Rectification using a modified model of rational function

Ramadan Khalil
Transportation Dept., Faculty of Eng., Alexandria University, Alexandria, Egypt

Nowadays remote sensing is used in map-production and in natural resources management. Also, it is considered as a helpful tool in the environment protection. Resolution of satellite-image plays a basic role in the accuracy of produced maps. The procedure of transformation of the satellite image to cadastral maps needs rectification of coordinates. This paper presents the rectification using a modified model of Rational function. The results of this way were compared by the results of other methods, like using neural networks, affine formulae and polynomial formulae. Using Rational function proved a better accuracy than the other methods. This is an advantage for it, but it has many unknowns, so it needs more points for getting the best parameters.

إن بيانات الاستشعار عن بعد تستخدم اليوم فى انتاج الخرائط وادارة الموارد الطبيعية للبلاد والتخطيط العام و أيضا تستخدم فى متابعة التغييرات الطبيعية التى تحدث فى الأراضى من زحف عمرانى أو تصحر وخلافه. وأيضا تعتبر وسيلة مساعدة لدراسة البيئة والمحافظة عليها. والدقة الهندسية للصور الجوية تلعب دورا أساسيا فى دقة الخرائط الناتجة، وعملية تحويل الصورة الفضائية الى خريطة تحتاج الى اجرائات وتصحيح لإحداثيات النقط بالاعتماد على ثوابت ربط. هذا البحث يعرض أسلوب من الاجرائات الرياضية التى تمكننا من تحويل الاحداثيات من الصورة الى الخريطة والحصول على خرائط مقبولة الدقة. النموذج الرياضى الذى تبناه هذا البحث هو نموذج معدل من الدالة الكسرية وقد قورنت نتائج هذه الطريقة بنتائج طرق أخرى تم تناولها فى أبحاث سابقة مثل استخدام الشبكات العصبية واستخدام دوال affine و polynomial ومن النتائج توصلنا الى أن الدالة الكسرية تعطى دقة أعلى فى عملية تعديل الصور مع وجود عيب وحيد لها وهو أن عدد المجاهيل بها كثير ولذلك نحتاج الى عدد كبير من النقاط للتعويض بها والحصول على أحسن معاملات تحويل.

Keywords: Remote sensing, Rectification, Rational function

1. Definition of rational function model

The Rational Function Model (RFM) represents the relationship between the image coordinates and the object coordinates with ratios of polynomials, as shown in eq. (1),

$$\begin{cases} x = \dfrac{P_1(X,Y,Z)}{P_2(X,Y,Z)} \\ y = \dfrac{P_3(X,Y,Z)}{P_4(X,Y,Z)} \end{cases}, \qquad (1)$$

where the polynomial Pi (i=1, 2, 3, and 4) has the following general form

$$\begin{aligned} P_1(X,Y,Z) &= a_1 + a_2X + a_3Z + a_4XY + a_5XZ + a_6YZ \\ &+ a_7X^2 + a_8Y + a_9Z2 + a_{10}XYZ + a_{11}X^3 + a_{12}XY2 \\ &+ a_{13}XZ^2 + a_{14}X^3Y + a_{15}Y^3 + a_{16}YZ^2 + a_{17}X^2Z \\ &+ a_{18}Y^2Z + a_{19}Z^3 \end{aligned}, \quad (2)$$

and where (x, y) are the column and row of each image point and (X, Y, Z) are, for example, the longitude and latitude (in degrees, WGS84) and ellipsoidal height (in meters, WGS84) of the corresponding ground point [1].

The RFM sensor model describes the geometric relationship between the object space and image space. It relates object point coordinates (X,Y,Z) to image pixel coordinates (r,c) or vice versa using 78 Rational Polynomial Coefficients (RPCs) that allow users to perform photogrammetric processing in the absence of the rigorous physical sensor model. For the ground-to-image transformation, the defined ratios of polynomials have the following form:

$$r_n \frac{P1(X_n, Y_n, Z_n)}{P2(X_n, Y_n, Z_n)} = \frac{\sum_{i=0}^{m1}\sum_{j=0}^{m2}\sum_{k=0}^{m3} a_{yk} X_n^i Y_n^j Z_n^k}{\sum_{i=0}^{n1}\sum_{j=0}^{n2}\sum_{k=0}^{n3} b_{yk} X_n^i Y_n^j Z_n^k};$$

$$c_n \frac{P3(X_n, Y_n, Z_n)}{P4(X_n, Y_n, Z_n)} = \frac{\sum_{i=0}^{m1}\sum_{j=0}^{m2}\sum_{k=0}^{m3} c_{yk} X_n^i Y_n^j Z_n^k}{\sum_{i=0}^{n1}\sum_{j=0}^{n2}\sum_{k=0}^{n3} d_{yk} X_n^i Y_n^j Z_n^k}, \quad (3)$$

where (*rn*,*cn*) are the normalized row (line) and column (sample) index of pixels in image space; *Xn*, *Yn*, and *Zn* are normalized coordinate values of object points in ground space; and the polynomial coefficients *aijk*, *bijk*, *cijk*, *dijk* are called Rational Function Coefficients (RFCs). The normalization (i.e., offset and scale) minimizes the introduction of errors during computation. The total power of all ground coordinates is usually limited to three, that is, the RFCs are assumed to be zero wherever $i+j+k>3$. In such a case, each numerator or denominator is of twenty-term cubic form. Although several versions of different permutations of the polynomial terms occur in the literature. In the RFM model, ratios of the first-order terms can represent distortions caused by the optical projection, while corrections such as Earth curvature, atmospheric refraction, and lens distortion , can be well modeled by the second-order terms. Other unknown and more complex distortions with high-order components may be absorbed by the third-order terms. The inverse form of the RFM expresses the planar coordinates (X, Y) of an object point as rational functions of the image coordinates (r,c) and the vertical object coordinate Z:

$$X = \frac{p5(r,c,Z)}{p6(r,c,Z)}; \quad Y = \frac{p7(r,c,Z)}{p8(r,c,Z)} \ . \quad (4)$$

However, the forward RFM is superior to the inverse RFM when approximating the physical sensor models and exploiting imagery in terms of resultant accuracies as presented.

The method by which the RFM coefficients are recovered depends on the availability of a physical sensor model. In cases where a physical model is provided a terrain independent scheme can be applied. This scheme is based on the generation of a 3D grid in object space, using the physical sensor model. The 3D grid should contain several layers of points and its characteristics are determined by the coverage of the image and the terrain relief differences. Then, based on eq. (1), a Least Squares solution of the RFM coefficients can be derived. Finally, an estimation of the quality of the derived RFM coefficients should be carried out based on an evaluation of the residuals in a higher density 3D grid. When a physical model is not available, a terrain dependent scheme is used. This scheme is based on utilizing known GCPs, from which an estimation of the RFM coefficients could be derived. As in this scheme it is not possible to generate a 3D grid, the solution is highly sensitive to the terrain relief, as well as to the distribution, number, and quality of the GCPs used. It should be noted that in most cases, it would be exceptionally difficult for users to generate their own estimation of the RFM coefficients especially where a physical sensor model is not provided due to the requirement for a large number of well-distributed GCPs. In other cases, even if GCPs are available, the terrain characteristics may not allow deriving a reliable estimation. In such cases, users are heavily dependent on data providers [2].

2. Modification of RFM

By using a Taylor expansion the rational polynomial coefficients can be found. But in this paper a modification of rational function model will be done. The problem can be solved as follows.

The inverse form of the rational function, which presents the coordinates from the image space to the ground space, is expressed by

$$X = \frac{P_5\,(x,y,Z)}{P_6\,(x,y,Z)} \ . \quad (5)$$

$$Y = \frac{P_7\,(x,y,Z)}{P_8\,(x,y,Z)} \ . \quad (6)$$

The rational function can be expressed as

$$X = \frac{a_0 + a_1x + a_2y + a_3Z + \ldots a_{18}yZ^2 + a_{19}Z^3}{1 + b_1x + b_2y + b_3Z + \ldots b_{18}yZ^2 + b_{19}Z^3} \ . \quad (7)$$

$$Y = \frac{c_0 + c_1 x + c_2 y + c_3 Z + \ldots c_{18} yZ^2 + c_{19} Z^3}{1 + d_1 x + d_2 y + d_3 Z + \ldots d_{18} yZ^2 + d_{19} Z^3} . \quad (8)$$

The number of the unknown coefficients in each of the above equations $(a_{19}\ a_{18}\ \ldots a_3\ a_2\ a_1\ a_0\ \ b_{19}\ , b_{18} \ldots b_1,\ c_{19}\ c_{18}\ \ldots c_3\ c_2\ c_1\ c_0\ , d_{19}\ ,\ d_{18} \ldots d_1)$ is: m=39

This function can be rewritten as follows.

$$\begin{aligned} &a_{19}Z^3 + a_{18}yZ^2 + \ldots + a_3Z + a_2y + a_1x + a_0 - b_{19}XZ^3 \\ &- b_{18}XyZ^2 - \ldots - b_3XZ - b_2Xy - b_1Xx - X = 0 \\ &c_{19}Z^3 + c_{18}yZ^2 + \ldots + c_3Z + c_2y + c_1x + c_0 - d_{19}YZ^3 \\ &- d_{18}YyZ^2 - \ldots - d_3YZ - d_2Yy - d_1Yx - Y = 0 \end{aligned} . \quad (9)$$

Each of the above algebraic equations can be expressed as a product of two matrices as follows.

$$(Z^3\ yZ^2\ \ldots\ Z\ y\ x\ 1\quad XZ^3\ XyZ^2\ \ldots .XZ\ Xy\ Xx\ X)$$
$$(a_{19}\ a_{18}\ ..a_3\ a_2\ a_1\ a_0\ \ -b_{19} - b_{18} \ldots - b_1)^T = X . \quad (10)$$

$$(Z^3\ yZ^2\ \ldots Z\ y\ x\ 1\quad YZ^3\ YyZ^2\ \ \ldots YZ\ \ Yy\ \ Yx\ \ Y)$$
$$(c_{19}\ c_{18}\ \ .c_3\ c_2\ c_1\ c_0\ \ -d_{19} - d_{18} \ldots - d_1)^T = Y . \quad (11)$$

If we substitute in eq. (10) by coordinate values of n control points (n >= 39) this equations reduces to

$$(Z_1^3\ y_1Z_1^2\ ..\ Z_1\ y_1\ x_1\ 1\ \ X_1Z_1^3\ X_1y_1Z_1^2\ ..X_1Z_1\ X_1y_1\ X_1x_1\ X_1)$$
$$(a_{19}\ a_{18} \cdot a_3\ a_2\ a_1\ a_0\ \ -b_{19} - b_{18} \ldots - b_1)^T = X_1$$

$$(Z_2^3\ y_2Z_2^2\ \ldots Z_2\ y_2\ x_2\ 1\ X_2Z_2^3\ X_2y_2Z_2^2\ \ldots X_2Z_2\ X_2y_2\ X_2x_2\ X_2)$$
$$(a_{19}\ a_{18}\ \ldots a_3\ a_2\ a_1\ a_0\ -b_{19} - b_{18} \ldots - b_1)^T = X2$$

……………………

……………………

$$(Z_n^3\ y_nZ_n^2\ \ldots .Z_n\ y_n\ x_n\ 1\ X_nZ_n^3\ X_ny_nZ_n^2\ \ldots X_nZ_n\ X_ny_n\ X_nx_n\ X_n)$$
$$(a_{19}\ a_{18}\ \ldots \ldots a_3\ a_2\ a_1\ a_0\ -b_{19} - b_{18} \ldots - b_1)^T = X_n$$

Comparing the above matrices by analogy with the well known least square matrix representation:

$$A_{(n*m)}\ X_{(m*1)} = L_{(n*1)} . \quad (12)$$

$$A^T_{(m*n)}\ A_{(n*m)}\ X_{(m*1)} = A^T_{(m*n)}\ L_{(n*1)} . \quad (13)$$

$$N_{(m*m)}\ X_{(m*1)} = A^T_{(m*n)}\ L_{(n*1)} . \quad (14)$$

$$X_{(m*1)} = N^{-1}_{(m*m)} A^T_{(m*n)}\ L_{(n*1)} . \quad (15)$$

The coefficients of the rational function $(a_{19}\ a_{18}\ \ldots \ldots a_3\ a_2\ a_1\ a_0\quad b_{19}\ , b_{18} \ldots b_1)$ correspond to the elements of X in the solution vector in eq. (15). These coefficients can be determined using the Least-Squares solution. In a similar way the coefficients $(c_{19}\ c_{18} \ldots c_3\ c_2\ c_1\ c_0\ , d_{19}\ ,\ d_{18} \ldots d_1)$ of eq. (11) can be determined.

3. Implementation of the rectification

A satellite IRS-1C image would be rectified using the modified rational function model. The reference rectified coordinates are known from an existing map 1:5000 scale. 52 points were detected in the image and also in the cadastral map. The coordinates from image and map were substituted in the previous equations and the best parameters (coefficients) of the rational function were obtained. These parameters were used in the rectification procedure. Some tests were carried out. The Ground Control Points (GCPs) and the test points are shown in table 1. Values of Z are simulated. The code of designed program for calculations is shown below.

Table 1
Ground control points and test points

GCP #	X image	Y image	X map5000	Y map5000	Z map5000
1	700	350	509753	943915	0.1
2	688	328	509718	943980	0.15
3	675	303	509678	944108	0.17
4	695	295	509773	944130	1
5	732	314	509943	943998	1.2
6	698	314	509778	944030	0.9
7	649	312	509533	944085	0.8
8	667	307	509630	944098	0.85
9	662	328	509578	943960	0.7
10	671	363	509603	943825	0.72
11	697	355	509740	943845	0.75
12	722	352	509863	943825	0.4
13	717	374	509815	943735	0.2
14	755	399	509980	943573	0.25
15	745	354	509980	943800	0.35
16	733	338	509928	943890	0.36
17	739	397	509908	943593	0.47
18	748	371	509975	943710	0.46
19	700	391	509718	943663	0.5
20	669	387	509575	943715	0.55
21	656	383	509510	943738	0.65
22	644	364	509465	943848	1.05
23	632	392	509380	943715	1.1
24	674	410	509573	943595	1.08
25	712	405	509763	943578	0.95
26	746	431	509910	943420	0.99
27	750	452	509908	943313	1.15
28	760	472	509943	943210	0.08
29	731	490	509783	943143	0.1
30	743	470	509858	943228	0.15
31	723	478	509750	943210	0.2
32	711	460	509710	943308	0.22
33	711	430	509738	943465	0.3
34	723	442	509785	943388	0.35
35	679	454	509555	943371	0.1
36	681	494	509528	943183	0.36
37	662	487	509443	943225	0.4
38	632	501	509288	943188	0.45
39	618	470	509243	943348	0.55
40	626	435	509313	943513	0.75
41	630	402	509365	943670	0.8
42	627	490	509270	943248	0.9
43	607	405	509253	943678	0.4
44	575	384	509110	943815	0.5

Table 1
Cont.

GCP #	X image	Y image	X map5000	Y map5000	Z map5000
45	577	405	509100	943710	0.6
46	572	438	509045	943553	0.15
47	595	468	509133	943383	0.2
48	569	519	508963	943150	0.11
49	574	331	509155	944070	0.8
50	605	389	509250	943763	1.1
51	620	378	509335	943800	0.8
52	543	316	509003	944165	0.6
Check Points	654	397	509488	943675	0.1
	714	393	509783	943640	0.15
	740	410	509890	943528	0.2
	729	420	509833	943485	0.35
	608	375	509273	943825	0.4
	663	465	509465	943333	0.6
	696	442	509650	943415	0.45

```
Function map(AG, data, n, data1, Ach1, data2)
MLputmatrix "AG", data
MLputmatrix "n", data1
MLputmatrix "Ach1", data2
mlevalstring "n1=size(Ach1,1)"
'mlgetmatrix "n1", "C5"
mlevalstring "Ach=[Ach1(:,2:4)]"
'mlgetmatrix "Ach", "A140"
'CASE 3 GCPS
mlevalstring
"A=[AG(:,2:4)];Lxm=[AG(:,5)];Lym=[AG(:,6)];Nafm=A'*A;AtLxm=A'*Lxm;AtLym=A'*Lym;Xm=Nafm\At
Lxm;Ym=Nafm\AtLym;Xm1=A*Xm;Ym1=A*Ym;EXm=Lxm-Xm1;EYm=Lym-
Ym1;EXms=EXm.*EXm;EYms=EYm.*EYm;MSEme=sqrt(EXms+EYms);MSEms=MSEme.*MSEme;s
MSEms=sum(MSEms);TMSEm=sqrt(sMSEms/52);Xm2=Ach*Xm;Ym2=Ach*Ym;EXm2=Ach1(:,5)-
Xm2;EYm2=Ach1(:,6)-
Ym2;EXms2=EXm2.*EXm2;EYms2=EYm2.*EYm2;MSEm2=sqrt(EXms2+EYms2);MSEms2=MSEm
2.*MSEm2;sMSEms2=sum(MSEms2);TMSEm2=sqrt(sMSEms2/7)"

mlgetmatrix "TMSEm", "K20"
mlgetmatrix "TMSEm2", "L20"
'polynomial Transformations
'seconed order
mlevalstring
"cxp2=[AG(:,2)];cx2p2=cxp2.*cxp2;cyp2=[AG(:,3)];cy2p2=cyp2.*cyp2;cxyp2=cxp2.*cyp2;c1p2=[AG(:
,4)];ACp2=[cy2p2 cx2p2 cxyp2 cyp2 cxp2 c1p2];AG1=[AG
ACp2];cxpp2=[Ach(:,1)];cx2pp2=cxpp2.*cxpp2;cypp2=[Ach(:,2)];cy2pp2=cypp2.*cypp2;cxypp2=cxp
p2.*cypp2;c1pp2=[Ach(:,3)];ACpp2=[cy2pp2 cx2pp2 cxypp2 cypp2 cxpp2 c1pp2]"
mlevalstring
"AA=[AG1(:,1:13)];A=AA(:,8:13);L1=AA(:,5);L2=AA(:,6);N1=A'*A;L3=A'*L1;L4=A'*L2;X1=N1\L3;Y1=N
1\L4;X2=A*X1;Y2=A*Y1;E1=L1-X2;E2=L2-
Y2;E3=E1.*E1;E4=E2.*E2;M1=sqrt(E3+E4);M2=M1.*M1;s1=sum(M2);T12nd=sqrt(s1/52);X3=ACp
```

```
p2*X1;Y3=ACpp2*Y1;E5=Ach1(:,5)-X3;E6=Ach1(:,6)-
Y3;E7=E5.*E5;E8=E6.*E6;M7=sqrt(E7+E8);M8=M7.*M7;s2=sum(M8);T22nd=sqrt(s2/7)"
mlgetmatrix "T12nd", "K22"
mlgetmatrix "T22nd", "L22"
'mlgetmatrix "AG1", "A80"
'third order
mlevalstring
"cxp3=[AG(:,2)];cx2p3=cxp3.*cxp3;cyp3=[AG(:,3)];cy2p3=cyp3.*cyp3;cxyp3=cxp3.*cyp3;c1p3=[AG(:
,4)];cx2yp3=cx2p3.*cyp3;cxy2p3=cxp3.*cy2p3;cx3p3=cx2p3.*cxp3;cy3p3=cy2p3.*cyp3;ACp3=[cy3
p3 cx3p3 cxy2p3 cx2yp3 cy2p3 cx2p3 cxyp3 cyp3 cxp3 c1p3];AG2=[AG
ACp3];cxpp3=[Ach(:,1)];cx2pp3=cxpp3.*cxpp3;cypp3=[Ach(:,2)];cy2pp3=cypp3.*cypp3;cxypp3=cxp
p3.*cypp3;c1pp3=[Ach(:,3)];cx2ypp3=cx2pp3.*cypp3;cxy2pp3=cxpp3.*cy2pp3;cx3pp3=cx2pp3.*cx
pp3;cy3pp3=cy2pp3.*cypp3;ACpp3=[cy3pp3 cx3pp3 cxy2pp3 cx2ypp3 cy2pp3 cx2pp3 cxypp3
cypp3 cxpp3 c1pp3]"
mlevalstring
"AA=[AG2(:,1:17)];A=AA(:,8:17);L1=AA(:,5);L2=AA(:,6);N1=A'*A;L3=A'*L1;L4=A'*L2;X1=N1\L3;Y1=N
1\L4;X2=A*X1;Y2=A*Y1;E1=L1-X2;E2=L2-
Y2;E3=E1.*E1;E4=E2.*E2;M1=sqrt(E3+E4);M2=M1.*M1;s1=sum(M2);T13rd=sqrt(s1/52);X3=ACpp
3*X1;Y3=ACpp3*Y1;E5=Ach1(:,5)-X3;E6=Ach1(:,6)-
Y3;E7=E5.*E5;E8=E6.*E6;M7=sqrt(E7+E8);M8=M7.*M7;s2=sum(M8);T23rd=sqrt(s2/7)"
mlgetmatrix "T13rd", "K24"
mlgetmatrix "T23rd", "L24"
'Rational Function
mlevalstring
"cxp3=[AG(:,2)];cyp3=[AG(:,3)];czp3=[AG(:,7)];cx2p3=cxp3.*cxp3;cy2p3=cyp3.*cyp3;cz2p3=czp3.*cz
p3;cxyp3=cxp3.*cyp3;cxzp3=cxp3.*czp3;cyzp3=cyp3.*czp3;cx3p3=cx2p3.*cxp3;cy3p3=cy2p3.*cyp
3;cz3p3=cz2p3.*czp3;cx2yp3=cx2p3.*cyp3;cx2zp3=cx2p3.*czp3;cxy2p3=cxp3.*cy2p3;cxyzp3=cxp3
.*cyzp3;cyz2p3=cyp3.*czp3;c1p3=[AG(:,4)];xm=[AG(:,5)];xcz3=xm.*cz3p3;xcyz2=xm.*cyz2p3;xcy2z=
xm.*cy2zp3;xcy3=xm.*cy3p3;xcxz2=xm.*cxz2p3;xcxyz=xm.*cxyzp3;xcxy2=xm.*cxy2p3;xcx2z=xm.*
cx2zp3;xcx2y=xm.*cx2yp3;xcx3=xm.*cx3p3;xcz2=xm.*cz2p3;xcyz=xm.*cyzp3;xcy2=xm.*cy2p3;xcx
z=xm.*cxzp3;xcxy=xm.*cxyp3;xcx2=xm.*cx2p3;xcz=xm.*czp3;xcy=xm.*cyp3;xcx=xm.*cxp3;xc1=x
m.*c1p3;Ax=[cz3p3 cyz2p3 cy2zp3 cy3p3 cxz2p3 cxyzp3 cxy2p3 cx2zp3 cx2yp3 cx3p3 cz2p3
cyzp3 cy2p3 cxzp3 cxyp3 cx2p3 czp3 cyp3 c1p3 xcz3 xcyz2 xcy2z xcy3 xcxz2 xcxyz xcxy2 xcx2z
xcx2y xcx3 xcz2 xcyz xcy2 xcxz xcxy xcx2 xcz xcy xc1]"
mlgetmatrix "cx2p3", "K26"
'mlevalstring
"cxp3=[AG(:,2)];cyp3=[AG(:,3)];czp3=[AG(:,7)];cx2p3=cxp3.*cxp3;cy2p3=cyp3.*cyp3;cz2p3=czp3.*cz
p3;cxyp3=cxp3.*cyp3;cxzp3=cxp3.*czp3;cyzp3=cyp3.*czp3;cx3p3=cx2p3.*cxp3;cy3p3=cy2p3.*cyp
3;cz3p3=cz2p3.*czp3;cx2yp3=cx2p3.*cyp3;cx2zp3=cx2p3.*czp3;cxy2p3=cxp3.*cy2p3;cxyzp3=cxp3
.*cyzp3;cyz2p3=cyp3.*czp3;c1p3=[AG(:,4)];ym=[AG(:,6)];ycz3=ym.*cz3p3;ycyz2=ym.*cyz2p3;ycy2z=
ym.*cy2zp3;ycy3=ym.*cy3p3;ycxz2=ym.*cxz2p3;ycxyz=ym.*cxyzp3;ycxy2=ym.*cxy2p3;ycx2z=ym.*
cx2zp3;ycx2y=ym.*cx2yp3;ycx3=ym.*cx3p3;ycz2=ym.*cz2p3;ycyz=ym.*cyzp3;ycy2=ym.*cy2p3;ycx
z=ym.*cxzp3;ycxy=ym.*cxyp3;ycx2=ym.*cx2p3;ycz=ym.*czp3;ycy=ym.*cyp3;ycx=ym.*cxp3;yc1=ym
.*c1p3;Ay=[cz3p3 cyz2p3 cy2zp3 cy3p3 cxz2p3 cxyzp3 cxy2p3 cx2zp3 cx2yp3 cx3p3 cz2p3 cyzp3
cy2p3 cxzp3 cxyp3 cx2p3 czp3 cyp3 c1p3 ycz3 ycyz2 ycy2z ycy3 ycxz2 ycxyz ycxy2 ycx2z ycx2y
ycx3 ycz2 ycyz ycy2 ycxz ycxy ycx2 ycz ycy yc1]"

'mlevalstring "Nx=Ax'*Ax;Atxm=Ax'*xm;x1=Nx\Atxm;x2=Ax*x1;Ex=xm-
x2;Ny=Ay'*Ay;Atym=Ay'*ym;y1=Ny\Atym;y2=Ay*y1;Ey=ym-
y2;Ex2=Ex.*Ex;Ey2=Ey.*Ey;M1=sqrt(Ex2+Ey2);M2=M1.*M1;s1=sum(M2);T1r=sqrt(s1/52)"
'mlevalstring
"cxpp3=[Ach(:,1)];cypp3=[Ach(:,2)];czpp3=[Ach(:,7)];cx2pp3=cxpp3.*cxpp3;cy2pp3=cypp3.*cypp3;cz
```

```
2pp3=czpp3.*czpp3;cxypp3=cxpp3.*cypp3;cxzpp3=cxpp3.*czpp3;cyzpp3=cypp3.*czpp3;cx3pp3=cx
2pp3.*cxpp3;cy3pp3=cy2pp3.*cypp3;cz3pp3=cz2pp3.*czpp3;cx2ypp3=cx2pp3.*cypp3;cx2zpp3=cx
2pp3.*czpp3;cxy2pp3=cxpp3.*cy2pp3;cxyzpp3=cxpp3.*cyzpp3;cyz2pp3=cypp3.*czpp3;c1pp3=[AG
(:,4)];Acpp3=[cz3pp3 cyz2pp3 cy2zpp3 cy3pp3 cxz2pp3 cxyzpp3 cxy2pp3 cx2zpp3 cx2ypp3
cx3pp3 cz2pp3 cyzpp3 cy2pp3 cxzpp3 cxypp3 cx2pp3 czpp3 cypp3
c1pp3];xa=[x1(1:20,1)];xb=[x1(21:40,1)];Acpp3xa=Acpp3*xa;Acpp3xb=Acpp3*xb;xmf=Acpp3xb\Ac
pp3xa;ya=[y1(1:20,1)];yb=[y1(21:40,1)];Acpp3ya=Acpp3*ya;Acpp3yb=Acpp3*yb;ymf=Acpp3yb\Acp
p3ya;Exc=Ach1(:,5)-xmf;Eyc=Ach1(:,6)-
ymf;Exc2=Exc.*Exc;Eyc2=Eyc.*Eyc;Mc=sqrt(Exc2+Eyc2);Mc2=Mc.*Mc;sc=sum(Mc2);T2r=sqrt
(sc/7)"
mlgetmatrix "T1r", "K26"
mlgetmatrix "T2r", "L26"
End Function
```

4. Data analysis

In this section some computations were performed to compare the potential of each transformation method for image rectification task and select the best method for this problem. For this purpose, fifty two GCPs were used from IRS – 1C image and map 1:5000 to perform this comparison between five different methods of transformation (Affine, 2^{nd} Polynomial, 3^{rd} Polynomial, Neural Network, and Rational Function Transformation). The mathematical procedures to perform this transformation for the other methods are found in detail in [3, 4].

By using Matlab, Excel, and Visual basic the comparison between all methods of transformations carried out, and the results are shown in table 2.

Table 2
RMSE for all methods of transformations

Transformation method	Check points RMSE (m)
Affine	5.95
2nd Polynomial	5.58
3rd Polynomial	5.26
Neural Network	4.51
Rational Function	3.23

5. Conclusions

Rational Functions have been used in the remote sensing community to replace the rigorous sensor models for two main reasons: firstly, because sensor models are kept confidential by vendors. Secondly, Rational Function provides a more accurate and fast method.

A new mathematical approach was developed and tested in this paper for solving the rational function to avoid the linearization of it. This method is described in section 2 and is based on matrix approach and least squares- solution.

This paper presents the rectification using a modified model of Rational function. The results of this way were compared by the results of other methods, like using neural networks, affine formulae and polynomial formulae. Using Rational function proved a better accuracy than the other methods. This is an advantage for it, but it has many unknowns, so it needs more points for getting the best parameters.

References

[1] Xutong Niu, Feng Zhou, Kaichang Di and Rongxing Li, "Geopositioning Accuracy Analysis Based on Integration of Quickbird and Ikonos Imagery", Mapping and GIS Laboratory, CEEGS, The Ohio State University li.282@osu.edu Commission I and IV (2006).

[2] Arie Croitoru, Vincent Tao, Yong Hu and Jeff Xu; "The Rational Function Model: A Unified 2d and 3d Spatial Data Eneration Scheme", GeoICT Laboratory, Department of Earth and Space Science and Engineering Faculty of Pure and Applied Science, York University 4700 Keele Street, Toronto, ON, Canada M3J 1P3 {arie, tao, yhu, zxu}@yorku.ca (2006).

[3] S. El-naghi, H. El-ghazouly, R. Khalil and A. El-naggar, (2004a), "Image Rectification Using Artificial Neural Network", Transportation Department, Faculty of Engineering, Alexandria University, 5th International Conference on Role of Engineering Towards a Better Environment, Alexandria, Egypt, 11-13 December, (2004).

[4] S. El-naghi, H. El-ghazouly, R. Khalil and A. El-naggar, (2004b), "Integration between Excel, Visual Basic, and Matlab for Selecting the best GCPs in map production (Data Sources are GPS, Remote Sensing, and an Existing map of Scale 1:5000", Transportation Department, Faculty of Engineering, Alexandria University, 5th International Conference on Role of Engineering Towards a Better Environment, Alexandria, Egypt, pp. 11-13 December (2004).

Received September 29, 2007
Accepted November 29, 2007