# Subpixel estimation of straight lines on noisy data

C Dong and P C K Kwok
Department of Computer Science
The University of Calgary
Calgary, AB, Canada T2N 1N4

## Abstract

A method for determining the position of straight lines in a grayscale image to subpixel accuracy is presented. It is based on Newton Iteration and is superior to other existing methods in terms of accuracy. It also guarantees convergence to the correct solution while other methods do not. The performance of the method in the presence of noise is evaluated. It is found that the method performs better than others in all cases.

## 1 Introduction

Digitization of documents, and in particular, of line drawings from hard copies is an important application of image processing. Digitization partitions the analog image into compact and convex subsets of points—pixels. Each pixel is assigned a gray value which represents the intensity of the points in that pixel. Usually, an edge detector locates an edge as a chain of pixels. The reconstruction of the original edge from the chain is performed by connecting the centres of pixels with straight line segments. The accuracy is limited by the resolution of the image and the reconstruction is considerably different from the original.

In recent years, there has been some interests towards the use of grayscale information to determine the location of edges[1-8]. This enables a better reconstruction of the original edges. Two approaches have been used to obtain subpixel edge estimation: (1) the derivative methods and (2) the structural methods. They are described in the next section.

## 2 Derivative methods

In a derivative method, a real valued function is first fitted to a set of pixels, and the edges are located according to the derivatives of that function. The derivative operators used can be classified into two broad classes: first derivative operators and second derivative operators. the corresponding methods are regarded as first derivative method and second derivative method respectively. First derivative operators respond with a broad peak at an edge location. Second derivative operators, on the other hand, respond with a zero-crossing at an edge location. Thus, if edges are related to the positions of such peaks or zero-crossings, then their locations can be computed to better than pixel accuracy.

Several first derivative methods have been developed. Fang and Huang[1] presented a method for locating the peak at the maximum point of the second-order polynomial function fitted to a set of values from the first derivative operator. Nomura *et al*[2] introduced a normal-distribution curve instead of a quadratic-curve in their method. It was claimed to be more suitable for a sharper edge than [1]. MacVicar-Whelan and Binford[3] developed a second derivative method where a Lateral inhibition operator was utilized. The zero-crossing point at a straight line obtained by linearly interpolating the second derivative of two adjacent pixels was considered to be the position of an edge. Huertas and Medioni[4] used a standard second derivative operator — Laplacian-Gaussian operator(LoG). Zero-crossings with pixel precision were first located according to LoG. Then a parametric polynomial function[9] was fitted to create a finer grid, on which zero-crossings were extracted again. It is shown that in a low noise environment the zero-crossing can be estimated with subpixel accuracy.

## 3 Structural approach

A function fitting approach makes the assumption that the image has spatially continuous differentiable intensities. This assumption does not always hold. Consider a scene that is made up of objects against a high contrast background. One would expect discontinuities near the boundary of the object from the digitized image. In such cases, structural methods are applicable. Assuming the edge to be recovered is a straight line or locally linear, a structural method estimates the parameters of a line defining the edge which best predicts the grayscale intensities of the pixels.

Hyde and Davis[5] first presented a structural approach in order to achieve a better estimation. They improved the simple least-squares method by making use of the grayscale values of the pixels. An iterative algorithm was proposed to solve the least-squares problem. Unfortunately, it yielded little improvement over a simple least-squares method.

Kiryati and Bruckstein[6] derived a set of simultaneous equations for a straight line. The intersection point of the equations identifies the parameters for the line. Theoretically, it was shown that error-free reconstruction of the original straight line was possible when the grayscale values were noise-free and unquantized. However, no methods were offered to solve the simultaneous equations.

We have developed a structural method for subpixel estimation[8]. Experiments were conducted in both straight line recovery and piecewise recovery of curves by applying our method to artificially generated images, where noise and quantization were not considered. Our method was demonstrated to be superior to other existing ones in terms of accuracy in both cases. This paper presents the experimental results of our method taking noise and quantization errors into account. Similar analyses were performed on the simple least-squares method and Hyde and Davis's method. The performance of the three methods were compared under the influence of noise and quantization errors.

# 4 Formulation of the subpixel estimation method

Consider a scene that is made up predominately of straight lines against a high contrast background. The scene is digitized using a scanner or camera. The result is a rectangular array of pixels with grayscale values. It is assumed that the lines are at least $3\sqrt{2}$ units in thickness. Under ideal lighting conditions, the background is at a constant grayscale value and the pixels completely covered by a line will be at another grayscale value. Near the edges, the grayscale value of a pixel is a function of the overlap between the straight line and the pixel. The exact relationship between the grayscale value and the overlap is dependent upon the scanner or the digitizer. It can be established by calibration.

In any case, the response of a sensor is assumed to be isotropic so that the grayscale value is a function only of the distance of the edge from the centre of the sensor, irrespective of the orientation.

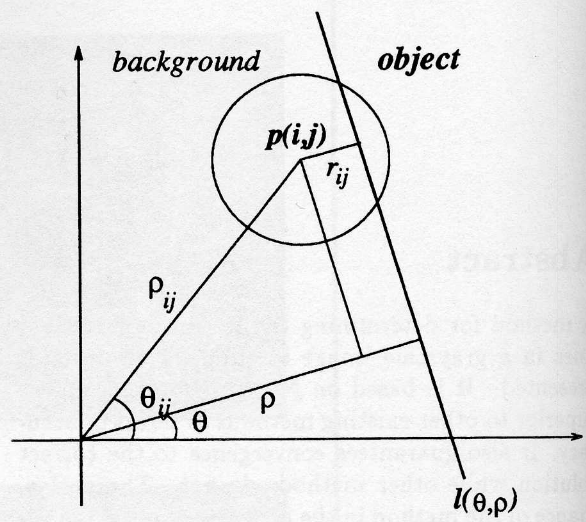


Figure 1: The edge of a straight line $l(\theta, \rho)$ overlapping with a pixel.

Figure 1 shows the edge of a straight line that divides the image into the object region and background. A pixel is represented by a circular disk. The grayscale value $g(i, j)$ of a pixel $(i, j)$ is assumed, for simplicity, to be equal to the distance $r_{ij}$ between the edge and the centre of the disk. The parameters $\theta$ and $\rho$ satisfy

$$\rho = r_{ij} + \rho_{ij} \times cos(\theta - \theta_{ij}) \qquad (1)$$

In Kiryati and Bruckstein's formulation, each pixel along the edge is associated with an equation in the $\theta/\rho$ space of a form given by equation (1). Since there are two unknowns in the equation, at least two pixels will be needed to define the straight line uniquely. In practice, many pixels may be partially covered by the same straight line. The equations for the pixels should intersect at a common point. The intersection point therefore identifies the parameters for the straight line. Unfortunately, since the equation involves a Cosine function, it is very difficult to find the intersection points of a set of Cosine functions with different offsets and amplitudes.

# 5 A method to reconstruct the straight line

We therefore propose to find the intersection using a different formulation. During reconstruction, the values of $\rho_{ij}$, $\theta_{ij}$ and the actual grayscale $g(i,j) = r_{ij}$ are given for every pixel $(i,j)$ partially covered by the line along the edge.

We place a test line with parameters $\theta$ and $\rho$. The test line gives rise to a grayscale value of

$$G(\rho,\theta,i,j) = \rho - \rho_{ij} \times \cos(\theta - \theta_{ij}) \qquad (2)$$

for a pixel $(i,j)$. We call $G$ the predicted grayscale value. If the test line is the same as the original line, there will be zero error between the actual value $g$ and the predicted value $G$, for all pixels along the edge. This is equivalent to the minimization of the sum of squared errors:

$$S(\theta,\rho) = \sum_{k=1}^{N}(g(i_k,j_k) - (\rho - \rho_{i_k j_k} \times \cos(\theta - \theta_{i_k j_k})))^2 \qquad (3)$$

The minimization of $S(\theta,\rho)$ is obtained by setting the derivatives to 0, and it is solved by Newton's method.

There are two methods to determine the initial values $\theta_0$ and $\rho_0$. In the first method, we can fit a least-squares straight line against the set of edge pixels, as suggested by Hyde/Davis. This will give the initial values sufficiently close to the correct result to warrant convergence. We have conducted experiments on 50,000 test cases of randomly generated edges and in every case, it converges to the correct result. This method of determining the initial values requires extra computations for the least squares straight line.

Figures 2 and 3 show an example of an edge used in the experiment. The results of three methods are shown: least-squares without gray value information (SLS), Hyde/Davis and the new method. The line determined by SLS was used as the initial values for the other two methods. In figure 2, 14 edge pixels were used to reconstruct the straight line. The differences among the three methods are not significant, but the new method is the only one providing error-free recovery. Figure 3 shows the result when the experiment was conducted on a $3 \times 3$ window. Only four edge pixels were involved in the estimation and the differences between the new method and the other two are very significant.

From experiments, we have found that the objective function (equation 3) is very sensitive to $\theta$ but not very sensitive to $\rho$. When $\theta_0$ is sufficiently close to the correct value, both $\theta$ and $\rho$ always converge to the correct values regardless of $\rho_0$. Otherwise, convergence to the correct result is not guaranteed.

For noiseless data, $S(\theta,\rho)$ should be zero if the result is correct. In practice, because of computation error, etc., $S(\theta,\rho)$ is compared with a predetermined threshold to determine whether the conrrect result has been attained. Thus in the second method, we perform a search in a one dimensional space for an appropriate $\theta_0$, while using an arbitrary value for $\rho_0$.

## 6 Subpixel Estimation on Noisy Data

For noisy case, a straight line is constructed locally to a $3 \times 3$ window. The objective function $S(\theta,\rho)$ is defined in the same form as equation (3) over the edge pixels in that window. But $S(\theta,\rho)$ will not be zero even for the best fitting and it is impossible to predetermine a threshold. Thus, the searching of $\theta_0$ is performed through the domain from $-\pi$ to $\pi$. The minimum of the minimum $S(\theta,\rho)$ gives the best fitting line.

## 7 Noise and Quantization

The gray level intensity at a pixel depends on many factors. Furthermore there may be errors or noise introdced during the digitization and quantization processes.

The noise $x$ is assumed to be uniformly distributed with mean 0 and variance $\sigma^2$. The absolute value of $x$ added to a pixel is supposed to be less than 10% of the exact output from an error-free digitizer at that pixel[1]. Therefore, the digitized image $g^n$ is

$$g^n(i,j) = g^a(i,j) + x(i,j)$$

where $g^a(i,j)$ is the accurate gray value at $g(i,j)$.

The variables $g(i,j)$ (either $g^n$ or $g^a$) are quantized uniformly, with the exception that the values are 0 or $\pi$. If $b$ bits are available for the quantization of $f(i,j)$, then the output of the digitizer are represented by $2^b$ levels. Let $g^q(i,j)$ denote the quantized value of $g(i,j)$, the maximum quantization error is

$$\delta g = max|g^q(i,j) - g(i,j)| = \frac{\pi}{2^b - 2}. \qquad (4)$$

$b$ is taken from 3 to 7 in the tests.

[1] Suppose $f$ is the accurate output,

$$\sigma^2 = \int_{-0.1g}^{0.1g} x^2 \times \frac{1}{0.2g} dx = \frac{1}{300} g^2$$

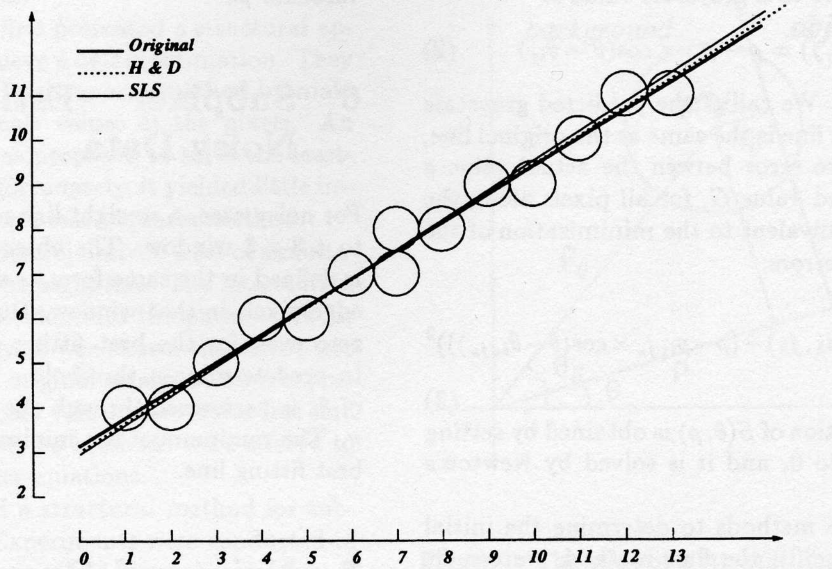


Figure 2: Straight line recovery

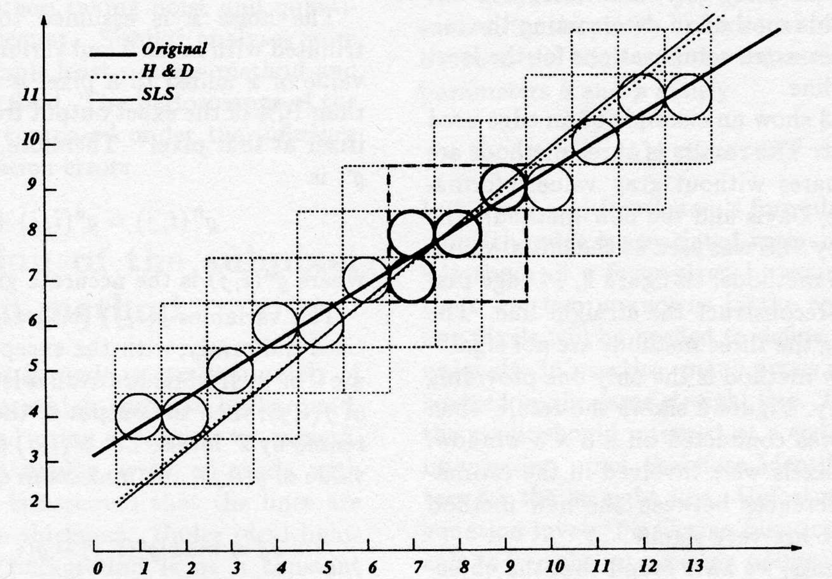


Figure 3: Straight line recovery from a 3 × 3 window

(Estimation using 4 pixels in the highlighted 3 × 3 window)

## 8 Performance Measures

The acccuracy of the method is estimated by the difference between the original and the reconstructed images. The most straightforward measure for the accuracy of the method is the value of the objective function. A better fitting line results in a smaller value of $S$. For a fair comparision, the measure is normalized to a per pixel basis:

$$\eta = \frac{1}{N}\sum_{k=1}^{N}(g(x_k, y_k) - G(x_k, y_k))^2, \qquad (5)$$

where $(x_k, y_k)$ are the coordinates of edge pixels.

In the case of straight line recovery, with the knowledge of the equations of original lines, the accuracy of the methods can also be estimated by the difference in the equation parameters between original and recovered lines: $d_\theta$ and $d_\rho$. Since the test lines are generated randomly, the recovery is accomplished by applying the methods to a randomly picked window along the line. Different positions of windows along the same line give different $d_\rho$s even when the values of $d_\theta$ are similar. Thus, we only consider $d_\theta$ as a measurement.

## 9 Results on Noisy Data

To evaluate the performance of the new methods, we also implemented the method of least-square without gray value information (SLS) and Hyde/Davis's method for comparison. It is not possible to make a comparison with Kiryati/Bruckstein's method, since they gave a formulation of the problem without solving it.

### 9.1 Noise

Figures 4 and 5 show the results on 8 sets of noisy lines. The $x$-axis is the number of lines in each set. The $y$-axis is the error measure in the form of $\eta$ and $d_\theta$ respectively. The values shown in the figures are taken as the average over the number of lines in each set. It is obvious our method is superior to the other two, though we used only a $3 \times 3$ window in our method while a $7 \times 7$ window is used in the other two methods.

### 9.2 Quantization

Similar experiments were conducted on straight line recovery when the grayscale values are quantized to a limited accuracy (3 to 7 bits). 8 sets of lines were tested for each quantization. In all cases, the proposed method has the best preformance. Quantization on both noise-free and noisy data were considered. Figures 6-9 are the results for the set of 12800 lines. A better estimation is obtained by our method in all the cases.

## 10 Conclusion

A method for the reconstruction of straight lines and curves from a grayscale image to subpixel accuracy has been presented. The method was tested using a large set of pixels as well as a small window of pixels, in the presence of noise. Its performance was compared with the other methods. It was shown that the new method performs better than the other methods in all cases. Furthermore, the other methods may not converge to the correct solution.

## 11 Acknowledgement

This work is supported by the Natural Sciences and Engineering Research Council of Canada.

## References

[1] J. Fang and T. S. Huang. Some Experiments on Estimating the 3-D Motion Parameters of a Rigid Body from Two Consecutive Image Frames. *IEEE Trans. Pattern Analysis and Machine Intelligence* **6**, 545-554, 1984.

[2] Y. Nomura, M. Sagara, H. Naruse and A. Ide. Edge Location to Subpixel Precision and Analysis. *Systems and Computers in Japan* **22**(9), 70-80, 1991.

[3] P. J. MacVicar-Whelar and T. O. Binford. Intensity Discontinuity Location to Subpixel Precision. *Proc. 7th Int. Joint Conf. Artificial Intelligence*, 752-754, 1981.

[4] A. Huertas and G. Medioni. Detecting of Intensity Changes with Subpixel Accuracy using Laplacian-Gaussian Masks. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **8**, 561 - 564, 1986.

[5] P. D. Hyde and L. S. Davis. Subpixel Edge Estimation. *Pattern Recognition*, **16**(4), 413 - 420, 1983.

[6] N. Kiryati and A. Bruckstein. Grey Levels Can Improve the Performance of Binary Image Digitizers. *CVGIP: Graphical Models and Image Processing* **51**(1), 31 - 39, 1991.

[7] J. Koplowitz and X. Lee. Edge Detection with Subpixel Accuracy. *SPIE: Automatic Objection Recognition*, **1471**, 453 - 458, 1991.

[8] C. Dong and P.C.K. Kwok. An iterative algorithm for the subpixel estimation of straight lines. *Proceedings of the Asian Conference on Computer Vision*, 554-557, November 1993.

[9] R. M. Haralick. Digital Step Edges from Zero Crossing of Second Directional Derivatives. *IEEE Trans. Pattern Analysis and Machine Intelligence*, **6**, 58-68, 1984.

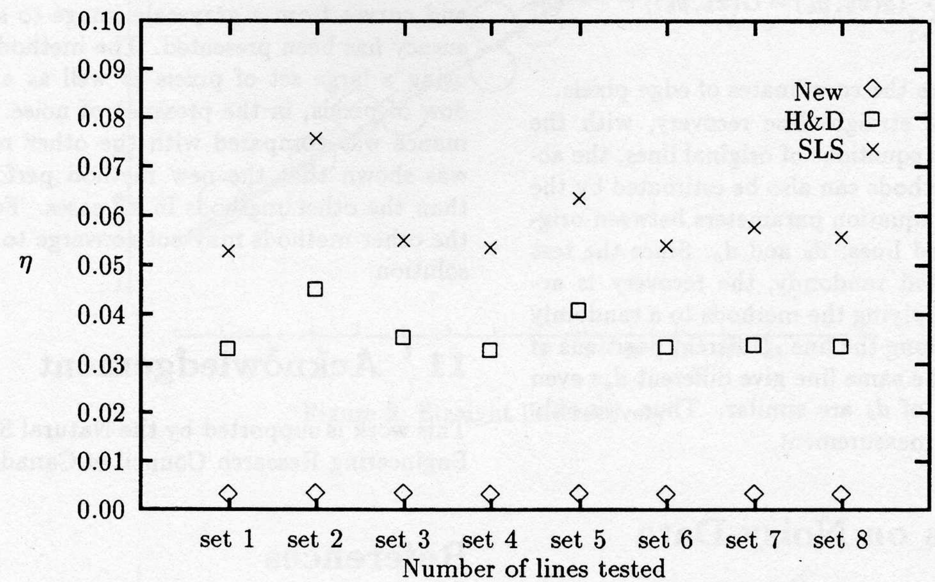


Figure 4: Noise: by measure $\eta$

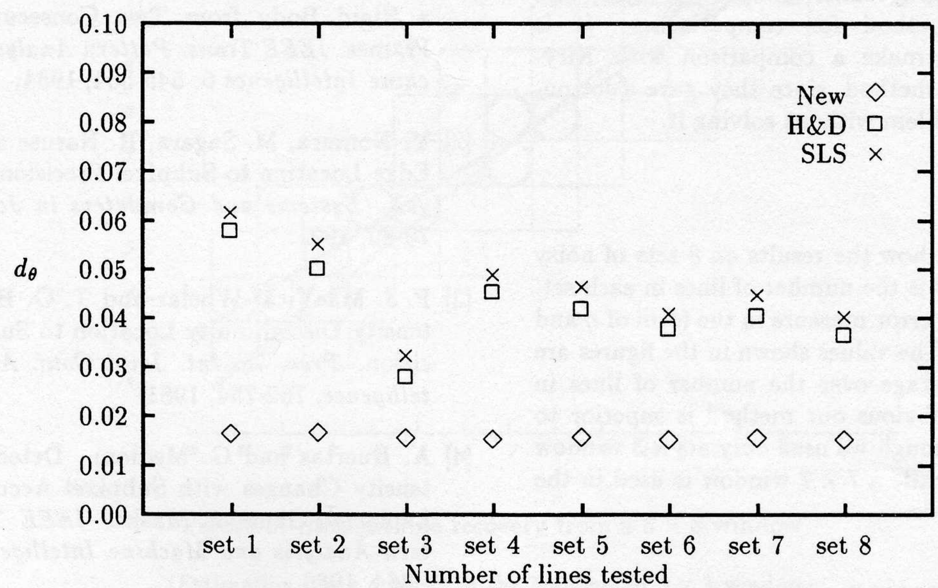


Figure 5: Noise: by measure $d_\theta$

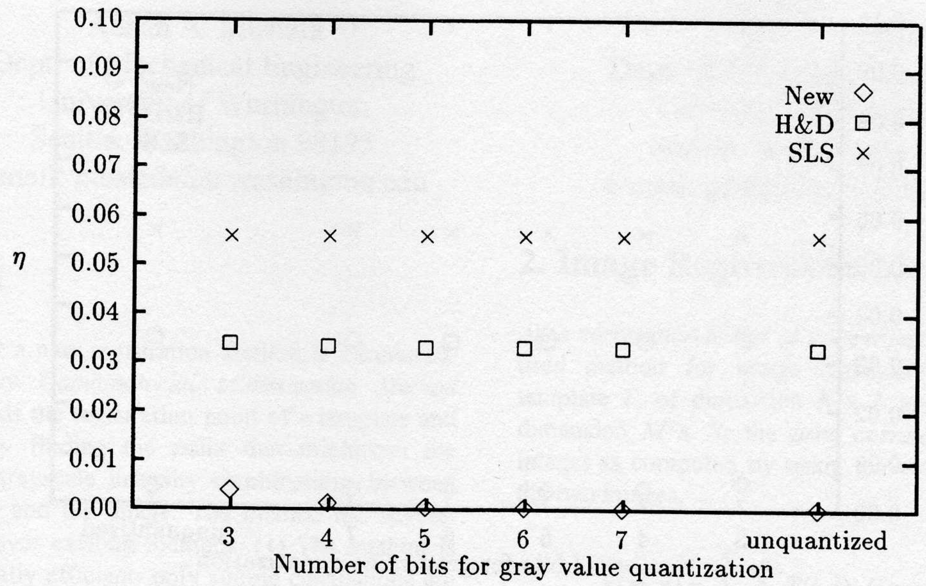


Figure 6: Quantizations: by measure $\eta$

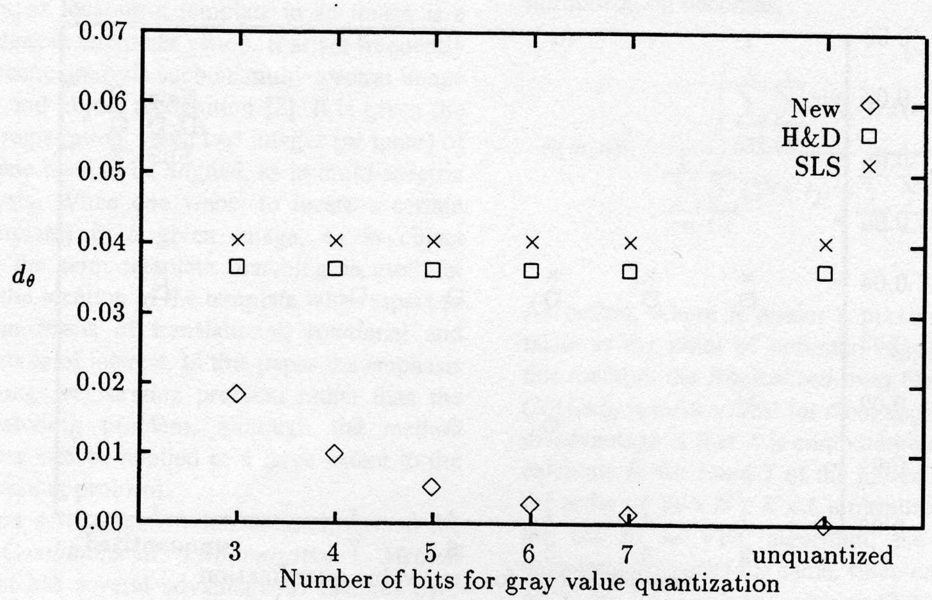


Figure 7: Quantizations: by measure $d_\theta$

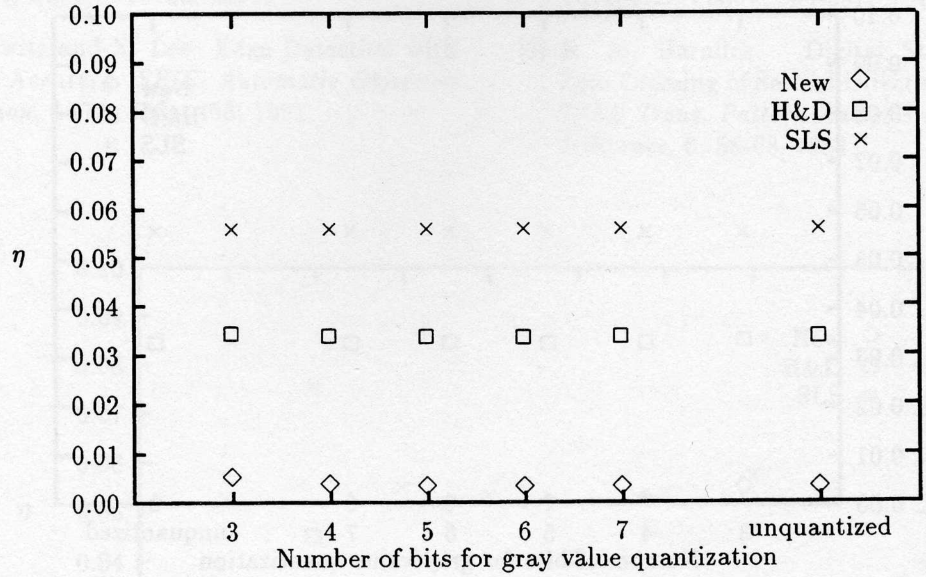


Figure 8: Noise, Quantizations: by measure $\eta$

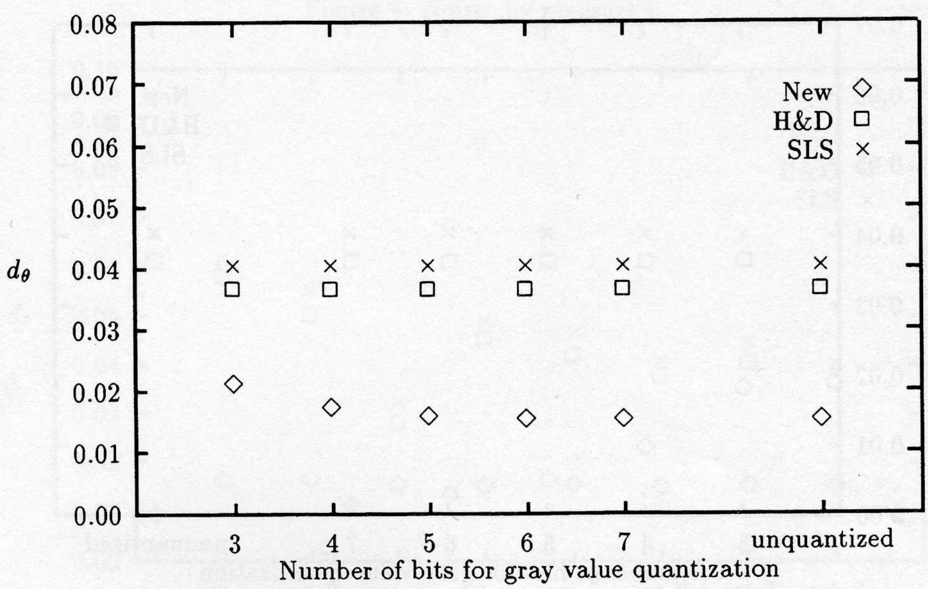


Figure 9: Noise, Quantizations: by measure $d_\theta$