Computer Science Depatrment of The University of Auckland
CITR at Tamaki Campus (http://www.tcs.auckland.ac.nz)

CITR-TR-31 October 1998

Surface Visualisation and 3D Feature Analysis in Confocal Microscopy Images

Fu Cheong Fong [1], Reinhard Klette [1], Anthony Poole [2], and Man Hon Wong [1]

Abstract

Chondrons form the fundamental biomechanical and metabolic unit of articular cartilage. But there is no accurate description of chondron volume which is known to change when articular cartilage is functionally loaded. This report discusses the 3D feature analysis of the confocal microscopy images of chondrons. The basic principle of confocal microscopy is described. The segmentation problem of confocal images is also discussed. Two different reconstruction algorithms, Marching Cubes and Dividing Cubes are also illustrated and their respectively ways of calculating volume and surface area are also discussed. The results of calculating 3D features of sperical polysterene beads using Marching Cubes algorithms are also given.

[1] The University of Auckland, Computer Science Department, CITR, Tamaki Campus (Building 731), Glen Innes, Auckland, New Zealand
[2] Department of Anatomy, Faculty of Medicine and Health Science, University of Auckland, Private Bag 92019, Auckland, New Zealand

Surface Visualisation and 3D Feature Analysis in Confocal Microscopy Images

Fu Cheong Fong [1], **Reinhard Klette** [1], **Anthony Poole** [2], **and Man Hon Wong** [1]

[1] CITR, University of Auckland
Tamaki Campus, Building 731, Auckland, New Zealand

[2] Department of Anatomy
Faculty of Medicine and Health Science, University of Auckland
Private Bag 92019, Auckland, New Zealand

Abstract: Chondrons form the fundamental biomechanical and metabolic unit of articular cartilage. But there is no accurate description of chondron volume which is known to change when articular cartilage is functionally loaded. This report discusses the 3D feature analysis of the confocal microscopy images of chondrons. The basic principle of confocal microscopy is described. The segmentation problem of confocal images is also discussed. Two different reconstruction algorithms, Marching Cubes and Dividing Cubes are also illustrated and their respectively ways of calculating volume and surface area are also discussed. The results of calculating 3D features of spherical polystyrene beads using Marching Cubes algorithm are also given.

Keywords: Confocal microscopy, chondrons, segmentation, Marching Cubes, Dividing Cubes, volume and surface area calculations in Marching Cubes and Dividing Cubes, Minimum Jordon Surfaces.

1 Introduction

In recent years, there are many 3D visualization and analysis techniques [Klette et al., 1998] applied to the field of microscopy . These techniques have been successfully applied in light and electron microscopy. But the advent of confocal microscopy has led to the rapid growth of 3D visualization of microscopic structures.

We examine the 3D structure of biological images produced by confocal microscopy. For this study, we use confocal images of cartilage chondrons.

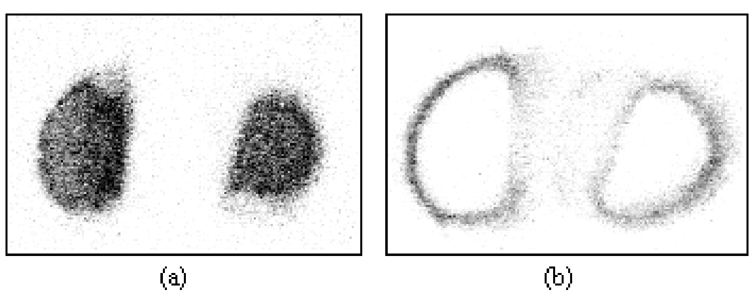

Figure 1: (a) Two live chondrocytes. (b) Pericellular microenvironment.

Chondrons form the fundamental biomechanical and metabolic unit of articular cartilage. They consist of two parts: the chondrocytes (see Fig. 1(a)), the cells of cartilage, which are surrounded by a specialized structure called the pericellular microenvironment (see Fig. 1(b)) [Poole, 1997].

A considerable number of confocal studies have now established the molecular anatomy of the chondron and it is now possible to accurately define both the chondrocyte and the pericellular microenvironment by confocal microscopy [Poole, 1997].

We do not however have an accurate description of chondron volume which is known to change when articular cartilage is functionally loaded [Poole, 1997].

We therefore aimed to calculate, using reconstructed confocal images, the values of the chondrocyte, the pericellular microenvironment and the total chondron volume. The unique new data resulting from this study will be used to model the hypothesis [Poole, 1997] that the chondron functions biomechanically to protect the vital role of the chondrocyte in the maintenance of normal cartilage. Failure of the chondron in its biomedical role will ultimately lead to osteoarthritis and immobility in old age.

Volume investigation requires four steps:

1. data acquisition,
2. segmentation,
3. three-dimensional visualization,
4. analysis of 3D features.

In this report, we discuss the basic principle of confocal microscopy and how the images are obtained. The problems encountered in segmentation will also be mentioned. Then, we illustrate the use of two different 3D reconstruction algorithms and how to calculate the volume and surface area of the chondron using these two algorithms.

2 Data Acquisition

All data images in this project are collected from confocal microscopy. The basic optical principle used in confocal microscopy is illustrated in Fig. 2.

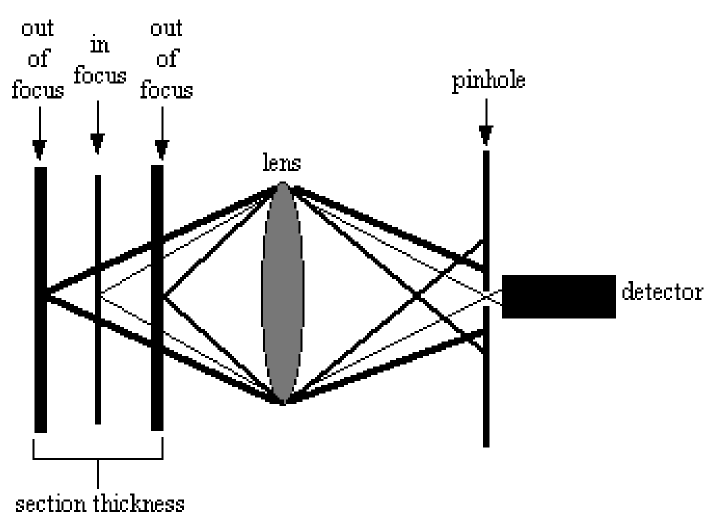

Figure 2: Optical principle of confocal microscopy.

As seen from the figure, the lens focuses all the light from the in-focus plane to the detector while the pinhole cuts away the light that comes from the out-of-focus planes. The whole mechanism is controlled by the computer and the focused beam is scanned in a raster fashion through a specimen. The signals received by the detector are used to generate a digital image of the specimen. In this mechanism, laser light is used instead of conventional light due to its high intensity, high degree of monochromaticity and extremely long coherent properties.

To get the image of the different structures of the same specimen, different dyes are applied to the different structures. Then a particular excitation/emission wavelength is used to pick up a particular dye each time and the signal received is used to generate the digital image of that structure. In this study, the dye CMFDA was used to image the chondrocyte while type VI collagen stained with Texas Red was used to image the pericellular microenvironment.

All data collected for this report are 8-bit images with a resolution of 512 x 512. For each scan of a specimen, a maximum of 70 slices can be obtained. But since the whole mechanism is controlled by the computer, it is possible to obtain 140 slices of the specimen by scanning it twice with a different displacement adjustment. In our project, two image sets are obtained for each chondron - chondrocyte image and capsule image. Chondrocyte image is used to allocate the interior while capsule image is used to allocate the exterior contour.

To ensure accurate analysis of the chondron volume, we prepare identical data set of spherical polystyrene beads (see Fig. 3) specifically manufactured to a diameter of 15μm $\pm$ 3% (Molecular Probes USA). These beads contain alternate layers of red and green fluorodromes which can be imaged separately, and provide a convenient, inert, non-biological specimen for comparative purposes. These image sets are used for verification of calculation method used in different 3D reconstruction algorithm since there is no information about the surface area and volume of a chondron.

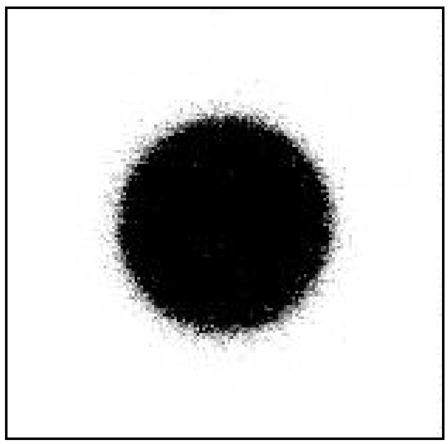

Figure 3: Confocal microscopy image of spherical polystyrene bead.

3 Segmentation

After the images have been collected, the next step is to define or locate the interested image regions. This is the most important but also difficult step in volume investigation. As seen in Fig. 1, the boundary of the chondrocytes and the pericellular microenvironment are very difficult to define or locate. Sometimes, preprocessing of the images is necessary in order to simplify the segmentation process.

We applied some local operators to enhance the images during the preprocessing, mean(average), median, maximum(dilation), minimum(erosion), opening and closing [Klette and Zamperoni, 1996]. These are basic operators in image processing. Results of applying these operators are illustrated in Fig. 4. The median operator seem to lead to a better image enhancement than the other local operators. After a median operator has been applied to the image, we can possibly locate the interested region. Since the chondron has a likely ellipse shape, an approximate way of describing the shape of chondron is ellipse fitting [Russ, 1995]. Ellipse fitting has an advantage of calculating the features of the chondron such as area and perimeter.

Since we are dealing with a 3D object, a local operator on a 2D image of a single slice may not be appropriated. A possible approach is to combine several images of slices to specify the input for a 3D

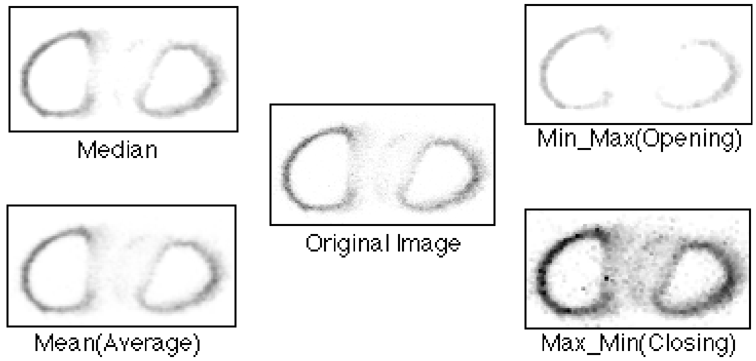

Figure 4: Results of four image operators.

filter to support the object segmentation. This approach will require a huge set of data,for example, one output image on same location will require 4 to 8 raw image. Another approach is combining fitting ellipse result on the volumetric data. The main idea is slicing the volumetric data into series of images according to x-axis, y-axis and z-axis, then apply fitting ellipse algorithm on them. In this case, we collect three series of ellipses with respect to different axis (Fig. 5), then apply each series of ellipses on a higher resolution of volumetric space, and then we can use the idea of maximum or minimum operators to combine these data into a new volume data, which is called cumulative space. A voxel or 3D data point of cumulative space is classified as inside the object by a minimum operator if and only if this point is included in either one of three series of fitted ellipses. The data point is included in the cumulative space by a maximum operator if and only if this point is included in all three series of fitted ellipses.

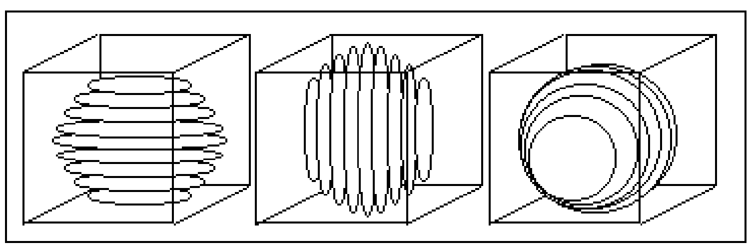

Figure 5: Ellipse fitting with respect to different axes.

The idea of a cumulative space can co-operate with different segmentation methods, e.g. by using cumulative space for interpolating curves or surfaces such as spline and b-splines [Foley et al., 1992] on the original volume data. The first reason for choosing spline and b-spline is that the result of fitting ellipse is not accurate for calculating 3D feature. Second, spline curve will maintain the C^1 continuity (end point) and C^2 continuity (first derivation). Using spline curves allows a higher resolution and better accuracy for 3D feature calculations. The reason of using cumulative space to recreate the volumetric data is that we can improve the original data into a higher resolution and reveal the missing details during the data collection.

4 Three-Dimensional Visualization

After the images have been preprocessed and/or segmented, a 3D model of these images has to be reconstructed. There are many different volume visualization algorithms that can be used to generate a 3D model of images sequence. Among these algorithms, Marching Cubes is the most common and widely used approach in volume reconstruction for images sequences.

4.1 Marching Cubes Algorithm

The first Marching Cubes algorithm was invented by W. E. Lorensen and H. E. Cline for isosurface extraction [Lorensen and Cline, 1987]. In this algorithm, two successive images are used to form a plane of cubes where the corners of cubes are the pixels of the images. The corners of cubes are called voxels.

The voxels of each cube are then classified according to the user defined threshold. The eight voxels of the cube are numbered 1 through 8 as in Fig. 6(a). If the voxel's value is greater than or equal to the threshold, the voxel is classified as 'in' and labeled '1'. Otherwise, it is classified as 'out' and labeled '0'. The eight values of the voxel are then put in eight consecutive bit locations to form an 8-bit code. This code is treated as binary and its decimal value becomes the index of the Lookup Table. The Lookup Table then returns a sequence of edges' number according to the triangular surface formed within the cube. In this algorithm, each cube is considered row by row, plane by plane. As the location of the considered cube marches through the data set, W. E. Lorensen and H. E. Cline called this a marching cubes algorithm.

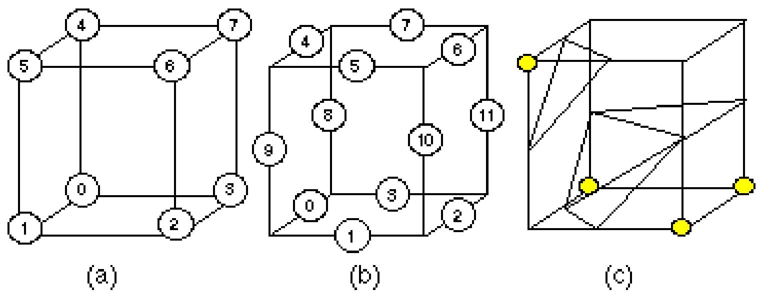

Figure 6: (a) A cube with numbered voxels. (b) A cube with numbered edges. (c) An illustrated example.

The cube in Fig. 6(c) have four voxels classified as 'in'. According to the numbering in Fig. 6(a), a bit code 00101101 is formed. Its decimal value (i.e. 45) is used as the index of lookup table. If the edges of the cube are numbered as in Fig. 6(b), the lookup table should return { 11, 10, 8, 8, 10, 0, 10, 1, 0, 5, 4, 9 }. From this, we can know that four triangular surfaces are generated and which edges are used to form each surface.

To find out where the edge is intersected by the isosurface, linear interpolation is applied. If a voxel have a value greater than the threshold while the other have a value less than the threshold, there should be a point between these two voxels having a value equal to the threshold. The location of this point is calculated using simple linear ratio.

Since each voxel can be labeled either '1' or '0', there totally $2^8 = 256$ different ways to label the voxels. But these 256 cases can be reduced to 14 by reflection, rotation and complementary symmetry. These 14 cases are shown in Fig. 7.

But these 14 cases are incomplete in the sense that they are generating surfaces which occasionally may have holes. By adding eight more cases, the continuity of the surface can be guaranteed. (see Fig. 8).

However, since it is now more than one way to define surfaces for a particular cube, it is difficult to decide which way should be used. To reduce the ambiguous cases, each cube can be broken up into five tetrahedra and each tetrahedron is checked with a different lookup table to get the intersected edges [Shirley and Tuckman, 1990]. This approach is called marching tetrahedra algorithm. Another way to reduce ambiguous cases is also described in [Nielson and Hamann, 1991].

4.2 Dividing Cubes Algorithm

The basic idea of this algorithm is similar to a hierarchical data structure called octree [Samet, 1993]. The voxels of a cube are classified as in the Marching Cubes algorithm. If all the voxels having a value greater than or equal to the threshold, the cube is said to be inside the object. If any voxel has a value

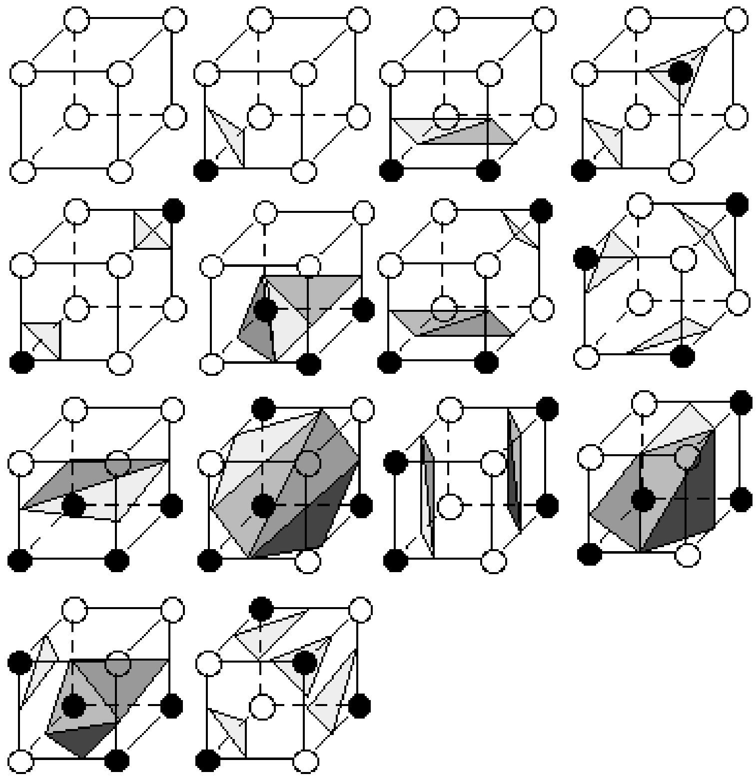

Figure 7: 14 major cases.

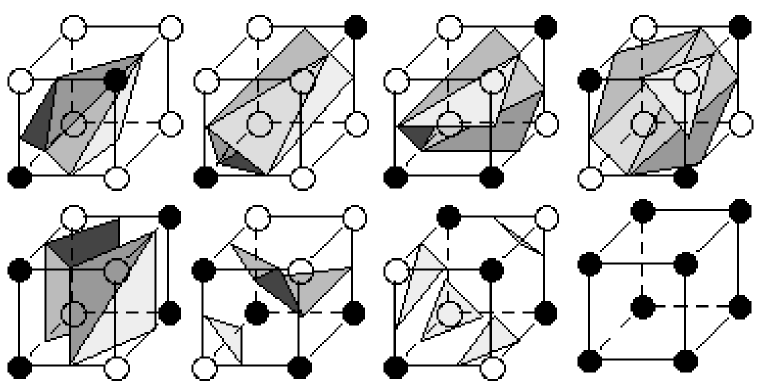

Figure 8: Eight more cases to improve the algorithm.

less than the threshold, the cube is subdivided into smaller cubes, and the smaller cubes are classified again. This process continues until the cube is subdivided into its minimum size. The minimum size is determined by the accuracy wanted. The smaller the minimum size used, the higher the accuracy will be. However, the processing time will increase exponentially when the minimum size used decreases.

This algorithm is easier to implement than the Marching Cubes algorithm since the Marching Cubes algorithm has 22 major cases to be considered while this algorithm has only one recursion rule to be handled.

Although this algorithm is easy to implement, there will be a problem in 3D feature analysis which will be discussed in the next section.

5 Analysis of 3D Features

The 3D features to be analysed are volume and surface area of the object. To make it more efficient, these features are analysed during reconstruction process. This makes the way to analyze 3D features depends on the reconstruction method used.

5.1 Marching Cubes Based Measurements

The Marching Cubes algorithm generates many triangular isosurfaces. The surface area of the object is just the sum of the area of these triangular isosurfaces. So we only need a formula to calculate the area of an arbitrary triangle.

Assume the length of the sides of an arbitrary triangle are a, b and c. The surface area of the triangle is equal to

$$\sqrt{p(p-a)(p-b)(p-c)}$$

where p = $\frac{1}{2}$ (a + b + c).

To find the length of the sides of a triangle, we can use the vector formula as the vertices of the triangle are in 3D space.

Let the vertices of one side of the triangle are (x_1, y_1, z_1) and (x_2, y_2, z_2). The length of the side is equal to

$$\sqrt{(x_2-x_1)^2+(y_2-y_1)^2+(z_2-z_1)^2}.$$

To calculate the volume of the object, we have to calculate the volume enclosed by the isosurface within each cube. The sum of these volumes gives the volume of the object. Since a tetrahedron is the basic structure of all regular volumetric shapes, a formula to calculate its volume is required.

Assume a tetrahedron formed by vertices (x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3) and the origin. The volume of the tetrahedron is equal to

$$\frac{1}{6}\begin{vmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix}.$$

This formula is used to calculate the volume of a tetrahedron formed by three vertices and the origin. If the tetrahedron is formed by four vertices with no origin, we have to transform the tetrahedron before applying this formula.

To calculate the volume of the object, we have to subdivide the volume enclosed by the isosurface into tetrahedra. Each tetrahedron is then transformed so that one of its vertices is at the origin before the formula is applied. The total volume of these tetrahedra gives the volume of the object.

This formula gives a signed volume of the region spanned by these three vertices and the origin. So we have to take the absolute value of the result of the formula before adding to the total volume.

5.2 Dividing Cubes Based Measurements

In Marching Cubes algorithm, surface area calculation is much easier than volume calculation. But in Dividing Cubes algorithm, the volume calculation is much easier.

To calculate the volume of the object, we just simply calculate all the volumes of the cubes with all voxels having the value greater than or equal to the threshold. The smaller the minimum size of the cube can be subdivided, the higher the accuracy of the object's volume will be.

To calculate the surface area of the object, we have to calculate all the surface area of the faces of cubes which are part of the object's surface. So even a cube is classified as an 'inside' cube, it may have no contribution to the surface area of the object. In other words, we have to consider if the cube is on the object's surface and which faces of the cube contribute to the object's surface.

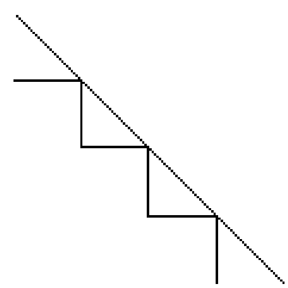

Figure 9: Illustrating the problem in Dividing Cubes algorithm.

A problem in this algorithm is illustrated in Fig. 9. In this figure, the length of the zigzag line is used to estimate the length of the straight line. However, the error is just the same whatever the size of the zigzag line is. Similarly in 3D cases, whatever the minimum size of the cubes is, the error in surface area calculation will be the same. However, a change in orientation of the object may cause a difference in the accuracy of surface area calculation. It is because this change may cause an alignment between the faces of the cubes and the object's surface. However, this depends on what the shape of the object is. But there is no such problem in volume calculation.

5.3 Measurement of spherical polystyrene beads using Marching Cubes Algorithm

For verifying the program, two sets of confocal microscopy images of spherical polystyrene beads are collected. Each set of data have 70 images with a resolution of 512 pixels x 512 pixels. The actual size of the data set is 99.645μm x 99.645μm x 18.354μm. The results are shown in the following tables.

The spherical polystyrene bead used for the images has a diameter of 15μm. So its volume is 1767.15μm^3 and its surface area is 706.86μm^2.

In the program, we can use either the exact intersection point or the middle point of the edge for calculating 3D features. The difference in 3D features using different points is shown in the tables. Besides this, as mentioned before, in some cases of lookup table, there are more than one way to define surfaces within a cube. Since one way of defining surfaces is always enclosing a larger volume than the other, the program can be set to use either larger volume configuration or smaller volume configuraton. The difference is also shown in the tables.

One data set is used to get the results in Table 1, 2 and 3 while the other data set is used in Table 4 and 5. The data set used in Table 1, 2 and 3 have a higher voltage of lighting during acquisition than the other. So the pixels of the images in these data set have higher density values. Comparing these tables, it is found that the threshold value is very critical. Different threshold is needed for different data set. Even for the same data set, different threshold can result in different accuracy of the volume and surface area calculated. So segemntation plays an important role in 3D analysis.

For each table, it is found that there is only a slightly difference between using larger volume configuration and using smaller volume configuration. Also, Using the middle point of the edge always results in larger volume than using the exact intersection point.

When comparing Table 2 and Table 3, it is found that the volumes have only a slight change while the surface area have a relatively larger increased. This is due to the fact that the median operator have a 'smoothing' effect on the boundary of the regions in the images.

		using exact intersection point	using middle point
using larger vol. config.	volume in μm^3(error in %)	1780.83(0.77)	1796.21(1.64)
	surface area in μm^2(error in %)	755.68(6.90)	822.64(16.38)
using smaller vol. config.	volume in μm^3(error in %)	1780.87(0.78)	1796.29(1.64)
	surface area in μm^2(error in %)	755.61(6.90)	822.55(16.37)

Table 1: Threshold = 240

		using exact intersection point	using middle point
using larger vol. config.	volume in μm^3(error in %)	1747.71(-1.10)	1762.38(-0.27)
	surface area in μm^2(error in %)	752.31(6.43)	817.91(15.71)
using smaller vol. config.	volume in μm^3(error in %)	1745.76(-1.21)	1762.49(-0.26)
	surface area in μm^2(error in %)	752.22(6.42)	817.91(15.71)

Table 2: Threshold = 245

		using exact intersection point	using middle point
using larger vol. config.	volume in μm^3(error in %)	1742.96(-1.37)	1760.06(-0.40)
	surface area in μm^2(error in %)	907.55(28.39)	1022.37(44.64)
using smaller vol. config.	volume in μm^3(error in %)	1748.33(-1.06)	1767.34(0.01)
	surface area in μm^2(error in %)	893.69(26.43)	1002.01(41.76)

Table 3: Threshold = 245, with no median operator

		using exact intersection point	using middle point
using larger vol. config.	volume in μm^3(error in %)	1961.29(10.99)	1987.15(12.45)
	surface area in μm^2(error in %)	1034.25(46.32)	1073.45(51.86)
using smaller vol. config.	volume in μm^3(error in %)	1963.14(11.09)	1989.28(12.57)
	surface area in μm^2(error in %)	1033.30(46.18)	1073.89(51.92)

Table 4: Threshold = 150

		using exact intersection point	using middle point
using larger vol. config.	volume in μm^3(error in %)	1930.91(9.27)	1977.50(11.90)
	surface area in μm^2(error in %)	1112.10(57.33)	1104.07(56.19)
using smaller vol. config.	volume in μm^3(error in %)	1933.26(9.40)	1980.01(12.05)
	surface area in μm^2(error in %)	1117.46(58.09)	1102.59(55.98)

Table 5: Threshold = 157

When comparing all tables, it is found that the errors in surface area are much larger than that in volume. So it is quite obvious that Marching Cubes algorithm is good for volume calculation but not for the surface area calculation. In the other words, an algorithm for calculating surface area is needed.

6 Conclusion

Segmentation of the confocal images of chondron is the most difficult steps in 3D feature analysis. It is difficult to define the boundary of chondrocyte and its pericellular microenvironment which in turn causes the difficulty in reconstructing an accurate chondron's model. This also affects the accuracy of the measurement of 3D features of the object.

The two reconstruction algorithms used in 3D visualization have their own advantages and disadvantages. Marching Cubes algorithm is difficult to implement but easy to calculate the surface area of the object. Ambiguous cases in this algorithm cause the problem in both reconstruction and calculation. It is found that this algorithm is good for volume but not surface area calculation. Dividing Cubes algorithm is much easier to implement and to calculate the volume of the object. However, the surface area calculated by this algorithm is just an approximation and the accuracy may vary with the orientation of the object. To calculate the surface area, the algorithm for minimum Jordon Surfaces [Klette et al., 1998] can be used.

A more complete 3D features analysis(including segmentation, 3D reconstruction and volume and surface area calculation using both algorithms) will be given in the forthcoming report.

References

[Cheng, 1997] S. L. Cheng: Estimation of Volume and Surface Area from Isosurface. Postgraduate Project in Computer Science, 415.780 FC, CITR Tamaki, University of Auckland (1997).

[Foley et al., 1992] J. Foley, A. van Dam, S. Feiner, J. Hughes: *Computer Graphics*, Addison Wesley, USA., (1002).

[Klette et al., 1998] R. Klette, F. Wu, S. Z. Zhou: Multigrid Convergence of Surface Approximation. Postgraduate Project in Computer Science, CITR-TR-25 Tamaki, University of Auckland (1998).

[Klette and Zamperoni, 1996] R. Klette, P. Zamperoni: *Handbook of Image Processing Operators*, Wiley, Chichester, (1996).

[Lorensen and Cline, 1987] W. E. Lorensen, H. E. Cline: Marching Cubes: A High Resolution 3D Surface Construction Algorithm. *ACM Computer Graphics* **21** (1987) 163–169.

[Nielson and Hamann, 1991] G. M. Nielson, B. Hamann: The Asymptotic Decider: Resolving the Ambiguity in Marching Cubes. *Proceedings of the IEEE Visualization '91 Conference* (1991) 83–91.

[Poole, 1997] A. Poole: Articular cartilage chondrons: form, function and failure. *J. Anat.* **191** (1997) 1–13.

[Russ, 1995] John C. Russ: *The Image Processing Handbook* , CRC Press, Florida, USA., (1995).

[Samet, 1993] H. Samet: *Applications of Spatial Data Structures*, Addison Wesley, USA., (1993).

[Shirley and Tuckman, 1990] P. Shirley, A. Tuckman: A Polygonal Approximation to Direct Scalar Volume Rendering. *Computer Graphics* **24** (1990) 63–70.