MULTI-PROCESSOR ROTATION AND COMPARISON OF OBJECTS

Brian V. Funt
Department of Computing Science
Simon Fraser University
Vancouver, British Columbia V5A 1S6

ABSTRACT

A parallel processing method of rotating and comparing three-dimensional objects is presented. The multi-processor has an abstract structure in which the Individual processors are located at the vertices of a geodesic dome. The design has been tested on the same type of object matching problems as Shepard (3) used in his mental rotation experiments with human subjects. The same linear "reaction time** behaviour is found*

I INTRODUCTION

Shepard*s experiments demonstrated that human subjects can determine that two identical objects in different orientations are the same, with a reaction time that is proportional to their angular separation. The objects are simple — composed of a few unit cubes joined at their faces -- and are presented to the subjects as line drawings* Shepard [4] uses his results to argue that people carry out mental rotations by an analog process, where by analog he means: "...a process in which the intermediate Internal states have a natural one-to-one correspondence to appropriate intermediate states in the external world*" This paper presents a parallel-process computational model for solving the object rotation and comparison task. It is relevant to machine vision and computer graphics problems where the modelling of three-dimensional objects Is involved*

The linear reaction time Is a direct result of the fact that the model involves parallel processing. The processors can directly communicate only with their immediate neighbours. Each processor has its own local memory. The representation of an object is distributed amongst the processors. The complete object is represented by the total Information stored by all the processors. In order to compare two objects it is first important to "align" them in the sense that each processor must hold Information from corresponding parts of the two objects* Once aligned, a test is made to determine how similar they are. In this test, each processor compares the Information it has about the local shape of the first object with that it has about the second*

II THE MODEL

The alignment process corresponds to the rotation of an object. To understand why the alignment process is proportional to the angular separation of the objects we must look further Into the way in which objects are modelled. The abatract structure of the multi-processor system is as shown in Figure 1. The processors are distributed over the surface of a sphere, like the tiles in the figure, and have direct communication links to their neighbours* Most are hexagons having six neighbours, but a few are, of necessity in this type of tesseiation, pentagons. The modelled object is considered to be inside this sphere of processors with its center of gravity at the center of the sphere. Each processor stores the radial distance of each intersection of the object's surface with the radial line from the processor's center to the center of the sphere. In addition, because a collection of points by itself does not define a surface, for each intersection point a link is stored to the neighbouring points on the object's surface. This representation is related to that of Brown [2]. Note that the spherical abstract structure of the multi-processor does not require a spherical physical structure — it is only necessary to have the same physical communication structure. The same object in two different orientations will have its shape information differently distributed amongst the processors. To align two objects requires shifting the information describing one of them* To shift the information means that messages must be sent between processors. The time taken for a message to get to its destination processor is a linear function of the number of intervening processors which have to relay it. The number of intervening processors is proportional to the angular separation of the source and destination processors because of the uniform distribution of processors over the sphere* The processors holding corresponding information from the two objects which are farthest apart are those along the "equator" of the rotation, and the angular difference between them will be the same as the angular difference between the two objects. In summary, the time taken to "rotate" one object Into another Increases linearly with the angular separation between the objects because the information describing it must be relayed through a linearly increasing sequence of processors.

III THE COMPUTER SIMULATION

A LISP program has been written which, while simulating the parallel hardware, carries out the parallel algorithms for rotating and comparing two Shepard-style objects. The Input to the program is two line-drawings expressed as a set of line segment endpolnt coordinates. The output is a statement of whether or not the two objects are the same or different. A brief description of the program follows.

The rotation mechanism is the most important part of the program. Before a rotation can be begun, the axis of the rotation must be found. This is a serious problem because If one object is to be rotated into correspondence with another In linear time, it is not possible to spend time searching for the correct axis of rotation unless the search time happens to be either constant or linearly increasing. Fortunately, it is possible to determine the unique axis and magnitude of rotation from the moments of inertia of the two objects (this observation was made at least as early as Baumgart [1)). An object's moments of inertia (a generalization of center of gravity) define an Inertlal ellipsoid, and the required axis of rotation is the axis which will align one object's Inertlal ellipsoid with the other's* Having Identical and aligned Inertlal ellipsoids does not guarantee that two objects are the same, but It does guarantee that if they are the same they will be oriented in the same direction. The moments-of-inertla computation is a parallel process.

To rotate an object about a given axis each processor first computes what its own new location (expressed in spherical coordinates) would be If the point where the axis intersects the sphere of processors were to become location (1,0,0) — the "north pole". This effectively makes the axis of rotation the north-south axis of a spherical coordinate system thereby reducing subsequent calculation. The object is rotated in steps where the step else is no larger than the maximum angular difference between two processors. After each step, each processor checks to see whether the coordinates of the points it is holding are nearer to its center or to the center of a neighbouring processor. If the point is actually closer to one of the neighbours, then it is transmitted as a message to the neighbouring processor which then takes over responsibility for it. The step size is limited so that a point never moves farther than the distance between neighbours.

Once one object has been rotated into correspondence with the other, each processor does a simple comparison of the points it is holding for the two objects. If the corresponding points are within a given tolerance of one another, then they are considered to be matching. If no processor discovers a mismatch of points then the two objects are reported to be Identical.

The one remaining aspect of the implementation is how the line drawings are processed to obtain a three-dimensional model and stored in the sphere of processors. In this, the limited nature of the domain of object types has been exploited. Almost every closed curve in one of the line drawings Is the side of a unit cube* Knowing that a closed curve is a perspective projection of a unit square is sufficient information to provide a set of four equations in four unknowns which can be solved for the exact location and orientation of the face it represents*

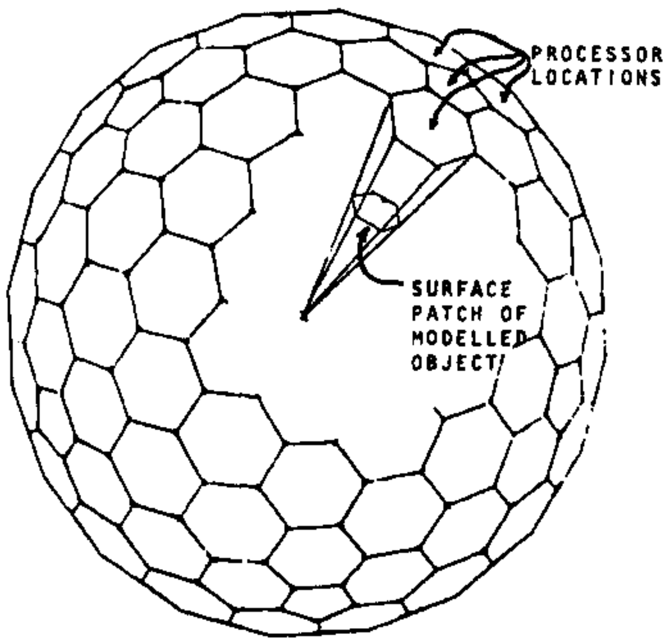

Figure 1: The Sphere of Processors.

IV CONCLUSION

The current simulation la baaed on a aet of 482 processors. In a truly parallel implementation the proceaalng time la guaranteed to be a linear function of the angular difference In the object's orientatlona by the fact that the rotationa are carried out in steps of fixed atep size as neceaaitated by the conatrainta of message propagation through a network of locally connected proceasors. We can aee — in contraat to what would be the caae during the execution of a serial matrix multiplication baaed algorithm — that at all times during the rotation process an object's repreaentation la in a consistent state. Thus the Internal atatea of the rotation have "a natural one-to-one correspondence to appropriate intermediate atatea in the external world". By both aatlafying Shepard'a definition of an analog proceaa and by exhibiting linear reaction-time behaviour, the parallel algorithm discussed above provides a computational model of the mental rotation phenomenon.

ACKNOWLEDGEMENTS

Edwin Bryant, Thomas Strothotte and Leigh Hunt Palmer have made aubatantlal contributions to thla paper. The financial support of the National Science and Engineering Reaearch Council of Canada under grant A4322 la gratefully acknowledged.

REFERENCES

[1] Baumgart, B.C., Geometric Modeling for Computer Vision, Ph.D. Thesis Stanford Univ. AIM 249, 1974.

[2] Brown, C.R., Faat Dlaplay of Well-Teaaelated Surfacea, Computers and Graphics, Vol 4 (1979) 77-85.

[3] Shepard, R., and Metzler, J., Mental Rotation of Three-Dimenalonal Objecta, Science, 171 (1971) 701-703.

[4] Shepard, R., The Mental Image, American Psychologist, February 1978, 135.