

DESIGNING SOFTWARE FOR 3D OBJECT MODELLING USING DIGITAL CLOSE RANGE PHOTOGRAMMETRY

A. Zamirroshan

Mahab Ghodss Consulting Engineers, Tehran, Iran- abz1352@yahoo.com , zamirroshan@mahabghodss.com

KEYWORDS : Software, Design, Point Cloud, Modelling, Non-metric, Calibration, Matching

ABSTRACT:

Digital close range photogrammetry is extensively used because of its efficiency and significant decreasing time needed to collect data and process them and also applicability in different environmental conditions. To use this technology in an optimum way it needs image processing and computations in the form of a software that can meet our requirements of applying this technique. In this paper different stages of designing a software which use Matlab and Visual Basic environments for preparing 3D model of an object in the form of point cloud is explained. The software uses sequence of images with the same resolution taken by different or the same non-metric cameras besides coordinates of control points as input. Then an operator must zoom and click on control points in images to prepare coordinate of control points in image coordinate system and select one image (generally middle image) as base or template image, this is the only stage that needs operator. To compute interior and exterior parameters of cameras, bundle adjustment is performed. Then using image matching technique and epipolar geometry points of edges of base image is searched in other images. Applying coordinates of corresponding points and collinearity equation, object space coordinate of point is computed in a least square adjustment and its position accuracy is evaluated. Then all accepted points (irregular point cloud of object) is filtered to prepare a regular point cloud. All these stages are done automatically without operator interference. In this paper above mentioned parts of software are explained and finally some practical applications of the software presented.

1. INTRODUCTION

Close Range Photogrammetry is practically applicable when the goal of using this technique and its necessary software are known. The purpose of designing presented software in this paper is to prepare 3D model of objects in scale of images taken by a simple non-metric digital camera. Finally 3D model is prepared in the form of point cloud. This process uses a digital camera as imaging device and a computer equipped with the software as processor and there is no need for stereo plotters or softcopy or other hardwares for 3D view of the object being modelled. To bypass 3D view of the object image matching technique and epipolar geometry exploited.

2. APPLICATIONS OF CLOSE RANGE PHOTOGRAMMETRY

Close range photogrammetry has many applications and it is growing as we go, because of its low cost and ease of use, short time needed to collect data and process them, compared with other methods. Some of interesting and well known applications of close range photogrammetry are :

- 3D reconstruction
- Human face modelling
- Deformation measurements
- Deformation monitoring
- Industrial metrology (car and aerospace industries)
- Implants design and reconstructive plastic surgery
- Wound measurements
- Digitization for cad modelling
- Circularity check and alignment
- Growth rate research

3. NECESSITY FOR DESIGNING SOFTWARE

In all above mentioned applications there is a need for image measurements and computations according to a particular task we consider. So to do this task accurately and in a minimum period of time and effectively we have to design a software. The core sections of this software is multi image bundle adjustment for computing camera interior and exterior parameters and image matching to find conjugate points in images. In case we want to find points automatically not manually to supply input for forward intersection.

4. DIFFERENT STAGES OF A CLOSE RANGE PROJECT

To apply close range photogrammetry (in a specific project), after determining the object which is supposed to be modelled, it is necessary to design control points. Control points are those points that are clearly visible in images, and their coordinates (X,Y,Z) are known in a reference coordinate system. This system is preferably defined as its XY plane perpendicular to horizon and Z axis towards location of cameras (and perpendicular to XY plane) as shown in Figure 1. Geometric positions of control points must form volume i.e. they must not lie on only one plane, otherwise DLT and collinearity equations will not obtain reasonable solution for unknowns. After designing control points, images have to be taken from different points of

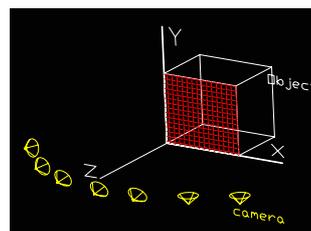


Figure 1. Reference Coordinate System Defining Object Space

view (at least two locations) the way object and control points can be seen in images. It is not necessary that all control points be visible in all images, but in each image at least 6 control points have to be seen to solve collinearity or DLT equations. The next stage is to extract image coordinates of control points in each image.

Assuming images have been taken by a non-metric camera we calibrate cameras using a suitable mathematical model (11 parameters DLT equations, or collinearity equations). In this process unknown parameters such as camera orientation angles (Ω, Φ, κ) and perspective center coordinates and focal length of each camera will be solved. Then if a real point in object space be visible in more than one image, object space coordinates of that point can be computed by method of intersection.

To extract image coordinates of points in images, one image is selected as middle image or base image then edges of this image is extracted by canny edge detection method and then points of edges of base image is searched in other images using image matching technique.

Finally we have image coordinates of a specific point in all images, so it is possible to calculate ground coordinates of that point by intersection method in a least square manner.

If texture on the object surface is not appropriate for applying image matching after determining camera orientation parameters it is possible to select two images and introduce orientation parameters to a stereo plotter or softcopy and after orientation of images start to measure the object and model it but in case surface texture of object be good enough to support matching, the need for 3D vision of images in a stereo plotter or softcopy will be discarded.

5. REQUIRED FUNCTIONS (SUBROUTINES)

All above mentioned stages after taking images are done by the software and will be explained.

5.1 BASIC COMPUTATIONAL FUNCTIONS

5.1.1 PREPARE SETTINGS

This function uses following parameters as inputs and prepare a setting file for other functions:

- Image number of middle image or base image
- coordinates of Search window corners in base image (boundary of object in base image)
- Images resolution (dimensions in pixel)
- Reference window size (in pixel)
- Max z in object space according reference coordinate system
- Min z in object space according reference coordinate system

5.1.2 DLT METHOD

This function computes direct linear transformation (DLT) parameters and then calculates camera orientations $x_0, y_0, f, \omega, \phi, \kappa$ from the 11 Parameters calculated by DLT. It uses image No as input. Finally computed values will be employed as starting values (initial values) for non linear least square adjustment in calibration process

5.1.3 NON LINEAR SYSTEM LSQ

Since solving unknown parameters in camera calibration by collinearity condition is based on a non linear system of equations, NON LINEAR SYSTEM LSQ function takes all non linear equations (for each image point two equations) in string form and also unknown parameters names and their initial values as input and then solve system of equations. Because of iterative nature of solving non linear systems, this function takes a value as tolerance or threshold to exit iteration loop. (if corrections to unknowns in each iteration is less than tolerance NON LINEAR SYSTEM LSQ will stop)

5.1.4 MULTI MATCH

This function uses two inputs:

- Base image No
- Vector of other images numbers

Then using Normalized Cross Correlation method it finds conjugate points in other images. Output file consist following files:

- ALLMATCHPOINTS file contains all conjugate image points coordinates
- LOCALP<base image No><j> file contains conjugate points between base image and image number j

5.1.5 MULTI IMAGE BUNDLE ADJ

This function designed to compute camera calibration parameters and camera orientations and locations with respect to reference coordinate system defined for object space. It exploits NON LINEAR SYSTEM LSQ to solve unknowns, input for this function is vector of image numbers and output is a file for each image that contains following parameters:

- (x_0, y_0, f) = camera interior orientation parameters
- (ω, ϕ, κ) = camera rotations in reference coordinate system
- (X_L, Y_L, Z_L) = location of camera in object space coordinate system
- (K_1, K_2, P_1, P_2) = symmetrical and decentering lens distortion parameters

5.1.6 MULTI GRND

After finding corresponding points this function uses output of MULTI MATCH and performs forward intersection to calculate ground coordinates of a point. It also yields RMS of X, Y, Z coordinates of points. So the inputs are:

- File ALLMATCHPOINTS
- Base image No
- Vector of other image numbers

and outputs are:

- MULTIGRND<base image No>.txt which contains X, Y, Z coordinates of object point cloud
- RMS of object point coordinates

5.1.7 CLOSEPREC

After intersection and computing all object points, to evaluate precision of intersection and number of points that lies in a specific range according their RMS in X, Y, Z, this function prepare a report on following values:

- RMS of sigma X's
- RMS of sigma Y's
- RMS of sigma Z's

- Mean of sigma X's
- Mean of sigma Y's
- Mean of sigma Z's
- Max sigma X
- Max sigma Y
- Max sigma Z
- Min sigma X
- Min sigma Y
- Min sigma Z
- No of points with accuracy of (X,Y,Z) Between :
 - ✓ 0 to 0.1 (in object space units)
 - ✓ 0.1 to 0.5
 - ✓ 0.5 to 1
 - ✓ 1 to 2
 - ✓ 2 to 4
 - ✓ 4 to infinity

5.1.8 GP

At the beginning of a project and computations we have to prepare image coordinates of control points in each image, with this function operator can easily zoom and click on control points then enter number of that point and do this for all control points in image to obtain input for DLT METHOD. input for this function is only image No. this function also prepare image resolution (dimensions) for PREPARE SETTINGS.

5.1.9 FILTER POINT CLOUD

After forward intersection and computing ground coordinates of object always there are some mismatched points or gross errors in points calculated by intersection, these points actually doesn't belong to object surface. this is because of wrong coordinates of conjugate points obtained by image matching. To remove some of these points and also obtain a regular point cloud on object, this function divides object space in cubes and replace all points that located inside one cube with cube corner. Cube size and frequency of points inside a single cube are selected by operator. this function works in Visual Basic 6 environment as an independent application.

5.1.10 CREATE DXF FILE

DXF file is a standard format that most graphical softwares support it. to visualize point cloud of object surface we need to prepare DXF file of resulting regular or irregular point cloud that can be seen in a CAD software such as AutoCad. This function that is written in Visual Basic 6 take an X,Y,Z text file of point cloud with format NO,X,Y,Z and prepare DXF file of that file.

5.2 SECONDARY COMPUTATIONAL FUNCTIONS

5.2.1 FORM ONE POINT COLLIN EQ

The function uses x,y (image coordinates) and (X,Y,Z) ground coordinates of a point and form two collinearity equations for each point. these equations are in terms of unknown parameters XL,YL,ZL, omega, fi, kapa, x0,y0,f, K1,K2,P1,P2

5.2.2 FORM ONE IMAGE BUNDLES

For each image point two collinearity equations can be formed, so to form all possible equations from all control points in one

image, this function uses image No and output of GP for that image as input and obtain all collinearity equations in image.

5.2.3 GET_IMAGE_COOR

As mentioned before in matching process, one image is selected as base image. to extract points on edges of this image this function uses black and white form of base image as input and yields an N by 2 matrix that is coordinates of points on edges in base image.

5.2.4 GROUND POINT

In matching process it is necessary to compute ground coordinate of a point in an image getting image No, image coordinates of point and its z coordinate in object space as input. This is done by collinearity equations and image orientation parameters. We have inputs as :

- Image No
- (x,y) : Image coordinates
- Z : (object space z coordinate of point)

And output is

- (X,Y) : object space coordinates of point

5.2.5 IMAGE POINT

In image matching process it is necessary to compute image coordinates of a point in a particular image given (X,Y,Z) ground coordinates of that point. this function uses following inputs :

- Image No
- (X,Y,Z) : ground coordinates of point

and the output is :

- (x,y) : image coordinate in given image (image No)

5.2.6 I2I

This function uses (x,y) image coordinates of point in one image and also z of that point in object space and calculate image coordinates of this point in another image, so inputs are:

- (x1,y1) : image coordinates of a point in image No1
- Image No1 : first image number
- Z coordinate of point in object space
- Image No2 : second image number that coordinates of point in it is required

And out put is

- (x2,y2) : image coordinates in image No2

5.2.7 FIND MATCH

To find conjugate point in other images, since searching all areas of other images is so time consuming, it is possible to limit the search area to a line. For each point in object space with coordinates (X,Y,Z1) we can calculate (x1,y1) in a particular image. in a similar way we can find (x2,y2) for that point in object space but with different Z coordinate (X,Y,Z2). If take Z1 and Z2 maximum and minimum Z in object space we have a line between (x1,y1) and (x2,y2).

This function uses (x,y) coordinates of point in one image and points of calculated line in image we want to find match as input and find the best match according normalized cross correlation method. so the inputs are :

- (x,y): image coordinates of a point in base image

- epipoints: points of line calculated according max and min z in object space
 - Limage :matrix of left image (Base image)
 - Rimage: matrix of right image (Other images)
 - Window size :matrix dimension of a window that normalized cross correlation is calculated in it
- And output is :
- (xi,yi) :best match of point (x,y) on line epipoints

5.3 OTHER FUNCTIONS

5.3.1 MAT2CAD

Matlab image coordinate system is defined as origin at the upper left of image and x axis is toward right side of image and y axis toward bottom of image as shown in figure 2.

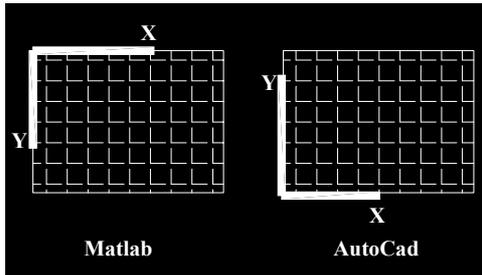


Figure 2. Matlab and Autocad Coordinate System

When operator zoom and click on images to get image coordinate of control points x,y coordinates are in matlab coordinate system .to transform these coordinates to autocad system (that origin is bottom left and x axis toward right side of image and y axis is toward top of image) MAT2CAD uses x,y coordinates in matlab and get x,y coordinates in autocad system.

5.3.2 CAD2MAT

This function is reverse of MAT2CAD and transform x,y coordinates in Autocad system to matlab image coordinate system. This is necessary because FIND MATCH in function MULTI MATCH uses coordinates in matlab system.

5.3.3 FUNCVAL

This function simply calculates value of a function according given parameter names and their values

5.3.4 GET KERNEL

To evaluate correspondence of two points in two images it is needed to calculate correlation coefficient between these two points but this is done in a window i.e correlation of two windows are evaluated . for each point on EPIPOINTS (explained in FIND MATCH) assuming points are at center of windows. To get that portion of image to compute correlation in it this function uses following inputs :

- Image matrix (matrix of grey values of each image pixel)
- (x,y) : coordinate of center of window
- window size (in pixel)

and output is a square matrix with size of window size that consists grey values of image. this matrix is input for GET CORCOEF.

5.3.5 GET CORCOEF

After taking matrices of grey values of two windows in left (for example base image) and right image it is possible to compute normalized cross correlation coefficient for these two windows. So the input for this function is windows (matrices) of left image and right image candidate points .

6. SOME EXPERIMENTS

To test the software , camera Dlink DSC 350F which is a simple non-metric digital camera selected , maximum resolution of this camera is 640x480 pixels and since sensor dimensions are not known all image coordinates used in computations are in pixel . to define object space coordinate system an object such as a small box almost in size of the object choosed or a calibration box prepared . some or all of images taken for the object modeling and resulted point clouds are shown in following figures :

6.1 Inner surface of a plate

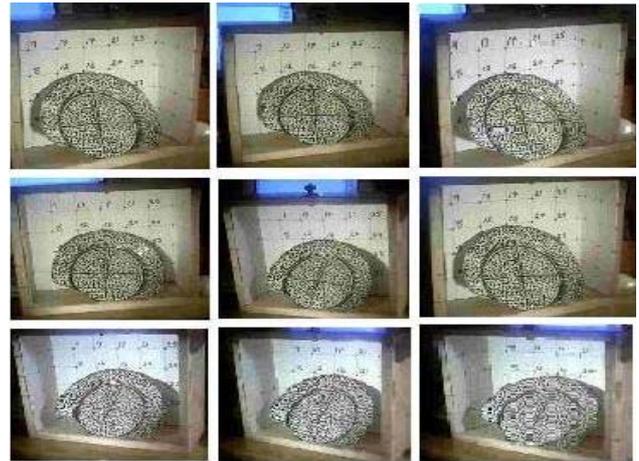


Figure 3. Images taken in different directions from a plate inside calibration object

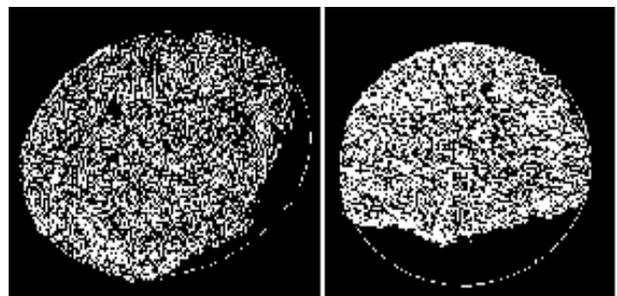


Figure 4. Resulting point cloud in two different views

6.2 3D model of a cellphone



Figure 5. Two selected images taken from a cellphone with a small box as calibration object

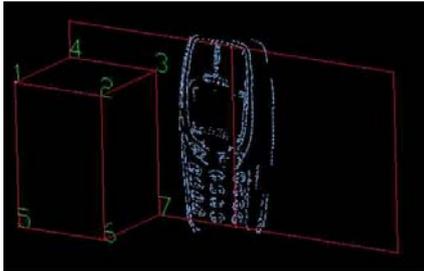


Figure 6. Point cloud of surface of cellphone after manual filtering of gross errors

6.3 modeling surface of an egg



Figure 7. Three Images of an egg and an small box as calibration object

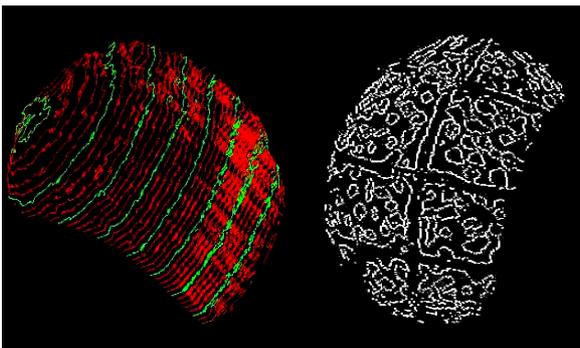


Figure 8. One mm contour lines of egg surface created from final point cloud after filtering gross errors

Figure 9. Final 3d point cloud of a portion of egg surface

7. CONCLUSION

The software test results shows that quality of object point cloud in reference coordinates system greatly depends on precision of image matching process and finding exact conjugate points ,so it is necessary the object surface to be

textured well . in above mentioned examples (except cellphone) the surface of objects were coloured to form a good texture on the object surface to support image matching .

This also can be done by a video projector that projects a random noise image on the object surface.

To check precision of geometric quality in object space ,some distances on the created point cloud and on the object compared and it proved that if camera calibration results and matching process be accurate enough , these results are also reliable.

8. REFERENCES

References from Books :

Tony C.T. Tang ,Digital Image Correlation

References from Websites :

A. Malian, F.A. Van del Heuvel ,A. Azizi ,A Robust Photogrammetric System For Wound Measurement

Ahmed Elkasher , Mohammed Elghazali ,Ashraf Sayed ,Yasser Elmanadilli ,Assessment Of Two Cheap Close Range Feature Extraction Systems

C. Rigotti , N.A. Borghese , S. Ferrari , G. Baroni , G. Ferrigno ,Portable And Accurate 3D Scanner For Breast Implants Design and Reconstructive Plastic surgery

Fabio Remondino , 3D Reconstruction Of Articulated Objects From Uncalibrated Images

Fabio Remondino , Image Sequence Analysis for Human Body Reconstruction

Fanq-Jen Chen , Application Of Least Squares Adjustment Technique To Geometric Camera Calibration and Photogrammetric Flow Visualization

Ibrahim Zeroual and AbdelKrim Lizaid , Use Of DLT In Photogrammetric Metrology

Jian Xu ,Z.P. Fang ,Andrew A. Malcolm,H. Wang , A Robust Close Range Photogrammetric system For Industrial Metrology

Ming Zhou ,Clive S. Fraser , Automated Surface Extraction In Real Time Photogrammetry

Nicola D'Apuzzo ,Human Face Modeling From Multi Images

Paul Siebert and Stephen Marshall , Human Body 3D imaging By Speckle Texture Projection Photogrammetry

Russel Morison ,Weird and Wonderful Applications Of Close Range Digital Photogrammetry

T.A. Clarke and Dr X. Wang , Extracting High Precision Information From CCD Image

Tim Suthau , John More , Albert Wiedemann and Jens Franzen , Close Range Photogrammetry With CCD Cameras and Matching Methods Applied To Fracture Surface Of An Iron Bolt

X. Wang and T.A. Clarke , An Algorithmic Method For Real Time 3D Measurement