Real-time texture boundary detection from ridges in the standard deviation space

Ray Hidayat
ray.hidayat@pg.canterbury.ac.nz
Richard Green
richard.green@canterbury.ac.nz

Computer Science and Software Engineering Department
University of Canterbury Private Bag 4800, Christchurch 8140, New Zealand

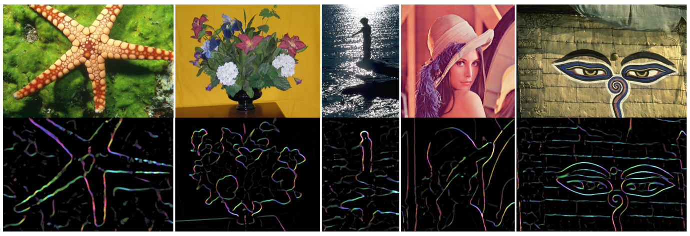

Figure 1: Example results (bottom row) of the proposed realtime texture boundary detector on some images (top row). Hue represents the detected boundary orientation.

1 Overview

This paper presents a novel texture boundary detector called the standard deviation ridge detector. At 43.29 frames per second, it is one of the few texture boundary detectors that can run in realtime. With its Berkeley segmentation benchmark F-statistic of 0.62, the algorithm outperforms all existing realtime texture boundary detectors. The use of this boundary detector would induce noticable improvements to all realtime machine vision applications. In addition, a ridge detector, which is better suited to realtime than other approaches, is presented as part of the proposed algorithm.

2 Results

The proposed detector was tested on the publicly available Berkeley segmentation benchmark [3], which algorithmically (and therefore objectively) rates segmentation algorithms using human-defined ground truths.

An algorithm's performance on this benchmark can be determined by its precision-recall curve. Figure 2 clearly shows the proposed standard deviation ridge algorithm outperforming all other algorithms with its precision-recall curve.

Another way to compare the precision-recall curves is to use the F-statistic – a single value that sums up an algorithm's entire performance on the Berkeley dataset. The proposed detector achieves an F-statistic of 0.62. This can be compared to other state-of-the-art algorithms such as the second moment matrix [2], which achieves 0.57, and the convolved variance ridge algorithm [1], which scores 0.59. Clearly, the proposed algorithm outperforms existing realtime texture boundary detectors.

The favourable results of the proposed detector on the Berkeley benchmark also prove the worth of the novel ridge detection algorithm that is used, and presented in this paper.

2.1 Algorithm

Most texture boundary detectors require multiple convolutions and high-dimensional analysis on an image. This produces state-of-the-art results, but unfortunately, is too computationally intensive to calculate in realtime. As its name suggests, the proposed standard deviation ridge detector works by finding ridges in standard deviation space. To achieve this in realtime, a new ridge detection method was developed, as existing methods were not sufficient. Some examples of this detector's results can be seen in figure 1. This ridge detection algorithm, plus the standard deviation ridge detector itself, are the two primary contributions of this paper.

The standard deviation ridge detector only takes one parameter, the window size parameter k, which in essence determines the cutoff wavelength between when variations should be considered texture (and would therefore be suppressed) as opposed to when they should be considered a boundary (and would therefore be emphasised).

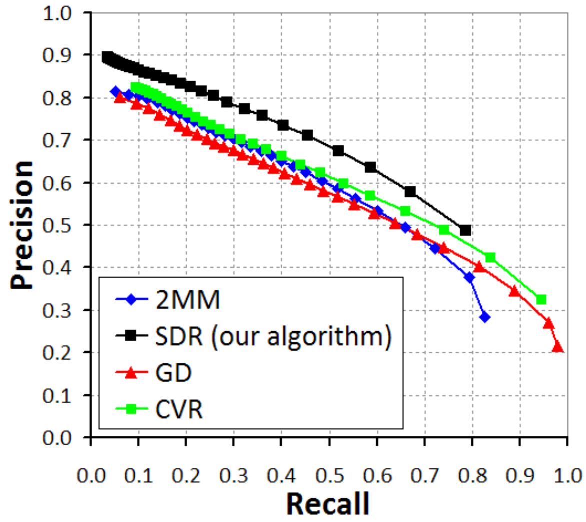

Figure 2: The Berkeley segmentation benchmark precision-recall curves of the proposed standard deviation ridge (SDR) detector against the best realtime detectors: the second moment matrix (2MM), a Gaussian derivative (GD), and the convolved variance ridges (CVR) algorithm. Each detector's parameters were chosen to optimise performance on the Berkeley benchmark.

The overall stages of the proposed algorithm can be expressed by the following equation:

$$b_k(I) = r_k(g_k^s(I)) - \|g_k^s(I)\| \quad \text{where} \quad g_k^s(I) = g_k(s_k(I)) \tag{1}$$

The algorithm begins with an image I, which would be presented to the algorithm in the CIELAB colour space. $s_k(.)$ calculates the standard deviation transform of the image. $g_k(.)$ then calculates the gradient vectors from the standard deviation image. For convenience, the name $g_k^s(.)$ is given for succession of these two operations. From here, the ridge detection is performed in two stages. $r_k(.)$ identifies the ridge potentials from the gradient image. Then basic non-maxima suppression is performed by subtracting the gradient magnitudes $\|g_k^s(I)\|$ from the ridge potentials. This produces the output of the detector.

3 Conclusion

The proposed standard deviation ridge algorithm can be used as a preprocessing stage to improve realtime vision applications by making them texture-aware. It is one of the few realtime boundary detectors available, meaning it is able to suppress unimportant texture variations while emphasising important texture boundaries. The algorithm's Berkeley benchmark F-statistic score of 0.62 and its realtime execution speed of 43.29 frames per second fully back up this claim.

[1] R. Hidayat and R. Green. Texture-suppressing edge detection in realtime. In *IVCNZ08*, 2008.

[2] S. Konishi, A. L. Yuille, J. Coughlan, and S. C. Zhu. Fundamental bounds on edge detection: An information theoretic evaluation of different edge cues. In *CVPR*, pages 573–579, 1999.

[3] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *ICCV*, 2001.