

Modeling of a Stereo Vision System using a Genetic Algorithm based Fuzzy Linear Regression

Hiaw San, Kho, Chee Peng, Lim*, Zalina Abdul Aziz, Anwar Hasni Abu Hassan
 School of Electrical and Electronic Engineering
 University of Science Malaysia, Engineering Campus
 14300 Nibong Tebal, Penang, Malaysia
 Email (corresponding author*): cplim@eng.usm.my

Abstract

In this paper a fuzzy linear regression (FLR) model integrated with a genetic algorithm (GA) is proposed. The proposed GA-FLR model is applied to modeling of a stereo vision system. A set of empirical data from stereo vision object measurement is collected based on the full factorial design technique. Three regression models, namely ordinary least-squares regression (OLS), FLR, and GA-FLR, are developed, and with their performances compared. The results show that the proposed GA-FLR model performs better than OLS and FLR in modeling of a stereo vision system.

Keywords—Fuzzy Linear Regression, Genetic Algorithm, Stereo Vision, Range Finder, Factorial Design

1. Introduction

Empirical data modeling is a common approach used by researchers to understand the relationship between input factors and output variables of a system under investigation. Traditionally the ordinary least-squares (OLS) regression method is used to approximate the true function of a system or process. The descriptive model is a crisp polynomial function, which can be applied only if the underlying statistical model assumptions are satisfied; e.g. the normality of error terms and the predicted value, and the equality of variances [1]. In many real world problems, the variables under consideration do not always distribute as stated in statistical properties and the systems tend to be complex. Violation of the assumptions implies an invalid model, which may not be able to precisely describe the investigated system.

Fuzzy regression analysis has emerged as an alternative for modeling of vague or imprecise systems where the application of statistical regression is not appropriate [2]. Recently, application of Fuzzy Linear Regression (FLR) has been reported in [3], [4], [5].

In fuzzy regression, the deviation between the actual value and the predicted value is said to be owing to the uncertainty naturally present in the system. The data samples are distributed based on possibilistic theory, instead of probabilistic theory. The predicted response from fuzzy regression is assumed to be a range of possible value rather than a crisp value. The estimated

regression coefficient is a fuzzy function coefficient [6].

In this study, we develop a system modeling method using the FLR model and apply it to a stereo vision object distance measurement system. We compare two types of FLR model, i.e. FLR using Peters' approach [7] and our proposed Genetic Algorithm (GA)-based FLR. The performance of the measurement system is evaluated using R^2 , and the results from OLS, FLR, and GA-FLR are compared and discussed.

This paper is organized as follows. Section 2 describes FLR and GA. Section 3 presents the integration of GA and FLR. The hardware and software design of the stereo vision system is described in section 4. Details of the experiment and the variables involved are explained in section 5. Validation experiments are also carried out to demonstrate the effectiveness of the proposed method for system modeling. Performance comparison of three modeling approaches is summarized and discussed. Finally, a summary of concluding remarks and further work is presented in section 6.

2. Concept of FLR and GA

2.1 Fuzzy Linear Regression with Fuzzy Interval

A general fuzzy linear function [8] is expressed by:

$$Y = A_0x_0 + A_1x_1 + A_2x_2 + \dots + A_nx_n = Ax \quad (1)$$

where $(x_0 := 1)$, $x = [x_1, x_2, \dots, x_n]^T$ is a crisp vector of input factors, $A = [A_0, A_1, A_2, \dots, A_n]$ is a fuzzy vector of the model coefficients. Fuzzy function A_j is a function of two parameters, α and c , denoted by

$$A_j = (\alpha_j, c_j), j = 0, 1, \dots, n \quad (2)$$

with $\alpha = [\alpha_0, \alpha_1, \dots, \alpha_n]^T$ and $c = [c_0, c_1, \dots, c_n]^T$, where α_j is the central value of the fuzzy number and c_j is the spread from the central value of the fuzzy number, denotes the fuzziness of the function.

From equation (2), a revised version of the fuzzy linear regression model can be derived, as follows

$$Y = (\alpha_0, c_0) + (\alpha_1, c_1)x_1 + (\alpha_2, c_2)x_2 + \dots + (\alpha_n, c_n)x_n \quad (3)$$

Tanaka et al [2] has proposed to solve equation (3) by formulating it as a linear programming (LP) problem. The objective of solving the LP problem is to minimize the total vagueness of the system, in order to estimate the

fuzzy coefficients, (α_j, c_j) . However, the major problem of Tanaka's approach is that it is too sensitive to outliers, resulting in an interval estimation of the dependent variable that is too wide for accurate prediction. Consequently, this model might lead to a wrong conclusion on the investigated system.

Peters [7] proposed a fuzzy linear regression model with fuzzy intervals to overcome the weakness of Tanaka's model. As described in [7], the following fuzzy linear programming formulation is derived to approximate the fuzzy coefficients, (α_j, c_j) :

$$\begin{aligned} \text{Max } \bar{\lambda} &= \frac{1}{M} \sum_{i=1}^M \lambda_i \\ \text{s.t. } (1 - \bar{\lambda})p_0 - \sum_{i=1}^M \sum_{j=0}^N c_j |x_{ij}| &\geq -d_0 \\ (1 - \lambda_i)p_i + \sum_{j=0}^N \alpha_j x_{ij} + \sum_{j=0}^N c_j |x_{ij}| &\geq y_i, \\ (1 - \lambda_i)p_i - \sum_{j=0}^N \alpha_j x_{ij} + \sum_{j=0}^N c_j |x_{ij}| &\geq -y_i, \\ -\lambda_i &\geq -1 \end{aligned} \quad (4)$$

$\alpha \in \mathfrak{R}$, $c_j, \lambda \geq 0$, $x_{i0} = 1$, $i = 1, 2, \dots, m$, $j = 0, 1, 2, \dots, n$, where, λ_i represents the membership degree to which the given data sample belongs to the set of 'good solution'. The λ_i value is restricted to $[0, 1]$, and d_0 is the desired value of the objective function, p_0 is the tolerance of the desired lower bound, p_i represents the width of the tolerance interval of data, y_i , and $\bar{\lambda}$ is the mean of all λ_i . The width of the estimated interval depends on the selection of parameters, d_0 , p_0 , and p_i . The solution of equation (4) returns the values of (α_j, c_j) and λ_i . In other words, the values of d_0 , p_0 , and p_i have to be provided in order to use equation (4). According to Peters [7], d_0 is set to zero ($d_0 = 0$) in order to obtain a model as crisp as possible, so that the desired value of the total vagueness function, $\sum_{j=0}^N c_j |x_{ij}| = 0$. The selection of p_0 and p_i is

context-dependent and is carried out on a trial and error basis. For simplicity, in [7], it is suggested to fix $p_1 = p_2 = p_3 = \dots = p_m$. The reason is the dimension of p_i is dependent on the number of data samples. For example, if a given data set contains 36 pairs of input-output samples, $[(x_1, y_1); (x_2, y_2); \dots; (x_{36}, y_{36})]$, there are 36 p_i values (p_1, p_2, \dots, p_{36}) to be decided for equation (4). This is time consuming and is not practical to determine those p_i values by trial-and-error. By following the approach in [7], each trial only has to decide two input parameters instead of 37 parameters.

There are several shortcomings associated with the FLR model in [7]. The user has to intuitively set p_0

and p_i without guidance. Therefore, it is necessary to assess the goodness of fit of the FLR model to the given data and the predictive performance by means of measurement index, such as R^2 , or Mean Square Error (MSE). Another problem is that it is difficult to justify the satisfactory level of the resulting model with respect to a given data set. Many trials with different p_0 and p_i values have to be carried out, and the goodness of fit to the respective data set are uncertain.

The performance of the estimated FLR model is greatly dependent on the proper selection of p_0 and p_i parameters. It is vitally important to introduce an effective method to help set the parameters automatically. Therefore, we propose to employ a GA to determine p_0 and p_i in FLR automatically. Besides that, we suggest to use R^2 as a measurement index for the FLR model evaluation.

2.2 Genetic Algorithms

A GA is a search technique whose inspiration is from natural biological evolution, with the purpose of finding true or approximate solution to optimization problems. The GA starts from a population of randomly generated potential solutions, and evolves into a final fittest generation as the optimal solution. In the GA, a chromosome is represented as a string, containing characteristic of a proposed solution. Each individual value in the string is known as gene. A collection of chromosome forms a population. The number of chromosome in a population is referred as the population size.

There are two common methods of chromosome encoding, namely binary encoding and continuous (real-number) encoding. The encoding method is decided based on problem domain. An objective function of the problem is required to be transformed into the fitness function, which relates each chromosome with the corresponding fitness value. The selection function is served as a tool to choose suitable parents from a population, for mating, with reference to the fitness value. One of the widely-used selection methods is the roulette wheel selection. In this method, the fitness value is transformed into a probabilistic value. The probability value is obtained by dividing the fitness value with the total fitness value in the population. Therefore, fitter chromosomes have better chances of being selected [9].

After selection, reproduction techniques such as crossover and mutation are used during evolution. Crossover exchanges a specify portion in a chromosome within a pair of chromosome, enabling the algorithm to extract the best genes from different chromosome and recombine them as a potentially superior children. Mutation randomly changes the genes of the selected chromosome, allowing random search. The reproduction process creates more chromosomes than the initial population. Since most GA models maintain a fixed-

sized population, replacement of population is necessary and biased replacement, such as the elitism replacement strategy, is usually adopted to keep the fittest chromosome continually survive through the next generation. The cycle of selection, reproduction, and population replacement continues until one of the following criteria is met:

- The population has converged
- The best fitness value is achieved
- The pre-defined maximum number of generations is achieved
- There is no improvement of fitness value for a specify interval of time

3. Integration of GA and FLR

In this section, we use the GA to determine the optimum p_0 and p_i values automatically. In addition, our approach allows variations in p_i values, or $p_1 \neq p_2 \neq p_3 \neq \dots \neq p_m$. The GA iteration employed in our work is presented in Figure 1.

Each solution of p_0 and p_i combination is represented as a single chromosome in the GA. In our work, continuous encoding is employed. The number of gene in each chromosome depends on the number of data sample plus one (as p_0 is counted in the string). The selection of p_0 and p_i values is generated by the GA random mechanism. From the first generation of population, the GA automatically evolves the solution of p_0 and p_i values towards the optimum. The optimum decision criterion is set by maximizing the $R^2_{prediction}$ value. In other words, $R^2_{prediction}$ is set as the fitness function in the GA iteration. Table 1 summarizes the GA configuration in our study. Determination of p_0 and p_i values is restricted to positive real numbers.

Table 1 GA configuration

Description	Setting
Coding of genes	Real number (positive value only)
Population size	20
Population initial range	1 to 1000
Number of genes	According to number of data sample + 1
Selection strategy	Roulette-Wheel selection
Replacement strategy	Elitism replacement strategy; (Elite count=2)
Crossover	Scattered-point crossover
Crossover rate	70%
Mutation	(Random mutation) uniform mutation
Mutation rate	1%
Maximum generation	100
Convergence	Maximum generation or population convergence
Population convergence	No improvement for 50 continuous generation

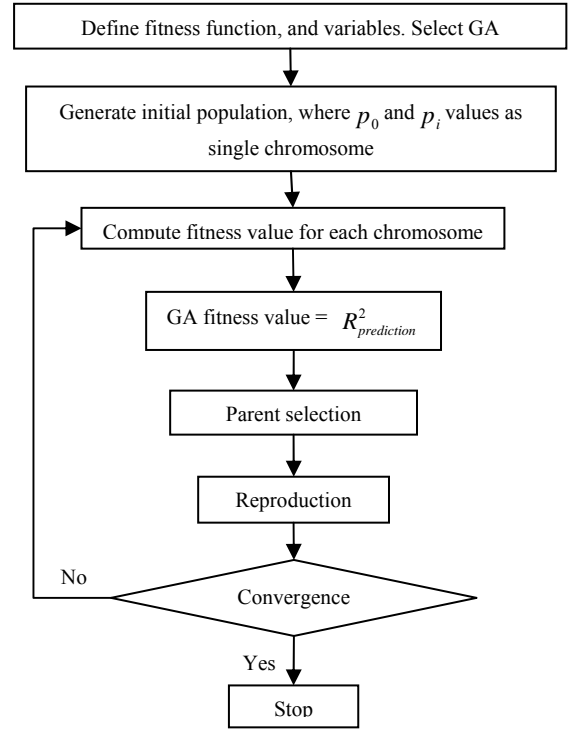


Fig. 1. Flowchart of the GA iteration

3.1 Model Performance Measurement Index

The fitted model and the prediction model are assessed using the R^2 value. The R^2 value is calculated as follows

$$R^2 = 1 - \frac{SSE}{SST} \quad (5)$$

$$SSE = \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

$$SST = \sum_{i=1}^N (y_i - \bar{y})^2$$

where, SSE is the error sum of squares, SST is the total sum of squares, y_i is the actual response value, \hat{y}_i is the predicted value for the respective y_i , and \bar{y} is the average of all y_i .

Note R^2 can take on any value between 0 and 1, with a value closer to 1 indicates a better fit. The fitted model is measured by R^2_{fit} ; on the other hand, the validation of the model is measured by $R^2_{prediction}$. R^2_{fit} is a statistic index that measures how successful the fit is in explaining the variation of the given data, while $R^2_{prediction}$ provides an indication of the predictive capability of the model constructed.

4. Stereo Vision Design

A case study on object measurement of a stereo vision system was conducted. The stereo vision system design was divided into two, i.e., software design and hardware design, as follows.

4.1. Hardware Design

4.1.1 Camera Specification

The stereo vision system consists of two webcams from the same manufacturer as the left and right cameras. The specifications of both the webcams are as follows

- (a) Resolution: 640 x 480 pixels
- (b) Image sensor size: 4.86 x3.86 square millimeter
- (c) Focal length: 6 mm
- (d) Video format: 24-bit RGB
- (e) Interface: USB
- (f) Dynamic range: < 72 dB

The webcams can be rotated freely at the horizontal and vertical axes. Therefore, calibration is necessary to ensure that the webcam lens are parallel relatively to each other and are normally located in front of the object. The webcams are tested for compatibility using MATLAB. The result shows that they are able to communicate via the USB interface connection.

4.1.2 Camera Holder Structure

Figure 2 shows the structure of the stereo vision system. A support frame is designed to hold the camera at a fix position for the purpose of reducing variability during baseline adjustments and object distance changes. The camera holder is made up of a few components: a metal bar is served as the base to clip on the webcam; a metal plate for putting an object to be measured is called the object platform; and a thread shaft for object distance adjustment. The object platform is welded onto the thread shaft.



Fig. 2 Stereo Vision Measurement System Design Layouts

4.1.3 Measurement Scale of Camera Holder

The measurement scale on the fixture is determined based on the number of rotation on the thread shaft. The relationship between the number of rotation on thread shaft and the object platform displacement is given by 20 rotation=4.7cm. Therefore, for 1cm displacement, it is 20/4.7=4.26 rotations. In our study, the scale of the fixture is marked based on every 10 cm increment. The range of the total length is from 0.5m to 2.0m. For each increment of 10 cm, approximately 42.6 rotations on the thread shaft are tuned. The webcam is clipped on the metal bar. The scale of baseline adjustment is from 0.10m to 0.18m, with an increment step of 0.04m.

4.2 Software Design

The stereo vision object distance measurement system involves image acquisition, image processing, and image analysis. The MATLAB “Image Acquisition Toolbox” is used to capture images directly into MATLAB from a PC-compatible imaging device. On the other hand, the MATLAB “Image Processing Toolbox” is used for image processing, analysis, visualization, and algorithm development. Besides that, a GUI is developed as an interactive tool between the user and the stereo vision measurement system. The GUI is shown in Figure 3.

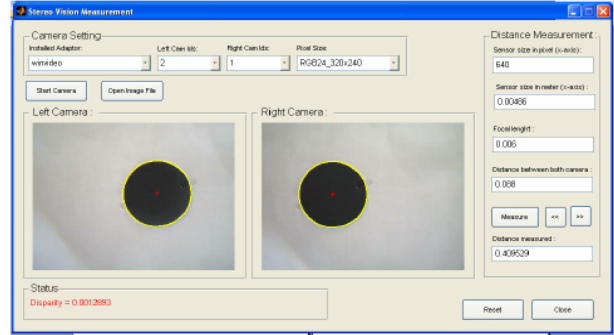


Fig. 3 GUI of the stereo vision measurement system

An object measurement system is developed based on the stereo camera geometry [10]. The measured object distance, Z , is the desired output of the system. The distance of an object from the stereo camera system is computed using equation (6):

$$Z = \frac{Bf}{(x_L - x_R)} \quad (6)$$

where $(x_L - x_R)$ is the difference in pixel location between matched features in the left and right camera images. It is also known as disparity. This disparity image together with the camera parameters can be used to calculate the stereo surface. Parameter B is the distance between center points of the left and right camera lens, is called the baseline. Parameter f is the focal length of the camera, and parameter Z is the distance of the object from the camera.

To obtain Z , parameters B (baseline value), f (focal length), and $(x_L - x_R)$ (disparity) must be provided. Baseline B is decided by the experimenter; $f=0.006$ is obtained from the camera specification data sheet. Disparity is computed from the data obtained by the MATLAB image analysis algorithm.

5. Experimental Setting

The performance of the stereo vision object distance measurement system is assessed based on 3 input factors and 1 output variable. The input factors are object distance from the webcams (A), baseline of the webcams (B), and angle of the webcams (C). The measured object distance, Z , is chosen as the output variable.

During the experiment, a total of 36 data samples were collected based on a full factorial design. Table 2 shows the input factors and their corresponding setting.

Two methods were used to estimate the regression model coefficients, i.e., OLS and FLR. Owing to the drawback in the FLR model of Peters' approach, we included GA-FLR in the current study for modeling of the stereo vision object distance measurement system.

Table 2 Input factors of the stereo vision system

Factors	Units	Levels (setting)			
Object distance (A)	M	0.5	1.0	1.5	2.0
Baseline (B)	M	0.10	0.14	0.18	
Webcam Angle (C)	degree	-15	0	15	

5.1 Factor Identification

Before fitting a regression model, it is essential to identify the important factors, to ensure that only factors which contribute significantly to the changes of the response are studied. Removing non-significant terms in a regression model could improve the prediction accuracy of the model and indirectly help save time and cost of analysis.

The analysis of variance (ANOVA) in factorial design analysis is applied to identify the significant factors involved in the stereo vision object distance measurement system. The Minitab software package is used to run ANOVA and to investigate the effect of the factors towards the response [11]. A significant level, α as 0.05, is chosen. Therefore, any factor with p -value smaller than α is significant.

Table 3 summarizes the ANOVA results. From Table 3, we can observe that factors which significantly affect the measured object distance (Z) are the main factors A, B, C, and the interaction factors between A and B (AB), A and C (AC), and B and C (BC), since their p -values are smaller than 0.05. Therefore, a total of 3 main factors and 3 interaction factors are used to construct an appropriate fitted model to the collected data.

Table 3 The ANOVA table

Source	Deg of Freedom	Sum of Squares	Mean square	F	P-value
A	3	21.27539	7.0918	7056.17	0.000
B	2	0.14548	0.07274	72.37	0.000
C	2	0.13408	0.06704	66.7	0.000
AB	6	0.0842	0.01403	13.96	0.000
AC	6	0.06534	0.01089	10.83	0.000
BC	4	0.02947	0.00737	7.33	0.000
ABC	12	0.01524	0.00127	1.26	0.281
Error	36	0.03618	0.00101		
Total	71	21.78537			

5.2 Model fitting

Two regression models, i.e. OLS and FLR, are employed to fit the collected data samples.

1) *Ordinary Least-squares model.* The OLS regression model for the experimental data collected is developed by using Minitab software package as follows:

$$\hat{y} = 0.0400 + 0.9914A + 0.2404B - 0.0010C - 0.1609AB - 0.00004AC + 0.0062BC \quad (7)$$

2) *Fuzzy Linear Regression model.*

The FLR model for the experimental data collected is as follows

$$Y = (\alpha_0, c_0) + (\alpha_1, c_1)x_1 + (\alpha_2, c_2)x_2 + (\alpha_3, c_3)x_3 + (\alpha_4, c_4)x_4 + (\alpha_5, c_5)x_5 + (\alpha_6, c_6)x_6$$

where, (α_j, c_j) , $j = 0, 1, \dots, 6$, are fuzzy parameters.

By employing Peters' approach, estimation of the FLR coefficients, (α_j, c_j) , is formulated as the following LP model:

$$\begin{aligned} \text{Max } \bar{\lambda} &= \frac{1}{36} \sum_{i=1}^{36} \lambda_i \\ \text{s.t. : } &(1 - \bar{\lambda})p_0 - \sum_{i=1}^{36} \sum_{j=0}^6 c_j |x_{ij}| \geq -d_0 \\ &(1 - \lambda_i)p_i + \sum_{j=0}^6 \alpha_j x_{ij} + \sum_{j=0}^6 c_j |x_{ij}| \geq y_i, \\ &(1 - \lambda_i)p_i - \sum_{j=0}^6 \alpha_j x_{ij} + \sum_{j=0}^6 c_j |x_{ij}| \geq -y_i, \\ &-\lambda_i \geq -1 \\ &\alpha \in \mathfrak{R}, c_j, \lambda \geq 0, x_{i0} = 1, i = 1, 2, \dots, 36, \\ &j = 0, 1, 2, \dots, 6 \end{aligned} \quad (8)$$

The MATLAB software package is used to solve the LP problem. Parameter d_0 is set to zero, ($d_0 = 0$), as the objective of the modeling process is to reduce the vagueness in the system function. Two approaches for choosing the optimum p_0 and p_i values and for maximizing $R^2_{prediction}$ are conducted, as follows.

a) The Trial and Error Approach

The selection of p_0 and p_i values are based on a trial and error basis. Sixteen trials are carried out, and the results are summarized in Table 4. Most of the $R^2_{prediction}$ value is larger than that of the OLS regression method (89.34%). Note that p_0 and p_i can assume any value from 0 to ∞ (positive numbers). This implies that the search space for suitable p_0 and p_i values is large. As such, it is a tedious task to set p_0 and p_i values by trial-and-error. From Table 4, the best $R^2_{prediction} = 91.65\%$ for FLR is obtained at $p_0 = 1500$, and $p_i = 1$. The resulting FLR fitted model is:

$$\hat{y} = (0.1245, 0.1303) + (0.9604, 0)A + (0.3128, 0)B + (-0.0024, 0)C + (-0.2695, 0)AB + (0.0018, 0)AC + (0.0111, 0)BC \quad (9)$$

b) GA Approach

As a total of 36 sample data are collected and used in constructing the estimation model, this implies that there are 36 different p_i values. Determination of p_0 and p_i values is accomplished by using MATLAB GA Toolbox. The GA parameters are set according to Table

1. Detail definition of each GA parameter can be obtained in [12].

In this case study, the fitness function for the GA is the $R^2_{prediction}$ value. Figure 4 shows the steps involved in the fitness function computation. The target of the GA optimization process is to maximize the $R^2_{prediction}$ value. The results from GA iterations are the optimum fitness value ($R^2_{prediction}$), and the optimal p_0 and p_i values that contribute to the respective $R^2_{prediction}$ value.

By applying GA-FLR, the best $R^2_{prediction}$ is 94.09%. The GA-FLR estimation model is as follows:

$$\hat{y} = (-0.3276,0) + (1.0206,0)A + (3.8245,0.0487)B + (0.0063,0.0003)C + (-0.993,0)AB + (0.0082,0.0001)AC + (-0.1084,0)BC \quad (10)$$

The corresponding p_0 and p_i values are shown in Table 5.

Table 4 Trial and error approach

p_0	p_i	R^2_{fit}	$R^2_{prediction}$
100	10	96.68%	85.98%
20	50	96.20%	83.51%
130	13	96.63%	85.70%
500	27	96.78%	87.00%
82	5	96.79%	87.61%
1500	1	96.27%	91.65%
1500	2	96.54%	89.96%
1400	1	96.30%	91.51%
1450	1	96.25%	91.63%
1480	1	96.26%	91.64%
5000	1	95.91%	89.87%
4800	1	95.92%	89.85%
4800	2	96.21%	90.14%
4800	3	96.36%	90.10%
3	345	96.18%	83.44%
578	1	96.64%	89.79%

Table 5 p_0 and p_i values corresponding to $R^2_{prediction} = 94.09\%$

p0	703.9685	p13	389.822	p26	658.6989
p1	605.3106	p14	521.527	p27	660.0588
p2	929.2059	p15	300.9104	p28	303.2557
p3	565.0434	p16	18.7383	p29	63.5294
p4	777.8088	p17	143.5406	p30	28.9656
p5	591.9815	p18	48.7708	p31	735.9622
p6	541.0455	p19	861.8783	p32	401.4261
p7	842.97	p20	903.5321	p33	482.7803
p8	87.2393	p21	163.3787	p34	977.41
p9	629.9248	p22	121.2331	p35	314.0178
p10	567.6793	p23	426.3456	p36	134.1613
p11	273.1338	p24	757.387	p37	303.2557
p12	778.2485	p25	590.5389		

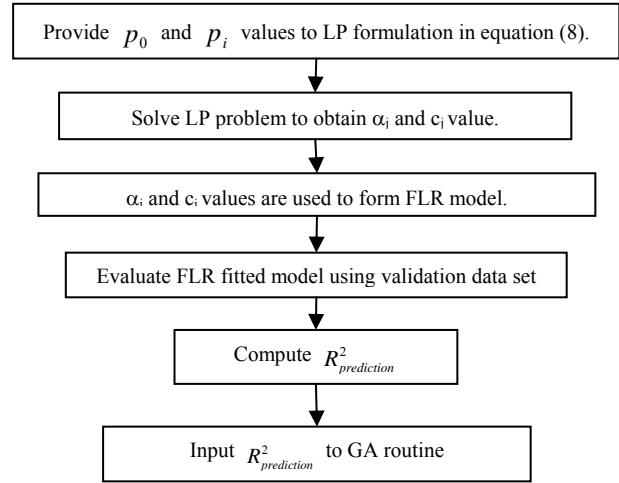


Fig.4 Flowchart of fitness function computation

5.3 Model validation

One of the methods to validate a regression model prediction performance is to compare the predicted values with a set of newly collected data. If the model can achieve higher prediction accuracy on the new data, the user would have more confidence on the model building process and using the model for prediction. When we have two or more alternative regression models, comparing the prediction performance of these models on new data can serve as a guideline for final model selection [13]. In the current study, we compare the OLS, FLR, and GA-FLR models by predicting the measured object distance, Z for new data. Thus, another set of 14 new data samples with different combinations of parameters is used for validation of the prediction model.

5.4 Discussion

Table 4 summarizes the results based on trial and error approach. It takes approximately 20 minutes to obtain the results of 16 trials shown in Table 4. The method of trial and error is done manually. More trials can be carried out if the experimenter do not satisfied with the prediction capability achievement. However, more time will be consumed and it is a tedious task. As a result, we propose application of GA to determine the p_0 and p_i values for achieving better prediction capability in FLR. The proposed GA-FLR method utilize simple GA algorithm to determine p_0 and p_i values automatically. The computation time to obtain the results shown in Table 5 is less than 5 seconds. One advantage of the GA method is there is various value of p_i as compared with the existing trial and error method. It has obviously shown that GA approach has significantly reduced the processing time in p_0 and p_i values selection process.

The OLS model (equation (7)), FLR model (equation (9)) and GA-FLR model (equation (10)) are used to give prediction for the validation data. The performances of

the respective regression models are measured by the R^2 value. Table 6 summarizes the R_{fit}^2 and $R_{prediction}^2$ values for the three regression models.

Table 6 Comparison of the regression models

	R_{fit}^2	$R_{prediction}^2$
OLS	97.12%	89.34%
FLR	96.26%	91.65%
GA-FLR	91.91%	94.09%

The results in Table 6 demonstrate that the GA-FLR model gives the best $R_{prediction}^2$ value (94.09%), although the R_{fit}^2 (91.91%) of GA-FLR is lower than those of OLS and FLR. Among the three regression methods, OLS produces the best estimated model with $R_{fit}^2 = 97.12\%$. However, this does not guarantee the usefulness of the model as a prediction model. This is proven by the result of $R_{prediction}^2 = 89.34\%$, which is the lowest among the three models.

6. Summary

In this paper, we have developed a system modeling method based on FLR and GA. The proposed approach has been evaluated using an object distance measurement system based on the stereo vision concept. Owing to the uncertainty in the system parameters setting, fuzzy regression has been employed to model the system. A study has been conducted whereby factorial design is used to collect the experimental data. The results show that FLR models provide a better predictive capability with a higher $R_{prediction}^2$ value than that of the OLS method. In addition, we have proposed a GA-FLR model which can achieve better predictive capability by stochastically choosing p_0 and p_i values.

The GA approach has been shown to be effective in determining p_0 and p_i values for FLR. However, more investigations are necessary in order to fully ascertain the efficacy of the proposed GA-FLR model. As such, further work will focus on how to further extend the GA-FLR model to other applications, as well as to use alternative objective functions, e.g. Mean Square Error (MSE) or error percentage, for the GA-FLR model.

References

[1] Montgomery, D.C., Peck, E.A., and Vining, G.G., Introduction to linear regression analysis, 3rd ed., New York, John Wiley & Sons, 2001, Chap.4, pp.131-151

[2] Tanaka, H., Uejima, S., and Asai, K., "Linear Regression Analysis with Fuzzy Model", IEEE Trans. Systems, Man, and Cybernetics SMC, vol. 12, (1982), pp. 903-907.

[3] Jorge, A.S., "Claim reserving with fuzzy regression and Taylor's geometric separation method", Insurance Mathematics and Economics, vol. 40, (2007), pp. 145-163.

[4] Chen, K.T., Rys, M.J., Lee, E.S., "Modeling of thermal comfort in air conditioned rooms by fuzzy regression analysis", Mathematical and computer Modeling, vol. 43, (2006), pp. 809-819.

[5] Xue Y., et al., "Fuzzy regression method for prediction and control the bead width in the robotic arc-welding process", Journal of Materials Processing Technology., vol. 164-165, (2005), pp. 1134-1139.

[6] Terano, T., Asai K., and Sugeno, M., Fuzzy systems theory & its applications, San Diego, CA, Academic Press, 1991, Chap. 4, pp.69-77.

[7] Peters G., "Fuzzy Linear Regression with Fuzzy Intervals", Fuzzy Sets. Syst., vol. 63, (1994), pp. 45-55.

[8] Ross, T.J., Fuzzy logic with engineering applications, 2nd ed., West Sussex, England, John Wiley & Sons, 2004, Chap. 14, pp.555-566.

[9] Haupt, R.L., and Haupt, S.E., Pratical Genetic Algorithms, 2nd ed., New Jersey, John Wiley & Sons, 2004, Chap.3 pp.51-65.

[10] Goshtasby, A.A., 2-D and 3-D Image Registration for medical, remote sensing, and industrial applications, New Jersey, John Wiley & Sons, 2004, Chap.10, pp.197-199.

[11] Montgomery, D.C., Design and analysis of experiments, 5th ed., New York, John Wiley & Sons, 2001, Chap. 6, pp.218-240.

[12] Genetic Algorithm and Direct Search Toolbox User's Guide, Version 1, The MathWorks.

[13] Montgomery, D.C., Peck, E.A., and Vining, G.G., Introduction to linear regression analysis, 3rd ed., New York, John Wiley & Sons, 2001, Chap.15, pp.529-545.