Q-warping: Direct Computation of Quadratic Reference Surfaces

A. Shashua

shashua@cs.huji.ac.il

Hebrew University
Institute of Computer Science
Jerusalem, Israel

Y. Wexler

wexler@cs.umd.edu

Center for Automation Research
University of Maryland
College Park, MD 20742-3275

Abstract

We consider the problem of wrapping around an object, of which two views are available, a reference surface and recovering the resulting parametric flow using direct computations (via spatio-temporal derivatives). The well known examples are affine flow models and 8-parameter flow models — both describing a flow field of a planar reference surface. We extend those classic flow models to deal with a Quadric reference surface and work out the explicit parametric form of the flow field. As a result we derive a simple warping algorithm that maps between two views and leaves a residual flow proportional to the 3D deviation of the surface from a virtual quadric surface. The applications include image morphing, model building, image stabilization, and disparate view correspondence.

1 Introduction

Visual motion estimation is an essential ingredient for applications across the hierarchy of visual levels. Applications range from 3D model building from multiple views, segmentation of the scene based on motion cues, object tracking, image stabilization, 3D representation of the scene with respect to a reference surface, recovery of camera ego-motion from image sequences, visual recognition by alignment and by learning networks methods, and morphing applications in vision and graphics — and the list is by no means complete.

The image flow field induced by the camera (or scene) motion is a product of the 3D structure of the scene, the 3D camera motion parameters and its intrinsic parameters. While efficient factorization of the flow into structure and motion is an active area of research, in many cases an implicit parametric form of the optic flow is sufficient, say for motion segmentation, image stabilization, coarse correspondence for model building, and for establishing reference surfaces for 3D scene representation. To that end, a hierarchy of parametric flow models have been developed in the past (whose most elegant description can be found in [3]) starting from pure global translation, image plane rotation, 2D affine, and 2D homography (8-parameter flow, also known as quadratic flow). These models have been used extensively and have been estimated directly from image spatio-temporal derivatives (known as *direct estimation*) using coarse-to-fine estimation via Laplacian (or Gaussian, or wavelets) pyramids. These methods search for the best (parametric) flow model out of the family of constrained flows (described above) that minimizes the square of change of image intensities (SSD) over the whole image — thus gaining robustness due to very highly over-constrained linear systems (each pixel contributes a linear constraint).

However, these models are applicable to scenes that are approximately planar or have small variations in depth, relative to the distance from the camera. The residual flow, followed by the nominal planar warp, is proportional to the depth variation from the scene to the virtual planar surface. In other words, the virtual surface plays a role of a *reference surface* and the residual flow represents the scene structure relative to the reference surface. In many of the applications mentioned above one would like the reference surface to approximate the structure of the scene. For example, a planar scene with a small number of objects protruding from the plane (such as a moving vehicle) is ideal for an affine or homography flow that will stabilize the plane and thereby enhance the position of the protruding objects. Small residual flow is also convenient for establishing correspondence (alignment) between disparate views. Since the nominal flow (corresponding to the parametric flow model) is highly over-constrained large image distances can be tolerated, and if the residual flow is small then a second round of optic flow (now unconstrained) can handle the remaining displacements.

These examples naturally suggest considering higher-order parametric flows in order to account for non-planar virtual reference surfaces. For example, by placing a virtual quadric surface (allowing for all degenerate forms including a plane) on the object would give rise to a smaller residual flow and in more general circumstances. Consider for example Fig. 3 displaying two widely separated views of a face. Notice the effect of a planar warp field, compared to the effect of a Quadric-based flow field. The image warped by the nominal flow is much less distorted and the residual flow is much smaller.

The idea of extending the planar models to quadric models was originally suggested in [14] but in the context of discrete motion. The quadric reference surface was recovered using explicit point matches, including the computation of the epipolar geometry, whereas here we wish to establish a "quadric warp field" using infinitesimal motion models and direct estimation. In [14] special attention was payed on how to overcome the multiple solution problem since a general ray from the camera meets a quadric twice, thus for every pixel in the first image there would be two candidate projections in the second image. Also special attention was payed to the type of image measurements that are sufficient for a solution (point matches only, points and an outline conic, see also [6]).

In this paper we introduce the derivation of a quadric-based nominal flow field, we call *Q-warping*, using the infinitesimal motion model and direct estimation. The multiple solution problem addressed by [14] via "opacity" constraint is approached differently here. Instead of using an opacity assumption, which is problematic to enforce in a parametric flow representation, we enforce the family of quadrics to include the center of projection of the first camera. We show that the assumption does not reduce the generality of the approach due to existence of hyperboloids of two-sheet (one sheet includes the camera center and the other wraps around the object), and that planar surfaces are included in this model in a general manner, i.e., the plane may be generally located in space. Therefore, our model extends the hierarchy of direct estimation parametric models of [3] without sacrificing "backward compatibility".

2 Background: Small Motion and Parametric Flow

The parametric flow models are based on combining three elements: (i) infinitesimal motion model, (ii) planar surface assumption being substituted into the motion model, and (iii) the parametric flow is integrated with the "constant brightness constraint". We will describe these elements in detail below.

2.1 Small Motion Model

We describe below a compact form of the Longuett-Higgins and Prazdny motion model [11]. Let $p = [x, y]^t = [X/Z, Y/Z]^t$ where $P = [X, Y, Z]^t$ is a world point in the coordinate system of the first (calibrated) camera and p is it's corresponding image point. Let P' be the coordinates of the same world point in the second camera coordinate frame. Since the camera motion is rigid, we have $P' = RP + t$ where R and t are the rotation and translation between the coordinate frames.

The rotation matrix R can be written as

$$R = I * cos\phi + (1 - cos\phi)ww^t + sin\phi * [\omega]_\times$$

where ω is a unit vector representing the screw axis, $[\omega]_x$ is the skew-symmetric matrix of vector products, i.e., $[\omega]_x v = \omega \times v$ for all vectors v, and ϕ is the angle of rotation around the screw axis. When ϕ is small, $cos\phi \rightarrow 1$ and $sin\phi \rightarrow \phi$, and in turn $R = I + [\omega]_\times$ where the magnitude of ω is the angle of rotation. Given the instantaneous rotation, the instantaneous motion of P is:

$$\begin{aligned} \dot{P} &= \frac{dP}{dt} \approx P' - P = RP + t - P = \\ &= (I + [\omega]_\times)P + t - P = [\omega]_\times P + t \qquad (1) \end{aligned}$$

Let $[u, v]^t$denote the image velocity from p to p'. We use $\dot{X}$ and $\dot{Z}$, as defined by (1) to get the following:

$$\begin{aligned} u &= \frac{dx}{dt} = \frac{d}{dt}(\frac{X}{Z}) = \\ &= \frac{\dot{X}Z - X\dot{Z}}{Z^2} = \frac{1}{Z}(\dot{X} - x\dot{Z}) = \\ &= \frac{1}{Z}[1,\ 0, -x]^t([\omega]_\times P + t) \end{aligned}$$

and $v = \frac{dy}{dt}$ is similarly derived. To summarize we have,

$$\begin{aligned} u &= \frac{1}{Z}s_1^t t + s_1^t[\omega]_\times p \\ v &= \frac{1}{Z}s_2^t t + s_2^t[\omega]_\times p \qquad (2) \end{aligned}$$

where $s_1 = [1, 0, -x]^t$ and $s_2 = [0, 1, -y]^t$.

2.2 Direct Estimation Equation

Assume the brightness constancy assumption,

$$I'(x, y) = I(x - u, y - v),$$

where $I(x, y), I'(x, y)$ are the observed grey-scale images at two successive time frames. Since the displacement u, v are assumed to be small (infinitesimal motion assumption), then the equation above can be simplified through the truncated (first-order) Taylor series expansion of $I(x, y)$ to what is known as the "constant brightness equation" [8]:

$$uI_x + vI_y + I_t = 0$$

where I_x, I_y are the x, y spatial derivatives, respectively, and $I_t = I'(x, y) - I(x, y)$ is the temporal image derivative. By substituting u, v with equations 2 we obtain a linear constraint on the flow u, v as a function of the spatio-temporal derivatives, the camera motion parameters ω, t and the depth variable Z:

$$\frac{1}{Z}(I_\times s_1 + I_y s_2)^t t + (I_x s_1 + I_y s_2)^t [\omega]_\times p + I_t = 0,$$

which after simplification becomes [9]:

$$\frac{1}{Z} s^t t + s^t [\omega]_\times p + I_t = 0 \qquad (3)$$

where

$$s = \begin{bmatrix} I_x \\ I_y \\ -xI_x - yI_y \end{bmatrix}.$$

2.3 Parametric Model: Planar Case

One can eliminate the parameter Z from equation 3 by assuming that the scene is planar [1], i.e., there exist scalars A,B,C such that $AX + BY + CZ = 1$ for all points $[X, Y, Z]^t$. By definition of p we have $X = xZ$, $Y = yZ$ and so we can rewrite it as $\frac{1}{Z} = Ax + By + C$. Substituting this in (2) yields:

$$\begin{aligned} u &= (Ax + By + C)s_1^t t + s_1^t [\omega]_\times p \\ v &= (Ax + By + C)s_2^t t + s_2^t [\omega]_\times p \end{aligned}$$

Written more explicitly, let $t = [\tau_1, \tau_2, \tau_3]^t$ and $\omega = [\alpha, \beta, \gamma]^t$ we get:

$$\begin{aligned} u &= (A\tau_1 - C\tau_3)x + (B\tau_1 - \gamma)y + C\tau_1 + \beta - \\ &\quad (B\tau_3 + \alpha)xy + (\beta - A\tau_3)x^2 \\ v &= (A\tau_2 + \gamma)x + (B\tau_2 - C\tau_3)y + C\tau_2 - \alpha + \\ &\quad (\beta - A\tau_3)xy - (B\tau_3 + \alpha)y^2 \end{aligned}$$

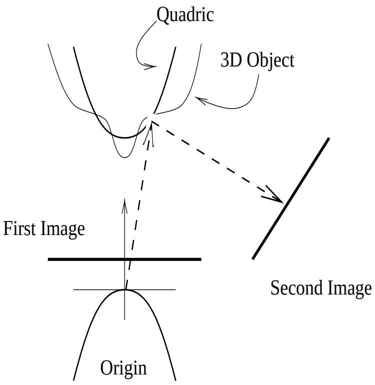


Figure 1: Q-warping fits a quadric around a general object, however the family of quadrics must contain the origin (the first camera center). A hyperboloid of two sheets meets this requirement by having one sheet coincide with the origin and the other sheet wrap around the object. All rays from the first camera intersect the quadric uniquely and the projection of the intersection point onto the second view is the result of the Q-warping flow. The residual flow is thus proportional to the deviation of the physical surface from the virtual quadric.

The terms above can be collected to give the 8-parameter flow model used for estimating the instantaneous motion of a plane:

$$\begin{aligned} u &= ax + by + c + gxy + hx^2 \\ v &= dx + ey + f + hxy + gy^2 \end{aligned} \qquad (4)$$

The direct estimation readily follows by substituting the above in $uI_x + vI_y + I_t = 0$ we obtain a linear constraint on the parameters $a, b, ..., h$. Every pixel with a non-vanishing gradient contributes one linear constraint for 8 unknowns, thus, making a highly over-constraint least-squares system for solving the warping function 4.

3 The Quadric Flow: Q-warping

Consider the family of quadric surfaces that contains the origin $[0, 0, 0]$, i.e., the center of projection of the first camera:

$$\begin{aligned} AX + BY + Z + DXY + EXZ + FYZ + \\ +GX^2 + HY^2 + KZ^2 = 0 \end{aligned} \qquad (5)$$

Note that we have normalized the coefficients assuming that the coefficient of Z is non-vanishing, one could choose other forms of normalization.

The reason we include the origin is to have a single intersection between the optical rays emanating from the first camera and the quadric surface. A single intersection is a necessary condition for obtaining a warping function. It is important to note that the inclusion of the origin does not limit the generality of the quadric because quadrics can break apart into two pieces, known as the hyperboloid of two sheets. Thus, one sheet will include the origin and the other sheet will wrap around the object. The location and shape of the sheet (parabolic, elliptic, and degenerate forms like cylinders and cones) will be determined by the image measurements of spatio-temporal derivatives (see Fig. 1). What is also important to show is that the inclusion of the origin is not a constraint that is carried to the degenerate form of a planar surface. In other words, a planar warping function, corresponding to any general position of a plane, should be a particular case of the Q-warping function — otherwise we will not be able to include Q-warping in the hierarchy of parametric models. We will show later that in case of planar objects, the quadric breaks down into two planes, one coincides with the physical plane in the scene and the other is the plane $Z = 0$. Taken together, there is no loss of generality by having the origin live inside the family of quadrics.

Using $X = xZ$, $Y = yZ$ and dividing by Z^2 and rearranging terms, we get:

$$\frac{1}{Z} = \frac{Dxy + Ex + Fy + Gx^2 + Hy^2 + K}{Ax + By + 1}$$

By substituting Z in equations 2 we obtain a parametric flow model (with 17 distinct parameters), which is our Q-warping function:

$$u = \frac{\alpha(x, y, a, ..., p)}{Ax + By + 1} \qquad (6)$$

$$v = \frac{\beta(x, y, a, ..., p)}{Ax + By + 1} \qquad (7)$$

where,
$\alpha(\cdot) = ax+by+c+dxy+ex^2+fy^2+gyx^2+hxy^2+px^3$
$\beta(\cdot) = jx+ky+l+mxy+nx^2+oy^2+pyx^2+gxy^2+hy^3$

And:

$$\begin{array}{ll}
a = \beta A - K\tau_3 + E\tau_1 & b = \beta B - \gamma + F\tau_1 \\
c = K\tau_1 + \beta & d = -\alpha - D\tau_1 - \gamma A - F\tau_3 \\
e = \beta + G\tau_1 - E\tau_3 & f = H\tau_1 - \gamma B \\
g = \beta B - D\tau_3 - A\alpha & h = -H\tau_3 - B\alpha \\
j = \gamma - A\alpha + E\tau_2 & k = F\tau_2 - K\tau_3 - B\alpha \\
l = K\tau_2 - \alpha & m = B\gamma + \beta + D\tau_2 - E\tau_3 \\
n = G\tau_2 + A\gamma & o = H\tau_2 - \alpha - F\tau_3 \\
p = \beta A - G\tau_3 &
\end{array}$$

Before we continue to the direct estimation equation, consider the case of a planar object. We would like to show the following:

Proposition 1 *The Q-warping flow model includes as a particular case the planar parametric model of equations 4.*

Proof: Consider the quadric: $EXZ + FYZ + KZ^2 + Z = 0$. By dividing by Z^2 followed by substitution in equations 2, we obtain:

$$\begin{aligned} u &= ax + by + c + dxy + ex^2 \\ v &= jx + ky + l + mxy + oy^2 \end{aligned}$$

where $d = o = -\alpha - F\tau_3$ and $e = m = \beta - E\tau_3$. □

In order to obtain a direct estimation using spatio-temporal derivatives, we multiply both sides of the Q-warping equations by $Ax + By + 1$ and obtain:

$$\alpha(x, y, a, ..., p)I_x + \beta(x, y, a, ..., p)I_y + (Ax + By + 1)I_t = 0. \qquad (8)$$

This is a linear equation in $A, B, a, b, ..., p$ per pixel with non-vanishing gradients. The least-squares estimation requires some care which will be described next. Also note that the motion model has 17 parameters, yet the minimal number of parameters required for describing a moving quadric passing through the camera center is $14 = 6 + 8$, where 6 comes from parameters of rotation and translation and 8 comes from the number of parameters representing the quadric (passing through the origin). Therefore, the 17 parameters must satisfy algebraic constraints, i.e., not every set of 17 numbers is admissible. However, this is a topic which will not be covered in the scope of this paper.

4 Issues of Implementation

The direct estimation equation 8 in the parameters $a, ..., p, A, B$ holds only for infinitesimal motion — both due to the motion model assumed and to the application of the constant brightness equation. In other words, the parametric flow model is a first-order approximation and, thus, the implementation framework must include a Newton iterative refinement, of the style suggested in [2, 3, 4, 12]. The iterative refinement procedure for parametric models up to the planar model are typically integrated with a coarse-to-fine framework (Laplacian pyramid [5], for example) with an image warping step — will be discussed later. In addition, the Q-warping model requires special care on how the iterative refinement should be defined. We will start with the latter issue and then proceed with the coarse-to-fine implementation issues.

4.1 Iterative Refinement

The iterative refinement process [2, 4] (and described in more detail later) gradually brings the two original images closer to each other, such that in the ideal case of constant brightness and a quadric surface, $I_t \rightarrow 0$ at the limit. Considering the estimation equation 8, the diminishing I_t is a serious problem because the coefficients A, B become under-determined. In other words, as we get closer to the solution, our system of equations gets increasingly unstable numerically.

We adopt the line of approach described in [7, 15] which is to rewrite the direct estimation equation 8 as a function of the final output flow instead of the incremental flow, shown next. Let $u(x,y), v(x,y)$ be the final flow (describing the displacement field between the original two images $I(x,y), I'(x,y)$ as a function of a quadric model) described parametrically in eqns. 6 and 7.

Let $\tilde{u}(x,y), \tilde{v}(x,y)$ be the flow field established in the last iteration (the initial guess for the current iteration). The incremental flow is defined by

$$\begin{aligned} \Delta u &= u - \tilde{u} \\ \Delta v &= v - \tilde{v} \end{aligned}$$

and satisfies the constant brightness equation:

$$\Delta u I_x + \Delta v I_y + I_t = 0.$$

After substitution we obtain the new direct estimation equation for the parameters $A, B, a, ..., p$ below:

$$\alpha(\cdot)I_x + \beta(\cdot)I_y + (Ax+By+1)(I_t - \tilde{u}I_x - \tilde{v}I_y) = 0. \quad (9)$$

Initially, $\tilde{u} = \tilde{v} = 0$. As the iterations proceed $\tilde{u}, \tilde{v}$ approach the desired flow u, v (eqns. 6,7).

Another issue worth noting regarding the iterative refinement concerns the correction of the statistical bias caused by the product with $Ax+By+1$ which was necessary for linearizing the Q-warp function to obtain equation 9. The product with the term $Ax + By = 1$ can be viewed as a weight, thus making the overall least-squares process (by collecting all the measurements per pixel) a weighted least-squares. The weights however may cause a bias in the solution. One way to "undo" the bias, is to "renormalize" by dividing equation 9 by the term $\tilde{A}x + \tilde{B}x + 1$ where $\tilde{A}, \tilde{B}$ are the corresponding coefficients determined from the previous iteration. Taken together, the direct estimation equation has the form:

$$\frac{\alpha(\cdot)I_x + \beta(\cdot)I_y + (Ax + By + 1)(I_t - \tilde{u}I_x - \tilde{v}I_y)}{\tilde{A}x + \tilde{B}x + 1} = 0. \quad (10)$$

In each iteration, the system of equations for the parameters $a, ..., p, A, B$ is defined by minimizing the least squared error:

$$\begin{aligned} Err &= \sum_{x,y} = \frac{1}{\tilde{A}x + \tilde{B}x + 1}[\alpha(\cdot)I_x + \beta(\cdot)I_y \\ &+ (Ax + By + 1)(I_t - \tilde{u}I_x - \tilde{v}I_y)]^2 \quad (11) \end{aligned}$$

where the sum is over the entire image. The system of linear equations is obtained by setting the partial derivatives of (11) with respect to each of the parameters $a, ..., p, A, B$ to zero.

4.2 Alignment (warping) Steps

Iterative steps are implemented by image alignment, referred to as "warping" steps. After an initial estimate of motion field $u(x,y), v(x,y)$ is made (through the least-squares system that minimize equation 11), the second image is shifted towards the first to compensate for the estimated displacement. The motion estimation procedure is then repeated between the original first image and the shifted second image to obtain the new updated estimate of the parameters $a, ..., p, A, B$ (which in turn determine the new updated motion field $u(x,y), v(x,y)$). These shift and estimate steps are iterated to bring the second image into alignment with the first, thereby progressively reducing the frame-to-frame displacement.

Let u_k, v_k be the motion field estimate after the k'th iteration of the alignment process. Let u_o, v_o be the a priori estimate before the analysis begins — typically we assume that $u_o = v_o = 0$. Steps of the alignment procedure during the k'th iteration are as follows:

1. The original second image $I'(x,y)$ is warped towards the first image $I(x,y)$ in accordance with the displacement u_{k-1}, v_{k-1} obtained on the previous iteration:

   $$I_{k-1}(x,y) = I'(x - u_{k-1}(x,y), y - v_{k-1}(x,y)).$$

2. The motion estimator (minimizing equation 11) is applied to the original first image $I(x,y)$ and the shifted second image $I_{k-1}(x,y)$ to obtain the flow field u_k, v_k. In other words, the new $I_t(x,y)$ is $I_{k-1}(x,y) - I(x,y)$.

4.3 Hierarchical Motion Estimation

The above estimation algorithm is meaningful only when the frame-to-frame displacements are a fraction of a pixel so that the first-order order term of the Taylor series is dominant. The range of displacements can be extended to large displacements by implementing the procedure within a multiresolution (pyramid) structure [5, 13].

A multiresolution pyramid is a sequence of copies of the original image in which both resolution and sample density are reduced by a power of 2. In a Gaussian pyramid, let G_l be the l'th pyramid level for image $I(x,y)$, where $G_o = I(x,y)$, obtained by convolving the $l-1$ level with a small kernel filter w followed by sub-sampling by 2 in both x and y, i.e., every other row and column are discarded.

A Gaussian (or Laplacian) pyramid is constructed for each of the original frames $I(x,y)$ and $I'(x,y)$. The motion analysis begins at the lowest resolution level of the pyramids. The sample distance at level l is 2^l times that of the original image, which in turn means that correspondingly larger image displacements can be estimated. At each successive iteration, the shift and estimate steps (described above) are performed on the next higher resolution level $l-1$. Thus, if level l is processed at iteration k with estimated flow-field u_k, v_k, then the warp step is applied to the pyramid level $l-1$ with the estimated flow $2u_k, 2v_k$ (twice the magnitude). Warping insures that the residual displacements remains less than a sample distance as the procedure moves to each higher resolution pyramid level until full resolution is reached.

Given u_k, v_k the estimated flow at level l of the pyramid, the flow $2u_k, 2v_k$ for the warp step at level $l-1$ can be estimated analytically as follows. Let $a^k, ..., p^k, A^k, B^k$ be the estimated parameters that determine the flow u_k, v_k. Instead of computing explicitly the flow u_k, v_k and expanding it down the pyramid, one can compute $a^{k+1}, ..., p^{k+1}, A^{k+1}, B^{k+1}$ that represent the flow $2u_k, 2v_k$:

$$\begin{array}{lll}
a^{k+1} = a^k & b^{k+1} = b^k & c^{k+1} = 2c^k \\
d^{k+1} = d^k/2 & e^{k+1} = e^k/2 & f^{k+1} = f^k/2 \\
g^{k+1} = g^k/4 & h^{k+1} = h^k/4 & p^{k+1} = p^k/4 \\
j^{k+1} = j^k & k^{k+1} = k^k & l^{k+1} = 2l^k \\
m^{k+1} = m^k/2 & n^{k+1} = n^k/2 & o^{k+1} = o^k/2 \\
A^{k+1} = A^k/2 & B^{k+1} = B^k/2 &
\end{array}$$

Thus, the estimated flow-field need not be represented explicitly only the estimated parameters $a, ..., p, A, B$ need to be represented during the process.

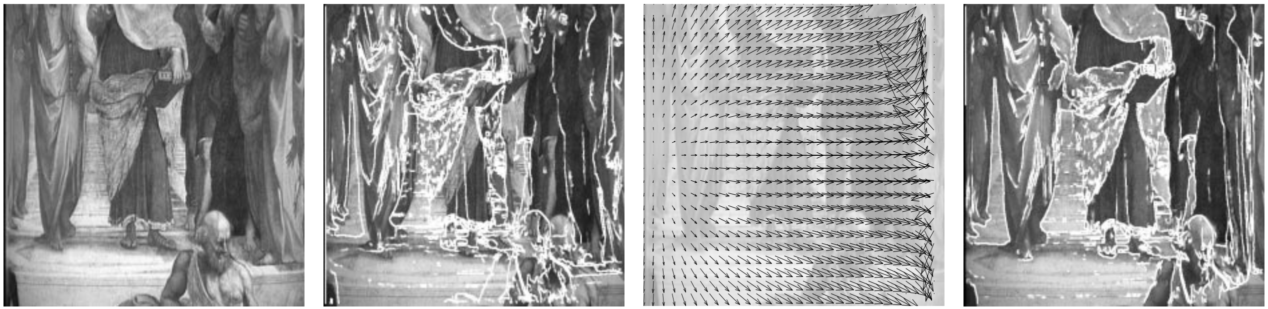
Finally, another implementation issue worth noting is that it is often recommended to apply low complexity motion models (pure translation, affine, etc.) to correspondingly low resolution levels of the pyramid. For example, when the lowest resolution level of the pyramid consists of a relatively small image, then the numerical support for a 17 parameter motion model would be rather weak. In that case a lower complexity model, like pure translation or affine would be more appropriate. Details on various schemes for moving across the pyramid structure with corresponding levels of motion model complexities can be found in [10].

5 Experiments

We have conducted a number of experiments both on quadrics and general objects. In our first example we have wrapped a poster on an approximately cylindrical surface. Here we expect the parametric flow recovered from the Q-warping function to match closely the true flow. Fig. 2 displays the results: the edges of the second view are overlayed on the first view in order to visualize the magnitude of displacements. The flow field recovered by our algorithm is displayed in Fig. 2c and the edges of the warped image overlayed on the second view shown in Fig. 2d match their true position up to sub-pixel accuracy. Note that this also demonstrates that the real surface need not contain the center of projection of the first camera because the Q-warping function can feet a hyperboloid of two sheets with one of the sheets wrapping around the physical surface.

We next apply Q-warping on a planar scene used for stabilization examples. Fig. 6 shows a sequence of images of a helicopter hovering above a planar scene. Typically, planar warp stabilizes the motion of the plane and thus enhances the region containing the helicopter. Q-warping applied to this sequence has reduced to a planar flow field (as shown in Proposition 1), thus illustrating the point that Q-warping is "backward" compatible — it will adjust the complexity of the warping function to the complexity of the scene.

The remaining examples are on general objects. Fig. 3 shows two disparate views of a face. One can see from the edge overlay that the distance between the views is fairly significant. Note the difference between the Q-warping and the planar warping. We have applied to versions of a planar warping. First, in the bottom row 4 matching points were selected (coming from an approximately planar configuration in space) and the 2D projective transformation determined by 4 matching points) was recovered. Note that the edges on the center of the face are closely aligned at the expense of the boundary curves (as the boundary curves away from the plane determined by the 4 matching points). Second, we applied the infinitesimal planar motion model (eqn. 4) in a direct estimation framework (row 3). Note that the planar warping has chosen a plane fitting the center of the face (where most of the strong gradients are) rotated around the vertical axis — the result is a distorted warped image due to the large deviation of the object from a planar surface. The Q-warping on the other hand has aligned the warped image with the second view, up-to a few pixels distance. Fig 4 demonstrates the application of Q-warping on another face — note that the warped



(a) The first view.

(b) Edges of the second view overlayed on the first.

(c) Optical flow induced by the quadric surface

(d) Edges of the warped (first) view overlayed on the second.

Figure 2: A poster was wrapped around a cylindrical surface. The recovered flow matches the true flow to sub-pixel accuracy.

image is closely aligned with the second view. Finally, Fig. 5 is an example of another general object, this time the statue of Venus. Note again that the overlayed edges of the warped image are closely aligned with the second view.

Taken together, the Q-warping algorithm generates a parametric flow field that performs well on general objects as well as on quadric surfaces (in the latter case the flow field is exact).

6 Summary

We have extended the parametric flow hierarchy to include flows induced by a virtual quadric. We have shown that the extension can be made feasible in the sense that the warping function is unique and includes planar warping as a particular case, when the family of quadrics contain the center of projection of the first camera. We have shown that containing the origin does not limit the generality of the quadric fitting and have proven that the planar case is included within the model. Experiments on real images of general objects illustrate the applicability of our method. Q-warping provides more flexibility in fitting flow fields to the scene than existing planar models, yet reduces to planar warping when the scene requires so.

References

[1] G. Adiv. Determining three–dimensional motion and structure from optical flow generated by several moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-7(4):384–401, 1985.

[2] J.R. Bergen and E.H. Adelson. Hierarchical, computationally efficient motion estimation algorithm. *Journal of the Optical Society of America*, 4:35, 1987.

[3] J.R. Bergen, P. Anandan, K.J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *Proceedings of the European Conference on Computer Vision*, Santa Margherita Ligure, Italy, June 1992.

[4] J.R. Bergen and R. Hingorani. Hierarchical motion-based frame rate conversion. Technical report, David Sarnoff Research Center, 1990.

[5] P. J. Burt and E. H. Adelson. The laplacian pyramid as a compact image code. *IEEE Transactions on Communication*, 31:532–540, 1983.

[6] G. Cross and A. Zisserman. Quadric reconstruction from dual-space geometry. In *Proceedings of the International Conference on Computer Vision*, Bombay, India, January 1998.

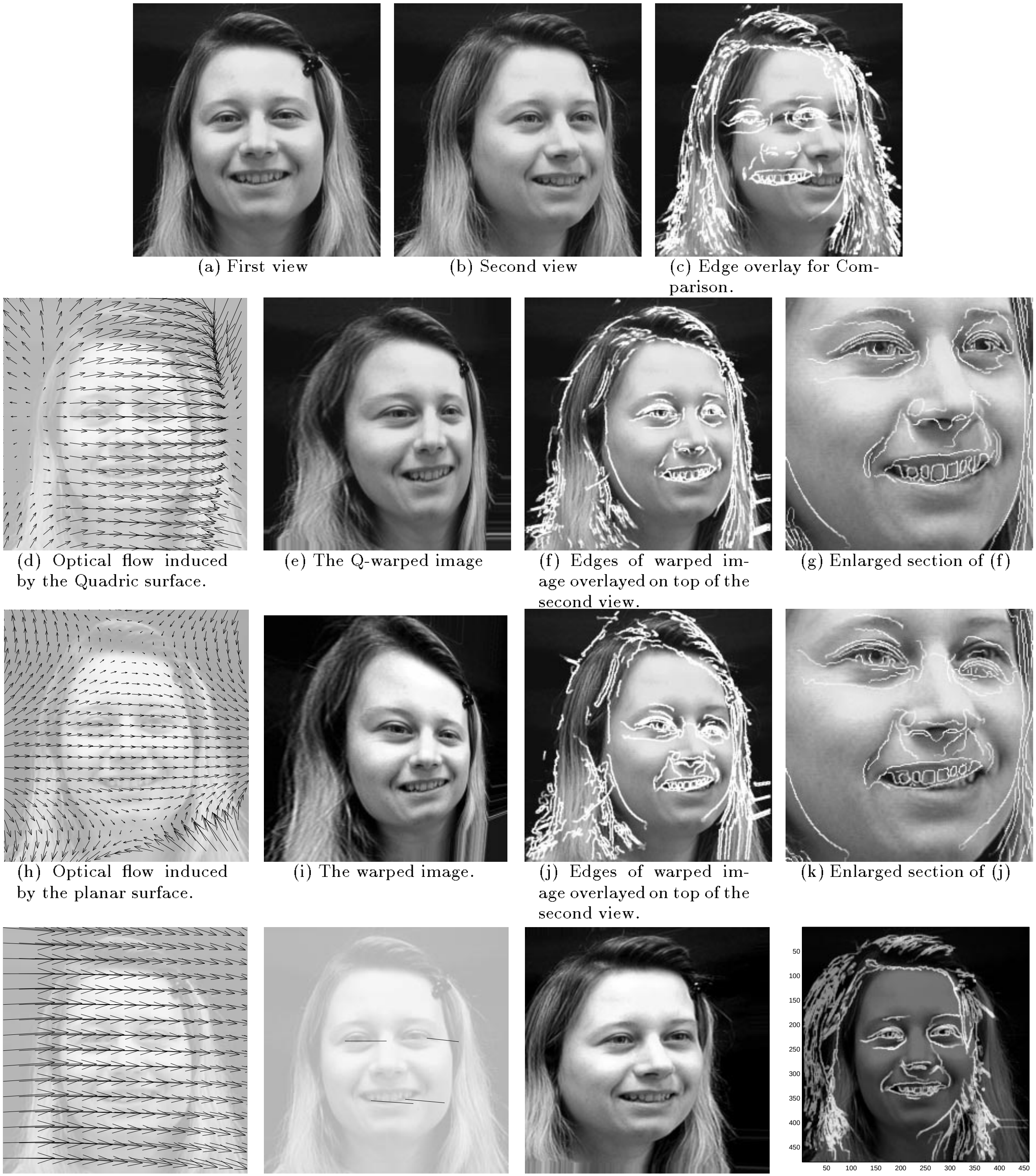
[7] K.J. Hanna and N.E. Okamoto. Combining stereo and motion analysis for direct estimation of scene structure. In *Proceedings of the International Conference on Computer Vision*, pages 357–365, 1993.

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

[9] B.K.P. Horn and E.J. Weldon. Direct methods for recovering motion. *International Journal of Computer Vision*, 2:51–76, 1988.

[10] M. Irani, B. Rousso, and S. Peleg. Computing occluding and transparent motions. *International Journal of Computer Vision*, 12(1):5–16, 1994.

[11] H.C. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. *Proceedings*



(a) First view

(b) Second view

(c) Edge overlay for Comparison.

(d) Optical flow induced by the Quadric surface.

(e) The Q-warped image

(f) Edges of warped image overlayed on top of the second view.

(g) Enlarged section of (f)

(h) Optical flow induced by the planar surface.

(i) The warped image.

(j) Edges of warped image overlayed on top of the second view.

(k) Enlarged section of (j)

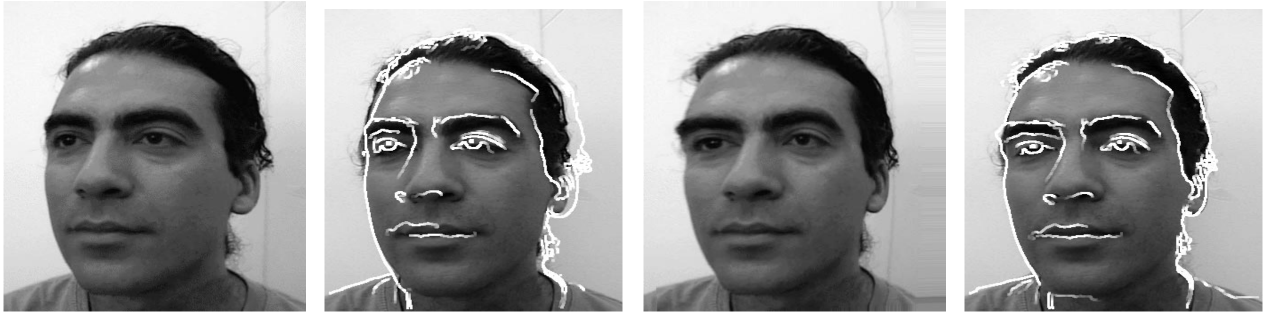
(l) Optical flow

(m) matches on 1st image

(n) Warped image

(o) Edges of warped on 1st image

Figure 3: Application of Q-warping on general objects. *Row 1* displays the original two views and the edge overlay in order to appreciate the distance between matching features. *Row 2* displays the Q-warping results. Note that the features are aligned up-to a few pixels. The alignment is not expected to be accurate because the object is not a quadric, but the small residual flow suggests that the fitted quadric was wrapped closely around the object. *Row 3* compares the results with a direct estimation planar flow (eqn. 4). *Bottom Row* compares the results with a planar flow recovered from discrete 4 point matches from the center of the face.



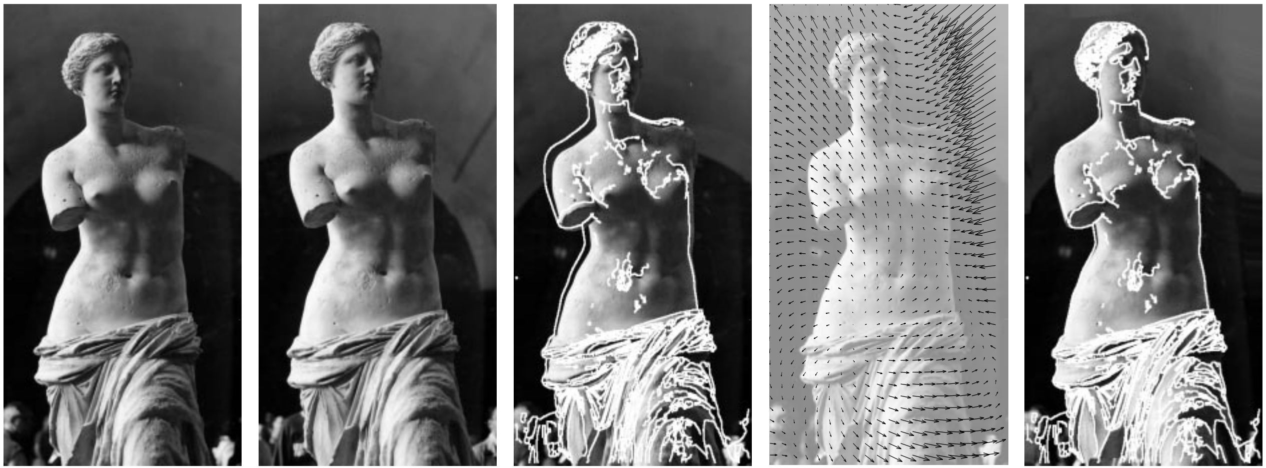
(a) First view

(b) Edges of the first view overlayed on the second view.

(c) The Q-warped image.

(d) Edges of warped image overlayed on top of the second view.

Figure 4: Another face example. Note that the warped image is in most regions closely aligned with the second view.



(a) First view

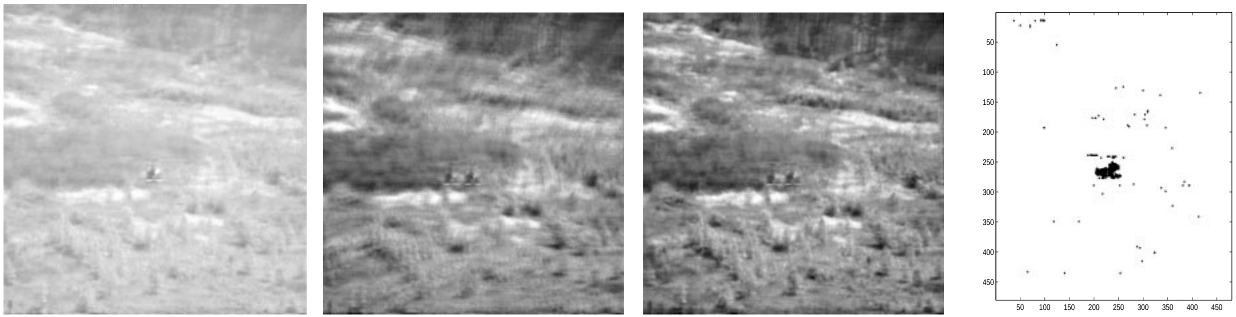
(b) Second image

(c) Edges of the first view overlayed on the second view.

(d) Optical flow induced by the quadric.

(e) Edges of warped image overlayed on top of the second view.

Figure 5: Another example on a general object. Note that the warped image is in most regions closely aligned with the second view.



(a) The first image in the sequence

(b) The average of (a) and the last image in the sequence

(c) The average of (a) and the stabilized last image

(d) Detected motion for the first frame

Figure 6: The Chopper sequence.

of the Royal Society of London B, 208:385–397, 1980.

[12] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings IJCAI*, pages 674–679, Vancouver, Canada, 1981.

[13] A. Rosenfeld. *Multiresolution image processing and analysis.* Springer-Verlag, 1984.

[14] A. Shashua and S. Toelg. The quadric reference surface: Theory and applications. *International Journal of Computer Vision*, 23(2):185–198, 1997.

[15] G. Stein and A. Shashua. Model based brightness constraints: On direct estimation of structure and motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Puerto Rico, June 1997.