



3D MODEL RECONSTRUCTION ALGORITHM AND IMPLEMENTATION BASED ON THE MOBILE DEVICE

¹CHONGWEN WANG ²MINGLI BAO ¹TONG SHEN

¹School of Software, Beijing Institute of Technology, 100081, Beijing, China

²School of Computer Science, Beijing Institute of Technology, 100081, Beijing, China

ABSTRACT

With the mobile phone has been developed rapidly now, 3D reconstruction applied in this mobile device becomes a possible. Based on the photometric stereo, a 3D model reconstruction algorithm had been put forward, which can be run on mobile-phone with android system. In the dark environment, through several light sources provided by the mobile-phone screen, multiple perspective images of an object can be captured, then each value of depth can be calculated. Finally based on depth data, OpenGL ES have been used to map out the object model. Through experiments, the proposed algorithm is effective.

Keywords: *Three-Dimensional Reconstruction, Photometric Stereo, 3D Model, Android Application*

1. INTRODUCTION

3D reconstruction, as human vision research mainly, is one important direction of the computer vision field. It is mainly to adopt certain technology to create the 3D shape of an object model. With the Marr's computing theory was pushed, 3D reconstruction won the breakthrough progress, emerged many substantial algorithms and reconstruction methods. Among them, the 3D reconstruction based on 2D images, due to the low cost, convenient modeling, fast rendering, etc, is widely respected to use, the main has the following three methods:

Reconstruct the Geometric Model Based on Silhouette Contour

When we use perspective projection method to observe an object in the space from a visual angle, a silhouette contour belongs to the object will appear in the screen, the silhouette contour and perspective projection center form this object's shell, when the observation angle is much, the shell will approach to the real shape of an object infinitely. Because this method requires a lot angle of the image, so it will bring problem to the image's calibration.

Using Three-dimensional Visual Method to Reconstruct the Geometry Model

Stereo vision method have been the favor of people, it is mainly through much image calibrated to rebuild the object model. Based on the triangle measuring principle, many measured points' projection connections would make a 3D coordinate belong to the object. All 3D coordinates will form the shape of an object together. Although

this method is easy relatively, but it requires the object not to be keep out.

The Modeling Method Based on Region Matching

This method, through the digital image processing technology, using the image segmentation method to find some blocks with the same texture feature in different pictures, then in this basis, calculate the depth value of pixel and its strips corresponding to three dimensional space, further to build out the 3D model. Although this method requires low image, it could not be effective to apply to the objects that the curvature of the surface don't change significantly.

Comprehensive consideration, this paper will use photometric stereo technology based on the stereo vision method to achieve the 3D reconstruction of the object in the mobile phone. Photometric stereo technology is a method about modeling 3D object through the illumination change on the surface of the object. Because the image has three components, light, surface normal and surface shader, so if we take the certain object, we just change the light conditions, through the relationship between the image brightness and illumination change it can be calculated and analyzed, we can get the object's surface accurate and depth. Finally according to the depth value, draw out the corresponding object model.

This paper will introduce the algorithm, and give the corresponding application implementation: in the dark environment, with the light provided by the mobile-phone screen, user photographs the object's multiple images, then through the algorithm it calculates the depth value and maps out

the 3D model of the object. Through the practical application, it proves that it is feasible to use this algorithm and way to achieve the object's 3D reconstruction

2. PHOTOGRAPHING PROGRAM REALIZATION

Because this paper will use the photometric stereo technique to make the 3D reconstruction of the object, according to the requirements, it should has the three images at last in the different light sources, this chapter will be focus on how to create the lights with the different positions on the cell phone, and get the corresponding image.

2.1 Choose the Light Source

In order to generate the three images that the object is under the different illuminations, the traditional way is to build three parallel light sources, in turn to illuminate the object surface, and shoot down the current state of the object in the same location. But for mobile phone equipment, from the point of view of users' accessibility, it is not practical to require users to provide light source by themselves. Although most mobile phones own the flash function, but it is the only one and can't reach the point of three lights. So, how to create the different location's light sources in the mobile phone devices is a very important problem and need to be solved.

Here, in this paper, the solution is using the cell phone's screen brightness as the light source. For today's mobile phones, its screen has reached the 2.0 inches above, and even the 3.7 inches and 4.3 inches, and if we transfer the screen brightness to the maximum in the dark environment, it can be an incandescent lamp to light up the object's surface. More convenient, in the Android system, we can use the programming to control the display mode of the screen brightness and provide the effective solution to create the different light sources, the display mode is like the following figure (2.1)

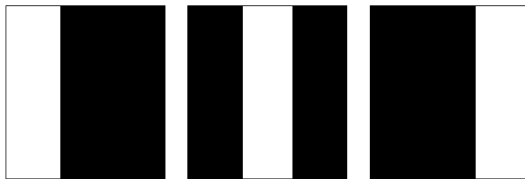


Figure 2.1: Create Three Different Light Sources By The Screen Brightness

(2.1) figure's way is set the background of the mobile phone's screen to white according to the order left, middle and right, the corresponding two

other positions is black, such as the left picture, when the left of background is white, the middle and right of the background is black. So, when we transfer the brightness of the screen to be the maximum, the background for white section will provide illume, and black section will not provide it, three different light sources will be created out, the positions are left, middle and right, they will be the different light sources to light up the surface of the object.

If necessary, we still create more light source's mode based on the screen brightness, it depends on the number of images that users want to photograph, as long as reasonable to use each position of the screen, we can create the corresponding different light sources. The following diagram (2.2) and (2.3) to create four different light source mode. The mode one divides the screen into the four parts, up, low, left and right, the mode two divides the screen into the four parts, upper left, upper right, lower left and lower right, the way is the same as creating three light sources, which in turn to set the background as white, and the other part is black.

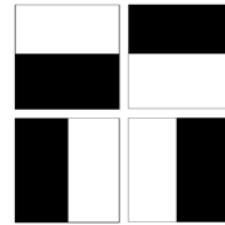


Figure 2.2 : The One Mode Which Create Four Light Sources

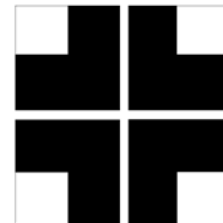


Figure 2.3: The One Mode Which Create Four Light Sources

2.2 The Switch Order of the Lights

As described in the last section, through divide the screen into the different parts, and make the background of each part to be white color in turn, we can realize the plan that it creates more lights. Finally, we hope to do is in the every light that we can photograph the current object images, so we need to adopt a mechanism to control the order of the light switch, for example using the first way

about four light sources, the first image is photographed in the first half part as a light source, then the light source switch to lower part as the light source for the program to photograph.

We need to take N pictures of the object $\{I_j : j = 1 \dots N\}$, and every image must be taken in the different light sources L_j , so to the every pixel on the screen in the current j picture, we can use the follow formula to determine the white point or black point:

$$L_j(x, y) = \text{sign}\left(x \cdot \cos\left(\frac{2\pi j}{N}\right) + y \cdot \sin\left(\frac{2\pi j}{N}\right)\right) \quad (1)$$

-1 represent the black, 1 represent the white, and (0,0) is the center of the screen, x and y coordinate rank according to it. For example taking four pictures, and use the first way using four light sources, then when we take the first picture, the x and y are the pixel located in the left side of the screen, due to the $x \cdot \cos\left(\frac{2\pi j}{N}\right) + y \cdot \sin\left(\frac{2\pi j}{N}\right)$

returns to positive, so the sign function will return 1, which is white; and the right pixels, because the return is negative, so the sign function returns -1, which is black. Through this, we can determine the left screen is white and right screen is black when we take the first image.

2.3 The Realization Of The Photograph Programming

Because the photometric stereo algorithm need three object images taken in three different light sources, so in the choice of the light source introduced before the section, we use the Camera interface in the Android platform to implement the function taken the object images in the different light sources.

The specific implementation procedure is as follows (like the 2.2 picture):

1. create the light source

Like (2.2) picture, we need to divide the screen into four parts about up, down, left, right, and make the background color to be white in turn. Here, create the up, down, left, right views to be the four specific screen parts, and set the parameter on the current layout, show them in the specific position in the screen, the following is the set about up:

```
<View
    android: id="@+id/up"
    android: layout_width = "wrap_content"
    android: layout_height = "fill_parent"
    android: layout_weight = "1" >
```

```
</View>
```

```
left = (View)findViewById(R.id.left)
```

Through it, we can create four views in the four parts of the screen, to each view, we can use the method to set them to be white in the need time:

```
left.setBackgroundColor(Color.WHITE)
```

set the four views to be white background in turn as the style (2.2) style, then adjust the screen brightness to the supreme, the screen can be four different light sources.

2. realize photograph function

The Camera interface in the Android platform allows us to call the function of the cell phone Camera.

First, we create the Camera object, and set the picture captured by camera as the prospect preview:

```
mCamera = Camera.open(1)
mCamera.startPreview()
```

Then call the touch events of the mobile phone, capture the operation of the screen real-time:

```
onTouchEvent(MotionEvent event)
```

Here, what we specify is photograph the object when the finger is pressed the phone screen:

```
case MotionEvent.ACTION_DOWN
    mCamera.takePicture(null, null, new
    PictureCallbackListener())
```

The parameter PictureCallbackListener() called by the takePicture method is the callback method about the camera image, through this method, we can get the picture taken by the camera, and operate them.

Finally, we realize the concrete operation of the callback method PictureCallbackListener():

```
public void onPictureTaken (byte[] data,
    Camera camera) {
    bitmap =
    BitmapFactory.decodeByteArray( data,
    0, data.length)
    .....
}
```

Through transfer the byte array of the storage image into Bitmap format, which becomes a

photograph of the object. Of course, we can also specify the path to store our pictures for photometric stereo algorithm:

```
new File
(Environment.getExternalStorageDirectory ()
+ "/CameraPicture", "p.jpg")
```

So far, we complete the photograph programming used to get the 3D reconstruction image, according to the requirements of the algorithm, as long as user is in the darkness environment, and put the object wanted to take photo on the mobile phone screen, through the photograph programming, we can get the pictures of the object in the different light sources. Picture 2.4 shows the four face images taken by the photograph programming in the four different light sources.



Figure 2.4: The Picture Of The Face In The Different Light Sources

3. 3D RECONSTRUCTION ALGORITHM BASED ON PHOTOMETRIC STEREO TECHNOLOGY

3.1 Obtain The Normal Value Of Each Surface Point From The Photometric Stereo Image

According to Lambertian scattering model we can learn that the brightness values of each point on the surface id depended on its reflectance value and normal value. The below formula (2) shows it, S_0 is the direction of the light source, N_0 is the unit vector of each point, I_0 represents each point's brightness value.

$$I_0 = \eta N_0 \cdot S_0 \quad (2)$$

Since the normal of each point is depended by the direction of x, y, z parts, so here we use vector to express the above N: $N = (N_x, N_y, N_z)^T$. Then the (2) is represent the equations about N_x , N_y , N_z , and carry them on the corresponding solution, we can get the final normal value of each surface point.

It suppose that the three images are $I = (I_1, I_2, I_3)$, and their three light source direction are:

$$S_1 = (s_{11}, s_{12}, s_{13}) \quad S_2 = (s_{21}, s_{22}, s_{23})$$

$$S_3 = (s_{31}, s_{32}, s_{33})$$

$$S = \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix}, \text{ according to formula (2),}$$

it can form a linear equations, each one represents the surface normal value of a light source.

$$\begin{cases} I_1 = \eta(s_{11}N_x + s_{12}N_y + s_{13}N_z) \\ I_2 = \eta(s_{21}N_x + s_{22}N_y + s_{23}N_z) \\ I_3 = \eta(s_{31}N_x + s_{32}N_y + s_{33}N_z) \end{cases} \quad (3)$$

In the above formula (3), we can set the direction of the light source. The brightness of the three images can also be calculated through the color value, so the only mystery η will become a obstacle about calculating the normal value. Because the η will be different depends on the model object, it is not fixed, so it is very difficult to be confirmed. But finally what we want is normal value represented each surface points, we don't care about that if it is the unit, so solving a detail is not necessary, just need to know the value of ηN .

Base on this, we can use matrix to change it:

$$I = \eta S \cdot N, \text{ and } \hat{N} = \eta N.$$

Through the matrix inverse we change the form of the formula:

$$\hat{N} = S^{-1}I \quad (4)$$

According to the above formula to calculate, we can get the final normal value of each surface point, the whole operation process is not difficult. All the operation process describes as follow, there are four steps:

1. Set the pixels of each image are $M \times N$, and the brightness is $I = (I_1, I_2, I_3)$, the direction of every light source is:

$$S_1 = (s_{11}, s_{12}, s_{13}) \quad S_2 = (s_{21}, s_{22}, s_{23})$$

$$S_3 = (s_{31}, s_{32}, s_{33})$$

Using matrix to describe, define

$$S = \begin{bmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{22} & s_{23} \\ s_{31} & s_{32} & s_{33} \end{bmatrix}$$

2. Normalize the brightness of the image,

$$I(i, j) = \frac{I(i, j)}{\max_{1 \leq i \leq M, 1 \leq j \leq N} \{I(i, j)\}}$$

$$(i = 1, 2, \dots, M, j = 1, 2, \dots, N).$$

3. $\hat{N}(i, j) = \eta N(i, j) = S^{-1}I$
($i = 1, 2, \dots, M, j = 1, 2, \dots, N$)

4. $\hat{N}(i, j) = \hat{N}(i, j) / \|\hat{N}(i, j)\|$, $\hat{N}(i, j)$ is the normal value of each surface point.

Using the above algorithm to solve the normal value of each surface point, relevant examples below figure (3.1) and figure (3.2). Through the recovery normal value, we can see that it has the 3D effect initially, it shows that it is a feasible scheme using the normal value to create object model.

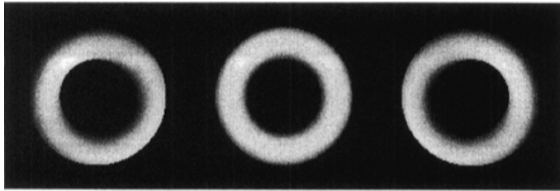


Figure 3.1: Three Ring Image In The Three Light Sources

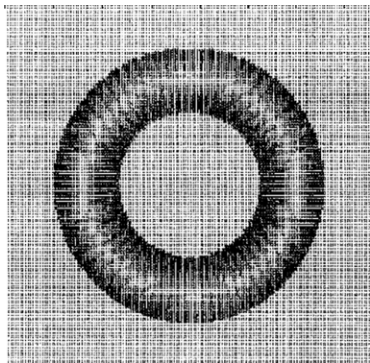


Figure 3.2: The Normal Value Of The Ring Calculated By The Algorithm

3.2 Using the Polydrons to Calculate the Depth Value of Each Point From the Normal Value

The polydrons is an object composed of many surface patches, such as triangles and so on. In the last section, through the formula we can get the normal value of each surface point, base on it, we use the polydrons to calculate the corresponding depth value.

To the object in the 3D space, its x and y coordinate information are known, only the depth information in the z coordinate is an unknown, we need to calculate. When we know the (x,y), the normal value in the last section is relate to the depth value of the z coordinate, base on this relationship, we can establish the linear equation to calculate the depth value of each point finally.

Using the polydron models, we assume that the object surface is made by many triangles, each one is corresponding to three points on the surface of the object:

$$P_0 = (x_0, y_0, z_0) \quad P_1 = (x_1, y_1, z_1) \quad P_2 = (x_2, y_2, z_2)$$

And the normal value of every patch is:

$$N = (N_x, N_y, N_z)^T.$$

Because the normal value of the patch relates to its x, y and z information, so we can make the follow equations:

$$\begin{cases} N_x = (y_1 - y_0)(z_2 - z_0) - (z_1 - z_0)(y_2 - y_0) \\ N_y = (z_1 - z_0)(x_2 - x_0) - (x_1 - x_0)(z_2 - z_0) \\ N_z = (x_1 - x_0)(y_2 - y_0) - (y_1 - y_0)(x_2 - x_0) \end{cases} \quad (5)$$

In the above equations, the unknown $z_i (i = 0, 1, 2)$ is the depth value we want finally, to each point, we always make the same equations, it will be the equations represented the depth value of the surface of the object. And these are ultra-linear equations, we can use the least square method to calculate them.

Assume that we use the images that their pixel is $M \times N$. We use the grid to resolve these points, forming $2(M-1) \times (N-1)$ triangle pitch. As shown in Figure (3.3), for each pitch, it has two unknown depth value, so $2(M-1) \times (N-1)$ triangle pitch will have $4(M-1) \times (N-1)$ equations.

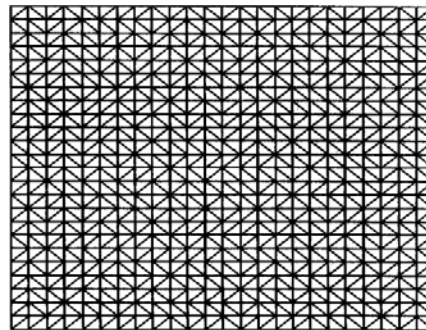


Figure 3.3: The Division Way To The Triangles Of The Surface

When we get the normal value of each surface point, we can use polyhedron model to calculate the

corresponding depth data, the algorithm process is as follow:

1. For each surface point, we use the way like picture (3.3) to form $2(M-1) \times (N-1)$ triangle pitches, and every pitch's normal value is N. Here we use T_k to express the k pitch.
2. Three points made up the k pitch are $P_0 = (x_0, y_0, z_0)$ $P_1 = (x_1, y_1, z_1)$ $P_2 = (x_2, y_2, z_2)$, and each point's normal value is N_1, N_2, N_3 .
3. According to the relationship about the normal value of each triangle pitch and the position of the three points, we establish the corresponding equations:

$$\begin{cases} aN_x = (y_1 - y_0)(z_2 - z_0) - (z_1 - z_0)(y_2 - y_0) \\ aN_y = (z_1 - z_0)(x_2 - x_0) - (x_1 - x_0)(z_2 - z_0) \\ aN_z = (x_1 - x_0)(y_2 - y_0) - (y_1 - y_0)(x_2 - x_0) \end{cases} \quad (6)$$

$$[N_x N_y N_z] = (N_1 + 2N_2 + N_3) / 4$$

4. Because of the convenience of the matrix calculating, we change the above formula to matrix, using matrix A to express depth value z and matrix B to express the normal value of the left side of the equations and coefficient, forming the matrix equations $Z = (A^T A)^{-1} \cdot (A^T B)$.
5. Using the least square method to solve this super linear equations.

Matrix A stores the factor of the depth value z, these factor is made by the value x and y, so the value of matrix A depends on the pixel of the image. To the result of A and $A^T A$, if every image's pixel is same, then it can be reused. And the matrix $A^T A$ has some features, as follow, assume that the size of images is $M \times M$:

- a) $A^T A$ is 0 out the diagonal line position, and it is the matrix made by pieces;
- b) $M \leq 5, |A^T A| = 0$;
 $M > 5, |A^T A| \neq 0$;
- c) $M \leq 200, cond(A^T A)$ is $10^{17} \sim 10^{18}$;

Because of the above features, using the images which pixel is $M < 200, N < 200$ will spend few time to calculate $A^T A$. If the pixel of images is out this scope, we can split these images into many small images which the pixel is in this scope.

As long as every image has the same pixel, then there is a rule about the structure of the matrix

$A^T A$, we assume the images' pixel is $M \times M$, then the structure is as follow:

$$A^T A = \begin{bmatrix} I_1 & E & & O \\ E & I_2 & \dots & \\ & \dots & \dots & \dots \\ & & \dots & I_2 & E \\ O & & & E & I_1 \end{bmatrix} \quad (7)$$

O is the zero matrix, I_1, I_2, E is the $M \times M$ matrix,

$$I_1 = \begin{bmatrix} 2 & -1 & & & O \\ -1 & 4 & \dots & & \\ & \dots & \dots & \dots & \\ & & \dots & 4 & -1 \\ O & & & -1 & 2 \end{bmatrix}$$

$$I_2 = \begin{bmatrix} 4 & -2 & & & O \\ -2 & 8 & \dots & & \\ & \dots & \dots & \dots & \\ & & \dots & 8 & -2 \\ O & & & -2 & 4 \end{bmatrix}$$

$$E = \begin{bmatrix} -1 & & & & O \\ & -2 & & & \\ & & \dots & & \\ O & & & & -1 \end{bmatrix}$$

By using this algorithm, we can calculate the depth value of each point when we know the normal value of each surface point, then through this depth value we can create the 3D model of the object. As shown in Figure (3.4) and (3.5), we can see that the effect is very good.

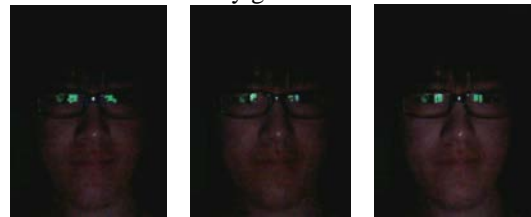


Figure 3.4: Three Images Of Face In The Different Light Sources

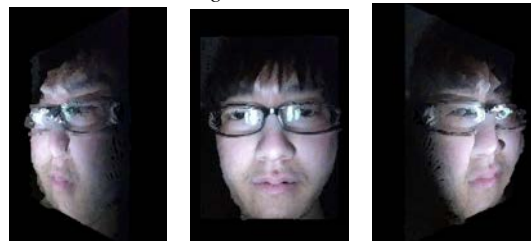


Figure 3.5: A 3D Mode Of The Face

4. DRAWING THE 3D MODEL BASED ON THE OPENGL ES

Through the camera program, we can take the image of object in some light sources. On the basis of these images, we use photometric stereo algorithm to get the pixel depth image of the object. Now, we need to do is to draw the 3D model through the depth information of the depth image. Here, we use the OpenGL ES graph library on the Android mobile system.

The steps are as follow:

1. Drawing $M \times N$ points;
2. Call `glDrawElements`, and use the pattern `GL10.GL_TRIANGLES`, so we can draw out a plane;
3. Use one of the images as the texture to put on the plane;
4. According to the depth value of each point, we adjust the pixel point on the Z coordinate;

After adjusting all the points, we can create the reality 3D model finally.

Using the camera program, we take the images of the object in the certain environment, the following Figure (4.1) is the four images of face in the four light sources:



Figure 4.1: The Images Of The Face In The Four Light Sources

Through using photometric stereo algorithm to calculate the four images, we get the depth value of the face, and create the face model by the OpenGL ES, like the Figure (4.2):

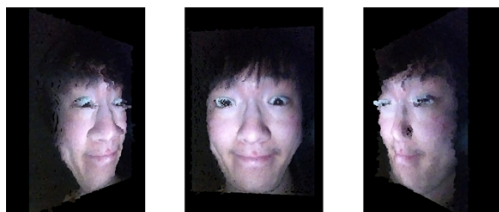


Figure 4.2: 3D Model Of The Face

From the above Figure (4.2), the face model's features have a very good expression, like the nose, mouth, eyes and so on. So we can verify that the photometric stereo algorithm explained in the chapter 3 can generate the precise 3D model by the

images in the different light sources, and it can be used on the telephone very well. The next Figure (4.3) shows more 3D effect, on the left it is the picture, on the right it is the model reconstructed.



Figure 4.3: 3 Reconstruction Examples

5. CONCLUSION

With the 3D areas such as game, film production, digital city, virtual reality and so on are in the rapid development, more and more people begin to pay attention to the 3D model technology. At the same time, in the recent years the development of the mobile phone is in advance by leaps and bounds, especially the smart phones, embedded CPU makes the phone has a handle function similar to the PC. Whether performance or hardware support, now mobile phone is powerful. So 3D reconstruction technique applies on the mobile phone is an effective way of 3D reconstruction and also a kind of reducing the cost of 3D reconstruction.

This article realizes the function which reconstructs the 3D object through the cell phone mobile. The way we use is: in the dark environment, through the light sources create by the programming with the screen, we use the camera of the phone mobile to take the images of the object, then use the photometric stereo algorithm to calculate these images and get the depth values, finally use the OpenGL ES in the Android platform to create the 3D model.

Overall, this paper not only determine the 3D reconstruction method applied to cell phone from the theoretical research, but also realize the application in the Android system, it has certain theoretical significance and application value. We can say that it is a try that 3D reconstruction application is used in the mobile phone.



REFERENCES:

- [1] Mariappan, M.B, Xiaohu Guo, Prabhakaran B, "A Computer Vision-based Gesture Recognition and 3D Gaming System for Android Mobile Devices", Proceedings of the 2011 IEEE International Symposium on Multimedia, 2011, pp. 19-26.
- [2] Yoon Dong-Hyeog, Yang Kyung-Soo, Choi Choon-Bum, "Three-dimensional wake structures and aerodynamic coefficients for flow past an inclined square cylinder", *Journal of Wind Engineering and Industrial Aerodynamics*, February 2012, pp. 34-42.
- [3] Chaurasia Hemant K, Thompson Mark C, "Three-dimensional instabilities in the boundary-layer flow over a long rectangular plate", *Journal of Fluid Mechanics*, August 25, 2011, pp. 411-433.
- [4] Bu`yu`katalay. S, Birgu`l. O, Halici. U, "Effects of light sources selection and surface properties on Photometric Stereo error", *2010 IEEE 18th Signal Processing and Communications Applications Conference*, 2010, pp. 336-9.
- [5] Guan Yudong, Wang Peng, Chen Hua, Wang, Yong, Zou Yang, "A system of three-dimensional reconstruction based on CT images", *2010 International Conference on Biomedical Engineering and Computer Science*, 2010.
- [6] Koch. L, Kearney. P, Erbel. R, Roth. Th, Ge. J, Brennecke. R, Meyer. J, "Three dimensional reconstruction of intracoronary ultrasound images: roadmapping with simultaneously digitised coronary angiograms", *Computers in Cardiology*, 1993, pp. 89-91.
- [7] Song LiMei, Wen YuHua, "Three-dimensional reconstruction and evaluation based on CT images", *Proceedings of the 1st International Workshop on Education Technology and Computer Science*, ETCS 2009, 2009, v2, pp. 44-48.
- [8] Jian Wang, Defeng Chen, Tuan Zhao, "Research of Three-Dimensional Reconstruction Based on Computer Vision", *International Conference on Wireless Communications, Networking and Mobile Computing*, 2010, pp. 4.
- [9] Kang Hyun-Deok, Jo Kang-Hyun, "3D reconstruction of planar objects using the properties of plane and vanishing points from a single panoramic image", *The 10th World Multi-Conference on Systemics, Cybernetics and Informatics, Jointly with the 12th International Conference on Information Systems Analysis and Synthesis*, ISAS 2006 – Proc, 2006, pp. 342-346.
- [10] Wei Jiang, Sugimoto S, Okutomi. M, "Panoramic 3d Reconstruction Using Stereo Multi-perspective Panorama", *International Journal of Pattern Recognition and Artificial Intelligence*, Sept. 2010.