IMAGE BANDWIDTH COMPRESSION BY DETECTION AND CODING OF CONTOURS

by

GARY ARNOLD WALPERT

S. B., Massachusetts Institute of Technology
(1964)

S. M., Massachusetts Institute of Technology
(1966)

E.-E., Massachusetts Institute of Technology
(1966)

SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June, 1970

Signature of Author
Department of Electrical Engineering, April 3, 1970

Certified by
Thesis Supervisor

Accepted by
Chairman, Departmental Committee on Graduate Students

IMAGE BANDWIDTH COMPRESSION BY DETECTION AND CODING OF CONTOURS

by

GARY ARNOLD WALPERT

Submitted to the Department of Electrical Engineering on April 6, 1970 in partial fulfillment of the requirements for the Degree of Doctor of Philosophy.

ABSTRACT

The results of the computer simulation to reduce the redundancy in a transmitted image by contour detection and coding are presented in this study. The two-dimensional image is divided into its low and high frequency components. The low frequency component is transmitted to the receiver at a reduced sampling rate corresponding to its low bandwidth. The high frequency component is approximated by locating the essential contours and fitting quadratic curves to these contours. The image is then reconstructed from the quadratic curves and the low frequency image as in a "synthetic highs" system. The resulting total number of bits required to describe the image is reduced by a factor of up to 12.1 when compared to 6-bit PCM.

THESIS SUPERVISOR: William F. Schreiber
TITLE: Associate Professor of Electrical Engineering

ACKNOWLEDGEMENT

The author wishes to thank Professor William F. Schreiber for his constant encouragement, guidance, and patience during this study.

The author also wants to thank his wife, Ellen, for her patience and understanding through some very trying times.

Computation for this study was done at the Information Processing Center at the Massachusetts Institute of Technology.

This work was done at the Research Laboratory of Electronics of Massachusetts Institute of Technology and was supported in part by the National Science Foundation and in part by the National Institutes of Health (Grant GM 14940-03).

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

1. BACKGROUND

1.1 Introduction

The transmission of pictorial data has become commonplace in a world with televised moon landings and intercontinental live television. Scientists and engineers have constantly strived not only to improve the quality of pictorial communication but also to reduce the channel capacity needed for the transmission of an image. This investigation examines one method of image transmission and the relationship of picture quality and channel capacity using that method.

In the area of digital facsimile transmission, the great wealth of literature and patents indicates an approach which is often independent of the psychophysical relationship between the human observer and the image itself. Techniques such as run-length coding[1], line to line correlation[2], and block encoding[3] are well known. Such techniques have also been used in the transmission of grey tone images; that is, pictures where the full range of black to white is required to be preserved. Bit plane encoding[4], methods of making use of a priori second or higher order statistics[5,6], and others[7] have been successful in reducing the required channel capacity by factors of 2 or 3 with varying degradations in picture quality.

None of these methods attempt to exploit to any substantial degree the human subjective response or the basic properties of pictures which are not statistical. The Mach phenomenon[8] which results from a subjective enhancement of sharp brightness changes of an image is one example of subjective distortion of a picture. Since the eye does not perceive contours, abrupt changes in brightness, accurately, and a significant amount of "information" in an image is contained in the contours[9], and all pictures of interest have contours, a logical approach to image transmission would involve altering the contours in some way so that the change would allow the image to be transmitted with a reduced channel capacity.

The method which was studied separates the contours from the body of the picture. The remaining portion of the picture, essentially a low-pass picture, is coded and transmitted at a reduced channel capacity. The contours are also coded and transmitted to achieve significant reductions in channel capacity; and at the receiver the low-pass picture and the contours are recombined to form the original picture. See Fig. 1.

This work represents an extension of work begun by Schreiber[10] and continued by Pan[11] and Graham[12]. They have shown that a considerable reduction in the channel capacity required to transmit an image can be realized compared to conventional 6-bit PCM. A signal, B(x,y), can be processed through the system shown in figure 2 without distortion; and specifically, if $B(x,y)$ represents the spatial brightness distribution of an image, then an image can be processed without distortion.

Schreiber has shown that a good choice for $h(x,y)$ is a two-dimensional low-pass filter. According to the Nyquist criterion, the output of this filter need not be sampled at as high a frequency as the input to retain all information. Graham[12] found for example that a 32 X 32 point sampling array was adequate to reconstruct the output of his two-dimensional Gaussian low-pass filter, operating on a 256 X 256 point image. The number of sample points which must be transmitted is thereby reduced by a factor of 64.

If $h(x,y)$ is a low-pass filter, then $g(x,y)$ must be a high-pass filter. The portions of a signal which contribute to the high frequency spectrum are those where the signal changes rapidly. In an image, this corresponds to what the eye perceives as edges, areas where brightness changes occur in small distances. Thus an image passed through $g(x,y)$ contains primarily the edges of a picture. By approximating $g(x,y)$ with an edge detector, a coder, a decoder, and a reconstruction filter, a distorted picture, $B'(x,y)$, is obtained (Fig. 3). The results of Graham's work demonstrated that this approximation is valid (in a subjective sense) and that an image may actually be subjectively sharpened by slightly accentuating the edges.

The research hereinafter described was performed using the IBM 360/65/40 at the M.I.T. Computation Center. The amount of available core storage restricted the size of the processed images to 200 X 200 points. The scanner used was constructed by Professor William Schreiber of the Cognitive Information Processing Group of the Research Laboratory of Electronics. The scanner is capable of resolving 256 levels of grey scale.

1.2 Area of Research

There are four areas of investigation. First, how should the contours or edges of a picture be located? Second, given the contour locations and the magnitude and direction of the change in brightness how should this information be coded using curve fitting techniques? Third, what pre- or post- processing may be desirable? Fourth, what are the subjective effects of the various coding and pre- or post- processing operations?

2. THE LOW-PASS FILTER AND RECONSTRUCTION FILTER

2.1 The Problem in One Dimension

The most efficient way to approach the two-dimensional problem is to analyze the one-dimensional problem. Consider a one-dimensional contour represented in Fig. 4(a) by the signal b(x). The contour is a unity step located at $x=x_o$. Passing the signal b(x) through a low-pass filter, with impulse response h(x), produces an output l(x) as shown in Fig. 4(b). Differentiating the signal b(x) results in an impulse of unity magnitude located at $x=x_o$ (Fig. 4(c)). One method of reconstructing the original signal b(x) requires a signal r(x) to be added to l(x) to produce b(x). Then r(x) = b(x) - l(x). By passing the impulse $u_o(x_o)$ through a reconstruction filter, with impulse response $b(x+x_o)$ - l(x+x) the desired signal r(x) is synthesized. (Fig. 4(d)) Thus in the one-dimensional continuous spatial domain, given an input a(x), a(x) being any real function of x, the system in Fig. 5 will produce a(x) as the output. In the system of Fig. 5, the differential operation is in fact an edge locating operation.

The scanning of an image to be processed in a digital computer results in a sampled signal and not a continuous signal. Likewise signal processing in a digital computer is not a continuous, but a discrete process. Consequently rather than using the system shown in Fig. 5, a corresponding system of difference equations must be solved.

2.2 The Problem in Two Dimensions

Analytically the two-dimensional problem is identical to the one-dimensional problem. The low-pass filter used throughout the investigation is the circularly symmetric, separable Gaussian low-pass filter:

$$\text{(Eq. 2.1)} \quad h(x,y)= \frac{1}{2\pi\sigma^2} \exp(- \frac{x^2+y^2}{2\sigma^2})$$

To avoid excessive computer processing time, σ was chosen to be equal to four sample points and the Gaussian function was sampled in one dimension at 19 points equally spaced at $\frac{1}{4}\sigma$ intervals. The two-dimensional filter was chosen so that the aperture was approximately 10% of the original picture and so that the ratio of the largest sample value to the smallest sample value was approximately 100:1.

The reconstruction filter is a function of the low-pass filter, h(x,y), used with the picture and the operator used to detect and locate contours. The entire process of edge detection, coding, and reconstruction will approximate a transfer function of 1 - h(x,y). If the edge detecting operator is denoted by e(x,y), and the reconstruction operator by r(x,y), then $e(x,y) * r(x,y) = 1 - h(x,y)$, where "*" denotes two-dimensional convolution. It is this equation which must be solved for r(x,y) with the various low-pass filters, h(x,y), and edge detectors, e(x,y), as parameters.

Graham[12] has worked out a method for obtaining the exact reconstruction filter for any low-pass filter. Using this method the reconstruction filter and low-pass filter are shown in Table 1. Note that the truncated low-pass filter is normalized so that

(Eq. 2.2) $$\int_{x,y}^{\infty,\infty} h(x,y)\,dx\,dy = 1$$

The output of the two-dimensional low-pass filter is an image which has no sharp changes in brightness. It is thus similar to an out of focus picture.

The Fourier transform of the Gaussian function is also a Gaussian function.

(Eq. 2.3) $$h(x,y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

$$H(\omega_x,\omega_y) = \exp\left(-\frac{\sigma^2}{2}(\omega_x^2+\omega_y^2)\right)$$

By truncating the low-pass filter h(x,y) at $\pm 2\frac{1}{4}\sigma$, the high frequencies of $H(\omega_x,\omega_y)$ are somewhat accentuated. However, 90% of the energy of the Gaussian function is located at frequencies less than $f_m = \frac{1.516}{2\pi\sigma} = 0.0602$; therefore the sampling theorem in two dimensions provides that sampling at a rate $f = 2f_m$ should provide sufficient sampling to reconstruct the low-pass picture. This requires sampling approximately every eighth point, and since the sampling array is 200 x 200 points, a 25 x 25 point sampling array is needed.

As a convention the brightness at boundary points will be assumed to continue beyond the boundary so that no sharp edges will appear on the perimeters of the picture.

3. CONTOUR DETECTION AND CONTOUR FOLLOWING

Many operators have been used to detect and locate edges or contours.[13,14,15] Graham used both the gradient and the Laplacian operators. The Laplacian operator, for example, operating on a contour (Eq. 3.1) results in a double line of points, one positive in magnitude and one negative in magnitude.

(Eq. 3.1) $h(x,y) = 4p(x,y)-p(x-1,y)-p(x+1,y)-p(x,y-1)-p(x,y+1)$

In principle the location of the edge can be determined by taking the midpoint between the double line of points. The magnitude of the edge would be the magnitude of either set of points which result from the Laplacian operation.

The gradient operator results in only a single line of points with each point being described by a vector magnitude and direction. The resultant single line of points makes contour location significantly simpler in computer simulation and therefore the gradient operator was used.

(Eq. 3.2) $G(x,y)=(p(x,y)-p(x-1,y))\bar{x}+(p(x,y)-p(x,y-1))\bar{y}$

(Eq. 3.3) $G(x,y)=G_x\,\bar{x} + G_y\,\bar{y}$

Although the magnitude of the gradient could have been found at each point, the number and type of operations needed to calculate the magnitude at each sample point required too much computer time. In order to reduce computer processing time the following procedure of contour following was used. The contour tracing procedure is described with reference to the grid pictured in Fig. 6 (see Appendix A for the explanation of the grid in Fig. 6). In order to judge whether a contour is important enough to transmit, an absolute threshold is established so that some point on the contour must have an "x" or "y" gradient component which exceeds the threshold. Once having thus "latched" onto a contour, the contour will be followed from the "latching on point", in both directions, until the end points of the contour are determined. These end points are defined as those contour points where no points within reach of the presently accepted contour points have a gradient component whose absolute value is higher than a second preselected absolute threshold. When the two end points of the contour have been found, the contour is coded for transmission.

The problem of finding the next point of the contour, given a point on the contour, is solved in the following manner. For every point on the contour there will be eight nearest neighbors. A contour point is defined as the gradient component of a sample point. This is true whether the present point is an "x" or a "y" component of the gradient. (See Fig. 8.) When a new contour point has been found, an erase procedure is used to prevent the tracing algorithm from reversing and returning to the point from which it had just come. For example, in Fig. 8, if the present contour point were the point marked "x", and if the new point were a corner point, then those points not accessable to the new point, i.e. the diagonally opposite corner point and its four closest neighbors, would be erased or set to zero. Thus for example, if the next point were point 1, points 3,4,5,6, and 7 would be erased. If the new point were point 4, a mid-point, then points 1,7, and 8, those points not accessible to point 4, would be erased. In addition the previous contour point is also erased. A special procedure is used for the starting or initial point of a contour, so that gradient information relating to the contour extending in the second direction is not destroyed.

4. CONTOUR CODING

4.1 Background

Once the location of the contours and the corresponding gradient values are specified at every point, the next problem is the efficient transmission of that data. Perhaps the most obvious method of transmission is to send every contour point followed by the full gradient information for that point. If the scanning raster is 200 x 200 points, sixteen bits are required to specify each contour point. Assuming that a comparison will be made vis-à-vis a six-bit PCM system, probably a minimum of seven bits will be desired to specify the value of each component of the gradient (six bits for magnitude and one bit for sign). Thus thirty bits would be required to specify each contour point. An average picture may contain 8000 contour points or more; thus 240,000 or more bits would be required only to transmit the contours. The end result of the preceding operation would be a picture lacking small brightness changes or "texture" while requiring more bits than the original picture. This result shows the need for a more efficient manner of transmission.

Graham[12] proposed a point by point coding scheme which took advantage of the following characteristics of contours. First, contour points are always adjacent to one another. Second, the magnitude of the gradient does not change rapidly.

Third, the direction of the gradient does not change rapidly. By making use of the probabilities, and restricting the magnitude of the gradient to one of four levels and the vector direction to one of eight equally spaced directions, he calculated that, using Huffman[16] coding, the expected number of bits needed to transmit a contour point location and the corresponding gradient information was approximately 6 bits per contour point. In addition, twenty-one bits are needed for the first contour point, sixteen bits to locate the first point, two bits for the gradient magnitude and three bits for the gradient direction. Thus for a picture with about 6200 contour points, with 200 contours, 40200 bits are needed to transmit the contour information. In general, for any particular curve of n points, 21 + 6(n-1) bits are needed to transmit the data. This is a considerable reduction from the previous method.

Graham's proposed coding scheme takes advantage of the nature of contours to generally have a continuous first derivative. The scheme also takes advantage of the characteristic of the gradient to have a slowly changing vector direction and a relatively unchanging magnitude. The number of bits required to transmit the contours thus increases linearly with the number of edge points. If the resolution is doubled, the number of bits required to transmit the picture by 6-bit PCM code increases fourfold, while the number of bits required to transmit the contours only doubles. Thus as the resolution of the scanning raster increases, an increase in the savings in channel capacity is noted compared to six-bit PCM.

Although the savings in channel capacity is significant, the increase in the number of bits needed to transmit the contour information as the resolution increases is troublesome. Aside from this annoyance, it is desirable to develop a more efficient coding mechanism. Intuitively the definition of a contour at one resolution does not seem to require more definition as the resolution increases. Although additional resolution may add new contours to the lower resolution picture, the old contour will generally be unchanged in shape. It is also recognized that many contours are long (30 or more points) and continuous. Therefore the idea of fitting one of a family of curves to the contour location and gradient data seemed appealing. The family of linear curves did not seem desirable due to the inherently non-linear character of most contours. See also Pan[11]. Use of families of higher order curves involves an increasingly more complicated problem and consequently is more time consuming. For this reason, simple second order curves of the form:

(Eq. 4.1) $x = a + by + cy^2$

(Eq. 4.2) $y = d + ex + fx^2$

were used to fit the picture contour locations and their corresponding gradient values.

The criteria to be used in curve fitting, determining whether a given curve sufficiently well "fits" the contour, must be incorporated in a mathematical formula which takes account of the physiological effects within the eye. Since the eye is very sensitive to discontinuities in the contour, one criterion must reflect this factor. A second factor must relate to the maximum allowable amount of deviation from the true contour location. To incorporate both these factors, an iterative approach was used employing a best mean square error fit.

4.2 A Procedure for Contour Coding

4.2.1 Weighting the Contour Points

The contour coding method used in this investigation fits a simple second order curve to the data points using a mean square error criterion. Initial attempts to fit artificial data (where all data points were weighted equally) indicated that the largest absolute errors occurred at the end points of the data. This unexpected occurrence results from a "built-in" weighting function inherent in a mean square error fit. Consider the case of a continuous curve. An absolute error of two units at an end point probably implies an error of "one unit" at the next to last point and maybe an error of $\frac{1}{2}, \frac{1}{4}$....as we approach the middle of the curve. The contributions to the squared error are then $2^2 + 1^2 + \frac{1}{2}^2 + \ldots \cong 5\frac{1}{4}$. If the same error were to occur in the middle of the curve, there would be contributions to the error not only from points on one side of the largest error point (where, for example, the error is two) but on both sides of the error point. The contribution to the error in this case may be 7. Thus the larger errors will tend to occur towards the ends of the curve where they are less costly, in a mean square error sense; and an absolute error criterion imposed on these end points will also tend to act as an error criterion along the whole length of the curve. In a normal curve fitting procedure, this would not cause a problem; however, when the data represents contour locations of an image, the miss-match at the end points is immediately sensed by the observer with a resultant subjective dislike. To counter this problem the end points are weighted by some factor generally greater than five. By changing this weighting factor, the location of the maximum absolute deviation of the fitted curve from the original curve can be shifted from the end points to the interior points; and the error at the end points can be made as small as desired. By holding the end points relatively fixed there is a trade-off; two of the three degrees of freedom in fitting the curve have been severly limited, and consequently, the deviations in the interior of the curve become greater than they would normally be. For a given contour, there will generally be an optimum value of the weighting factor which provides the desired distribution of the error along the contour. This optimum value will depend upon the number of points in the contour, the shape of the contour, and the error criteria. In an actual picture where contours are generally of random length and shape with unknown a priori probability distribution, only an approximate best choice can be made. Thus, for the picture shown in Fig.12c,Table 2 gives experimental results for the number of curves needed to fit the contours of the picture for various error criteria. When one simple second order curve could not fit the entire contour, two, three, or more curves were added to the first curve until the entire contour was described.

4.2.2 The Error Criteria

The error criterion used throughout is an absolute criterion. Thus if the contour data points are (x_i, y_i), $i=1,n$, and the approximating curve is $x=a + by + cy^2$, then if any error e_i, ($e_i = x_i - x_i$; $x_i = a + by_i + cy_i^2$; $i=1,n$) is greater than a predetermined value, the approximation is rejected. Since the eye places an especially tight tolerance on the end points of the contours, a two-step error criterion was considered for the error detection process. The two-step error criterion would place one error limit on all points except the end points and a second smaller error limit on the end points. It was found that this approach could effectively be approximated by weighting the end points appropriately and using the single error criterion. Therefore the two-step approach was not used.

4.2.3 Continuing Curves

In normal operation, one second order curve will often not fit a complete contour. Thus a continuation curve will be used to fit as much of the remaining contour as possible. Originally all continuation curves, that is, curves which continue coding the contour at the point where a previous curve stopped, were fitted so that the first derivative of the preceding curve and the first derivative of the following curve were equal at their common point. This produced a smooth contour in accordance with what the eye was expected to prefer. A side effect of this procedure, however, was a decrease in the average number of contour points each curve was able to fit, when compared to the same curve fitting procedure where the curves were allowed to be discontinuous in the first derivative at their common point. A second side effect was a distinct waviness in the fitted curves as a result of restricting the first derivative. Pictures are compared later in this study as to the subjective and practical effects of both procedures.

4.2.4 The Iterative Approach

It was discovered early in this investigation that a single second order curve would generally not fit a complete contour. Consequently an iterative procedure was developed whereby initially only the first three points of the contour were fit. Then the appropriate change or perturbation in the coefficients of the second order equation were made to add the fourth point of the contour. If the second order curve fit the contour points within the allowed error criterion, another point was added. When the error criterion was exceeded, the curve was terminated at the previous point, and a second curve was initiated at that point.

4.2.5 Pre-scan

A pre-scan of the contour points is used to determine whether a curve of the form $a + bx + cx^2$ or $d + ey + fy^2$ would result in a better "fit". Occasionally the wrong choice is made. To provide for the contingency, if fewer than a specified number of points are fit by the chosen second order curve, the other version of the simple second order curve is used, and the results are compared. The curve fitting the larger number of points is used.

The method of calculating the coefficient for a second order curve yielding the best mean square fit to a given set of data can be found in many reference books.[17,18,19] The general set of equations is presented in Appendix B.

5. GRADIENT CODING AND DECODING

5.1 Coding

Once the contour location information has been coded by fitting curves to the contour location points, there still remains the problem of coding the gradient magnitude and direction. As will be seen in the gradient decoding, it is tacitly assumed that the direction of the gradient is quantized yielding an angle of $\frac{n\pi}{4}$, n=0, 1,...,7, therefore only the magnitude need be coded. It will also be assumed, as was substantiated from Graham's work[1,2] that the magnitude and direction of the gradient are slowly varying functions of distance. Thus, it should be reasonable to code the gradient in the same manner as the contour location. If the contour location is fit by the curve $x=f(y)$, then the gradient is coded by the curve $g(x) = f'(y)$ where $g(x)$ is the absolute value of the gradient component at each contour point x, and $f'(y)$ is the simple second order curve $a + by + cy^2$. Similarly, if the contour location is fit by the curve $y = f(x)$ the absolute value of the gradient is fit by the curve $g(y) = f'(x)$. In fitting the second order curves to the gradient information, the best mean square error fit of the entire contour is used.

An approximation of the gradient magnitude occurs when fitting the second order curve to a series of points whose gradient information has for example, an "x" component of the gradient equal to +2 and the "y" component equal to -3. The mean square error fit will give a value of $2\frac{1}{2}$ to both components of the gradient; this has the effect of quantizing the vector direction as mentioned above, but in addition, the magnitude of the gradient is also distorted though not enough to give a subjectively dissonant appearance to the picture. In the example the magnitude is changed from 13 to 12.5, a change of only 2%.

In order to preserve the proper sign for all gradient information (it will be remembered that the absolute value of the gradient information is coded) an extra bit of information must be transmitted. This bit of information, the vector bit, indicates which side of the contour has a higher average brightness level and which side of the contour has a lower brightness level. Thus, if for the contour x=y, starting at y=2 and ending at y=7, $g(\bar{x})=5$ and the vector bit indicates that going from y=2 to y=7 that portion of the picture to the right of the contour is brighter than that portion to the left of the contour, the resulting gradient component would be as shown in Fig. 10, where the gradient is defined as in Eq. 3.2.

The problem of determining the value of the vector bit is a difficult task. Many errors were noted in the earlier pictures, an example of which appears in Fig.24. Three different algorithms were used during this investigation. In the first algorithm, a single vector bit value was assigned to the entire contour no matter how many curves were required to code the contour.

The vector bit value is initially determined by the first and last pair of points of the contour. If those values agree, the vector bit is set to that value. If the values do not agree, then values are taken at various points along the contour until there is substantial agreement in the vector value. If there is no such agreement, the best estimate is made on the basis of majority rule and that contour is noted in the diagnostic computer print out. It has been found, however, that the ends of a contour do not reliably indicate the value of the vector value bit. Therefore, a second method was developed.

The second method of determining the vector bit values rested on the assumption that the value determined by using the entire contour was misleading and that only a middle portion of the contour should be used. The same procedure as used in the first method is used in determining the vector bit value in the second method except that the first pair of points used is located a distance equal to $\frac{1}{3}$ of the total contour length from the beginning of the contour and the second point pair is located about $\frac{1}{3}$ from the end of the contour. The procedure of the first method was followed if the values indicated by these pairs of points did not agree.

The third method used to determine the vector bit value resulted from an increase in errors noted in the process of coding the "cameraman" pictures. The contours of these pictures tended to be very long and often included touching natural contours which were in fact unrelated. Thus "one" contour might include the side of the tripod, the coat of the cameraman,and the head of the cameraman. In some instances, a single vector bit value could not describe all of the components. When several curves were needed to fit the entire contour, the intersection of two natural contours often occurred at points where one curve would end and a second curve would begin. Thus it was a natural though not necessarily obvious extension of the first or second method described above to assign a vector bit value to each curve which was used to fit the entire contour. This method results in a picture with the least number of errors.

5.2 Decoding

The decoding of the "x" and "y" gradient information uses the same two-dimensional grid as described in section 2 and shown in Fig. 6. The picture elements will eventually be located at points (x,y) where both "x" and "y" are even integers. The "x" gradients will be located at points (x,y) where "x" is odd and "y" is even. The "y" gradients are located at points where "x" is even and "y" is odd. Thus a gradient component can never fall on points where both "x" and "y" are even or where both "x" and "y" are odd. Initially the points specified by the second order curves are listed in order. Thus if the curve is $x=y^2$ and the curve begins at y=2 and ends at y=4 then the points listed are (4,2), (9,3), and (16,4). If any of the listed points are located where a gradient point cannot exist, e.g. (4,2), that point is moved in the "x" direction or the "y" direction to the nearest acceptable gradient location. The direction in which the point is moved is determined by the second order curve which designated that point. If the curve has the form x=f(y), which due to the curve fitting procedure used generally describes a contour which has a relatively small range of "y" values and a larger range of "x" values, the point is moved to the closest "x" gradient location in the direction of the nearest point. Thus the point (4,2), referred to above, would be moved to (4,3), (9,3) would be moved to (8,3) and (16,4) would be moved to (16,3). This causes a further distortion in the approximation process; however, since the curve itself may be a minimum of 2 or 3 points in error, the added error does not appear to cause a problem. When all the listed points are located at valid gradient points, they are connected in sequence by a straight line approximation with the added points being placed on "x" and "y" gradient locations so as to form a continuous contour. For the example outlined above, the whole contour is located at valid gradient locations on the line beginning at (4,3) and ending at (16,3). If the two listed points had different "x" and "y" coordinates, for example (4,3) and (12,5), then the line of added gradients would appear as in Fig. 10.

After all the gradient points for a given curve have been located, values are given to the gradient components. The vector bit value is introduced at this point to provide the sign of the various gradient components. If, for example, the gradient information were given by the curve g(x)=5 and the vector bit value described the contour as having greater brightness on the right-hand side of the contour, then the edge of Fig. 10 would have gradient values as shown in Fig. 11.

After all of the gradient components are built up from the second order curve information, the "x" and "y" components are separated from each other and convolved with the appropriate reconstruction filters as previously discussed (See Sec. 2). The results of the separate convolutions, the reconstructed or "synthetic" highs, are then added together with the low-pass, interpolated filtered picture to form the completed picture.

6. THE OUTPUT PICTURES

6.1 Basis of Comparison

The result of any coding and decoding operation is the output copy and the channel capacity needed to transmit the copy. Some picture coding procedures produce an error-free output so that the only basis for comparison is the channel capacity required.[20,21,22] For the curve fitting/contour coding procedure, approximations are continuously being made; therefore some feeling must be acquired in a multi-dimensional space; the quality of the processed picture versus the number of bits required to transmit the picture versus the quality of the original 6-bit PCM picture.

The amount of channel capacity which is used by a 200 X 200 point, 6-bit PCM picture will be the standard against which the channel capacity required for the contour coded pictures will be compared. The standard picture requires a channel capacity of 240,000 bits. In order to compare the processed picture with the standard picture, the number of bits required to transmit it will be calculated as follows:

1. A 25 X 25 point matrix will be used to transmit the low-pass picture. Each element of the low-pass picture will be described by 6 bits. Thus the total

number of bits required to transmit the low-pass filtered picture equals 3750 bits.

2. To transmit the information relating to one curve, the following information must be transmitted:

   a. One bit to describe which version of the second order curve is being sent; that is, a curve of the form x=f(y) or a curve of the form y=f(x).

   b. One bit to describe whether the curve is a continuation of a previous curve (and therefore continuous in the first derivative) or not.

   c. One bit to describe the vector bit value.

   d. Eight bits to describe the beginning value of the independent variable and eight bits to describe the final value of the independent variable. If the curve is a continuation of a previous curve, only the final value of the dependent variable need be sent since the initial value for the continued curve was the final value for the previous curve.

e. Twenty-seven bits are required to specify each initial curve in the contour fitting process. In order to perform the curve fitting process, each contour is normalized to begin at (0,0). The unusually great precision required to perform curve fitting with data values of the various contours requires such normalization. The simple second order curve $a+bx+cx^2$ has three degrees of freedom and can be specified either by giving the values of "a","b", and "c", denoting three points along its contour, or by an appropriate combination of the two methods. Since the initial contour point is defined as (0,0), and since the final value of the independent variable is already specified in subsection "d" above, the latter method was chosen. The value of the dependent variable is specified at the initial, mid-point, and final sections of the curve. The value of the independent variable at the mid-point is defined as the mean of the initial and final values of the independent variable, therefore no additional channel capacity is required to transmit its value. Nine bits are used to transmit the value of each dependent variable, thus an overall accuracy of one sample point is achieved.

f. Fifteen bits are required to specify each curve in the gradient curve fitting process. The method is identical to "e" above. The value of the gradient is specified at the initial, mid-point and final sections of the contour. Five bits are required to transmit each of the required gradient values, thus an overall precision of one part in sixteen is achieved. As with the curve location information, the independent variable has been translated to begin at zero.

Thus a total of sixty-one bits is required to describe the gradient value and location information conveyed by each curve if the curve is not a continuation of a previous curve. If the curve is the continuation of a previous curve, the value of "b", the slope at the initial point, (0,0), of the continued curve, will be determined by the slope of the previous curve at its final point. The initial value of the independent variable will be given by the previous curve, and the initial value of the dependent variable will deviate no greater than twice the maximum allowable error from the value given by the previous curve. In order to adjust the value of the dependent variable at the initial point of the continuing curve, three additional bits are transmitted. Thus a continued curve may be transmitted using thirty-nine bits. In addition it is possible to code the gradient information for a whole contour using only one second order curve by relying on Graham's evidence that the gradient need not be specified to more than two significant binary places. This premise would allow each continued curve to save an additional fifteen bits. In the calculation of channel capacities which follows, it is assumed that the complete gradient information is specified for each continued curve. The rationale for this apparent waste of channel capacity results from the experimental evidence that a large gradient change often occurs at the beginning of a continued curve.

The total number of bits required to transmit the complete picture equals 3750 + 61n + 39m where n equals the number of new curves and m equals the number of continued curves.

6.2 Picture Processing Techniques and Options

A variety of picture processing methods were used to investigate their effects on the final pictorial content. Some methods were used in combination with others to attempt to understand how the various techniques interact with each other. The various techniques are described individually in the following sections.

6.2.1 Thresholds

By varying the two threshold levels, the number of distinct contours and the number of edge points can be varied significantly. Specifically, changing the upper threshold varies the number of contours found, and consequently the number of edge points located. This threshold must be placed low enough to allow each "important" contour to be found, and yet high enough to reject those contours which resemble texture in the picture, or which are in reality noise contours. For a given picture a precise determination seems to be possible only by experimentation; however, a threshold level of approximately 20 (out of 256 levels) seems to work quite well for most pictures.

The lower threshold generally determines the length of the average contour; that is how many points the average contour will contain. It must be set low enough so that an entire contour will be traced, yet high enough so that the contour tracing routine will neither consider noise in the picture as part of the contour nor use noise in the picture to "jump" from one contour to some part of a distinct separate contour. Experimentally a level of approximately 9 seems to work well. Figure 13 illustrates the result as the upper threshold is increased. Note the disappearance of some of the "important" contours as the threshold is raised to 30.

6.2.2 Logarithmic Processing

The eye is characteristically more sensitive to changes in black or dark areas than it is to changes in white or bright areas. The use of a constant high threshold, however, does not take account of this fact. Thus important contours in the darker areas, although subjectively equal to important contours in the bright areas, will not be found because the magnitude of the brightness change for these darker contours is below the fixed threshold. If the threshold level is lowered to catch the "important" contours in the darker areas, then there is the danger that either "texture" or noise in the brighter areas will be mistaken for contours.

One method of solving this dilemma uses an adaptive threshold based on the brightness of the immediately surrounding areas. An easier solution however is to change the original, linear picture, to a quasi-logarithmic picture using the transformation of Eq. 6.1, where p(x,y) is the original picture, "k" is a constant, "c_1" and "c_2" are constants, and p'(x,y) is the logarithmic picture.

$$\text{(Eq. 6.1)} \quad p'(x,y) = c_1 \log_2 (p(x,y) + k) + c_2$$

Since the values of p(x,y) can range from 0 to 255, the values of p'(x,y) must also range from 0 to 255, hence the need for "c_1" and "c_2". Although the darker areas are to be emphasized, if the constant "k" was not present, the emphasis would be excessive, thus "k" is chosen to be a positive number which will provide emphasis in the dark areas, but not so much that the brighter areas are overly compressed. In the pictures which follow, the range of brightness corresponding to levels 0 and 1 was expanded sevenfold. The values of all the constants are thereby fixed.

6.2.3 Curve Error Parameters

The curve error parameters, the weighting on the end points and maximum allowable error, affect the efficiency of the curve fitting process. Generally a weight of five and a maximum error of three was used in the pictures which follow. In one series of experiments using the portrait of the girl, it was found that for an error criterion of 3, a weighting factor of 5 produced the best results; that is, the fewest curves were needed to fit the contours at these values. See Table 3. As the allowable error increases there does not seem to be a definite minimum, although there does seem to be an asymptotic behavior. One problem which results from using a larger error value is a possible curviness in the curve fitting process which is subjectively undesirable. This is illustrated in Fig. 14.

6.2.4 Gradient Intensification

6.2.4.1 Pre-processing

Most processed pictures were 200 X 200 points square. One of the pictures was scanned at a resolution of 512 X 512 points of which only a 200 X 200 point section was used. Whether the higher or lower resolution was used, the contour was rarely a true step, but exhibited the characteristic of a step function convolved with an impulse response, characteristic of CRT scanners, consisting of a narrow Gaussian pulse plus "tails" due to the halo effect. Normally, the curve tracing procedure ignored the outer gradient components because of their low value, and therefore the gradients which were transmitted were smaller than they should have been. One available option was to pre-process the "x" and "y" gradients separately, so that these side lobes were added to the main or most significant gradient value. For example the gradients of Fig. 15(a) were changed as shown in Fig. 15(b).

The result of this procedure does not seem to justify its use. Subjectively, a little sharpening of the picture does occur; however this may also be accomplished as described in section 6.2.4.2. The desire not to use this procedure arises not because the same result can be achieved in another and perhaps simpler way, but because this method of pre-processing increases the number of curves required to fit the contours. See Table 4 and Fig. 16. Thus although the pre-processing sharpens the contours, it also introduces discontinuities into the contour structure with which the tracing algorithm is unable to cope.

6.2.4.2 Post-processing

Graham[12] has shown that by adding extra synthetic highs to the picture, a subjective sharpening takes place. This is similar to the result of the pre-processing of section 6.2.4.1. Graham's work indicated that a "highs" accentuation by a factor of 1.5 did not produce objectionable results while sharpening the image. Accentuating the "highs" which result from the curve fitting operation by more than a factor of 1.5 produces extreme contouring as shown in Fig. 17. As was mentioned earlier, the gradient values which are transmitted without pre-processing are smaller than they should be. Thus a picture in which the synthetic highs are not accentuated by about 25% is lacking in the high frequency component. This effect is demonstrated by the shadowing around the cameraman's coat in Fig. 17(a). Since the preprocessing of Section 6.2.4.1 adds the side lobe to the gradient, the "highs" of a preprocessed picture do not require accentuation.

6.2.5 Curve Continuation

It was presumed that the eye is very sensitive to discontinuities along a contour either in its spatial distribution or in its slope. Although this sensitivity is present, if the error criterion is sufficiently tight, the use of curves continuous in the first derivative can be abandoned. The result does not appear to be subjectively undesirable, while a savings of at least 10% in the number of curves needed to fit the contours can be expected. This savings of channel capacity results from the extra degree of freedom which the previous continued curves now have. An example of such a picture is shown in Fig. 18.

7. RESULTS

7.1 Channel Capacity

The results of the research are illustrated in figures 13, 14, 16, 17, 18, 19, and 21. Table 4 shows the type of processing used and the required channel capacity for each picture. The channel capacity was calculated in two different ways. The first method is that outlined in section 6.1. The results using this method are listed in Table 4 under the heading Channel Capacity/Curves. The lengths of the individual curves, however, indicate that for some picture processing methods, 10% or more of the curves may be less than five points long. Since 61 bits are required to transmit a starting curve and 39 bits are required to transmit a continued curve, the method described by Graham[12] for coding the gradient and contour information is advantageous for the shorter curves. Thus, the results of using a combination coder, one which uses one of two different codes, is shown in Table 4 under the heading Channel Capacity/Combination. Use is made of Graham's picture statistics to apply Huffman coding; therefore, the number of bits required for a beginning curve is 25 + 7(n-1) where n is the number of points on the contour. The fixed overhead of 25 bits is used to describe the starting point (eight bits for each coordinate) to communicate to the decoder the type of code which is used to describe the curve, (one bit), and to transmit the initial gradient component (five bits, four data bits plus one sign bit), and to fix the location of the next contour point (three bits). Note that the gradient is again assumed to require a precision of no more than 1 part in 16. Thus, without taking account of statistical correlations, 5 bits would be required to transmit the gradient information and 3 bits would b e required to indicate the direction of the next contour point. Some savings would be available because of the nature of contours, (see Graham's data); a savings of one bit per contour point will be assumed. If the curve is a continuation curve, the fixed overhead will only be nine bits since both starting coordinates are derived from the previous curve.

The required channel capacity for the combination coder is calculated by first observing the length of the curve. If the number of points fit by an initial curve is less than seven, the point by point code is used. Likewise, if the number of points fit by a continuation curve is less than six, the point by point code is used. The results are tabulated in Table 4.

Using the channel capacity/combination calculations the channel capacities required to transmit the picture of the girl range from 32,441 (Fig. 14(e)) to 42,991 (Fig. 16(d)) bits. The standard picture, defined to be the picture processed as in Fig. 22(b), requires 39,964 bits. Similarly the range for the close-up of the cameraman if from 19,944 (Fig. 14(d)) to 27,330 (Fig. 21(a)) bits. The close-up of the cameraman with standard processing (Fig. 22(a)) requires 25,952 bits. The range for the distant shot of the cameraman is from 20,778 (Fig. 13(d)) to 32,097 (Fig. 22(d)) bits. The standard for the distant cameraman picture is 26,800.

7.1.1 Effects of Various Methods of Processing on Channel Capacity

As the upper threshold is raised, thereby decreasing the number of contours which are transmitted, the required channel capacity decreases. See Fig. 13 and Table 4.

Pre-processing greatly increases the number of contours found with a corresponding decrease in the average length of the contours (Fig. 16). The number of curves needed to code these contours decreased but the required channel capacity increased. This implies an increase in the number of initial curves needed to code the pre-processed pictures.

Removing the requirement of a continuous first derivative at the end of one curve and the beginning of the next curve reduced the number of required curves; however, the channel capacity remains almost constant because of the extra information needed to specify the curves (Fig. 18).

The channel capacity required to transmit the pictures decreased markedly as the allowable error was increased from three to five. The decrease in channel capacity was less notable and the maximum error increased to seven (Fig. 14).

The combining of two processes had an additive effect. Increasing the maximum error and removing the requirement of a continuous first derivative resulted in a channel capacity almost equal to the average of the two processes separately. Where both processes individually increased the channel capacity ,when compared to the standard, the combination further increased the required channel capacity (Fig. 21(c)). When one process by itself, increased the required channel capacity vis-a-vis the standard, and the second process, by itself, decreased the required channel capacity vis-a-vis the standard, the combination of the two processes resulted in a channel capacity somewhere between their individual channel capacity (Fig 21(b)).

7.2 Subjective Effects

The subjective effects of the contour coding process should be separated into two groups. The first group contains those effects resulting from the nature of the process itself. The second group contains those effects resulting from the nature of some specific contour coding technique such as pre- or post-processing.

There seem to be four subjective effects due to the nature of the process itself. First, there is loss of texture. This occurs in such areas as the girl's face and hair and the foreground in the full-length picture of the cameraman.

Second, there appears to be a subjectivly undesireable effect from loss of shading, slow changes in brightness over a distance of three or more sample points. The contour coding procedure either sharpens this shading into one or more edges or ignores it completely.

Third, there is the effect due to quantization of the gradient direction. At the low resolution at which these pictures are printed, this tends to show up as "digital ratiness" in the picture. In the cameraman picture, where there are straight oblique lines, this effect is prominent especially where the lines are long.

Fourth, there is the effect due to errors in matching the end-point of continuous curves. The results are clearly seen in the full-length cameraman picture (e.g. Fig. 13(b)) as dark flares eminating from the dark areas into the whiter areas, especially around the head and elbow. This effect results from gaps in the fitted curves, areas where no "synthetic highs" are added, thereby leaving the spread out low-pass picture untouched.

Fifth, there seems to be an over-all effect, probably a combination of the previously described errors, to remove character from the picture. This effect is particularly prominent in the picture of the girl where the character of the facial expression is lost in the processed pictures.

As a result of pre-processing, numerous false contours may appear or existing contours may be over accentuated or "sharpened." See Fig. 16. The false contours occur as a result of the process illustrated in figures 15(a) and 15(b) where the five "x" gradient values are 9 and the pre-processing builds up a contour with a brightness change of forty-five levels where a gradual shading previously appeared.

The result of increasing the maximum allowable error affects the cameraman pictures significantly more than the picture of the girl. The long straight lines and naturally curved lines seem relatively unaffected; however, the shorter lines in the cameraman pictures and especially areas of high detail are often distorted.

The loss of detail that occurs as the upper threshold is changed (Fig. 13) demonstrates the loss of background detail. If the threshold had been raised further, the detail in the foreground would begin to disappear.

8. CONCLUSIONS

The questions which now must be answered are whether this work has any practical use and whether this approach to the "synthetic highs" technique should be continued. The answer to both questions is a qualified yes.

This investigation leads to the conclusion that contour coding has its place in image transmission, however, only in certain areas. The contour coding technique, as developed in this research, breaks down in areas of high detail because the contour trace routine erases the nearby gradients, destroying the detail. This can be a severe problem in a low resolution system such as the one used here since more areas in such a system fall in the category of detail easily noticed by the eye. The contour coding technique is also poor in areas of "texture." Some of the gradients in the "texture" areas may exceed the upper threshold and, thus, incorrectly cause a curve to be fit to that data.

To overcome the deficiencies in contour coding, it is desirable to isolate those areas where high detail, "texture", or very thin (one point in width) lines, a special type of detail, exist. These exceptions should be reported by a separate coding technique. The remaining areas can then be contour coded using a relatively low upper threshold since the noise in the scanned picture should be no greater than four levels, and there is no need to worry about areas of texture.

As a practical matter, not many receiving or transmitting stations have the computing capability or memory capacity to perform the indicated contour coding and reconstruction. The need for two-dimensional filtering and large-scale memories is only half the problem. The average time, twelve minutes, required to perform the simulation of one complete picture on the IBM 360/65/40 is the other half of the problem.

Notwithstanding the hardware and real time difficulties, the coding problem changes into a combination of pattern recognition and coding selection. The problem of locating the areas where different coding techniques will operate seems to become the major difficulty. In the facsimile area where a variety of codes were developed to transmit an entire document, methods were later developed to select one of a group of codes to encode different portions of a document.[23] Likewise, in two-dimensional grey tone pictures different codes will be advantageous in different areas. Therefore, areas of detail may be sent in an uncoded mode; then lines may be transmitted by applying curve fitting techniques to the sample point values instead of the gradient values; and areas of "texture" may be characterized by some special code.[24] The remaining areas will be transmitted by a contour coding technique. It is in this context that contour coding will provide significant reductions in channel capacity.

APPENDIX A

Gradient Grid Construction

A typical gradient grid as shown in figure 6 is used to present gradient and sample point information. Using the 3 X 3 point sample array shown in figure 7 for illustration, the corresponding gradient grid is as follows. Each sample point (x',y'), x', y' = 1,2,3 is located in the gradient grid at (2x',2y') = (x,y); thus, the sample points are located at (2,2), (2,4), (2,6), (4,2), (4,4), (4,6), (6,2), (6,4), and (6,6). The gradient information is then added in between the sample points. Thus, the gradient for sample point now located at (4,4), and given by (Eq. 3.2), is modified as in equation (A-1.1) where p(x,y) are the sample values.

$$\text{(Eq. A-1.1)} \quad \bar{G}(4,4) = \left[p(4,4) - p(2,4)\right]\bar{x} + \left[p(4,4) - p(4,2)\right]\bar{y}$$

The "x" component of the gradient G_x for the point (x,y) is added at location (x-1,y) and the "y" component of the gradient G_y, for the sample point (x,y) is added at location (x,y-1).

Thus, all sample points are located where both the "x" and "y" values are even integers; the "x" component of the gradient is located at points where the "x" value is an odd integer, and the "y" value is an even integer; and the "y" component of the gradient is located at points where the "x" value is even and the "y" value is odd. The locations where both the "x" and "y" values are odd are prohibited.

APPENDIX B

Fitting Quadratic Curves by a Least Squares Criterion

Methods for fitting a quadratic which has the formula $y=a + bx + cx^2$ to a set of data ((x_i, y_i), $i=1,n$) are well known. The solution for the particular case of a least mean square error error criterion is given below:

$$\begin{aligned}
&\text{(i)} \quad an + b\sum x_i + c\sum x_i^2 = \sum y_i \\
&\text{(ii)} \quad a\sum x_i + b\sum x_i^2 + c\sum x_i^3 = \sum x_i y_i \qquad \text{(Eq. B-1)} \\
&\text{(iii)} \quad a\sum x_i^2 + b\sum x_i^3 + c\sum x_i^4 = \sum x_i^2 y_i
\end{aligned}$$

APPENDIX C

Some Comments on the Output Pictures

Figure 12 shows the input pictures. The spatial resolution is 200 X 200 points; the video has 256 brightness levels. Since 64 brightness levels are usually considered adequate to display a picture without noticeable degradation from quantization, all comparisons of channel capacity are based on a 6-bit PCM picture. The channel capacity required for the 6-bit PCM picture is 240000 bits.

The pictorial output of the low-pass filter are shown in figure 20. Figure 23 represents the "x" component of the gradients by three levels. The brightest points represent gradient components whose values are greater than eight; the black points represent gradient components whose values are less than minus eight. The remaining gradient values are represented by the grey areas. Figure 19 shows the results of the curve tracing algorithm. The top halves of the pictures show the "x" gradient components of the contours after transmission but before reconstruction. The bottom halves show the "y" gradient components.

Figure 24 shows one type of low channel capacity picture. It is constructed by portraying a 50 X 50 point picture on a 200 X 200 point raster by repeatedly displaying each of the low resolution sample points at each point of a corresponding 4 X 4 point square in the higher resolution picture. These pictures, with a channel capacity of 25000 bits, can be compared with the pictures which result from the coding process.

Where appropriate, the channel capacities for the various pictures have been placed in parentheses beside the corresponding figure designations.

TABLE 1

Gaussian Filter

The Gaussian filter, $g(x,y)$ is circularly symmetric, and therefore only the one-dimensional version, $g'(x)$ is presented below.

$$g(x,y) = g'(x)g'(y) \quad \text{(Eq. T-1)}$$

$g'(i)$, $i=1,19$ is given below:

i	g'(i)
1	0.00807425
2	0.01373476
3	0.02194808
4	0.03294798
5	0.04646410
6	0.06155493
7	0.07660633
8	0.08956188
9	0.09836447
10	0.10148692
11	0.09836447
12	0.08956188
13	0.07660633
14	0.06155493
15	0.04646410
16	0.03294798
17	0.02194808
18	0.01373476
19	0.00807425

TABLE 2

Gradient Reconstruction Filter

The gradient reconstruction filter corresponds to the Gaussian low-pass filter shown in Table 1. Its construction is presented in Graham's thesis[12]. The reconstruction filter is symmetric about one axis and anti-symmetric about the other axis, therefore only one quadrant is shown below.

.0	.0	.0	.0	.0	.0	.0	.0	.00003
.0	.0	.0	.0	.0	.0	.0	.00024	.00014
.0	.0	.0	.0	.0	.0	.00096	.00072	.00018
.0	.0	.0	.0	.0	.00294	.00240	.00072	.00027
.0	.0	.0	.0	.00759	.00651	.00203	.00101	.00038
.0	.0	.0	.01706	.01517	.00472	.00269	.00134	.00050
.0	.0	.03415	.03121	.00944	.00588	.00335	.00167	.00062
.0	.06156	.05755	.01654	.01103	.00687	.00392	.00195	.00072
.10092	.09608	.02571	.01817	.01216	.00755	.00430	.00215	.00079
.24743	.03561	.02652	.01875	.01250	.00778	.00444	.00221	.00082

TABLE 3

Curves Required	Maximum Error	Weighting Factor
114	3	2
101	3	5
110	3	8
115	3	11
116	3	14
113	5	2
87	5	5
75	5	8
72	5	11
71	5	14
113	7	2
87	7	5
71	7	8
74	7	11
69	7	14

Relationship of the number of curves required to code a picture of the type Fig. 12(c) to the maximum allowed error and the weighting factor.

TABLE 4

Picture Statistics

Fig.	Max. error	Weighting factor	Post-Processing %	Pre-Processing	Log/ Lin	Upper Threshold/ Lower Threshold
13(a)	3	5	125	no	log	15/9
(b)	3	5	125	no	log	20/9
(c)	3	5	125	no	log	30/9
(d)	3	5	125	no	log	40/9
14(a)	5	9	125	no	log	20/9
(b)	7	12	125	no	log	20/9
(c)	5	9	125	no	log	20/9
(d)	7	12	125	no	log	20/9
(e)	5	9	125	no	log	20/9
16(a)	3	5	100	no	log	20/9
(b)	3	5	100	no	log	20/9
(c)	3	5	125	no	log	20/9
(d)	3	5	100	no	log	20/9
(e)	3	5	125	no	log	20/9
17(a)	3	5	100	no	log	20/9
(b)	3	5	125	no	log	20/9
(c)	3	5	150	no	log	20/9
18(a)	3	5	115	no	log	20/9
(b)	3	5	115	no	log	20/9
21(a)	3	5	115	yes	log	20/9
(b)	5	9	125	no	log	20/9
(c)	3	5	125	yes	log	20/9
22(a)	3	5	125	no	log	20/9
(b)	3	5	125	no	log	20/9
(c)	3	5	125	no	log	20/9
(d)	3	5	125	no	lin	20/9

TABLE 4 (cont.)

Fig.	Continued curves option	Contours/ Average length	Curves needed	Channel Capacity needed-(bits) Curves	Comb.	Graham*
13(a)	no	310/15	554	32066	30535	41880
(b)	no	215/20	489	28101	26800	37720
(c)	no	139/27	434	24878	23664	32523
(d)	no	95/34	366	21566	20778	28070
14(a)	no	215/20	369	22761	22372	37720
(b)	no	215/20	330	21020	20906	37720
(c)	no	147/24	356	21550	21295	31092
(d)	no	147/24	319	19909	19844	31092
(e)	no	358/16	551	32983	32441	50290
16(a)	no	478/9	487	30135	28576	42468
(b)	no	515/7	429	27609	26459	38255
(c)	no	515/7	429	27609	26459	38255
(d)	no	896/7	755	45405	42991	63782
(e)	no	896/7	755	45405	42991	63782
17(a)	no	215/20	489	28101	26800	37720
(b)	no	215/20	489	28101	26800	37720
(c)	no	215/20	489	28101	26800	37720
18(a)	yes	215/20	399	28089	27026	37720
(b)	yes	358/16	631	42241	40649	50290
21(a)	yes	515/7	406	28516	27330	38255
(b)	yes	358/16	506	34616	34015	50290
(c)	yes	478/9	449	31139	29578	42468
22(a)	no	147/24	480	27398	25952	31092
(b)	no	358/16	774	42494	39964	50290
(c)	no	215/20	489	28101	26800	37720
(d)	no	268/16	606	33852	32097	38590

* Calculated on a point-to-point basis as indicated in Graham's thesis[12].

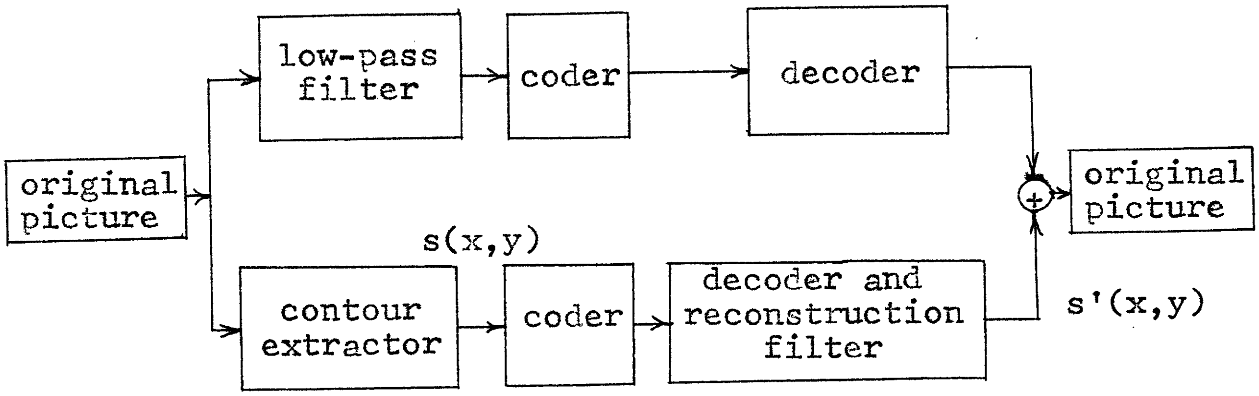


Figure 1
Block diagram of contour coding system

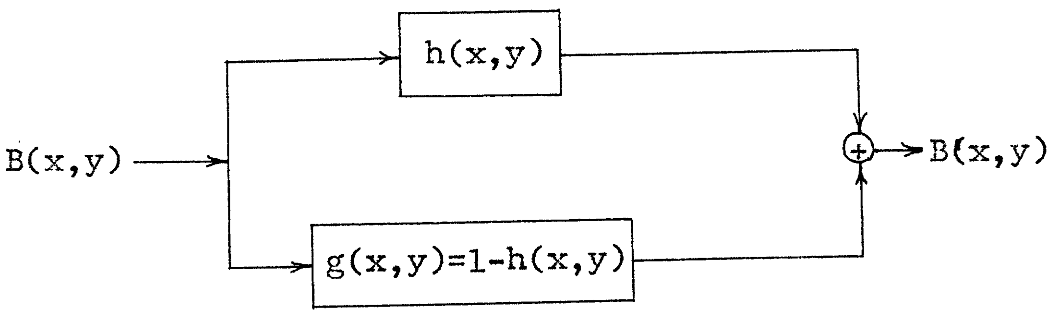


Figure 2
A distortionless system

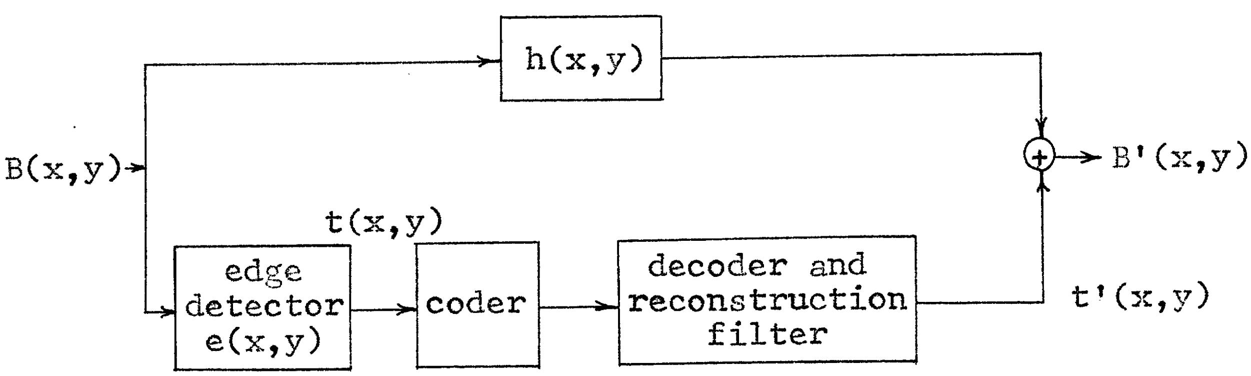


Figure 3
An approximation to the distortionless system

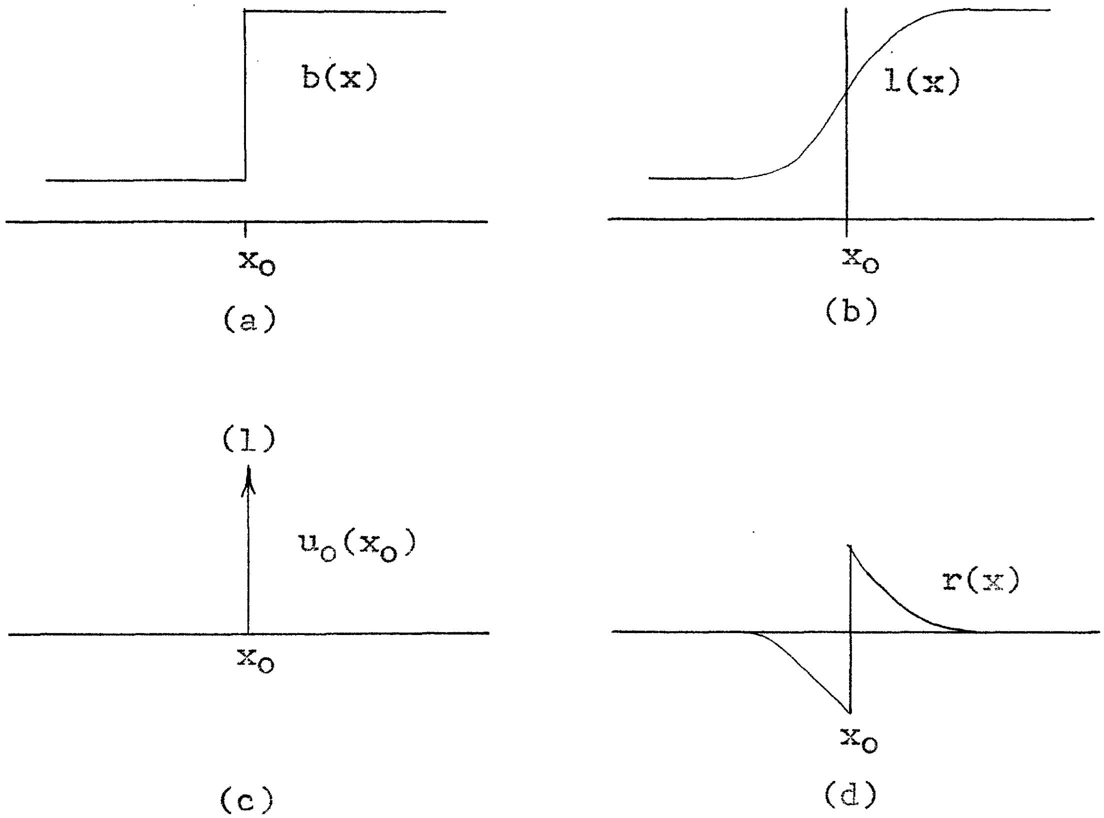


Figure 4
A one-dimensional view

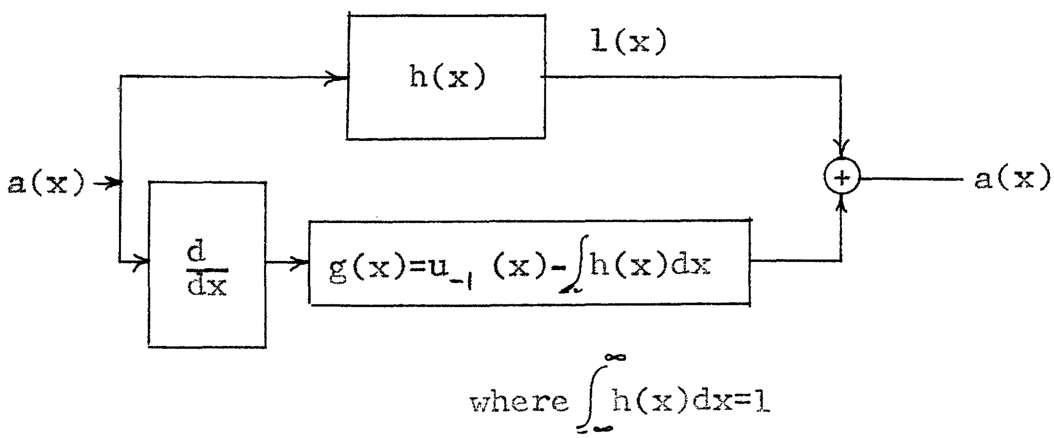


Figure 5
The one-dimensional system

	1	2	3	4	5	6
1		0		0		0
2	0	5	4	9	-8	1
3		0		0		0
4	0	5	4	9	-8	1
5		-4		-8		0
6	0	1	0	1	0	1

Figure 6
Gradient Grid

	1	2	3
1	5	9	1
2	5	9	1
3	1	1	1

Figure 7
Sample point grid

		1		
	8		2	
7		X		3
	6		4	
		5		

Figure 8
Direction determination

	1	2	3	4	5	6	7	8
1								
2			5					
3				-5				
4					5			
5						-5		
6							5	
7								

Figure 9
An example of gradient reconstruction

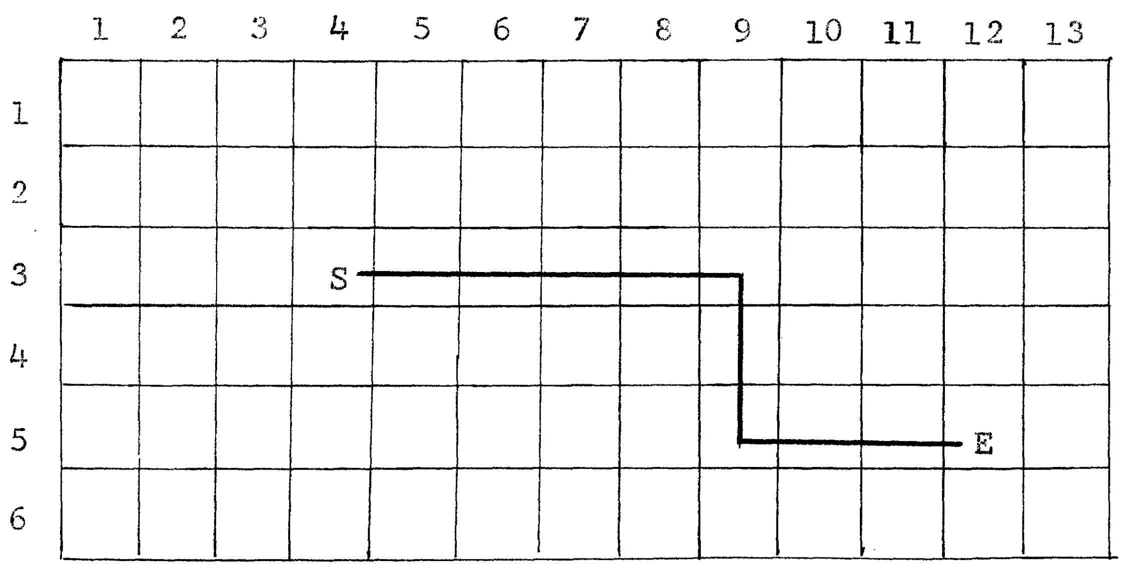


Figure 10
An example of gradient point connection

	1	2	3	4	5	6	7	8	9	10	11	12	13
1													
2													
3				5		5		5					
4									-5				
5										5		5	
6													

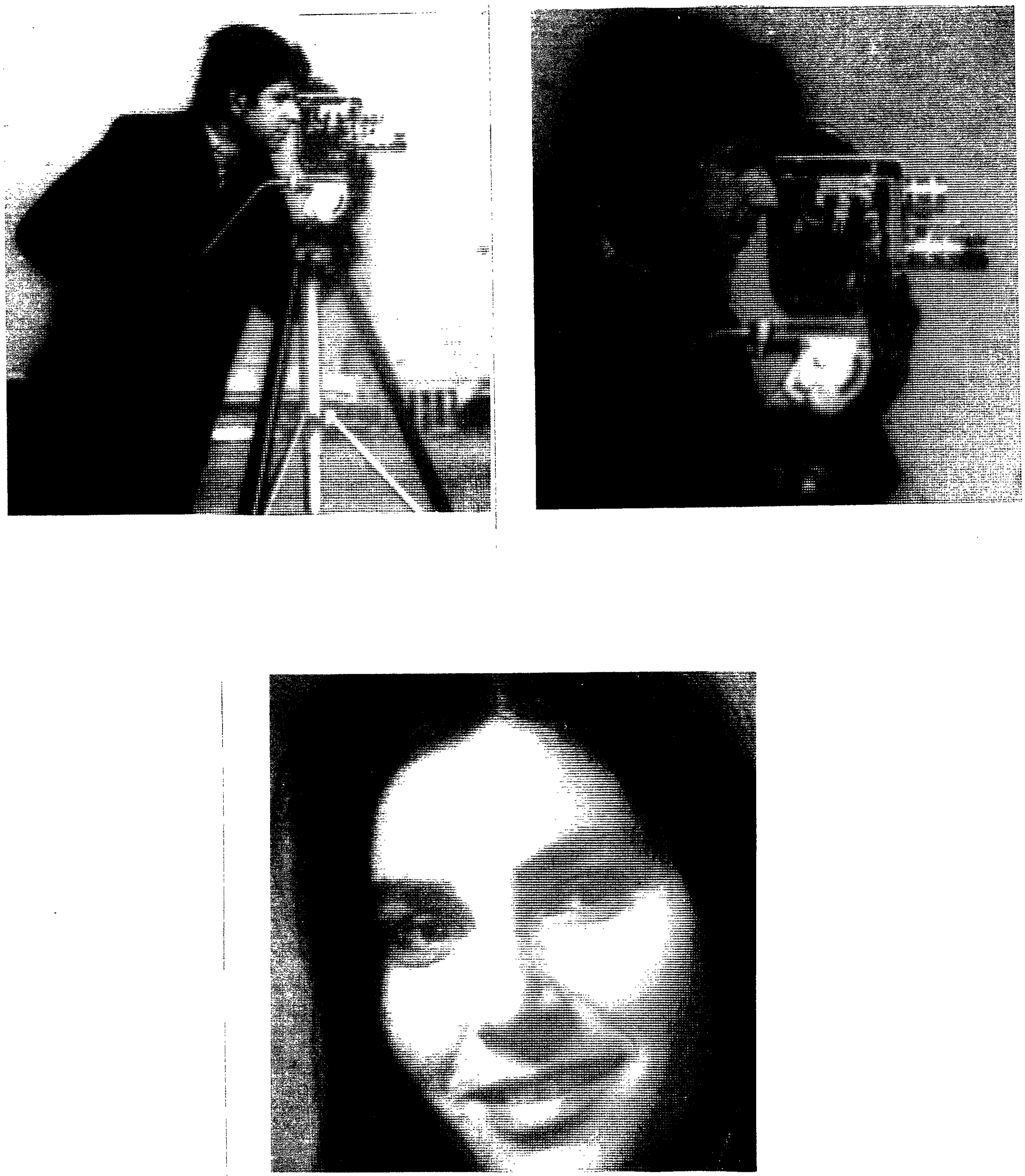
Figure 11
An example of gradient reconstruction

	1	2	3	4	5	6	7	8	9	10	11	12	13
1													
2			1		2		10		2		1		
3													
4			1		2		9		2		1		
5													
6					2		10		2		1		
7													

Figure 15(a)
Gradients before pre-processing

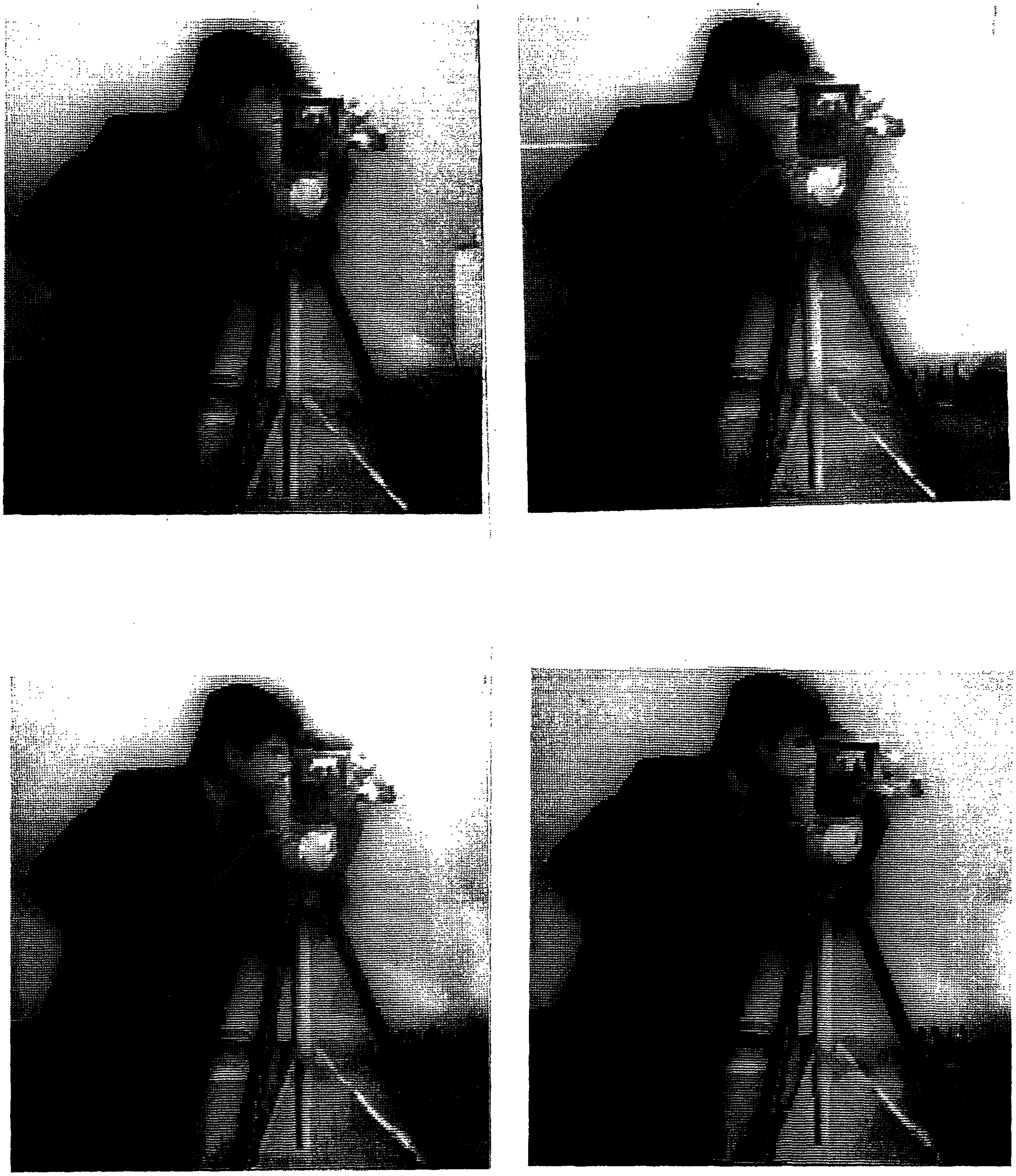
	1	2	3	4	5	6	7	8	9	10	11	12	13
1													
2							16						
3													
4							15						
5													
6							16						
7													

Figure 15(b)
Gradients after pre-processing



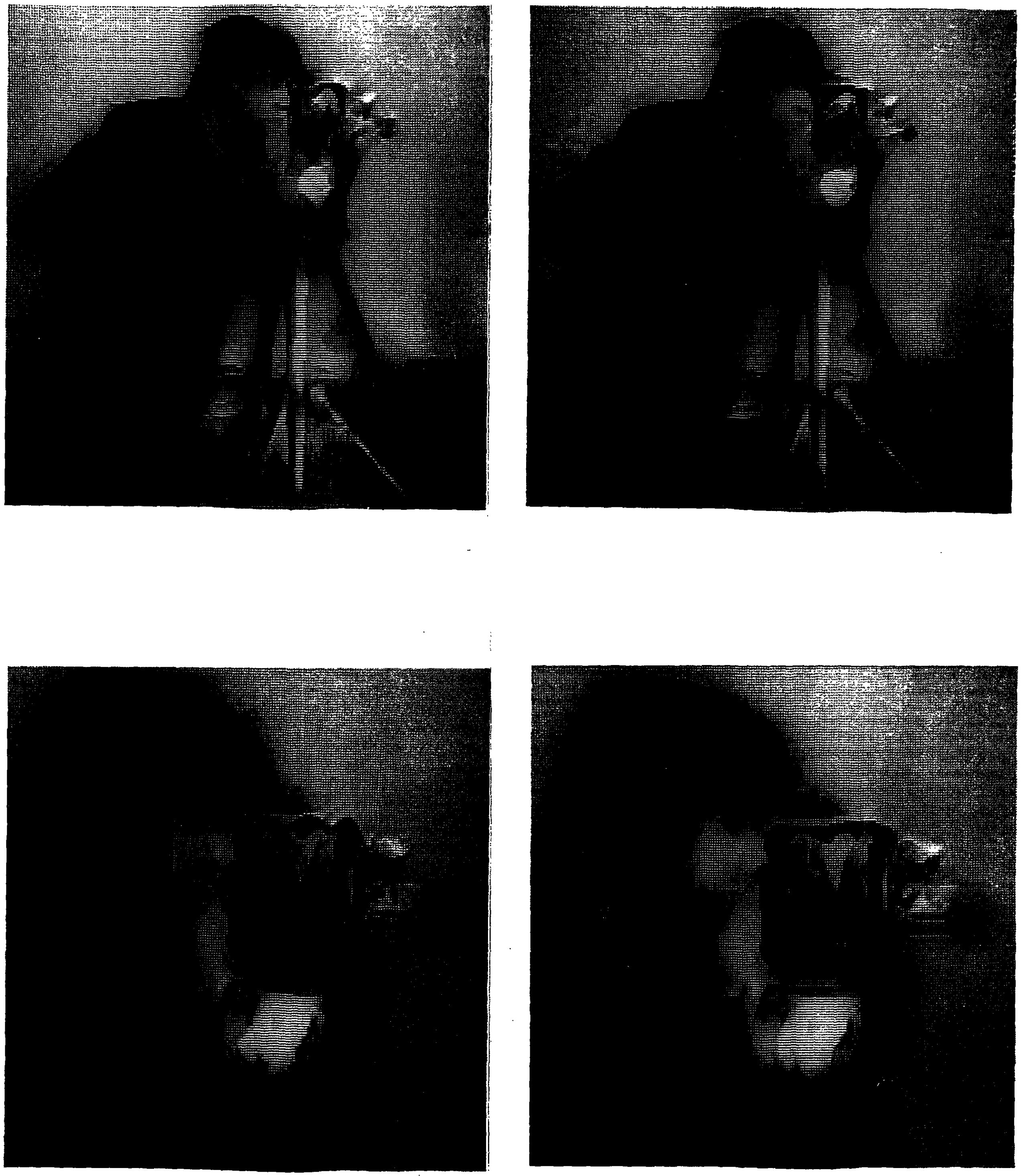
(a) (240,000 bits) (b) (240,000 bits)
(c) (240,000 bits)

Figure 12
Original Pictures



(a) (30,535 bits) (b) (26,800 bits)
(c) (23,664 bits) (d) (20,778 bits)

Figure 13
Pictures at Varying Thresholds

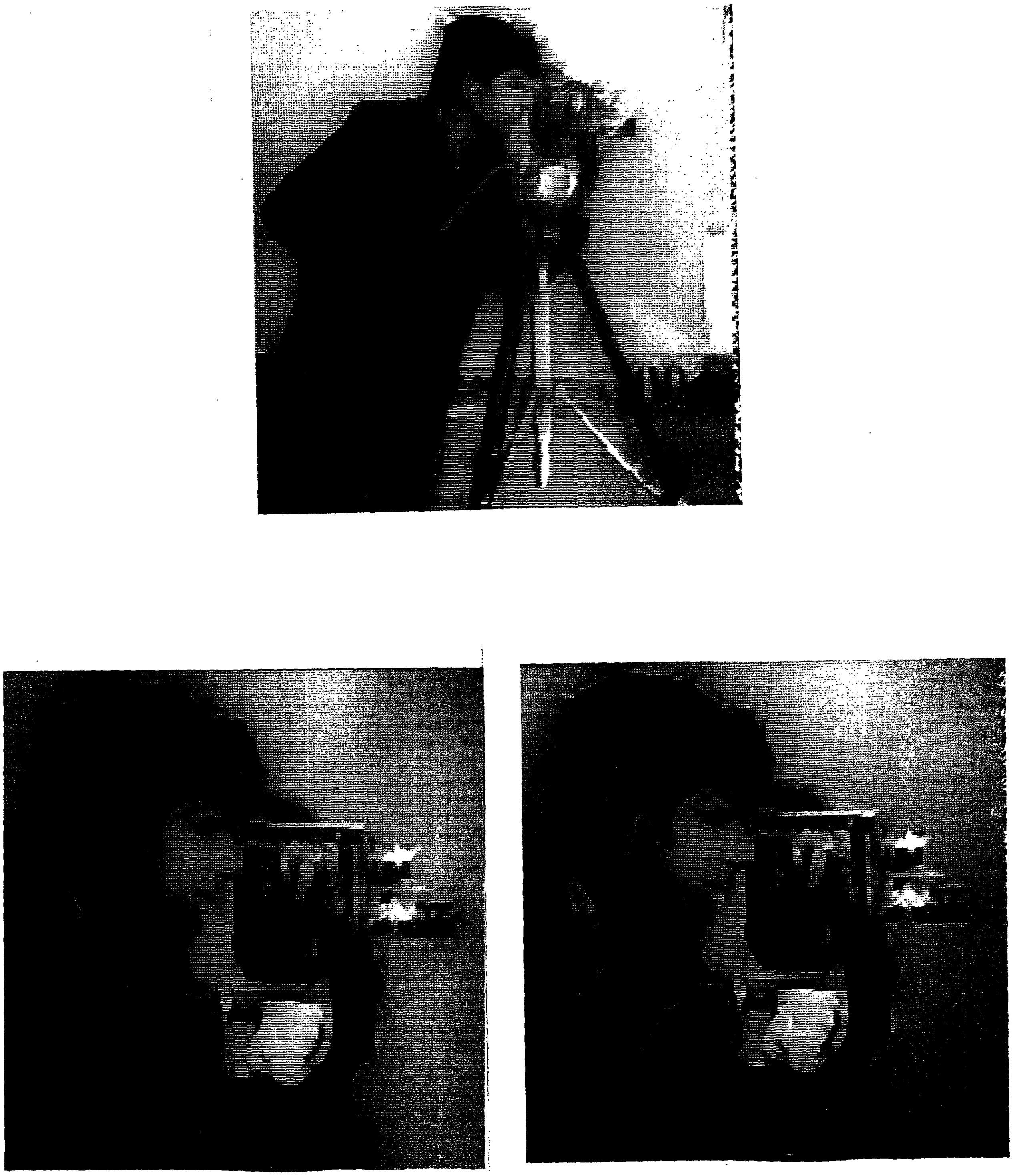


(a) (22,372 bits) (b) (20,906 bits)
(c) (21,295 bits) (d) (19,844 bits)

Figure 14
Pictures at Varying Maximum Errors

(e) (32441 bits)

Figure 14 (cont.)



(a) (28,576 bits)

(b) (26,459 bits)

(c) (26,459 bits)

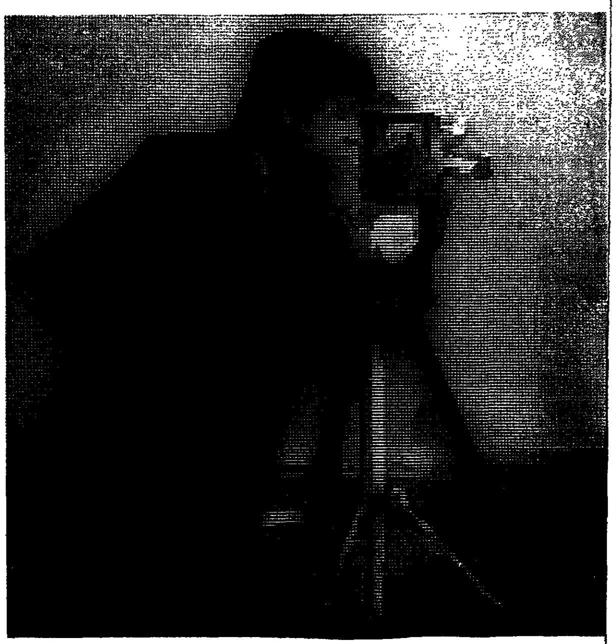
Figure 16
Pictures with Pre-processing

(d) (42991 bits)   (e) (42991 bits)

Figure 16 (cont.)

(a) (26,800 bits) (b) (26,800 bits)
(c) (26,800 bits)

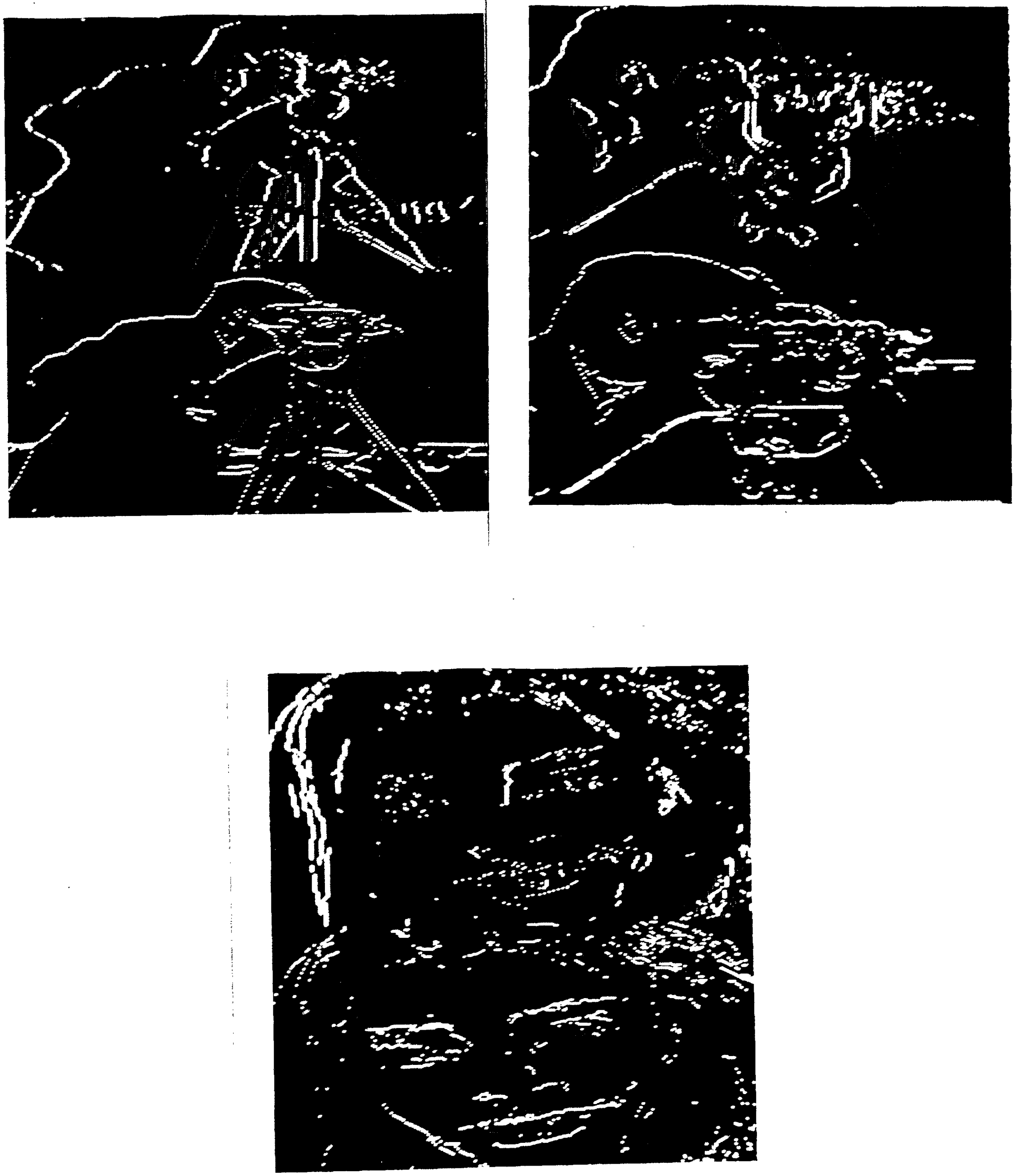
Figure 17
Pictures with Post-processing



(a) (27,026 bits)

(b) (40,649 bits)

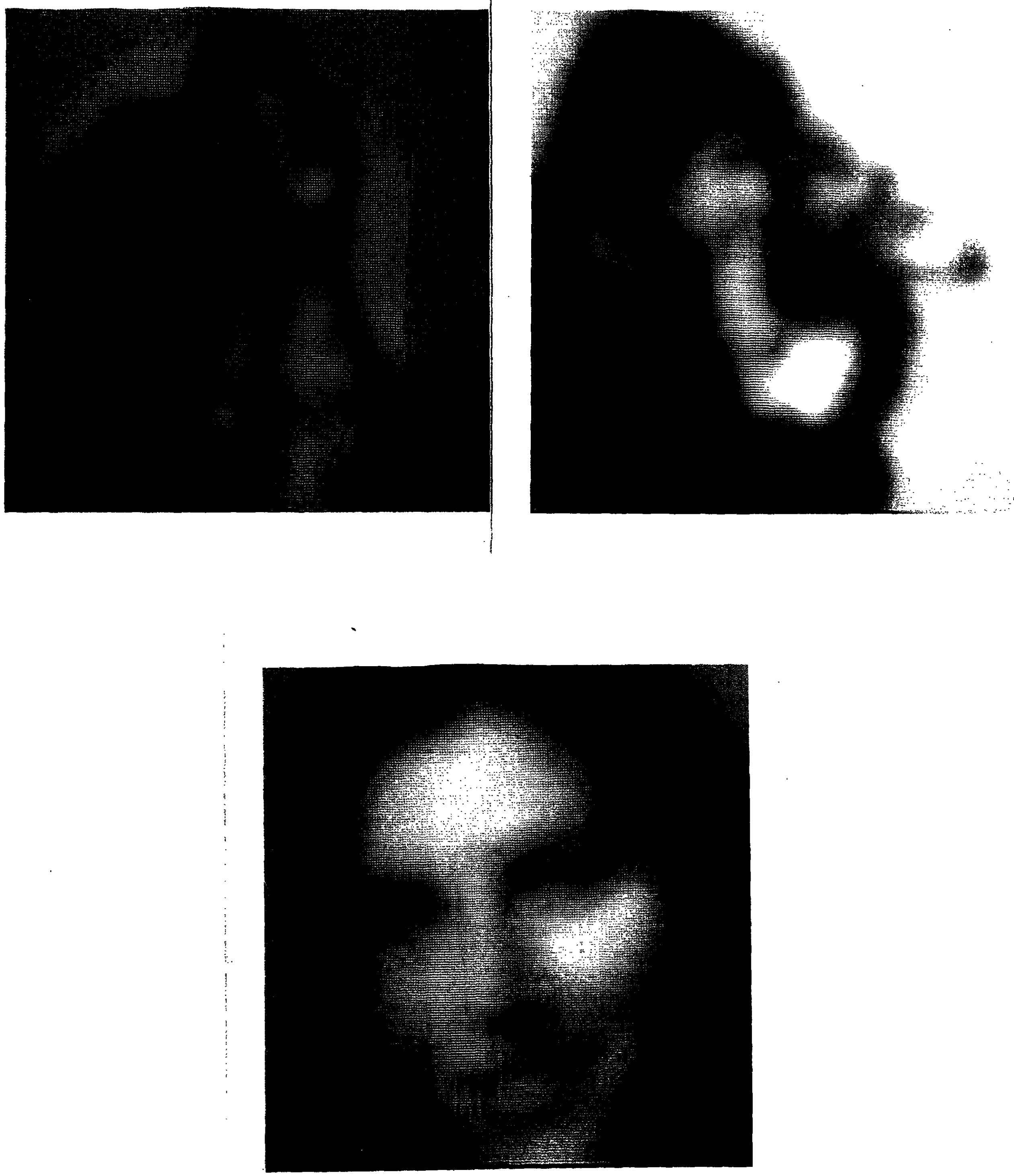
Figure 18
Pictures without Continuous First Derivatives



(a) (b)

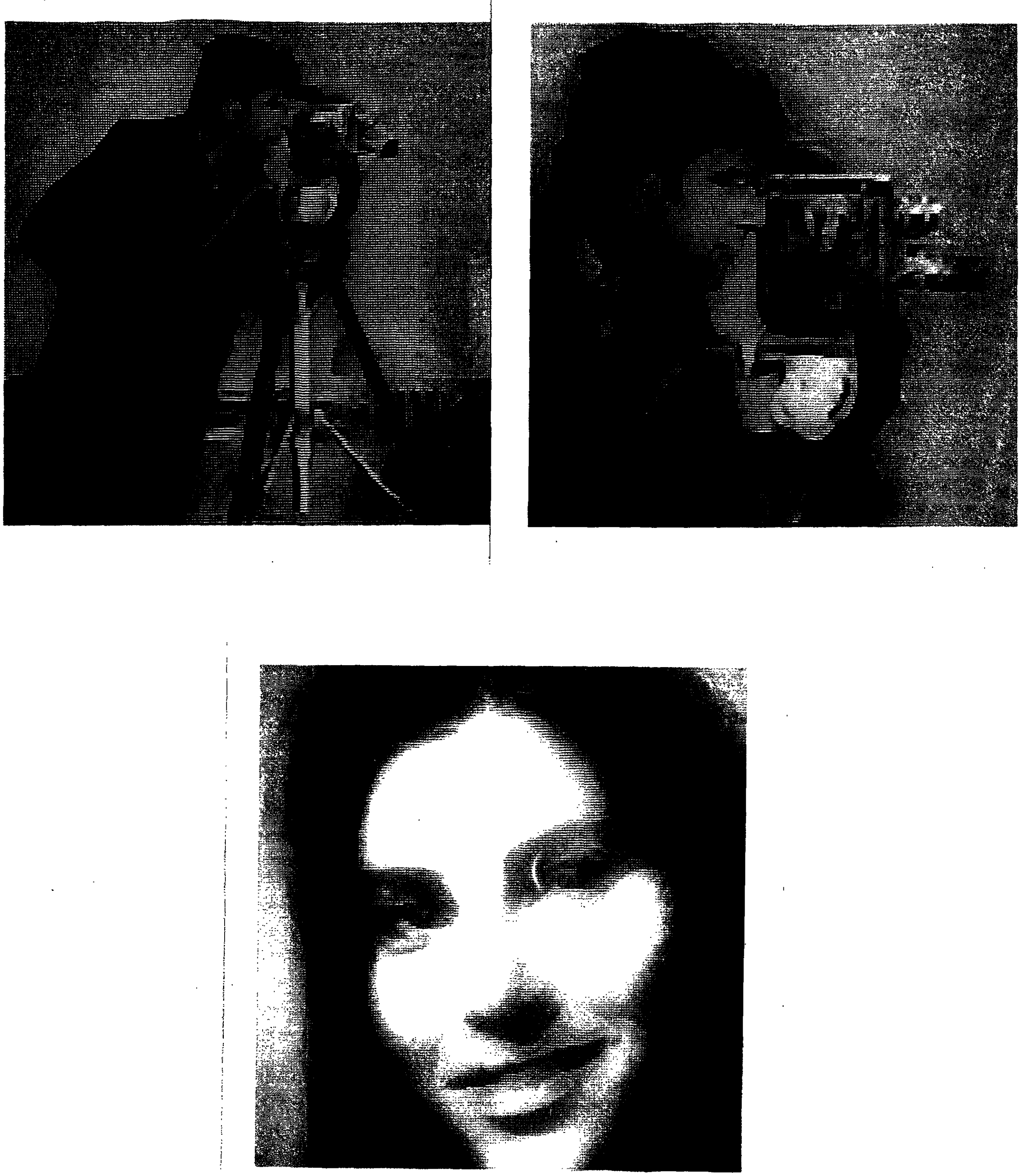
(c)

Figure 19
Results of Curve Tracing Algorithm



(a) (b)
(c)

Figure 20
Low-pass Pictures



(a) (27,330 bits) (b) (34,015 bits)
(c) (29,578 bits)

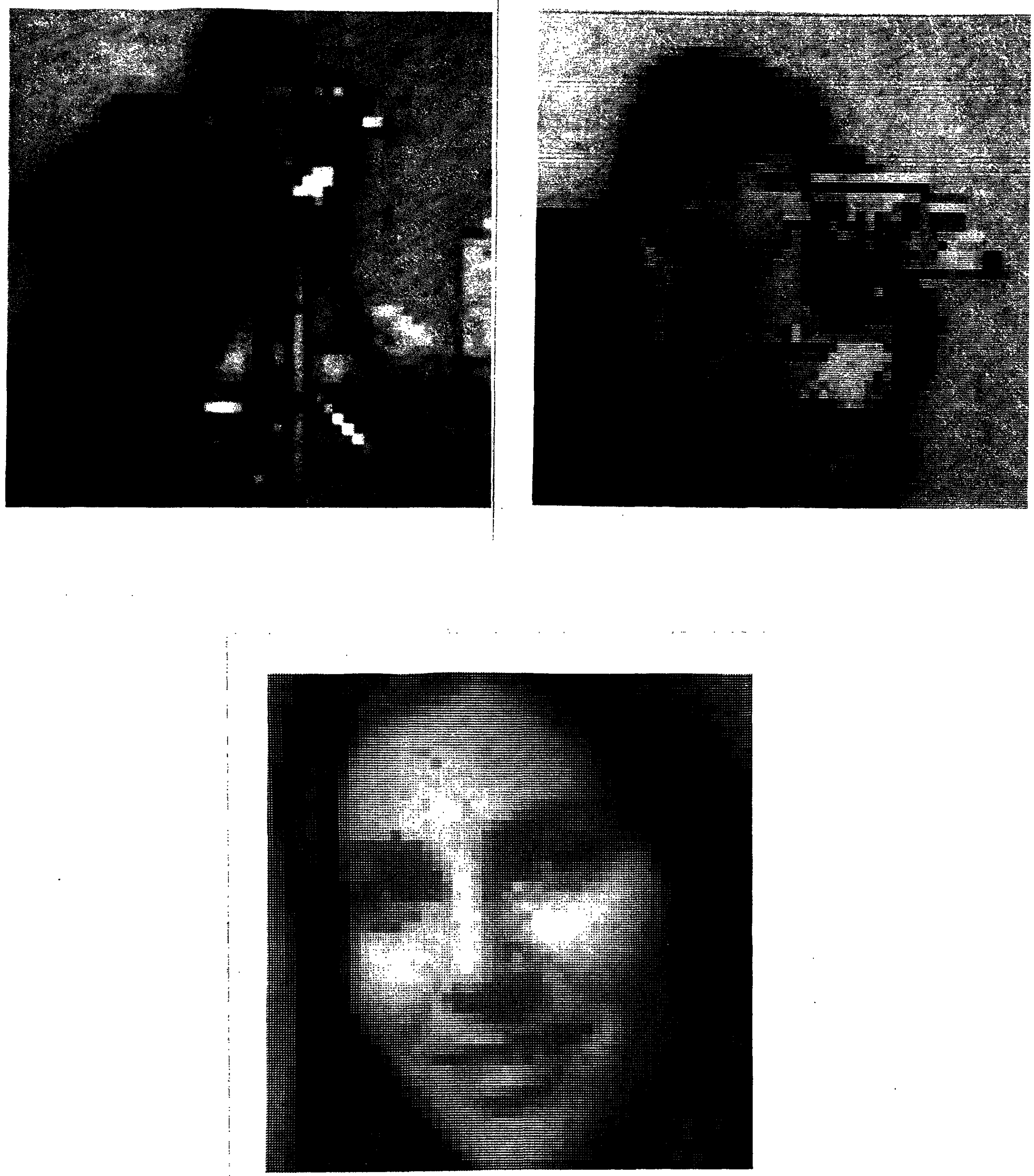
Figure 21
Pictures with Combination Processing

(a) (25,952 bits) (b) (39964 bits)
(c) (26,800 bits) (d) (32097 bits)

Figure 22
Standard Pictures- Log/Lin

(a) (b)
(c)

Figure 23
Pictures Representing "x" Gradients



(a) (25,000 bits) (b) (25,000 bits)
(c) (25,000 bits)
Figure 24
Low Channel Capacity Pictures

BIBLIOGRAPHY

1. Limb, J. O. and Sutherland, I. G., "Run-Length Coding of Television Signals", Proc. IEEE, Vol. 53, Feb., 1965, pp. 169-170.

2. Kagan, S. and Rodgers, F. A., "Method for Transmitting Digital Data in Connection with Document Reproduction System", United States Patent #3,347,981, Patented Oct. 17, 1967.

3. Gallager, R. G., Information Theory and Reliable Communications, John Wiley and Sons, New York, 1968.

4. Spencer, D. R., "Analysis of Facsimile Transmission by Bit-Plane Encoding", S.M. Thesis, Department of Electrical Engineering, September, 1968, MIT.

5. Kretzmer, E.R., "Statistics in Television Signals", Bell System Technical J., Vol. 31, July, 1952, pp. 751ff.

6. Michel, Fleckenstein, and Kretzmer, "A Coded Facsimile System", IRE Convention Record, 1957, Part 2, p. 84.

7. O'Neal, J. B., Jr., "Predictive Quantizing Systems (Differential PCM) for the Transmission of Television Signals", BSTJ, Vol 45, No. 5, May-June, 1965, p.689.

8. Ratliff, F., Mach Bands, Holden-Day, 1965.

9. Neuburger, A., "Characterization of Pictures by Essential Contours", S.M. Thesis, MIT, Department of Electrical Engineering, January, 1966.

10. Schreiber, W. F., Knapp, C. F., and Kay, N. D., "Synthetic Highs - An Experimental TV Bandwidth Reduction System", SMPTEJ, Vol. 68, 1959, p. 525.

11. Pan, J. W., "Reduction of Information Redundancy in Pictures", Sc.D. Thesis, MIT, Department of Electrical Engineering, September, 1962.

12. Graham, D. N., "Image Transmission by Two-Dimensional Contour Coding", Ph.D. Thesis, MIT, Department of Electrical Engineering, September, 1966.

13. Julesz, B., " A Method of Coding Television Signals Based on Edge Detection", BSTJ, July, 1959.

14. Gouriet, G. G., "Bandwidth Compression of a Television Signal", Proc. I.E.E., Vol 104, Part B, May, 1957, p. 265.

15. Cherry, E. C., and Gouriet, G. G., "Some Possibilities for the Compression of Television Signals by Recoding", Proc. I.E.E., Vol. 100, Part C, January, 1963, p. 9.

16. Huffman, D. A., "A Method for the Construction of Minimum Redundancy Codes", Proc. IRE, Vol. 40, 1952 pp. 1098-1101.

17. Meyer, P. L., Introductory Probability and Statistical Applications, Addison-Wesley, Reading, Mass., 1965.

18. Mack, C., Essentials of Statistics for Scientists and Engineers, Plenum Press, New York, 1967.

19. Hoel, P. G., Introduction to Mathematical Statistics, John Wiley and Sons, New York, 2nd Ed., 1962.

20. Quinlan, R.V., "Information Transmission System and Method", United States Patent #3,461,231, Patented Aug. 12, 1969.

21. De Groat, P. H., "Selective Encoding Technique for Bandwidth Reduction in Graphic Communication Systems", United States Patent #3,483,317, Patented Dec. 9, 1969.

22. Chatelon, A. E. J., "Reduced Time Transmission System", United States Patent 33,378,163, Patented Nov. 11, 1969.

23. Wernikoff, R., Epstein, P., and Schreiber, W. F., "Method of and Apparatus for Code Communication", United States Patent #3,394,352, Patented July 23, 1968.

24. Anderson, G. B., Ph.D. Thesis, MIT, Department of Electrical Engineering, Feb., 1970.

25. Schreiber, W.F., "Picture Coding", Proc. IEEE, Vol. 55, No. 3, March 1967, pp. 320-330.

26. Schreiber, W. F. and Knapp, C. F., "TV Bandwidth Reduction by Digital Coding", IRE Convention Record, Vol. 6, Part 4, 1958, pp. 88-99.

27. O'Brian, V., "Contour Perception, Illusion, and Reality", JOSA, Vol. 48, No. 2, Feb., 1958, p. 112.

28. Roberts, L. G., "Machine Perception of Three-Dimensional Solids", Ph.D. Thesis, MIT, Department of Electrical Engineering, 1963.

28. Lowery and DePalma, "Sine Wave Response of the Visual System, I. The Mach Phenomenon", JOSA, Vol. 51, July, 1961, p. 740.

29. Seyler, A. J., "Detail Perception after Scene Changes in Television Image Presentations", Proc. IEEE, IT.-11, p. 31, 1965.