

# An Analytical Design of High-Speed Pixel Transformation for Object Boundary Enhancement

KP Lam and A Furness

Automatic Identification Research Laboratory,  
Centre for Electronic Engineering, University of Keele,  
STAFFS ST5 5BG, UK

**Abstract.** Sophisticated algorithms for edge detection with built-in noise filtration and good localisation properties often require an inherently iterative and/or adaptive processing structure, which is difficult to parallelise for high speed operation. This paper describes a boundary enhancement technique which can be adopted as a cascaded precursor for edge detection. In addition to the demonstrable efficacy in enhancing edge features, the precursor offers the distinct advantage of exploiting both the image and operator parallelism. Further, unlike many other enhancement methods, its characteristics can be studied analytically. The performance of the proposed detector is examined and compared with established techniques.

## 1 Introduction

Improvements in speed and efficiency for edge detection continue to pose a challenge in 2-D real-time image processing. The gradient operators most commonly used as a measure of edge profiles fall into three general categories: (1) Operators such as Robert and Sobel which approximate image function derivatives using spatial differences with convolution masks. (2) More sophisticated algorithms with built-in noise filtering capabilities, as exemplified by Marr-Hildreth [12] and Canny [1] operators, based on the zero crossings of image function second derivatives. (3) Parametric operators, such as that of Hueckel [13] and Tan et al [16], which match and/or optimise an image function to a specific model of the edge profiles. Computationally many of these techniques apply an appropriate sequence of neighbourhood operations simultaneously on all pixels of an image, thus enabling them to exploit image parallelism when implemented on parallel computers [4]. In terms of algorithmic complexity, handling the first class of techniques is relatively simple, whilst the other two generally require significantly more iterative as well as adaptive processing [2]. In general, the approach to edge enhancement, followed by thresholding in a variety of forms, and characterising operators from the first class in particular, is largely heuristic in nature. Similarly the higher-order derivative and parametric approaches, although mathematically formulated, also suffer from this type of setback.

This paper presents a parallel, computationally efficient boundary enhancement technique which can be adopted as an effective precursor for edge detection.

Here the precursor significantly steepens the sheerness of image discontinuities, thus enabling edges with widely varying strengths and qualities to be determined using a single, non-interactive thresholding operation after the application of a gradient-based detector. In terms of computational efficiency, the technique takes advantage of the combined robustness of non-linear mathematical techniques and the simplicity of gradient operators. Adding to these the uniformity of design and regularity of operations, the characteristic blueprints for efficient parallel algorithms for low-level image processing, makes the proposed precursor technique a practical approach to high-speed edge detection.

The non-linear technique of generalised nearest-neighbour (GNN) transformation which forms the mathematical basis of the enhancement method is discussed in Section 2. An analytical treatment which leads to the development of the proposed precursor is also presented. Sections 3 and 4 examine and discuss techniques to exploit the intrinsic parallelism of the detector method devised. Section 5 describes the analytical tests and results for the proposed detector. Finally, concluding remarks are included in Section 6.

## 2 Generalised Nearest-Neighbour Transformation

The enhancement method of low pass filtering to suppress noise and eliminate spurious edges have introduced two major problems: Firstly, edges that appear in close proximity are merged and thus become indistinguishable. Secondly, the widening of an edge due to smoothing leads to a much weakened gradient amplitude, which in turn restricts the practical range of thresholds at which edges can be detected. The boundary enhancement method presented here overcomes these limitations. On the one hand, it facilitates the minimisation of the local variance of pixel gray-level values, thus counteracting the undesirable effects of smoothing. On the other hand, it maximises the separation of boundaries between relatively homogeneous image areas, thus causing the useful ranges of detection thresholds to overlap.

### 2.1 Rank filters

Earlier work on the properties and applications of these filters for noise suppression and image enhancements can be found in [7,9-10]. The extremum operators,  $\min_n$ , and  $\max_n$ , which compute rank positions at the two extremes, have the important effect of reducing the gray-scale variance of an image. Here, for a square neighbourhood window  $W_P$  of  $N(=L^2)$  pixels,  $P = \{p_{ij}^k, k = 1..N\}$ , centered upon pixel  $p_{ij}$ , we evaluate  $\min_P = R_1(P)$  and  $\max_P = R_N(P)$ , where  $R_r$  gives the  $r$ -th element of the  $N$  ordered values such that  $p_{ij}^1 \leq p_{ij}^2 \leq \dots p_{ij}^r \leq \dots p_{ij}^N$ . Without loss of generality, the effect of  $\max_P$  and  $\min_P$  on the image gray-levels can be analysed with a 1-D sequence, say  $\{x_i, i = 1..n\}$ , using a moving window  $\max_e$  of extent 3. This is depicted in Fig. 1 below. With  $x_{k+2} = x'_m = \max\{x_i, k-1 \leq i \leq k+5\}$ , it is clear that  $\max_e$  has the net effect of

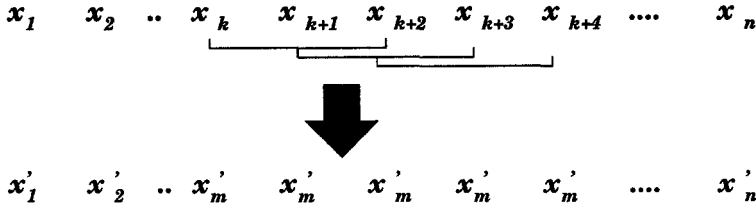


Fig. 1. The effect of  $\max_e$  operating on  $\{x\}$

reducing the statistical variance  $Var(x)$  of  $\{x\}$ , since,  $Var(x) = E(x^2) - E^2(x)$ , is a decreasing function which attains its minimum ( $=0$ ) when

$$x_1 = x_2 = x_3 = \dots = x_n \quad (1)$$

It can be shown by induction proof that the equality of (1) is the only condition whereby  $Var(x)$  is a minimum.

## 2.2 Integrating the max and min operators

The *min* and *max* operators can be integrated to take advantage of the characteristics illustrated in the preceding section. Here the principal objective is to maximise separation of the gray-scale boundaries between the relatively homogeneous image areas achieved by the min/max transformation. The Nearest-Neighbour (NN) approach offers a mathematically robust technique to achieve this. Analytically, let  $d(x_i, x_j)$  be the distance between  $x_i$  and  $x_j$  in the 1-D Euclidean space, the operator  $g(x)$ , defined as,  $g(x) = \min\{d(x, x_1), \dots, d(x, x_n)\}$ , represents the nearest neighbour of  $x$  amongst  $\{x_i\}$ . The generalisation of NN provides a key to the unification of the two operators for the characterisation of individual image points. The essence of the principle is illustrated in Fig. 2, where pixels are categorised with reference to the *a priori* measurement of focal points within a predefined neighbourhood of pixels. Individual pixel gray-levels can then be determined according to their proximity to these focal points computed by the extremum operators.

The net effects of integrating the min/max operators using the NN method are two-fold; Firstly, variations in brightness values across different edges is significantly reduced. Secondly, characterisation of the boundaries of gray-scale discontinuities representing the separation of distinctively monotonic image areas is achieved. These results are also confirmed by the analytical experiments presented in Section 5. Combining the discussion above with that of the preceding section, the GNN-based precursor transformation can now be stated as follows. If  $S$  and  $D$  represents the source and destination image respectively, then;

$$D_{i,j} = \begin{cases} M, & \text{if } (S_{i,j} - m) > (M - S_{i,j}) \\ m, & \text{otherwise.} \end{cases} \quad (2)$$

where  $M = \max[S_{x,y}; (x,y) \in W_P]$  and  $m = \min[S_{x,y}; (x,y) \in W_P]$ . Equation (2) gives a rounding down condition if  $S_{i,j}$  is equidistant from  $m$  and  $M$ .

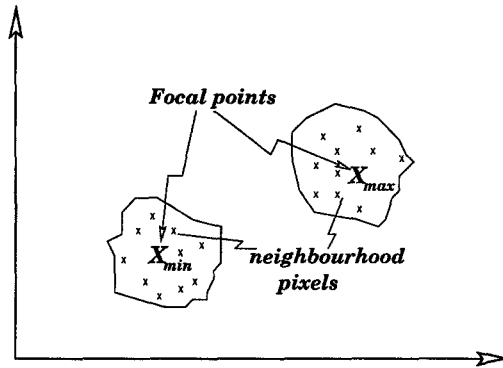


Fig. 2. Principle of the GNN method

### 3 Parallelism within the Cascaded Precursor

The boundary enhancement algorithm comprises of two stages of point-wise image operation; an edge-preserving smoothing and the Generalised Nearest-Neighbour transformation (GNN) as discussed earlier. The first processing stage facilitates a noise-smoothing filtration for the source image. As well as possessing both powerful noise reduction and good edge-preserving properties [3], the K-AVE operator selected and the succeeding GNN transformation are both derived from the same class of mathematical techniques and so they are highly compatible in terms of computational characteristics and requirements. The latter consideration is particularly desirable from a theoretical viewpoint, since the GNN is formulated to “sharpen” the spatial features of the K-AVE filtered image, thereby counteracting the effect of the much weakened edge gradients that critically limits the practicality of subsequent gradient calculations. To facilitate demonstrating the technique as an effective precursor to edge detection, the well-known Sobel operator [14] is chosen in this work primarily because of its relatively high accuracy and robustness [5]. More importantly, the convolution operation required is well suited to high-speed (parallel) implementation [11].

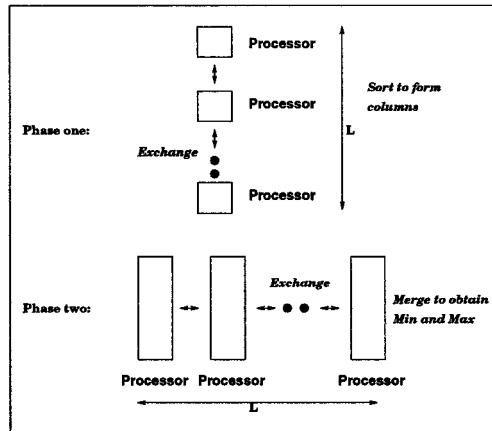
The detector described above is inherently parallel; the directly cascable operation of the GNN precursor results in a computationally efficient processing structure which can exploit two levels of parallelism, namely the *image* and *operator* parallelism [4].

1. *Image Parallelism* is directly derivable from the K-AVE, GNN and Sobel operators and offers a fine-grained parallelism which is characteristic of many low-level image processing algorithms. Specifically individual processors can apply the appropriate sequence of computations specified by each operator to different (sets of) pixels concurrently.
2. *Operator Parallelism* is achievable by pipeline processing successive stages of the K-AVE, GNN and Sobel computation. This offers a coarse-grained parallelism in that individual sets of processors may be cascaded in the form of a pipeline whereby each set operates on its inputs independently.

As with many iconic image transformations, it is evident that the relative multiplicity of data arising from each of the spatial operators in 1. above determines the maximum parallelism achievable, *ie.* the totality of data in a practical image offers maximum units of parallelism [15]. In this respect, the most CPU-intensive is the calculation of different ranks required by the K-AVE filter and the GNN transformation. Adopting the notations in Section 2, the former entails the selection of  $R_j$ ,  $j = 1..K$ , from the defined window neighbourhood of  $W_P$ , whilst the latter requires the determination of  $R_1$  and  $R_N$ .

## 4 Sort-And-Merge Technique

Previous work has demonstrated that the data-parallel model of computation can be used effectively in realising a general class of rank filters on massively parallel platforms [8]. For the GNN transformation, it can be shown that the two-phase divide-and-conquer algorithm described in [8] can easily be adapted to facilitate an efficient data-parallel implementation. Specifically the computation of the extrema in a  $L \times L$  window is performed by first sorting  $L$  subsequences along one dimension, say the column, of the moving window, then merging them in the orthogonal direction. A simple sorting method is used as  $L$  is small. Similarly, values of  $R_1$  and  $R_N$  are determined by comparing the smallest and largest elements of the appropriate groups of  $L$  sorted columns respectively. A representation of the method is depicted in Fig. 3. Since each of the  $L$  sorted



**Fig. 3.** The two-stage Sort-and-Merge selection method

columns can again be used for computing a further  $L - 1$  calculations of extrema in the neighbourhood, the method thus facilitates  $O(k_1 L^2)$  comparisons to be executed on  $L$  processors with an  $O(k_2 L)$  complexity, where  $k_1$  and  $k_2$  are small constants.

In performing the K-AVE filtration, it should be pointed out that the sorted subsequences obtained in the first phase have to be re-computed by their  $L - 1$  immediate neighbours, since the reference centre pixel  $p_{i,j}$  from which the nearest distances are derived changes as the window  $W_P$  moves across the image. Further, it should also be noted that, for a relatively large  $K$ ,  $\frac{N}{2} \ll K < N$ , it would be more advantageous to compute the  $N - K$  largest elements in  $P$ , since  $\sum_{i=1}^K R_i = \sum_{i=1}^N R_i - \sum_{i=K+1}^N R_i$ . Consequently, the K-AVE operator can be computed as a spatial convolution with unity mask weights (*cf.* Average-filter), less the sum total of the individual ranks  $R_{K+1} + \dots + R_N$ . These rank values are determined using the Sort-and-Merge method depicted in Fig. 3.

## 5 Performance and Discussion

The approach of Pratt [14] is used to quantify the efficacy of the proposed GNN precursor. A  $64 \times 64$  gray-scale (256 levels) image is constructed, which contains a steep ramp edge of height 30, separating two uniform regions of gray values 100 and 130 respectively in fixed steps of 15. The noise used is additive Gaussian,  $N(0, \sigma)$ , with signal-to-noise ratio  $(SNR) = (h/\sigma)^2$ , where  $h (= 30)$  represents the edge height. The quality of edge detection,  $\psi$ , is defined as:

$$\psi = \left( \frac{100}{I_N} \right) \sum_{i=1}^{I_A} \frac{1}{1 + \alpha d_i} \quad (3)$$

where  $I_N = \max(I_I, I_A)$ ,  $I_I$  = number of ideal edge points,  $I_A$  = number of detected edge points,  $\alpha$  = scale factor to penalise offset edge points, taken as  $1/9$  here and in [14], and  $d$  = distance of the detected edge point from the true edge. Of particular interest here is the variation of  $\psi$  with the threshold  $T$  of edge detection following the Sobel operator. Using a  $3 \times 3$  window mask, *ie.*  $L = 3$ , with  $K = 6$  for computational efficiency and good results [6], plots of  $\psi$  *vs.*  $T$  are presented in Fig. 4 for  $SNR = 7.5$  and  $15$  respectively. These plots clearly show that whilst the K-AVE filtration smooths edges, the GNN precursor significantly widens the range of detectable thresholds as predicted. In particular, a plateau generally develops, showing that edges can be detected with over 90% quality from about a wide range of thresholds  $30 \leq T \leq 120$ .

For comparison purposes, the cascaded precursor approach is also considered with the Canny operator, using several real-world digitised images. The scene was captured using a portable camera unit whose frame buffer size is  $754 \times 502$ , with 0-255 gray-levels. Thus picture degradation from input circuitry provides an approximately Gaussian additive noise field. Computationally the GNN-based detector method is inexpensive; using the techniques described in Section 4 where applicable, the total processing time required to produce a good edge image ranges 1.26–1.99 seconds on a P166 CPU, the maximum corresponds to a double application of K-AVE and GNN operators. The Canny algorithm typically requires 34–52 seconds, depending upon the image and  $\sigma$  used. In all cases, the GNN process has kept close edges narrow, with weak edges steepened

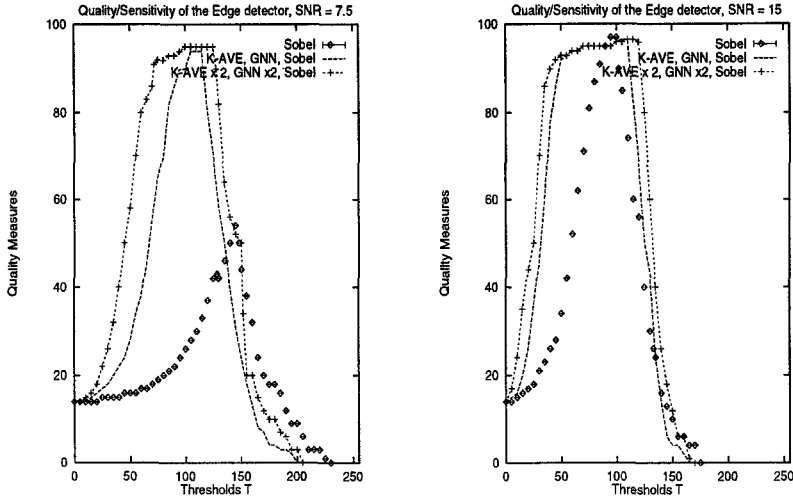


Fig. 4. SNR=7.5 and SNR=15 respectively

and the noise suppressed sufficiently so that they become distinguishable within the overlapped detection ranges for strong edges.

## 6 Summary and Conclusion

The detection of edges is simplified if the edge strengths are suitably enhanced. Many edge operators approximate local derivatives with spatial differencing techniques, which emphasise high frequency features including false edges due to noise points. If the noise is smoothed by low-pass filters, the strength of edges detected can be significantly weakened. Also such a non-selective smoothing causes close edges to merge, limiting the threshold range at which true edges may be detected. The proposed GNN precursor resolves these problems by adopting a non-linear mathematical technique to enhance spatial discontinuities of an image. When combined with the edge-preserving K-AVE filter, it has been shown to exhibit the desirable properties of sharpening edge features, thus enabling them to be determined irrespective of the widely varying strengths and qualities. These characteristics have been shown analytically, and the preliminary experimental results obtained confirm the analysis.

Computationally the characteristics of the proposed GNN match well with that of the K-AVE, both being derived from the same class of operators. By adopting the data-parallel technique developed in an earlier work, it has been shown that the uniformity of the K-AVE and GNN computations has the important advantage that they could both be implemented efficiently on highly parallel platforms. Further, given the the relative robustness and ease of parallelisation of the Sobel operator, the precursor-based edge detection method has considerable potential for real-time applications in in practical situations. Moreover, the

investigation of precursor method also raises the interesting possibilities that the general class of rank operators could be further developed to facilitate selective removal and retention of image edge features for optimal detection (*cf.* WMF in [8]). This warrants further research.

## References

1. Canny JF.: A computational approach to edge detection. *IEEE Trans. PAMI.* **8** (1986) 1-14
2. Chen SS., Chou JS., Lin WC., Pelizzari CA.: Edge and surface searching in medical images. *Medical Imaging II* - SPIE (1988) 594-599
3. Chin RT., Yeh CL.: Quantitative evaluation of some edge-preserving noise-smoothing techniques, *Computer Vision, Graphics and Image Processing Vol 23* No. 1 (1983) 64-91
4. Danielsson PE., Levialdi S.: Computer architectures for pictorial information systems, *IEEE Computer*, **14**(11) (1981) 53-67
5. Davies ER., Circularity - A new principle underlying the design of accurate edge orientation operators. *Image and Vision Computing* **2** (1984) 134-142
6. Davis LS., Rosenfeld A.: Noise cleaning by iterative local averaging. *IEEE Trans. Syst. Man Cybern. SMC-7* (1978) 256-261
7. Hodgeson RM.: Properties, implementations and applications of rank filters. *Image and Vision Computing* **3** (1985) 3-14
8. Lam KP., Horne E.: A data-parallel approach to the implementation of weighted medians technique on parallel/super-computers. *Procs. of Fifth International Symposium on Parallel and Distributed Processing* (1993) 734-738
9. Lee WH.: Signal processing techniques for CCD image sensors. *Applied Optics* **23** December (1984) 4280-4289
10. Levy-Mandel et al: Knowledge-based land-marking of cephalograms. *Computers and Biomedical Research* **19** (1986) 282-309
11. Manning LJ., Dew PM., Wang H.: Programming models for VLSI Array Processors with application to low-level vision processing. *Procs. of Parallel Architecture and Computer Vision Workshop, UK* (1987) UK1.5
12. Marr D. and Hildreth E.: Theory of edge detection. *Procs of the Royal Society London B***207** (1980) 187-217
13. Nevatia R.: Evaluation of simplified Hueckel edge-line detector. *Computer Graphics and Image Processing.* **6**(6) (1977) 582-588
14. Pratt WK., Abdou IE.: Quantitative design and evaluation of enhancement/thresholding edge detectors. *Procs IEEE Vol 67-5* May (1979)
15. Quinn MJ., Hatcher PJ.: Data-parallel programming on multi-computers. *IEEE software* **7** (1990) 69-76
16. Tan HK., Gelfand SB., Delp EJ.: A cost minimization approach to edge detection using simulated annealing, *IEEE Trans of PAMI* **14**(1) (1992) 42-56