THE UNIVERSITY OF MICHIGAN

COMPUTING RESEARCH LABORATORY

SHAPE MATCHING IN PYRAMIDS

JANUARY 1984

Room 1079, East Engineering Building
Ann Arbor, Michigan 48109
USA
Tel: (313) 763-8000

SHAPE MATCHING IN PYRAMIDS

William I. Grosky
Intelligent Systems Laboratory
Computer Science Department
Wayne State University
Detroit, Michigan 48202

and

Ramesh Jain
Department of Electrical and Computer Engineering
University of Michigan
Ann Arbor, Michigan 48109

Abstract

This paper presents a technique for region matching. An image is represented as a function $f(X,Y)$ and a region in an image is approximated by an elliptical paraboloid. The coefficients of the elliptical paraboloid are used in matching regions. It is shown that the coefficients can be easily computed in a pyramid type architecture using proposed pyramid linking algorithms. Our experiments demonstrate that the proposed approach is more robust than moment-based approaches for region matching. Another feature of the approach is that it can be used for region splitting and merging, if required.

1. INTRODUCTION

Matching plays a vital role in many applications of image understanding. At a low level, token matching, commonly called correspondence, is used in stereo and structure from motion; object recognition using structural information uses graph isomorphism at a high level. The tracking of objects and change detection are required in many applications. Cross-correlation and statistical techniques have been used for matching regions in such situations. If the segmentation of images has been already performed, then correlation has been suggested as a particularly powerful technique for matching and displacement. A serious limitation of correlation, however, is that it does not work well in the case of rotation.

In many applications, object recognition is performed based only on the regions, or borders of regions, representing the mask of the object. Usually, several masks for each object are stored and recognition is achieved utilizing matching of the unknown region with each model. In tracking, also, a region for an object is given, or obtained in a frame, and then unknown regions are matched to it.

Assuming segmentation has been already accomplished, many of the most wide used techniques for the shape description of regions are moment-based [5,7,10,15,16,21,24,25]. These methods extract features which are invariant with respect to the object's position, size, and orientation. The initial impetus in this area was the work of Hu[16] in which 7 algebraic invariants, calculated from n-th order moments of the image, for $0 \leq n \leq 3$, are described.

In this paper, we introduce a least-squares based technique which computes various features, and compare it with that of Hu[16]. We interpret an image as function $f(X,Y)$ which maps X and Y coordinates to graylevels. This allows us to consider an image, or a region representing a sub-image, as a surface $Z=f(X,Y)$. A region in a frame is represented by finding a best-fitting elliptical

parabol to the graylevel surface corresponding to the region. It is observed that this representation allows the reliable matching of regions, and by-products of the matching are the parameters describing the 2-D transformation of the surfaces. We discuss a method to fit the elliptical paraboloid to the regions and demonstrate the efficacy of this representation. We also show how to embed the calculations in a pyramid architecture [1,4,8,9,22,23] so that the representation of each region in an image is at the highest possible level in the pyramid structure. It is shown that the parameters of the best-fit surfaces can be computed in the pyramid structure while forming the connected components. Thus, by using a special architecture, it is possible to compute the parameters of the graylevel surfaces very quickly, so that the matching of the surfaces can be performed without going to the raw images.

Our method is more powerful than moment-based approaches for several reasons. Moment-based techniques lose all contact with the underlying graylevel surface. They have no notion of the errors involved in representing a segment by a set of real numbers. Thus, there cannot be any embedded control mechanism which uses the error to invoke numerous possible split-and-merge techniques to compute less errorful features. Our technique, on the other hand, has a built-in notion of error; namely, the difference in volume between the actual graylevel surface and the fitted one. Thus, we may use various split-and-merge techniques to lessen the error if we so desire. This paper discusses just the overall efficacy of our approach, so we do not cover this topic, but in future reports such techniques will become extremely important.

Our technique also generalizes to 3-dimensions in that we may parametrize coefficients of our quadric surfaces in a similar manner as we are here parametrizing the graylevel values. Many views of a 3-D object will then be representable in a very compact manner as a k-dimensional subspace of an m-dimensional space, for $k<m$, while a single view would be a point in this m-

dimensional space. We may then use existing techniques to efficiently find the best-match to an arbitrary view of some unknown object. Moment-based techniques, on the other hand, cannot be generalized in such a direct fashion to handle this problem.

Finally, as we shall later demonstrate, our method produces better results. We have performed numerous experiments where various images were translated, rotated, underwent size changes, had noise introduced, and underwent changes in lighting. These experiments show that our approach produces better overall clusters than the moment-based ones.

Fitting a parametrized surface to a graylevel surface has been done by other researchers[11,12,13,19]. They, however, do so for reasons other than ours, which causes their methods to be quite dissimilar to ours. Paton[19] uses Legendre polynomial while Haralick and his co-workers use planes [11] and bicubic surfaces [12,13]. These fittings are done locally, however. For each pixel p, a fit is done over some small neighborhood of p. These researchers are trying to discover local features such as peaks, pits, ridges, ravines, saddles, flats, and hillsides in order to use them as a means for describing an image. Ostensibly, these features can be used for matching purposes. Their use in such a setting has not been explored, however. Also, their robustness under various levels of noise has not been examined. Our method, on the other hand, works quite well under noise. Since matching is our primary goal, we set out to discover global rather than local features, and global features should be more robust under noise than local features. Thus, we fit a single surface to the entire region, rather than many surfaces locally.

An overview of our scheme now follows. We interpret an image as a surface $Z=f(X,Y)$ in 3-space. Each node of the pyramid will then contain the equation of a second-order surface, an elliptic paraboloid, which is fit through the center-

of-mass of the level-1 sub-image corresponding to this node and which minimizes the square-error between itself and the surface $Z=f(X,Y)$ over an appropriate area. This will be discussed more fully in Section II.

For each region in the image, we now want to find the highest node in the pyramid, that is, that node closest to the apex, whose corresponding level-1 sub-image completely contains this region and contains no other region. If we can find such a node, the equation contained in it will convey information regarding the shape of the region, its position and its orientation. If we cannot find such a node, we change some of the links of the pyramid so that a level-k node, $k>1$, now determines a not necessarily square region. This connected region will consist of squares of sizes $2^{k-1} \times 2^{k-1}$, $2^{k-2} \times 2^{k-2}$, ..., $2^0 \times 2^0$ which are joined together across their edges. This will be discussed in Section III. See [2,14,17,18,20] for other applications of pyramid relinking.

The embedding of the above calculations in the pyramid data structure will be discussed in Section IV.

In Section V, we will discuss the matching phase of our algorithm. This will consist of constructing pyramids for each frame. In each pyramid, we identify nodes corresponding to each region, as above. Two nodes whose equations are equivalent after a coordinate translation and rotation will correspond to the same region. This translation and rotation will then give information regarding the translational and rotational components of the 2-D orientation, respectively, of this region.

Finally, the various experiments we performed, as well as our conclusions, will be discussed in section VI.

2. SECOND-ORDER IMAGE APPROXIMATION

For illustration, let us assume an 8 x 8 grid of pixels. Our coordinate origin will be placed in the exact center of the grid. Thus, the point (i,j) will be in the lower left-hand corner of the pixel in row $4-j$ and column $i+5$. This pixel will also have address (i,j). See Figure 1. Let us assume that this grid contains only 1 surface (4-connected region). Now, $Z=A(X-X_0)^2+B(Y-Y_0)^2+C(X-X_0)*(Y-Y_0)+D$ is the equation of an elliptic paraboloid centered at point (X_0,Y_0). Let us find the values of A,B,C, and D which minimizes the square-error between this surface and the surface $Z=f(X,Y)$ over a circular region centered at (X_0,Y_0) and with radius R larger than the diameter of any possible region in our grid. For our example, any $R>8\sqrt{2}$ will suffice. (Note that $f(X,Y)$ is assumed to be 0 outside the grid.) We have experimented with fixed values of R as well as variable values. In the latter case, R is taken to be the maximal distance from (X_0,Y_0) to a boundary point.

We are going to put $X_0=X_{CM}$ and $Y_0=Y_{CM}$, the coordinates of the center-of-mass of the region. The reason for centering this surface at the center-of-mass and fitting it over a circular region is that when this is done, the position and orientation of the surface in the grid will not affect the final equation after a coordinate transformation is done which puts the origin over (X_{CM},Y_{CM}) and the axes in such an orientation as to eliminate the cross-product term.

Letting $x=X-X_0$ and $y=Y-Y_0$, we want to minimize expression (1)

$$\int_{-R}^{+R}\int_{-\sqrt{R^2-y^2}}^{+\sqrt{R^2-y^2}} [Ax^2+By^2+Cxy+D-f(x+X_0,y+Y_0)]^2 dxdy \qquad (1)$$

which is equivalent to expression (2):

$$\int_{-R}^{+R}\int_{-\sqrt{R^2-y^2}}^{\sqrt{R^2-y^2}} [A^2x^4+B^2y^4+C^2x^2y^2+2ABx^2y^2+2ACx^3y+2Bxy^3 \qquad (2)$$
$$+2ADx^2+2CDxy+2BDy^2+D^2+f^2(x+X_0,y+Y_0)-2Df(x+X_0,y+Y_0)$$
$$-2Ax^2f(x+X_0,y+Y_0)-2By^2f(x+X_0,y+Y_0)-2Cxyf(x+X_0,y+Y_0)]dxdy$$

We finally end up minimizing

$$\int_{-R}^{+R}\int_{-\sqrt{R^2-y^2}}^{\sqrt{R^2-y^2}} f^2(x+X_0,y+Y_0)-(2Ax^2+2By^2+2Cxy+2D)\,f(x+X_0,y+Y_0)]dxdy \quad (3)$$

$$+(\frac{\pi R^6}{8})(A^2+B^2)+(\frac{\pi R^6}{24})(C^2+2AB)+(\frac{\pi R^4}{2})D(A+B)+\pi R^2D^2$$

Setting the partial derivatives of expression (3) with respect to A,B,C, and D equal to 0, we end up with equations (4a)-(4d):

$$(\frac{\pi R^6}{4})A+(\frac{\pi R^6}{12})B+(\frac{\pi R^4}{2})D=2\int_{-R}^{+R}\int_{-\sqrt{R^2-y^2}}^{+\sqrt{R^2-y^2}} x^2 f(x+X_0,y+Y_0)dxdy \quad (4a)$$

$$(\frac{\pi R^6}{12})A+(\frac{\pi R^6}{4})B+(\frac{\pi R^4}{2})D=2\int_{-R}^{+R}\int_{-\sqrt{R^2-y^2}}^{+\sqrt{R^2-y^2}} y^2 f(x+X_0,y+Y_0)dxdy \quad (4b)$$

$$(\frac{\pi R^6}{12})C = 2\int_{-R}^{+R}\int_{-\sqrt{R^2-y^2}}^{+\sqrt{R^2-y^2}} xyf(x+X_0,y+Y_0)dxdy \quad (4c)$$

$$(\frac{\pi R^4}{2})A+(\frac{\pi R^4}{2})B+2\pi R^2D = 2\int_{-R}^{+R}\int_{-\sqrt{R^2-y^2}}^{+\sqrt{R^2-y^2}} f(x+X_0,y+Y_0)dxdy \quad (4d)$$

with solutions (5a)-(5d):

$$A = (\frac{6}{\pi R^6})\int_{-R}^{+R}\int_{-\sqrt{R^2-y^2}}^{+\sqrt{R^2-y^2}}(y^2+3x^2-R^2)f(x+X_0,y+Y_0)dxdy \quad (5a)$$

$$B = (\frac{6}{\pi R^6})\int_{-R}^{+R}\int_{-\sqrt{R^2-y^2}}^{+\sqrt{R^2-y^2}}(x^2+3y^2-R^2)f(x+X_0,y+Y_0)dxdy \quad (5b)$$

$$C = (\frac{24}{\pi R^6})\int_{-R}^{+R}\int_{-\sqrt{R^2-y^2}}^{+\sqrt{R^2-y^2}} xyf(x+X_0,y+Y_0)dxdy \quad (5c)$$

$$D = \int_{-R}^{+R}\int_{-\sqrt{R^2-y^2}}^{+\sqrt{R^2-y^2}} [\frac{4}{\pi R^2}-\frac{6}{\pi R^4}(x^2+y^2)]f(x+X_0,y+Y_0)dxdy \quad (5d)$$

From Theorem 7.8 in Apostol[3], it may easily be verified that the above solutions do minimize (1).

Changing back to (X,Y) coordinates, letting $W=R^2-(Y-Y_{CM})^2$, and putting $X_0=X_{CM}$ and $Y_0=Y_{CM}$, where

$$X_{CM}=\frac{\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} Xf(X,Y)dXdY}{\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} f(X,Y)dXdY}$$

and

$$Y_{CM} = \frac{\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} Yf(X,Y)dXdY}{\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} f(X,Y)dXdY},$$

we have

$$A = \left(\frac{6}{\pi R^6}\right)Y2MEAN + \left(\frac{18}{\pi R^6}\right)X2MEAN - \left(\frac{6}{\pi R^4}\right)MEAN \tag{6a}$$

$$B = \left(\frac{6}{\pi R^6}\right)X2MEAN + \left(\frac{18}{\pi R^6}\right)Y2MEAN - \left(\frac{6}{\pi R^4}\right)MEAN \tag{6b}$$

$$C = \left(\frac{24}{\pi R^6}\right)XYMEAN \tag{6c}$$

$$D = \left(\frac{4}{\pi R^2}\right)MEAN - \left(\frac{6}{\pi R^4}\right)[X2MEAN + Y2MEAN] \tag{6d}$$

for

$$MEAN = \int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} f(X,Y)dXdY \tag{7a}$$

$$X2MEAN = \int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} X^2 f(X,Y)dXdY \tag{7b}$$

$$- \frac{\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} [Xf(X,Y)dXdY]^2}{MEAN}$$

$$Y2MEAN = \int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} Y^2 f(X,Y)dXdY \tag{7c}$$

$$- \frac{\left[\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} Yf(X,Y)dXdY\right]^2}{MEAN}$$

$$XYMEAN = \int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} XYf(X,Y)dXdY \tag{7d}$$

$$- \frac{\left[\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} Xf(X,Y)dXdY\right]\left[\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} Yf(X,Y)dXdY\right]}{MEAN}$$

Thus, to compute the appropriate minimal surface, we need only compute the zero-th, first and second order moments of the image. These moments can easily be calculated in terms of the graylevel value at each pixel, since, for i,j integers, $f(i,j)=f(i+a,j+b)$ for $0\le a,b<1$. Specifically, it can be shown that,

$$\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} f(X,Y)dXdY = \sum_{i,j} f(i,j) \tag{8a}$$

$$\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} Xf(X,Y)dXdY = \sum_{i,j}(i+\frac{1}{2})f(i,j) \tag{8b}$$

$$\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} Yf(X,Y)dXdY = \sum_{i,j}(j+\frac{1}{2})f(i,j) \tag{8c}$$

$$\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} X^2f(X,Y)dXdY = \sum_{i,j}(i^2+i+\frac{1}{3})f(i,j) \tag{8d}$$

$$\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} Y^2f(X,Y)dXdY = \sum_{i,j}(j^2+j+\frac{1}{3})f(i,j) \tag{8e}$$

$$\int_{Y_{CM}-R}^{Y_{CM}+R}\int_{X_{CM}-W}^{X_{CM}+W} XYf(X,Y)dXdY = \sum_{i,j}(i+\frac{1}{2})(j+\frac{1}{2})f(i,j) \tag{8f}$$

Thus, to develop the equation of the best fitting elliptic paraboloid to a region, we need only be able to efficiently calculate

$$\sum_{i,j} f(i,j) \tag{9a}$$

$$\sum_{i,j} if(i,j) \tag{9b}$$

$$\sum_{i,j} jf(i,j) \tag{9c}$$

$$\sum_{i,j} i^2(i,j) \tag{9d}$$

$$\sum_{i,j} j^2(i,j) \tag{9e}$$

$$\sum_{i,j} ijf(i,j) \tag{9f}$$

for this region.

3. SEGMENTING AN IMAGE VIA PYRAMID LINKING

Consider the image of Figure 2. If one constructs a standard pyramid in which each node above level 1 has precisely 4 sons, one cannot find any node in this pyramid whose corresponding sub-image contains one and only one region. We thus relax this requirement. Each node can have as few as 1 and as many as 16 sons. The way this is done is that, as necessary, nodes are relinked to neighboring fathers. Given the 2 x 2 template of Figure 3, area 1 can be linked to either its own father or to a neighbor of this father to the west, to the north or

to the northwest. Similar statements can be made for areas 2, 3, or 4.

We first label the regions on level 1. We use the standard recursive labelling scheme, but it may be done in parallel by having each pixel with a graylevel above 0 pass its address to its northern, southern, eastern and western neighbors. When a pixel with a 0 graylevel receives inputs, it ignores them. When a pixel with a non-zero graylevel receives inputs, it changes its address to the minimal address received. Thus, after a time proportional to the diameter of a region, that region is labelled by the smallest address of any pixel in that region. Then, starting at level 1 and continuing as long as necessary we do the following. Each 2 x 2 sub-image is examined to see how many region labels are included among its nodes. If this sub-image contains nodes from at most 1 region, none of its nodes are relinked to another another father, while if it contains nodes from 2, 3, or 4 different regions, these nodes are relinked, if possible. This relinking is done in a disciplined fashion:

1) If 3 nodes are from 1 region, while the 4-th node is from a different region, this latter node is the one to be relinked.

2) If 2 neighboring (4-connected) nodes are from one region, these 2 nodes are relinked to the same father.

3) Relinking is only done to a father all of whose sons are either of graylevel 0 or from the same region.

4) We first try to relink nodes to a father whose sons contain the same region as the to be relinked node. If this can't be done, we then try to relink them to a father all of whose sons are of graylevel 0.

5) After all the relinking is accomplished, each son of a given father has a 0 region label or an equal positive region label. The father inher region label.

6) Relinking is not done to a node if there is no neighboring node of the same region label. If this is the case, we have found a node in the pyramid whose corresponding sub-image completely contains just 1 region. The information

concerning this region will be kept at this node and a flag will be set indicating that this node is the root node of an individual connected region, but its region label is changed to 0, so that higher level nodes will see it as being of graylevel 0. Thus, higher level nodes will treat this node as background for any future relinking operations, even though there is an entire connected region encompassed by this node. This latter region is called a *hidden region* , and all nodes with such regions are flagged appropriately.

7) Before the relinking starts on level 1, we place a border of graylevel 0 around the original image. In our case, the 8 x 8 image is put in the center of a 16 x 16 image. This is done as it may be necessary to relink into this border.

Let us trace the steps taken using the image of Figure 2. Each of Figures 4-8 illustrates higher levels of the pyramid. Neighboring nodes with similar cross-hatching have the same father node. Also, nodes an R written in them have their root flag set, while nodes with an H written in them have hidden regions included among their children.

4. EMBEDDING THE NECESSARY CALCULATIONS IN THE PYRAMID

We now indicate how to embed the calculations of (9a)-(9f) for each 4-connected region in the pyramid.

Each node of the pyramid 'sees' only a single 4-connected region, aside from any hidden regions which may occur at various children of the given node. Call this region the node's *primary* region. We will develop a method whereby each node of the pyramid calculates (9a)-(9f) for that part of its primary region which it encompasses. We will use recurrence formulas. Specifically, each node will calculate its value of (9a)-(9f) in terms of the values of (9a)-(9f) of its children, after all relinking has been done at that level.

We will illustrate this method on level 3 of the pyramid for the calculation of (9f). Figure 9 illustrates our example. We want to calculate (9f) for node N. This

node has 4 children after all the relinking has been done on level 2: its northwest, southeast and southwest children as well as the southeast child of its western neighbor. Areas p q, r and s are the regions in the image encompassed by these children, respectively. We then have that (9f) for node N is

$$\sum_{i,j \in p} (i-2^0)(j+2^0)fp(i,j)+\sum_{i,j \in q} (i-3*2^0)(j-2^0)fq(i,j) \qquad (10)$$
$$+\sum_{i,j \in r} (i-2^0)(j-2^0)fr(i,j)+\sum_{i,j \in s} (i+2^0)(j-2^0)fs(i,j)$$

where in each region p, q, r or s the origin of the row, column indices i,j is the respective child. Thus, what is pixel (0,0) with respect to the northwest son of node N is pixel (-1,1) with respect to node N. The functions fp, fq, fr and fs are function f with these coordinate transformations.

Now, (10) reduces to

$$\sum_{i,j \in p} ijfp(i,j)+\sum_{i,j \in q} ijfq(i,j)+\sum_{i,j \in r} ijfr(i,j)+\sum_{i,j \in s} ijfs(i,j) \qquad (11)$$
$$+2^0(\sum_{i,j \in p} ifp(i,j)-\sum_{i,j \in q} ifq(i,j)-\sum_{i,j \in r} ifr(i,j)-\sum_{i,j \in s} ifs(i,j))$$
$$+2^0(\sum_{i,j \in p} jfs(i,j)-\sum_{i,j \in q} jfp(i,j)-\sum_{i,j \in r} jfr(i,j)-3\sum_{i,j \in s} jfq(i,j)$$
$$+(2^0)^2(\sum_{i,j \in p} fr(i,j)-\sum_{i,j \in q} fp(i,j)-\sum_{i,j \in r} fs(i,j)+3\sum_{i,j \in s} fq(i,j)$$

Thus, (9f) at node N is computed in terms of (9a), (9b), (9c) and (9f) at the children of node N.

In general, in computing (9a)-(9f) at a level n node, for n1, we use 2^{n-3} instead of 2^0 in (11). Also, the initializations of (9a)-(9f) at pixel (i,j) of level 1 are $f(i,j)$, $\frac{-f(i,j)}{2}$, $\frac{-f(i,j)}{2}$, $\frac{f(i,j)}{4}$, $\frac{f(i,j)}{4}$, and $\frac{f(i,j)}{4}$, respectively.

5. SHAPE MATCHING

In each node of our pyramid which completely encompasses a region, we have the equation $Z=AX^2+BY^2+CXY+D$ of a best-fitting elliptical paraboloid through the center-of-mass of the region. In order to use this surface for match-

ing purposes, we find the angle which the major axis of the ellipse which results from the intersection of the elliptic paraboloid with the (X,Y)-plane makes with the positive X-axis. This angle is restricted to be in the first or fourth quadrant, and is found by calculating through what angle to rotate the (X,Y) axis so that the CXY term is eliminated.

The resultant X^2, Y^2 and constant coefficients will be used in matching regions from frame to frame. The feature we actually use is the eccentricity of the ellipse which results from the interesection of the elliptic paraboloid with the (X,Y)-plane. This feature should remain constant under rotation, translation and scale changes as well as under the addition of a constant graylevel. Once we find similar regions in 2 frames, we use the center-of-mass and the difference in the above angles to find the 2-D translational and rotational components of motion, respectively. We assume that the rotational component is small, as this method does not give unique results for arbitrary rotations. That is, 2 different rotations may lead to the same answer. For example, if in frame 1, the major axis of the above ellipse makes an angle of 30 degrees with the positive X-axis, while in frame 2 the angle is 60 degrees, the rotational component could be -150 degrees, 30 degrees, 210 degrees or 330 degrees.

It is easily derived that if the axes are rotated by RAD radians, for $RAD = .5 * arctan(\frac{C}{(A-B)})$, the CXY term disappears. Also, the equation of the aforementioned ellipse is $AX^{2+BY^2}+CXY+D=0$. Thus, $\frac{dY}{dX} = \frac{-(2AX+CY)}{(2BY+CX)}$ or $\frac{dY}{dX}|_{Y=0} = \frac{-2A}{C}$. This implies that if $sign(A)=sign(C)$ then its major axis is in the 4-th quadrant, while if $sign(A)sign(C)$, its major axis is in the 1-st quadrant. Thus, for $RAD = .5 * arctan(\frac{C}{(A-B)})$, if $RAD<0$ and $sign(A) \neq sign(C)$, we add $\frac{\pi}{2}$ to RAD, while if $RAD>0$ and $sign(A)=sign(C)$, we subtract $\frac{\pi}{2}$ from RAD.

6. EXPERIMENTS AND CONCLUSIONS

To see how well our method performed, we generated data on which we tested how our approach compared to that of the standard moment-based approach of Hu[16]. The data was formulated as follows:

1. We started out with 19 binary images of airplanes. Each of our images was of 64 x 64 resolution.

2. Each of the 19 images was rotated by 0, 15, 30, and 45 degrees, resulting in 76 images.

3. These 76 binary images were transformed to 64-graylevel images by generating random gray-values and smoothing twice with a window size of 3 x 3.

4. 5 ranges of noise were added to each of these 76 images by adding random values to each of the pixel values. These ranges were plus-minus 0, 5, 10, 15, and 20 graylevels.

5. Each graylevel g>0 of each of the 76 graylevel images was transformed to graylevel min(63,g+12).

6. Each of the 76 graylevel regions was reduced to 32 x 32 resolution by averaging once over a 2 x 2 window.

7. Each of the 76 graylevel regions was reduced to 32 x 32 resolution by averaging once over a 2 x 2 window, and then transformed by changing each graylevel g>0 to graylevel max(0,g-4).

We thus had a total of 1520 64-graylevel images, each of resolution 64 x 64. For each image, we calculated the eccentricity of the resulting elliptical paraboloid using both a fixed and a variable radius, as well as the 7 invariants of Hu[16]. Some sample images and their resulting parameters are shown in Figures 10-12.

In order to compare the various approaches, we formed 19 clusters, 1 cluster per airplane, and calculated the eigenvalues of $(SW^{-1}) * SB$, where SW is

the

within-cluster scatter matrix and SB is the *between-cluster scatter matrix* as described in Duda and Hart[6]. The largest eigenvalues were 32.1, 1.3, and 44.0 for Hu's 7 invariants, our method with a fixed radius, and our method with a variable radius, respectively. Note that partitions giving larger eigenvalues are more favorable. Also, note that our method, using a variable radius and only moments of up to second-order, outperforms Hu's method which uses third-order moments.

We have thus demonstrated that our least-squares based method is robust and outperforms moment-based techniques. We have also introduced a pyramid-based architecture in which the computations are done quite rapidly, especially for images containing multiple regions.

We are currently expanding this method to include both general quadric surfaces and error estimation techniques. The latter will be used to control various split-and-merge schemes to be used for images of objects having multiple surfaces.

7. REFERENCES

[1] Adelson, E.H. and Burt, P.J., 'Image Data Compression with the Laplacian Pyramid,' *Proceedings of the Conference on Pattern Recognition* and Image Processing, Dallas, Texas, 1981, pp. 218-223

[2] Antonisse, H.J., 'Image Segmentation in Pyramids,' *Computer* Graphics and Image Processing, Volume 19 (1982), pp. 367-383

[3] Apostol, T., *Mathematical Analysis*, Addison-Wesley, Reading, Massachusetts, 1960

[4] Burt, P.J., *Tree and Pyramid Structures for Coding* Hexagonally Sampled Binary Images, Technical Report TR-814, University of Maryland, College

Park, Maryland, October 1979

[5] Casey, R.G., 'Moment Normalization of Handprinted Characters,' *IBM Journal of Research and Development*, Volume 14 (1970), pp. 548-557

[6] Duda, R.O. and Hart, P.E., *Pattern Classification and Scene* Analysis, Wiley-Interscience, New York, 1973

[7] Dudani, S.A., Breeding, K.J., and McGhee, R.B., 'Aircraft Identification by Moment Invariants,' *IEEE Transactions on Computers* Volume 26 (1977), pp. 39-46

[8] Dyer, C.R. and Rosenfeld, A., *Cellular Pyramids for Image* Analysis, TR-544, Computer Science Department, University of Maryland, Colle May 1977

[9] Dyer, C.R. and Rosenfeld, A., *Cellular Pyramids for Image* Analysis, 2, TR-596, Computer Science Department, University of Maryland, Co November 1977

[10] Hall, E.L., Crawford, W.O., and Roberts R.E., 'Computer Classification of Pneumoconiosis from Radiographs of Coal Workers,' *IEEE* Transactions on Biomedical Engineering, Volume 22 (1975), pp. 518-527

[11] Haralick, R.M., 'Edge and Region Analysis for Digital Image Data,' *Computer Graphics and Image Processing*, Volume 12 (1980), pp. 60-7

[12] Haralick, R.M., 'Ridges and Valleys on Digital Images,' *Computer Vision, Graphics, and Image Processing*, Volume 22 (1983), pp. 28-38

[13] Haralick, R.M., Watson, L.T., and Laffey, T.J., 'The Topographic Primal Sketch,' *The International Journal of Robotics* Research, Volume 2 (1983), pp. 50-72

[14] Hong, T.H., Shneier, M., and Rosenfeld, A., 'Border Extraction Using Linked Edge Pyramids, *IEEE Transactions on Systems, Man, and Cyberneti* Volume 12 (1982), pp. 660-668

[15] Hsia, T.C., 'A Note on Invariant Moments in Image Processing,' *IEEE Transactions on Systems, Man, and Cybernetics*, Volume 11 (1980), pp. 831-834

[16] Hu, M.K., 'Visual Pattern Recognition by Moment Invariants,' *IRE Transactions on Information Theory*, Volume 8 (1962), pp. 179-187

[17] Kasif, S. and Rosenfeld, A., 'Pyramid Linking is a Special Case of ISODATA,' *IEEE Transactions on Systems, Man, and Cybernetics* Volume 13 (1983), pp. 84-85

[18] Narayanan, K.A. and Rosenfeld, A., 'Approximation of Waveforms and Contours by One-Dimensional Pyramid Linking,' *Pattern Recognition* Volume 15 (1982), pp. 389-396

[19] Paton, K., 'Picture Description Using Legendre Polynomials,' *Computer Graphics and Image Processing*, Volume 4 (1975), pp. 40-54

[20] Pietikainen, M. and Rosenfeld, A., 'Image Segmentation by Texture Using Pyramid Node Linking,' *IEEE Transactions on Systems, Man and* Cybernetics, Volume SMC-11 (1981), pp. 822-825

[21] Reeves, A.P. and Rostampour, A., 'Shape Analysis of Segmented Objects Using Moments,' *Proceedings of the Conference on Pattern Recognition* and Image Processing, Dallas, Texas, 1981, pp. 171-174

[22] Tanimoto, S.L., *Towards Hierarchical Cellular Logic: Design Consideration for Pyramid Machines*, TR 81-02-01, Computer Science Department University of Washington, Seattle, Washington, February 1981

[23] Tanimoto, S.L. and Klinger, A., Editors, *Structured Computer* Vision: Machine Perception Through Hierarchical Computation Structures Academic Press, New York, 1980

[24] Wiejak, J.S., 'Moment Invariants in Theory and Practice,' *Image and Vision Computing*, Volume 1 (1983), pp. 79-83

[25] Wong, R.Y. and Hall, E.L., 'Scene Matching with Invariant Moments,' *Computer Graphics and Image Processing*, Volume 8 (1978), pp. 16-24

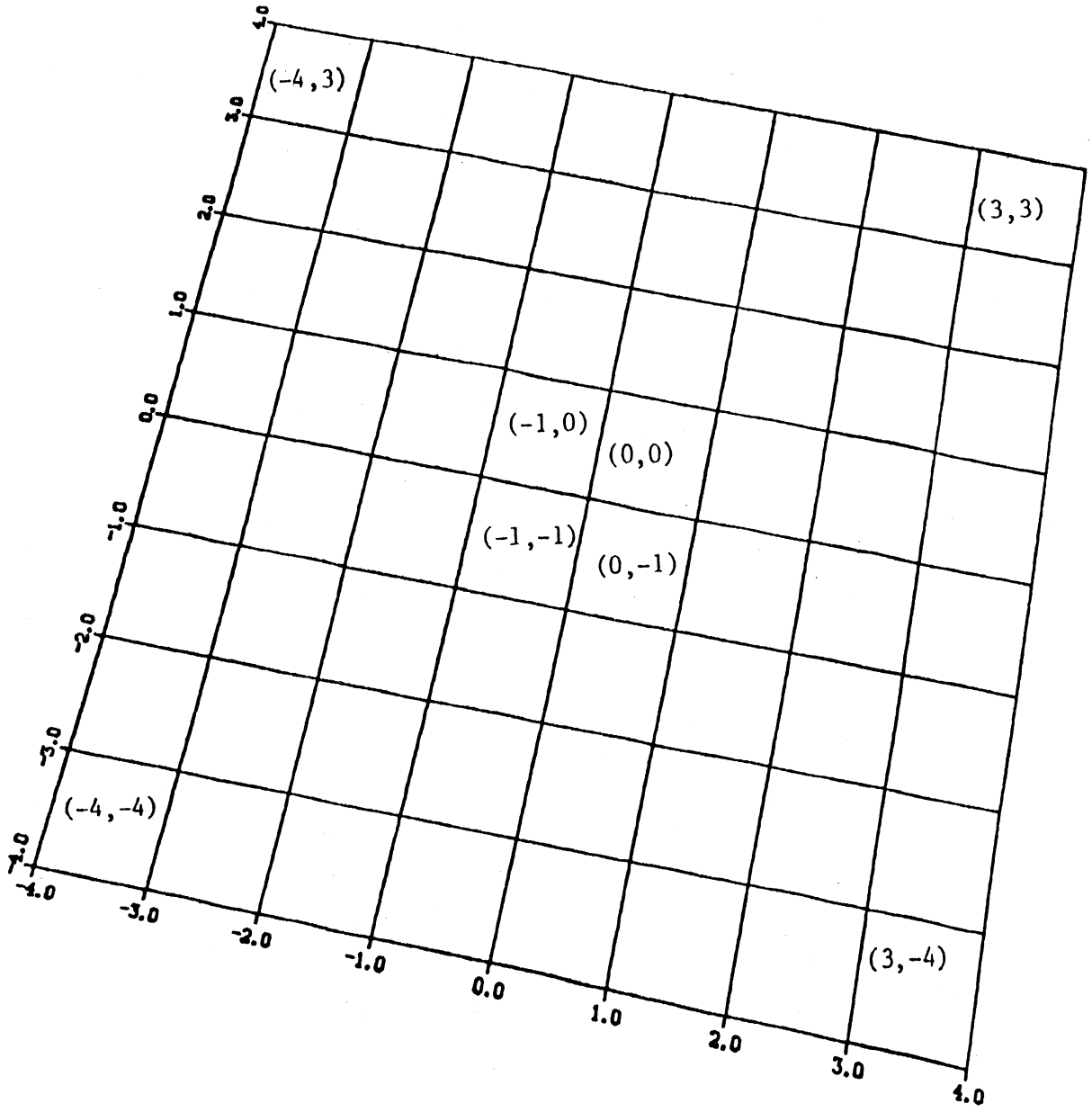


FIGURE 1 - AN 8 x 8 COORDINATE GRID

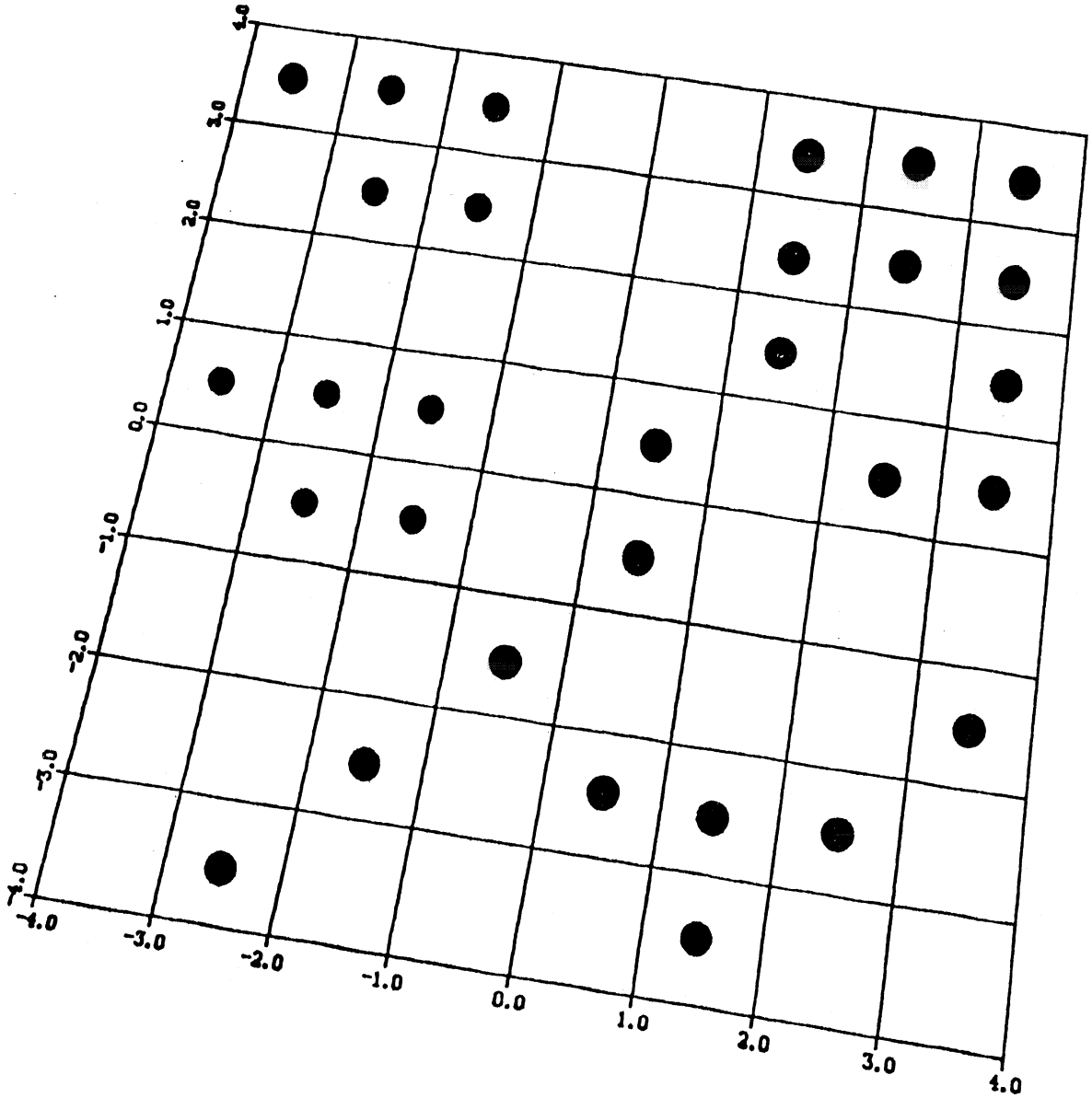


FIGURE 2 - A BINARY IMAGE

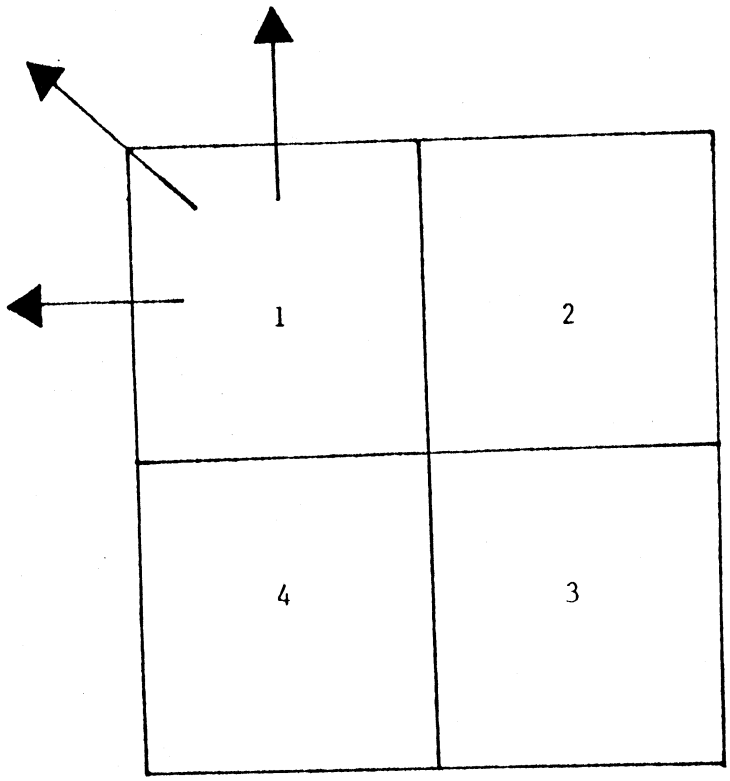


FIGURE 3 - A 2 x 2 TEMPLATE

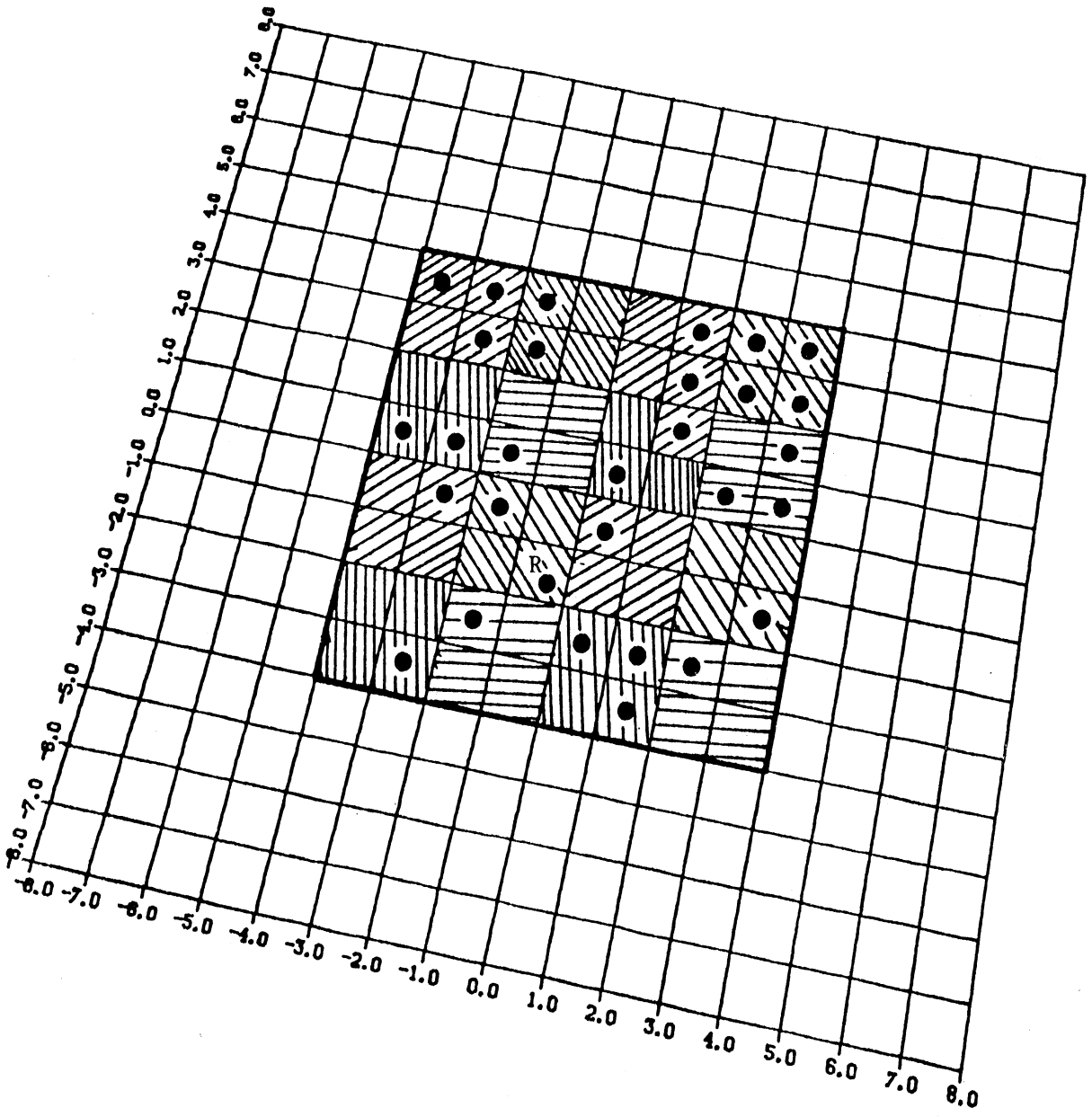


FIGURE 4 - LEVEL 1 OF THE IMAGE PYRAMID

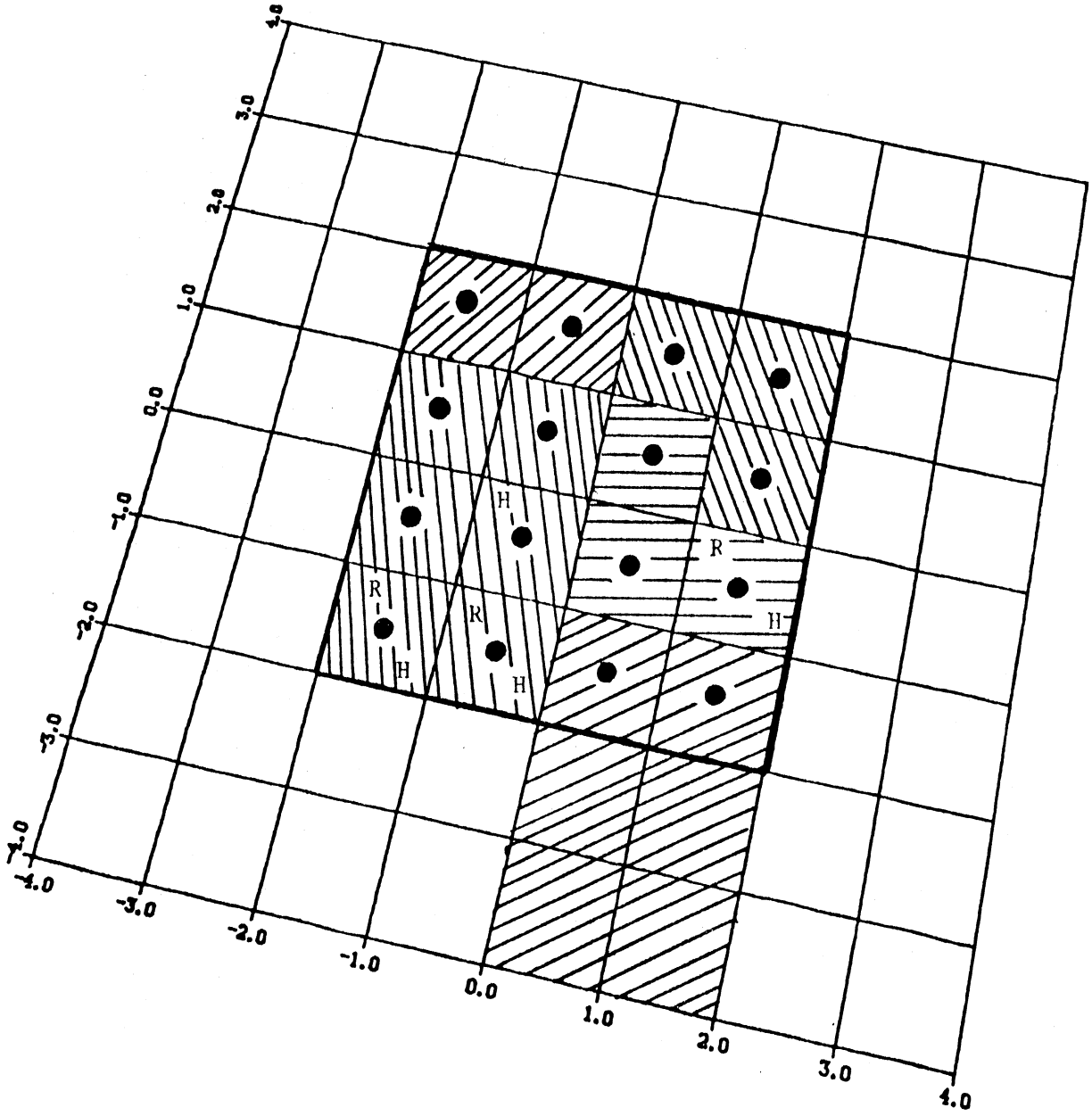


FIGURE 5 - LEVEL 2 OF THE IMAGE PYRAMID

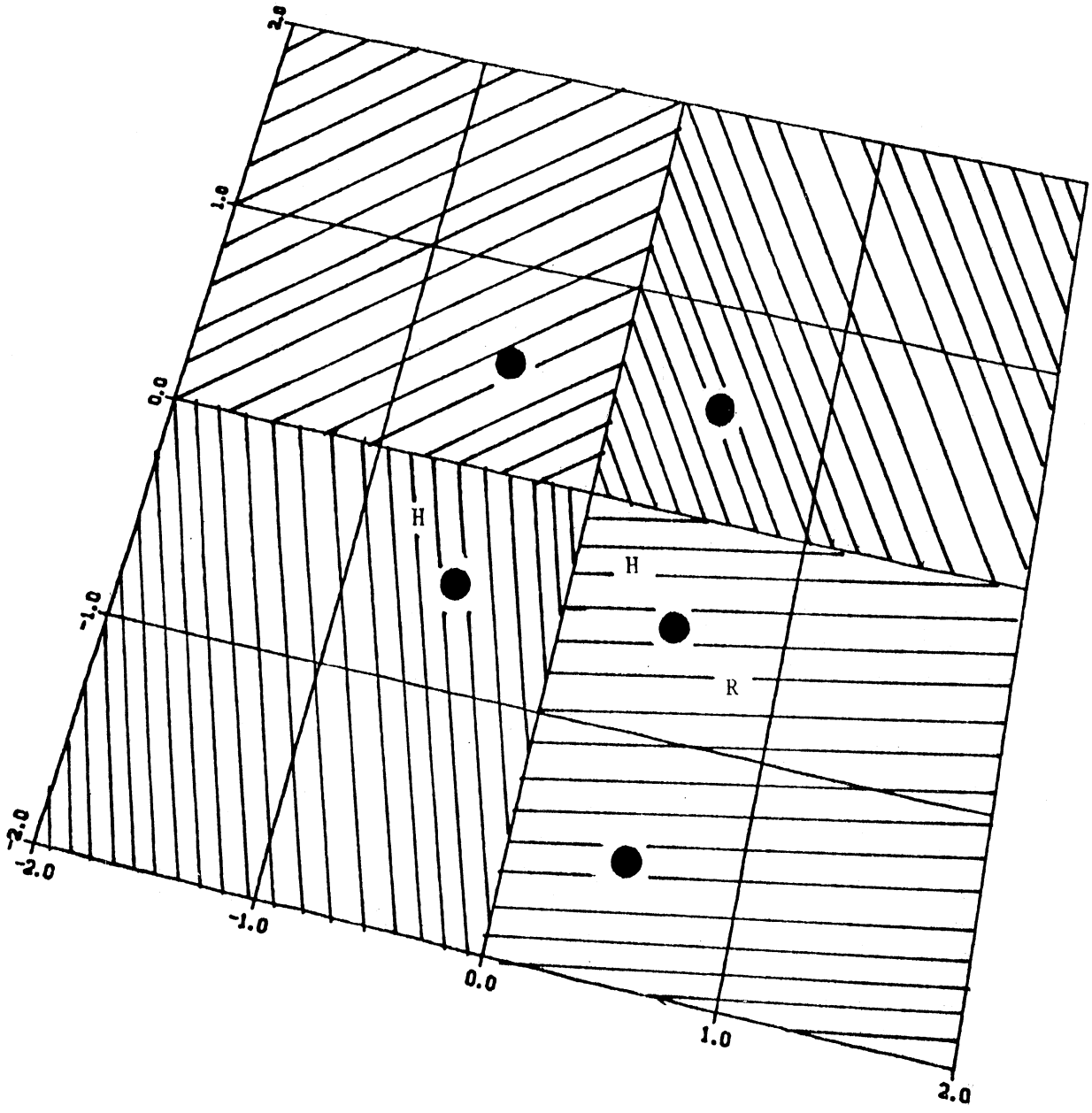
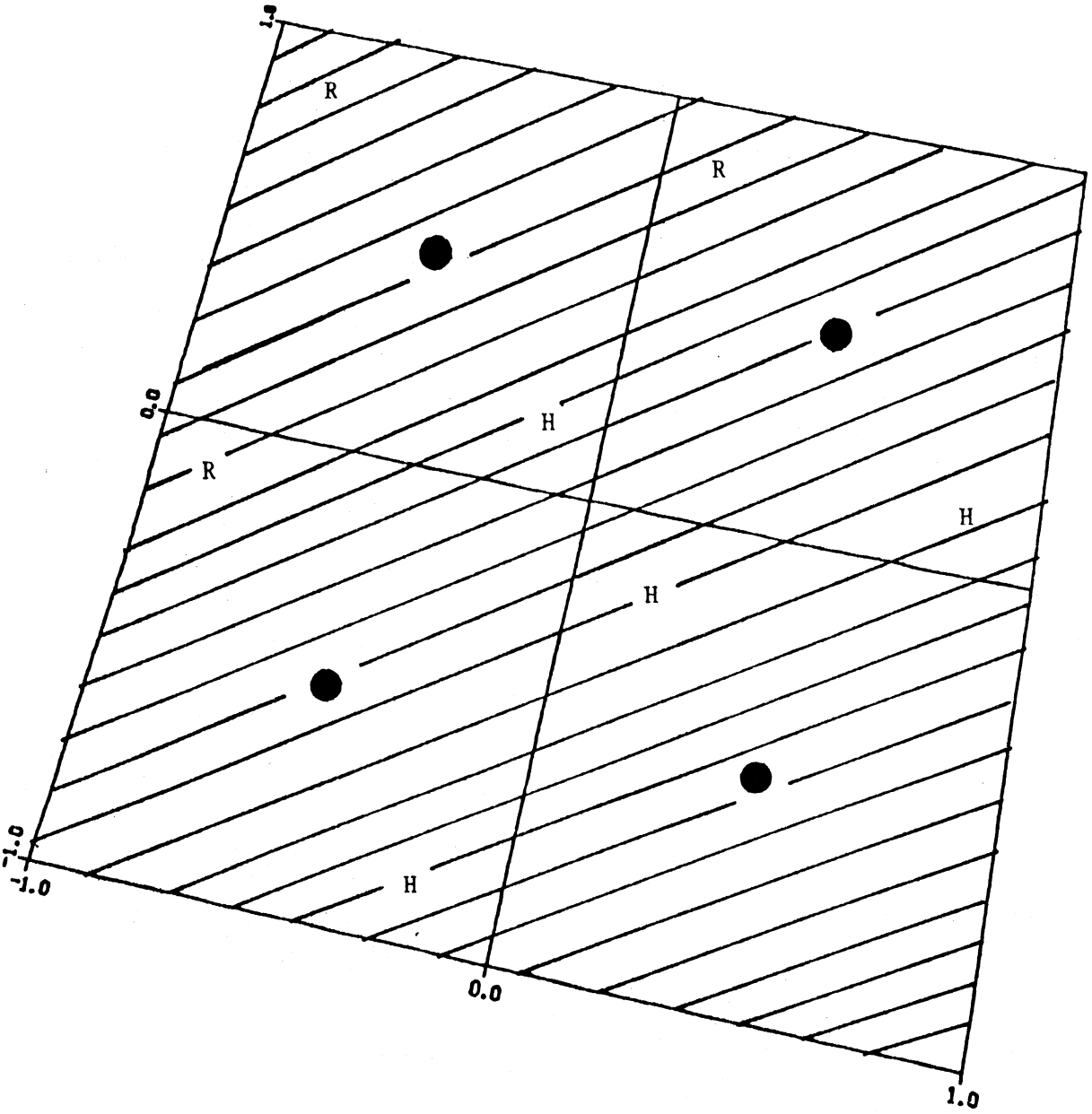


FIGURE 6 - LEVEL 3 OF THE IMAGE PYRAMID


R
R
H
R
H
H
H
1.0
0.0
-1.0
-1.0
0.0
1.0

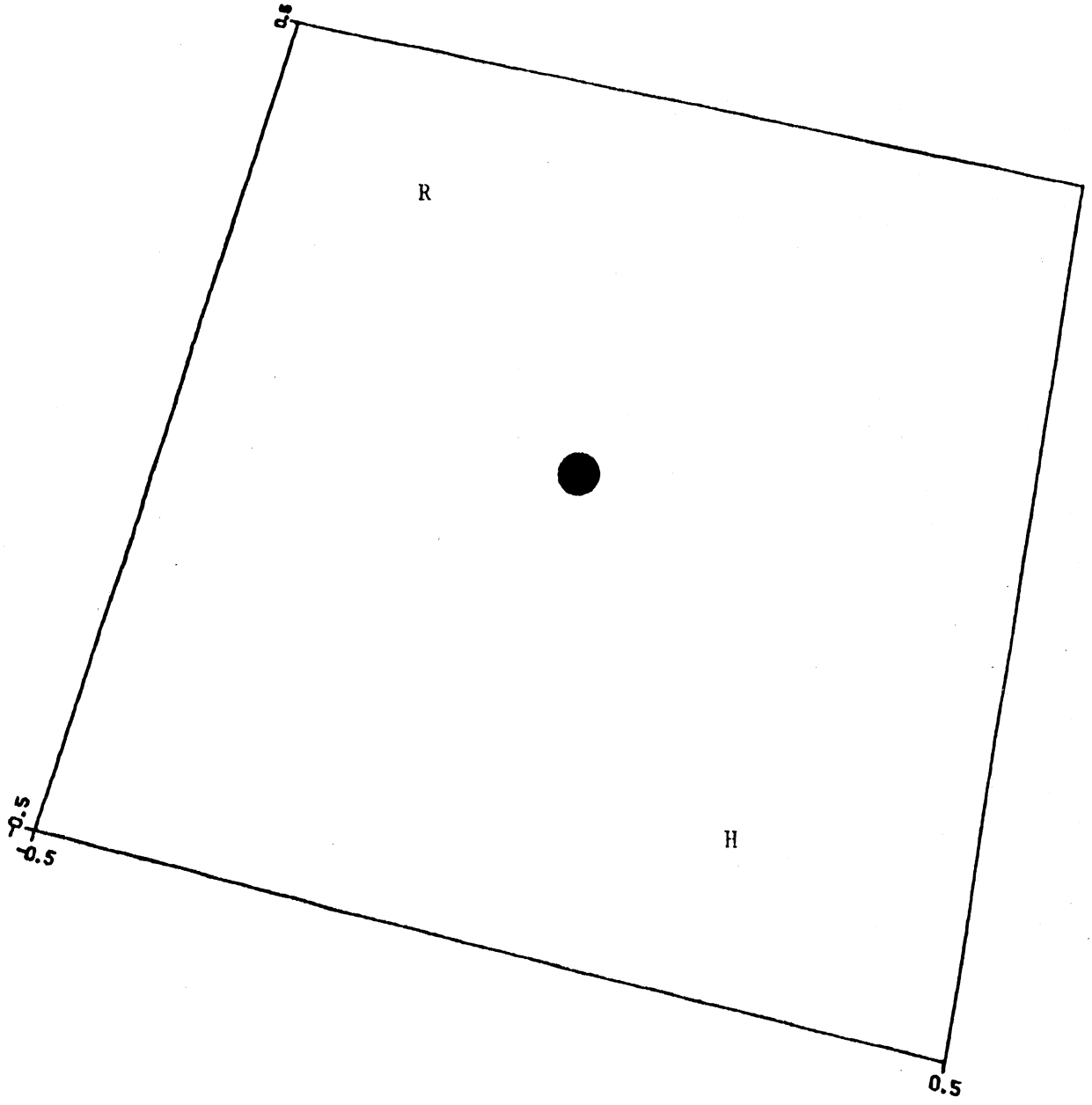


FIGURE 8 - LEVEL 5 OF THE IMAGE PYRAMID

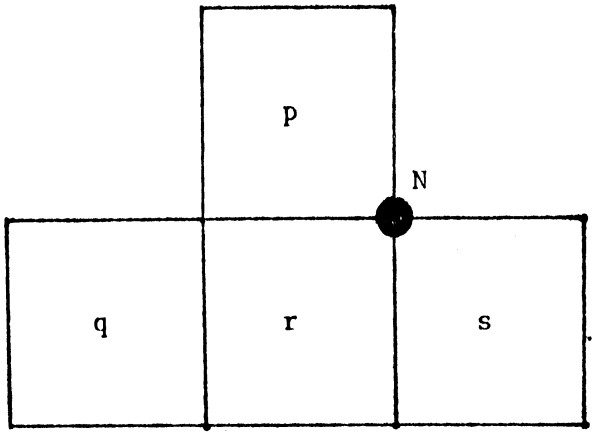
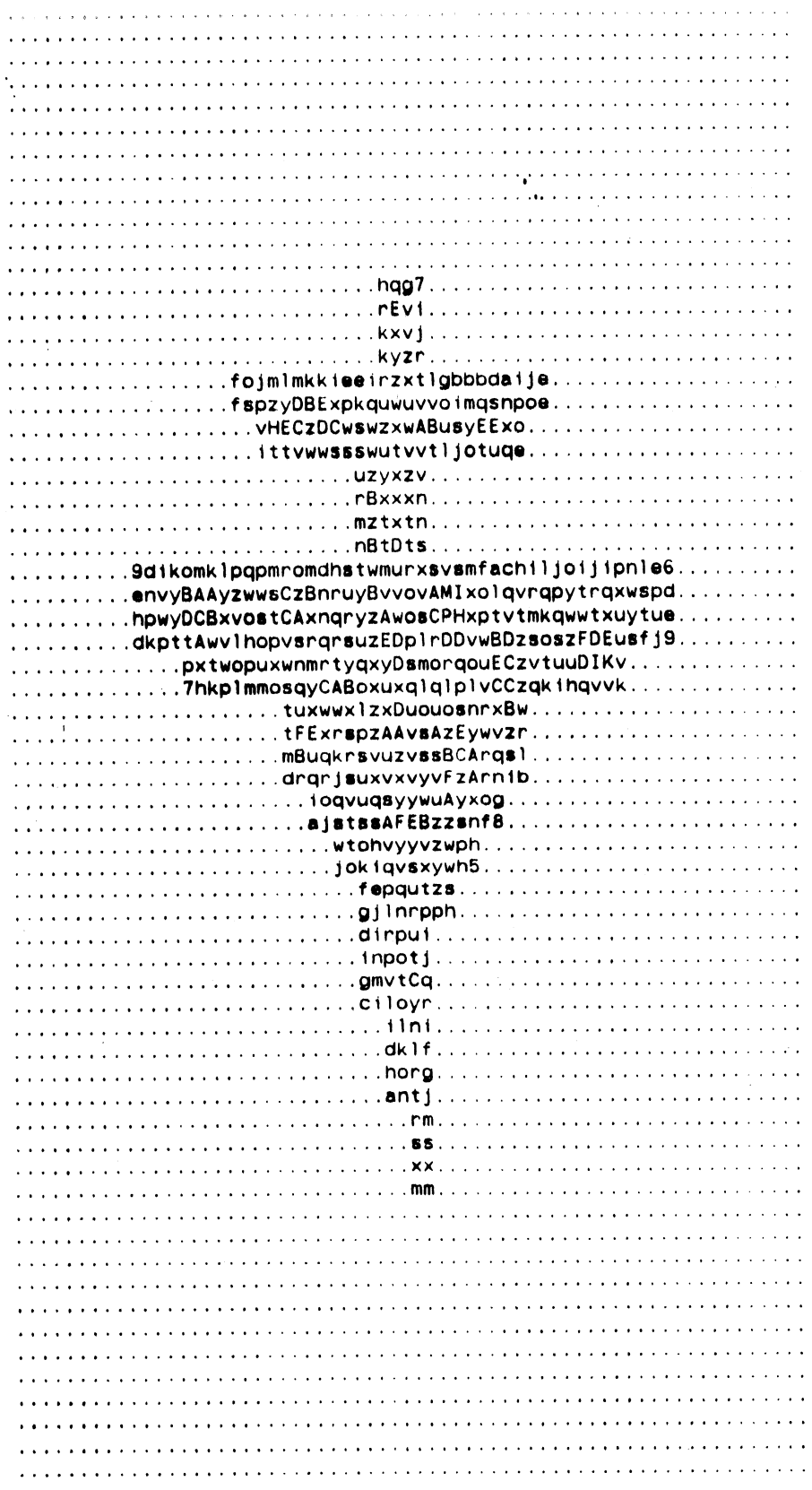


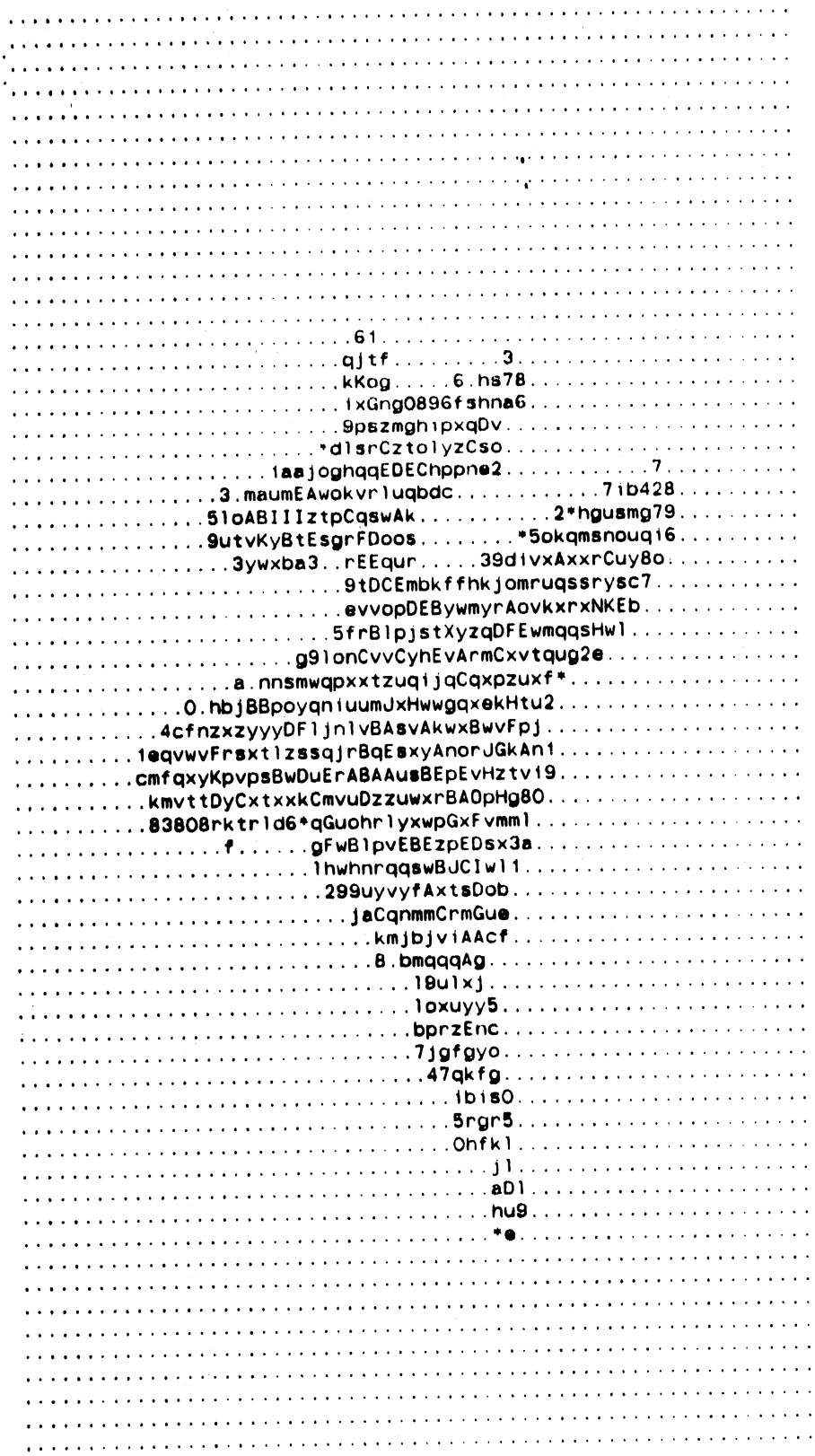
FIGURE 9 - THE PYRAMID CALCULATIONS



Fixed radius eccentricity = $.368 \times 10^{-2}$

Variable radius eccentricity = $.108 \times 10^{0}$

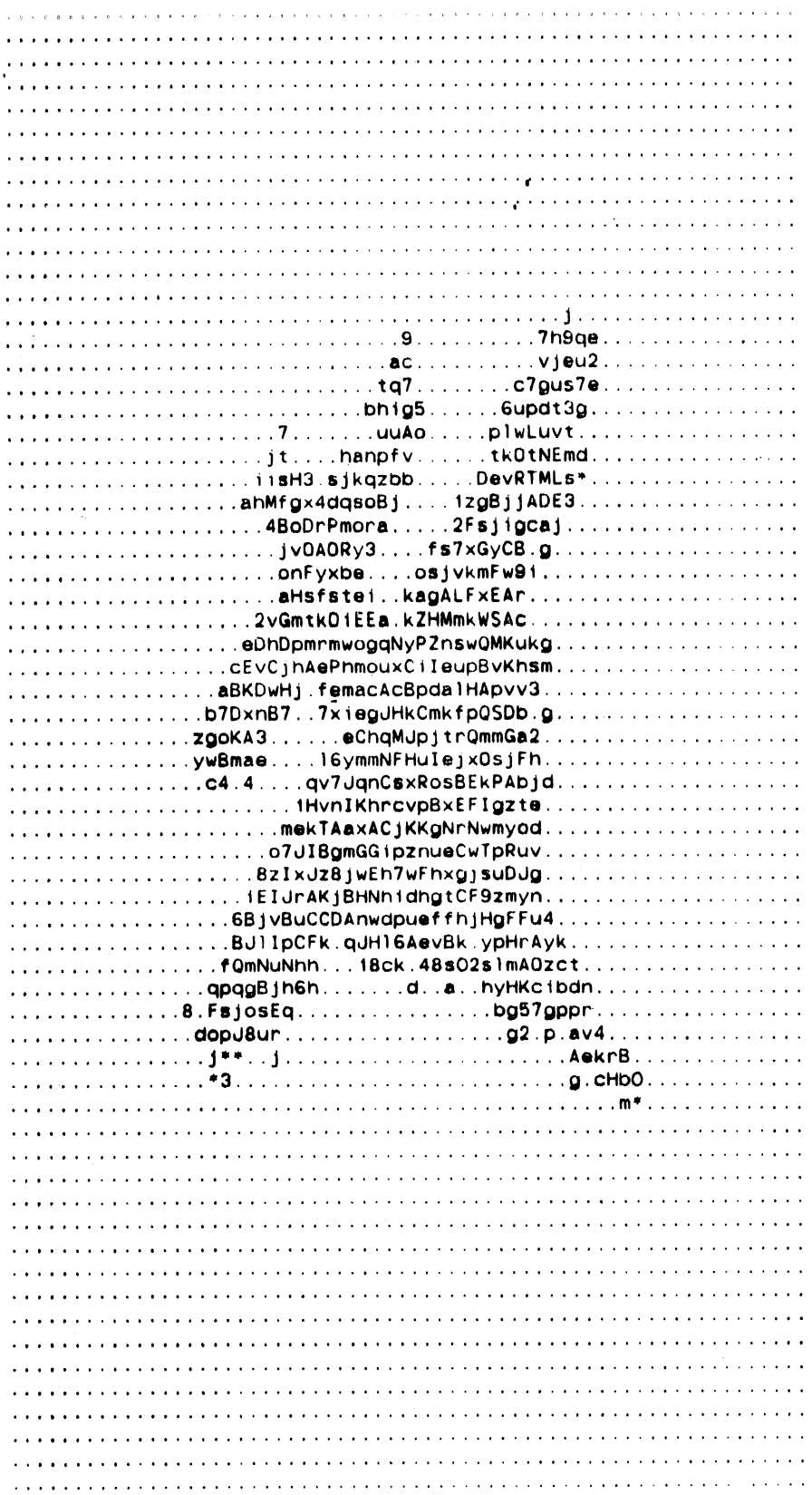
FIGURE 10 - IMAGE #3



Fixed radius eccentricity = .348 x 10^{-2}

Variable radius eccentricity = .946 x 10^{-1}

FIGURE 11 - IMAGE #3 ROTATED 15 DEGREES WITH $\pm$ 10 UNITS OF NOISE



Fixed radius eccentricity = $.428 \times 10^{-2}$

Variable radius eccentricity = $.120 \times 10^{0}$

FIGURE 12 - IMAGE #3 ROTATED 45 DEGREES WITH $\pm$20 UNITS OF NOISE