

A Simple Method for 3-Dimensional Photorealistic Modeling and Consideration of Reconstructing Error

Ippei Torii, Yousuke Okada, Masayuki Mizutani, and Naohiro Ishii

Faculty of Information Science, Aichi Institute of Technology,

Yachikusa 1247, Yakusa-cho, Toyota-shi, Aichi, Japan

mac@aitech.ac.jp, x07232xx@aitech.ac.jp, x09120xx@aitech.ac.jp, ishii@aitech.ac.jp

doi: 10.4156/jdcta.vol3.issue3.1

Abstract

The process of creating photorealistic 3-dimensional computer graphic (3DCG) images is divided into two stages, i.e., modeling and rendering. Automatic rendering has gained popularity, and photorealistic rendering is generally used to render different types of images. However, professional artists still model characters manually. Moreover, not much progress has been achieved with regard to 3-D shape data acquisition techniques that can be applied to facial modeling; this is an important problem hampering the progress of 3DCG. Generally, a laser and a highly accurate camera are used to acquire 3-D shape data. However, this technique is time-consuming and expensive. Further, the eyes may be damaged during measurements by this method. In order to solve these problems, we have proposed a simple method for 3-D shape data acquisition using a projector and a web camera. This method is economical, simple, and less time-consuming than conventional techniques. In this paper, we describe the setup of the projector and web camera, shape data acquisition process, image processing, and generation of a photorealistic model. Moreover, we show reconstructing error and improved accuracy.

Keywords

Photorealistic 3DCG, facial modeling, reconstructing error

1. Introduction

Recent developments in 3-dimensional computer graphics (3DCG) have made it possible to generate photorealistic images. Research is being conducted on various 3DCG applications and generation of photorealistic images is one of the main research topics. The generation of photorealistic images is divided into two stages, namely, modeling and rendering. Modeling is a process which defines and creates the data of facial shape and rendering is a process whereby the final image is generated from the

modeling data. The rendering is comparatively easy for automatic creation and it can generate photorealistic images from complicated form easily. However, modeling must be carried out manually by an artist, and its efficient improvement is late. Achieving highly precise geometric modeling is an important challenge in 3DCG[1]. Conventional methods such as stereo imaging and 3-D scanning are used for 3-D modeling. Stereo imaging makes use of a stereo camera that can simulate binocular vision, and 3-D scanning involves the use of a laser scanner and a CCD camera. In 3-D scanning, the vertical planes of an object are measured using the laser scanner by the 2-D scanning method, in which a method of measuring the object shape of carrying the cross section is developed. However, conventional methods have some disadvantages. Stereo imaging is difficult to use under varying light conditions because the shadow of the object is misinterpreted by the turbulence light. In 3-D scanning, we can acquire large high-density data by only one scan in short time; however, a large amount of memory is required because it is necessary to scan the object in all directions. In 3-D scanning, the object size in the passage direction is misinterpreted when the speed and direction of objective change in the measurement domain[5]. In this paper, we propose a new method that reproduces the shape data of a person's face on the computer. In comparison with the laser scan method, the proposed method is simple[9, 4].

2. Facial Modeling

It can be confirmed that the line is horizontal even if it is observed from anywhere when the horizontal line projected to the plane. However, the line has been often curved when this line is projected to complex ups and downs, and observed from the upper part. The shape of a face can be acquired by comparing and analyzing this difference with a web camera.

3. Stages in Facial Modeling

3.1 Environment

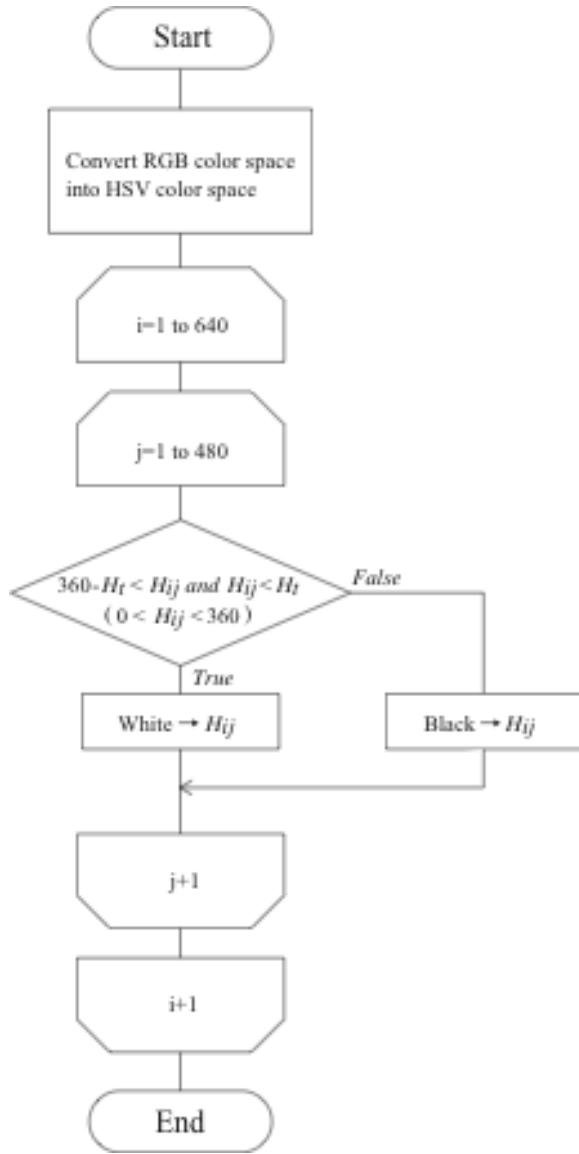


Figure 4. Flow of analysis of scan lines

3.4 Computation of coordinate values

The scanned data must be sampled along the x-axis, and the amount of data must be determined. Minute facial contours are considered as noise. According to the sampling theorem[8], the x-axis should be divided into at least 5 intervals. It calculates the coordinates on the basis of the analyzed image after sampling. The values of lc , ln , and hc have already been measured. The scan line which is projected from the projector, becomes a real image at point y_i on the base panel in Figure 5. Point y_i is also measured in section 3.2. However, when the object is placed in front of the base panel, it becomes a real image at point P of the object.

It holds the same as the point observed by point y' on the base panel by the web camera. And, the straight line of a , b , and c is linear function, therefore it is easy to estimate an intersection coordinates (x_P, y_P, z_P) . We show a schematic diagram of the coordinates calculation in Figure 5.

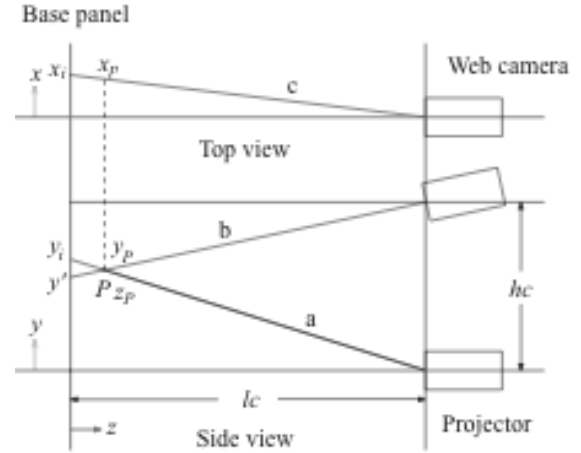


Figure 5. Computation of coordinate values

$$y = -\frac{y_i}{lc}z + y_i \dots\dots\dots (1)$$

$$y = \frac{hc - y'}{lc}z + y' \dots\dots\dots (2)$$

$$x = -\frac{x_i}{lc}z + x_i \dots\dots\dots (3)$$

$$-\frac{y_i}{lc}z_p + y_i = \frac{hc - y'}{lc}z_p + y'$$

$$-y_i z_p + lcy_i = (hc - y')z_p + y'lc$$

$$(hc - y' + y_p)z_p = lc(y_i - y')$$

$$z_p = \frac{lc(y_i - y')}{hc + y_i - y'} \dots\dots\dots (4)$$

$$y_p = \left(-\frac{y_i}{lc}\right)\left(\frac{lc(y_i - y')}{hc - y' + y_i}\right)$$

$$y_p = -\frac{y_i}{lc} z_p + y_i \dots\dots\dots(5)$$

$$y_p = -\frac{y_i(y_i - y')}{hc + y_i - y'} + y_i \dots\dots\dots(6)$$

$$x_p = -\frac{x_i}{lc} z_p + x_i \dots\dots\dots(7)$$

$$x_p = -\frac{x_i(y_i - y')}{hc + y_i - y'} + x_i \dots\dots\dots(8)$$

$$P = \begin{pmatrix} x_p \\ y_p \\ z_p \end{pmatrix} = \begin{pmatrix} -\frac{x_i(y_i - y')}{hc + y_i - y'} + x_i \\ -\frac{y_i(y_i - y')}{hc + y_i - y'} + y_i \\ \frac{lc(y_i - y')}{hc + y_i - y'} \end{pmatrix} \dots\dots\dots(9)$$

3.5 Conversion of modeling data to polygonal data

The acquired facial data are converted to polygonal data by a general software application. Representing objects in the form of polygons is a standard modeling technique used in 3DCG as the object can be easily edited. Therefore, we adopt the DXF-3DFACE[6] file format. The simple file structure of DXF simplifies the process of mapping facial data to polygon coordinates. Furthermore, since many software applications support DXF, it is possible to import DXF data from one application to another. We show rendered image in Figure 10 of Appendix.

4. Verification, Comparison, and Evaluation of the Acquired Data

4.1 Error estimation

We defined equations for error estimation from section 3.4.

$$C = \frac{y_i - y'}{hc + y_i - y'} \dots\dots\dots(10)$$

$$z_p = lcC \dots\dots\dots(11)$$

$$y_p = -y_iC + y_i \dots\dots\dots(12)$$

$$x_p = -x_iC + x_i \dots\dots\dots(13)$$

We calculate the error values. We can see that the accuracy of zP depends on lc. Therefore, in order to reduce the error in zP, the value of lc needs to be controlled. We can also see that xP depends on xi, which means sampling internal (number of partitions in vertical direction) of x-axis. Further, yP (accuracy of projector and web camera) depends on yi (capture angle).

In our method, equation has to satisfy yi-y'>0 and hc>0. However, if yi-y'>0 and hc>0 are not large enough, then it will result in a large error margin from Table 2, Table 3 and Table 4 of Appendix.

4.2 Verification of acquired data by geometric form

It is necessary to measure the dimensions of a known object and verify them with the acquired data to measure the accuracy of the proposed method. In this paper, we verify the accuracy of our proposed method by using a geometric form. A geometric form is a solid model used for sketches etc., whose sizes are known. The accuracy of acquisition data is evaluated by using two geometric forms. The accuracy of this method is verified by comparing the actual dimensional values of the solid model and the measurement data acquired by this method. The actual dimensional values and the measurement data acquired by this method are shown in Figure 6, Figure 7 and Table 1.

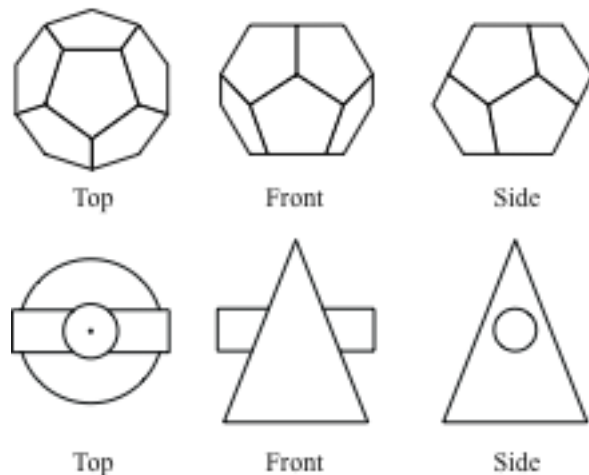


Figure 6. Drafting of geometric form

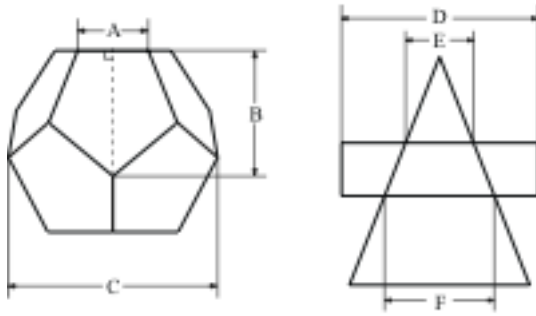


Figure 7. Dimensions of the solid model

Table 1. Measurement data

	A	B	C
Actual size	70 mm	105 mm	175 mm
Measured value	68 mm	103 mm	171 mm
Difference	- 2 mm	- 2 mm	- 4 mm
	D	E	F
Actual size	185 mm	80 mm	100 mm
Measured value	182 mm	77 mm	98 mm
Difference	- 3 mm	- 3 mm	- 2 mm

According to Table 1, line c (Figure 5) can be confirmed that the margin of error is the largest when full scale is compared with measurement scale. Therefore, x_P is related to equations (7), (8), (10), and (13) which means length of x-axis. From equations (10) and (13), the possibility that the problem occurs is low because x_P does not depend on l_c . Moreover, y_i and y' has the influence to all variables. As a result, it is guessed that there is a factor in x_i . The x_i means sampling interval of x-axis (number of partitions in vertical direction). This error margin in accuracy is due to less sampling rate. Therefore, the sampling rate needs to be increased in order to reduce the error margin.

4.3 Improved accuracy of reconstruction by using stereo matching

We have increased the number of cameras from one to two[2]. Our aim is improved accuracy. We show in Figure 8 and Figure 9. We measure the length between the two cameras (lh), and we get values of θ and lt using equation (14). This is the calibration. Using this method we can easily adjust the position, because our method of calibration does not depend on the angle of the cameras. After that, we conduct computation of

value from camera A (Figure 9(a)) and camera B (Figure 9(b)). We correct the image in camera B (Figure 9(b)) from equation (15)(16). (X, Y, Z) are coordinates of a 3D point in the world coordinate space. (u, v) are coordinates of point projection in pixels. (c_x, c_y) is a principal point (that is usually at the image center). f_x and f_y are focal lengths expressed in pixel-related units. The joint rotation-translation matrix $[R|t]$ is called a matrix of extrinsic parameters. k_1 and k_2 are radial distortion coefficients, p_1 and p_2 are

$$r^2 = \frac{x^2}{z} + \frac{y^2}{z}$$

tangential distortion coefficients. We have shown the correct data as image in Figure 9(c). We show experimented data of scan lines in Figure 11 and Table 5 of Appendix.

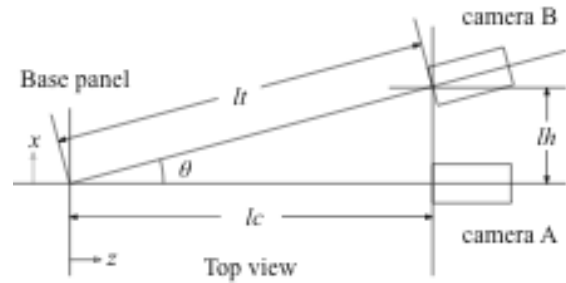


Figure 8. Two camera setup

$$lt = \frac{lc}{\cos(\arctan \frac{lh}{lc})} = \frac{lh}{\sin(\arctan \frac{lh}{lc})} \dots \dots \dots (14)$$

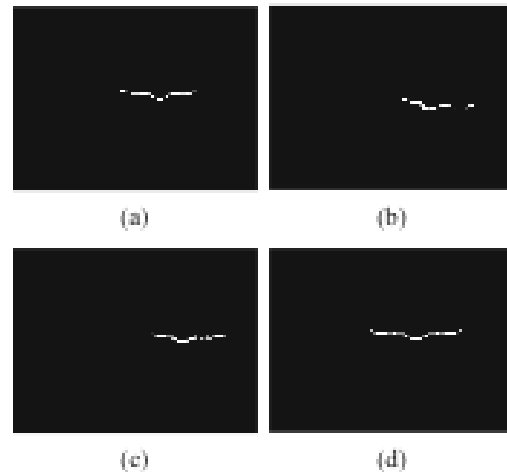


Figure 9. Image processing to correct two camera

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = R \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + t \dots\dots\dots(15)$$

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} fx[\{\frac{x}{z}(1+k_1r^2+k_2r^4)\} + \frac{2p_1xy}{z^2} + \\ p_2\{r^2 + 2(\frac{x}{z})^2\}] + cx \\ fy[\{\frac{y}{z}(1+k_1r^2+k_2r^4)\} + \frac{2p_2xy}{z^2} + \\ p_1\{r^2 + 2(\frac{y}{z})^2\}] + cy \end{pmatrix} \dots\dots(16)$$

Then, we synthesize the image in camera A to correct the image (the image in camera B) based on the point of maximum y-scale (that point being the top of nose) to get a more accurate image in Figure 9(d) and data in Table 6. According to the Table 5, we have a satisfactory result as accuracy with two cameras has improved ±0.5mm, as compared with one camera. Stereo matching with two camera proposed here, shows higher accuracy than reference [2].

5. Conclusion

In this paper, an economical, simple, and less-time consuming method for generation of photorealistic images is proposed. The image processing method and calculations for this study are described. Furthermore, muscle animation[3] can also be applied if muscular motions of the face are compiled into a database[7]. The simplicity of our method of facial modeling will be applied to the entrance checking information, the interface of the robot and the medical support apparatus etc[10].

6. Reference

[1] T. Akimoto, Y. Suenaga, R. Wallace, and K. NTT. Automatic creation of 3D facial models. *IEEE Computer Graphics and Applications*, 13(5):16–22, 1993.
 [2] D. An, A. Woodward, P. Delmas, G. Gimelfarb, and J. Morris. Comparison of active structure lighting mono and stereo camera systems: Application to 3d face acquisition. In *ENC '06: Proceedings of the Seventh Mexican International Conference on Computer Science*, pages 135–141, Washington, DC, USA, 2006. IEEE Computer Society.
 [3] B. Bickel, M. Lang, M. Botsch, M. Otaduy, and M. Gross. Pose-space animation and transfer of facial details. In

2008 ACM SIGGRAPH. In *Eurographics Symposium on Computer Animation*, 2008.
 [4] A. Ghosh, T. Hawkins, P. Peers, S. Frederiksen, and P. Debevec. Practical modeling and acquisition of layered facial reflectance. In *SIGGRAPH Asia '08: ACM SIGGRAPH Asia 2008 papers*, pages 1–10, New York, NY, USA, 2008. ACM.
 [5] Y. Lee, D. Terzopoulos, and K. Walters. Realistic modeling for facial animation. In *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 55–62, New York, NY, USA, 1995. ACM.
 [6] S. Ochiai. *DXF handbook*. Ohmsha, Ltd, Tokyo, 2003.
 [7] S. Platt and N. Badler. Animating facial expressions. *ACM SIGGRAPH Computer Graphics*, 15(3):245–252, 1981.
 [8] S. Smale and D. Zhou. Shannon sampling and function reconstruction from point values. *BULLETIN-AMERICAN MATHEMATICAL SOCIETY*, 41(3):279–306, 2004.
 [9] D. Terzopoulos and K. Waters. Physically-based facial modeling, analysis, and animation. *Journal of visualization and Computer Animation*, 1(2):73–80, 1990.
 [10] T. Vetter and V. Blanz. A morphable model for the synthesis of 3D faces. In *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, pages 187–194, 1999.
 [11] CV Reference Manual, http://opencv.jp/opencv-1.0.0/org/docs/ref/opencvref_cv.htm

Appendix

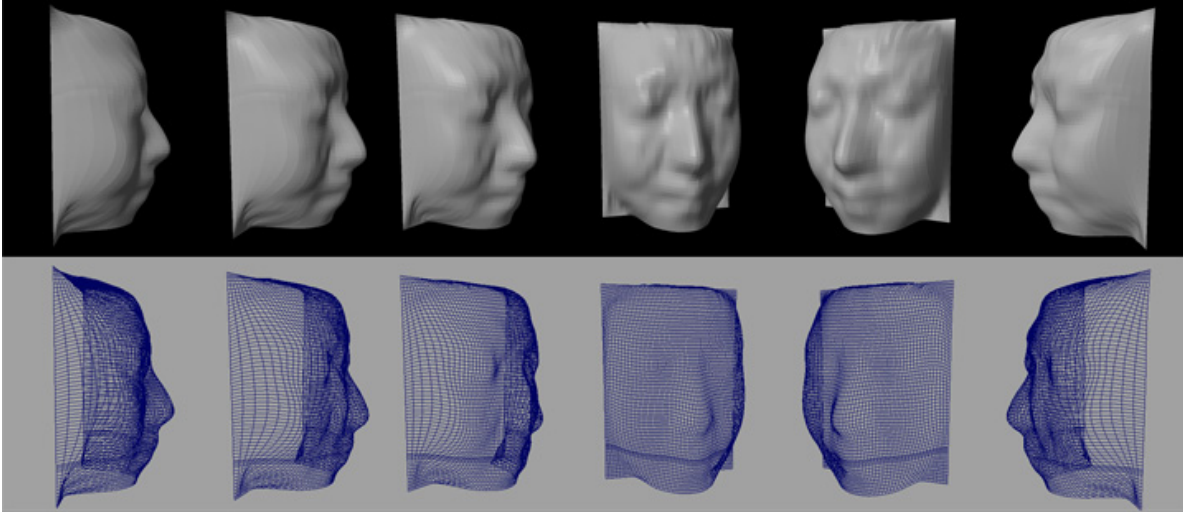


Figure 10. Render image of outputted DXF

Table 2. Experimented data about lc

Result of current value			$lc+2$		
x_p	y_p	z_p	x_p	y_p	z_p
0.000000	33.016228	0.926123	0.000000	33.016228	0.928287
.....				
60.522446	33.016228	0.926123	60.522446	33.016228	0.928287
58.480278	33.521816	29.778305	58.480278	33.521816	29.847879
.....				
105.589394	33.521816	29.778305	105.589394	33.521816	29.847879
109.276642	33.016228	0.926123	109.276642	33.016228	0.928287
.....				
178.765381	33.016228	0.926123	178.765381	33.016228	0.928287

Table 3. Experimented data about hc

Result of current value			$lc+2$		
x_p	y_p	z_p	x_p	y_p	z_p
0.000000	33.016228	0.926123	0.000000	33.016132	0.920665
.....				
60.522446	33.016228	0.926123	60.522831	33.016132	0.920665
58.480278	33.521816	29.778305	58.492283	33.518845	29.608698
.....				
105.589394	33.521816	29.778305	105.611069	33.518845	29.608698
109.276642	33.016228	0.926123	109.277336	33.016132	0.920665
.....				
178.765381	33.016228	0.926123	178.766525	33.016132	0.920665

Table 4. Experimented data about yi

Result of current value			$lc+2$		
x_p	y_p	z_p	x_p	y_p	z_p
0.000000	33.016228	0.926123	0.000000	35.090595	5.965381
.....				
60.522446	33.016228	0.926123	60.165764	35.090595	5.965381
58.480278	33.521816	29.778305	58.147194	35.523708	34.484161
.....				
105.589394	33.521816	29.778305	104.987991	35.523708	34.484161
109.276642	33.016228	0.926123	108.632637	35.090595	5.965381
.....				
178.765381	33.016228	0.926123	177.711853	35.090595	5.965381



Figure 11. An experimented 3-D object and data

Table 5. Data of scan lines : Experimented 3-D object

x_p	y_p	z_p	x_p	y_p	z_p
0.000000	127.920967	0.845682	82.713150	115.064682	138.407913
0.000000	127.920967	0.845682	82.713150	115.064682	138.407913
0.666008	127.920967	0.845682	82.576210	114.953651	139.595963
.....				
71.262894	127.920967	0.845682	90.428383	115.400002	134.819946
71.262894	127.920967	0.845682	90.990051	115.400002	134.819946
66.295120	121.661209	67.825035	93.453835	115.966393	128.759598
66.295120	121.661209	67.825035	93.611160	116.080811	127.535301
.....				
69.112564	121.393837	70.685913	96.937454	116.426399	123.837494
69.597870	121.260880	72.108597	97.266609	116.658752	121.351379
69.097168	120.733826	77.748100	100.283791	117.965408	107.370171
69.453445	120.473129	80.537537	100.457596	118.086662	106.072708
.....				
70.382339	119.575226	90.145050	103.103210	118.699310	99.517395
70.853401	119.448769	91.498146	103.283775	118.823128	98.192566
70.603920	119.197189	94.190056	105.609222	119.197189	94.190056
70.947411	118.947380	96.863068	106.389832	119.322762	92.846466
.....				
71.752228	118.575920	100.837624	107.553261	119.702133	88.787140
72.340355	118.575920	100.837624	108.342873	119.829491	87.424423
72.548653	118.208344	104.770760	109.329834	120.085571	84.684395
73.006973	118.086662	106.072708	109.930550	120.085571	84.684395
.....				
75.664986	117.844566	108.663139	111.325020	120.603241	79.145332
75.534531	117.724144	109.951645	112.131363	120.733826	77.748100
75.336220	117.010246	117.590370	112.943909	121.260880	72.108597
75.207878	116.892677	118.848312	113.554420	121.260880	72.108597
.....				
77.856705	116.195625	126.306847	115.613205	121.795631	66.386749
78.292976	116.080811	127.535301	117.082794	122.338242	60.580776
79.327255	115.512527	133.615967	126.541588	127.920967	0.845682
79.889862	115.512527	133.615967	127.207596	127.920967	0.845682