Computer Science
Technical Report

Colorado State University

Optical Linear Feature Detection Based on Model Pose *

Mark R. Stevens and J. Ross Beveridge

December 16, 1995

Technical Report CS-96-110

Computer Science Department
Colorado State University
Fort Collins, CO 80523-1873

Phone: (970) 491-5792   Fax: (970) 491-2466
WWW: http://www.cs.colostate.edu

*This work was sponsored by the Advanced Research Projects Agency (ARPA) under grant DAAH04-93-G-422, monitored by the U. S. Army Research Office.

Optical Linear Feature Detection Based on Model Pose *†

Mark R. Stevens
Colorado State University
stevensm@cs.colostate.edu

J. Ross Beveridge
Colorado State University
ross@cs.colostate.edu

Abstract

Low-level edge detection in optical imagery can be problematic in the ATR domain where highly complex scenes are the norm. Feature detection algorithms typically take a global approach, resulting in the discovery of many fragmented lines which are not directly related to stored model information. For this domain, we have taken a top-down approach which searches an optical image for the locally optimal features based on the current hypothesized object pose. The resulting linear features can then be matched against a CAD model.

1 Introduction

Edge detection in the Automatic Target Recognition (ATR) domain should be driven by the expectation of which model features are assumed to be visible in a given image. Using a hypothesized model pose to predict visible features from a CAD model [Mar96, Ste95], a local optimization procedure is used to find the corresponding and consistent data features in the image.

The process differs from the traditional low-level bottom-up edge detection process [MH80, Hil83] which can be highly error-prone [Cla89]. The main problem with bottom-up detection is the inability to deal with large amounts of scene complexity and clutter. Color imagery in the ATR domain often contains many different structural events taking place simultaneously (camouflage on military vehicles set against natural terrain is an excellent example). Current edge detection algorithms [BHR86, LB83, FL88] do not deal well with these type of scenes and will produce many small fragmented line segments which can easily distract a model-based matching system.

Furthermore, most ATR algorithms require that the edges supplied have some physical significance relative to the vehicle in the scene [Cla89]. The model matching process will be more robust if there is a one-to-one correspondence between extracted data lines and model features. Our experience suggests bottom-up feature extraction can not meet the requirements of this domain.

Consequently, we take a top-down approach in which the current set of model features drives the search for line segments in color images. Using a method to predict visible model lines for a hypothesized pose [Mar96], the model features can be projected into a given image using known sensor characteristics [BHP94]. Local search then maximizes the segment orientation and position based on the current gradient response. A similar approach has been applied using gradient descent to perturb the line segment [SWF95].

2 Local Search

The model-driven approach is initialized by projecting the predicted 3D model edges [Mar96] into the color image. An error function uses a gradient mask oriented to the direction of the model edge to determine the underlying changes in pixel intensity. The error function is then used to guide a local search algorithm in the selection of a better edge position.

2.1 Oriented Gradient Mask

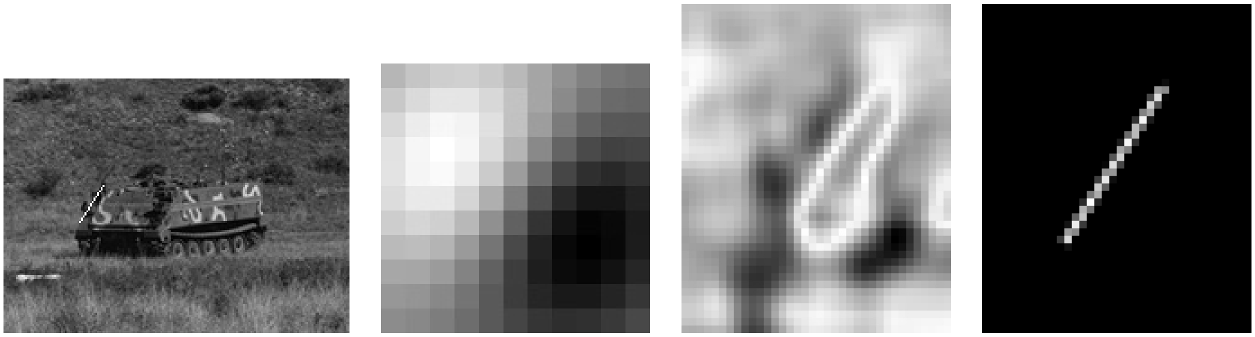
The gradient mask is constructed by rotating the first derivative of a bi-variate Gaussian to match the orientation of the current model edge. There are many precedents both for using tuned edge masks [Can86] and the first derivative Gaussian [TP86]. Others have also used different methods to obtain gradient estimates based on steerable filters [Shu94, FA91] for use in bottom-up edge detection. However, contrary to other approaches [FL88], we are not searching for the maximum gradient of a line of an arbitrary orientation, but rather the gradient for the orientation of the current model edge.

The horizontal first derivative of a bi-variate Gaussian is given by:

$$G(a,b) = -2ye^{-(\frac{a^2}{\sigma_b}+\frac{b^2}{\sigma_b})} \tag{1}$$

*This work was sponsored by the Advanced Research Projects Agency (ARPA) under grant DAAH04-93-G-422, monitored by the U. S. Army Research Office.

†Appears also in the Proceedings of the 1996 ARPA Image Understanding Workshop.



a. Silhouette Line **b.** Gradient Mask **c.** Gradient Response **d.** Weight Mask

Figure 1: Gradient Mask and Response

where (a, b) represents the current position in the filter coordinate system. In order to maximize the response to an arbitrary orientation, the function is rotated to the orientation of the given model line:

$$a = i \cos\phi + j \sin\phi \tag{2}$$
$$b = -i \sin\phi + j \cos\phi \tag{3}$$

where ϕ is the angle of rotation required, and (i, j) are the gradient mask positions being calculated. Figure 1a shows a model edge projected into a color image, along with the gradient mask used (Figure 1b) to obtain the gradient response (Figure 1c).

2.2 Defining the Error Function

The weight mask (shown in Figure 1c) for the current model edge is then convolved with the response to the gradient mask for each pixel lying under the line:

$$\hat{G}_{Line}(k) = \frac{\sum_{i=Line_{xa}}^{Line_{xb}} \sum_{j=Line_{ya}}^{Line_{yb}} |Grad(i,j)| \cdot w(i,j)}{\gamma \cdot \sum_{i=Line_{xa}}^{Line_{xb}} \sum_{j=Line_{ya}}^{Line_{yb}} w(i,j)} \tag{4}$$

where $(Grad(i, j))$ is the gradient mask response, and $w(i, j)$ is a weighting mask based on the distance of the pixel from the true line, thus allowing the computation of $\hat{G}_{Line}(k)$ with sub-pixel accuracy [Pin88]. A threshold for $w(i, j)$ neglects pixels lying outside some radius. The γ term is the largest expected gradient possible for the current mask, and will normalize $\hat{G}_{Line}(k)$ to the range $[0, 1]$ for each line segment. The gradient response is then converted to an error term:

$$E_{Line}(k) = (1 - \hat{G}_{Line}(k)) \tag{5}$$

2.3 Defining the Neighborhoods

The local search algorithm uses the set of moves shown in Figure 2 to perturb each model edge. The error, $E_{Line}(k)$, for each move is calculated, and the best move in the set becomes the new model edge position. The initial step and rotation sizes are set manually. Once a local optimum is achieved, the move sizes are halved and the process continues. Once they fall below a certain threshold, and no further improvement can be made, the current position of the edge is returned as the data line corresponding to the current model edge.

3 Results

The local search algorithm is currently being used in a multi-sensor object recognition algorithm here at Colorado State University [Ant96]. The results of the search are shown in Figure 3. Figure 3a shows the predicted model edges thought to be visible in the image for the given pose hypothesis. Figure 3b shows the data segments extracted for matching to those model features. As can be seen, several of the data lines do not correspond directly to the features desired for matching. However, they are good enough to move the model closer to the desired location, where a new correspondence can be generated. As the model moves closer to the correct position, the local search will find better features in the data for matching.

References

[Ant96] Anthony N. A. Schwickerath and J. Ross Beveridge. Coregistering 3D Models, Range, and Optical Imagery Using Least-Median Squares Fitting. In *Proceedings: Image Understanding Workshop*, page (to appear), Los Altos, CA, February 1996. ARPA, Morgan Kaufman.

[BHP94] J. Ross Beveridge, Allen Hanson, and Durga Panda. Integrated color ccd, flir & ladar based object modeling and recognition. Technical report, Colorado State University and Alliant Techsystems and University of Massachusetts, April 1994.

[BHR86] J. Brian Burns, Allen R. Hanson, and Edward M. Riseman. Extracting Straight Lines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(4):425–455, July 1986.

[Can86] John Canny. A Computation Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–697, November 1986.

[Cla89] James J. Clark. Authenticating Edges Produced by Zero-Crossing Algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-11(1):43–57, January 1989.

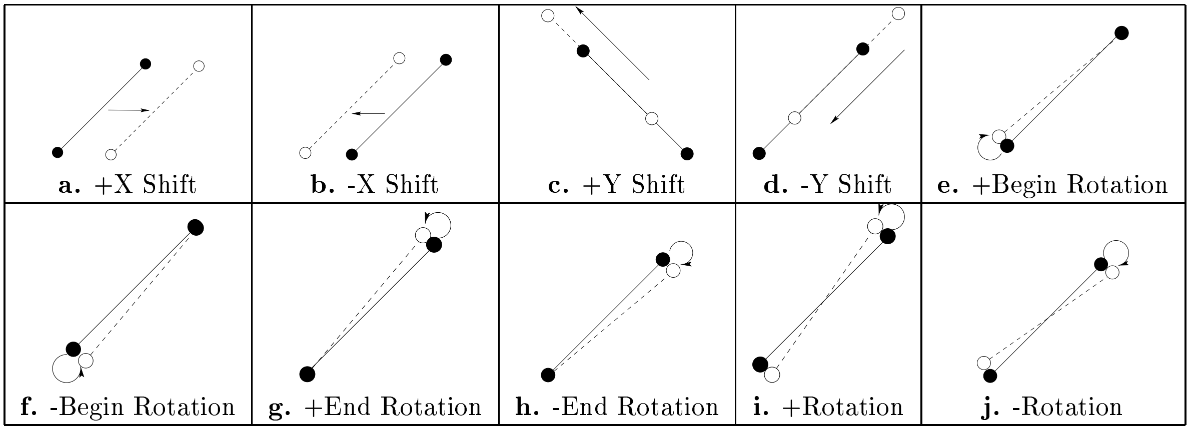


Figure 2: Various Line Movements

a. Model Lines **b.** Data Lines

Figure 3: Linear Features Detected

[FA91] William T. Freeman and Edward H. Adelson. The Design and Use of Steerable Filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-13(9):891–906, September 1991.

[FL88] Pascal Fua and Yvan G. Leclerc. Model driven edge detection. In *Proceedings: Image Understanding Workshop – 1988*, pages 1016 – 1021. DARPA, Morgan Kaufmann, April 1988.

[Hil83] Ellen C. Hildreth. The Detection of Intensity Changes by Computer and Biological Vision Systems. *Computer Vision, Graphics, and Image Processing*, 22:1–27, 1983.

[LB83] David G. Lowe and T. O. Binford. The Perceptual Organization of Visual Images: Segmentation as a Basis for Recognition. In *Proceedings Image Understanding Workshop, Stanford*, pages 203 – 209, June 1983.

[Mar96] Mark R. Stevens and J. Ross Beveridge. Interleaving 3D Model Feature Prediction and Matching to Support Multi-Sensor Object Recognition. In *Proceedings: Image Understanding Workshop*, page (to appear), Los Altos, CA, February 1996. ARPA, Morgan Kaufman.

[MH80] David Marr and Ellen C. Hildreth. Theory of Edge Detection. *Proceedings of the Royal Society of London*, B207:187–217, 1980.

[Pin88] Juan Pineda. A Parallel Algorithm for Polygon Rasterization. In *Proceedings of Siggraph '88*, pages 17–20, 1988.

[Shu94] Alexander Shustorovich. Scale Specific and Robust Edge/Line Encoding with Linear Combinations of Gabor Wavelets. *Pattern Recognition*, 27(5):713–725, 1994.

[Ste95] Mark R. Stevens. Obtaining 3d shilhouettes and sampled surfaces from solid models for use in computer vision. Master's thesis, Colorado State Univeristy, Fort Collins, Colorado, September 1995.

[SWF95] G.D Sullivan, A.D. Worrall, and J.M Ferryman. Visual Object Recognition Using Deformable Models of Vehicles. In *Workshop on Context-Based Vision*, pages 75–86, june 1995.

[TP86] Vincent Torre and Tomaso A. Poggio. On Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(2):147–164, March 1986.