
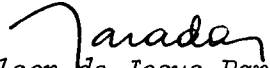


1. Publication Nº <i>INPE-3386-PRE/657</i>	2. Version	3. Date <i>Jan., 1985</i>	5. Distribution <input type="checkbox"/> Internal <input checked="" type="checkbox"/> External <input type="checkbox"/> Restricted
4. Origin <i>DIN/DPD</i>	Program <i>DENUME</i>		
6. Key words - selected by the author(s) <i>SPLINE INTERPOLATION IMAGE PROCESSING</i>			
7. U.D.C.: <i>621.376.5</i>			
8. Title <i>NOISE CORRECTION ON LANDSAT IMAGES USING A SPLINE-LIKE ALGORITHM</i>		10. Nº of pages: <i>07</i>	11. Last page: <i>06</i>
9. Authorship <i>Luiz Alberto Vieira Dias Nandamudi Lankalapalli Vijaykumar</i>		12. Revised by  <i>Sergio R.M. Pellegrino</i>	
Responsible author <i>N.L. Vijaykumar</i>		13. Authorized by  <i>Nelson de Jesus Parada Director General</i>	
14. Abstract/Notes <i>Many applications using LANDSAT images face a dilemma: the user needs a certain scene (for example, a flooded region), but that particular image may present interference or noise in form of horizontal stripes. During automatic analysis, this interference or noise may cause false readings of the region of interest. In order to minimize this interference or noise, many solutions are used, for instance, that of using the average (simple or weighted) values of the neighbouring vertical points. In case of high interference (more than one adjacent line lost) the method of averages may not suit the desired purpose. The solution proposed here is to use a spline-like algorithm (weighted splines). This type of interpolation is simple to be computer implemented, fast, uses only four points in each interval, and eliminates the necessity of solving a linear equation system. In the normal mode of operation, the first and second derivatives of the solution function are continuous and determined by data points, as in cubic splines. It is possible, however, to impose the values of the first derivatives, in order to account for sharp boundaries, without increasing the computational effort. Some examples using the proposed method are also shown.</i>			
15. Remarks <i>This work has been presented at IV Plenary Meeting of SELPER (Remote Sensing Latinamerican Experts Society), held at Santiago, Chile, on November, 12-16th, 1984.</i>			

NOISE CORRECTION ON LANDSAT IMAGES USING A SPLINE-LIKE ALGORITHM

Luiz Alberto Vieira Dias (1)

Nandamudi Lankalapalli Vijaykumar (2)

ABSTRACT

Many applications using LANDSAT images face a dilemma: the user needs a certain scene (for example, a flooded region), but that particular image may present interference or noise in form of horizontal stripes. During automatic analysis, this interference or noise may cause false readings of the region of interest. In order to minimize this interference or noise, many solutions are used, for instance, that of using the average (simple or weighted) values of the neighbouring vertical points. In case of high interference (more than one adjacent line lost) the method of averages may not suit the desired purpose. Immediate solutions that may be thought of are polynomial or spline interpolations. The polynomial interpolation has the disadvantage of presenting oscillations (depending on the polynomial degree), while, in the second method, one has to solve a linear system that may be large depending on the number of points and consequently takes more time. The solution proposed here is to use a spline-like algorithm (weighted splines). This type of interpolation is simple to be computer implemented, fast, uses only four points in each interval, and eliminates the necessity of solving a linear equation system. In the normal mode of operation, the first and second derivatives of the solution function are continuous and determined by the data points, as in cubic splines. It is possible, however, to impose the values of the first derivatives, in order to account for sharp boundaries, without increasing the computational effort. Some examples using the proposed method are also shown.

1. INTRODUCTION

In order to minimize the noise or interference in form of horizontal stripes in LANDSAT imagery, many solutions are used, like substitution of the stripe pixel values for the average (simple or weighted) values of the neighbouring vertical points. This solution may not suit if more than one adjacent line is lost or if the pixel value variation is not linear (as observed, the variations are seldom linear).

Sometimes the user needs a certain scene, and the particular images are noisy or with interference. It is interesting to have a nonlinear interpolator, that has not the disadvantages of the high degree polynomial interpolations (oscillations) or splines (need of solving a linear system of equations, thus making the calculations slow and expensive). The spline solution,

(1) INPE - Instituto de Pesquisas Espaciais - Av. dos Astronautas, 1758, Caixa Postal 515 - 12200 - São José dos Campos - SP - Brasil. Tel. (0123) 22-9977 Ext. 384. Computer Scientist.

(2) INPE - Instituto de Pesquisas Espaciais - Av. dos Astronautas, 1758, Caixa Postal 515 - 12200 - São José dos Campos - SP - Brasil. Tel. (0123) 22-9977 Ext. 291. Computer Scientist.

however, is technically very good, and if the cost/benefit is worth, maybe in certain applications one should really use the splines.

The solution proposed in this work is the weighted splines method (Costa, 1980; Pereira and Vijaykumar, 1984) that has the advantage of low computer cost since it does not have to solve a linear system of equations, produces a 3rd degree polynomial based on four data points calculated through a single formula and is very convenient to be computer implemented. As a further advantage, it is possible to impose the values of the first derivatives, in order to account for sharp boundaries, without increasing the computational effort or to keep the first and second derivatives of the solution function continuous as determined by the data points (Pereira and Vijaykumar, 1984). In the examples no sharp boundaries were considered.

The weighted spline method is described and a few examples are presented to illustrate it. The advantages and shortcomings are discussed in the conclusions.

2. WEIGHTED SPLINES METHOD

The interpolation in each interval i is obtained using only four points. Given n pair of points $\{x_i, y_i\}$ such that $i=1, 2, 3, \dots, n$ and $x_i < x_{i+1}$, the following hypotheses are assumed:

- (i) $f(x_i) = y_i$;
- (ii) $f'(x)$ and $f''(x)$ are continuous;
- (iii) in each interval $[x_i, x_{i+1}]$ $f(x)$ must be defined by a polynomial whose degree does not depend on n ;
- (iv) in each interval $[x_i, x_{i+1}]$, $f(x)$ must be defined only as a function of x_j and y_j , where $j=i-1, i, i+1, i+2$.

Initially, two second degree functions h_1 and h_2 are defined such that h_1 interpolates by x_{i-1}, x_i and x_{i+1} and h_2 interpolates by x_i, x_{i+1} and x_{i+2} . The following conditions are given to these functions:

$$h_1(x_{i-1}) = y_{i-1}; h_1(x_i) = y_i; h_1(x_{i+1}) = y_{i+1}. \quad (2.1)$$

$$h_2(x_i) = y_i; h_2(x_{i+1}) = y_{i+1}; h_2(x_{i+2}) = y_{i+2}. \quad (2.2)$$

Now consider a function $f(x)$ defined as

$$f(x) = P_1(x) h_1(x) + P_2(x) h_2(x), \quad (2.3)$$

where $P_1(x)$ and $P_2(x)$ are weight functions.

For $x=x_i$ and $x=x_{i+1}$ the function $f(x)$ in Equation 2.3 yields, respectively:

$$f(x_i) = (P_1(x_i) + P_2(x_i)) y_i,$$

$$f(x_{i+1}) = (P_1(x_{i+1}) + P_2(x_{i+1})) y_{i+1}.$$

From Equations 2.1 and 2.2 it is known that $f(x_i) = y_i$. With this condition the following can be obtained:

$$P_1(x_i) + P_2(x_i) = 1;$$

$$P_1(x_{i+1}) + P_2(x_{i+1}) = 1.$$

Deriving Equation 2.3, one may obtain:

$$f'(x) = P_1'(x) h_1(x) + P_1(x) h_1'(x) + P_2'(x) h_2(x) + P_2(x) h_2'(x). \quad (2.4)$$

In order to guarantee the continuity of the first derivative, the following has to be satisfied:

$$f'(x_i) = h_1'(x_i),$$

$$f'(x_{i+1}) = h_2'(x_{i+1}).$$

The following conditions satisfy the above:

$$P_1'(x_i) + P_2'(x_i) = 0, \quad (2.5)$$

$$P_1(x_i) = 1, \quad (2.6)$$

$$P_2(x_i) = 0, \quad (2.7)$$

$$P_1'(x_{i+1}) + P_2'(x_{i+1}) = 0, \quad (2.8)$$

$$P_1(x_{i+1}) = 0, \quad (2.9)$$

$$P_2(x_{i+1}) = 1. \quad (2.10)$$

For $f''(x)$, the second derivative of $f(x)$, the continuity must also be guaranteed and the following conditions are necessary:

$$P_1''(x_i) + P_2''(x_i) = 0, \quad (2.11)$$

$$P_1'(x_i) = 0, \quad (2.12)$$

$$P_2'(x_i) = 0, \quad (2.13)$$

$$P_1''(x_{i+1}) + P_2''(x_{i+1}) = 0, \quad (2.14)$$

$$P_1'(x_{i+1}) = 0, \quad (2.15)$$

$$P_2'(x_{i+1}) = 0. \quad (2.16)$$

Changing our interval to $[0,1]$, and changing the variable from x to z , the following relations hold for $x \in [x_i, x_{i+1}]$ and for $z \in [0, 1]$:

$$z = \frac{x - x_i}{x_{i+1} - x_i}$$

$$x_i < x_{i+1}$$

The above system of equations in z becomes:

$$P_1(0) = 1; P_2(0) = 0.$$

$$P_1(1) = 0; P_2(1) = 1.$$

$$P_1'(0) = 0; P_2'(0) = 0.$$

$$P_1'(1) = 0; P_2'(1) = 0.$$

$$P_1'(0) + P_2'(0) = 0.$$

$$P_1'(1) + P_2'(1) = 0. \quad (2.17)$$

$P_1(z)$ and $P_2(z)$ can be expressed as fourth degree polynomials:

$$P_1(z) = a_1z^4 + b_1z^3 + c_1z^2 + d_1z + e_1,$$

$$P_2(z) = a_2z^4 + b_2z^3 + c_2z^2 + d_2z + e_2. \quad (2.18)$$

Substituting Equation 2.18 in Equations 2.17 and solving the system, the following is obtained.

$$P_1(z) = \left(1 + \frac{t}{2}\right)z^4 - tz^3 + \left(\frac{t}{2} - 2\right)z^2 + 1,$$

$$P_2(z) = -\left(1 + \frac{t}{2}\right)z^4 + tz^3 - \left(\frac{t}{2} - 2\right)z^2,$$

where t is a parameter to be determined. It can be noticed that

$$P_1(z) = 1 - P_2(z) \text{ or } P_2(z) = 1 - P_1(z).$$

The Expression 2.3 can be written:

$$f(z) = P_1(z)h_1(z) + (1 - P_1(z))h_2(z),$$

which is the interpolating polynomial. It was demonstrated by Costa (1980) that if $t \in [-8, 4]$ the oscillations are avoided.

Returning from z to x , the final formula is

$$f(x) = P_1(x)h_1(x) + P_2(x)h_2(x),$$

where

$$P_1(x) = \left(1 + \frac{t}{2}\right) \left(\frac{x - x_j}{x_{i+1} - x_j}\right)^4 - t \left(\frac{x - x_j}{x_{i+1} - x_j}\right)^3 + \left(\frac{t}{2} - 2\right) \left(\frac{x - x_j}{x_{i+1} - x_j}\right)^2 + 1$$

$$P_2(x) = 1 - P_1(x)$$

$$t \in [-8, 4].$$

One can also notice that when $t = -2$, the above functions are reduced to third degree polynomials.

The functions $h_1(x)$ and $h_2(x)$ are:

$$h_1(x) = \frac{(x - x_j)(x - x_{i+1})}{(x_{i-1} - x_j)(x_{i-1} - x_{i+1})} y_{i-1} + \frac{(x - x_{i-1})(x - x_{i+1})}{(x_j - x_{i-1})(x_j - x_{i+1})} y_j +$$

$$+ \frac{(x - x_{i-1})(x - x_i)}{(x_{i+1} - x_{i-1})(x_{i+1} - x_i)} y_{i+1},$$

$$h_2(x) = \frac{(x - x_{i+1})(x - x_{i+2})}{(x_i - x_{i+1})(x_i - x_{i+2})} y_i + \frac{(x - x_i)(x - x_{i+2})}{(x_{i+1} - x_i)(x_{i+1} - x_{i+2})} y_{i+1} +$$

$$+ \frac{(x - x_i)(x - x_{i+1})}{(x_{i+2} - x_i)(x_{i+2} - x_{i+1})} y_{i+2}.$$

3. CONCLUSIONS

In order to test the method, a computer program was developed having as input a pixel matrix 8x8, with each pixel in the range 1(one) to 255 (two hundred and fifty-five). The level 0 (zero) was considered noise or interference.

Depending on the target, the weighted splines method can produce better or worse results. If the missing line is the upper or lower border, the averages method does not give results and the weighted splines method does. If there is more than one adjacent lines lost, again the averages method does not work, and the weighted splines method does.

To choose the better reconstruction, the elements of the vectors for the reconstructed lines were simply added, the lower the sum, the better the reconstruction.

Two examples were considered:

1) Using the artificial image below (pixel values from 1 to 255),

220	220	220	220	220	220	220	220
60	60	60	60	60	60	60	60
50	50	50	50	50	50	50	50
90	90	90	90	90	90	90	90
110	110	110	110	110	110	110	110
70	25	110	143	128	250	70	70
110	32	180	45	250	250	110	110
85	110	215	138	214	85	92	13

the lines 3(three) and 7(seven) were removed and substituted for zero (assumed to be noise), and then the average method was applied for reconstruction. The results were:

for line 3: 75 75 75 75 75 75 75 75
for line 7: 77 67 162 140 171 167 81 41,

and the sum of the line 3 produced 200, while the sum of the line 7 produced 441.

Applying the weighted splines method to the example, it was possible to obtain:

for line 3: 44 44 44 44 44 44 44 44
for line 7: 61 21 144 152 162 241 63 37,

and the sums for lines 3 and 7 resulted in 48 and 408, respectively.

In this case the weighted splines method showed to be better. It has to be said that line 3 was chosen such that average method did not work. However, for line 7 a random generation was used to produce the pixel values and, as one can notice, the weighted splines method was better.

2) Using a portion of a Thematic Mapper Image (this image belongs to Southern Bahia (Brazil) region, with oscillating pixel values),

64	71	64	74	59	51	59	69
59	59	59	56	59	48	51	69
59	56	61	59	61	61	59	71
61	61	71	64	61	59	69	71
69	59	64	64	66	61	59	59
94	94	94	66	74	74	71	61
145	94	74	59	87	105	107	102
59	48	38	54	87	102	107	105

the lines 3 and 7 were removed and substituted for zero. The following results were obtained using the averages method and weighted splines method:

Averages method:

line 3: 60 60 65 60 60 53 60 70
line 7: 76 71 66 60 80 88 89 83
sum of line 3: 21
sum of line 7: 159

Weighted Splines Method:

line 3: 57 58 65 56 59 52 60 72
line 7: 90 90 85 62 81 87 87 76
sum of line 3: 24
sum of line 7: 143

The averages method presented a better result for line 3, and the weighted spline method presented a better result for line 7.

Further tests have to be made for better and precise conclusions. However, it is expected that in near-random fluctuations of pixel variation, the two methods would yield similar results. In nonlinear variations (as in the case of most number of images), the weighted splines method would be better. In linear variations the averages method presents better results.

As far as computer effort is concerned, both the averages method and the weighted splines method present the same complexity (order n), although the former is slightly faster.

REFERENCES

- 1) COSTA, L.A.Z. da Spline Ponderado - Um novo algoritmo, In: II Simpósio sobre Aplicações Gráficas por Computador e Sistemas Gráficos Interativos, 1980, 2:445-464.
- 2) PEREIRA, J.A.G.; VIJAYKUMAR, N.L. Um interpolador rápido com especificação de inclinações, abril de 1984, São José dos Campos. (INPE-3058-PRE/458)

ADDITIONAL BIBLIOGRAPHY

- 1) COONS, S.A. Surfaces for Computer-Aided Design of Space Forms, MAC-TR-41, June 1967.
- 2) PEREIRA, J.A.G.; VIJAYKUMAR, N.L. Geração de superfícies livres por meio de "Splines" ponderados. Apresentado no 19 Congresso Latino-Americano de Automática/59 Congresso Brasileiro de Automática, Campina Grande, setembro de 1984.
- 3) VIJAYKUMAR, N.L. Geração de superfícies livres usando "Spline" ponderado. Dissertação de Mestrado. No prelo.