International Journal of Multimedia and Ubiquitous Engineering
Vol.11, No.1 (2016), pp.115-124
http://dx.doi.org/10.14257/ijmue.2016.11.1.12

Optimal Viewpoint Extraction of 3D Model Based on AdaBoost Iterative Algorithm

Dong Tao

Eastern Liaoning University
Keaton2011@163.com

Abstract

According to the limitations of a single measurement algorithm in the current 3D models' viewpoint extraction, this essay puts forward a viewpoint extraction algorithm based on AdaBoost iterative algorithm, which can make the features adaptive automatically. It, firstly, extracts 3D models' feature descriptor and feature vector in the model library and adopts AdaBoost iterative algorithm to establish rules about classification and matching from geometric features and various viewpoint extraction algorithm; then, it constructs decision classifier in order to extract optimal viewpoint. In query process, the model obtains viewpoint extraction algorithm which can suit its geometric feature through decision classifier and then gets its best view by calculation. The experimental result shows this algorithm extraction effect is superior to the one by a single measurement algorithm.

Keywords: *Optimal viewpoint, Geometric feature, 3D model, AdaBoost iterative algorithm*

1. Introduction

Cognitive Recognition Theory put forward by Liu Group [1] points out 3D model is shown by choosing a two-dimensional image in human's brain. 3D model's optimal viewpoint extraction indicates that 3D model's perspective chosen by the computer automatically will make this model, which contains maximum information and suits human being's visual perception, reflect on this two-dimensional image correctly. The classical Canonical Viewpoint Theory [2] refers that most people will choose almost the same viewpoint to observe the same object, that is the optimal viewpoint in theory. This technique is applied widely in the fields of computer vision's shape recognition and classification [3], the production of models' thumbnails in 3D model library, the editing of 3D model, *etc*.

Recent years, many investigators put forward various algorithms to extract optimal viewpoint. Vazquez Group [4] make good use of surface area entropy to measure 3D model's geometric features, and then to select the perspective containing maximum area entropy as optimal viewpoint; Polonsky Group [5] think visible area ratio can be used for measuring 3D model's geometric features and viewpoint's quality; Lee Group [6] propose that mesh saliency algorithm should be used with gauss weighted average based on curvature, in order to improve key point's capturing capabilities; Yang Group [7] come up with the idea that geometric features based on visual plane's curvature can be used for calculating the entropy related with perspective to judge the optimal viewpoint; Cao Group [8] presents algorithm can be chosen by calculating distance histogram's entropy to standardize viewpoint's quality, based on 3D model's optimal viewpoint on distance histogram; Bonaventur Group [9] focus the viewpoint on 3D model's outermost spherical surface, they think the algorithm should be chosen based on the viewpoint of polygon information projection; Mortara Group [10] use semantic-oriented segmentation method to segment 3D model, the parts segmented in different methods will show different

ISSN: 1975-0080 IJMUE

geometric features' weight, thus, candidate perspective set can be evaluated through weighting function to get the optimal viewpoint; Considering perspectives should show 3D model's geometric features as many as possible, Zhang Group [11] put forward a different method, in which particle swarm optimization is applied, to choose model's optimal viewpoint.

According to 3D model's geometric features, the researches above suggest various effective algorithms for extracting viewpoint. But because of 3D models' different shapes, a single measurement extraction algorithm cannot be suitable for all of them. Models' multiple geometric features are also considered in the method of references [10-11], while owing to the complicated extraction process, more manual interventions need carrying out.

2. Research Method Summary

The method here is mainly studying on how to choose viewpoint extraction algorithm adaptively, according to 3D model's geometric features, in order to realize the extraction of optimal viewpoints. The algorithm process, shown in Figure 1, consists of training process and viewpoint extraction process. In the training process, firstly, the model's SDF feature descriptor and eigenvector are extracted from 3D model library; meanwhile, the optimal viewpoint extraction algorithm can be designated from different models; then, the matching rule, which suits geometric features and optimal viewpoint extraction algorithm, can be naturally established. And the optimal viewpoint classifier [12] can be obtained by using AdaBoost iterative algorithm for training. In the process of viewpoint extraction, well-constructed decision classifiers should first be applied to classify input models. Thus, the algorithm can be got to suit model's geometric features, and then model's best view can also be shown by this algorithm.

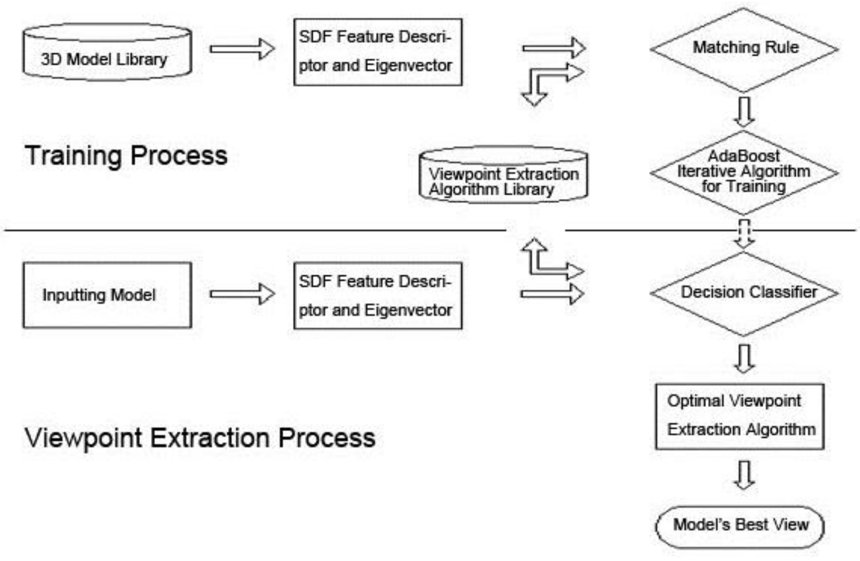

Figure 1. Algorithm Process

3. Geometric Features and Viewpoint Extraction

The essay aims to extract optimal viewpoints adaptively, and the optimal algorithm choosing needs to be realized by decision classifier, whose construction is based on model's geometric features in AdaBoost iterative and viewpoint extraction algorithm in algorithm library, thus, matching and training study can be well performed.

3.1. Extraction of SDF Feature Descriptor and Eigenvector

As in this heading, they should be Times New Roman 11-point boldface, initially capitalized, flush left, with one blank line before, and one after.

SDF [13] is a kind of 3D model's geometric feature descriptors, it is defined as a real-valued function on each point of 3D model surface, and its core thoughts are shown as Figure 2. SDF calculation steps are as followings: suppose any vertex v departs from 3D model, cone C with a given field angle θ can be constructed in the negative direction of this vertex's normal. Then in this cane C , m rays $r_i, i = 1, \ldots, m$ are launched from the vertex to the bottom, and these rays r_i can produce corresponding intersection q_i with the surface on the other side of the model, the distance l_i from vertex v to intersection q_i can be calculated as the length of the ray segment. Analyze the mean value $l = \frac{1}{m}\sum_{i=1}^{m} l_i$ of all ray segments r_i length l_i and variance $\sigma = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(l_i - l)}$. Then choose a ray segment whose length l_i is limited within mean's one variance in ray segment r_i, that is, r_i will be met all in $|l_i - l| \le \sigma$, meanwhile, $i = 1, \ldots, s$, and s is regard as the sum of chosen ray segments. Calculate the chosen ray segments weighted mean, which is presented as q , that is, $q = \sum_{i=1}^{s}\omega_i l_i \Big/ \sum_{i=1}^{s}\omega_i$; and regard this q as the value of SDF feature on vertex v , in which weight ω_i is the reciprocal of angle between the corresponding ray segment l_i and cone's medial axis .

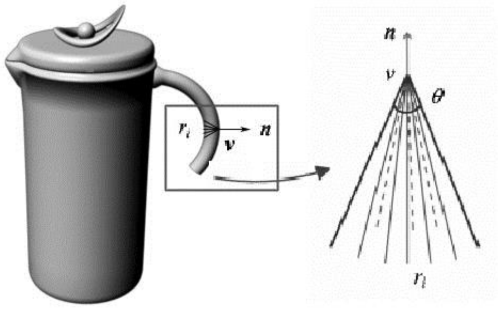

Figure 2. SDF's Core Thoughts

Because the weighted average length of ray segment changes weakly in the process of SDF evaluation, when 3D model's posture is changing, SDF is insensitive to model's posture. In addition, if all the vertexes' SDF values on the surface get normalized, $q^* = \frac{q - \min(q)}{\max(q) - \min(q)}$ can be formed; thus, the influence of model's size on q can be eliminated, meanwhile, the size of q can also keep invariance. SDF's advantages fully prove it is suitable for 3D model's classification mark.

This essay uses the following methods to construct 3D model's eigenvector: First, calculate the value of central point q^* on any face f_i of model's surface M , and mark it as q_i^*, then calculate the area η_i of f_i as weight. Next, get weighted histogram statistics of q_i^* on all the faces, that is, divide the interval of $[\min(q^*), \max(q^*)]$ equally as L subintervals, and then sum up the corresponding weights η_i of q_i^* among j subintervals $j = 1, \ldots, L$, and mark it as $W_j = \sum \eta_i$. Finally, process all the W_j in the normalization, then get W_j', that is, $W_j' = W_j \big/ (\sum W_j)$. As Figure 3 shows, this essay puts $[W_1', \ldots, W_L']$ as the eigenvector of 3D model M, and then gets its dimension $L = 100$.

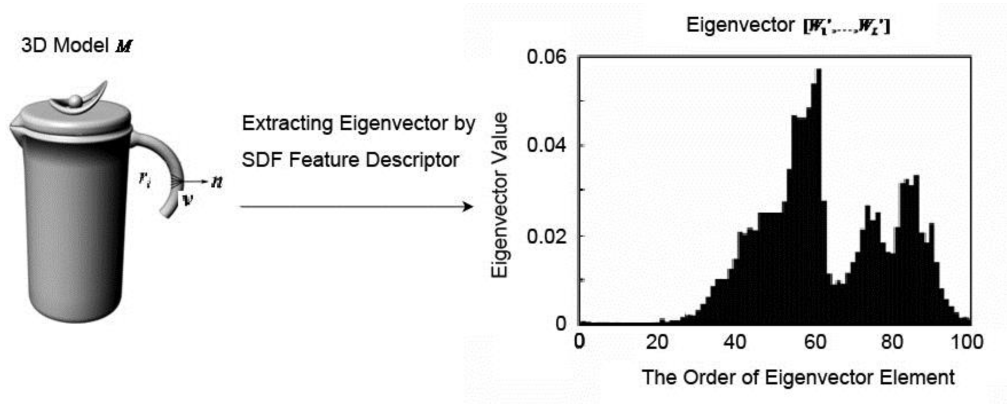

Figure 3. SDF Eigenvector Extraction

3.2. Construction of Viewpoint Extraction Algorithm Library

This essay uses the method of mixing similarities [14] to optimize the key points set checked, that is, to keep the data points in the same area, and then to eliminate the others, the normal of data points can be regarded as candidate viewpoints. The applicability of various viewpoints extraction algorithm being considered comprehensively, the essay chooses maximum projection area, surface area entropy, visible area ratio and mesh saliency algorithm to construct viewpoints extraction algorithm. Different algorithms have a good effect on some different 3D model viewpoint extraction.

Viewpoint extraction algorithm of maximum projection area is to measure the optimal viewpoint by calculating model's projection area in different perspectives. A good view usually has a larger projection area, and this algorithm defines the optimal viewpoint is the biggest perspective of projection area. And as long as this algorithm can suit the simple 3D model, then it can extract model's optimal viewpoint quickly.

Surface area entropy [4] can reflect model surface's degree of visibility and changeability. From some perspective p, surface area entropy can be shown as $N_e(p) = -\sum_{n=0}^{N_s} \frac{A_n}{A} \ln \frac{A_n}{A}$, N_s represents the number of visible surfaces in a scene; A_n is the projection area of visible surface n; and A is the total projection area of all surfaces visible. This algorithm can comprehensively show the users optimal viewpoint under the visible degree, especially in the viewpoint extraction of hand and tables models.

The algorithm of visible area ratio [5] can measure viewpoint's quality through the visible area ratio of corresponding viewpoint. Thus, the visible area ratio of viewpoint p can be shown as $N_r(p) = \frac{A_n}{A}$, A_n represents the area of visible surface n, A is the area of 3D model. This algorithm can be applied for the 3D model in which there is positive correlation between the visible area of optimal viewpoint and model's information amount.

Based on curvature, mesh saliency algorithm [6] is combined with gauss weighted average, which improve key point's capturing capabilities. In it, $\ell(v)$ represents the average curvature of vertex v, $N(v,\sigma) = \{v' \| \|v' - v\| < \sigma\}$ represents point set of vertex v within the range of distance σ, v' is the point on mesh model, $G(\ell(v),\sigma)$ is gauss weighted average value of average curvature on vertex v, $\zeta(v)$ is the saliency of v,

$$G(\ell(v),\sigma) = \frac{\sum_{v' \in N(v,2\sigma)} \ell(v') \exp[-\|v' - v\|^2 / (2\sigma^2)]}{\sum_{v' \in N(v,2\sigma)} \exp[-\|v' - v\|^2 / (2\sigma^2)]},$$

$$\zeta(v) = |G(\ell(v),\sigma) - G(\ell(v),2\sigma)|,$$

$$U(p) = \sum_{v' \in P(p)} \zeta(v'),$$

$P(p)$ is the set of visible vertexes in the perspective of p, $U(p)$ is the saliency under p. Compare the saliency values calculated from various viewpoints, then get the highest one, thus the viewpoint needed can be obtained. Mesh saliency algorithm is suitable for the 3D models which have geometric features, *e.g.* hair, clothing wrinkle, *etc.*

3.3. Construction of Decision Classifier and Viewpoint Extraction

According to geometric feature set of 3D model extracted, Adaboost iterative algorithm is applied to train and assemble various weak classifiers, then to construct the final decision classifier. This algorithm is done by changing data distribution to ascertain the weights of every sample, on the condition that every sample's classification in every training set and the last total classification are correct. The new data set got by corrected weights will be sent to lower classifier for training, and then combines all the classifiers got from every training to construct the final one.

Apply N models divided into Category M to construct training set, $\{(W_n^m, y_n^m), n \in 1,\dots,N, m \in 1,\dots,M\}$, W_n^m represents feature vector of the n^{th} model sample, $y_n^m \in \{-1,1\}$ is used to judge whether this model belongs to the mark of the m^{th} category. According to every model category, Adaboost binary classification algorithm is used to train a binary classifier. To geometric feature W_n of a inputting model, if the existing category mark is k and $H(W_n^k) = 1$, this model belongs to the category whose mark is k; if not, it belongs to another category, and it should enter the next binary classifier to be judged. According to the training set of binary classifier $\{(W_n, y_n), n = 1,\dots,N\}$, W_n is model's geometric feature, $y_n \in \{-1,1\}$ is the classification result, 1 represents right, -1 wrong. The training process is as followings,

Step 1 $D_1(n) = 1/N, n = 1,\dots,N$, sample weight of initialized training data.

Step 2 To weak classifier $t = 1,\dots,T$:

Step 2.1 Search the data samples whose weight is $D_t(n)$, which can produce the weak classifier h_t whose error value is the lowest in classification process;

Step 2.2 Evaluate $h_t = \arg \min_{h_i \in H} \varepsilon i$, in it $\varepsilon i = \sum_{n=1}^{N} D_t(n)(y_n \neq h_i(W_n))$, if the lowest error value is $\varepsilon_i < 0.5$, then continue; if not, withdraw from it and adjust training data feature again;

Step 2.3 Set up the weight of h_t, $a_t = \frac{1}{2}\log[(1-\varepsilon_t)/\varepsilon_t]$, in it ε_t is the lowest error value in Step2.2;

Step 2.4 Renew training data's sample weight: $D_{t+1}(n) = \frac{1}{Z_t} \cdot D_t(n)\exp(-a_t y_t h_t(W_n))$, Z_t will normalize data point's weight.

Step 3 Decision classifier H is superimposed by various weaker ones, and then $H = sign(\sum_{t=1}^{T} a_t h_t)$ will be got.

For inquiring 3D model, first extract the geometric feature W_x, then obtain optimal viewpoint extraction algorithm by decision classifier, finally extract the best view which is suitable for model's query.

4. Examples and Analysis

Combined with Matlab2014, algorithm example applies VS2013 to be realized on windows platform, hardware system configuration is CPU Intel Corei5 3450, RAM is Kingston 8GB DDR3 1600, graphics card is EISA-V GTS450 1GB DHV. Example data is got from 3D model library supplied in references [15], which is divided into 12 categories including human bodies, cups, glasses, chairs, octopus, pincers, birds, vases and so on, every category owns 20 models.

4.1. Analysis of Subjective Visual Sense Matching

The effect comparison of the best views got by various viewpoints extraction algorithm is shown as Figure 4, the optimal viewpoint extraction algorithm applied in this essay which is based on AdaBoost iterative algorithm not only fully represents different algorithms are suitable for their own particular models, but also makes up the limitations of a single measurement algorithm, then realizes algorithm's adaptive selection to model's geometric feature, finally extracts model's 2D view optimally. AdaBoost interactive algorithm applied here aims to do training and studying, while to the model categories which are not included in model library during training process, it also has the strong ability to study, and then can be done for classification study.

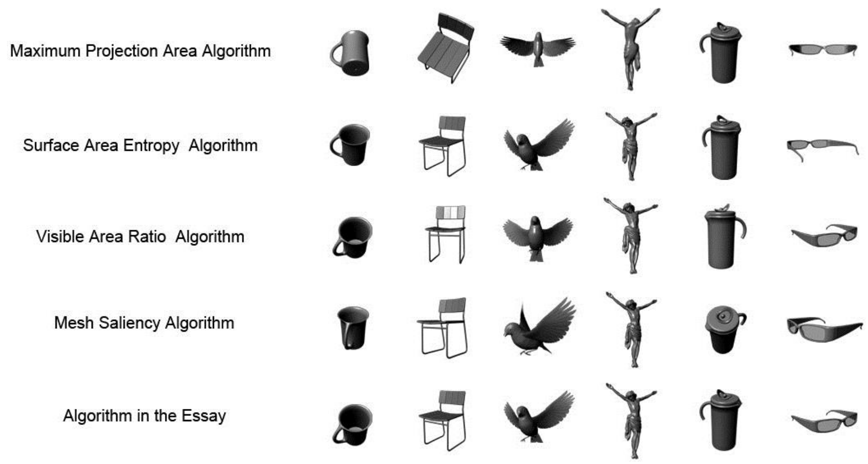

Figure 4. Comparison of Various Viewpoint Extraction Algorithms' Effect

4.2. Statistical Analysis of Function Comparison

This essay applies VSE [16] to do the statistical analysis on function for various viewpoint extraction algorithms:

$$M_s(v_m^a) = \min_v \frac{1}{\pi} GD(v, v_m^a),\ v \in \sum v_m^s,$$

$$vse = \frac{1}{S} \sum_s M_s(v_m^a)。$$

In it, $M_s(v_m^a)$ represents the lowest value of geodesic distance [17] $GD(v, v_m^a)$ between the optimal viewpoint v_m^a extracted from model m by algorithm and the one selected by user s , $\sum v_m^s$ shows the equivalent symmetrical point set of optimal viewpoint also selected by user s . Error measure VSF of final viewpoint selected is done by the average value of viewpoint selection error $M_s(v_m^a)$ from all of S test objects, the lower the value is, the higher the fitting degree of users' selection habit is. The statistical function comparison among the algorithm in this essay and the maximum projection area, surface area entropy, visible area ratio, mesh saliency, curvature entropy, polygon information projection, particle swarm optimization algorithm is shown as Table 1, which clearly performs the algorithm here is better than others on function.

Table 1. Statistical Analysis on Algorithms' Function

Viewpoint extraction algorithm	VSE
maximum projection area	0.521
surface area entropy	0.394
visible area ratio	0.469
mesh saliency	0.414
curvature entropy	0.485
Viewpoint extraction algorithm	VSE
polygon information projection	0.476
particle swarm optimization	0.420
algorithm in the essay	0.363

4.3. Analysis of Algorithm Stability

In order to test and verify the stability of algorithm used here, the best view is extracted again from 3D model which has been edited and revised. Because this essay applies AdaBoost iterative algorithm to select the optimal one for extracting the best view by training and studying, and this algorithm has the strong generalization ability, it also can be well suitable for new samples. In addition, model resolution ratio almost has no effect on geometric features; therefore, this algorithm also has the strong stability under different resolution ratio.

5. Conclusions

Based on AdaBoost iterative algorithm, this essay puts forward optimal viewpoint extraction algorithm, which has the strong adaptability for 3D model's geometric features. By this algorithm, the query models can be analyzed through classification study to get the viewpoint extraction algorithm which suits their geometric features, thus, the problem of semantic gap between users' perception and the best view can be solved effectively. And this algorithm also overcomes viewpoint optimization algorithm's limitation of a single geometric measurement in universality.

In the following research, viewpoint algorithm library should be further enriched to improve the ability of model classification, and then to promote matching accuracy between model features and viewpoint extraction algorithm. Furthermore, how to judge 3D model's view direction is also the problem which is worth focusing on in further studies.

References

[1] L. Yongjin, F. Qiufang, L. Ye and F. Xiaolan, "J. A distributed computational cognitive model for object recognition", Science China Information Sciences, vol. 9, no. 56, **(2013)**.

[2] V. Blanz, M. J. Tarr and H. H. Balthoff, "J. What object attributes determine canonical views", Perception, vol. 5, no, 28, **(1999)**.

[3] S. Abbasi and F. Mokhtarian, "Editors. IEEE Computer Society Press. Automatic view selection in mufti-view object recognition", Proceedings of the 15th International Conference on Pattern Recognition, Los Alamitos, January 13-16, **(2000)**.

[4] P. P. Vazquez, M. Feixas and M. Sbert, "Editors. ACM Press, Viewpoint selection using viewpoint entropy", Proceedings of the Vision Modeling and Visualization Conference, New York, November **(2001)**, pp. 273-280.

[5] O. Polonsky, G. Patane and S. Biasotti, "J. What's in an image?" The Visual Computer, vol. 8, no. 21, **(2005)**.

[6] C. H. Lee, A. VarShney and D. W. Jacobs, "J. Mesh saliency", ACM Transactions on Graphics, vol. 3, no. 24, **(2005)**.

[7] Y. Liming, W. Wencheng and W. Enhua, "J. Viewpoint selection by feature measurement on the viewing plane", Journal of Computer-Aided Design & Computer Graphics, vol. 9, no. 20, **(2008)**.

[8] C. Weiguo, H. Ping and L. Hu, "J. Canonical viewpoint selection based on distance histogram", Journal of Computer-Aided Design & Computer Graphics, vol. 9, no. 22, **(2010)**.

[9] X. Bonaventur, M. Feixas and Sbert M., "J. Information measures for object understanding", Signal Image and Video Processing, vol. 3, no. 7, **(2013)**.

[10] M. Mortara and S. Mortara, "J. Semantics-driven best view of 3D shapes", Computers & Graphics, vol. 3, no. 33, **(2009)**.

[11] Z. Yejia, S. Zhengxing and L. Chenxi, "J. A Method of Best View Selection of 3D Shapes Based on PSO", Journal of Computer-Aided Design & Computer Graphics, vol. 12, no. 26, **(2014)**.

[12] P. Xiang, C. Ao and Z. Guodong, "J. Best view selection based on statistics classification and edge features", Journal of Image and Graphics, vol. 8, no. 18, **(2013)**.

[13] L. Shapira, A. Shamir and D. Cohen-Or, "J. Consistent mesh partitioning and skeletonization using the shape diameter function", The Visual Computer, vol. 4, no. 24, **(2008)**.

[14] M. R. Ruggeri, G. Patane and M. Spagnuolo, "J. Spectral-driven isometry-invariant matching of 3D shapes", International Journal of Computer Vision, vol. 2/3, no. 89, **(2010)**.

[15] X. B. Chen, A. Golovinskiy and T. Funkhouser, "J. A benchmark for 3D mesh segmentation", ACM Transactions on Graphics, vol. 3, no. 28, **(2009)**.

[16] H. Dutagaci, C. P. Cheung and A. Godil, "Editors. ACM Press A benchmark for best view selection of 3D objects", Proceedings of the ACM Workshop on 3D Object Retrieval, New York, **(2010)**, pp. 45-50.

[17] L. Shapira, S. Shalom and A. Shamir, "J. Contextual part analogies in 3D objects", International Journal of Computer Vision, vol. 2/3, no. 89, **(2010)**.

Authors

Dong Tao, He received his master (2009) from China West Normal University. Now he is full lecturer at Digital Media Department in Engineering School of Eastern Liaoning University since 2002, mainly teaching the subjects of Graphic Image Processing, Three Dimensional Animation and other related ones. His current research interests include Computer Vision, Computer Graphic Image, and Three Dimensional Production Technology.