International Journal of Physical Sciences Vol. 5(4), pp. 313-320, April 2010
Available online at http://www.academicjournals.org/IJPS
ISSN 1992 - 1950 © 2010 Academic Journals

Full Length Research Paper

A novel 3D spatial neighbor points coupling surface modeling method for scattered points

Huipu Xu*, Ying Hu and Yuqing Chen

Automation Research Center, Dalian Maritime University, Dalian, CO 116026 P. R. C., China.

Accepted 11 January, 2010

This paper reports a novel 3D surface modeling method based on nonlinear auto-regressive moving average with exogenous input (NARMAX) in time domain - 3D surface modeling based on spatial neighbor points coupling (SMSNPC) for scattered data points. The 3D model of complex surface can be reconstructed through the mutual coupling of neighbor points. The algorithm of adjustment for model parameters is introduced in this paper based on ANN. To demonstrate the efficient applicability of the developed prototype, two groups of point cloud data are presented. One collected by the robot measure system is used to validate the reconstruction result and the other is used to compare with conditional non-coupling ANN modeling in reconstruction precision. From the results, the new approach presented in this paper validates the expectation that the surface modeling based on spatial neighbor points can match the real surface satisfactorily and it is particularly suitable for modeling of complex surface which cannot be finished by non-coupling ANN modeling. And from the comparative studies it is shown the reconstruction precision of SMSNPC is much higher.

Key words: Surface modeling, scattered points, coupling, ANN, SMSNPC.

INTRODUCTION

It has been witnessed that the application of computer aided design (CAD) technology has been expanding rapidly in the field of industry with the development of technology of computer. No doubt, engineering drawings can be created and modified easily by using CAD packages. Accordingly, the CAD databases have been linked to provide direct inputs to the computer aided process planning (CAPP) systems and numerically control the tool path to generate software codes. In some cases where CAD models are not existed, extensive effort is usually required to identify a proper model for describing a specified 3D surface through reverse engineering, such as the reshaping and restoring of relics and so on.

Over the last 20 years, the development in machine vision technology has triggered some new applications, such as inspection and measurement, location analysis, part recognition and robot guidance, which establishes firm base for reverse engineering. However, the CAD model reconstruction for existing complex freeform part still requires more efforts in which model reconstruction is key.

The obtainment of point cloud from the object is the precondition and most important task of model reconstruction. The robotics 3D laser measurement system (Ma et al., 2006) is preferred for collecting the point cloud of the 3D complex surface and the presented robot vision system is obviously different from the traditional approach. In the new measuring approach, a measured object is placed in an experimental area in stand still or on positioner; all the surface information of the object is obtained from a once off scanning due to the agility of the robot with six - freedom degree and the path planning can be done to solve the laser measurement blind points problem so that the whole of point cloud signals of the measured part without any blank space can be obtained directly and automatically.

The good measurement system is the basis of obtaining the precise point cloud and also provides the basic conditions for model reconstruction. In recent years, a lot

*Corresponding author. E-mail: hpx1212@newmail.dlmu.edu.cn, hpx1212@163.com.

of works have been performed to develop the methods for getting mathematical model of optical 3D points, in order to facilitate handling of them on existing CAD/CAM system, for example, model reconstruction using ANN for the capability of its approximation to nonlinear function. However, these mathematical tools are not suit to reconstruct model for the objects with complex surface.

Some research topics about modeling in control field arouse our new ideas for model reconstruction. For instance, nonlinear auto-regressive moving average with exogenous inputs (NARMAX) model is put forward by Billings etc. in 1982 (Leontaritis and Billings, 1985) and the proof for describing general nonlinear systems is given in 1989(Chen and Billings, 1989).This model almost includes all nonlinear model such as bilinear model, Hammerstain model, Wiener model, nonlinear time sequence model, ARMAX model, Output Affine model and so on. It is used successfully in actual engineering, for example, automation control (Val and Hiri, 2008), iatrology (Abdul et al., 2007), environment (Harish et al., 2007; Hagan et al., 1995) and so on.

In this paper, a novel 3D Surface Modeling based on Spatial Neighbor Points Coupling (SMSNPC) is presented which is based on NARMAX model in time domain. The model of surface for complex object can be built in this way. The algorithm of parameters adjustment based on gradient descent is introduced in detail. To prove the effectiveness of the proposed scheme, one group of data points is presented to validate the reconstruction result. And the other group of data points is used to be compared with the traditional ANN model reconstruction of non-coupling for the precision validation and the simulation results shows the former has a prominent advantage than the latter in modeling and has a higher modeling accuracy.

Algorithm theory

NARMAX model theory

NARMAX model is introduced by Billings and his workmate in 1981. It is a discrete system expression and is used to describe the discrete-time nonlinear stochastic control system (leontaritis and Billings, 1985). The model is as follows:

$$y(t) = f(y(t-1),..., y(t-n_y), u(t-1), ..., u(t-n_u), e(t-1),..., e(t-n_e)) + e(t) \quad (1)$$

Where $y(t), u(t), e(t)$ are the output, input and noise of system respectively, and n_y, n_u, n_e is the maximum lag of output, input and noise which is also called as orders of output, control input and noise. $e(t)$ is assumed to be gauss white noise. $f(.)$ is a nonlinear function. (.) could be any combination of input, output and noise. The formula (1) is expressed as the polynomial form, gives

$$y(t) = \sum_{0}^{m} \theta_m p_m(t) + e(t) \quad (2)$$

Where m is the number of unknown parameters θ_m is the parameter which will be recognized and these parameters are fixed for specific system. $p_m(t)$ are the terms in formula (1).

Formula (2) can be divided into two parts where the combination of terms without noise is called the term of process model and the combination with noise is called the term of noise model. In practical applications, only the term of process model is considered because of the unknown to the term of noise model. Then, formula (1) is simplified as following.

$$y(t) = f(y(t-1),..., y(t-n_y), u(t-1),..., u(t-n_u)) \quad (3)$$

SMS*NPC theory*

Formula (3) can be regarded as a stochastic MISO nonlinear system, where $y \in Y \in R^1$. The modeling idea using NARMAX in control system can be extended the 3D surface model reconstruction. So the process of 3D surface model reconstruction can be considered a process of modeling of MISO system and the scattered point cloud from the surface can be seen as a discrete system coming from a 3D continuous surface system after discretization. So let

$$z_k(x_i, y_j) = f(x_i, x_{i\pm1}, \cdots, x_{i\pm n_x}, y_j, y_{j\pm1}, \cdots, y_{j\pm n_y}, z_{k\pm1}, \cdots, z_{k\pm n_z}) \quad (4)$$

Where $z_k(x_i, y_j)$ is the coordinate value at i point of X-Y plane in x direction y_i is the coordinate value at j point of X-Y plane in y direction. $i = 1,..., l_x, j = 1,...l_y$. n_x , n_y , n_z are orders of the neighbor of 3D point according to x direction, y direction and z direction respectively. Formula (4) is the expression of 3D surface in space and the z-value of the point (x_i, y_j) in X-Y plane can be considered as the mutual coupling of the point (x_i, y_j) and its neighbors.

The next work is how to recognize the model parameters in formula (4). As we all know, the artificial neural network can simulate the function of human brain through learning mechanism (Hagan et al., 1995; Haykin, 1994). It is well known that ANN is a very effective modeling tool for representing nonlinearities. According to some sample data, a model can be built through learning and can also match these sample data well. So ANN can be used in the case that the system model can not be expressed by means of mathematical methods. ANN is very flexible and strong in some system modeling, especially in the approximation of nonlinear function. The 3D surface model described by formula (4) has strong non-linear and it is very difficult to obtain the model through general methods. Accordingly, ANN can be trained into a network model to finish this task due to its good approximation to nonlinear function. Of course, the point information has to be sufficient to include the whole feature of a 3D surface.

The multilayer feed forward network is general of all ANN and it is structured with 3 layers, input layer, hidden layer and output layer.

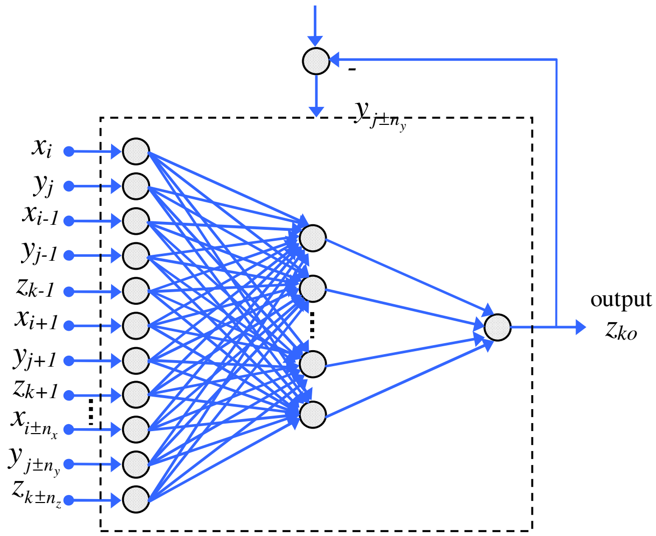

Figure 1. The structure of parameters recognition.

Every node in same layer is connected with the node in previous layer through weighs and these weighs are initialized at the beginning of the training. A BP algorithm for parameter estimation is given in (Rumelhart and Mcclelland, 1986).

To take characteristics of 3D surface model and network convergence rate into account, the ANN for parameters recognition of the 3D surface model based on spatial neighbor points coupling can be designed with 3 layers, input layer, hidden layer with l nodes and output layer with only one node. It is shown in Figure 1.

In Figure 1, z_{ko} is the coupling output of the inputs of network and z_k is the ideal value. The inputs of the network are the values of 3D point x_i, y_j, z_k in x-y plane and its neighbor in space.

The input of *mth* node in hide layer is

$$r_m = \sum_{a=i-n_x}^{i+n_x} {}^1w_{ma} x_a + \sum_{b=j-n_y}^{j+n_y} {}^1w_{mb} y_b + \sum_{c=k-n_z}^{k-1} {}^1w_{mc} z_c + \sum_{c=k+1}^{k+n_z} {}^1w_{mc} z_c \quad (5)$$

Where $m = 1,2,\cdots,l$ and l is the number of node in the hide layer. ${}^1w_{ma}, {}^1w_{mb}, {}^1w_{mc}$ are the weights from a-type, b-type and c-type node in input layer to *mth* node in hide layer respectively.
The output of *mth* node in hide layer is

$$s_m = f_1(r_m - \theta_m) \quad (6)$$

Where f_1 is activation functions and θ_m is the threshold of *mth* node. Then the input of node in output layer is

$$r_o = \sum_{m=1}^{l} {}^2w_m s_m \quad (7)$$

Where ${}^2w_m$ is the weight from the *mth* node in hider layer to the node in output layer. Then the output of network which the value is

$$z_{ko}(x_i, y_j) = f_2(r_o - \theta_o) \quad (8)$$

Where f_2 is activation function of node in output layer and θ_o is the threshold of node in output layer.

When the *pth* sample data is as the input of network and z_{ko} is the output of network after the weights have been adjusted, so the object function of network (L_2 norm is taken) is

$$E_P = \frac{1}{2}\left\| z_k - z_{ko} \right\|_2^2 = \frac{1}{2}(z_k - z_{ko})^2 = \frac{1}{2} e_p{}^2 \quad (9)$$

The whole object function of the network is

$$J = \sum_{p=1}^{L} E_p \quad (10)$$

Where L is the number of sample data.

The weight adjustment algorithm from hide layer to output layer after the training of t epochs is

$$\begin{aligned} {}^2w_m(t+1) &= {}^2w_m(t) - \eta_2 \cdot \frac{\partial J}{\partial\, {}^2w_m} \\ &= {}^2w_m(t) - \eta_2 \cdot \sum_{p=1}^{L} \frac{\partial E_p}{\partial\, {}^2w_m} \end{aligned} \quad (11)$$

Where η_2 is the learning step of ${}^2w$, also known as learning operator.

$$\begin{aligned} \frac{\partial E_p}{\partial\, {}^2w_m} &= \frac{\partial E_p}{\partial r_o} \cdot \frac{\partial r_o}{\partial\, {}^2w_m} \\ &= \frac{\partial E_p}{\partial z_{ko}} \cdot \frac{\partial z_{ko}}{\partial r_o} \cdot \frac{\partial r_o}{\partial\, {}^2w_m} = -e_p \cdot f_2' \cdot s_m \end{aligned} \quad (12)$$

Namely,

$$\begin{aligned} {}^2w_m(t+1) &= {}^2w_m(t) - \eta_2 \cdot \frac{\partial J}{\partial\, {}^2w_m} \\ &= {}^2w_m(t) - \eta_2 \cdot \sum_{p=1}^{L} (-e_p \cdot f_2' \cdot s_m) \end{aligned} \quad (13)$$

The weight adjustment algorithm from input layer to hide layer after the training of t epochs is (a-type node is taken as a example for explaining)

$$\begin{aligned} {}^1w_{ma}(t+1) &= {}^1w_{ma}(t) - \eta_1 \cdot \frac{\partial J}{\partial\, {}^1w_{ma}} \\ &= {}^1w_{ma}(t) - \eta_1 \cdot \sum_{p=1}^{L} \frac{\partial E_p}{\partial\, {}^1w_{ma}} \end{aligned} \quad (14)$$

Where η_1 is the learning operator of ${}^1w$.

$$\begin{aligned}\frac{\partial E_p}{\partial^1 w_{ma}} &= \frac{\partial E_p}{\partial r_m}\cdot\frac{\partial r_m}{\partial^1 w_{ma}} \\ &= \frac{\partial E_p}{\partial s_m}\cdot\frac{\partial s_m}{\partial r_m}\cdot\frac{\partial r_m}{\partial^1 w_{ma}} \\ &= \frac{\partial E_p}{\partial s_m}\cdot\frac{\partial s_m}{\partial r_m}\cdot x_a\end{aligned} \tag{15}$$

$$\begin{aligned}\frac{\partial E_p}{\partial s_m} &= \frac{\partial E_p}{\partial r_o}\cdot\frac{\partial r_o}{\partial s_m} = \frac{\partial E_p}{\partial r_o}\cdot\frac{\partial}{\partial s_m}\sum_{m=1}^{l}{}^2 w_m s_m \\ &= \frac{\partial E_p}{\partial r_o}\cdot({}^2 w_m) = \frac{\partial E_p}{\partial z_{ko}}\cdot\frac{\partial z_{ko}}{\partial r_o}\cdot({}^2 w_m) \\ &= -e_p\cdot f_2'\cdot({}^2 w_m)\end{aligned} \tag{16}$$

$$\frac{\partial s_m}{\partial r_m} = f_1' \tag{17}$$

Namely,

$$\begin{aligned}{}^1w_{ma}(t+1) &= {}^1w_{ma}(t) - \eta_1\cdot\frac{\partial J}{\partial^1 w_{ma}} \\ &= {}^1w_{ma}(t) - \eta_1\cdot f_1'\cdot x_a\cdot\sum_{p=1}^{L}(-e_p\cdot f_2'\cdot{}^2 w_m)\end{aligned} \tag{18}$$

The weight adjustment algorithm from b-type node, c-type node to hide layer by analogy is

$$\begin{aligned}{}^1w_{mb}(t+1) &= {}^1w_{mb}(t) - \eta_1\cdot\frac{\partial J}{\partial^1 w_{mb}} \\ &= {}^1w_{mb}(t) - \eta_1\cdot f_1'\cdot y_b\cdot\sum_{p=1}^{L}(-e_p\cdot f_2'\cdot{}^2 w_m)\end{aligned} \tag{19}$$

$$\begin{aligned}{}^1w_{mc}(t+1) &= {}^1w_{mc}(t) - \eta_1\cdot\frac{\partial J}{\partial^1 w_{mc}} \\ &= {}^1w_{mc}(t) - \eta_1\cdot f_1'\cdot z_c\cdot\sum_{p=1}^{L}(-e_p\cdot f_2'\cdot{}^2 w_m)\end{aligned} \tag{20}$$

The search algorithm for neighbor point in 3D space

The search algorithm for neighbor point in 3D space is an optimization algorithm based on K-D tree. K-D tree is abbreviation of multi-dimensional binary search tree and it is extending of one-dimensional binary search tree in multi-dimensional space (Gaede and Gunther, 1998). 1D or 2D data is no longer the key of node in k-d tree, but multi-dimension array (when the k-d tree is used in 3D space, k=3 and the 3D coordinates of point (x, y, z) is the key of

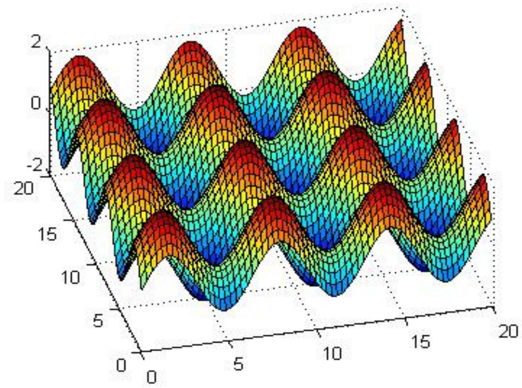

Figure 2. Wave surface.

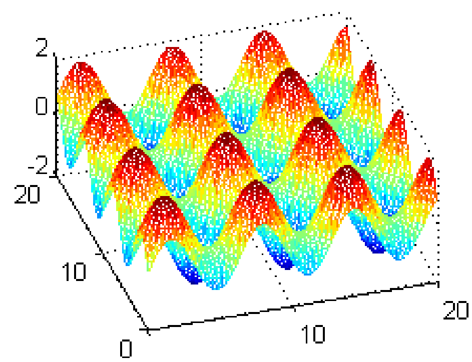

Figure 3. Reconstruction of 3D wave surface.

node in k-d tree to split the search space.). First, the set of elements in three-dimensional search space must be described as the structure of k-d tree when k-d tree search algorithm is used, namely, the optimal k-d tree must be built in search space. Then, the k-nearest neighbor is built in 3D space based on k-d tree.

EXPERIMENTAL RESULTS

Simulation results

In this section, we compare the performance of traditional ANN and SMSNPC by simulation. The point data group 1 is sampled from the surface of 3D wave (Figure 2), this group data includes 4489 sample points. The model reconstruction of SMSNPC is shown in Figure 3.

The conventional Artificial Neural Network (ANN) scheme, which has been applied for 3D surface reconstruction, was used to implement model reconstruction again for data group 1. The comparison is finished when the two approaches have the same network structure. The model reconstruction accuracy of two schemes of the case are shown in Figure 4 respectively with different number of neurons in hide layer, where

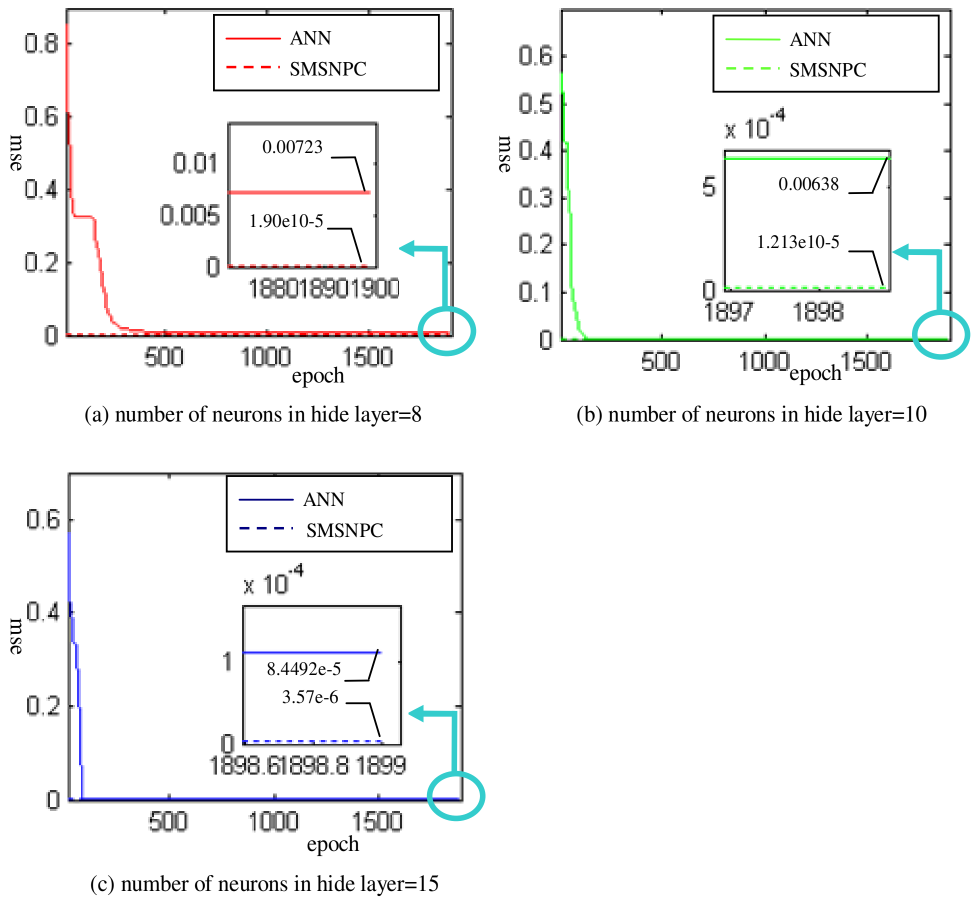

Figure 4. Comparison of model reconstruction of SMSNPC and ANN of data group 1.

x-axis denotes the running epoch and the y-axis denotes the Mean Square Error (MSE) between the output of network model (ANN or SMSNPC) and the real sample points.

In Figure 4, $n_x = n_y = n_z = 3$ in the approach of SMSNPC. First, data group 1 was used to train ANN. When the number of neurons in hide layer is taken as 8 and the training epoch is arriving at 1900, it is shown that the convergent rate will not change in Figure 4(a), now, the model reconstruction accuracy of ANN (mse) is only 0.00723, but the model reconstruction accuracy of SMSNPC (mse) is up to 1.90e-5 and the reconstruction accuracy is improved to two orders of magnitude. From Figure 4(b), when the number of neurons is chosen as 10 with the same training epoch (1900) and the reconstruction accuracy of ANN is improved to 0.00638 at convergence, but the reconstruction accuracy of SMSNPC is up to 1.213e-5. In Figure 4(c), the reconstruction accuracy of SMSNPC is up to 3.57e-6, but the accuracy of ANN is only 8.4492e-5. Based on the data group 1, it is shown that the reconstruction accuracy of SMSNPC can be higher than the accuracy of ANN by two orders of magnitude. From the simulation results, it is known that the approach of SMSNPC is a better tool for model reconstruction.

Real object results

In this section, two real objects are taken as examples for demonstration of the effectiveness of the presented scheme. The point data group 2 comes from the upper surface of a real object (Figure 5). In Figure 6, it is shown the object in the process of scanning by the robotics 3D laser measurement system (Ma et al., 2006) and the group data collected from the system includes 3429 sample points; The point data group 3 come from the face surface of plaster head portrait model of Beckham (Figure 10) and the data set includes 10827 sample points (Figure 11).

The model reconstruction of data group 2 with SMSNPC is shown in Figure 7. where $n_x = n_y = n_z = 5$. The same data are trained with traditional ANN to reconstruct the

Figure 5. Real object.

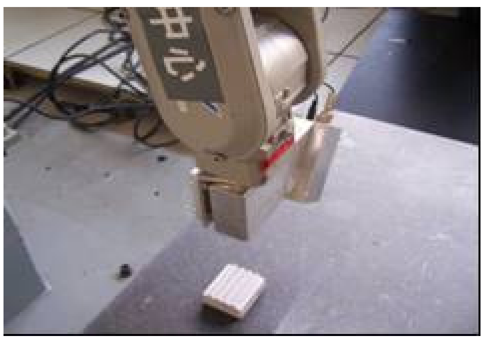

Figure 6. Object in scanning.

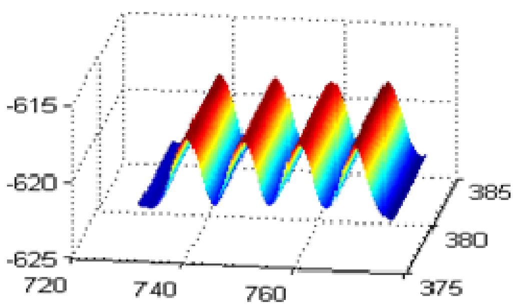

Figure 7. Model reconstruction of SMSNPC.

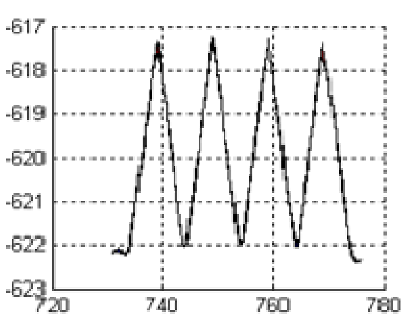

Figure 8. Cross section of the reconstruction with SMSNPC.

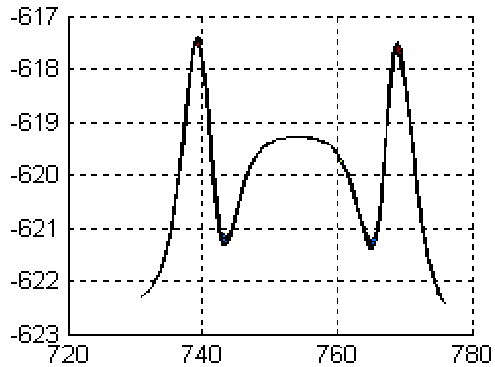

Figure 9. Cross section of the reconstruction with ANN.

Figure 10. Plaster head portrait of Beckham.

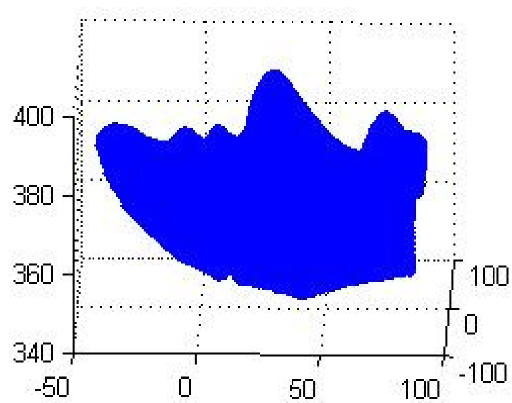

Figure 11. Point data of plaster head portrait.

model. The model reconstruction results of SMSNPC and ANN in cross section are shown in Figures 8 and 9 respectively. From the results, it can be shown that the approach with ANN failed obviously because the curvature of this type of surface varied sharply and it is very difficult for ANN to model for this type of surface. When the number of neurons in hide layer of ANN for parameters

Table 1. The reconstruction results of ANN.

Number of neurons in hider layer	epochs	Running time(s)	Reconstruction accuracy
5	2500	460	11.7
6	2500	473	14.2
7	2500	483	17.7
8	1720	341	9.95
9	2500	502	7.16
10	2500	516	5.97
11	2574	546	6.08
12	2500	555	4.78
13	2500	581	5.42
15	2500	609	3.7

Table 2. The reconstruction results of SMSNPC

Number of neurons in hider layer	epochs	Running time	Reconstruction accuracy
5	2289	522	0.0485
6	2505	628	0.0476
7	2502	625	0.0471
8	2500	822	0.0480
9	2500	822	0.0434
10	1949	567	0.0463
11	2500	792	0.0462
12	1271	441	0.0396
13	2500	930	0.0413
15	2500	986	0.0406

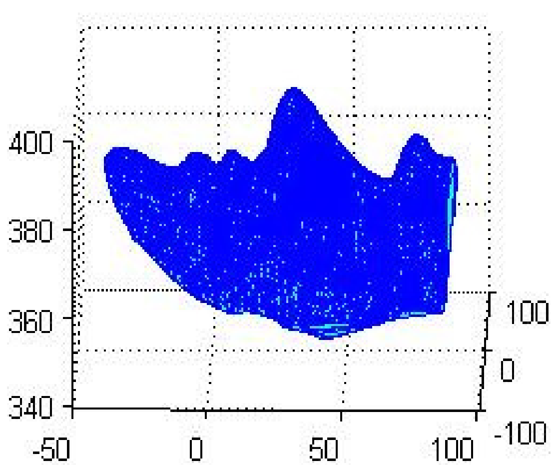

Figure 12. Model reconstruction of plaster head portrait with SMSNPC.

recognition is taken as 15, the accuracy of SMSNPC is up to 0.00128 (MSE: Mean Square Error). So we can see that SMSNPC is an effective 3D surface reconstruction approach.

The model reconstruction of face surface of Beckham is shown in Figure 12 with SMSNPC. From the result, it is shown that the face profile is restored well with the reconstructed 3D digital model. Many products can be manufactured based on the digital model, such as mask, waxwork of famous person, plaster model and so on.

For showing the principal advantage of the presented algorithm on real surface, the data group 3 is taken as sample points using ANN and SMSNPC respectively. Here, the 3-layer structure is taken for ANN and the same structure is taken for recognition of SMSNPC parameters ($n_x = n_y = n_z = 3$). Tables 1 and 2 are reconstruction accuracy of these two approaches. From the tables, we can see that the accuracy of each network structure for SMSNPC is higher than ANN and the precision is improved more than an order of magnitude.

Above all, we can say that the presented algorithm in this paper – SMSNPC is more suitable for the model reconstruction of real surface than traditional ANN.

Conclusion

This paper puts forward a novel 3D Surface Modeling based on Spatial Neighbor Points Coupling for scattered points. This is the first time SMSNPC is applied for model reconstruction for scattered data. It can be flexibly quickly used to reconstruct digital models with any point density. Of course, the denser the point cloud is, the higher the precision of the reconstruction model is. But too much dense point, namely, the data file is too big, can reflect on rapidity of reconstruction. In addition, the choice for the orders of the neighbors of 3D point according to x direction, y direction and z direction respectively, namely, n_x , n_y , n_z should be appropriate and the values can't be chosen too large which can impact the speed of training and too small which can influence the accuracy of model reconstruction. As a matter of experience, these values are taken from 3 to 8. The application example shows that the precision of the SMSNPC is satisfactory. Moreover, the convergence rate of the SMSNPC is faster than non-coupling ANN. The most important point is that SMSNPC is suitable for the real surface with complex. By comparing the data predicted from SMSNPC with the real samples at the same position points, the reconstructed model can match the real object surface very well.

REFERENCES

Abdul H, Rahim FI, Taib MN (2007). "A Novel Preciction System in Dengue Fever Using NARMAX Model", International Conference on Control, Automation and Systems 2007 , Oct. 17-20, 2007 in COEX, Seoul, Korea 305-309.

Chen S, Billings SA (1989). "Representation of Non-linear Systems: the NARMAX Model" [J]. Int. J. Control (S0020-7179) 49(3): 1013-1032.

Gaede V, Gunther O (1998). "Multidimensional access methods" [J]. ACM Computing Surveys 30(2): 170-231.

Hagan MT, Demuth HB, Beale M (1995). Neural Network Design [M]. Boston, MA: PWS.

Harish J, Palanthandalam M, Biju E, Dennis S, Bernstein WM, Aaron JR (2007). "NARMAX Identification for Space Weather Prediction Using Polynomial Radial Basis Functions", Proceedings of the 46th IEEE Conference on Decision and Control, New Orleans, LA, USA, Dec. 12-14.

Haykin S (1994). Neural Networks: A Comprehensive Foundation [M]. New York: Mc Millan.

Leontaritis IJ, Billings SA (1985). "Input-output Parametric Models for Non-linear Systems" [J]. Int. J. of Control (S0020-7179) 41(2): 311-344.

Ma Z, Xu HP, Hu Y, Huang J, Zhang X (2006). "Robot vision system and artificial neural network for model reconstruction in reverse engineering". Proceeding of the 6th World Congress on Control and Automation 2: 9073-9078.

Val OMM, Hiri RM (2008). "An Approach to Polynomial NARX/NARMAX Systems Identification in a Closed-loop with variable structure control", Int. J. Automation Comp. 05(3): 313-318.

Rumelhart DE, Mcclelland JL (1986). Parallel Distributed Processing, Explorations in the Microstructure of Cognition [M], Cambridge, MIT Press Vol. 1.