A PATH METRIC FOR SEQUENTIAL SEARCH AND ITS APPLICATION IN EDGE LINKING

Aly A. Farag
Engineering Mathematics \\ Computer Science Dept.
speed scientific school
University of Louisville
Louisville, KY 40292.

Edward J. Delp
Computer Vision & Image Processing Lab.
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907.

Abstract: In this paper we develop a new path metric for sequential search based on the linear model. The metric forms the heart of an edge linking algorithm that combines edge elements enhanced by an optimal filter. From a starting node, transitions are made to the goal nodes by a maximum likelihood metric. This metric requires only local calculations on the search space and its use in edge linking provides more accurate results than other linking techniques.

1. INTRODUCTION

A typical search problem is formed from three major components: state description, operators (metrics) that guide transitions between states, and a search strategy. The focus of this research is to develop a metric that can be estimated from local measurements on the search space. This metric forms the heart of a sequential search algorithm that is used to perform complete object boundary allocation on discrete two-dimensional images. The object boundary allocation problem is commonly referred to as edge detection which is a two-stage system: edge enhancement followed by edge linking (Fig. 1). Modern techniques for edge enhancement are based on optimal filtering; for example, the Laplacian of the Gaussian (the \(V_G\) operator) [1] and the gradient of the Gaussian (the \(V_G\) operator) [2]. These filters can optimally enhance step edges; however, an optimal procedure for the selection of the enhancement filter's spatial support is needed. Also, a procedure that can optimally enhance other edges in the scene, e.g., roof, corners, and stepped edges is needed. Articles [3] and [4] are of interest.

In the second stage, edge linking, the enhanced edge elements are linked together to form a closed contour that describes the object's boundary. The methods used to enhance the edges also provide a clue for linking. For example, in the \(V_G\) operator, the locations of zero-crossings provide the decoding method used to link the edges. Similarly, when enhancing with the VG operator, the points of maximum gradient (normal to the edge direction) provide the decoding used to link the edges.

Since real world scenes contain various types of edges, the application of a particular enhancement filter will not enhance all edges; thus, errors (e.g., edge displacement) usually result. Hence, the linking stage is not at all trivial. In fact, a good linking procedure should also correct some of the errors introduced by the enhancement step. In this paper we are concerned with the linking of edges enhanced by a gradient operator (e.g., the VG operator), Canny [2], used the minimum suppression and thresholding (NONMAX) for deciding upon edge pixels. In this approach only pixels which have a maximum directional derivative (normal to an edge direction) are considered. The problem with this approach is that the resulting edge map can be full of broken edges (streaks).

The use of quantitative sequential search techniques to link edges has been shown to provide better object boundaries and edge localization than the NONMAX approach [8]. Articles [5]-[7] are also of interest.

In this paper, a new linking algorithm that uses tree search to link edges enhanced by a gradient operator is introduced. Section 2 states the edge linking problem as a tree search. Section 3, the main contribution of the paper, contains a new search metric. Section 4 describes a linking algorithm (LINK) that used the new search metric and the \(A^*\) algorithm [9]. Section 5 contains experimental results and Section 6 presents the conclusions.

II. EDGE LINKING AS A TREE SEARCH

Let the spatial support of an \(M \times N\) discrete image be \(S\), that is,

\[
S = \{(x,y) : 0 \leq x \leq M-1, 0 \leq y \leq N-1\}.
\]  

Any internal pixel location (a site, or node) \(s = (x,y)\) has a unique set of eight nearest neighbors. Given a node \(s\) on an edge path, the path can be extended in eight possible directions, each extension leading to one of the eight neighbors of \(s\). The next node on the path can then be extended similarly through any of its neighbors, and so on. What evolves is a tree structure with each node possessing eight outgoing branches. The depth into the tree indicates the position along the path. As Ashkar and Modesto [6] noted, one of the eight nearest neighbors of any node along the path (except the root node) leads back

563

Consider an edge path \( p \) and a pixel (node) \( r = (x', y') \) on \( p \). The eight nearest neighbors are:

\[
\{(x'-1, y'), (x'+1, y'), (x', y'-1), (x', y'+1), (x'-1, y'-1), (x'-1, y'+1), (x'+1, y'+1)\}.
\]

One of these eight nodes must be on the path \( p \) and is known before node \( r \). Let this be \( r' = (x'y'1) \), i.e., the straight line between nodes \( r' \) and \( r \) is horizontal. Fig. 2 illustrates this situation. Knowing that nodes \( r' \) and \( r \) are on the edge path \( p \), how can we extend the path an extra node? That is, how should we choose among the seven possible nodes? This paper develops a maximum likelihood criterion for this selection. This will be addressed shortly, but first let us examine the search space.

An edge path \( p \) of length \( Q \) will be described by the cartesian co-ordinates of the sites forming it. The size of the search space for a path of \( Q \) nodes is \( 2^Q \). Therefore, an exhaustive search is not feasible for any practical sized image. The computational burden can be drastically reduced if the number of actual extensions of an edge path from a given node (on the path) is reduced. Further reduction in the size of the search space is possible if what is considered to be an edge path is selected according to a specific rule. This rule, or optimality criterion, is denoted by the path metric. The path definition in [8] will be adopted in this paper. It is formally defined as follows:

**Definition:** A path is a connected set of nodes with the following property: For any subset of three nodes on the path, the direction defined by the first two nodes and by the second two nodes differ by \( \pi/4 \) or less.

Fig. 3 provides a few examples of possible edge paths, and the tree is constructed as shown in Fig. 4. On this tree structure, search will be performed using the A*-algorithm [9]. The components of sequential search are: root node selection, path metric, and goal node definition which specifies the stopping criterion. By far the most important component is the path metric. In the following section, we introduce our path metric.

**III. A METRIC FOR NODE EXTENSION**

Let \( p^o = \{r_0, r_1, r_2, \ldots, r_Q, \theta \} \) be a sequence of nodes along the \( i^o \) path up to level \( Q \) in the tree, and let \( \beta^o \) be a cost measure for the transition from node \( r_j \) to node \( r_i \) on this path. The cost of the transitions along the sequence of nodes \( p^o \) will be the cumulative cost for the \( Q \)-transitions along the \( i^o \) path from a start node \( r_0 \). The path metric is a function of this cost. We propose a metric of the following form:

\[
\gamma_0(p^o) = \sum_{i=0}^{Q} \beta_i(p^o)
\]

where \( \beta_i(p^o) \) is a measure for the selection of one of the \( Q \) possible transitions along the \( i^o \) branch of the path \( p^o \). This metric will be derived from the linear model.

**A. The Linear Model**

Let the observed edge enhanced image (the gradient magnitude) be a sample function of a random field \( G = \{G_s, s \in S\} \). Consider another random field \( E \) in which the random variables \( \{E_s, s \in S\} \) have zero-mean Gaussian distribution with common variance \( \sigma^2 \). The linear model equation is [10]:

\[
g = A\theta + e,
\]

where \( g \) and \( e \) are \( L \times 1 \) vectors representing an observation from the random fields \( G \) and \( E \), respectively. The matrix \( A \) is \( L \times k \) with \( L > k \), and \( \theta \) is a \( k \times 1 \) vector of unknown parameters.

We will consider the case where the coefficients of the matrix \( A \) are deterministic. Let us define the two hypotheses:

\[
H_0: \text{ } B\theta = e, \quad H_1: \text{ } B\theta \neq e,
\]

where \( B \) is a given \( q \times k \) matrix of rank \( q \leq k \), \( e \) is a given \( q \times 1 \) vector, and \( B\theta = e \) is a consistent set of linear equations.

Given the values of the parameters \( \theta \) and \( \sigma^2 \), the random vector \( g \) is Gaussian with mean \( A\theta \) and diagonal covariance matrix, \( \sigma^2 I \), where \( I \) is an \( L \times L \) identity matrix, i.e.,

\[
p(g|\theta, \sigma^2) = (1/(2\pi \sigma^2))^{L/2} \exp \left( -(g - A\theta)'(g - A\theta)/(2\sigma^2) \right).
\]

The natural logarithm of (6) is the following quadratic function

\[
\psi(\sigma^2, \theta) = -\ln p(g|\theta, \sigma^2)
\]

\[
\psi(\sigma^2, \theta) = -\frac{1}{2} \ln 2\pi - \frac{1}{2} \ln \sigma^2 - \frac{1}{2} (g - A\theta)'(g - A\theta)/2\sigma^2.
\]

The MLE of \( \sigma^2 \) and \( \theta \) can be obtained from the differentiation of (6) (or its natural logarithm (8)) and equating the derivatives to zero. It can be easily shown that these estimates are:

\[
\hat{\sigma}^2 = \frac{1}{N} (g - \hat{A} \hat{\theta})' (g - \hat{A} \hat{\theta})\text{ and}
\]

\[
\hat{\theta} = \frac{1}{N} (A\hat{\theta})'
\]

Let's define the following likelihood ratio:

\[
LRT = p(g|\hat{\theta}, \hat{\sigma}^2) / p(g|\theta, \sigma^2).
\]
From (9) and (10), it’s easy to show that
\[ LRT = \left( \sigma_0 \hat{\sigma}_0 \right)^2, \]  
(12)

independent of the parameter \( \Theta \), where \( \sigma_0^2 \) is the variance of the Gaussian distribution under hypothesis \( H_0 \); \( i = 1, 2 \).

Let’s also define the following statistic:
\[ \Lambda = \left( LRT^{(i)} \right)^{1/2} \left( \frac{L - k}{q} \right). \]  
(13)

It is readily seen that:
\[ \Lambda \sim \left( \sigma_0^2 \sigma_i^2 \right)^{1/2} \left( \frac{L - k}{q} \right). \]  
(14)

The statistic \( \Lambda \) has an F-distribution with \( (q, (q - k)) \) degrees of freedom [11][12]. It will be used to evaluate the parameter \( \beta_i \) in (3).

B. Definition of Edge Hypothesis

On a \( 3 \times 3 \) neighborhood, edge directions are usually quantized into eight directions (multiples of \( 45^\circ \)). Hence, we can define the set \( \{ H, V, D_1, D_2 \} \) of four edge-models (edge hypotheses) on this neighborhood as shown in Fig. 5. To use the statistic \( \Lambda \) in (14), we need to specify \( L, k, \) and the parameters \( \sigma_0^2 \) and \( \sigma_i^2 \). The matrix \( A \) is specified by fitting a linear model to each edge model (edge hypotheses). Fig. 6 shows the components of the linear model fit to edge model \( D_1 \). The linear model equation for this edge model is written as
\[ g = A \cdot h + e, \]  
(15)

where
\[ g = \begin{bmatrix} g_1 & g_2 & g_3 & \cdots & g_8 \end{bmatrix}^T. \]
(16a)
\[ e = \begin{bmatrix} e_1 & e_2 & e_3 & \cdots & e_8 \end{bmatrix}^T. \]
(16b)
\[ A = \begin{bmatrix} 0.5 & 1 & 0 & 0.5 & 1 & 0 & 0.5 & 1 \\ 0.5 & 0 & 1 & 0.5 & 0 & 1 & 0.5 & 0 \end{bmatrix}. \]
(16c)
\[ h = \begin{bmatrix} h_1 \\ h_2 \end{bmatrix}. \]
(16d)

and the script \( \cdot^T \) in (16) denotes matrix transposition.

C. Parameter Estimation

We now address the problem of parameter estimation. As we indicated before, for proper definition of the linear model, it is necessary that the matrix \( B \) and the vector \( e \) in (5) be such that the \( q \times k \) matrix \( B \) must have a full rank (i.e., \( \text{rank} = k \)), and the set of equations \( Bh = e \) must have a consistent set of solutions. A simple choice for the matrix \( B \) and the vector \( e \) which satisfies these requirements is the following:
\[ B = \begin{bmatrix} 1 & -1 \end{bmatrix} \quad \text{and} \quad e = 0. \]  
(17)

This choice for \( B \) and \( e \) is also similar to that in [12] and [13].

With the above choice for \( B \) and \( e \), the hypothesis \( H_0 \) and \( H_2 \) are now written as follows:
\[ H_0: \begin{bmatrix} 1 & -1 \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} = 0 \quad \text{and} \quad H_2: \begin{bmatrix} 1 & -1 \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} \neq 0. \]  
(18)

From the above specifications of \( A, B, \) and \( e \), we have \( L = 9, \ k = 2, \) and \( q = 1 \). Hence, the statistic \( \Lambda \) in (14) is
\[ \Lambda = \frac{1}{7} \left( \frac{\sigma_0^2 - \sigma_i^2}{\sigma_0^2} \right). \]  
(19)

The maximum likelihood estimation (MLE) for the parameters \( \sigma_0 \) and \( \sigma_i \) can be easily calculated as follows: For the above choice of \( B \) and \( e \), \( H_0 \) is the no-edge hypothesis, that is, \( h_1 = h_2 = h \). Hence, (4) becomes
\[ g = A \cdot e + \epsilon, \quad \text{where} \quad A' = A \begin{bmatrix} 1 \\ 1 \end{bmatrix} h. \]  
(20)

Equation (20) is the classic deterministic signal and additive noise problem. The MLE is
\[ \hat{\sigma}_i^2 = \frac{1}{2} \sum_{k} \left( g - \hat{h} \right)^2 \quad \text{where} \quad \hat{h} = \frac{1}{2} \sum_{k} h^*. \]  
(21)

For hypothesis \( H_2 \), the edge hypothesis, the MLE for \( \sigma_i^2 \) is obtained from the minimization of the natural logarithm in (8). For the \( D_2 \) edge model, this minimization can be written in the following quadratic form:
\[ J = \sum_{t=1}^{N} \left( g - \hat{h}_1 \right)^2 + \sum_{t=1}^{N} \left( g - \hat{h}_2 \right)^2 + \sum_{t=1}^{N} \left( g - \hat{h}_3 \right)^2. \]  
(22)

Differentiating (22) with respect to \( h_1 \) and \( h_2 \) and equating the derivatives to zero, it is simple to show that the MLE for \( h_1 \) and \( h_2 \) are as follows:
\[ \hat{h}_1 = w_1 g, \quad \hat{h}_2 = w_2 g, \]  
(23)

where
\[ w_1 = \frac{1}{18} \begin{bmatrix} 2 & 5 & 5 & -1 & 2 & 5 & -1 & -1 & 2 \end{bmatrix} \]  
(24)

and
\[ w_2 = \frac{1}{18} \begin{bmatrix} 2 & -1 & 5 & 2 & -1 & 5 & 5 & 2 \end{bmatrix}. \]  
(25)

Hence, the estimate \( \hat{\sigma}_2 \) has the following form:
\[ \hat{\sigma}_2^2 = \sum_{t=1}^{N} \left( g - \hat{h}_1 \right)^2 + \sum_{t=1}^{N} \left( g - \hat{h}_2 \right)^2 + \sum_{t=1}^{N} \left( g - \hat{h}_3 \right)^2. \]  
(26)

The MLE for the parameters \( \sigma_0 \) and \( \sigma_i \) can be easily obtained [12][13]. The table provides a summary of the equations needed to calculate the statistic (19) for the four edge models.
TABLE: THE TEST STATISTIC $\Lambda$ FOR 4-EDGE MODELS

$$\Lambda = \sum_{i=1}^{4} \left( \sum_{j \in \mathbb{Z}} (g_i - \bar{g})^2 \right)$$
$$\bar{g} = \frac{1}{4} \sum_{i \in \mathbb{Z}} g_i$$
$$\bar{g}_i = \frac{1}{4} \sum_{j \in \mathbb{Z}} g_{i,j}$$

Edge model $H$

$$\sigma_1^2 = \sum_{i \in \mathbb{Z}} (g_i - \bar{g})^2$$
$$w_1 = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Edge model $V$

$$\sigma_2^2 = \sum_{i \in \mathbb{Z}} (g_i - \bar{g})^2$$
$$w_2 = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

D. The Path Metric

The quantity $\beta_i$ in (3) is evaluated from the linear model as follows: First, knowing the predecessor of the current node, select the next node by choosing the edge model (among the three possible models) that has a maximum value of $\Lambda$ in (19). Then, set $\Lambda$ to be equal to the value of $\Lambda$. For example, in the situation shown in Fig. 2, we need to evaluate $\Lambda$ for edge models $\{H, D_0, D_1\}$ and choose the model with the maximum. Ties are selected arbitrarily, but always in favor of goal nodes. The cost of the $i$th path having $Q$-nodes is the additive cost of all the nodes forming it. The cost associated with node $j$ (the $j$th branch) is obtained from (19). It is obvious that only local calculations are needed to obtain this cost and, therefore, the metric in (3) is very easy to calculate.

Kay and Lemay [12] have also used the linear model in their study. However, our approach differs from theirs in four main points: First, they consider only two edge models, the horizontal and the vertical $\{H, V\}$. Second, and most importantly, they detect edges solely based upon the value of the ratio in (19); that is, a threshold is set, and an edge pixel is declared if the ratio exceeds this threshold. As a result, their approach suffers from the well-known problems associated with edge detection by thresholding. Third, their approach gives no consideration to edge orientation; thus, the issue of edge localization is totally ignored. This, in addition to the previous point, explains why the authors had poor edge detection results. Finally, our approach uses the linear model as a part of the linking algorithm on enhanced edges and not on the original image as in (12).

IV. THE LINKING ALGORITHM (LINK)

The sequential linking algorithm that uses the metric in (3) and the $\Lambda^*$ algorithm is outlined in the following steps.

Step 1. Perform edge enhancement using a gradient operator of suitable spatial support.

Step 2. Choose a root node and find the corresponding initial direction from the gradient angle.

Step 3. Transitions on a path $i$ are carried out by the $\Lambda^*$ algorithm depending upon the value of $\gamma_i$ in (3).

Step 4. Stop the search when all goal nodes have been examined.

In this paper, we used the VG operator for edge enhancement. Therefore, the edge information (gradient vector) is represented in terms of a magnitude map and an angle map. We selected the root node to be the point of highest gradient magnitude. The goal nodes were selected such that: (i) The nodes considered by the search algorithm have corresponding gray level pixel values (gradient magnitude) within 30% of the maximum (that defines the root node); (ii) paths do not intersect the image boundary; and (iii) isolated short paths (less than 18 pixels long) are ignored.

V. RESULTS

The LINK algorithm has been applied to various test images and natural scenes. We will provide a few examples here to show the relative performance of LINK to the SEL algorithm [13], as well as in the NONMAX approach of Canny [2].

Fig. 7 shows the results on the steps image for different signal-to-noise ratios, $\text{SNR} = (\Delta, \sigma)$, where $\Delta$ is the edge contrast and $\sigma$ is the variance of the additive Gaussian noise. The top row shows the original image for SNR values of 100 (left-most), 10, 1, and 0.5 (right-most). The second row shows the results of NONMAX for various SNR, the third row shows the results of the SEL algorithm, and the fourth row shows the results of LINK. The size of the VG was 9 x 9. At $\text{SNR} = 10$, SEL produced an incomplete contour in parts of the second circle from the outside. On the other hand, few false boundaries were
detected by LINK because the threshold set on A was slightly low. As the SNR is reduced, the performance of the algorithms deteriorates.

Fig. 8 shows the results on the discs image [15] for different signal-to-noise ratios, SNR = (A/N)2, where A is the edge contrast and \sigma is the variance of the additive Gaussian noise. The top row shows the original image for SNR values of 100 (left-most), 10, 1, and 0.5 (right-most). The second row shows the results of NONMAX for various SNR, the third row shows the results of the SEL algorithm, and the fourth row shows the results of LINK. The size of the VO was also 5 × 5. At SNR = 10, SEL produced an incomplete contour in parts of the second circle from the outside. On the other hand, few false boundaries were detected by LINK because the threshold set on A was slightly low. As the SNR is reduced, the performance of the algorithms deteriorates.

Fig. 9 shows the results on the girl image. In this figure, the original picture is shown in the upper left, the NONMAX results are shown in the upper right, the SEL results are shown in the lower left, and the results of LINK are shown in the lower right. Edges were enhanced by a VO operator of size 7 × 7 and an upper bound threshold of 25 for the three linking algorithms. The picture here was less crowded, and the contrast between the object and the background was great, which resulted in a very good enhancement map. This was reflected on the response of the three linking algorithms. The performance of SEL and LINK were nearly the same on this particular image.

VI. CONCLUSION

In this paper we examined the application of sequential search techniques for edge linking. A new metric based on the linear model was developed. A search algorithm (LINK) that used this metric and the A* algorithm for edge linking was constructed. The metric is easy to compute and provides more accurate results than the NONMAX technique and the SEL algorithm at comparable execution time. The LINK algorithm has been used on numerous other images, and the results clearly indicate that it is very adequate for edge linking. Currently, we are considering other path definitions and the application of the linear model on wider block sizes than the 3 × 3 blocks in this paper.

ACKNOWLEDGMENT

Dr. Paul Eichel of Sandia National Laboratory provided the software of the SEL algorithm.

REFERENCES


![Figure 1. Edge detection system.](image-url)
Fig. 2 Nearest neighbors

Fig. 3 Path definition

Fig. 4 Tree structure

Fig. 5 Edge models

Fig. 6 A linear model fit to edge $D_2$

Fig. 7 Results of the steps image

Fig. 8 Results of the discs image

Fig. 9 Results of the girl image