INVESTIGATING THE EFFECT OF PSO ALGORITHM ON REDUCING CONTROL POINTS IN CAMERA CALIBRATION

Seyed Arvin Fakhri¹*, Seyed Arya Fakhri²

¹ Department of Photogrammetry and Remote Sensing, Faculty of Geodesy and Geomatics Engineering, K. N. Toosi University of Technology, Tehran, Iran – arvin.fakhri@email.kntu.ac.ir

² School of Surveying and Geospatial Engineering, College of Engineering, University of Tehran, Tehran, Iran

Commission VI, WG VI/4

KEY WORDS: PSO, Calibration, non-metric camera, close range, control point, collinearity equation

ABSTRACT:

Over the past decades, non-metric cameras have been utilized in functions with less accuracy requirement or even in precise works with the progress of photogrammetric cameras developed technology. One of the reasons to use these kinds of cameras is due very much to their lower costs in comparison with the metric ones. Since we have always error in measurements, there is no exception in Photogrammetry, which is more in the non-metric cameras than the metric ones. Some of these errors are systematic and the only way to cope with them is to model. So far, many models have been proposed to investigate and modify the behavior of errors. Some of these models are linear and some others are non-linear. The number of parameters in each model is different based on the complexity of error in each image. Since the picture need to be connected to earth to calculate the calibration parameters of the camera, therefore this connection is generally made through points known as the ground control points, and more of these control points are needed in the complex models with more sophisticated calculations. Using a method to reduce the need for less control points and achieving a suitable accuracy is beneficial due to the high cost and time-consuming process of preparing the control points. One of the methods that could be used in solving the calibration equations is the Particle Swarm Optimization (PSO) algorithm. Images of a few targets are captured in this research by a non-metric camera and the collinearity equations are used by adding further correction terms in order to calibrate the camera. The results of PSO method are compared with the classical mathematical methods in each step by reducing the number of control points, which indicated that the performance of using the PSO algorithm is better than the conventional proration methods in reducing the number of points and could be utilized in projects suffering from the lack of control points.

1. INTRUDUCTION

Photogrammetric images, like other observations, have errors. One of the sources of this error can be camera-related errors (Remondino, Fraser, & Sciences, 2006). In order to communicate between the image and the ground, it is necessary to identify and model the error behaviour in order to predict and model the error handling. How to model these errors is one of the issues we have always been looking for, and we have always been looking for a precise method despite its simplicity. Therefore, in any photogrammetric project, camera calibration is important. Camera calibration means estimating the internal and external parameters of the camera model. The solution to the calibration problem is to use the relationships between the 3D coordinates of space, camera, and image to extract these parameters. Many methods have been introduced for calibration, each depending on the type of system requirements, imaging environment, accuracy, speed and facilities. The methods are usually based on the pattern dimensions used in the method (Duane, 1971; Fraser & Sensing, 2013; Luhmann, Fraser, Maas, & Sensing, 2016; Z. J. I. T. o. p. a. Zhang & intelligence, 2000). In this section, we will briefly review the performance of methods and then discuss computational and artificial intelligence methods.

1.1 An Overview of Calibration Methods

The first batch uses basic camera calibration methods, which are traditional or classic methods, and use a 3D object that has orthogonal plates (such as a box) and has small squares like a chessboard on their faces. The algorithms presented for this method are of two types (Z. Zhang, 1999): 1- Direct calculation of parameters, 2- Using projection matrix

The third category is the use of one-dimensional objects, which are actually thin and tall objects such as television antennas, rulers and thin wood (Z. Zhang, 2004). The fourth category of calibration methods is self-calibration or non-dimensional calibration, only by calibration between surrounding images (Ji & Dai, 2004). We will need to calibrate each project separately. Therefore, Calibration on the Job and Self-Calibration methods will be more important in close range photogrammetry (Brown, 1976).

1.1.1 Direct Linear Transformation (DLT): The mathematical model of the DLT is considered as the basic equation of the calibration predecessor as the object space and image linker (Zhizhou & Scholar). The mathematical equation of the DLT, along with the additional parameters used for external calibration and justification, is formulated as equations 1 and 2 (Chen, 1997):

^{*} Corresponding author

$$x = -\Delta x + \frac{L_1 X + L_2 Y + L_3 Z + L_4}{L_9 X + L_{10} Y + L_{11} Z + 1}$$
⁽¹⁾

$$y = -\Delta y + \frac{L_5 X + L_6 Y + L_7 Z + L_8}{L_9 X + L_{10} Y + L_{11} Z + 1}$$
(2)

Where:

$$\begin{split} \Delta x &= a_1 x + a_2 y + a_3 x^2 + a_4 x y + a_5 y^2 \\ &+ \frac{x}{r} (c_1 x^2 + c_2 x y + c_3 y^2) \quad (3) \\ &+ x (k_1 r^2 + k_2 r^4 + k_3 r^6) \\ &+ p_1 (y^2 + 2 x^2) + 2 p_2 x y + \delta_{x_0} \\ &+ \frac{x}{c} \delta_c \end{split}$$

$$\begin{split} \Delta y &= b_1 x + b_2 y + b_3 x^2 + b_4 x y + b_5 y^2 \\ &+ \frac{y}{r} \left(c_1 x^2 + c_2 x y + c_3 y^2 \right) \quad (4) \\ &+ y (k_1 r^2 + k_2 r^4 + k_3 r^6) \\ &+ 2 p_1 x y + p_2 (x^2 + 2 y^2) + \delta_{y_0} \\ &+ \frac{y}{c} \delta_c \end{split}$$

$$r &= \sqrt{x^2 + y^2} \end{split} \tag{5}$$

Where X, Y and Z are the three-dimensional coordinates of the point in the object space in the photo coordinate system. By L_1 , ..., L_{11} eleven transformation parameters establish the relation between the object point and the image point. Extra parameters, in addition to accurately calculating external justification parameters, model systematic errors or mathematical model defects for a non-metric camera. These parameters bring the equations closer to the mathematical equations of an ideal camera in order to correct the mathematical equations of a non-metric camera. These parameters are defined as the relationship of 3 and 4 (Zhizhou & Scholar). By linearizing Equations 1 and 2 with additional parameters and 11 unknown DLT transforms and solving least squares, the external justification parameters and camera calibration parameters are calculated.

1.1.2 Collinearity equations: In analytical photogrammetry collinearity equations are used to communicate the object space and the photographic space. The mathematical model of the collinearity equations along with the additional parameters is a suitable mathematical model for correcting systematic errors and increasing the accuracy of the bundle adjustment. The mathematical model is described in Equations 5 and 6. $x - x_0 + \Delta x_0$

$$= c_x \frac{(X_j - X_{oi})m_{11} + (Y_j - Y_{oi})m_{12} + (Z_j - Z_{oi})m_{13}}{(X_i - X_{oi})m_{31} + (Y_i - Y_{oi})m_{32} + (Z_i - Z_{oi})m_{33}}$$
(5)

$$y - y_o + \Delta y_p$$

$$= c_y \frac{(X_j - X_{oi})m_{21} + (Y_j - Y_{oi})m_{22} + (Z_j - Z_{oi})m_{23}}{(X_i - X_{oi})m_{31} + (Y_i - Y_{oi})m_{32} + (Z_i - Z_{oi})m_{33}}$$
(6)

Where Δx_p and Δy_p are corrections along the x and y axes.

$$\Delta x_p = \delta r_x + \delta p_x + \delta g_x \tag{7}$$

$$\Delta y_p = \delta r_y + \delta p_y + \delta g_y \tag{8}$$

Various types of additional parameters are proposed (Brown, 1976) and (Moniwa, 1977) and so on.

Generally, the calibration parameters of a camera (except internal orientation) are:

$$\delta_{r_x} = (x - x_0) \times (k_1 r^2 + k_2 r^4 + k_3 r^6 + \dots + k_n r^{2n})$$
⁽⁹⁾

$$\delta_{r_y} = (y - y_0) \times (k_1 r^2 + k_2 r^4 + k_3 r^6 + \cdots$$
(10)

$$r = \sqrt{(x - x_o)^2 + (y - y_o)^2}$$
(11)

Where k_1 , k_2 to k_n are the radial distortion parameters of the lens.

Tangential Distortion Model:

$$\delta p_x = p_1 [r^2 + 2(x - x_0)^2] \tag{12}$$

$$+ 2p_2(x - x_0) \times (y - y_0)$$

$$\delta p_y = p_2[r^2 + 2(y - y_0)^2]$$
(12)

$$+ 2p_1(x - x_0) \times (y - y_0)$$
 (13)

Where $p_1 \mbox{ and } p_2$ are the parameters of the tangential lens distortion.

Affinity Distortion Model:

$$\delta g_x = A(x - x_0)$$

$$\delta g_{\rm y} = {\rm B}({\rm y}-{\rm y}_0) \tag{15}$$

Where A and B are the parameters of the film dimension change along the x and y axes. By omitting the higher terms, we will have:

$$\Delta x_{p} = (x - x_{0}) \times (k_{1}r^{2} + k_{2}r^{4} + k_{3}r^{6}) + p_{1}[r^{2} + 2(x - x_{0})^{2}] + 2p_{2}(x - x_{0}) \times (y - y_{0}) + A(x - x_{0}) \Delta y_{p} = (y - y_{0}) \times (k_{1}r^{2} + k_{2}r^{4} + k_{3}) + p_{2}[r^{2} + 2(y - y_{0})^{2}] + 2p_{1}(x - x_{0}) \times (y - y_{0}) + B(y - y_{0})$$
(17)

In each step, a computation is performed to adjust the unknown values using control points. One of the most common methods of adjustment is the least squares method.

1.2 Least Squares adjustment

The concept of least squares estimation was introduced by Legendre and Carl Friedrich Gauss in the early 1800s (Bretscher). Based on the least square's method, since observations and measurements always have errors, we cannot arrive at the actual amount of observations. In this case, by minimizing sum of the square of the errors, we arrive at a suitable approximation for the observations (Leon, Bica, & Hohn, 1980). The least squares problem can generally be defined as a set of linear equations. Given the set Ax = b with dimensions $m \times n$ such that m > n we cannot generally expect to obtain the vector $x \in \mathbb{R}^n$ of the equation Ax = L. Instead, we can

(18)

(14)

(16)

reach \boldsymbol{x} where $\boldsymbol{A}\boldsymbol{x}$ is close to L. In relation 18 this difference is calculated.

V = L - Ax

The difference between Ax and b is as follows: (19) |L - Ax| = |V|

We want to set the vector x such that |V| Minimize. Minimize |V| It is equivalent to minimizing $||V||_2$. The estimated value of the unknowns is denoted by x, and the answer is obtained by the least square's method (Boyd & Vandenberghe, 2015).

1.3 Artificial Intelligence Based Techniques

Optimization is an important part of our daily lives. Many engineering and other science issues have parameters that aim to provide the most desirable result (Van Den Bergh, 2001). The purpose of optimizing is to define the best solution to bound problems. In the past decades, several researchers have proposed solutions for linear and nonlinear optimizations (Konar, 2006). Nowadays, using classical methods and computational intelligence has many applications in optimization problems. One of the methods of computational intelligence is the particle swarm algorithm.

Particle Swarm Optimization: The particle swarm 1.3.1 algorithm, which originates from social-psychological theories, is used as a powerful solution to nonlinear and multidimensional problems (Kennedy & Eberhart, 1997). The initial idea of particle swarm was from Kennedy (social psychologist) and Eberhart (electrical engineer) whose goal was to produce computational intelligence using simple theories of social interaction rather than purely individual cognitive abilities. A number of scientists have created simulations based on interpretations of birds and fish. The initial particle swarm algorithm (Eberhart & Kennedy, 1995) evolved from simulations of social behavior of birds described by (Reynolds, 1987) and (Heppner & Grenander, 1990). In PSO, members are known as particles that search for a multidimensional space to find the desired location (Sierra & Coello, 2005). The PSO algorithm works by repeatedly searching for a region based on its previous best particle success, its previous best neighbor success, its current particle position and its previous velocity (Mendes, Kennedy, & Neves, 2004). The concept of particle swarm optimization is that at each time step, the velocity of each particle shifts to pbest and gbest, and the acceleration is weighted at random with pbest and gbest (Eberhart & Kennedy, 1995). The variable g is defined as the value of the index of the best current performance of any particle in the neighborhood. Therefore, the original version of particle motions is represented by (Kennedy & Eberhart, 1997).

PSO algorithm flowchart

Figure 1 illustrates the six steps the particles are trying in the PSO algorithm to find appropriate answers in the search space.



Figure 1. The PSO algorithm flowchart *Parameters*

The basic PSO described above has a few parameters that need to be determined. One of these parameters is population size, which is often empirical and depends on the size and complexity of the problem. Parameters ϕ_1 and ϕ_2 are defined by random forces for the best personal p_i and the best p_g in the neighborhood, and are often referred to as acceleration coefficients. The behavior of a PSO varies substantially with values of ϕ_1 and ϕ_2 . In the initial PSO studies, the value was $\phi_1 = \phi_2 = 2.0$ (Das, Abraham, & Konar, 2008).

• Contraction coefficient: In 2002, Clerc and Kennedy introduced a compatible PSO model that uses a new parameter called the Contraction coefficient. It also ignores the weight of inertia w and the maximum velocity V_{max} (Ratnaweera, Halgamuge, & Watson, 2004). (20)

$$\begin{cases} \overrightarrow{v_{i}} \leftarrow \chi(\overrightarrow{v_{i}} + \overrightarrow{U}(0, \phi_{1}) \otimes (\overrightarrow{p_{i}} - \overrightarrow{x_{i}}) + \overrightarrow{U}(0, \phi_{2}) \otimes (\overrightarrow{p_{g}} - \overrightarrow{x_{i}})) \\ \overrightarrow{x_{i}} \leftarrow \overrightarrow{x_{i}} + \overrightarrow{v_{i}} \end{cases}$$

Where:

$$\phi = \phi_1 + \phi_2 > 4 \tag{21}$$

$$\chi = \frac{2}{\phi - 2 + \sqrt{\phi^2 - 4\phi}} \tag{22}$$

When using the Clerc's control method, the value of ϕ is usually 4.1. $\phi_1=\phi_2$ and the constant coefficient χ is about 0.7298. Therefore, the previous velocity is multiplied by 0.7298 and each of the two (p - x) is multiplied by 1.49618 \approx 0.7298 \times 2.05.

• Inertia weight: The inertia weight w controls the movement of the particles (Das et al., 2008). (Shi & Eberhart, 1998) have had several research papers on w that show that when V_{max} is not small (<3) an inertial weight of 0.8 is a good choice.

• **Maximum speed:** The maximum velocity (V_max) means the maximum number of changes in its position during each iteration.

• **Papulation size:** A common method in PSO is to limit the number of particles to 20 to 60 particles (Kennedy

& Eberhart, 1997; Shi & Eberhart, 1998). Eberhart and Shi (1998) showed that particle size has little effect on PSO performance.

• **Reflection method:** The reflection method reflects the position of a particle that crossed the search space boundary. (Mostaghim, Halter, & Wille, 2006). Figure 2 illustrates this idea in a two-dimensional space.



Figure 2. Reflection - A particle from the excluded range is then moved to a new position inside the search space.

$$\begin{split} &\text{if } X_{t+1} > \ X_{max} \longrightarrow X'_{t+1} = X_{max} - (X_{t+1} - X_{max}) \\ &\text{if } X_{t+1} < \ X_{max} \longrightarrow X'_{t+1} = X_{min} + (X_{min} - X_{t+1}) \end{split}$$

1.4 Artificial Neural Networks

Artificial neural networks are a mathematical model of the biological system (Sharma, 2012). Which is basically inspired by the human nervous system (Dayhoff, 1990). Nervous systems are an algorithm for optimizing and learning freely based on concepts inspired by research into the nature of the brain. The brain organizes its structural components using a feature known as a neuron, making it perform much more computationally faster than a digital computer (Katiyaar & Sharma).

Neuron

The neuron is the smallest unit of an artificial neural network that forms the function of neural networks (Fausett, 1994).



Figure 3. The structure of a neuron

The artificial neural network acts as a function that captures the number of input neurons and outputs the number of output neurons (Hagan, Demuth, Beale, & De Jesús, 1996). Artificial neural networks are a way of approximating functions and predicting the future states of a system. This feature of neural networks is used in many applications (Chansarkar, 2000).

Artificial neural network architecture

The neural network consists of at least one input layer and one output layer. Depending on the complexity of its architecture, it contains one or more hidden layers. The input layer is userdefined data that enters the neural network. The output layer is where the final results of the network are sent. Neural networks can have different architectures. A widely used architecture is back-propagation. (Merwin, Cromley, Civco, & Science, 2002).

Weight

The weights in the artificial neural network are estimated using training algorithms. This algorithm randomly selects the initial weights and compares the calculated output with the actual output. The difference between the calculated output and the actual output is estimated using the mean squared error. After all observations have been trained on the network, the observed error rate is distributed among the different nodes to keep the network error rate low (Nguyen & Widrow, 1990).

2. MATERIAL AND METHOD

2.1 Image Specification

Figure 4 is used to evaluate the different calibration methods proposed in this study. To this end, Target's terrain coordinates were first obtained by the Australis program, assuming that these points were error-free, 20% of them were eliminated for evaluation, and the remainder of the model training process was performed.



Figure 4. Study image

2.2 Artificial Neural Network Modeling

Multilayer perceptron neural network in MATLAB software was used for geometric modeling of image. In this process, data inputs are image points with x and y coordinates, and data outputs are ground 3D points with X, Y, and Z coordinates. According to Figure 5 the architecture of this network is that it has two layers with tansig and purelin transfer functions.



Figure 5. Artificial neural network architecture in this study

For network training, 60% of data were used for training data, 40% for data validation. In the first step, each of the validation and training data was randomly selected by the network and then maintained in the next steps. And this structure was used every time to evaluate the amount of network error and performance. In the next step, which was to obtain the appropriate number of neurons, the number of neurons was calculated by try and error. It was evaluated for the number of neurons from 2 to 30 neurons. To do this, each network architecture was trained 20 times and the RMSE value of the test data was calculated each time. Because the bias weights and components in each case are determined by the network itself, it may be inaccurate at some stages. For this purpose, at the end of the network learning process, for each situation, the 10 RMSEs that had the highest values were discarded, and thus in each network architecture, 10 RMSEs were obtained that were most similar to the actual RMSE value of that architecture. Other network settings are as follows:

Train Function: trainIm Performance Function: mse Maximum Epochs:1000 Performance Goal:0 Minimum Gradient: 1e-07 Maximum Validation Checks:6 Mu:0 .001 Mu Decrease Ratio:0 .1 Mu Increase Ratio:10 Maximum mu: 1e+10

2.3 Modeling using the PSO method

The cost function

Due to the least squares method, the objective is to estimate the unknowns by minimizing the squares of the errors. Therefore, the cost function used in the PSO algorithm is to minimize the value of $V^T W V$. Here V is calculated by putting the DLT equation on one side. Since the value is not equal to zero, the second soft of this difference is introduced as a cost function. Weights were also taken for all observations constant and equal to one.

The number of decision variables

Here, since the objective is to estimate the two-dimensional coordinates of points, so the number of decision variables is obtained from the following relation: 2x11 = number of decision variables

The range of variables

Determining the range of variables in the use of PSOs to estimate the coordinates of points depends on the amount of distortion and error in the image. Since the parameters L4 and L8 are related to the transfer, these two parameters are always slightly higher than the other parameters. Taking a range for all parameters causes it to never converge. Therefore, in this research, a technique was used to make convergence very fast. For this purpose, each of the above coefficients was scaled by the parameter k to fit the other coefficients.

Maximum iterations and population size

These parameters are empirical and can be obtained by applying the algorithm several times.

Inertia weight

As mentioned earlier, studies show that by applying the χ parameter to the inertia weight, the algorithm performs better.

Coefficients C1 and C2

According to studies in this field, its values can be obtained through the following relationships: $\phi_1 = \phi_2 = 2 C_1 = \chi \times \phi_1 C_2 = \chi \times \phi_2$

W_{damp}

This value is empirical as well, with the algorithm running several times to obtain its proper value.

Maximum and minimum speed

The appropriate value of maximum velocity in this paper is calculated according to the following relation:

 $V_{max} = k(Var_{max} - Var_{min})$

Where k is an experimental value. Reflective methods have also been used to prevent problems from moving particles out of range.

3. RESULT AND DISCUSSION

Initially the coordinates of each point were controlled by the CAD output of the Australis program.



Figure 6. Location of control points in the space Cad

Then the corresponding image and ground coordinate values were obtained. The following results were obtained by evaluating the models by test points.

Coefficient	: L1	L2	L3	L4	L5	L6
		-	-			-
value	0.263329	0.25813	0.00256	258.9444	0.077049	0.07685
L7	L8	L9	L10	L11		
0.001407	76.62051	0.001027	-0.001	0.0000059	1	

Table	1.	Estimation	of	DLT	unknown	coefficients	with	least
square	s s	olution						

Coefficient	L1	L2	L3	L4	L5	L6
		-	-			-
value	0.263329	0.25813	0.00256	258.9502	0.077049	0.07685
L7	L8	L9	L10	L11		
			-			
0.001407	76.62052	0.001027	0.00095	0		

Table 2. Estimation of DLT unknown coefficients by PSO

Method	LS	PSO	ANN		
RMSE	0.57	0.58	7		
THE DISE I GUILING					

Table 3. RMSE value of test points

Also, these methods were calculated in different states of the number of points, the results of which are shown in Figure 1.



Figure 7. RMSE of methods with variable number of points

As can be seen in the case of the number of control points greater than 1, all three methods have the same efficiency. And then the neural network shows a high dependence on the number of control points. At fewer points, the PSO algorithm shows better robustness. This is an indication of its strength against the number of points. The above results show that the estimation of DLT equation coefficients by PSO method is very accurate and in complex equations, the PSO method is very suitable. The results of artificial neural networks showed that this model has a lower accuracy than the DLT method but overall its accuracy is good. Due to the high dependency of artificial neural networks on the number of training points, some applications may not be economically viable.

REFERENCES

Boyd, S., & Vandenberghe, L. J. R. D., October. (2015). Vectors, matrices, and least squares (Working Title). 25.

Bretscher, O. Linear Algebra with Applications.(2005). In: Prentice Hall, NJ.

Brown, D. (1976). The bundle adjustment-progress and prospect. Paper presented at the XIII Congress of the ISPRS, Helsinki, 1976.

Chansarkar, M. J. G. S. (2000). Neural networks in GPS navigation. 4(2), 14-18.

Chen, F.-J. (1997). Application of least-squares adjustment technique to geometric camera calibration and photogrammetric flow visualization.

Das, S., Abraham, A., & Konar, A. (2008). Particle swarm optimization and differential evolution algorithms: technical analysis, applications and hybridization perspectives. In Advances of computational intelligence in industrial systems (pp. 1-38): Springer.

Dayhoff, J. E. (1990). Neural network architectures: an introduction: Van Nostrand Reinhold Co.

Duane, C. B. J. P. E. (1971). Close-range camera calibration. 37(8), 855-866.

Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. Paper presented at the Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on.

Fausett, L. V. (1994). Fundamentals of neural networks: architectures, algorithms, and applications (Vol. 3): Prentice-Hall Englewood Cliffs.

Fraser, C. S. J. P. E., & Sensing, R. (2013). Automatic camera calibration in close range photogrammetry. 79(4), 381-388.

Hagan, M. T., Demuth, H. B., Beale, M. H., & De Jesús, O. (1996). Neural network design (Vol. 20): Pws Pub. Boston.

Heppner, F., & Grenander, U. J. T. u. o. c. (1990). A stochastic nonlinear model for coordinated bird flocks. 223.

Ji, Q., & Dai, S. (2004). Self-calibration of a rotating camera with a translational offset. IEEE Transactions on Robotics

Automation, 20(1), 1-14.

Katiyaar, V., & Sharma, I. K. Use of Data Mining & Neural Network in Commercial Application.

Kennedy, J., & Eberhart, R. C. (1997). A discrete binary version of the particle swarm algorithm. Paper presented at the Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on.

Konar, A. (2006). Computational intelligence: principles, techniques and applications: Springer Science & Business Media.

Leon, S. J., Bica, I., & Hohn, T. (1980). Linear algebra with applications: Macmillan New York.

Luhmann, T., Fraser, C., Maas, H.-G. J. I. J. o. P., & Sensing, R. (2016). Sensor modelling and camera calibration for closerange photogrammetry. 115, 37-46.

Mendes, R., Kennedy, J., & Neves, J. J. I. t. o. e. c. (2004). The fully informed particle swarm: simpler, maybe better. 8(3), 204-210.

Merwin, D. A., Cromley, R. G., Civco, D. L. J. C., & Science, G. I. (2002). Artificial neural networks as a method of spatial interpolation for digital elevation models. 29(2), 99-110.

Moniwa, S. (1977). Advanced photogrammetric system with self-calibration and its applications. Ph. D. Dissertation, Department of Surveying Engineering, UNB, Fredericton,

Mostaghim, S., Halter, W., & Wille, A. J. S. o. o. c. s. (2006). Linear multi-objective particle swarm optimization. 31, 209-237.

Nguyen, D., & Widrow, B. (1990). Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. Paper presented at the Neural Networks, 1990., 1990 IJCNN International Joint Conference on.

Ratnaweera, A., Halgamuge, S. K., & Watson, H. C. J. I. T. o. e. c. (2004). Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients. 8(3), 240-255.

Remondino, F., Fraser, C. J. I. A. o. P., Remote Sensing, & Sciences, S. I. (2006). Digital camera calibration methods: considerations and comparisons. 36(5), 266-272.

Reynolds, C. W. (1987). Flocks, herds and schools: A distributed behavioral model. Paper presented at the ACM SIGGRAPH computer graphics.

Sandmann, D. (2013). Design and Implementation of a precision Three-dimensional binocular Image tracker for departing aircraft.

Sharma, I. K. (2012). Use of Data Mining & Neural Network in Medical

Industry. Lecturer B.N College of Engineering & Technology, 3.

Shi, Y., & Eberhart, R. (1998). A modified particle swarm optimizer. Paper presented at the Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on.

Sierra, M. R., & Coello, C. A. C. (2005). Improving PSO-based multi-objective optimization using crowding, mutation and∈dominance. Paper presented at the International Conference on Evolutionary Multi-Criterion Optimization.

Van Den Bergh, F. (2001). An analysis of particle swarm optimizers. University of Pretoria South Africa,

Zhang, Z. (1999). Flexible camera calibration by viewing a plane from unknown orientations. Paper presented at the Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on.

Zhang, Z. (2004). Camera calibration with one-dimensional objects. J IEEE transactions on pattern analysis

machine intelligence, 26(7), 892-899.

Zhang, Z. J. I. T. o. p. a., & intelligence, m. (2000). A flexible new technique for camera calibration. 22.

Zhizhou, W. J. B., Press of Wuhan Technical University of Surveying, & Scholar, M. G. Principles of photogrammetry. 1990.