EUROGRAPHICS '94 / M. Daehlen and L. Kjelldahl (Guest Editors), Blackwell Publishers

Volume 13, (I994), number 3

A Method for Displaying Metaballs by using Bézier Clipping

Tomoyuki Nishita
Fukuyama University
Higashimura-cho, Fukuyama, 729-02 Japan

Eihachiro Nakamae
Hiroshima Prefectural University
Nanatsuka-cho, Shoubara City, 727 Japan

Abstract

For rendering curved surfaces, one of the most popular techniques is metaballs, an implicit model based on isosurfaces of potential fields. This technique is suitable for deformable objects and CSG model.

For rendering metaballs, intersection tests between rays and isosurfaces are required. By defining the higher degree of functions for the field functions, richer capability can be expected, i.e., the smoother surfaces. However, one of the problems is that the intersection between the ray and isosurfaces can not be solved analytically for such a high degree function. Even though the field function is expressed by degree six polynomial in this paper (that means the degree six equation should be solved for the intersection test), in our algorithm, expressing the field function on the ray by Bézier functions and employing *Bézier Clipping,* the root of this function can be solved very effectively and precisely.

This paper also discusses a deformed distribution function such as ellipsoids and a method displaying transparent objects such as clouds.

Keywords : Metaballs, Blobs, Soft objects, Density function, Ray tracing, Bézier Clipping, Deformable objects, Geometric Modeling, Photo-realism

1 Introduction

The representation of free-form surfaces can be classified into two: parametric surfaces and implicit surfaces. For the former, Bézier patches, B-spline patches, and NURBS are used. For the latter, a set of density functions such as metaballs[4] (blobs[1] or soft objects[11]) is often used; in this method, a curved surface is defined by an isosurface which is a set of points having the equi-potential field value. The field value at any point is defined by distances from the specified points in space. The features of the metaballs are as follows: (1) the required data for metaballs is typically at least two to three orders of magnitude smaller than that modeled with polygons[2], (2) it is suitable for use in the CSG model, *(3)* it is suitable for the representation of deformable objects, so it is useful for animations. (4) it is well suited for modeling of human bodies, animals, organic models, and liquids.

Because of such a usefulness, many commercial software packages implement such modeling techniques [2]. The metaball technique has become a most indispensable technique in 3-D graphics software these days, but its improvement in its modeling capability and calculation time are still desired.

The main task for rendering metaballs is intersection tests between rays and isosurfaces. When the field function is defined by a high degree of function, the modeling capability becomes rich; we can expect the surface to become much smoother. However, since the intersection between a ray and isosurfaces can not be solved analytically for such a high degree function, a robust method with high precision is needed. In this paper, the field function is expressed by degree six polynomial, so the degree six of equation should be solved for the intersection test. In the proposed algorithm, the field function on the ray is expressed by Bézier functions, so the root of this function is effectively and precisely solved by using *Bézier Clipping* [6] which uses the convex hull property of Bézier curves.

It is well known that if we use ellipsoid metaballs instead of spheres, the number of elements to represent free-form surfaces can be decreased in some cases. The proposed method is also applicable to ellipsoids. The metaball technique is also useful for displaying transparent objects such as clouds. We demonstrate the usefulness of the proposed method by using various examples.

2 Previous Methods for Field Functions

In the metaball technique, a free-form surface is defined as an isosurface(equi-potential surface) of field function; The field value at any point is defined by distances from the specified points in space. The task of the user is to specify the center position of each metaball, its density at the center, field function, and color. This modeling technique was first developed by Blinn[1], and he called it *blobs* (or blobs molecules). In Japan, Nishimura et al.[4] independently developed it, and they called it *metaballs.* Recently Wyvill et al. [9][10][11] have also developed a display method for field functions, and he called it *soft objects.* The main differences in these previous work are in the shapes of field functions and the methods solving for ray/isosurface intersections. For the field functions, the following four functions have been developed: exponential[1], piecewise quadratic, degree four polynomial, degree six polynomial[11]. That is, the field function f_i for metaball i is expressed by the following:

1) exponential function (Blinn[1] 1982)

$$f_i(r) = exp(-ar^2). \tag{1}$$

2) piecewise quadratic (Nishimura[4] 1983)

$$f_i(r) = \begin{cases} 1 - 3(\frac{r}{R_i})^2 & (0 \le r \le \frac{R_i}{3}) \\ \frac{3}{2}(1 - (\frac{r}{R_i}))^2 & (\frac{R_i}{3} \le r \le R_i) \end{cases}. \tag{2}$$

3) degree four polynomial (Murakami [3] 1987)

$$f_i(r) = (1 - (\frac{r}{R_i})^2)^2. \tag{3}$$

4) degree six polynomial (Wyvill [9] 1986)

$$f_i(r) = -\frac{4}{9}(\frac{r}{R_i})^6 + \frac{17}{9}(\frac{r}{R_i})^4 - \frac{22}{9}(\frac{r}{R_i})^2 + 1. \tag{4}$$

where R_i is the effective radius of metaball i, r the distance between a point from the center $P_i(x_i, y_i, z_i)$. For all functions except (1), $f_i(r) = 0$ in the range of $r > R_i$.

For n metaballs, the shape of the curved surface is defined by the points satisfying the following equation.

$$f(x,y,z) = \sum_{i=0}^{n} q_i f_i - T = 0, \tag{5}$$

where T is a threshold, q_i the density values (negative values are acceptable). The normal vector at an isosurface can be calculated by $(-\frac{df(x,y,z)}{dx}, -\frac{df(x,y,z)}{dy}, -\frac{df(x,y,z)}{dz})$.

The drawback in Blinn'sfunction is that the function is not zero even at a long distance from the center point. Wyvill et al. modified Blinn'stechnique nicely. In their model, for $T = 0$, $f(0) = 1, f'(0) = 0, f(R_i) = 0, f'(R_i) = 0$ and $f(\frac{R_i}{2}) = \frac{1}{2}$ (see Fig.1). $T = 0.5$ works well. That is, when $T = 0.5$, the radius of a single ball is exactly one-half. If two balls are placed at the same location, it has twice the volume of the isosurface for a single ball (i.e., when metaballs merge, their volumes are added together). Thus, for geometric modeling, degree six polynomial function is useful.

One of the difficulties for rendering metaballs is to get an analytical solution of intersections between the ray and isosurfaces. Blinn solved this problem by using "regula falsi" and the Newton method.

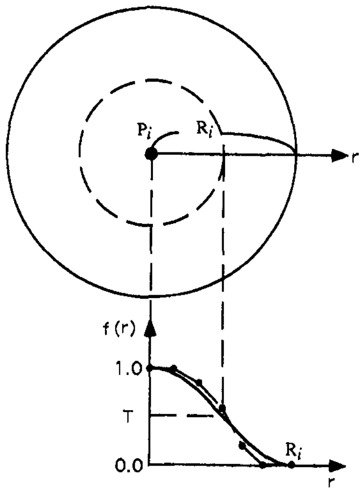


Figure 1: The field function of degree six polynomial (7 dots denoted the control points of Bézier curve, T=threshold).

In Nishimura's method, even though the degree of the field function with respect to the radius is low, the density distribution on the ray is a function having square root, so he employed approximation functions. Then he uses piecewise functions; even for the intersection test between the ray and a single ball, to express the density function on the ray, the ray should be divided into three sections. This results in the number of sections to be solved becoming huge for multiple metaballs.

Murakami employed degree four polynomial; as he employed the bisectional method, it is not robust and has calculation errors. Wyvill employed Laguerre's method[8] for the intersection test. Laguerre's method can get every real roots; but this method has a redundancy factor, because we need only one real root (a minimum value in the specified interval) for ray tracing. According to Wyvill, solving every root is useful for the CSG model, but even in this case, only one or two roots are required in one interval.

We employed Wyvill's degree six polynomial for the field function (see Fig.1) because of its capability as mentioned before. In the proposed algorithm, the density function on the ray is expressed by Bézier functions, a single root of this function is effectively and precisely solved by using *Bézier Clipping* [6], which was developed for ray tracing of Bézier patches. This is an iterative method which utilizes the convex hull property of Bézier curves, and converges more robustly to the polynomial's solution than does Newton's method. The proposed method is more than ten times faster than Laguerre's.

3 Intersection Test Between a Ray and Metaballs

As the density is defined within the sphere of metaball, first the intersection test between the ray and the sphere is done, followed by the intersection test between the ray and the isosurface.

As is well known, point P on the ray is expressed by using parameter t (the distance from the viewpoint):

$$\mathbf{P} = \mathbf{V}t + \mathbf{P}_0, \tag{6}$$

where $\mathbf{V}$ is the unit viewing vector, $\mathbf{P}_0$ the viewpoint.

The intersection between the ray and the sphere, whose center is P_i and its effective radius R_i, is solved by

$$At^2 - 2Bt + C = 0, \tag{7}$$

where $A = 1$, $B = (\mathbf{P}_i - \mathbf{P}_0) \cdot \mathbf{V}$, $C = \overline{\mathbf{P}_0}^2 - R_i^2$. the discriminant of this equation is given by

$$D = B^2 - AC. \tag{8}$$

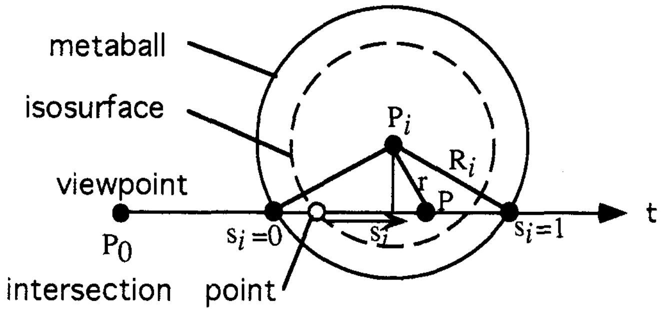


Figure 2: Intersection between a ray and a single metaball.

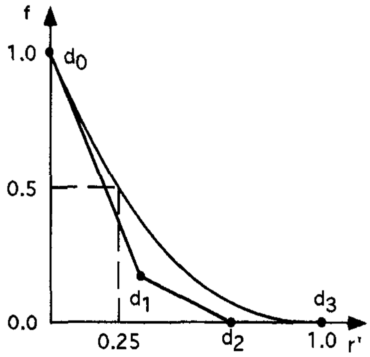


Figure 3: Field function expressed by Bézier curve with respect to r^2 (4 dots are the control points of a cubic Bézier curve).

When the ray intersects with the sphere ($D \geq 0$), the intersected interval in the parametric domain is expressed by (see Fig.2)

$$B - \sqrt{D} \leq t \leq B + \sqrt{D}. \tag{9}$$

For the above interval, the density distribution on the ray should be obtained to solve equation (5). In previous methods, as the density distribution is expressed by functions parameter t, the equation is complex, while in the proposed method, the function is reparameterized to the parameter within the intersected interval of eq.(9); the function is simplified by expressing a Bézier curve within the interval.

3.1 Intersection test between a ray and a single metaball

Because there is only a single metaball, an isosurface is a sphere (see Fig.2). Let's consider metaball i with radius R_i. The intersection test is enough to find the intersection points between the ray and a sphere with a relatively small radius ($< R_i$): Its radius is calculated as follows:

Assuming $r' = (\frac{r}{R_i})^2$, the expression in equation (4) becomes a cubic polynomial in r'. By converting to a Bézier function, equation (5) is expressed by

$$f(r') = \sum_{k=0}^{3} d_k^i B_k^3(r') - T, \tag{10}$$

where $(\frac{k}{3}, d_k^i)$ ($k = 0, 1, 2, 3$) is the coordinates of the control points of the Bézier curve (see Fig.3); $d_0^i = 1, d_1^i = \frac{5}{27}, d_2^i = d_3^i = 0$. B is the Bernstein polynomial and is expressed by $B_k^n(u) = \binom{n}{k} u^k (1-u)^{n-k}$ for degree n. Fig.3 shows the control points of equation (10) for $T = 0$. In this case, the isosurface is

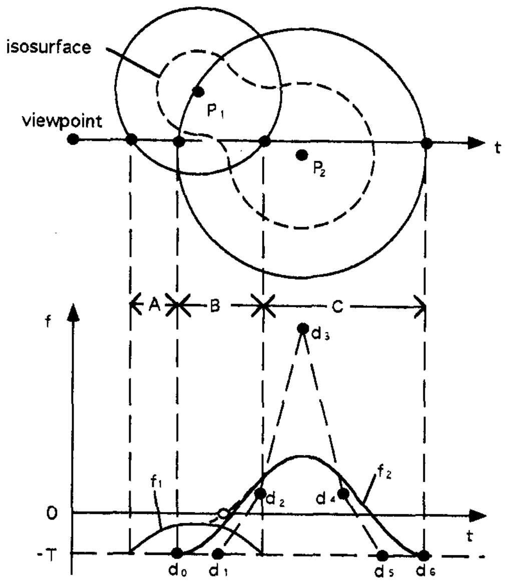


Figure 4: Density distributions expressed by Bézier curves on the ray.

a complete sphere (the dotted circle in Fig.2). The r' satisfying equation (10) can be solved by Bézier Clipping, then the intersection point between the sphere and the ray is calculated. As shown in Fig.3, the radius r for $T = 0.5$ is exactly one-half of R_i because of $r' = 0.25$.

3.2 Intersection test between a ray and multiple metaballs

First we consider metaball i with radius R_i. The length of the intersected interval is $2\sqrt{D}$ (see equation (8) for D). Let's consider a parameter $s_i (0 \leq s_i \leq 1)$ within the intersected interval between the ray and the sphere (see Fig.2). Assuming $a_i = \frac{D}{R_i^2}$ and r to be the distance between point P on the ray and the center of the sphere, P_i, the radius ratio r' $(= (\frac{r}{R_i})^2)$ is expressed by

$$r'(s_i) = 4a_i s_i^2 - 4a_i s_i + 1. \tag{11}$$

By substituting this equation to equation (4), the density function f_i on the ray is expressed by

$$f_i(s_i) = -\frac{256}{9}a_i^3 s_i^6 + \frac{768}{9}a_i^3 s_i^5 + \frac{16}{9}(5 - 48a_i)a_i^2 s_i^4 + \frac{32}{9}(8a_i - 5)a_i^2 s_i^3 + \frac{80}{9}a_i^2 s_i^2. \tag{12}$$

By converting equation (12) to a Bézier curve, f_i is expressed by

$$f_i(s_i) = \sum_{k=0}^{6} d_k^i B_k^6(s_i), \tag{13}$$

where $(\frac{k}{6}, d_k^i)(k = 0, 1, .., 6)$ are coordinates of control points; d_k^i are as follows:

$$d_0 = d_1 = d_5 = d_6 = 0, \quad d_2 = d_4 = \frac{16}{27}a_i^2, \quad d_3 = \frac{(8a_i + 5)a_i^2}{45}, \tag{14}$$

When $s_i = 0.5$, the function has a maximum value, $f_i(0.5) = \frac{(4a_i+5)a_i^2}{9}$, where $r'(0.5) = 1 - a_i$. That is, this function is a symmetric function centered at $s_i = 0.5$, and is a simple function because only three of d are non-zero; for example, when $a_i = 1$, $d_k = \{ 0, \ 0, \ 0.59, \ 2.31, \ 0.59, \ 0, \ 0 \}$ (see

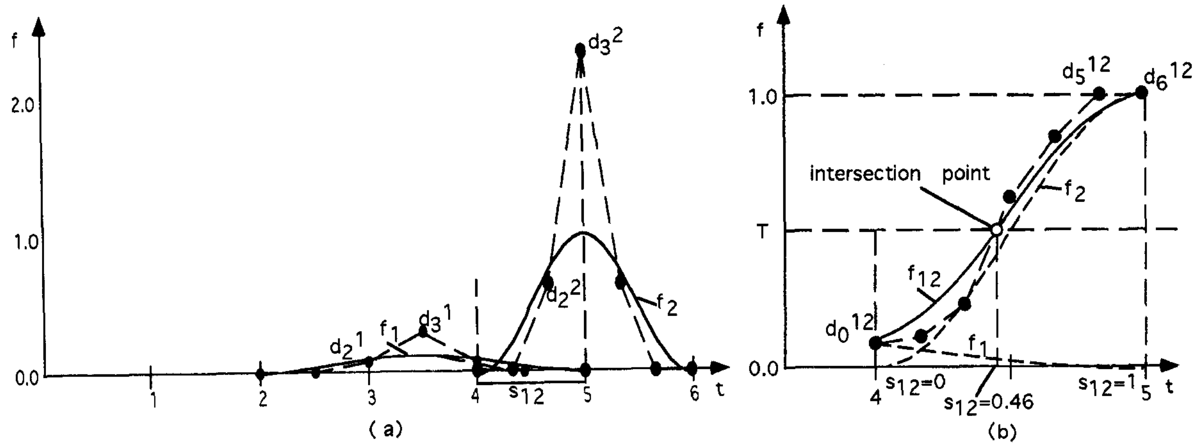


Figure 5: Calculation example of composition of two density distributions.

control points d_k in Fig.4). Though this method is applied to the degree six field function (see eq.(4)), it can be applied to field functions with any even number of degree (for odd number of degree, the density distribution can not be expressed by r'). Then the method is also applied to the degree four polynomial of eq.(3). For degree four filed function, the density function on the ray is more simple (see Appendix).

For a section which intersects with a single sphere (section A in Fig.4), the intersection is calculated by the method described in 3.1. But for a section which intersects with multiple spheres (section B in Fig.4), the intersection test is carried out as follows:

As shown in Fig.4, Bézier curves, f_1 and f_2, are clipped by the interval to be tested (i.e., section B in Fig.4), then both of the clipped curves are composited. This composite process of the curves is very simple, it is performed by simply adding each control points d_k^i belonging to f_1 and f_2; i.e., $d_k^{12} = d_k^1 + d_k^2 (k = 0,..,6)$. After compositing the curves, the new curve, f_{12}, is expressed by a degree six Bézier curve, then the roots are found by using Bézier Clipping. The intersection test *is* done in order for each interval from the viewpoint: The minimum root of s in the interval is found.

In Bézier Clipping, if there is a possibility of having multiple roots, the curve is split in half and Bézier clipping is resumed on each half. In this step, by first processing the half with the smaller parameter, the smallest root can be found first. The well known de Casteljau subdivision algorithm is applied for the subdivision. For opaque objects, only one intersection point closest to the viewpoint is required. Once the first root (a minimum real root) is found, the process is stopped. For displaying transparent objects, every root in each interval is required. But fortunately this method can output the roots in order of distances from the viewpoint, so the sorting process is unnecessary.

For clarifying our model, a calculation example having two metaballs, B_1 and B_2, on the ray is described below (see Fig.5). The density functions, f_1 for metaball B_1 and f_2 for metaball B_2 on the ray, exist on the intervals [2 5] and [4 6], respectively, in the parameter domain (*t* in this case). The intersection test for interval [4 5], which is the intersection of f_1 and f_2, *is* described here. f_1 and f_2 are expressed by parameters s_1 and s_2, respectively. And the composited function, f_{12}, is expressed by parameter s_{12}. Let'sconsider the density values, q_1 and q_2 for both metaballs as 1 (i.e., $q_1 = q_2 = 1$), and a used in equation (11) for B_1 and B_2 to be 0.5 and 1 (i.e., $a_1 = 0.5, a_2 = 1$), respectively. By using (14), the coordinates of the control points of density functions are $d_k^1 = \{0,\ 0,\ 0.148,\ 0.4,\ 0.148,\ 0,\ 0\}$ and $d_k^2 = \{0,\ 0,\ 0.59,\ 2.31,\ 0.59,\ 0,\ 0\}$. Toexpress thesefunctions by parameter s_{12} defined in the interval [4 5], f_1 and f_2 should be subdivide at $s_1 = 0.666$ and at $s_2 = 0.5$, respectively. After subdivision we can get new functions, $\{0.15,\ 0.12,\ 0.08,\ 0.04,\ 0.01,\ 0,\ 0$

} and { 0, 0, 0.15, 0.51, 0.83, 1, 1 }. By adding these coordinates of the control points, those of the composited function f_{12} are { 0.15, 0.12, 0.23, 0.55, 0.84, 1, 1 } (see Fig.5(b)).

The root, $s_{12} = 0.46$, is obtained by applying Bézier Clipping to $f_{12} - T = 0$, where $T = 0.5$. By converting from s_{12} to t, distance t (=0.46(5-4)+4: see Fig.5(b)) from the viewpoint is obtained, and (x, y, z) coordinates at the intersection point can be calculated by using equation (6).

3.3 Process for ellipsoids

In this subsection we demonstrate that the proposed method using Bézier curves can be applied to ellipsoids.

Let's denote the viewpoint $P_0(x_0, y_0, z_0)$, and unit viewing vector $V(V_x, V_y, V_z)$. For simple description, consider an ellipsoid with center at the origin and radii a, b, and c in x, y, z axes. Then the coefficients A, B, C in equation (7) for the ellipsoid are expressed by

$$A = (\frac{V_x}{a})^2 + (\frac{V_y}{b})^2 + (\frac{V_z}{c})^2, \quad B = \frac{V_x x_0}{a^2} + \frac{V_y y_0}{b^2} + \frac{V_z z_0}{c^2}, \quad C = (\frac{x_0}{a})^2 + (\frac{y_0}{b})^2 + (\frac{z_0}{c})^2 - 1. \qquad (15)$$

By using these coefficients A, B, and C, the discriminant is obtained from equation *(8)*, and the length between two intersections of the ray and the ellipsoid is $2\frac{\sqrt{D}}{A}$. $r'(s)$ is obtained by using equation (11) by setting $a_i = \frac{D}{A}$, and the density function on the ray is obtained by equation (13). For arbitrary ellipsoids, A, B, C are calculated after rotation and translation of V and P_0.

4 Comparisons of Computation Time

The comparisons between our method and Laguerre's in terms of computation time is discussed here. Wyvill employed Laguerre's method *[8]* to find roots of degree six polynomial. The comparisons are as follows:

(1) In Laguerre's method, the roots necessary are selected after finding all roots: n roots should be found for the degree n equation. In the ray tracing method, however, one positive minimum root is enough. In our method, only the minimum root within the specified interval can be solved. In the case of rendering transparent objects, all roots required in a specified interval can be solved in order of distances from the viewpoint.

(2) For finding sections having no root, as the density function on the ray is expressed by Bézier curves in our method, it is easy to test by using the convex hull property of Bézier curves; if every value of d_k in eq.(13) is positive (or negative), no root exists. In Laguerre's method, however, the result is only obtained after finding all roots.

(3) Bézier clipping uses only linear equations in each iteration, while Laguerre's method includes a square root in each iteration.

Many overlapped metaballs are usually required to get smooth surfaces in practical applications. As a result, the number of metaballs on a ray is large and many intervals to be solved exist, but fortunately only very few intervals have roots. In some cases, more than 50 metaballs exist on a ray. In our examples (see section 6) only 10 to 30 % of the intervals have effective roots (10% would mean that the first time a root is met is the 10th interval from the viewpoint). A quick test as to whether each interval has some roots or not is very important in metaball technique. For this reason, the proposed method is very effective.

The comparison of computation time for testing our method and Laguerre's is as follows: Table 1 shows the computation time of the degree six polynomial of equation (5) in several thresholds Ts; equation (14) is used for the control points ($a = 1$, d_k={ 0, 0, 0.59, 2.31, 0.59, 0, 0 }; see f_2 in Fig.5). As shown in Table 1, our method is almost ten times faster than the Laguerre's method in the case $T = 0.5$. For $T = -0.2$ and $T = 1.2$, any root does not exist in a specified interval (i.e.,

$0 \le s \le 1$). In the case an interval having no roots, our method is 30 to 200 times faster than the Laguerre'smethod.

More than half of the intervals have no root as described above, so the total processing time of our method, irrespective whether roots exist or not, is very short. As calculated from Table 1, in the case of 30% of intervals having roots, our method is 25 to 200(i.e.,1.18x70%+1.17x30%: 0.04 x 70% +0.12 x 30%) times faster than Laguerre's.

Table 1: Comparison of cpu times for filed function of degree six polynomial (terminate condition : tolerance is 0.001)

T	Laguerre's method (msec.)	Bézier Clipping (msec.)
-0.2	0.934	0.006
0.5	1.17	0.12
1.2	1.18	0.04

5 Rendering Transparent Objects

Rendering transparent objects is required to display inner structure such as in a human body or density distributions such as clouds. The intersection test between rays and isosurfaces can be carried out by using the technique described in the previous section, but the subject in this section is how to calculate the intensity of distributed density taking into account scattering/absorption effects due to particles in space. The density distribution of particles play an important part. In our model multiple scattering is ignored.

Intensity arriving at one'sviewpoint P_v from point P_b, I_v, is calculated by

$$I_v = I_0 exp(\tau(L_{vb})) + \int_{P_v}^{P_b} I_p F(\phi) \rho exp(\tau(l)) dl, \tag{16}$$

where I_0 is the intensity at P_b (behind the transparent objects), $\tau(L_{vb})$ the optical length between P_v and P_b (τ is obtained by integration of densities), L_{vp} the distance between P_v and P_b, I_p the incident at P on the ray, $F(\phi)$ the phase function (this is the function of angle ϕ between the incident direction and the scattering direction), ρ the density of particles, l the distance between P and P_v (i.e., integration variable).

There are two cases, uniform density and non uniform density of particles. For uniform density, the analytical solution can be obtained, while for non-uniform density the numerical integration along the ray is required. For the calculation of the density distribution on the ray equation (13) can be used. The main purpose of this paper is not to discuss a shading model, so the details of the integrations are omitted. See reference [5],[7], because the basic ideas are from these references.

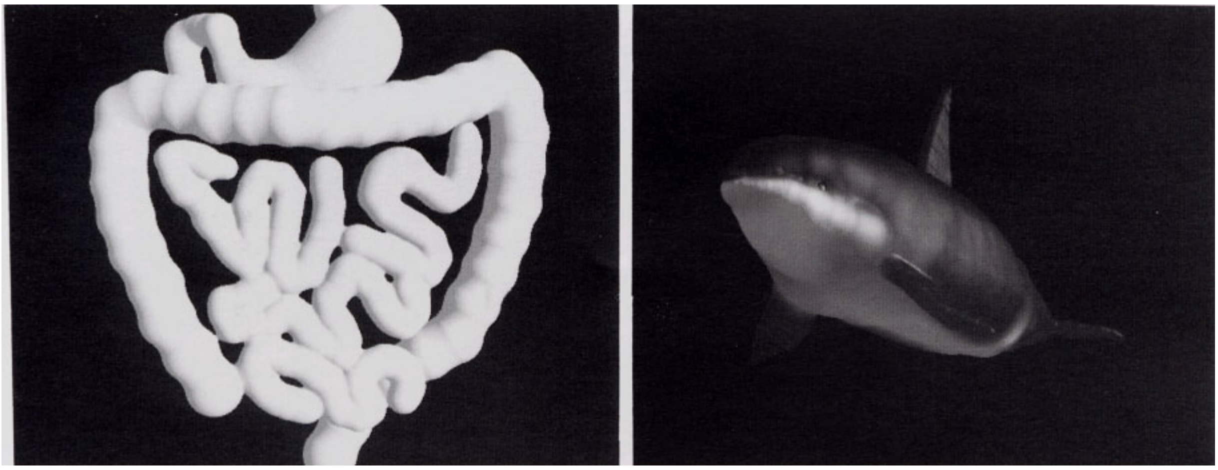
6 Examples

Fig. 6 shows some examples. Fig.(a) shows the stomach and the intestines consisting of 190 metaballs. Fig.(b) and Fig.7(a)*show a killer whale (the killer whale consists of 890 metaballs and the water bottom consists of 740 metaballs). Fig.7(b)*shows an example of ellipsoids; the hand consist of 94 metaballs (the ellipsoids are used for the fingers).

The calculation was done on an IRIS CRIMSON. The computation times for Fig.6(a),(b), Fig.7(a), and (b) were 25.9 sec., 35.4 sec., 159.8sec., and 28.8 sec, respectively (image width=512 pixels). In these examples the average number of iteration for a single intersection is only 2.0.

Fig.8 shows an example of transparent objects with nonuniform density; the clouds consist of 77 metaballs. The computation time is 226.9sec. In this case, the cpu time is relatively slow because the numerical integrations of the density are included.

* **See page C–533 for Figure 7.**



(a) (b)

Figure 6: Examples of metaballs.

As is shown in these examples, the proposed method is very effective for rendering objects consisting of a large number of metaballs.

7 Conclusions

As shown in the examples, the proposed method gives us photo-realistic images with smooth surfaces. In the proposed algorithm, the field function on a ray is expressed by Bézier functions, the root of this function is effectively and precisely solved by using *Bézier Clipping.* The conclusions of the proposed method are as follows:

(1) In this method, the density function on the ray is reparameterized to the parameter within an intersected interval; the function is simplified by expressing the Bézier curve within the interval, and the root finder is accelerated while maintaining the required precision.

(2) Intersection tests are performed by Bézier Clipping resulting in few iterations. Especially, it is very fast in the case of them being no root in an interval.

(3) The proposed method can be applied to field function expressed by any even number of degree of polynomial; even if more ideal density functions are found in the future, the basic idea of the proposed method can be applied to them.

Acknowledgment The authors would like to acknowledge Mr. I. Toshimiysu and Mr. T. Maeda for their help in making the data. We also would like to acknowledge Prof. Sederberg of Brigham Young University for discussion on root finders, and Mr. Mel Spencer for supporting some of the programs for timing tests.

References

[1] J.F. Blinn, "A Generalization of Algebraic Surface Drawing," ACM Transaction on Graphics, Vol. 2, No.3, pp. 235-256, (1980).

[2] G.L. Graves, "The Magic of Metaballs,", Computer Graphics World, pp.27-32, (1993 May).

[3] S. Murakami, H. Ichihara, "On a 3D Display Method by Metaball Technique,"Journal of papers given by at the Electronics Communication, Vol.J70-D, No.8, pp.1607-1615, (1987) (in Japanese).

Figure 8: An example of a transparent object by using metaballs.

[4] H. Nishimura, M. Hirai, T. Kawai, T. Kawata, I. Shirakawa, K. Omura, "Object Modeling by Distribution Function and a Method of Image generation,", Journal of papers given by at the Electronics Communication Conference ' 8 5J68-D(4) pp.718-725 (in Japanese).

[5] T. Nishita, Y. Miyawaki, E. Nakamae, "A Shading Model for Atmospheric Scattering Considering Distribution of Light Sources,"*Computer Graphics,* Vol. 21, No. 4, pp. 303-310, (1987).

[6] T. Nishita, T.W., Sederberg, M. Kakimoto, "Ray Tracing Rational Trimmed Surface Patches,'' Computer Graphics, Vol.24, No.4, pp.337-345, (1990).

[7] T. Nishita, T. Shirai, K. Tadamura, E. Nakamae, "Display of The Earth Taking into Account Atmospheric Scattering,"*Computer Graphics,* Vol. 27, No. 4, pp. 175-182, (1993).

[8] A. Ralston, "A First Course in Numerical Analysis,"McGraw Hill ,pp.368-371, (1965).

[9] B. Wyvill, C. McPheeters, G. Wyvill, "Data structure for soft objects,"The Visual Computer, 2, pp.227-234, (1986).

[10] B. Wyvill, C. McPheeters, G. Wyvill, "Animatingsoft objects,"The Visual Computer, 2, pp.235-242, (1986).

[11] G. Wyvill, A. Trotman, "Ray-Tracing Soft Objects,"CG International ' 9 0 ,pp.439-475, (1990).

Appendix : Distribution on a ray in case of degree four field function

By substituting r' in equation (11) for the field function of equation (3), the density distribution on the ray expressed by parameter s_i is

$$f_i(s_i) = 16a_i^2(s_i^4 - 2s_i^3 + s_i^2) \tag{17}$$

By converting equation (17) to a Bézier curve, coordinates of control points in equation (13), d_k, are as follows:

$$d_0 = d_1 = d_3 = d_4 = 0, \quad d_2 = \frac{8}{3}a_i^2. \tag{18}$$

The equation (18) is a very simple function because only d_2 is non-zero. When $s_i = 0.5$, the function has a maximum value, $f_i(0.5) = a_i^2$.