Evaluating the Range Flow Motion Constraint

Hagen Spies
ICG-III: Phytosphere
Research Center Jülich
52425 Jülich, Germany
h.spies@fz-juelich.de

John L. Barron
Department of Computer Science
University of Western Ontario
London, Ontario, N6G 5B7 Canada
barron@csd.uwo.ca

Abstract

The instantaneous three-dimensional velocity field of a moving surface can be computed from a sequence of dense range data sets. Here we discuss the computation of the underlying motion constraint equation. This involves the evaluation of derivatives of the depth coordinate with respect to the other world coordinates. As these are not evenly sampled the sampling has to be taken into account explicitly. We quantitatively compare four methods to compute derivatives based on the validity of the resulting constraint equation.

1. Introduction

The three-dimensional velocity field of deformable surface can be recovered from a sequence of range data sets [6, 7, 8, 13]. We denote this velocity field *range flow* and the constraint equation used to compute it *range flow constraint equation*. This equation involves derivatives with respect to world coordinates but the data is only available in sensor coordinates. Because the data is generally unevenly sampled the direct use of derivative filters is questionable.

Previously this limitation has sometimes been resolved by resampling the data. One can for example use a membrane model [3] or thin plate splines [9] to interpolate the data on a regular grid. Then the condition of evenly sampled data is met again and standard derivative filters can be used. Other approaches use finite element models [10, 13].

Here we present an experimental comparison of surface fits using ordinary least squares and total least squares and a novel formulation of the constraint equation using derivatives only. This is also compared to a direct derivative computation that ignores the uneven sampling.

2. Range Flow Constraint Equation

A time varying surface may be viewed as a depth function $Z(X, Y, t)$. If the object under consideration is made up of local planar patches then this function can be expressed by its first order Taylor series expansion:

$$Z(X, Y, t) = Z(0, 0, t) + Z_X X(t) + Z_Y Y(t) \ , \qquad (1)$$

where the Z_X, Z_Y denote the partial derivatives with respect to X and Y respectively. Here X, Y are the local world coordinates around the point of interest. The change in depth with time then becomes:

$$\frac{\mathrm{d}Z}{\mathrm{d}t} = Z_t + Z_X \frac{\mathrm{d}X}{\mathrm{d}t} + Z_Y \frac{\mathrm{d}Y}{\mathrm{d}t} + X \frac{\mathrm{d}Z_X}{\mathrm{d}t} + Y \frac{\mathrm{d}Z_Y}{\mathrm{d}t} \ . \quad (2)$$

Under the assumption that the infinitesimal motion of the patch is a pure translation, i.e. the slope does not change ($\frac{\mathrm{d}Z_X}{\mathrm{d}t} = \frac{\mathrm{d}Z_Y}{\mathrm{d}t} = 0$), this can be written as:

$$W = Z_X U + Z_Y V + Z_t \ . \qquad (3)$$

The local displacements form the range flow denoted by: $\boldsymbol{f} = [U\ V\ W]^T = \frac{\mathrm{d}}{\mathrm{d}t}[X\ Y\ Z]^T$. Equation (3) will be called *range flow motion constraint* (RFMC), the same equation has also been termed *elevation rate constraint equation* [3] and *range flow constraint equation* [13]. It is the analogon to the *brightness change constraint equation* used in optical flow calculation [4].

3. Derivative Estimation

In order to evaluate the RFMC equation (3) the partial derivatives of the depth function with respect to the other world coordinates have to be computed. This is not entirely straightforward for unevenly sampled data. A direct estimation discards the sampling problem and evaluates derivatives directly on the depth map ($Z_X = Z_x;\ Z_Y = Z_y$). A better approach is to compute the derivatives from a surface fit using either ordinary least squares OLS (Sect. 3.1) or total least squares TLS (Sect. 3.2). A novel approach that uses derivative kernels is presented in Sect. 3.3.

3.1. OLS Estimation

The derivatives can be computed from a local first order approximation of the surface (depth function):

$$\begin{aligned} &Z(X+\Delta X, Y+\Delta Y, t+\Delta t) = \\ &\qquad Z(X,Y,t) + Z_X \Delta X + Z_Y \Delta Y + Z_t \Delta t\,. \end{aligned} \tag{4}$$

If we assume the derivatives to be constant in a small local neighbourhood we can estimate them by solving a linear system of equations:

$$\underbrace{\begin{bmatrix} \Delta X_1 & \Delta Y_1 & \Delta t_1 \\ & \vdots & \\ \Delta X_N & \Delta Y_N & \Delta t_N \end{bmatrix}}_{\boldsymbol{A}} \underbrace{\begin{bmatrix} Z_X \\ Z_Y \\ Z_t \end{bmatrix}}_{\boldsymbol{x}} = \underbrace{\begin{bmatrix} \Delta Z_1 \\ \vdots \\ \Delta Z_N \end{bmatrix}}_{\boldsymbol{b}}\,, \tag{5}$$

where $\Delta Z_i = Z(X+\Delta X_i, Y+\Delta Y_i, t+\Delta t_i) - Z(X,Y,t)$. Equation (5) is easily solved using the pseudo inverse: $\boldsymbol{x} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}$. This approach implicitly assumes the data matrix $\boldsymbol{A}$ to be error free. While this is true, at least for practical purposes, for the temporal changes it certainly does not hold for the spatial differences. However the accuracy in X and Y is typically one order of magnitude better than that in the depth estimate [2]. It might still be advisable to use an error in variables model as presented next.

In order to give more influence to the central pixels we use a Binomial as weights, this will be denoted WOLS.

3.2. TLS Estimation

Due to the data acquisition at least X and Y are also error-prone and even the sampling rate might not be known accurately. Therefore TLS is the estimation technique of choice here. We recast Eq. (4) for TLS as $\boldsymbol{d}^T\tilde{\boldsymbol{f}} = 0$ with:

$$\boldsymbol{d} = [\Delta X\ \Delta Y\ \Delta t\ {-\Delta Z}]^T \quad \text{and} \quad \tilde{\boldsymbol{f}} = [Z_X\ Z_Y\ Z_t\ 1]^T\,. \tag{6}$$

This can of course be easily extended to incorporate higher orders. The sought derivative values are then found from the eigenvector to the smallest eigenvalue of the scatter matrix:

$$\boldsymbol{J} = \boldsymbol{B} * (\boldsymbol{d}\boldsymbol{d}^T)\,. \tag{7}$$

Here $\boldsymbol{B}$ represents the local integration via an averaging filter. Because the error in the data vector $\boldsymbol{d}$ is not the same for all entries we need to scale accordingly [11].

As for OLS it is sometimes desirable to weight the central region more than the outer pixels. This is easily achieved using a Binomial integration filter $\boldsymbol{B}$ (WTLS).

3.3. Derivative Filters

Because of their fast application via convolutions and the availability of highly optimised filter kernels we would like to employ derivative filters. The question then becomes if the deviation due to uneven sampling can be compensated for. Here the objects of interest are 2D surfaces in the 3D world $Z = Z(X,Y,t)$. The data points are sampled at locations on the sensor array which in turn depend on the 3D data points observed: $x = x(X,Y,Z)$; $y = y(X,Y,Z)$.

The range sensors considered here do not yield a depth function in terms of X and Y but rather produce one data set for each of X, Y and Z on a sampling grid ($X = X(x,y,t)$ etc.). Here sensor coordinates are denoted by (x,y). For the total variation of the three data sets we obtain:

$$\mathrm{d}X = X_x\,\mathrm{d}x + X_y\,\mathrm{d}y + X_t\,\mathrm{d}t\,, \tag{8}$$

$$\mathrm{d}Y = Y_x\,\mathrm{d}x + Y_y\,\mathrm{d}y + Y_t\,\mathrm{d}t\,, \tag{9}$$

$$\mathrm{d}Z = Z_x\,\mathrm{d}x + Z_y\,\mathrm{d}y + Z_t\,\mathrm{d}t\,. \tag{10}$$

Eliminating $\mathrm{d}x$ and $\mathrm{d}y$ from (10) results in:

$$\begin{aligned} \mathrm{d}Z &= \left(\frac{Y_yZ_x - Y_xZ_y}{Y_yX_x - X_yY_x}\right)\mathrm{d}X \\ &+ \left(\frac{X_xZ_y - X_yZ_x}{Y_yX_x - X_yY_x}\right)\mathrm{d}Y \\ &+ \left(\frac{\frac{\partial(X,Y,Z)}{\partial(x,y,t)}}{Y_yX_x - X_yY_x}\right)\mathrm{d}t\,. \end{aligned} \tag{11}$$

Here $\frac{\partial(X,Y,Z)}{\partial(x,y,t)}$ denotes the Jacobian of (X,Y,Z) with respect to (x,y,t). As Eq. (11) has to be equal to $\mathrm{d}Z = Z_X\,\mathrm{d}X + Z_Y\,\mathrm{d}Y + Z_t\,\mathrm{d}t$ we can identify the partial derivatives of Z with respect to the other world coordinates. These awkward expressions can be considerably simplified if the sensor coordinate system is aligned with the world coordinate system, as is often the case. This implies $X_y = Y_x = 0$ and the spatial partial derivatives of Z become:

$$Z_X = \frac{Y_yZ_x - Y_xZ_y}{Y_yX_x - X_yY_x} = \frac{Z_x}{X_x}\,, \tag{12}$$

$$Z_Y = \frac{X_xZ_y - X_yZ_x}{Y_yX_x - X_yY_x} = \frac{Z_y}{Y_y}\,. \tag{13}$$

However we can not generally assume that the X,Y measurements are independent of Z and will thus use Eq. (11) below. The derivatives with respect to the sensor coordinates are computed via convolutions with derivative kernels. Here we use special filters that have been optimized to minimize the directional error [5].

4. Experiments

In the following we want to experimentally determine which of the described derivatives is best suited for range flow estimation. To do so in a quantitative fashion the residual of the RFMC (3) is considered as error measure:

$$r = Z_XU + Z_YV + Z_t - W\,. \tag{14}$$

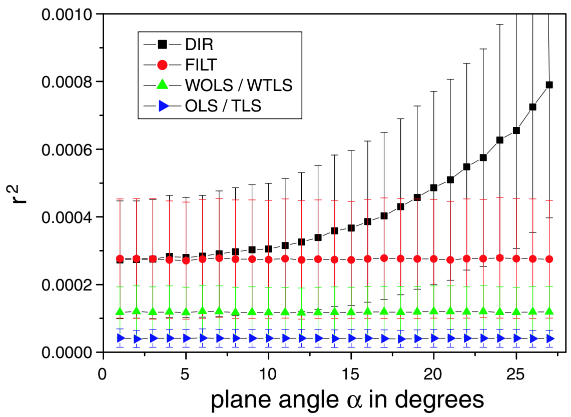

Figure 1. Result on the plane test sequences for an increasing slope of the plane.

For averaged quantities we either use r directly as a measure for the bias or r^2 to quantify the overall deviation.

In all experiments we use a small 3-tap Binomial spatial pre-filter on the range data in order to reduce the high frequency content for which the local planarity assumption does not hold. The used filters are all 5-tap and the fits are performed on a $5 \times 5 \times 5$ spatio-temporal volume. We use the following abbreviations for the different methods:

- OLS - ordinary least squares (Sect. 3.1).
- WOLS - weighted ordinary least squares (Sect. 3.1).
- TLS - total least squares (Sect. 3.2).
- WTLS - weighted total least squares (Sect. 3.2).
- FILT - derivative filters (Sect. 3.3).
- DIR - direct estimation: $Z_X = Z_x,\ Z_Y = Z_y$.

4.1. Synthetic Data

To investigate the performance on realistic test data we model a sensor using perspective projection, which corresponds to all sensors that employ a standard camera. Here we use a focal length of $f = 12\,mm$, a pixel size of $7.4 \times 7.4\ \mu m^2$ and 256×256 sensor elements. For testing a planar target as well as a sphere are used.

For the synthetic plane we choose a viewing distance of $300\,mm$ and typically an angle α between the surface normal and the Z-axis. The second type of synthetic data used consists of a sphere with a radius of $350\,mm$ with its center initially placed $500\,mm$ away from the camera. Note that in order to obtain the correct range flow at each sensor location the velocity has to be projected into the sensor frame.

In these experiments we use a realistic noise regime of $\sigma_X = \sigma_Y = 0.01\,mm$ and $\sigma_Z = 0.1\,mm$ [2]. Random noise following a normal distribution with the given standard deviations is added to the data arrays.

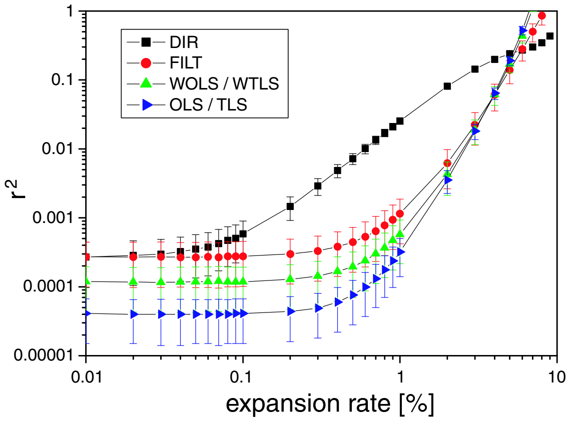

Figure 2. Result on the sphere test sequences for an increasing expansion rate.

Figure 1 shows the squared residual r^2 for a motion of the plane of $[0.1, 0.2, 0.3]^T$mm/frame and an increasing vertical angle α. We clearly see that for higher angles the direct estimation introduces larger errors as has to expected. The other methods yield constant error values. Note that OLS(WOLS) and TLS(WTLS) produce nearly identical results (shown together), because the accuracy in X, Y is much better than that in Z. We find that the surface fits yield somewhat better results than the filter method. Also, the use of weights slightly increases the residual on such ideal data. This is a direct result of the better noise suppression of the average. On data with faster variations the better characteristics of the Binomial comes into play.

Similar results are also found for the sphere, where we use an increasing expansion rate (Fig. 2). For higher expansion rates the direct estimation deteriorates much faster than the other methods. However, in this case the other methods also fail for high expansion rates, This is caused by the temporal aliasing accompanied with large movements.

4.2. Real Data

We present work on data captured with a *Biris* laser range finder [12, 2] and a structured light (phase shifting) system using a miniature line projector with grating made of chrome on glass [1]. Both are used at a working distance of about 300mm and a field of view of $\approx 100 \times 150$mm^2. Figure 3a shows the Biris real test sequence, the correct movement of $[-0.16, 0.19, -0.32]^T$mm/frame was achieved by placing the object (a toy tiger) on a system of linear positioners. Figure 3b shows the structured light test sequence (a crumpled sheet of paper), again the correct movement of $[0.0, 0.2, 0.3]^T$mm/frame is created by linear positioners. Note that because we are using special filters optimized for minimal directional error the zero movement in X-direction

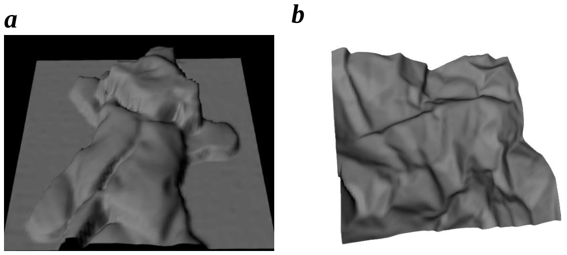

Figure 3. Real test sequences (rendered): a) Biris data of a toy tiger and b) structured light data of a crumpled sheet of paper.

in the second case should only make a minor difference [5].

We also report the density d and relative error E_r in the range flow magnitude as estimated by a total least squares technique detailed in [6, 7]. The following results are found on the Biris data (Fig. 3a):

method	r	r^2	E_r [%]	d [%]
OLS	-0.19 ± 0.04	0.03 ± 0.01	6.0 ± 6.6	16.1
WOLS	-0.19 ± 0.04	0.04 ± 0.02	11.0 ± 9.2	13.3
TLS	-0.19 ± 0.04	0.04 ± 0.01	6.0 ± 6.5	16.1
WTLS	-0.19 ± 0.04	0.04 ± 0.02	11.0 ± 9.1	13.3
FILT	-0.19 ± 0.04	0.05 ± 0.04	3.6 ± 1.6	24.3
DIR	-0.20 ± 0.05	0.06 ± 0.05	3.2 ± 1.6	23.5

We note that all methods exhibit the same bias, indicated by a negative mean r. This is probably caused by the alignment of the sensor and the positioner system, which has been done manually. Hence the correct movement is not entirely accurate. On the squared residual we find little difference. However there is a significant improvement on the computed range flow and the density when either filters or the direct derivative estimation is used. Also note that weighting the OLS and TLS method reduces the accuracy, as was also found on the synthetic data. On the structured light data (Fig. 3b) we obtain:

method	r	r^2	E_r [%]	d [%]
OLS	-0.007 ± 0.007	0.007 ± 0.0001	0.29 ± 2.89	12.3
WOLS	-0.007 ± 0.007	0.007 ± 0.0001	0.09 ± 0.11	13.5
TLS	-0.007 ± 0.007	0.007 ± 0.0001	0.29 ± 2.88	12.3
WTLS	-0.007 ± 0.007	0.007 ± 0.0001	0.09 ± 0.11	13.5
FILT	-0.007 ± 0.007	0.007 ± 0.0001	0.09 ± 0.22	14.0
DIR	-0.006 ± 0.008	0.007 ± 0.0002	0.53 ± 0.35	14.2

Here no bias can be found and the averaged squared residual is the same for all methods. As the same positioning system used for calibration is used for the movement there are no alignment problems. On this data weighting does improve the accuracy when using OLS and TLS. We also note that the direct estimation gives less accurate range flow. Yet with a relative error of 0.5% the result is still very good. Hence we conclude that on the real sequences used here one might as well compute the derivatives directly from the depth data. However one should bear in mind that here higher depth slopes only occur at a few places. In particular for the first sequence the slope in these places becomes so high that no reliable range flow estimation is possible in the first place. This violation of the local planarity assumption is probably also the reason why the results are much better on the second data set.

5. Conclusions

We examined four different techniques to evaluate the range flow motion constraint equation and have presented an empirical accuracy analysis. For higher surface slopes one should take the uneven sampling into account and use either a surface fit by OLS or TLS or the derivatives obtained by a change of coordinates from the sensor to the world frame. The latter method also gives excellent results on two tested real sequences. However if the depth slope remains moderate the direct use of filters on the depth data alone seems to be a good alternative to the more expensive other methods.

References

[1] ABW GmbH. 72636 Frickenhausen, Germany.

[2] J. Beraldin, S. F. El-Hakim, and F. Blais. Performance evaluation of three active vision systems built at the national research council of canada. In *Conf. on Optical 3D Measurement Techn. III*, pages 352–361, Vienna, Austria, Oct. 1995.

[3] B. K. P. Horn and J. Harris. Rigid body motion from range image sequences. *CVGIP*, 53(1):1–13, January 1991.

[4] B. K. P. Horn and B. Schunk. Determining optical flow. *Artificial Intelligence*, 17:185–204, 1981.

[5] B. Jähne, H. Scharr, and S. Körkel. Principles of filter design. In *Handbook of Computer Vision and Applications*, volume 2, pages 125–151. Academic Press, 1999.

[6] H. Spies, H. Haußecker, B. Jähne, and J. L. Barron. Differential range flow estimation. In *DAGM*, pages 309–316, Bonn, Germany, September 1999.

[7] H. Spies, B. Jähne, and J. L. Barron. Regularised range flow. In *ECCV*, volume 2 of *LNCS 1843*, pages 785–799, Dublin, Ireland, June/July 2000. Springer.

[8] H. Spies, B. Jähne, and J. L. Barron. Surface expansion from range data sequences. In *DAGM*, LNCS 2191, pages 163–169, Munich, Germany, September 2001. Springer.

[9] R. Szeliski. Estimating motion from sparse range data without correspondence. In *ICCV*, pages 207–216, 1988.

[10] L. Tsap, D. Goldgof, and S. Sarkar. Multiscale combination of physically-based registration and deformation modeling. In *CVPR*, pages 422–429, Hilton Head, SC, June 2000.

[11] S. Van Huffel and J. Vandewalle. *The Total Least Squares Problem: Computational Aspects and Analysis*. Society for Industrial and Applied Mathematics, Philadelphia, 1991.

[12] Vitana Corporation. Ottawa, ON K1H 1E1, Canada.

[13] M. Yamamoto, P. Boulanger, J. Beraldin, and M. Rioux. Direct estimation of range flow on deformable shape from a video rate range camera. *PAMI*, 15(1):82–89, January 1993.