, 20(13–14):981–991, December 2002.