

AUTOMATED EXTRACTION OF BREAK LINES IN TLS DATA OF REAL ENVIRONMENT

K. Kitamura^{a,*}, N. D'Apuzzo^b, N. Kochi^a, S. Kaneko^c

^a Imaging and Measuring Laboratory, R&D Center, Topcon Corporation, 75-1, Hasunuma-cho, Itabashi-ku, 174-8580 Tokyo, Japan - (k_kitamura, n.kochi)@topcon.co.jp

^b HOMETRICA CONSULTING, Culmannstrasse 59, CH-8006 Zurich, Switzerland

^c Hokkaido University Graduate School of Information Science and Technology, Kita14, Nishi9, Kita-ku, Sapporo, 060-0814, Hokkaido, Japan

Commission V, WG V/3

KEY WORDS: TLS, Point Cloud, Segmentation, Surface, Extraction, Algorithm, Three-dimensional, Measurement

ABSTRACT:

We are making research to find a novel method to extract the break-lines of a building from the point clouds obtained through a Scanner (Terrestrial Laser Scanner: TLS). The algorithm we have developed this time is to segment the plane surfaces and point clouds and to extract the line of two intersecting planes, as the break-line. But since our algorithm is not yet completely free from the errors in calculating of algorithm and measuring of TLS, it is used as yet primarily for pre-processing steps, such as creation of a plan or a modeling or assisting to understand object form or making rough sketch of an object. But even for these preliminary works, up to now, we were obliged to resort to manual works and to the use of various applications. So, now we have developed a novel system to extract all automatically the surface and break-lines, which would greatly improve the efficiency of the whole operation. We have also developed a system to convert the point clouds data into the polygon data of surface and lines, much to the alleviation of the setback due to the heavy data accumulation.

Besides, in order to facilitate the measurement using TLS in the real environment, our algorithm makes it possible to segment the planes and to extract the break-lines of objects, while eliminating automatically all the disturbing noises created by pedestrians, cars, trees, and grasses intervening between the TLS and the objects.

1. INTRODUCTION

In the concrete real situation, when we use TLS to take in the point clouds, we are often disturbed by the noises, created by humans, cars and plants coming in between the TLS and the objects. There are already many valuable reports written on the extractions of break-lines of building structures out of the point clouds and on the production of their modelling (Stamos, 2002; Chen, 2007; Konno, 2007). Some of them even speculate on sensor noises (Jiang, 1999) or creating noises artificially (Zhang, 2001; Mitra, 2003), but there is no report, so far, which deals effectively with the noises of humans, cars and plants like our present report. In the past, we removed the noises only manually, but now this is of no avail any more with all these recent developments of scanning technology which creates hundreds of millions of point clouds in a short period of time. Besides, the ever growing size of the objects demands more than one scan. So, we have made an automatic system to cope with all the related problems.

The result of the indoor and outdoor experiments of our system will be explained in the following order:

2. GENERAL OUTLINE OF PROCESS

With our algorithm we extract the planar and break-line in the following order.

First: the production of 2dimensional range image out of the point cloud data. Second: the extraction of planar / curved surface as the pre-processing step of the segmentation. Third:

the segmentation. Fourth: the extraction of the break-line (See: Figure 1).

The parameter in the explanation below is the value obtained experimentally out of several samples.

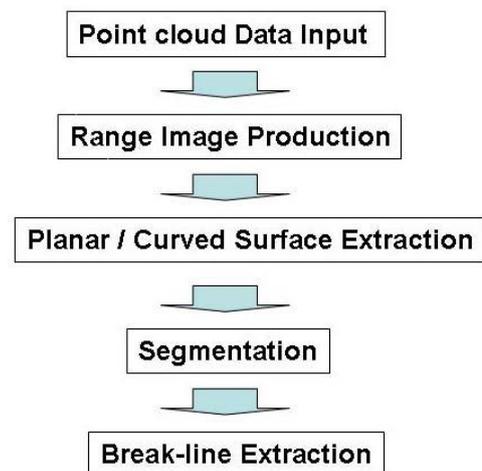


Figure 1. Flow chart of Process

* Corresponding author. This is useful to know for communication with the appropriate person in cases with more than one author.

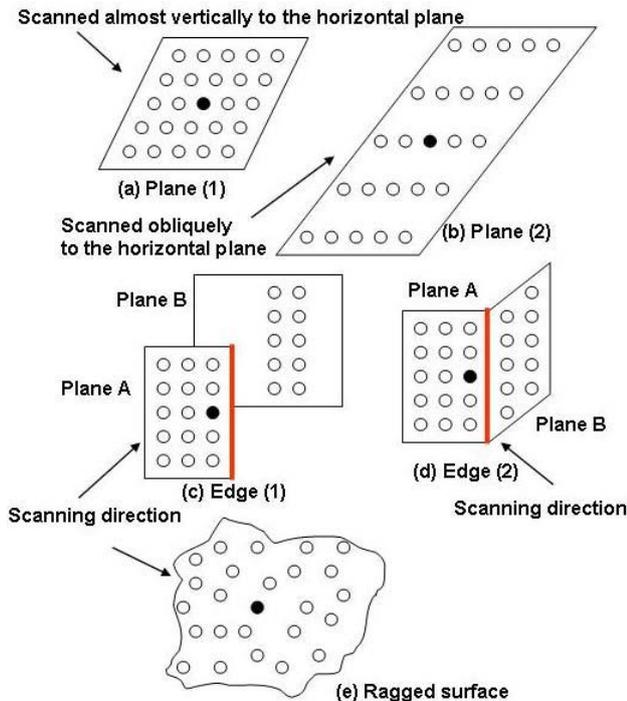


Figure 2. The Patterns of Local Plane
 The Black Dot is the Center of Local Plane
 The Arrow Shows the Direction of Scan

2.1 Normal Vector

Here we explain the calculation of the normal vector of the local plane, used in the segmentation and its pre-processing steps. We are calculating the local plane by the Least Square Method, using the 3Dimensional coordinates of the point $n \times n$ in the neighborhood of the interest point. In our experiment we are making "n" 5 or 7. We make this calculation with all the point clouds. The Figure.2 shows how the local plane looks like when we calculate 5×5 points. The central point which we use for the calculation of the local plane is shown here as a black Dot (interest point) and the neighbor points, as white points. The arrow mark indicates the direction of Scan to the plane. With both (a) and (b) we are scanning a plane. With (a), since the scan is done in the angle fairly close to the normal vector of the object plane, the distance between the points is very short, whereas with (b), since the scan is done in the angle oblique to the object plane the distance between the points is greater in depth. In both cases, however, since the points lie on a plane, the local plane can be calculated accurately. Now, (c) and (d) are the result of scan on the Edge region. The (c) shows the black Dot at the border of the plane A, which is closer to the scanner, and the plane B which is behind A. We call the part like this "Jump Edge". The (d) shows that the black Dot is at the border of the plane A and B, which are, unlike to (c), connected to each other with the Edge as their boundary. The (e) shows a ragged plane, which is neither planar nor curved surface.

2.2 Segmentation and its Pre-Processing Step

There are three different methods of segmentation of the point cloud. The Model Based Method put the point clouds on the existing model (Wang, 2003; Min, 2005). The Edge Based Method detect the Edge out of a range image and determine the region surrounded by the Edges as one Region (Zhang, 2001;

Bellon, 2002, Sappa, 2001). And the Region Based Method label all the points, which are similar in local plane, with the same label, judging from the normal vectors and Laser Intensity (Stamos, 2002; Yu, 2001; Chen, 2007; Konno, 2007).

Both Model Based and the Edge Based Methods are fitted for the segmentation of a big simple form object such as the façade. However, as we are interested in the segmentation of the detailed features, such as window frame, ceiling, column or pillar, we apply the Region Based Method which uses the local normal vectors. Now, Konno and his group (Konno, 2007), also applying Region Based Method, are extracting the feature lines in order to determine the proper starting point of labeling. But since this requires the feature lines to be clear and exact, they can work only on a planar. Besides, often times in the real survey site, the boundary of the planes cannot necessarily be extracted as an exact feature line. But this could easily undermine the accuracy of the labeling.

Again, Stamos and Allen (Stamos, 2002) have also challenge to clear. They try to determine the point, which is used for normal calculation, by the fixed threshold value of the distance from the interest point. But as the Figure 2 (b) indicates, when scan is made in the oblique angle to the local plane, the distance between points gets longer even on the same plane with the result that its difference from the jump edge(c) becomes inevitably blurred and the labeling result would not meet the necessary accuracy.

In our system, however, we have cleared all these challenges. As a pre-processing step for labeling we have invented a system by which the starting point should always appear on a planar or curved surface. We are judging whether the starting point is on the planar or curved surface by examining (1) the fitting accuracy (goodness of local plane fitting) at the interest point, (2) the curvature and (3) the distance to the neighbor points. By this examination we can determine the pertinent the starting point of labeling. And the initial labeling is given only to the point which is judged to be on such surface. And calculating the difference of the direction of the normal vector of interest point and that of its neighboring points, we put the same label (number) to them, if the difference is less than the threshold value.

Then next, we proceed to expand the labelling (Region Growing Procedure) to the points which have not yet been labeled. The judgment of applicability of this labeling is made by using the distance between the interest point and the local plane which is calculated from the labeled points neighboring to the interest point. And lastly, taking the two different planes, we calculate their intersecting line and determine it as the break-line.

3. RANGE IMAGE PRODUCTION

The measuring sphere of TLS is controlled by the azimuth angle "h" of the beam direction on the horizontal plane and by the elevation angle "v" on the vertical plane. And by scanning we can obtain for each point the 3 Dimensional coordinates (x, y, z) and Reflection Intensity as well as the color data of RGB. And as in the Figure.3, if we look at this projection also as the projection onto the plane with h and v as its axes, we have the 2 Dimensional image (h, v), which corresponds to the 3 Dimensional coordinates (x, y, z). We call this 2D image "Range Image". As a result, for each point we have 3 Dimensional coordinates (x, y, z), the Range Image coordinates (h, v), Reflection Intensity and RGB.

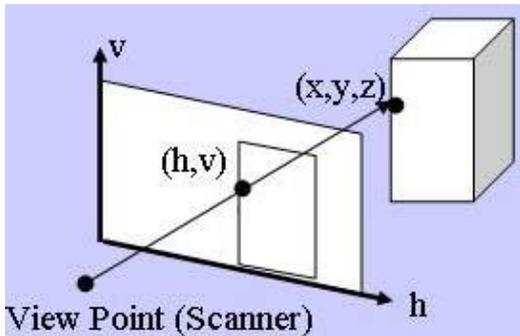


Figure 3. Coordinates of Range Image and 3 Dimensional Coordinates of an Object.



Figure 4. Top: Scanned Data (3D)
 Bottom: Range Image (2D)

With some TLS, which provides the h and v at an equal interval in the projecting direction, we can use the data obtained by the TLS directly as the range image data. However, with the data edited by the operator, it happens sometime that some of the coordinates data are dropped from the range image or are not given at an equal interval. In such cases we must reconstruct the range image. For this reason we always make this range image first in our algorithm. We make a regular grid with designated resolution on the projection plane and register the point closest to the mesh intersecting point. With this in hand, we can perform later the calculation of normal vectors and segmentation process like the calculation of 2 Dimensional image.

When we scan an object from the different angles, we must make a different range image each time. As to the resolution degree, usually we determine it by the sphere of the projecting direction and the number of the data points. But we can change the resolution if we so wish. At the top of Figure 4 you will find the 3 Dimensional picture of the original point cloud data and at the bottom, the picture of the range image created from these data.

4. PLANAR / CURVED SURFACE EXTRACTION

This is a pre-processing step for determining the pertinent starting point of labeling needed for the segmentation in the next chapter. The segmentation is the process to paste the same label (number) on the points belonging to the same planar or curved surface. The determining factor is the difference of the direction of the normal vectors between an interest point and its neighbor points. But if the interest point is on the ragged plane (e) as in the Figure 2, or on the Edge or near the Edge (c) (d), the normal vector calculation is liable to error. So, if we make labeling with such point as the starting point the result could be erroneous. We use the next three judging methods, therefore, to eliminate beforehand the points which do not belong to the planar / curved surface and after that we proceed to paste the label.

4.1 Judging by Goodness of Fitting

To determine the accuracy of Fitting we used the maximum distance from the local plane which was made Fit at the interest point to each of the neighbor points, which we used for Fitting. If the distance was longer than the threshold value d_{max} which had been set up beforehand, we judged that the fitting made at the interest point had not been correct. The purpose of this process is to eliminate the points on the ragged plane or the points on the Edge (Figure 2. (c) (d)). In our experiment we made the threshold value as $d_{max} = 0.02[m]$.

4.2 Judging by Curvature

First at the interest point, we obtain the standard deviation ($\sigma_{NVx}, \sigma_{NVy}, \sigma_{NVz}$) of the size of the normal vector of the neighbor $n \times n$ points in the direction of each axis. And using them, we calculate the Curvature C by the equations below. Since near the Edge, as in Figure 2. (d), the direction of normal vector is diversified, this value becomes bigger. Accordingly, the points in such region are eliminated. As to this calculation, first for the whole of the Range Image, we calculate the curvature of each point and obtain the maximum C_{max} and minimum C_{min} . And also we calculate the threshold value C_{th} by the equations below. As the Parameter we set γ within the range of 0.0 ~ 1.0. In our experiment we set $n = 5$ and $\gamma = 0.5$.

$$C = \sqrt{\sigma_{NVx}^2 + \sigma_{NVy}^2 + \sigma_{NVz}^2} \quad (1)$$

$$C_{th} = (C_{max} - C_{min}) \times \gamma + C_{min} \quad (2)$$

4.3 Judging by the Distance from the Surrounding

Lastly we make the judgment by using the distance between the interest point and the neighbor points. At the jump Edge as shown in the Figure 2.(c), the distance between the points changes greatly. So, we obtain the maximum distance and minimum distance and use their ratio (maximum value / minimum value) for judgment. In our case we made threshold value as $R_{th} = 2.0$. And if it was greater than this, we judged the point to be near the Jump Edge and eliminated accordingly.

5. SEGMENTATION

5.1 Initial Labeling

We make this operation with the points which were judged to be on the Planar / Curved surface in the previous chapter. As explained there, we compare the direction of the normal vector between the interest point and that of its neighbor points. And if the difference is less than the threshold value θ_{th} , we paste the same label (number). And we search for the point to be processed from the upper left side of the Range Image and make it the starting point. Since we use the range image, we can operate with the same algorithm as the processing of 2 D labeling. Here we have $\theta_{th} = 2.0[\text{deg}]$.

5.2 Region Growing Procedure

Now we work on the points which were not labeled by the initial labeling in order to expand the region. In the previous operation the points near the Edge were not labeled. But some are still on the neighboring plane. So, the purpose of this process is to redeem them by labeling. First we look at the labeled point and unlabeled point adjacent to it. Using the labeled point in the neighborhood ($k \times k$) of the interest point, we calculate the local plane with the point which has the same label. If we have more than one label, we calculate the local plane for each one of them. And then we calculate the distance between the interest point and the local plane of each point of the same label and select the closest label. But if the distance is bigger than the threshold value L_{Eth} we ignore the point.

Furthermore, if the change (θ_{Eth}) of the direction of the normal vector of the local plane before and after the Region Growing Procedure is within the threshold value, and if the ratio (R_{Eth}) of the distance between the interest point and neighboring points, and the average point interval of the neighbor region which we have used to calculate its local plane, is within the threshold value, we judge that the interest point belongs to the plane of neighboring labels and redeem it. Here we have $k = 7$, $L_{Eth} = 0.01[\text{m}]$, $\theta_{Eth} = 1.5[\text{deg}]$, $R_{Eth} = 2.0$.

All other points which have not been labeled are considered to be that of noise. But if we enlarge the value of θ_{th} for them, it may be possible to redeem them.

6. BREAK- LINE EXTRACTION

Picking up the two labeled which are next to each other using range images, we calculated out their intersection line as their break-line. The position of the both ends of the line were located by projecting the points in the close proximity of the intersecting line onto the intersecting line itself. But if two planes are parallel, or if they are separated in the 3D space, they can not have the intersecting line. Therefore, we developed the system to display at the same time the boundary line (the line

connecting the points on the contours). Since the boundary line surrounds the labels, it comes out always as an enclosed circuit.

7. RESULT

We applied our algorithm to three kinds of object: an indoor object (Figure 5.), an intersection of streets (Figure 6.) and a building (Figure 4. and 7.). The results of their Segmentation are shown with different color for each plane of the same label. For TLS we used Topcon GLS-1000. For PC we used CPU: Inter Core2 Duo P8700 2.53GHz, RAM: 2.8GB.

The Table 1. shows the number of the points of each data and the time required for processing. The Table 2. shows the original data size (txt file format) and the size after the break-line and boundary are output with the dxf file format.

Figure 5. and 6. show how the noises (black marks) of the pedestrians and cars passing between TLS and the buildings, are eliminated.

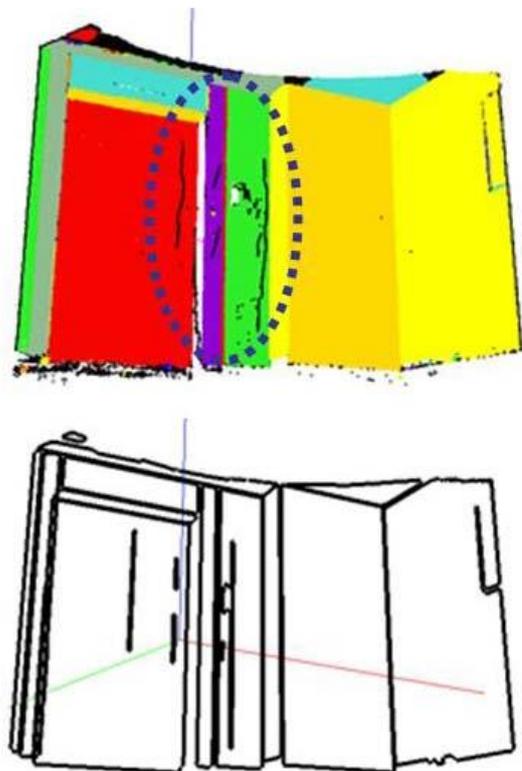


Figure 5. Indoor Sample

Top: Segmentation Result: Black shows noise. The area surrounded by dotted line is the noise of pedestrians
Bottom: Break-line and Boundary

Especially the Figure 6. of the intersection data shows that there are large amount of noises by the cars, which look like grasses growing from the ground. But our algorithm cleans all of them, making the segmentation of the roads possible (Figure 6. Bottom)

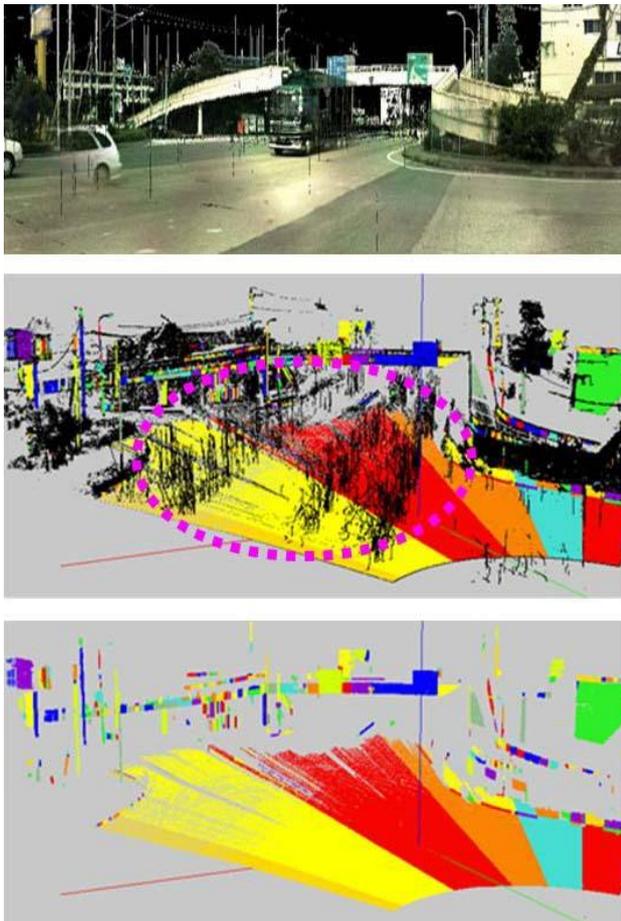


Figure 6. Intersection Sample
 Top: Range Image
 Middle: Segmentation Result. The area surrounded by dotted line is the area of the noises by passing cars
 Bottom: The top Sample after the elimination of noises

Sample Name	Point Number (,000)	Processing Time
Indoor	181	9sec
Intersection	1276	1 min. 34 sec
Building	10673	38 min. 12 sec

Table 1. Number of Points and Processing Time

Sample Name	Input Size (txt)	Output Size (dxf)
Indoor	4	1
Intersection	63	7
Building	378	66

Table 2. Input & Output Size [MB]

	x	y
RMS	25.9	26.2
MAX	51.5	59.8

Table 3. Comparison of the PC output Data and CAD Plan
 RMS and MAX [mm] of the Deviance on the plan

Table 1 shows each Sample with its number of Points and Processing Time. The Building Sample took long time, because it had over ten million Points. But you can shorten the time by relaxing the resolution density when we produce the range image, as explained in the Chapter 2. Though accuracy diminishes, it would still be clear enough to grasp the outlook. Table 2. shows the data size of input and output. We can see that the size can be compressed down to 1/4 ~ 1/9.

As the data of Building have CAD plan (Figure 7. Bottom), we measured the deviance between the CAD plan, and the boundary and break-lines produced by our algorithm (Figure 7. Top). We selected 32 Points at the corner of a window frame both from the CAD plan and from the PC output data and calculated the deviance by our Topcon software “Image Master (Old Name is PI-3000 and We have also developed Human Body Measurement using this technology)” for 3 D measurement (Kochi, 2003; 2009; Kitamura, 2009). Table 3. shows its result.

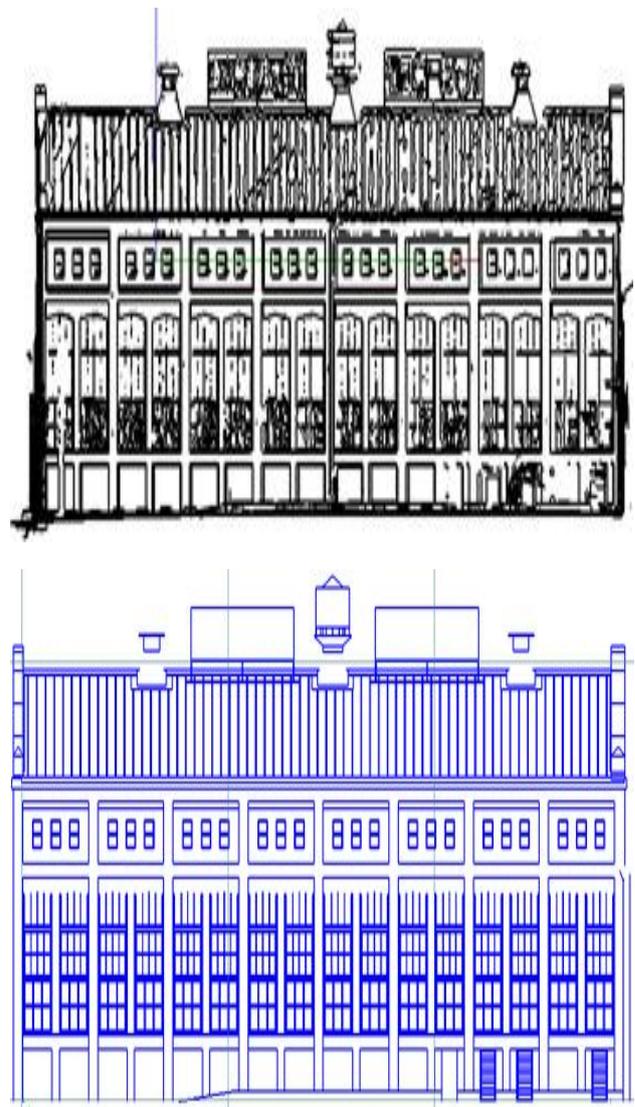


Figure 7. Top: Boundary and Break-lines produced by our algorithm
 Bottom: CAD Plan

8. RECAPITULATION

We created an algorithm to segment the plane and to extract the break-line, while eliminating automatically the noises of the pedestrians, cars and plants which come in between the TLS and the objects while scanning. We can see the elimination of noises in the samples of an indoor (Figure 5.) and intersection (Figure 6.). In the past, especially at the location where we had so much noises like at an intersection, we had to spend several hours with manual work. With our new algorithm, however, we can do that only in 1.5 minutes, thus economizing enormous amount of time.

As to the deviance in the data of Building, we found it to be 25~26mm in RMS between the CAD data and data obtained by our algorithm. This means, with our algorithm, at its present stage of development, we can already obtain the result of draft level accuracy from the Point Cloud Data.

Besides, by transposing the Point Cloud Data to the break-line or boundary line, the data size can be compressed down to 1/4~1/9. This enables us to recognize the exterior features and shape of an object, even without high performance PC, which is very helpful to understand the form on the real site. The processing time of a Building took us nearly 40 minutes, but if we tame the intensity of resolution in making range image, we can reduce the time considerably.

Our ultimate goal is the automatized production of the perfect plan out of the Point Clouds. But as we cannot yet totally eliminate the measuring or calculation errors, our algorithm is not self sufficient, at this point, for finalizing the total process. Nevertheless, it is practical enough for making pre-processing steps, such, for example, as to assist the understanding of the object form, or making draft image. And it has proved how efficiently it can simplify and expedite such process.

Our next step is to develop the algorithm which can automatically integrate the operations of more than one TLS scan, unifying all the Point Cloud data thus gathered from different angles (stand points).

REFERENCES

Bellon, O. R. P., and Silva, L., 2002. New Improvements to Range Image Segmentation by Edge Detection. *IEEE Signal Processing Letters*.

Chen, C., and Stamos, I., 2007. Range Image Segmentation for Modeling and Object Detection in Urban Scene. *The 6th International Conference on 3-D Digital Imaging and Modeling*, Montreal, Canada, August 21-23,

Jiang, X. and Bunke, H., 1999. Edge Detection in Range Images Based on Scan Line Approximation. *Computer Vision and Image Understanding* Vol.73, No.2, , pp.183-199,

Kitamura , K., Kochi , N., Watanabe ,H., Yamada ,M., Kaneko ,S., 2009, Human Body Measurement by Robust Stereo-matching, *9th Conference on Optical 3-D Measurement Techniques*, July 1-4, Viena University of Technology, Vol.2 pp254-263

Kochi,N., Ito,T,Noma,T., Otani,H., Nishimura,S., Ito,J., 2003, PC-based 3D Image Measuring Station with digital camera an example of its 24 actual application on a historical ruin, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXIV, Part 5/W12, pp. 195-199.

Kochi, N., 2009, Photogrammetry, *Handbook of Optical Metrology : Principles and Applications*, Yoshizawa,T.'(Ed.), Taylor and Francis, NewYork, Chapter 22.

Konno, T., Konno, K., and Chiba, N., 2007. Ridge Lines Extraction by Hierarchical Planer Segmentation of Measured Point Clouds. *Nicograph*, Vol6, No4, pp.197-206,

Min, J., and Bowyer, K., 2005. Improved range image segmentation by analyzing surface fit patterns. *Computer Vision and Image Understanding*, Vol.97, Issue.2, pp.242-258,

Mitra, N., and Nguyen, A.,. 2003. Estimating Surface Normals in Noisy Point Cloud Data. *Proc. Of the 9th Annual Symposium on Computational Geometry*,

Sappa, A. D. and Devy., M. 2001. Fast Segmentation by an Edge Detection Strategy. *In Proc. 3rd International Conference on 3-D Digital Imaging and Modeling*, Quebec, Canada, 2001. pp292-299

Stamos, I., and Allen, P. K., 2002. Geometry and Texture Recovery of Scenes of Large Scale. *Journal of Computer Vision and Image Understanding* ..

Wang, H., and Suter, D., 2003. A model-based range image segmentation algorithm using a novel robust estimator. *In Proc, 3rd International Workshop on SCTV*,

Yu, Y., Frensz, A., and Malik, J., 2001. Extracting objects from range and radiance images. *IEEE Trans. Visualization and Computer Graphics*, Vol.7, No.4, pp.351-364,

Zhang, Y., Sun,. Y., Sari-Sarraf, H., and Abidi, .M., 2001. Impact of Intensity Edge Map on Segmentation of Noisy Range Images. *In Proc. SPIE Conf. On Three-Dimensional Image Capture and Applications III*, Vol.3958m, pp.260-269, Quebec, Canada.