Paper presented at the 22nd Asian Conference on Remote Sensing, 5 - 9 November 2001, Singapore.

COMPARISON OF NEWTON-GAUSS WITH LEVENBERG-MARQUARDT ALGORITHM FOR SPACE RESECTION

Yao Jianchao and Chia Tien Chern
Signal Processing Laboratory, DSO National Laboratories
20 Science Park Drive, Singapore 118230
Tel: (65)7727179 Fax : (65)7759011
{yjiancha,ctienche}@dso.org.sg

KEY WORDS: Space resection, Non-linear Model-fitting, Photogrammetry, Levenberg-Marquardt Algorithm

ABSTRACT: Based on the perspective view of non-linear model fitting, a new algorithm for space resection based on Levenberg-Marquardt algorithm was developed in this paper. The relationship between the new algorithm and the current one, which is commonly implemented in the commercial software, was also discussed. The experimental evaluation of both algorithms with different level of inaccurate initial approximation was conducted, which demonstrates the superiority of the algorithm.

1. INTRODUCTION

Space Resection is the process of determining the elements of exterior orientation and position of sensor from ground control points and their image plane co-ordinates. It is the prerequisite for Digital Elevation Model reconstruction and object localization. The most common method of computation is by use of collinearity equations, which rely on the principle that, the ground point, its imaged point and the center of projection, all lie on a straight line. For every control point, we can obtain two equations. Since there are six exterior parameters (three for orientation and three for translation), at least 3 controls point are required to solve the system. As usual we desire more than 3 points, hence the least squares computation technique is applied to determine the most probable value for the six parameters. This technique has been well established and has been widely used in various commercial software.

Although the space resection based on collinearity equations is quite effective in application, there is an inherent limitation associated with the approach. Because the whole process starts with linearization of a non-linear mathematical model, it is usually necessary to repeat the computations using improved value for the initial approximation of the unknowns. If the initial approximation of sensor's parameter is not good enough, the algorithm will diverge. A lot of effort was placed on establishing good initial approximate value of unknowns, so as to make the algorithm converge to the true solution (P.R.Wolf al. 2000). For instance, with near-vertical photography, the value for ω and ϕ will be 0; κ may be estimated by identifying the direction of north on the photograph. However, for the general tilted photograph, it is inconvenient or even impractical. Thus two issues about current space resection technique are naturally raised. One is the performance of the algorithm with different level of inaccuracy of the initial approximation; the other is how to improve it. To seek for the answer of these issues consititute the bulk of the material presented in the paper.

In this paper, instead of linearisation of collinearity equations, we look at the issue of space resection from the perspective view of the non-linear model fitting. Following the steps of the non-linear model fitting, we developed another estimation framework for space resection based on Levneberg-Marquardt algorithm. Furthermore, we show that the space resection based on the linearization is nothing but a special case of our new estimation framework in which the time-varying stabilisation parameter λ is taken to zero, which is actually the Newton-Gauss algorithm. Therefore, two estimation approaches for space resection are united under the same estimation framework. Furthermore, we analysis the performance of algorithm with different level of inaccuracy of the initial approximation with simulation data coming from real scenario. The experimental result demonstrated that the performance of space resection based on Levenberg-Marquardt algorithm is superior to that of the Newton-Gauss algorithm in terms of tolerance of inaccuracy for initial approximation. The fact suggests us the replacement of the Newton-Gauss based algorithm with Leverberg-Marquardt for the functionality of space resection in various commercial software.

2. COLLINEARITY EQUATION AND ITS APPLICATION TO SPACE RESECTION

Suppose an object point $P=(X_A, Y_A, Z_A)$ in a 3D global co-ordinate system O-XYZ is projected to an image point $p=(x_a, y_a)$ in 2D image co-ordinate system o-xy. The perspective projection from P to p through the perspective center is characterized by the following collinearity equation

$$x_a = x_0 - f\frac{m_{11}(X_A - X_L) + m_{12}(Y_A - Y_L) + m_{13}(Z_A - Z_L)}{m_{31}(X_A - X_L) + m_{32}(Y_A - Y_L) + m_{33}(Z_A - Z_L)}$$

$$y_a = y_0 - f\frac{m_{21}(X_A - X_L) + m_{22}(Y_A - Y_L) + m_{23}(Z_A - Z_L)}{m_{31}(X_A - X_L) + m_{32}(Y_A - Y_L) + m_{33}(Z_A - Z_L)} \qquad (1)$$

where (x_0, y_0) represents the principal point of camera, f focal length, (X_L, Y_L, Z_L) position of camera, $m_{ij}\,(1 \le i, j \le 3)$ are elements of an orthogonal matrix defined by three successive rotation angles ω,ϕ, κ, or roll , pitch and yaw respectively.

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & \sin\omega \\ 0 & -\sin\omega & \cos\omega \end{bmatrix} \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\kappa & \sin\kappa & 0 \\ -\sin\kappa & \cos\kappa & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (2)$$

We observe that equation (1) is non-linear and involve nine unknowns: the three rotation angles which are inherent in the *m*'s; the three exposure station co-ordinates X_L, Y_L and Z_L; and the three object point co-ordinates X_A, Y_A and Z_A. Note, however, that since the object point co-ordinates of the control points are known in the issue of space resection, the number of unknowns reduces to six. On the other hand, the photo co-ordinate measurements (x_a, y_a) are constant terms, as well as the calibration parameters x_0, y_0 and f which are considered to be constants in most applications of collinearity. The non-linear collinearity equations are linearized by using Taylor's theorem. In linearizing them, equations (1) are written as follows:

$$F = x_0 - f\frac{r}{q} = x_a; \quad G = y_0 - f\frac{s}{q} = y_a \qquad (2)$$

Where

$$q = m_{31}(X_A - X_L) + m_{32}(Y_A - Y_L) + m_{33}(Z_A - Z_L);$$
$$r = m_{11}(X_A - X_L) + m_{12}(Y_A - Y_L) + m_{13}(Z_A - Z_L)$$
$$s = m_{21}(X_A - X_L) + m_{22}(Y_A - Y_L) + m_{23}(Z_A - Z_L)$$

By using Taylor's expansion, Eqs. (2) may be expressed in linearized form by taking partial derivatives with respect to the unknowns:

$$(F)_0 + (\frac{\partial F}{\partial x_a})|_0\, dx_a + (\frac{\partial F}{\partial \omega})|_0\, d\omega + (\frac{\partial F}{\partial \phi})|_0\, d\phi + (\frac{\partial F}{\partial \kappa})|_0\, d\kappa + (\frac{\partial F}{\partial X_L})|_0\, dX_L + (\frac{\partial F}{\partial Y_L})|_0\, dY_L + (\frac{\partial F}{\partial Z_L})|_0\, dZ_L = x_a;$$

$$(G)_0 + (\frac{\partial G}{\partial y_a})|_0\, dy_a + (\frac{\partial G}{\partial \omega})|_0\, d\omega + (\frac{\partial G}{\partial \phi})|_0\, d\phi + (\frac{\partial G}{\partial \kappa})|_0\, d\kappa + (\frac{\partial G}{\partial X_L})|_0\, dX_L + (\frac{\partial G}{\partial Y_L})|_0\, dY_L + (\frac{\partial G}{\partial Z_L})|_0\, dZ_L = y_0;$$

Since the photo co-ordinates (x_a, y_a) are measured values, if the equations are to be used in a least square solution, residual terms must be included to make the equations consistent. Therefore, we can obtain two linearised collinearity equations including these residuals for every control point given

$$b_{11}d\omega + b_{12}d\phi + b_{13}d\kappa - b_{14}dX_L - b_{15}dY_L - b_{16}dZ_L = J + v_{x_a}$$
$$b_{21}d\omega + b_{22}d\phi + b_{23}d\kappa - b_{24}dX_L - b_{25}dY_L - b_{26}dZ_L = K + v_{y_a}$$

The coefficients of the equations can be derived easily. For example,

$$b_{11}=\frac{\partial F}{\partial \omega}=\frac{f}{q^2}[r(-m_{33}\Delta Y+m_{32}\Delta Z)-q(-m_{13}\Delta Y+m_{12}\Delta Z)];....;J=x_a-x_0+f\frac{r}{q}$$
$$b_{21}=\frac{\partial G}{\partial \omega}=\frac{f}{q^2}[s(-m_{33}\Delta Y+m_{32}\Delta Z)-q(-m_{23}\Delta Y+m_{22}\Delta Z)];...;K=y_a-y_0+f\frac{s}{q} \tag{3}$$

and $\Delta X = X_A - X_L; \Delta Y = Y_A - Y_L; \Delta Z = Z_A - Z_L;$

If we are given *n* (*n*>=3) points, we have a total of *m=2*n* linearized equations containing 6 unknowns:

$$\begin{cases} a_1 d\omega + b_1 d\phi + c_1 d\kappa + d_1 dX_L + e_1 dY_L + f_1 dZ_L - L_1 = v_1 \\ a_2 d\omega + b_2 d\phi + c_2 d\kappa + d_2 dX_L + e_2 dY_L + f_2 dZ_L - L_2 = v_2 \\ \cdots\cdots\cdots\cdots\cdots\cdots \\ a_m d\omega + b_m d\phi + c_m d\kappa + d_m X_L + e_m dY_L + f_m dZ_L - L_m = v_m \end{cases} \quad \text{or} \quad A\Delta \vec{x} = L + V \tag{4}$$

By using least square techniques, the correction $\Delta\vec{x} = [d\omega, d\phi, d\kappa, dX_L, dY_L, dZ_L]^T$ can be obtained from

$$(A^T A)\Delta \vec{x} = (A^T L) \tag{5}$$

The Eq. (5) is called normal equations. Since the normal matrix corresponding to the normal equation is symmetric, the unknowns $d\omega, d\phi$ and $d\kappa$, etc., can be obtained via LU decomposition. Based on above derivation, the space resection algorithm using linearized collinearity equations can be constructed as follows:

1. Assign initial values of estimated vector $\vec{x} = [\omega, \phi, \kappa, X_L, Y_L, Z_L]^T$, set the sum of residuals $s_0 = \infty$;
2. Based on current estimate $\vec{x}$ and given control points and their image projections, forming the normal equations (5);
3. Conserving current sum of residuals $s_{old} = s_0$; Solving the normal equations to get correction to the estimation vector $\Delta\vec{x} = [d\omega, d\phi, d\kappa, dX_L, dY_L, dZ_L]^T$, and doing the correction $\vec{x} = \vec{x} + \Delta\vec{x}$;
4. Computing the sum of residuals s_0; if $(s_0 > s_{old})$, the algorithm diverges.
5. If $\Delta\vec{x} \approx 0$; output the estimated vector, else go back to 2

The above algorithm was adopted and implemented in all the commercial software, such as ErDas. The algorithm is, in fact, based on the Newton-Gauss Algorithm. The main idea of the Newton-Gauss algorithm is to approximate the non-linear residual surface (6) with quadratic function. The correction of $\vec{x}$ causes a jump directly to the minimum of the approximated quadratic function, thus the converging speed of the algorithm is faster, compared to the gradient steepest descent approach. The rationale of the approach is based on the fact that any non-linear residual surface can be approximated by quadratic function, at least when $\Delta\vec{x}$ is small. However, it is not true when $\vec{x}$ is far away from the its optimal solution. When the non-linear residual surface function around current estimate $\vec{x}$ cannot be well approximated by the quadratic function, i.e. $(s_0 > s_{old})$, the algorithm will diverge. Therefore, successful application of above algorithm implies a strict condition, which dictates the assumed initial value of $\vec{x}$ cannot be far away from its optimum solution.

3. SPACE RESECTION BASED ON LEVENBURG-MARQUART ALGORITHM

In this section, we deal with the space resection problem from perspective view of non-linear model fitting. The advantage of new representation of the old problem is to set up an united estimation framework that includes both Newton-Gauss and Levengure-Marquart as its components. As opposed to the approach in the last section, there is no obvious linearization process involved. In the following, based the steps of generic non-linear model fitting

(Wilian H. P al. 1992), we take the issue of space resection as an example to elaborate the framework, so as to discuss the relation between these two estimating approaches in more details

In this framework, the first step is to define the residual function. For our space resection problem, the residual is defined as follows

$r_i = \begin{bmatrix} F_i - x_a^i \\ G_i - y_a^i \end{bmatrix}$, where (x_a^i, y_a^i) is the image projections of *i*th control points, regarded as the measurement values in data model fitting. (F_i, G_i) is computed from the formula (2) given current estimate of $\vec{x}$. The error function can be formed based on the residual $e(\vec{x}) = \sum_i r_i^2$. In the second step, we calculate the gradient and Hessian matrix of error function. Under the assumption that error function is well approximated by a quadratic function, the gradient and hessian matrix are determined by

$$\frac{\partial r_i}{\partial \vec{x}} = \begin{bmatrix} \frac{\partial r_i}{\partial x_1} & \frac{\partial r_i}{\partial x_2} & \frac{\partial r_i}{\partial x_3} & \frac{\partial r_i}{\partial x_4} & \frac{\partial r_i}{\partial x_5} & \frac{\partial r_i}{\partial x_6} \end{bmatrix}^T = \begin{bmatrix} \frac{\partial F_i}{\partial x_1} & \frac{\partial G_i}{\partial x_1} & \cdots & \frac{\partial F_i}{\partial x_6} & \frac{\partial G_i}{\partial x_6} \end{bmatrix}^T \tag{6}$$

$$H = [h_{kl}]_{6x6} = [\sum_i (\frac{\partial r_i}{\partial x_k})(\frac{\partial r_i}{\partial x_l})^T]_{6x6} \tag{7}$$

where $\vec{x} = [x_1, x_2, x_3, x_4, x_5, x_6]^T = [\omega, \phi, \kappa, X_L, Y_L, Z_L]^T$

Every component of gradient vector can be obtained from equation (2). For example,

$$\frac{\partial r_i}{\partial x_1} = \begin{bmatrix} \frac{\partial F_i}{\partial x_1} \\ \frac{\partial G_i}{\partial x_1} \end{bmatrix} = \begin{bmatrix} b_{11}^i \\ b_{21}^i \end{bmatrix}; \ldots; \frac{\partial r_i}{\partial x_6} = \begin{bmatrix} \frac{\partial F_i}{\partial x_6} \\ \frac{\partial G_i}{\partial x_6} \end{bmatrix} = \begin{bmatrix} b_{16}^i \\ b_{26}^i \end{bmatrix} \tag{8}$$

In the third step, the set of linear equation is formed, which allows us to estimate the correction to the $\vec{x}$ either according to the LM algorithm or Newton-Gauss algorithm. The form of this set of linear equation is as follows:

$$(H + \lambda I)\Delta \vec{x} = \vec{b} \tag{9}$$

And $\vec{b} = \sum_i r_i \frac{\partial r_i}{\partial \vec{x}}$, and λ is a time-varying stabilization factor.

In formula (9), if we select $\lambda = 0$, the algorithm performs the Newton-Gauss minimization framework. In this situation, the formula (9) boils down to the normal equation (5). The fact demonstrates that the space resection algorithm using linearized collinearity equations described in the last section is equivalent to the Newton-Gauss minimization algorithm.

In the case when $\lambda \neq 0$, the algorithm becomes the Levenberg-Marquardt algorithm. It is an elegant method for varying smoothly between the Newton-Gauss and steepest descent method. When the initial value of $\vec{x}$ is far from the minimum, the error surface may not be well approximated by quadratic function, steepest descent was used, dealing with the more generic non-linear minimization issue. When current estimate $\vec{x}$ approaches the minimum, the functionality of the algorithm switches continuously to the Newton-Gauss approach. In this way, we can relax the strict demand for the initial value of $\vec{x}$, enabling the algorithm convergence even the initial estimate is not so close to the optimum solution.

Based on above steps, the space resection algorithm based on the Levenberg-Marquardt algorithm can be constructed as follows:

1. Assign initial values of estimated vector $\vec{x} = [\omega, \phi, \kappa, X_L, Y_L, Z_L]^T$, set the sum of residuals $s_0 = \infty$; and picking a modest value for λ, say λ=*0.01*.

2. Based on current estimate $\vec{x}$ and given control points and their image projections, forming the gradient vector and Hessian matrix based on equation (6) and (7). Conserving current sum of residuals $s_{old} = s_0$;
3. Solving the equations (9) to get correction to the estimation vector $\vec{\Delta x} = [d\omega, d\phi, d\kappa, dX_L, dY_L, dZ_L]^T$.
4. If $e(\vec{x} + \Delta \vec{x}) \geq e(\vec{x})$, increase λ by a factor of 10 and go back to 3.
5. If $e(\vec{x} + \Delta \vec{x}) < e(\vec{x})$, decrease λ by a factor of 10, conduct the correction $\vec{x} = \vec{x} + \vec{\Delta x}$ and $s_0 = e(\vec{x} + \Delta \vec{x})$
6. If $\vec{\Delta x} \approx 0$; output the estimated vector, else go back to 2.

4. PREFORMANCE ANALYSIS OF TWO ALGORITHMS

In this section, two experiments were conducted to analyze the performance of Newton-Gauss and Levenburg-Marquardt algorithms for space resection. The data used in experiments are simulation data, but it was obtained based on real scenario.

In the first experiment, the image was taken by a frame camera, whose focal length is 153.124 mm. The exterior parameter of camera when taking the picture is shown in Table1. From Table 1, we understand the image is the vertical photo. There are five control points available, whose co-ordinates are shown in Table 3. In order to show the robustness of Newton-Gauss and LM Algorithms for space resection, we put different noise levels on the initial values to see if the algorithm can be converged to the true values. In this case, we are able to set up the error bound for these two algorithms. Table 4 and Table 5 demonstrate the performance of Newton-Gauss and Levenburge-Marquart respectively. We observed that the performance of LM algorithm is superior to that of Newton-Gauss' s algorithm.

Table 1. The exterior information and interior orientation of the camera

	Omega	Phi	Kappa	X_L	Y_L	Z_L
Value	0.0132°	-0.0556°	90.3866°	666716.9974	115919.2083	8794.7161

Table 2. The co-ordinates of control points and their projections

	image co-ordinates (mm)		Co-ordinates in 3D global coordinate system (m)		
	x_a	y_a	X_a	Y_a	Z_a
1	-19.968142	33.563984	665228.955	115012.472	1947.672
2	70.815665	50.595992	664456.22	119052.15	1988.82
3	60.847086	-36.285546	668338.22	118685.9	1886.712
4	62.105090	-93.341627	670841.48	118696.89	2014.0
5	-25.316595	-94.049006	670970.45	114815.23	1891.888

Table 3. The performance of Newton-Gauss algorithm for space resection

Noise level	Conver-gence	Iteration no.	Initial values (position +orientation)
5 %	Yes	5	(700052.8473 121715.1687 9234.4519) (0.0139° -0.0584° 94.9059°)
10 %	Yes	6	(753388.6971 127511.1291 9674.1877) (0.0145° 0.0012° 99.4253°)
20 %	Yes	7	(800060.3969 139103.05 10553.6593) (0.0158° -0.0667° 108.4639°)
30 %	Yes	7	(866732.0966 150094.9708 11433.1309) (0.0172° –0.0723° 117.5026°)
40 %	Yes	8	(933403.7964 162286.8916 12312.6025) (0.0185° –0.0778° 126.5412°)
45 %	No	2	(966739.6462 168082.8520 12753.3383) (0.0191° -0.0826° 131.0606°)

Table 4. The performance of Levenburge-Marquardt algorithm for space resection

Noise level	Conver-gence	Iteration no.	Initial values (position +orientation)
5 %	Yes	4	(700052.8473 121715.1687 9234.4519) (0.0139° -0.0584° 94.9059°)
10 %	Yes	5	(753388.6971 127511.1291 9674.1877) (0.0145° 0.0012° 99.4253°)
20 %	Yes	5	(800060.3969 139103.05 10553.6593) (0.0158° -0.0667° 108.4639°)
30 %	Yes	6	(866732.0966 150094.9708 11433.1309) (0.0172° –0.0723° 117.5026°
40 %	Yes	7	(933403.7964 162286.8916 12312.6025) (0.0185° –0.0778° 126.5412°)

50 %	Yes	10	(1000075.4961 173878.8124 13192.0742) (0.0198° -0.0834° 135.5799°)
55%	Yes	23	(1033411.346 179674.346 13631.81) (0.0205° -0.0862° 142.0992°)

In the second experiment, instead of capturing a vertical photo, we take the picture in oblique direction. The camera position and control points remain the same as the first example. The only difference is the orientation angles of the camera, as shown in Table 5. Due to the orientation variation, the image co-ordinates of the projection of control points are also changed accordingly, as shown in Table 6. The performance analysis of the two algorithms is tabulated in Table 7 and Table 8. As expected, the LM algorithm for tilted photograph is advantage over the Newton-Gauss algorithm.

Table 5. The exterior information of the camera

	Omega	Phi	Kappa	X_L	Y_L	Z_L
Value	10.0132°	-5.0556°	70.3866°	666716.9974	115919.2083	8794.7161

Table 6. The co-ordinates of control points and their projections

	Image co-ordinates (mm)		Co-ordinates in 3D global coordinate system (m)		
	x_a	y_a	X_a	Y_a	Z_a
1	-63.24033	29.620801	665228.955	115012.472	1947.672
2	17.913679	73.322803	664456.22	119052.15	1988.82
3	36.460923	-8.386838	668338.22	118685.9	1886.712
4	53.792589	-56.241781	670841.48	118696.89	2014.0
5	-20.428644	-92.703080	670970.45	114815.23	1891.888

Table 7. The performance of Newton-Gauss algorithm for space resection

Noise level	Conver -gence	Iteration no.	Initial values (position +orientation)
1 %	Yes	6	(673384.1674 117078.4004 8882.6633) (10.1133° -5.1062° 71.0905°)
5 %	Yes	8	(700052.8473 121715.1687 9234.4519) (10.5139° -5..3084° 73.9059°)
7 %	Yes	10	(713387.1872 124033.5529 9410.3462) (10.7141° -5.4095° 75.3137°)
9 %	No	3	(726721.5272 126351.9370 9586.2405) (10.9144° -5..5106° 76.7214°)

Table 8. The performance of LM algorithm for space resection

Noise level	Conver -gence	Iteration no.	Initial values (position +orientation)
5 %	Yes	10	(700052.8473 121715.1687 9234.4519) (10.5139° -5..3084° 73.9059°)
9 %	Yes	13	(726721.5272 126351.9370 9586.2405) (10.9144° -5..5106° 76.7214°)
9.25 %	Yes	16	(728388.3197 126641.7351 9608.2273) (10.9394° -5..5232° 76.8974°)

Compared with these two examples, we observed that the robustness of either Newton or LM algorithm are extremely better in processing vertical photo than in tilted photo. This is because The formula (2.1), for the vertical photograph in which ω=0, ϕ=0 and κ=90°, the error function can be well approximated by a quadratic function, which is the convex function in nature. Since it is a convex function, the algorithms will converged its global minimum, regardless of the initial value assigned.

5. CONCLUSION

Two contributions were made in the paper. First of all, we developed a new algorithm for space resection based on the non-linear model fitting. The performance of the algorithm is illustrated to be superior to the current approach from both theory and experimental results. Secondly, perhaps more importantly, the instance of solving the space resection issue discussed in the paper gives the reader the possibility to take a fresh look at a number of old problems. There may exist some more efficient way to solve them, beside the popular approach recognized

REFERENCES

Paul R. Wolf , Bon A.Dewitt, 2000, Elements of Photogrammetry with Application in GIS. McGraw-Hill, pp. 237-244.

Wiliam H.Press al. 1992, Numerical Recipes in C. Cambridge University Press, pp. 681-688

Robert M. Harlick, 1989, Determining Camera Parameters from the perspective Projection of A Rectangle. Pattern Recognition, Vol. 2, No3, pp225-230