ACQUISITION AND MODELING OF HUMAN BODY FORM DATA*

SAISIE ET REPRESENTATION DE DONNEES SUR LES FORMES DU CORPS HUMAIN

Henry Fuchs, Joe W. Duran, Brian W. Johnson, and Zvi. M. Kedem
The University of Texas at Dallas
Richardson, Texas (U.S.A.)

Abstract

We describe some of our recent work on the computer-assisted acquisition, interaction, manipulation and display of three-dimensional data in general, and the human body form in particular. The work encompasses four distinct areas : 1) the automatic acquisition of 3-D surface points using a light-scanning technique, 2) the automatic construction of mathematical surface descriptions from point-contour data, 3) interaction and manipulation of 3-D data with multiple hand-mounted sensors, and 4) real-time stereo display of computer-synthetized continuous-tone images on special head-mounted display devices.

We present some of our preliminary results in each of these areas and outline how they will be combined into a single comprehensive system. This system we expect will significantly enhance the computer's capabilities to assist in working with complex three-dimensional structures such as human bodies. As a simple example, we expect the system should be able to automatically digitize a human model, and within a few minutes generate a three-space approximating surface conforming to the human model's surface. Another person, wearing a helmet-like display device, will see through his helmet's visor not only the human model, but also the computer-generated approximating surface superimposed on it. As this human viewer moves around the model, the computer-generated surface will appear to stay on the model, and he'll be able to reach out and modify any portion of it or indicate that the digitizing system should re-scan parts of the surface whose approximation he deems unsatisfactory.

Other applications of the system — for diagnostic medicine, architecture, and molecular modeling — will also be suggested.

Résumé

Nous décrivons ici quelques uns de nos récents travaux sur la saisie, la manipulation interactive et la visualisation assistée par ordinateur de données en trois dimensions, plus particulièrement de données concernant le corps humain. Ces travaux recouvrent quatre domaines distincts : 1) la saisie automatique des coordonnées de points dans l'espace à trois dimensions, utilisant une technique de balayage lumineux, 2) la construction automatique de modèles mathématiques de surface à partir de données sous forme de contours (courbes de niveau), 3) l'interaction et la manipulation des données dans l'espace à l'aide de détecteurs tenus à la main, 4) la visualisation stéréographique en temps réel d'images en «tons continus» synthétisées par ordinateur sur des écrans montés sur la tête de l'observateur.

Nous présentons quelques-uns de nos résultats préliminaires dans chacun de ces domaines et indiquons comment ils seront intégrés dans un seul système. Nous pensons que ce système améliorera de façon significative l'aide que l'ordinateur pourra apporter dans les travaux sur des structures à trois dimensions aussi complexes que celles du corps humain. Pour donner un exemple simple, ce système devrait être capable de représenter numériquement et automatiquement, un sujet humain et de sortir en quelques minutes une surface en trois dimensions, proche de la surface réelle du sujet. Une personne, portant sur la tête deux écrans de visualisation, comme un casque, verra sur ces écrans non seulement le sujet, mais, superposée à lui, la surface synthétisée par l'ordinateur. Si l'observateur se déplace autour du modèle, la surface ainsi synthétisée apparaîtra fixe par rapport au sujet, et il pourra le «toucher» et en modifier n'importe quelle partie, ou indiquer au système qu'il désire un nouveau balayage de la section dont la représentation numérique ne semble pas satisfaisante.

D'autres applications du système aux diagnostics médicaux, à l'architecture et à la représentation des molécules seront également évoquées.

---

* This research was supported in part by NSF Grant MCS77—03905 and a University of Texas Organized Research Grant.

ACQUISITION AND MODELING OF HUMAN BODY FORM DATA

Introduction

One of the most exciting potential uses of computers involves their interaction with three-dimensional data and three-dimensional objects. From visualizations of neural structures to simulations of weather patterns around mountain ranges, we are confronted with problems involving three-dimensional objects. One basic difficulty has been the disparity between the inherent three-dimensional complexity of many objects and the two-dimensional nature of our usual communication media such as paper and photographic film. Any reduction to such media involves either distortion or loss of information of fragmentation of information into disjoint components (e.g., maps, cross-sectional images of human anatomy).

The flexibility of the computer in representing information presents, perhaps for the first time, a medium in which such 3D information can be kept intact.

From the computer scientist's perspective, measurement and interaction with human body form data presents a challenging problem. The data must be collected from an object which is irregular, non-rigid, and may be moving. Worse, it is data with which everyone is familiar and so any flaws or distortions are readily apparent. (Not many people will recognize a simulated retinal bipolar cell as being represented badly!). The work to be described in this paper represents an attempt to deal with such difficult structures as the human body. It is hoped that the results of such basic investigations will be found useful in a variety of applications.

Automatic Digitization of Three-Dimensional Surface Points

The system we are currently developing for this purpose consists of a computer controlled (spot) light deflection system and a variable number of simple linear light sensors (typically 3 to 10), enabling a three-dimensional point computation by a simple triangulating technique (Fuchs, Duran, and Johnson, 1977). The sensor design is based on linear arrays of CCD (charged coupled device) light sensors (Figure 1a). Such a device consists of a series of light sensitive cells which may be digitally controlled in a maner similar to a shift register. The amount of light energy detected over a period of time in each cell can then be converted to a digital values (say, 9-bit) yielding a vector of digital values (Figure 1b). Our present sensors each consists of one of these arrays with a knife edge positioned a short distance in front (Figure 2).

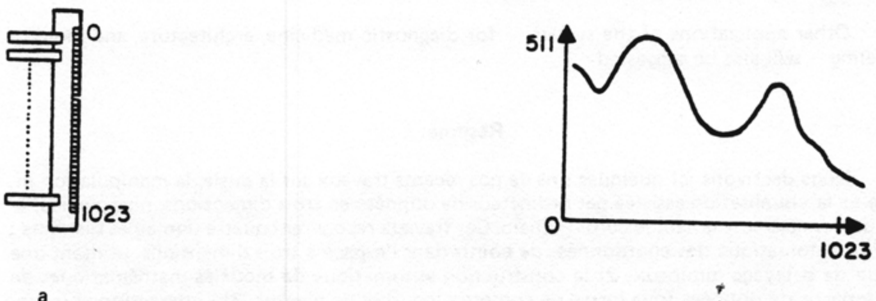


Figure 1 : a) CCD sensing array and b) the digitized signal

When this arrangement faces the environment to be measured and in this environment there is a single point source of light, the knife edge will cast a shadow onto the column of CCD elements (as in Figure 2). The vector of resulting digital values is a characteristic step function in that those CCD elements in shadow region will not receive as much light as those elements illuminated by the light source (Figure 3).

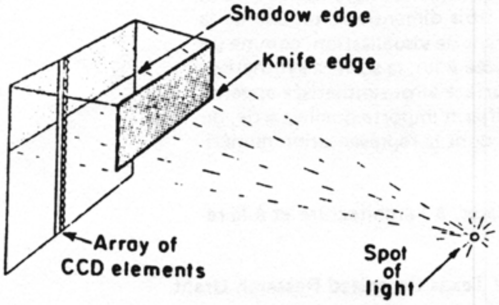


Figure 2 - Basic sensor design.

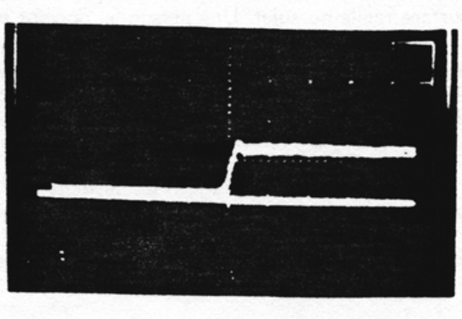


Figure 3 - CCD output from «shadow» sensor.

The location of the edge of the shadow on the CCD and the line of the blade edge together define a plane in the environment which must contain the point source of light. When several independent such sensors each detect a shadow from the same light source, then the location of the light source must be at the intersection of all the planes (Figure 4). (A similar system was developed by Burton and Sutherland (1974).

We are currently exploring variations on the sensor design. One of these which appears promising is to substitute a cylindrical lens for the knife edge in front of the sensing array. In such a variation (see Figure 5) the array of values would contain a peak rather than an edge at the location of the light source, and further, would increase the sensitivity of the unit by concentrating a larger amount of light onto the sensing elements.

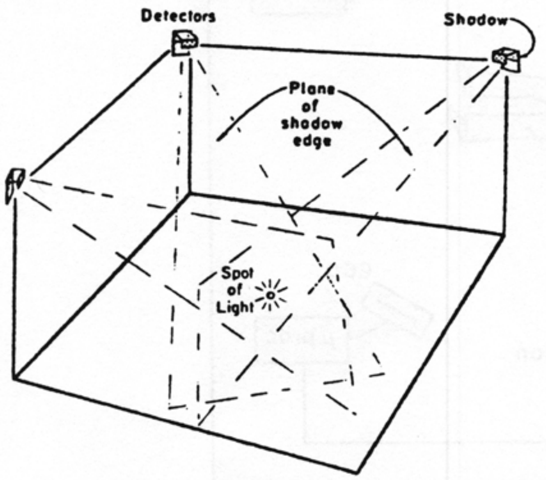


Figure 4 - 3-D position from 3 linear sensors.

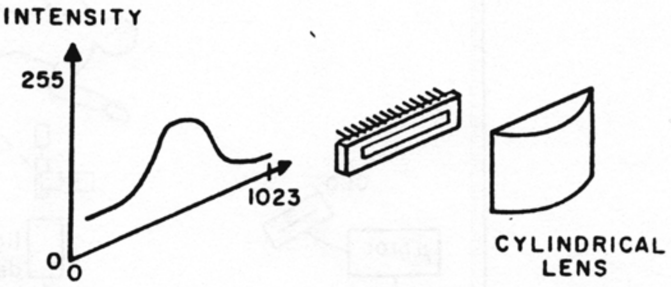


Figure 5 - Cylindrical lens sensor design.

The overall system can also be used in an entirely different mode to digitize a large number of positions concurrently, for instance a dozen points located around the body of a human dancer. In this mode a separate light source is placed at each location of interest and each light source is turned on in sequence, with only a single light being on during any one measurement cycle.

In a still different mode the system can be used to digitize an entire surface automatically. The principle behind this mode is based on the fact that the point source of light whose 3-D position is to be measured need not necessarily be a simple light source such as an LED, as in the modes described above. Rather, it could be a surface scattering of a narrow beam of light originating from some convenient location such as the ceiling of the room (Figure 6). In this way, after a particular point on the surface has been digitized the light beam can be quickly deflected (of course, under computer control) to another position of interest. The entire visible surface of an object can be digitized in this manner by deflecting the beam through a sequence of positions, either in a fixed pattern or in a pattern that depends on the local contour change – the point density being increased in regions of rapid change, and reduced in relatively flat regions. The sequence of beam deflections can be chosen to emphasize regions of interest, such as the face, or to acquire a minimal set of «interesting» points such as silhouette edges from some particular point of view.

To make the sensing system as easy to use as possible, the design incorporates an extensive amount of distributed processing. Each sensor is controlled by its own microprocessor which, on command from the central controller, scans out some sequence of intensity values from its CCD chip, digitizes each of them, then determines the most likely location of the edge. This single value – representing the edge location, with a special nil code if no likely edge is found – is the only information which needs to be returned to the central controller (Figure 7).

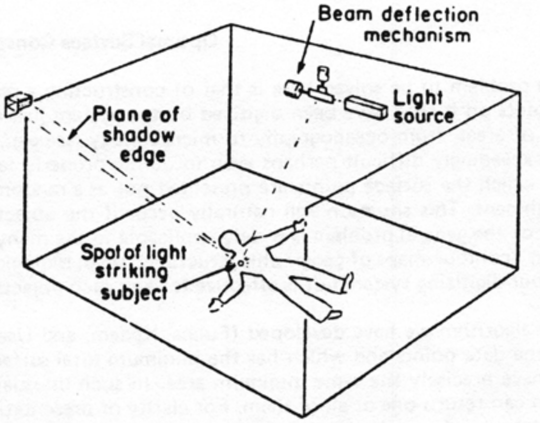


Figure 6 - Automatic surface digitization mode.

The central controller, in addition to coordinating communications with the individual sensing units, sets the deflection parameters for the focused light source or, in the multi-point digitization mode, sequences the various point light sources in the proper order.

After the central controller receives the edge location value from each sensing unit, it can compute the intersection in the 3-D environment coordinate system. This result, an XYZ triple, is the basic information provided by the overall system to the host computer. As such the entire system can be treated as a single input device, connected to the host computer via a standard asynchronous serial interface just like a standard terminal. Further, any sensing unit can be readily repositioned since the calibration to the environment coordinate system is fast and simple; the unit and the central controller simply have to compare the measured value and the known position of approximately seven locations in the environment. In fact, the entire system — consisting simply of breadbox-sized sensing units and the small central controller, connected by light terminal cables — can be easily transported to a new location and connected to virtually any computer's terminal connector.

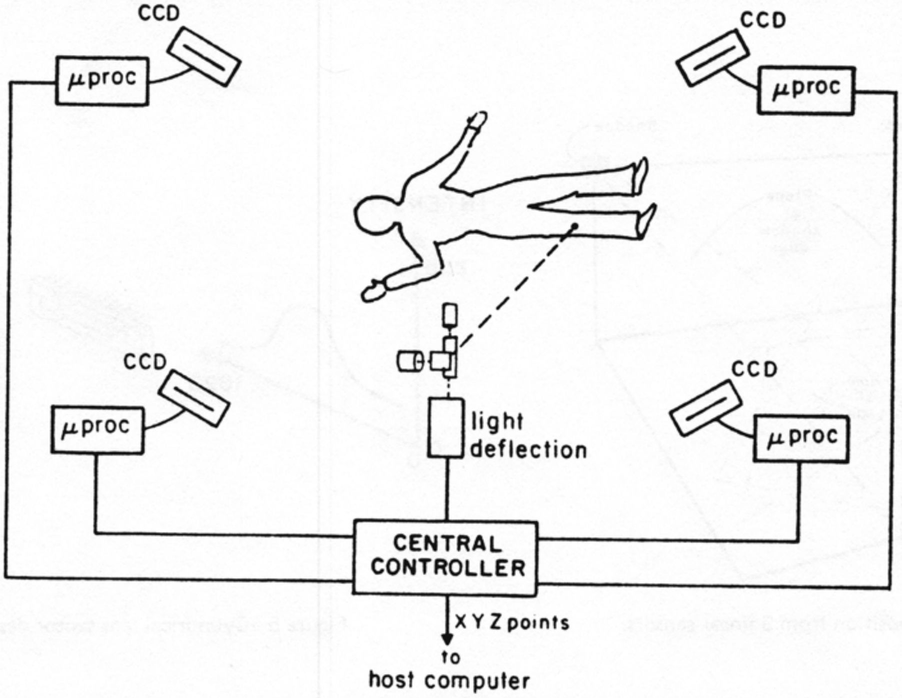


Figure 7 - Distributed system interconnection.

We next describe some of our recent results as to how data points from such a device may be used to construct and object's surface description.

Optimal Surface Construction from Limited Data

The problem to be solved here is that of constructing a reasonable approximation of the surface of the objects being studied, from the set of points on it that have been digitized by the system just described. This problem of fitting a surface to some points crops up in a wide variety of areas, from oceanography to microbiology (see e.g., Keppel (1975) and Gold, Charters, and Ramsden (1977)). The problem, however, is exceedingly difficult perhaps even to define properly let alone to solve in the general case. The variation we have been working on is one in which the surface points are presented not as a random set, but rather as a distribution on a finite set of planes not intersecting in the environment. This situation will naturally occur if the object is scanned in some rasterlike fashion (Figure 8). Further, a solution to this variance of the general problem is widely applicable to the many applications which represent a 3-D object by a set of contours on parallel planes, e.g., contour maps of geographic structures, small biological structures cut into thin slices. Our solution then is not limited to data acquired by our digitizing system but is effective for any such objects presented as a series of contours (Figure 9).

The algorithm we have developed (Fuchs, Kedem, and Uselton (1977)) will always find that surface (composed of triangles) which contains all the data points and which has the minimum total surface area. (Technically, it is possible that more than one such interpolating surface will have precisely the same minimum area. In such unusual cases we presumably don't care which of these optimal surfaces is used; the algorithm can return one or all of them. For clarity of presentation, however, we'll refer in the text to «the» optimal surface). This minimum interpolating surface criterion is roughly analogous to a sheet being tied down to all the points and pulled as tightly as possible.

The method is based on determining the optimal surface between each pair of consecutive contours. A surface between two such contours is simply a set of triangles, each with two vertices being points on one coutour and the third vertex a point on the other contour. The three sides of such a triangle will be referred to as the «left span», «right span», and «contour segment», as illustrated in Figure 10. In order for this set of triangles to form an «acceptable» surface, the surface must contain no «gaps» or «overlaps» (Figure 11).

This implies that the set of triangles has to satisfy the following conditions :

1) - If a span is used as the left edge of a triangle in the set, it also has to be used as a right edge of some triangle in the set.
2) - Each contour segment of each contour has to be used in exactly one triangle in the set.

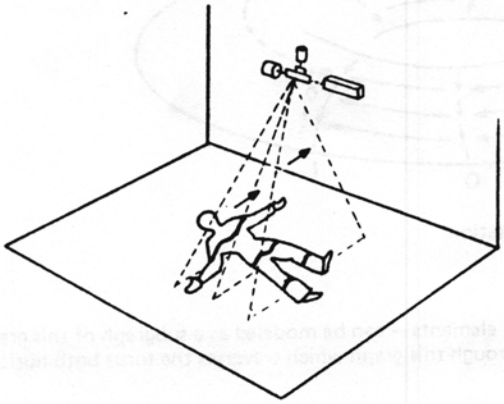


Figure 8 - Raster scan of object.

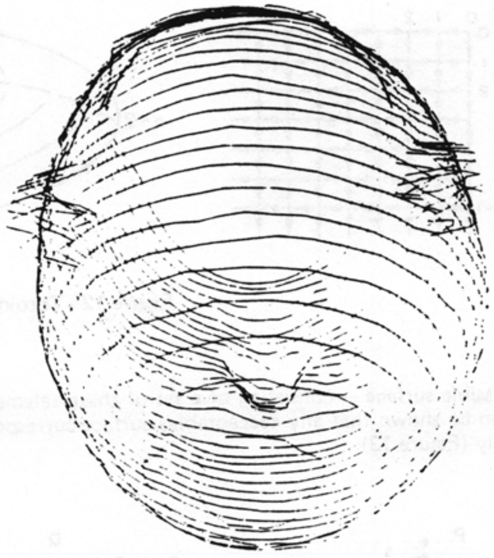


Figure 9 - Contour data of human head.

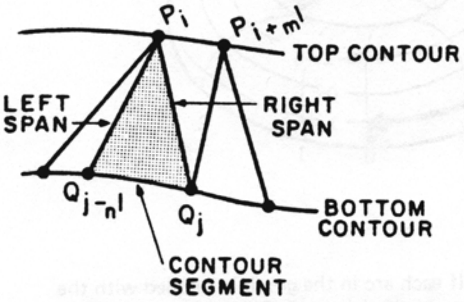


Figure 10 - Surfacing tiles.

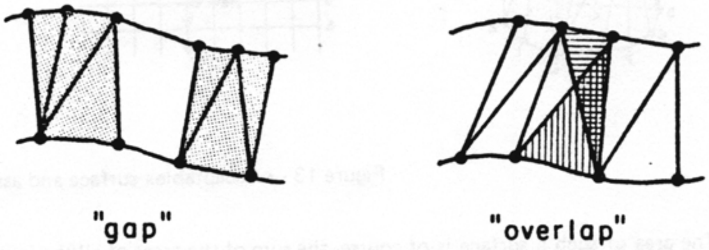


Figure 11 - Non-acceptable surfacings.

The search for the optimal set of triangles can be conveniently modeled by the following directed graph, $G = < V,A >$, where V is the set of vertices and A the set of arcs (for graph-theoretic terms see e.g., Harary (1969)).

Formally, let

$P_0, P_1, \ldots\ldots, P_{m-1}$ be the sequence of points on the top contour,

$Q_0, Q_1, \ldots., Q_{n-1}$ be the sequence of points on the bottom contour,

then : $V = [\, v_{ij} \;|\; i = 0, 1, \ldots., m - 1;\; j = 0, 1, \ldots., n - 1 \,]$, and

$$A = [\, <_{k\ell}, v_{st} > \;|\; \text{either } s = k \text{ and } t = (\ell + 1) \bmod n$$

$$\text{or } s = (k + 1) \bmod m \text{ and } t = \ell \,]$$

Vertex V_{ij} corresponds to the span between points P_i et Q_j, and the arc $<v_{k\ell}\ v_{st}>$ corresponds to the triangle with the left span P_kQ_l and the right span P_sQ_t.

It is easy to see that this is a toroidal graph — i.e., one which lies on the surface of a torus (Figure 12).

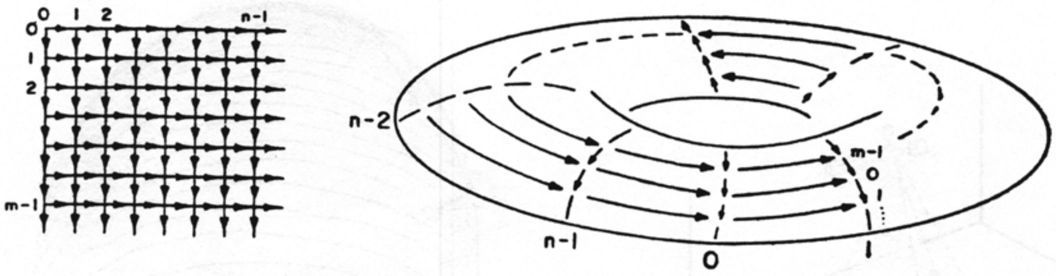


Figure 12 - Toroidal graph representation.

Any possible surface — consisting of a set of these «elementary» triangular elements — can be modeled as a subgraph of this graph. Specifically it can be shown that any «acceptable» surface corresponds to a cycle through this graph which traverses the torus both horizontally and vertically (Figure 13).

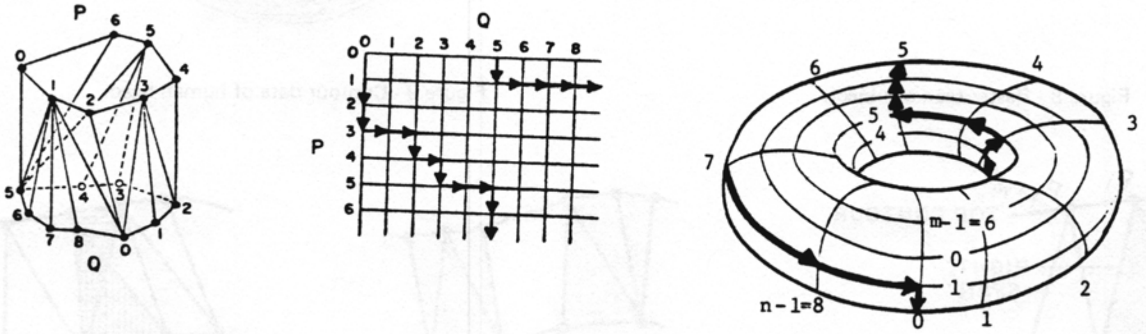


Figure 13 - «Acceptable» surface and associated cycle.

The area of such a surface is of course, the sum of the areas of all the involved triangles. If each arc in the graph is labelled with the area of its associated triangular tile, then the area of a cycle's associated surface is simply the sum of the labels of the involved arcs. Thus search for the optimal surface is reduced to finding the minimum traversing cycle. The problem of determing this can be shown to be a special case of finding shortest paths (Kedem and Fuchs (1977), Johnson (1977)). The required cycle can then be found in approximately 2 (m.n. log (min m,n)) steps. Figure 14 demonstrates the result of applying the surfacing algorithm to the data of Figure 9.

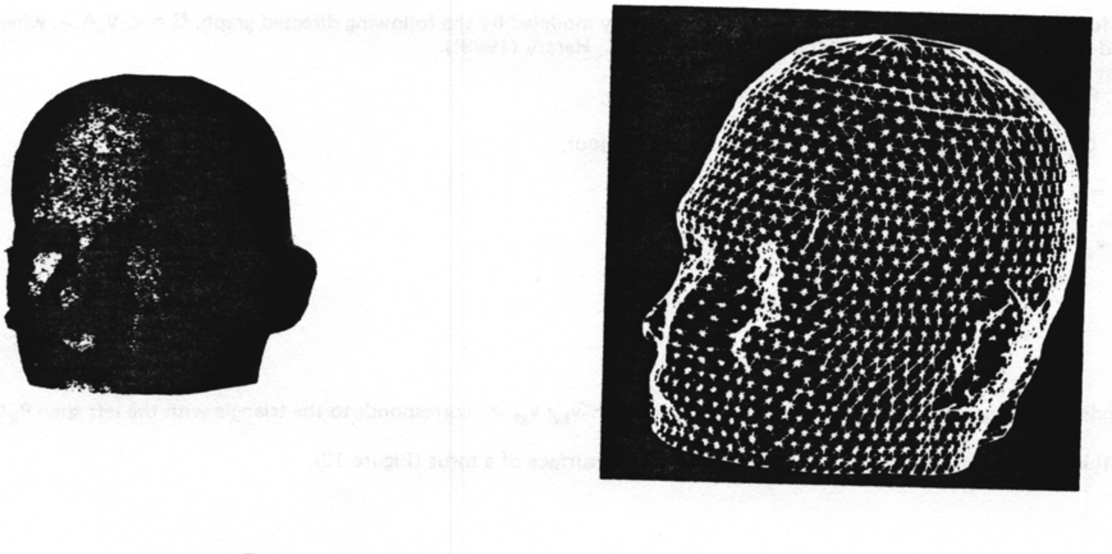


Figure 14 - Reconstructed surface of human head data.

We are currently extending and generalizing this surfacing algorithm to objects the data for which are only available in a less structured form than a sequence of contours.

Real-time Video Head-mounted Display and Interaction

Acquisition of 3-D surface points and construction of surface object definitions naturally leads to questions about interaction with this 3-D data, both the initial monitoring of the digitizer and the surface reconstruction algorithm and also for a variety of subsequent applications. We are currently developing a flexible 3-D viewing and interaction system for these purposes. The system is based on the head-mounted display idea introduced by Sutherland (1968) and developed further by Vickers (1974). Our design involves a small TV receiver mounted on each side of the helmet, with optics and half-silvered mirrors enabling each eye to see both the room environment as well as its assigned TV image (Figure 15). Placing three light sources on top of the helmet will allow our digitizing system to track the helmet's position and orientation in the room. From this information the system can precisely determine, for the person wearing the helmet, the location and viewing direction for each eye. It can then generate the appropriate image, for each eye, of the simulated objects in the environment. The half-silvered mirrors will cause these simulated images to appear superimposed onto the actual scene, floating in space in a ghost-like fashion. Further, if the user moves (and thus the helmet moves), the system will modify the images so that the simulated objects (if so desired) will appear stationary in the environment, enabling the user to walk around them or even to go inside them.

Since the video images can be transmitted via standard broadcast methods and the light sources can easily be sequenced under radio control, there will be *no* physical connections whatsoever between the helmet and the rest of the system. The resulting freedom of movement should make the entire system much more convenient to use than any previous attempts at this kind of solution. Further, the sequencer mounted on top of the helmet to control the three tracking lights can easily control several more of them; these additional lights can be hand-held like wands or mounted on fingertips and connected via thin wire to the helmet. They can then be used for a variety of interactive positioning and pointing tasks. As one example, the head-mounted display could be used to monitor and guide the progress of the digitizing system. As points are measured by it, they could be displayed on the TV's worn by the observer viewing the scene (Figure 16).

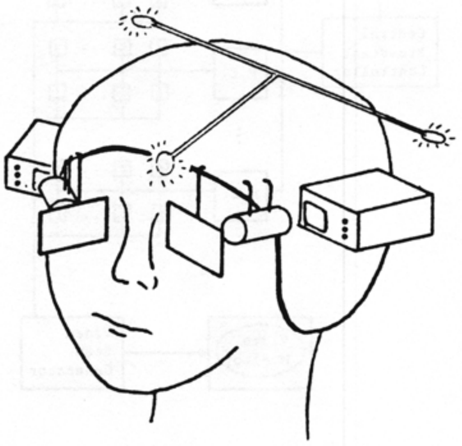


Figure 15 - Video head-mounted display.

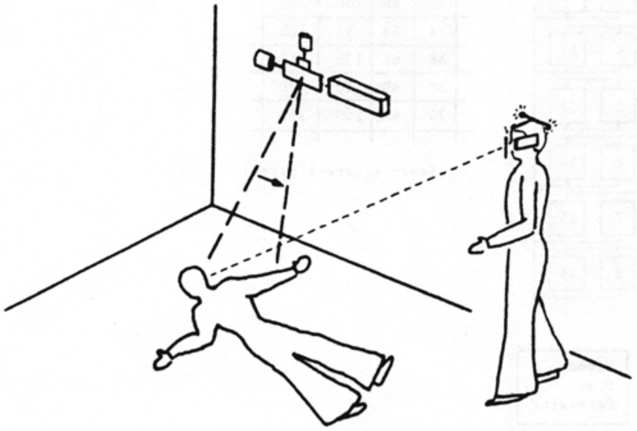


Figure 16 - Head mounted display monitoring digitization.

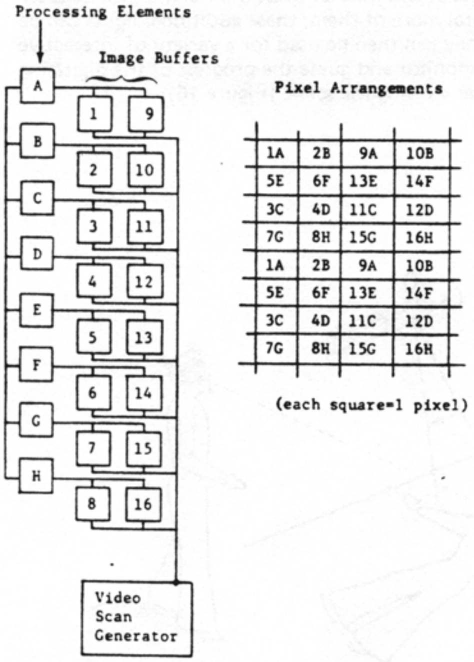
It would be natural to display these points in the environment coordinate system in their original locations; this would result in their appearing on the model's surface and thus any errors would be readily apparent. The observer could then easily point them out to the system with his «wand». The system could then attempt to re-digitize the region in question.

Similar monitoring could also be achieved for the surface reconstruction algorithm. Assuming the model to be relatively stationary, the observer could readily compare the reconstructed surface mesh with the model's and indicate any regions, for instance, where the disparity was unacceptable. (The variety of other uses of the system, for interactive design, etc., will not be discussed here).

Rapid Generation of Continuous-tone Images

In order for the head-mounted display system to be useful, the graphic system generating the images needs to be able to generate them in «real-time» in order for there to be no perceptable delay between the time the user moves his head and the new image is drawn on the TV's. This implies that each scan of the TV screen (1/30 sec.) needs (possibly) to have a new image. Such rapid processing is possible for line-drawing images, but is today much too costly (approx. S 500.000 for most users. We have just begun to look at this problem. We sketch here some of our preliminary results. The crucial difference between generation of line-drawing, «wire frame» images and the more realistic images of solid objects, composed of planar polygonal surfaces such as triangular tiles from our surfacing algorithm, is that for solid objects a visibility/obstruction calculation has to be performed at each pixel (picture element) to determine which of the surfaces is in front and therefore visible and obstructing all other surfaces at this pixel. Our solution, described in Fuchs (1977) and Fuchs and Johnson (1978), is to distribute these calculations among as many (micro-) processors as can be afforded for the system. The video image is also physically distributed, among a set of image memory modules. The distributed image is interlaced, both in the X and Y coordinates, so that any contiguous region on the screen such as one covered by a polygonal surface title (e.g., one side of a simulated box) would affect pixels in many different image memory units. These memory units are divided among the processing elements — the more processing elements, the fewer units assigned to each (Figure 17). All the processing elements are connected to a central controller which, after performing the geometric transformations on each surface tile, broadcasts its coordinates to all the processors. Each processor, knowing which memory units and thus which pixels are under its control, determines whether any of its pixels are affected by this particular tile and for each such pixel determines the tile's distance from the viewer. For each pixel in the image buffer there is an associated buffer (called a «Z» or a depth buffer) in which is kept the distance of the closest tile so far encountered for this pixel (Figure 18). At any time, if the current tile's Z value is larger than the one in the buffer, then the current tile is behind some already encountered tile and thus the processing element ceases to consider this tile for this pixel. On the other hand, if the current tile's Z value for this pixel is less than the one in that pixel's Z buffer then the current tile is closer and thus its new Z value is placed in the Z buffer and its color/ intensity value is put in the image buffer at this pixel.

The speed of the overall system is determined by the number of memory units (i.e., the resolution of the image), the number of processing elements, and the number of tiles in the simulated objects. We expect that such realistic simulation of solid objects will considerably enhance the utility of the system.



1A	2B	9A	10B
5E	6F	13E	14F
3C	4D	11C	12D
7G	8H	15G	16H
1A	2B	9A	10B
5E	6F	13E	14F
3C	4D	11C	12D
7G	8H	15G	16H

Figure 17 - Processor/image distribution and image interlacing.

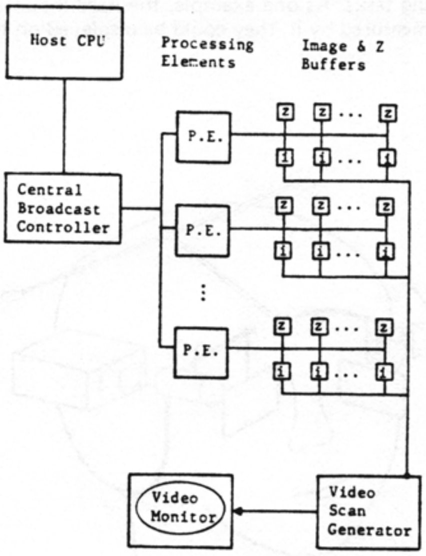


Figure 18 - Distributed system architecture with «Z» buffers.

Implementation

At present only isolated parts of the system have been completed. Four prototype sensing units have been constructed and are about to be tested, and the surfacing algorithm is running and is being enhanced. We have just begun detailed design of the head mounted display. The basic design of the distributed multiprocessor display system has been bread-boarded but a final design has yet to be constructed.

Summary

The system described above will allow any user, even one not familiar with computers, to digitize a human body or other 3-D object, view the digitization results in the actual room environment, and interact with the system via simple hand movements. We hope the completed system will become useful as a flexible, interactive, three-dimensional technical and creative medium.

Acknowledgment

We wish to thank Professor R.E. Herron of Baylor University for providing the data for figures 9 and 14.

References

1 - Burton R.P. and Sutherland I.E., *Twinkle Box-A Three Dimensional Computer Input Device.* National Computer Conf., AFIPS Proceedings, Vol.43, 1974.

2 - Fuchs H., *Distributing a Visible Surface Algorithm Over Multiple Processors.* Proc. 1977 ACM Annual Conf.

3 - Fuchs H., Duran J. and Johnson B., *A System for Automatic Acquisition of Three-dimensional Data.* National Computer Conf., AFIPS Proceedings, Vol.46, 1977.

4 - Fuchs H. and Johnson B., *A Multi-Microprocessor System for Video Graphics.* Tech. Rep. MMS-31, Math. Sciences, University of Texas at Dallas (in revision), 1977.

5 - Fuchs H., Kedem Z.M. and Uselton S.P., *Optimal Surface Reconstruction From Planar Contours.* Comm. A.C.M., Vol. 20, No.10., 1977.

6 - Gold C.M., Charters T.D. and Ramsden J., *Automated Contour Mapping Using Triangular Element Data Structures and an Interpolant Over Each Irregular Triangular Domain.* Proc. of ACM-SIGGRAPH Fourth Annual Conf. on Comp. Graphics and Interactive Techniques, 1977.

7 - Harary F., *Graph Theory.* Addison-Wesley, Reading, Mass., 1969.

8 - Johnson D.B., *Efficient Algorithms for Shortest Paths in Sparse Networks,* J.ACM, Vol.24, No.1, Jan.1977.

9 - Kedem Z.M. and Fuchs H., *A Fast Method for Finding Several Shortest Paths in Certain Graphs,* Tech. Rep. MMS-13, Math. Sciences, University of Texas at Dallas, 1977.

10- Keppel E., *Approximating Convex Surfaces by Triangulation,* IBM-J. of Research and Development, Vol.19, No.1, 1975.

11- Sutherland I.E., *A Head-Mounted Three-Dimensional Display.* Fall Joint Comp. Conf. AFIPS Proceedings, Vol.33, 1968.

12- Vickers D.L., *Sorcerer's Apprentice : Head-Mounted Display and Wand,* Ph. D. Th., Comp. Sci., U. of Utah (UTEC-C. Sc.–74–048), 1974.