3D Distance from a Point to a Triangle

Mark W. Jones
Technical Report CSR-5-95
Department of Computer Science, University of Wales Swansea

February 1995

Abstract

In this technical report, two different methods for calculating the distance between a point and a triangle in 3D space will be described. It will be shown that it is far more efficient to calculate the distance by using a rotation to make the problem 2D. Both methods are tested with relation to the production of 3D data from polygonal meshes (voxelisation).

1 Introduction

The problem of finding the distance from a point to a triangle in 3D space arose with repsect to producing 3D data from a polygonal mesh. In this case the input to the process of voxelisation [1] is a triangular mesh, and the output is voxel data – a 3D array of data values. Voxelisation is needed to overlay synthetic objects into collected data. For examples with application to the integration of objects and medical data see Wang and Kaufman [2] and for the integration of objects with terrain data for flight simulation see Cohen and Gotsman [3].
The problem is that of finding the distance from a point P_0 to a triangle $P_1P_2P_3$, where P_i is a point in three dimensional space. This is more difficult than it seems at first due to the different possibilities that exist. The point P_0 could be closest to the plane of the triangle, closest to an edge, or closest to a vertex. In Section 2 a method of calculating that distance in 3D is given, and in Section 3 a method which reduces the problem to 2D is described. Section 4 presents results demonstrating that the 2D method is more efficient than the 3D method.

2 3D Method

Approaching the problem in three dimensions requires the projection of P_0 onto the plane of triangle $P_1P_2P_3$ to create P_0' (Figure 1).

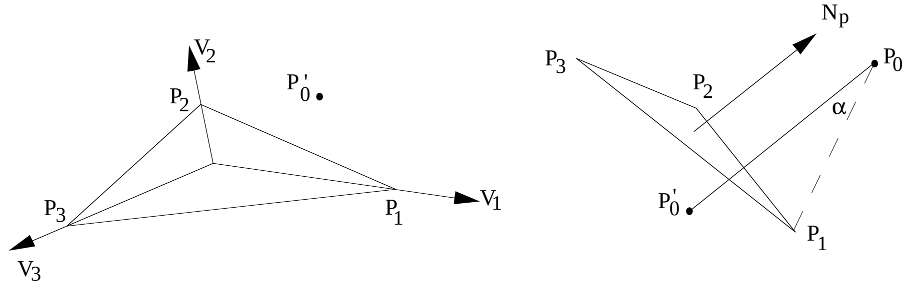

Figure 1: Calculating the distance of P_0 from $P_1P_2P_3$.

The normal N_p of $P_1P_2P_3$ can be calculated as

$$N_p = P_1P_2 \times P_1P_3 \tag{1}$$

The angle, α between the normal N_p and P_1P_0 is calculated

$$cos\alpha = \frac{P_1P_0 \cdot N_p}{|P_1P_0||N_p|} \tag{2}$$

The length of the vector P_0P_0' can be found using

$$|P_0P_0'| = |P_0P_1|cos\alpha \tag{3}$$

The vector P_0P_0' can then be determined

$$P_0P_0' = -|P_0P_0'|\frac{Np}{|Np|} \tag{4}$$

(The negative sign is used since P_0P_0' has direction opposite to that of N_p.)
P_0' can then be calculated as

$$P_0' = P_0 + P_0P_0' \tag{5}$$

If it was the case that the projection of P_0 onto the plane lay within the triangle $P_1P_2P_3$, the length $|P_0P_0'|$ (Equation 3) would be the distance of P_0 from $P_1P_2P_3$.
If instead P_0' falls outside the triangle, the distance to the triangle is the distance to the closest edge or vertex to P_0'. In order to determine which edge or vertex P_0' is closest to, the position of P_0' in relation to the three vectors V_1, V_2 and V_3 must be found, (Figure 1) where

$$V_1 = \frac{P_2P_1}{|P_2P_1|} + \frac{P_3P_1}{|P_3P_1|}, \quad V_2 = \frac{P_3P_2}{|P_3P_2|} + \frac{P_1P_2}{|P_1P_2|}, \quad V_3 = \frac{P_1P_3}{|P_1P_3|} + \frac{P_2P_3}{|P_2P_3|} \tag{6}$$

If $f_1 = (V_1 \times P_1P_0') \cdot N_p$, then $f_1 > 0$ if P_0' is anticlockwise of V_1. Similarly, f_2 and f_3 can be calculated for the other vectors. Using f_1, f_2 and f_3 the position of P_0' in relation to the vectors V_1, V_2 and V_3 can be determined. It further has to be determined if P_0' is inside the triangle, and if so the distance from the triangle is the distance calculated in Equation 3.
If P_0' is clockwise of V_2 and anticlockwise of V_1, it is outside the triangle if

$$(P_0'P_1 \times P_0'P_2) \cdot N_p < 0 \tag{7}$$

and similarly for the other cases.
If P_0' is found to be outside the triangle, it is either closest to a vertex, or the side. For example, assume P_0' is closest to P_1P_2. (it is easy to apply the following to the remaining edges)
The direction R, of P_0' to P_0'' is given by

$$R = (P_0'P_2 \times P_0'P_1) \times P_1P_2 \tag{8}$$

and the angle γ is

$$cos\gamma = \frac{P_0'P_1 \cdot R}{|P_0'P_1||R|} \tag{9}$$

The length $P_0'P_0''$ is calculated using

$$|P_0'P_0''| = |P_0'P_1|cos\gamma \tag{10}$$

and $P_0'P_0''$ is

$$P_0'P_0'' = |P_0'P_0''|\frac{R}{|R|} \tag{11}$$

The point P_0'', which is the projection of P_0' onto the line P_1P_2 is

$$P_0'' = P_0' + P_0'P_0'' \tag{12}$$

Let

$$t = \frac{P_0'' - P_1}{P_2 - P_1} \tag{13}$$

If $0 \leq t \leq 1$, P_0'' is between P_1 and P_2 and the distance of P_0' from the line is $|P_0'P_0''|$ as calculated in Equation 10. The distance of P_0 to P_1P_2 is $\sqrt{(|P_0'P_0''|^2 + |P_0P_0'|^2)}$.
If $t < 0$, P_0 is closest to vertex P_1 and can be calculated as the distance $P_1 - P_0$.
If $t > 1$, P_0 is closest to P_2.
Therefore the distance of P_0 to $P_1P_2P_3$ has been calculated. It should be noted that Equations 1 and 6 can be precalculated for efficiency.

3 2D Method

The second approach is that of converting the problem into a two dimensional problem. The simplest way to achieve this is by calculating the translation and rotation matrix to rotate the triangle $P_1P_2P_3$ so that P_1 lies on the origin, P_2 lies on the z axis, and P_3 lies in the yz plane. This transformation matrix can be calculated once for each triangle in a preproccessing step, and can then be used to transform P_0 to P_0'. P_0 can be trivially projected onto the triangles plane giving P_0' by ignoring its x coordinate since the triangle is in the yz plane. If P_0' is inside the triangle $P_1P_2P_3$, the distance from the point to the triangle is simply the x coordinate of P_0.
Using P_0', determination of the closest part of triangle $P_1P_2P_3$ to P_0 can be found by using the edge equation [4], and once determined the distance can be found in a standard way. The edge equation is simply:

$$E(x, y) = (x - X)\,dY - (y - Y)\,dX \tag{14}$$

for a line passing through (X, Y) with gradient $\frac{dY}{dX}$ with respect to a point (x, y). If $E < 0$ the point is to the left of the line, if $E > 0$ to the right, and if $E = 0$, it is on the line. In Figure 2 we see the different possibilities.

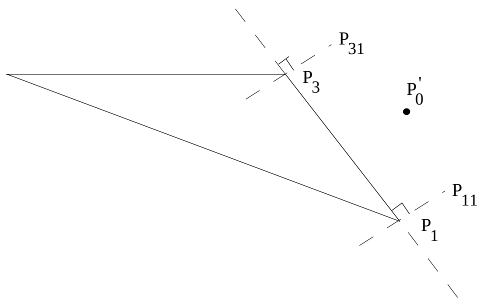

Figure 2: Calculating point position relative to triangle.

If P_0' is left of P_3P_1 it is closest to P_3P_1 if it is to the right of P_3P_{31} and to the left of P_1P_{11}. The proximity to the other edges of the triangle can be similarly determined. For P_0' to be closest to vertex P_1 it must be right of P_1P_{11} and left of P_1P_{12}, where P_1P_{12} is defined at right angles to P_1P_2. Using just these edge equations, the closest vertex or edge of $P_1P_2P_3$ to P_0' can be determined. The lines P_1P_{11}, P_1P_{12}, P_3P_{31}, P_3P_{32}, etc. and their directions can be precomputed, thus enabling simple applications of the edge equation to determine which part of the triangle the point P_0' is closest to. Once determined, the distance can be calculated to that part in the normal way.

4 Results

The voxelisation process was implemented as in [1], with two distance functions – one the 3D version of Section 2, and one the 2D version of Section 3. The voxelisation process was carried out using both distance functions, and times were compared for various datasets. It can be seen from Table 1 that the 2D method was consistently faster than the 3D method, and in fact took a quarter of the time to compute the same dataset. The reason for such a marked difference in computation times could be put down to the number of square roots that must be computed. For the 3D method four square roots are required ($|P_1P_0|$, $|R|$, $|P_0'P_1|$ and $|P_0'P_0''|$), as opposed to one final distance calculation square root for the 2D method.

Data set	No. of triangles	Distance computations	Time 2D method	taken 3D method	Ratio 3D ÷ 2D
Octahedron	8	99576	1.017	2.917	2.87
Dodecahedron	36	1170754	8.016	33.949	4.24
Soccerball	116	3850470	25.649	111.496	4.35
Teapot	252	5717484	38.032	159.744	4.20
King	3080	17740398	144.294	519.546	3.60
Queen	2600	16378497	130.028	491.447	3.78
Bishop	2360	14092809	114.112	410.334	3.60
Pawn	1600	14791639	116.912	420.033	3.59
Knight	1524	10191575	86.113	286.889	3.33
Rook	1600	18501156	144.544	541.462	3.75

Table 1: Table showing voxelization timings

In conclusion, a two dimensional method for computing the distance of a point from a triangle has been compared against a standard three dimensional method. The results indicate that the two dimensional method performs far more efficiently than the three dimensional method, and is suited for any application where distance measurements to three dimensional triangles are required. The method was tested with particular attention to the process of voxelisation, which requires intensive use of point to triangle measurement.

References

[1] M. W. Jones. Voxelisation of polygonal meshes. In *Proceedings of 13th Annual Conference of Eurographics (UK Chapter) (Loughborough, March 28-30, 1995)*, pages 160–171, March 1995.

[2] S. W. Wang and A. E. Kaufman. Volume sampled voxelization of geometric primitives. In *Proc. Visualization 93*, pages 78–84. IEEE CS Press, Los Alamitos, Calif., 1993.

[3] D. Cohen and C. Gotsman. Photorealistic terrain imaging and flight simulation. *IEEE Computer Graphics and Applications*, 14(2):10–12, March 1994.

[4] J. Pineda. A parallel algorithm for polygon rasterization. In *Proc. SIGGRAPH '88 (Atlanta, Georgia, August 1-5, 1988)*, volume 22(4), pages 17–20. ACM SIGGRAPH, New York, August 1988.