**Gross error detection in Digital Elevation model Using Genetic Algorithm and Bees Algorithm**

**F. Samadzadegana, M. Ghandeharib**

**a Dept. of Geomatics Engineering, University of Tehran, North Amirabad Street, Tehran, Iran -**

**samadz@ut.ac.ir**

**b Dept. of Geomatics Engineering, University of Tehran, North Amirabad Street, Tehran, Iran -**

**ghandehary@ut.ac.ir**

**ABSTRACT**

The detection and deletion of gross error of DEM data is a great concern in geospatial data analysis. In fact many statistical techniques are sensitive to the presence of gross errors, but there are many ambiguities in the results. This paper presents a new algorithm for detecting gross errors of irregular DEM data using Metaheuristic algorithms. We have used a classic and a modern tool of Artificial Intelligence (AI) in this research: The Genetic Algorithm (GA) and the Bees Algorithm (BA), respectively. The gross error methods we proposed are characterized by a common localization procedure: we examine the entire dataset by consideration iteratively only a small subset at a time; for each step we take into account the data belonging to a moving square window and we handle separately each data set. In each testing window we fit a bilinear surface to all surrounding points and estimate the residuals for each point. By the use of Genetic and Bee algorithms we try to minimum sum of squared residuals to towards detection of gross error point. Three simulation studies are conducted to verify our method. We compare the result of our blunder detection method with other implemented methods of geostatistical toolbox in ARC/INFO GIS software. The results demonstrated that the proposed method is more effective and feasible than those existing algorithms to detect single and cluster blunders with high accuracy and without any statistical techniques.

**KEY WORDS:** Digital Elevation Models (DEMs), Genetic algorithm, Bees foraging algorithm, Metaheuristic algorithms,Gross errordetection

**1. INTRUDUCTION**

The Digital Elevation Models (DEMs) is the basic data source for developing geospatial information model and the database of Geographic Information System (GIS). Many researchers’ attention had been brought to predict and evaluate the quality of DEMs (Tang, 2000). Some people employed error modeling to estimate the DEM quality level in order to inform judgments by users as to the suitability of the spatial data for specific tasks (Fisher 1998). Many investigators have demonstrated that the rate of uncertainty of the derived geospatial information is dependent on the quality of DEMs (Hunter 1997, Lee 1992, Fisher 1996). One of the criteria on DEM quality is its accuracy. The factors of affecting DEM data accuracy are classed into two types: the accuracy of the original data, such as the density of sampling and measurement errors, and the accuracy of interpolation, such as the methods of interpolation and terrain types. According to the property of the errors, the DEM errors can be classified into three types: (1) gross errors, (2) systematic errors, and (3) random errors. Gross errors are caused by carelessness or inattention, or mistake of the observer while using equipment. In some practice GIS projects, it is always assumed in the accurate estimation of DEMs that the gross errors are excluded, which means that the estimation of the accuracy of DEMs only considers the impacts of the random errors and systematic errors. This probably is because the detection of gross error in DEMs is difficult. However, the fact is that gross errors implicitly exist in DEM data, as a result, distorting the image of the spatial variation present in DEMs. In a severe case, totally undesirable and unacceptable results for practice GIS project may be produced due to the existence of such gross errors in DEMs. So it is essential to make a study of methods for detecting and correcting DEM gross errors. The related effort can be back to Hannah (1981), who proposed gross errors detection devices for regular DEM data based on a grid point comparison with its neighboring points. Parameters, slope values, and change-slope values are used as threshold values in order to detect conflictive data points. Östman (1987) pointed out that there exists no unique criteria or a single measure for improving the “quality” of DEMs. The accuracy in elevation, slope and also curvature, at least, should be considered. Felicísimo (1994) analyzed the differences between the elevation of a grid point and its value interpolated from the neighbors. He supposed the differences belong to a Gaussian distribution, and sets a statistical value of the differences by means of a standard Student t test. His experimental results demonstrated that some outlying values could be unmasked through large values of the statistic. The developed algorithms for detecting DEM gross errors are basically sorted into (1) method based on slope, and (2) method based on point wise. The former is suitable for grid DEM data type, and the latter is suitable for the irregular data type. The method 1 takes slope as a basic attribute of a point on the ground and assumes the continuous terrain of the ground, i.e., The major part of ground smoothly varies in elevation, and any point occurring sharp discontinuity in the elevations or sudden changes in the surface slopes are suspected of being in error. This method contains three procedures: firstly, define the slope between point and its eight neighbors; secondly, define the Slope Change (SC); thirdly, define the Difference in Slope Change (DSC) (Hannah, 1981). The algorithm based on the slope information is designed for grid-based DEMs. If the original DEMs is irregular, e.g., Lidar data, it will have to be interpolated into a grid form prior to the application of this method. Obviously, if the point contains gross error in original data, interpolated grid data will be affected by the gross error, which means it is difficult to detect the gross error. On the other hand, the statistical values of hypothesis test in the computational process of gross error detection are statistically correlated because they are from the same point source. Therefore, the algorithm is not appropriate for the irregular data set. In fact many statistical techniques are sensitive to the presence of gross errors but there are many ambiguities in the result.[6] This paper presents a new algorithm for detecting gross errors of irregular DEM data based on Metaheuristic search concepts. We have used a classic and a modern tool of Artificial Intelligence (AI) in this research: The Genetic Algorithm (GA) and the Bees Algorithm (BA), respectively. In this method after dividing the region into sub region, we search each part separately. In each local area (testing window) we fit a bilinear surface to all points and compute the residuals vector, by the use of Genetic and Bee algorithms we try to minimum sum of squared residuals to towards detection of gross error point. In our preprocessing method an automatic procedure to optimize window size is set up. Three simulation studies are conducted to verify our method. We compare the result of our blunder detection method with other implemented methods of geostatistical toolbox in ARC/INFO GIS software. The simulated results demonstrated that the proposed method is more effective and feasible than those existing algorithms to detect single and cluster blunders with high accuracy and without any statistical techniques.

**2. APPLIED STRATEGIES**

In this section, three methods of gross error detection are explained. First genetic algorithm, then bees foraging algorithm. Finally we give a brief explanation on global and local gross error detection in ARC GIS software:

**2.1. GENETIC ALGORITHM**

Genetic algorithm is a solution to complex optimization problems considered as an evolutionary computation method. The idea first came out at 1970's (Holland, 1975). It describes the problem by a set of parameters; interprets them as artificial genes and the genes as blueprints of individuals. Then it applies evolution rules to individuals and saves better fitting ones to the next generation. In general form, GA works as follows. It is initialized by a randomly generated population of solutions and their fitness to the objective function is evaluated. During a recursive process, best genes of the population are selected out, recombined and mutated to form the members of a new generation. Applying recombination and mutation is the way the algorithm avoids local optima instead of true global one (Engelbrecht, 2007). GA pseudo code is represented in figure 1.

|  |
| --- |
|  |
| Figure 1. Genetic algorithm pseudo code |

In case of our gross error removal problem, we initially select a population of proper size randomly via a uniformly distributed PDF. Each chromosome had a fixed length equal to the number which was defined by the user. The length should be more than number of blunder in a testing window. Each gene in each chromosome was an integer number between 0 and number of existing points in a local area which showed the selected points that are volunteers for our blunder detection test. This kind of encoding made some problem in reproduction step, like duplicated gene in one chromosome. To avoid these problems, a constrained crossover operator developed. In the case of constrained crossover operator, the offspring made by recombination of parents checked for probable repetitive genes. If there were any repetitive genes, crossover repeated again with another crossover mask. If this task does not fix the problem, the parents are changed. The algorithm is repeated until the best fitness is satisfied or the last generation is arrived.

In order to select parents for the next generation, we applied roulette wheel selection method. The method considers a slot for each solution on the roulette wheel proportional to its fitness value; the fitter the solution, the larger the slot. Then necessary solutions are randomly picked up from the wheel as parents for next generation. The fitness function we consider to do the task is sum of squared residuals in least squares solution of points in that local area that fit with a bilinear surface.

Note that for each chromosome, the points in that chromosome are not selected to fit function because the fitness function became minimum when there isn’t any blunder in that chromosome. The fitness function defined as:

(1)

(2)

In the above equations, V is residuals vector and C is experimental cofactor. By this cofactor the value of fitness is independent of the number of points in a chromosome.

There are several crossover operators which one point crossover is used here. Mutation operator chooses some gens randomly and changed their value to zero. In this research we use an inventive operator that has an important role. This operator makes new chromosome by use of residuals vector. Points that have larger residual have more chance to create these new chromosomes.

Finally the new population is ready to be evaluated. This recursive process is done until the stopping criteria are met. The elite of such population is the final solution to the gross error detection problem.

**2.2. BEES FORAGING ALGORITHM**

Swarm Intelligence employs the collective behaviors in animal societies to design optimization algorithms. Description of the collective intelligence based on bees’ behavior was first performed by (Yonezawa et al., 1996). Methods of artificial bee colony optimization consist of three major algorithms: honey bee mating, honey bee pollination and bee foraging. We applied the last one, since its characteristics fit our application. Bee colonies in nature consist of different sorts of bees. Regarding foraging, scout bees are responsible for food searching. They randomly search some food patches and evaluate their quality. When returned to the hive, they inform other bees of quality of each patch through Woggle dance. This makes followers to go back to those patches with higher quality at each sweep and finally reach the best patch. What goes on BA is much like the scenario in nature (Karaboga et al., 2008). The whole process is briefly explained in figure 2 via its pseudo code.

|  |
| --- |
|  |
| figure 2. Bees foraging algorithm pseudo code |

The parameters to be determined in this method are listed in table 1.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Parameter | Description | | n | Population size | | m | Best selected out patches | | e | Elite patches out of m selected patches | | n1 | Searches around (m-e) patches | | n2 | Searches around e patches | | ngh | Radius for neighbourhood search | | imax | Maximum number of iterations | |
| Table 1. Description of Bees foraging parameters |

Concerning our gross error application, initially, a population of size "n" is randomly generated from the dataset. Each bee (agent) has a dimension between 0 and k that k defined by the user. K should be more than the number of blunders in a testing window. This means if there isn’t any blunder, the bee that has zero dimensions should have the best fitness or if there are m blunder, the bee that has m dimensions should have the best fitness. Then like GA fitness function are computed for each bee. Next, the elites and other better sites are chosen according to a predefined number of elites and betters. Now n2 bees are sent to the neighborhood of elite site and n1 bees are recruited around the other better sites. However In this method neighborhoods have not any spatial relation. The combination of points with more residual are used to create neighborhoods. Then the fitness value for all n2 (n1) is computed and the bee with the elite (better site) is replaced by the bee with better fitness value. In the next step, other scout bees (n-m) with no proper fitness again scattered in search of space randomly. This task prevents trapping in local optima. These processes repeated until stopping criteria is met. In this research stopping criterion is defined as no change in elite’s fitness in some iteration. Best member of such population is the final output of the algorithm.

**2.3. LOOKING FOR GLOBAL AND LOCAL GROSS ERRORS IN ARC GIS SOFTWARE**

Global gross error is a measured sample point that has a very high or a very low value relative to all of the values in a dataset. For example, if 99 out of 100 points have values between 300 and 400, but the 100th point has a value of 750, the 100th point may be a global gross error .

A local gross error is a measured sample point that has a value that is within the normal range for the entire dataset, but, if you look at the surrounding points, it is unusually high or low. For example, the diagram below is a cross section of a valley in a landscape. However, there is one point in the center of the valley that has an unusually high value relative to its surroundings, but it is not unusual compared to the entire dataset.[11]

|  |
| --- |
|  |
| Figure 3.local gross error |

**2.3.1. LOOKING FOR GROSS ERRORS THROUGH THE HISTOGRAM**

The [Histogram tool](mk:@MSITStore:C:\Program%20Files%20(x86)\ArcGIS\Help\geostatistical_analyst.chm::/histogram.htm) enables you to select points on the tail of the distribution. The selected points are displayed in the ArcMap data view. If the extreme values are isolated locations (for instance, surrounded by very different values), they may require further investigation and be removed if necessary[2].

|  |  |
| --- | --- |
|  |  |
| figure 4. looking for gross errors through the histogram | |

**2.3.2. LOOKING FOR GROSS ERRORS THROUGH VORONOI MAPPING**

A Voronoi diagram divides (or tiles or tessellates) the space into areas (cells) around each sample point that are shaped so that the borders of the regions are equidistant from the two nearest points. All the unsampled points inside the cell are closer to the sample point in the cell than any other sample points outside the cell itself.

Voronoi maps, colored using the cluster and entropy methods, can be used to identify possible gross errors.

Entropy values provide a measure of dissimilarity between neighboring cells. In nature you would expect that things closer together are more likely to be more similar than things farther apart. Therefore, local gross errors may be identified by areas of high entropy.

The cluster method identifies those cells that are dissimilar to their surrounding neighbors. You would expect the value recorded in a particular cell to be similar to at least one of its neighbors. Therefore, this tool may be used to identify possible gross errors[2].

|  |  |
| --- | --- |
|  |  |
| figure 5. looking for gross errors through voronoi mapping | |

**2.3.3. LOOKING FOR GROSS ERRORS THROUGH TREND ANALYSIS**

The Trend Analysis tool provides a three-dimensional perspective of the data. The locations of sample points are plotted on the x,y plane. Above each sample point, the value is given by the height of a stick in the z dimension. The unique feature of the Trend Analysis tool is that the values are then projected onto the x,z plane and the y,z plane as scatter plots. This can be thought of as sideways views through the three-dimensional data. Polynomials are then fit through the scatter plots on the projected planes. An additional feature is that you can rotate the data to isolate directional trends through the values. There are many other features that allow you to rotate and vary the perspective of the whole image, change size and color of points and lines, remove planes and points, and select the order of the polynomial that is to fit the scatter plots[2].

|  |  |
| --- | --- |
|  |  |
| figure 6. looking for gross errors through trend analysis | |

**3. RESULTS AND DISCUSSIONS**

**3.1 DATA SETS**

This strategy has been successfully tested both on synthetic examples as well as on actual cases. In order to evaluate the efficiency of the new algorithm, three irregular DEMs in which represent three typical terrains: smooth terrain, hilly area, and mountain area, are simulated. The parameters used for experiments are listed in Table 1. Assuming that the original data are error-free.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | |  | Minimum height | Maximum height | Mean height | Number of points | | Smooth terrain | 1 | 30 | 18.32 | 20000 | | Hilly area | 1 | 300 | 138.35 | 20000 | | Mountain area | 1 | 2000 | 845.09 | 20000 | |
| Table 2. Characteristics of data sets |

**3.2 PARAMETERS SETTING**

Proposed metaheuristic methods have their own specific parameters that have direct effects on the performance of blunder detection process. In this research work, the optimum values of these parameters are defined through trial and error strategies by considering the best performance of other investigation in similar literatures.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | | Algorithm | Parameters | Value | | GA  Algorithm | Maximum iteration  Population size  Elite probability  Crossover probability  Mutation probability | 30  20  0.1  0.2  0.6 | | BEE  Algorithm | Maximum iteration (imax)  Population size (N)  Best selected out patches (M)  Elite patches (E)  Searches around (m-e) patches (N1)  Searches around e patches (N2) | 30  20  10  6  4  6 | |
| Table 3. Parameters of GA and BEE algorithms |

As shown in Fig.4, BA established a faster functionality rather than GA. The x-axis and

Y-axis shows the number of iterations and fitness values, respectively.

|  |
| --- |
|  |
| Figure 7. Convergence plot, BEE ( left) and GA (right) |

**3.3. RESULTS**

In this section, we discuss the results obtained by performing three methods of gross error detection: Genetic algorithm, Bees foraging algorithm and described methods of geostatistical toolbox of ARC GIS software.

Here we show the gross error detection result for a massive irregular elevation data set.

|  |
| --- |
|  |
| Figure 8. Gross error detection result of a massive irregular elevation data set |

By using the Trend Analysis function, we can visualize the results in 3D, which is a more intuitive way and it can be used for check the result of other gross error detection methods.

In Voronoi mapping only some gross errors are correctly detected, also other points, that are not gross error, are detected.

We randomly select about 1% of total number of data points and assign them intentional gross errors with magnitude of 2‐20 times larger than the original mean height of the area. The selected data points as well as the assigned errors are recorded in order to compare with the results of the error detection process. In table 4 and 5 we present a sensible results to detect single a cluster blunder and make a comparison for different test areas and different methods.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | |  |  | Proposed methods | | ARC GIS software | | | | Genetic | BEE | Histogram | Voronoi mapping | Trend analysis | | No. Of correctly detected error points | Smooth terrain | 30 | 30 | 26 | 21 | 19 | | Hilly area | 29 | 30 | 22 | 20 | 12 | | Mountain area | 27 | 28 | 20 | 15 | 13 | | No. Of incorrectly detected error points | Smooth terrain | 0 | 0 | 3 | 10 | 4 | | Hilly area | 2 | 1 | 4 | 14 | 2 | | Mountain area | 3 | 2 | 5 | 15 | 2 | | N0. Of undetected | Smooth terrain | 0 | 0 | 4 | 9 | 11 | | Hilly area | 1 | 0 | 8 | 10 | 18 | | Mountain area | 3 | 2 | 10 | 15 | 17 | |
| Table 4. Results for single gross error detection in different test areas and different methods. |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | |  |  | Proposed methods | | ARC GIS software | | | | Genetic | BEE | Histogram | Voronoi mapping | Trend analysis | | No. Of correctly detected error points | Smooth terrain | 49 | 50 | 44 | 33 | 30 | | Hilly area | 47 | 47 | 40 | 29 | 31 | | Mountain area | 45 | 47 | 38 | 27 | 28 | | No. Of incorrectly detected error points | Smooth terrain | 1 | 1 | 5 | 13 | 6 | | Hilly area | 3 | 3 | 7 | 15 | 5 | | Mountain area | 4 | 3 | 9 | 21 | 7 | | No. Of undetected | Smooth terrain | 1 | 0 | 6 | 17 | 20 | | Hilly area | 3 | 3 | 10 | 21 | 19 | | Mountain area | 5 | 3 | 12 | 23 | 22 | |
| Table 5. Results for Cluster gross error detection in different test areas and different methods. |

It is illustrated from table 4 and Table5 that performance of detecting gross error will decrease with increasing terrain undulating. But the terrain undulating have negligible effect on our proposed methods.

The maximum number of gross errors, which can be correctly detected, is estimated as 50‐80% of the total number of gross errors existed in the DTM source data, but our methods support detecting of more than 95% of gross errors.

Figure 9 compares the performance of proposed methods that discussed in this article.

|  |
| --- |
|  |
| Figure 9. Performance of proposed methods that discussed in this article |

Most gross error detection methods use Statistical tests. A popular test which has widespread use is to reject an observation if its absolute residual exceeds three times the standard deviation of the observations. To represent the deficiency of statistical methods we simulate an area with 50 points and the residual vectors evaluated for four different states. (figure10, 11, 12, 13)

As you can see the choice of optimal threshold value are rather difficult to achieve. We didn’t use any statistical parameters in our method.

|  |  |
| --- | --- |
|  |  |
| Figure 10. Simulated area with 50 points and four gross errors | Figure 11. Simulated area with 50 points and three gross errors |
|  |  |
| Figure 12. Simulated area with 50 points and two gross errors | Figure 13. Simulated area with 50 points and one gross errors |

As future work, we decide to use local search and k-order voronoi diagram to improve the performance of our methods and reduction of computation cost because it is essential for practical applications of this method to a large and dense data set.

The GIS technique is also useful in analyzing the spatial pattern of anomalous points and identifying anomalous regions. The location and spatial pattern of gross errors may provide important clues for reasoning potential causes for outlying observations. The random, clustered, or linear pattern of anomalous data often corresponds to Different error causes. If anomalous data points are randomly distributed, they are most likely caused by random recording errors or white noise of measuring instruments. If they are clustered in some regions, the gross errors may be related to some systematic errors, for instance, the malfunctioning of measuring instruments in a special environment, miscalculations, edge effects when assembling smaller data sets into a big data set, or a similar type of circumstance. Identification of each cause surely renders an opportunity for improving the quality of the data-gathering process.

Access to GIS software is also very easy, either using freeware options (GRASS) or commercial ones (ARC GIS). Geographic Information Systems are one of the fastest growing markets in software today (Anon 1994). That implies that more people have access to proper tools, and then are able to manipulate and produce data. The first objective of this research has been to develop and test automatic methods suitable to help GIS data users and data producers to find as many errors as possible in their datasets at minimum effort with best performance. We will implement our error detection method in the ARC/INFO GIS software in future works. The integration leads to a powerful exploratory data analysis tool for checking and analyzing anomalous values in a GIS environment.

**4. CONCLUSIONS**

From our investigation, the following conclusions can be drawn up:

1. The experimental results demonstrate that Capability of our algorithm in detecting gross error for irregular DEMs more efficient than the other existing methods, especially for cluster blunders.
2. Hypsography of terrain is an important factor of affecting the efficiency of gross error detection, but our result showed that the proposed algorithms are robust.
3. The rate of gross error cannot affect the efficiency of our algorithms, although the efficiency of other algorithms is relevant to rate of gross error.
4. The methods proposed can be applied not only for single blunders; It has very good result for cluster blunders as well.
5. The Bees Algorithm generally outperformed other techniques that were compared with in terms of speed and accuracy.
6. By using the Trend Analysis function, we can visualize the results in 3D, which is a more intuitive way and it can be used for check the result of other gross error detection methods.

**4. REFRENCES**

1. C. Lopez, On the improving of elevation accuracy of Digital Elevation Models: a comparison of some error detection procedures, Scandinavian Research Conference on Geographical Information Science (ScanGIS), Stockholm, Sweden,

(1997) 85.

1. http://resources.arcgis.com/
2. Hunter, G J and Goodchild M F. 1997. Modeling the uncertainty of slope and aspect estimates derived from spatial databases. Geographical Analysis, vol. 29-1, pp. 35-49.
3. Lee, J., Snyder P K, and Fisher P F. 1992. Modeling the effect of data errors on feature extraction from digital elevation models. Photogrammetric Engineering and Remote Sensing, vol. 58-10, pp. 1461-1467.
4. M. A. Broveli , D. Triglione and G. Venuti. GIS techniques for digital surface models outlier detection, DIIAR- Politecnico di Milano- Facoltà di Como, via Valleggio 11, 22100 Como
5. Minyi Cen, Xiaoyun Yang, Liya Gu and Guoqing Zhou. Gross error detection of irregular dem data for gis error propagation analysis. ASPRS 2005 Annual Conference
6. Randy L. Haupt, Sue Ellen Haupt,Practical Genetic Algorithm,WILEY,2004
7. Tang, G. 2000. A Research on the Accuracy of Digital Elevation Model. Science Press, Beijing, New York, 2000.
8. Tran Quoc Binh.On the detection of gross errors  in digital terrain model source data, VNU Journal of Science, Earth Sciences 23 (2007) 213-219
9. Z.L. Li, Sampling Strategy and Accuracy Assessment for Digital Terrain Modelling, Ph.D. thesis, The University of Glasgow, 1990.
10. Z.L. Li, Q. Zhu, C. Gold, Digital terrain modeling: principles and methodology, CRC Press, Boca Raton, 2005.