Jump Detection in Regression Surfaces Using Both First-Order and Second-Order Derivatives

Jingran Sun[1] and Peihua Qiu[2]

[1]Countrywide Financial Corporation, Calabasas, California

[2]School of Statistics, University of Minnesota

Abstract

We consider the problem of detecting jump location curves of regression surfaces. In the literature, most existing methods detect jumps in regression surfaces based on estimation of either the first-order or the second-order derivatives of regression surfaces. Methods based on the first-order derivatives are usually better in removing noise, whereas methods based on the second-order derivatives are often superior in localization of the detected jumps. There are a number of existing jump detection procedures using both the first-order and second-order derivatives. But they often use the two types of derivatives in separate steps, and estimation of the derivatives is usually based on intuition. In this paper, we suggest a new procedure for jump detection in regression surfaces, which combines the two types of derivatives in a single criterion. Estimation of the derivatives is based on local quadratic kernel smoothing. Theoretical justifications and numerical studies show that it works well in applications.

Key Words: Edge detection; Gradient; Image processing; Jump location curves; Nonparametric regression; Performance measure; Surface estimation; Thresholding; Zero-crossing.

1 Introduction

Surface estimation is an important problem in many scientific areas, including image processing, geology, meteorology, oceanography, and so forth. In many surface estimation problems, the true surface has jumps, which are often called edges in the image processing literature. Jumps in a surface often represent outlines of objects (e.g., boundary locations of objects in an image), or places where structure of the related 2-D process changes abruptly (e.g., places where an equi-temperature surface in high sky or deep ocean changes dramatically). Therefore, it is important to locate jump positions accurately from observed data, which is the focus of the current paper.

In the literature, most existing 2-D jump detection methods are based on the following properties of the first-order or second-order derivatives of a regression surface. Figure 1(a) shows a 1-D profile of a surface around a jump which is slighted blurred. Its first- and second-order derivatives are displayed in Figures 1(b) and 1(c), respectively. It can be seen from the plots that (1) the first-order derivative peaks at the jump position, (2) the second-order derivative equals zero at the jump position, and (3) the second-order derivative changes its signs from positive to negative when the surface jumps from a lower level to a higher level. In the literature, the second and third properties are called the *zero-crossing* properties of the second-order derivative. In the 2-D setup, the directional first- and second-order derivatives of the regression surface in the direction perpendicular to a given jump location curve would have similar properties. For a more detailed discussion, see Qiu (2005, Chapter 6).

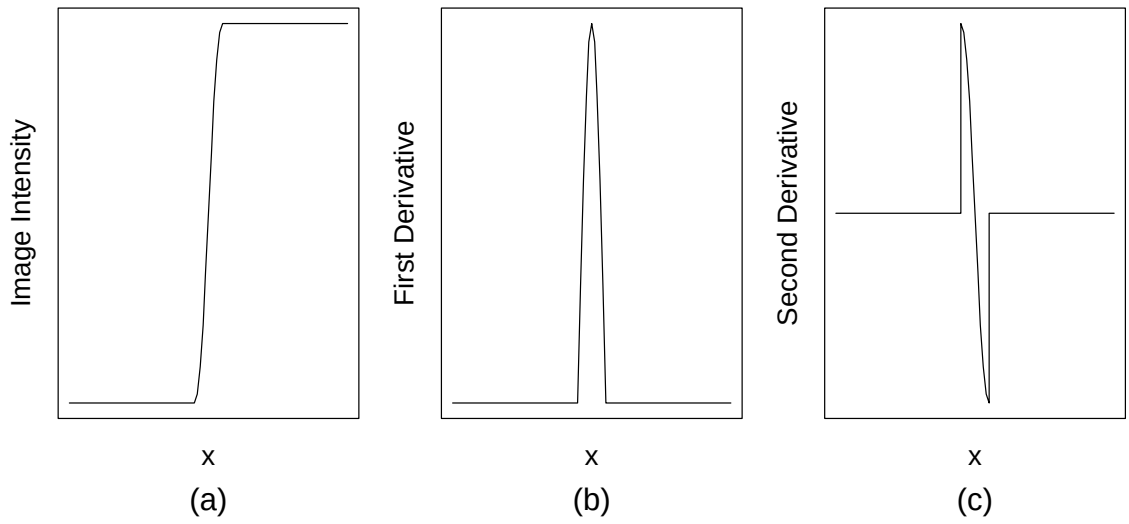

Figure 1: (a): 1-D profile of a surface around a jump that is slightly blurred. (b): First-order derivative of the 1-D profile. (c): Second-order derivative of the 1-D profile.

In the statistical literature, existing 2-D jump detection methods based on estimation of the first-order derivatives include the minimax estimation procedure (Korostelev and Tsybakov 1993), the rotational difference kernel estimation procedure (Müller and Song 1994, Qiu 1997), the estimation procedure using a contract statistic (O'Sullivan and Qian 1994), and the jump tracking algorithms (Hall *et al.* 2001, Hall and Rau 2000, 2002), among some others. All these methods assume that there is only one jump location curve (JLC) in the design space and the JLC satisfies some smoothness conditions. Methods based on the first-order derivatives and capable of detecting arbitrary jumps include the local least squares estimation procedure (Qiu and Yandell 1997), the

wavelet transformation method (Wang 1998), and the generalized Sobel detector (Qiu 2002).

In the image processing literature, many traditional edge detectors are based on estimation of the first-order derivatives, which include the Roberts operators, the Sobel, Prewitt, and Frei-Chen masks, the truncated pyramid masks, the derivative of Gaussian operators, the operators using multilevel masks, and so forth (see, e.g., Gonzalez and Woods 2002, Pratt 1991, Qiu and Bhandarkar 1996, Rosenfeld and Kak 1982). Edge detectors using the zero-crossing properties of the second-order derivatives are also popular. Most such methods are based on the Laplacian operator or the Laplacian of Gaussian (LoG) operator. See Haralick (1984), Huertas and Medioni (1986), Marr and Hildreth (1980), Nalwa and Binford (1986), Torre and Poggio (1986), Clark (1989), among many others. It has been demonstrated that methods based on the second-order derivatives have some good properties, compared to methods based on the first-order derivatives, which include good localization of the detected edges and closed edge contours (e.g., Torre and Poggio 1986). However, these methods are usually sensitive to noise, although pre-smoothing is often performed before edge detection (e.g., Nalwa and Binford 1986). Canny (1986) suggests a benchmark edge detector based on the first-order derivatives and his three criteria for measuring edge detection performance. To deal with spurious edges, thin the thick edges, and produce smooth and connected edges, the "non-maxima suppression" and "thresholding with hysteresis" techniques are also suggested by Canny. As explained and illustrated by Fleck (1992a,b), this edge detector and some of its alternatives have various artifacts in their outputs. To overcome this drawback, Fleck (1992b) suggests combining finite differences with different widths in edge detection. Selection of threshold parameters based on statistical theory is discussed by Rakesh *et al.* (2004). Performance evaluation of various edge detectors is discussed by Heath *et al.* (1997, 1998).

Some authors in the image processing literature suggest using both the first-order and second-order derivatives in edge detection. For instance, besides Fleck (1992b) mentioned above, Haralick (1984) suggests detecting edges by identifying zero-crossings of the directional second-order derivative of the surface in the gradient direction. Then, a zero-crossing pixel is marked as an edge pixel only when the directional first-order derivative in the same direction is not zero. Both the first-order and second-order derivatives used in that procedure are estimated by local orthogonal polynomial surfaces fitting. Clark (1988, 1989) distinguishes phantom edges from authentic edges using both zero-crossing properties of the second-order derivatives and a contrast function defined by the first-order derivatives.

In this paper, we suggest a jump detection procedure using both the first-order and second-order derivatives of the regression surface, which tries to combine the major strengths of the two types of methods mentioned above and to overcome their major limitations at the same time. The suggested procedure uses estimated first-order derivatives for identifying candidate edge pixels and then uses estimated second-order derivatives to produce thin edge segments. One novelty of the procedure is that the two types of derivatives are combined in a single edge detection criterion in a reasonable way; thus, it is convenient to use. Instead of using finite differences to estimate derivatives as did in some existing procedures in the image processing literature, we suggest estimating them using local quadratic kernel smoothing, which is optimal in minimizing the mean squared errors of the estimated derivatives under some regularity conditions (cf., Fan and Gijbels 1996, Chapter 2). To best remove noise and detect true edges, we suggest choosing the window width using a data-driven bootstrap procedure. After the local noise level is properly estimated, a formula for choosing the threshold parameter used in the proposed procedure is provided. Both theoretical justifications and numerical studies show that it would work well in applications.

The rest part of the paper is organized as follows. In Section 2, our jump detection procedure would be described in detail. Its theoretical properties are discussed in Section 3. Numerical studies are presented in Section 4. Several remarks conclude the paper in Section 5. Some technical details are provided in appendices.

2 Jump Detection by Local Quadratic Kernel Smoothing

Assume that n^2 observations $\{Z_{ij},\ \ i,j=1,2,\ldots,n\}$ are generated from the following 2-D regression model:

$$Z_{ij} = g(x_i, y_j) + \varepsilon_{ij},\ \ i,j = 1,2,...,n, \tag{1}$$

where $\{(x_i, y_j),\ \ i,j = 1,2,...,n\}$ are equally spaced design points in the design space $\Omega = [0,1] \times [0,1]$, $\{\varepsilon_{ij},\ \ i,j = 1,2,...,n\}$ are i.i.d. random errors with mean 0 and unknown variance σ^2, and g is an unknown nonparametric regression function. It is further assumed that g is continuous in the design space except at some curves, which are called the jump location curves (JLCs) hereafter. For a mathematically rigorous definition of the JLCs, see Qiu (1998).

For a given design point $(x,y) \in H_n = [h_n, 1-h_n] \times [h_n, 1-h_n]$, we consider its circular

neighborhood:

$$N_n(x,y) = \left\{(u,v) : (u,v) \in [0,1] \times [0,1], \sqrt{(u-x)^2 + (v-y)^2} \leq h_n\right\},$$

where $h_n > 0$ is a bandwidth parameter. In this neighborhood, a local quadratic surface is fitted by the following local polynomial kernel smoothing:

$$\min_{a,b,c,d,e,f} \sum_{(x_i,y_j)\in N_n(x,y)} \{ Z_{ij} - [\, a + b(x_i - x) + c(y_j - y) + d(x_i - x)(y_j - y) + e(x_i - x)^2 + f(y_j - y)^2 \,] \}^2 K\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n}\right), \quad (2)$$

where K is a radially symmetric bivariate density function defined in the unit circle centered at the origin. Let the solution to $\{a, b, c, d, e, f\}$ of problem (2) be denoted as $\{\widehat{a}(x,y), \widehat{b}(x,y), \widehat{c}(x,y), \widehat{d}(x,y), \widehat{e}(x,y), \widehat{f}(x,y)\}$. Then, $(\widehat{b}(x,y), \widehat{c}(x,y))$ is an estimator of the gradient of the true regression surface g at (x,y), and $(2\widehat{e}(x,y), 2\widehat{f}(x,y))$ is an estimator of (g''_{xx}, g''_{yy}). Therefore, if (x,y) is on a JLC, then the estimated gradient magnitude of g at (x,y), which is

$$M(x,y) = \left(\widehat{b}(x,y)^2 + \widehat{c}(x,y)^2\right)^{1/2},$$

should be large; the estimator of the Laplacian of g (i.e., $\nabla^2 g(x,y) = (\partial^2/\partial^2 x + \partial^2/\partial^2 y) g(x,y)$), which is

$$L(x,y) = 2\left(\widehat{e}(x,y) + \widehat{f}(x,y)\right),$$

should have the zero-crossing properties around (x,y), as discussed in Section 1.

2.1 Proposed jump detection procedure

Generally speaking, if arbitrary JLCs are detected, then estimated JLCs by procedures based on the first-order derivatives of the regression surface are thick around the true JLCs and have breaks at some places as well, due to the nature of local smoothing and thresholding used in such procedures. Although there exist some edge thinning and edge linking procedures in the literature (see e.g., Qiu 2005, Section 6.6, Rosenfeld and Kak 1982, Section 11.2), they are often *ad hoc* in nature and it is not guaranteed that we can always obtain good modified results by them. On the other hand, estimated JLCs by procedures based on the second-order derivatives generally have good localization and they are often closed curves in the design space as well (Torre and Poggio 1986). However, such estimators are quite sensitive to noise. As a matter of fact, it is not difficult to

derive the following asymptotic results:

$$\mathrm{Var}(\widehat{b}) = \mathrm{Var}(\widehat{c}) = O\left(\frac{1}{n^2 h_n^4}\right)$$

and

$$\mathrm{Var}(\widehat{e}) = \mathrm{Var}(\widehat{f}) = O\left(\frac{1}{n^2 h_n^6}\right).$$

Because the bandwidth h_n is often chosen in a way that $\lim_{n\to\infty} h_n = 0$, variability of the second-order derivative estimators $\widehat{e}$ and $\widehat{f}$ is much larger than that of the first-order derivative estimators $\widehat{b}$ and $\widehat{c}$.

To overcome the major drawbacks of the two types of jump detection procedures mentioned above, we suggest using properties of both the first-order and second-order derivatives of the regression surface for jump detection. More specifically, in the proposed procedure, a point (x, y) is classified as a jump point only if the estimated gradient magnitude $M(x, y)$ is large *and* the estimated Laplacian $L(x, y)$ has zero-crossing around (x, y). The first condition is often used in jump detection procedures based on the first-order derivatives, whereas the second condition is often used in jump detection procedures based on the second-order derivatives. In the proposed procedure, we suggest using both of them in a single criterion. Our proposed jump detection procedure is described in an algorithm as follows.

Proposed Jump Detection Procedure

- At a given design point (x_i, y_j), for $i, j = 1, 2, \ldots, n$, solve the minimization problem (2) and obtain the solution vector $\{\widehat{a}(x_i, y_j), \widehat{b}(x_i, y_j), \widehat{c}(x_i, y_j), \widehat{d}(x_i, y_j), \widehat{e}(x_i, y_j), \widehat{f}(x_i, y_j)\}$.

- Define the jump detection criterion

$$C_{ij} = \mathrm{sign}(L_{ij}) M_{ij},$$

  where $L_{ij} = L(x_i, y_j)$, $M_{ij} = M(x_i, y_j)$, and $\mathrm{sign}(x)$ is the sign function defined to be 1 when $x > 0$, 0 when $x = 0$, and -1 when $x < 0$.

- Define the categorical version of the jump detection criterion C_{ij} as follows:

$$D_{ij} = \begin{cases} 1, & \text{if } C_{ij} > T \\ -1, & \text{if } C_{ij} < -T \\ 0, & \text{otherwise,} \end{cases}$$

  where $T > 0$ is a threshold parameter.

- Identify detected edge pixels using zero-crossings of the categorical jump detection criterion D_{ij}. That is, the $(i,j)^{th}$ pixel is classified as an edge pixel if $D_{ij} \leq 0$ and the eight values $\{D_{i+s,j+t} : \ s,t = -1,0,1, \ (s,t) \neq (0,0)\}$ have different signs.

Note that the value of D_{ij} is not required to be exactly 0 at a detected edge pixel in the above procedure. Instead, it is required to be nonpositive. That is because, in discrete cases, it might happen that no pixels have their D_{ij} values exactly 0.

2.2 Selection of the threshold parameter

As mentioned in Section 1, jump detection based on the second-order derivatives alone is sensitive to noise, because false zero-crossings can be easily generated by noise. To overcome this difficulty, in our jump detection procedure, information about the first-order derivatives is used for eliminating such false zero-crossings. More specifically, if the estimated surface gradient magnitude is smaller than a threshold, then the detected zero-crossing is regarded as a false zero-crossing. Therefore, selection of the threshold parameter is important, which is discussed below.

It can be shown that the estimated surface gradient $(\widehat{b}(x,y), \widehat{c}(x,y))$ has the following expressions:

$$\begin{aligned} \widehat{b}(x,y) &= \frac{1}{K_{20}} \sum_{i=1}^{n} \sum_{j=1}^{n} (x_i - x) K\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n} \right) Z_{ij} \\ \widehat{c}(x,y) &= \frac{1}{K_{02}} \sum_{i=1}^{n} \sum_{j=1}^{n} (y_j - y) K\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n} \right) Z_{ij}, \end{aligned}$$

where $K_{s_1 s_2} = \sum_{i=1}^{n} \sum_{j=1}^{n} (x_i - x)^{s_1} (y_j - y)^{s_2} K\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n} \right)$, for $s_1, s_2 = 0,1,2,3,4$. Note that some $K_{s_1 s_2}$ values are not used here, but they will be used later. Therefore, their definitions are also given here. From the above expressions, both $\widehat{b}(x,y)$ and $\widehat{c}(x,y)$ are linear combinations of the observations. Thus, they are asymptotically normally distributed with means g'_x and g'_y, respectively (cf. Theorem 3.1 in Section 3). Furthermore, it is easy to check that $\widehat{b}(x,y)$ and $\widehat{c}(x,y)$ are asymptotically independent of each other. So,

$$\frac{\widehat{b}^2(x,y) + \widehat{c}^2(x,y)}{\widehat{\sigma}_n^2}$$

is approximately $\chi_2^2(\delta_n)$ distributed, where the noncentrality parameter is

$$\delta_n = \left([g'_x(x,y)]^2 + [g'_y(x,y)]^2 \right) / \widehat{\sigma}_n^2,$$

and

$$\widehat{\sigma}_n^2 = \widehat{\mathrm{Var}}(\widehat{b}(x,y)) = \widehat{\sigma}^2 \frac{\sum_{i=1}^n \sum_{j=1}^n (x_i - x)^2 K^2\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n}\right)}{K_{20}^2}.$$

In the above expression, $\widehat{\sigma}^2$ denotes a consistent estimator of σ^2, which can be obtained by procedures such as the one proposed by Qiu (2004) using piecewisely linear kernel smoothing. Then, a natural choice for the threshold value of C_{ij} is

$$T_n = \widehat{\sigma}_n \sqrt{\chi^2_{2,1-\alpha_n}(\delta_n)},$$

where $\chi^2_{2,1-\alpha_n}(\delta_n)$ is the $1-\alpha_n$ quantile of the $\chi^2_2(\delta_n)$ distribution. The above formula for T_n depends on the unknown surface gradient, used in the noncentrality parameter δ_n, which should be estimated from data. To this end, we suggest using the smallest estimated surface gradient among four such gradients obtained in the four quadrants of a neighborhood of (x, y) by the local piecewisely linear kernel smoothing procedure, as suggested by Qiu (2004).

2.3 Selection of the bandwidth

The proposed jump detection procedure described in Subsection 2.1 depends on the bandwidth parameter h_n. In 1-D cases, Gijbels and Goderniaux (2004) has demonstrated that this parameter plays an important role in their 1-D jump detection procedure. Based on our numerical experience, this is also true for the current 2-D procedure. In this subsection, we propose a bootstrap procedure for choosing the bandwidth parameter h_n in 2-D cases.

The 2-D bandwidth selection problem is more complicated than its 1-D counterpart. In order to choose the bandwidth properly in 2-D cases, we first need to have a metric to measure the distance between the pointset D of the true jumps and its estimator $\widehat{D}_n$ with the given bandwidth h_n. Qiu (2002) investigated this performance measure problem carefully, and suggested the following performance measure:

$$d^*(\widehat{D}_n, D; h_n) = \omega \frac{|\widehat{D}_n \setminus D|}{|\Omega \setminus D|} + (1-\omega)\frac{|D \setminus \widehat{D}_n|}{|D|},$$

where $0 \le w \le 1$ is a weight, Ω is the entire design space $[0,1]\times[0,1]$, $A \backslash B$ denotes the set of points in A but not in B, and $|A|$ denotes the number of design points in the point set A. Obviously, d^* is a weighted average of the false positive rate and the false negative rate of the related jump detector. In applications, Qiu suggested using $D^* = \{(x_i, y_i) : d((x_i, y_i), D) \le 1/(2n)\}$ to replace D for the purpose of calculating d^*, where $d(\cdot,\cdot)$ denotes the Euclidean distance. That is, all design

points whose Euclidean distances to the true JLCs are less than or equal to $1/(2n)$ (i.e., half of the distance between two horizontally or vertically neighboring design points) are considered in D.

Our bootstrap procedure for choosing the bandwidth h_n is described in the form of a pseudo-algorithm as follows.

Bandwidth Selection via Bootstrap

1. The proposed edge detection procedure is applied to the original dataset $\{(x_i, y_j, Z_{ij}),\ \ i, j = 1, 2, ..., n\}$. The estimated pointset $\widehat{D}_n$ of D is obtained in this step.

2. Construct an estimator $\widehat{g}(x, y)$ of the true regression function $g(x, y)$ and define residuals $\widehat{r}_{ij} = Z_{ij} - \widehat{g}(x_i, y_j)$.

3. Draw with replacement a bootstrap sample of size n^2, denoted as $\{\widehat{r}^*_{ij},\ \ i, j = 1, 2, ..., n\}$, from the residual set $\{\widehat{r}_{ij},\ \ i, j = 1, 2, ..., n\}$, and then obtain a bootstrap dataset $\{(x_i, y_j, Z^*_{ij}),\ \ i, j = 1, 2, ..., n\}$, where $Z^*_{ij} = \widehat{g}(x_i, y_j) + \widehat{r}^*_{ij}$.

4. The edge detection procedure with the same bandwidth as that used in step 1 is then applied to the bootstrap dataset, and an estimator of D from this bootstrap dataset can be obtained.

5. Steps 3 and 4 are repeated for N times. The detected sets of jump points from the N bootstrap datasets are denoted as $\widehat{D}^*_1, \widehat{D}^*_2, \ldots, \widehat{D}^*_N$, respectively. Then, the bootstrap estimator of $d^*(\widehat{D}_n, D; h_n)$ is defined as

$$\widehat{d}^{BT}(\widehat{D}_n, D; h_n) = \frac{1}{N}\sum_{i=1}^{N} d^*(\widehat{D}_n, \widehat{D}^*_i; h_n).$$

   The optimal bandwidth is then approximated by the minimizer of $\widehat{d}^{BT}(\widehat{D}_n, D; h_n)$ with respect to h_n.

In step 2, $\widehat{g}(x, y)$ can be chosen to be any jump-preserving surface estimator proposed in the literature. In all numerical examples presented in this paper, the local piecewisely linear kernel estimator, suggested by Qiu (2004), is used for this purpose.

3 Some Statistical Properties

In this section, we discuss some statistical properties of the jump detection procedure described in Section 2.1. Lemma A.1 in Appendix A gives some asymptotic properties of the local quadratic kernel estimators in regions where the true regression surface is continuous, whereas Lemma A.2 in Appendix A gives some properties of these estimators in regions around the JLCs. The following theorem shows that the set of detected jump points converges almost surely to the set of true jump points in Hausdorff distance.

Theorem 3.1 *Besides the conditions stated in Lemmas A.1 and A.2 in Appendix A, we further assume that* $\frac{(\log(n))^{\frac{1}{2}+\eta}}{nh_n} = o(1)$ *for* $\eta > 0$ *and that the significance level* α_n *satisfies* $\alpha_n = o(1)$ *and* $\frac{Z_{1-\alpha_n}(\log(n))^{\eta}}{nh_n} = O(1)$. *Then,* $\frac{1}{h_n}\lim_{n\to\infty} d_H(\widehat{D}_n \cap \widetilde{\Omega}_\rho, D \cap \widetilde{\Omega}_\rho) = 0$, *a.s., where* $\widetilde{\Omega}_\rho = [\rho, 1-\rho] \times [\rho, 1-\rho]$, $\rho > 0$ *is a constant,* $d_H(A, B)$ *is the Hausdorff distance between two point sets* A *and* B *in the design space defined by*

$$d_H(A,B) = \max\left\{ \sup_{(x,y)\in A} \inf_{(x',y')\in B} \|(x,y)-(x',y')\|, \sup_{(x,y)\in B} \inf_{(x',y')\in A} \|(x,y)-(x',y')\| \right\},$$

and $\|\cdot\|$ *is the Euclidean distance.*

After checking the conditions stated in Theorem 3.1, it can be seen that the convergence rate of $\lim_{n\to\infty} d_H(\widehat{D}_n \cap \widetilde{\Omega}_\rho, D \cap \widetilde{\Omega}_\rho) = 0$, a.s., can reach $o\left(n^{-1}(\log(n))^{\frac{1}{2}+\eta}\right)$, which is the same, up to a logarithm factor, as the minimax convergence rate (see e.g., Korostelev and Tsybakov 1993).

4 Numerical Studies

In this section, we present some results regarding the numerical performance of the proposed procedure described in Section 2, which are organized in two subsections. Subsection 4.1 includes some simulation examples related to edge detection by criteria M, L, and D, respectively. Subsection 4.2 compares the proposed edge detector D with Canny's (1986) procedure using both artificial and real images.

4.1 Numerical performance of edge detectors M, L, and D

Suppose that the true regression function is $g(x,y) = -(x-.5)^2-(y-.5)^2+1$ if $(x-.5)^2+(y-.5)^2 < .25^2$ and $g(x,y) = -(x-.5)^2-(y-.5)^2$ otherwise. Then, g has a unique JLC, which is a circle with the expression $(x-.5)^2+(y-.5)^2 = .25^2$. Observations are generated from model (1) with noise

level σ and sample size n^2. The kernel function is chosen to be the following modified, bivariate, Gaussian density function:

$$K(x,y) = \begin{cases} \frac{1}{2\pi - 3\pi e^{-.5}} \left[\exp\left(-\frac{x^2+y^2}{2} \right) - e^{-.5} \right], & \text{when } x^2 + y^2 \leq 1 \\ 0, & \text{otherwise,} \end{cases}$$

which has the support $\{(x,y) : x^2 + y^2 \leq 1\}$ and which is a continuous function in R^2. We first let $n = 100$ and $\sigma = .25$. One realization of the data is shown in Figure 2(a). In this case, it is known that the true JLC includes 168 pixels. Figures 2(b), 2(c) and 2(d) show the detected edges by the edge detector using the criterion $L(x,y)$, the edge detector using the criterion $M(x,y)$, and our proposed edge detector $D(x,y)$ which uses both $L(x,y)$ and $M(x,y)$, respectively. The first edge detector considered is a representative of all edge detectors using second-order derivatives, whereas the second one is a typical edge detector based on first-order derivatives. The bandwidth h_n used in each procedure is fixed to be 5 pixels. The thresholding parameter is chosen such that there are exactly 168 detected pixels in each case. From the plots, it can be seen that detected edges in plots (c) and (d) are less noisy than those in plot (b), and that detected edges in plot (d) are thinner and have less breaks than those in plot (c). These results confirm our intuition and theoretical justifications discussed before.

Next, we try to measure the performance of the related jump detectors quantitatively using the performance measure $d^*(\widehat{D}_n, D; h_n)$ defined in Subsection 2.3, in which ω is fixed at 0.5 in this section. Without loss of generality, the bandwidth h_n is assumed to take the value of k/n, where $k < n$ is a positive integer. We fix $n = 100$ and let k and σ change. Three edge detectors using $L(x,y)$, $M(x,y)$, and $D(x,y)$ are considered as before. For each edge detector, the type I error probability α_n is set to be .01. When calculating the performance measure d^*, a localization error of Euclidean distance of 2 pixels is tolerated using the algorithm proposed by Liu and Haralick (2002). For each σ value, the best bandwidth h_n is chosen such that the performance measure d^* is minimized. For the proposed detector, the bandwidth selected via bootstrap is also computed. The optimal bandwidths and the corresponding values of d^* are presented in Table 1, when σ equals .25, .5, .75, or 1. These results are based on 100 replications in each case. When computing the bootstrap bandwidth, the bootstrap sample size is fixed at 50 in each replication. From Table 1, it can be seen that (1) the proposed procedure performs much better than the other two procedures in terms of d^*, (2) the bandwidth needs to be chosen larger for noisier data, which is especially true for the proposed procedure, and (3) the bandwidth selection procedure via bootstrap works well,

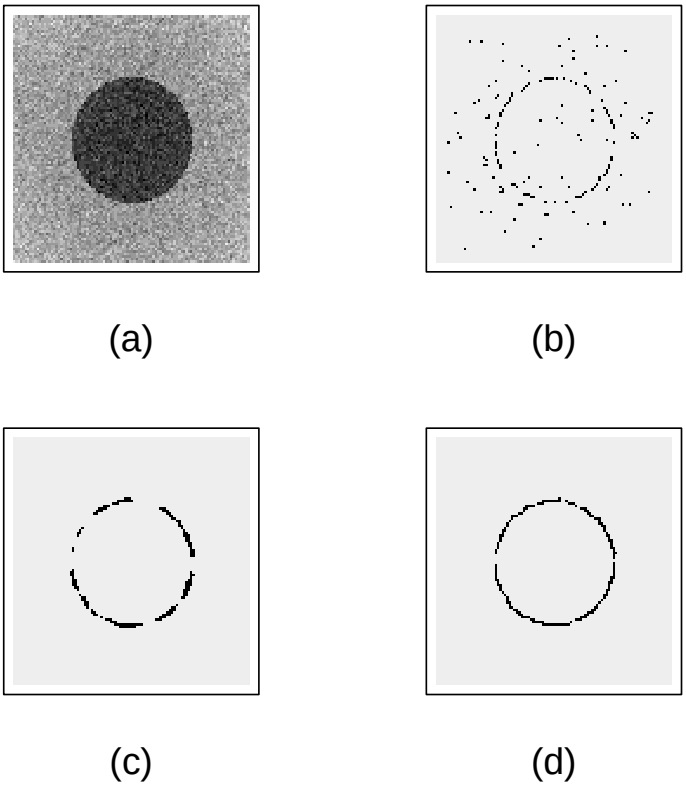

Figure 2: (a): Observed image. (b): Detected edges using criterion $L(x, y)$. (c): Detected edges using criterion $M(x, y)$. (d): Detected edges by the proposed procedure.

especially when σ is small to moderate, and it provides close-to-optimal results in terms of d^*.

4.2 Comparison to Canny's edge detector

Canny's (1986) edge detector is well received in image analysis applications. In this subsection, we compare our proposed edge detector D with Canny's edge detector using both artificial and real images. The source C codes of Canny's edge detector used in this paper were downloaded from the web page http://marathon.csee.usf.edu/edge/edge_detection.html, maintained by Dr. M. Heath. See Heath *et al.* (1997) for a detailed description about Canny's edge detector and its implementation.

In the first example, an artificial image is generated from the following true image intensity function: $g(x, y) = 512x$, when $(x, y) \in [0, 0.5]\times[0, 0.5]$; $g(x, y) = 250$, when $(x, y) \in (0.5, 1]\times[0, 0.5]$; $g(x, y) = 244$, when $(x, y) \in (0.5, 1] \times (0.5, 1]$; and $g(x, y) = 122$, when $(x, y) \in [0, 0.5] \times (0.5, 1]$. Then, g has the following four edge segments: (i) $\{(x, y) : x = 0.5, y \in [0, 0.5]\}$ with constant,

Table 1: For each σ value and each edge detector, the bandwidth and the corresponding d^* value (in parenthesis) are presented.

σ	0.25	0.5	0.75	1.0
Edge detector using L with best bandwidth	0.03 (.478)	0.03 (.499)	0.03 (.500)	0.03 (.500)
Edge detector using M with best bandwidth	0.02 (.010)	0.03 (.014)	0.04 (.022)	0.06 (.035)
Proposed edge detector with best bandwidth	0.05 (.002)	0.09 (.002)	0.10 (.003)	0.12 (.010)
Proposed edge detector with bootstrap bandwidth	0.05 (.002)	0.10 (.002)	0.12 (.004)	0.13 (.011)

small jump magnitude 6, (ii) $\{(x, y) : y = 0.5, x \in (0.5, 1]\}$ with constant, small jump magnitude 6, (iii) $\{(x, y) : x = 0.5, y \in (0.5, 1]\}$ with constant, large jump magnitude 122, and (iv) $\{(x, y) : y = 0.5, x \in [0, 0.5]\}$ with variable jump magnitude $-122 + 512x$. Figure 3 presents g in a 3-D plot.

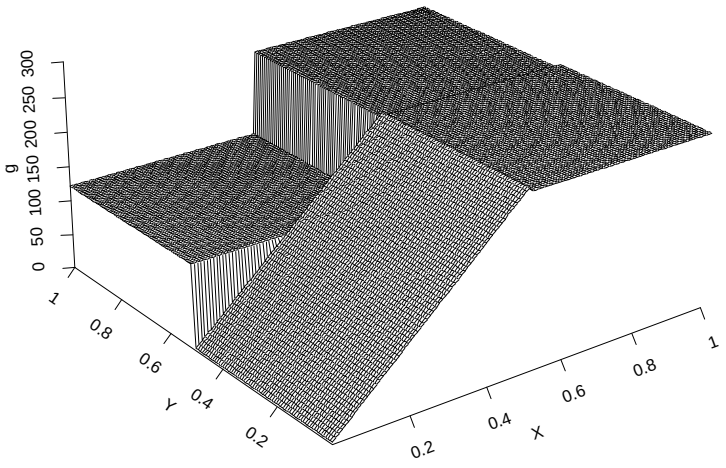

Figure 3: 3-D plot of g.

When $n = 128$, the true image intensities $\{g(i/n, j/n) : i, j = 1, 2, \ldots, n\}$ are shown in Figure 4(a) by an image. We then apply both the proposed edge detector and Canny's edge detector to the image in Figure 4(a). The detected edges are shown in Figures 4(b) and 4(c), respectively. In the proposed procedure, $h_n = 5/n$ and $\alpha_n = 0.01$. In Canny's procedure, there are three parameters σ_G, T_l, and T_h, denoting the standard deviation of the Gaussian pre-filter, the low and high hysteresis thresholds, respectively. We tried all combinations of these three parameters listed in Table 5 of Heath *et al.* (1997), which were found best for the twenty-eight test images in that paper, and selected ($\sigma_G = 0.6, T_l = 0.3, T_h = 0.3$) that produced results with best visual

impression. From the plots, it can be seen that the proposed procedure detects all edges well, except a couple edge pixels around the crossing point $(0.5, 0.5)$ and the point $(0.25, 0.5)$ at which the jump magnitude is 0. Canny's procedure has difficulty to handle these two places too; further, the detected edge segment corresponding to the true edge segment $\{(x, y) : x = 0.5, y \in [0, 0.5]\}$ seems shifted several pixels to left. We then add i.i.d. random errors from $N(0, 0.5^2)$ distribution to all pixels with $j \leq n/2$ and i.i.d. random errors from $N(0, 10^2)$ distribution to all pixels with $j > n/2$. The noisy image is displayed in Figure 4(d). The detected edges by the proposed procedure with $h_n = 7/n$ and $\alpha_n = 0.01$ are displayed in Figure 4(e). From the plot, it can be seen that, overall it detects edges well, the detected edge segments corresponding to the true edge segments $\{(x, y) : x = 0.5, y \in [0, 0.5]\}$ and $\{(x, y) : y = 0.5, x \in (0.5, 1]\}$ have several breaks due mainly to the small jump magnitudes in the true image intensities, and the breaks corresponding to the true edge segment $\{(x, y) : y = 0.5, x \in (0.5, 1]\}$ are wider due to the fact that the noise level on the upper side of that segment is much larger than the noise level in the pixels around the other edge segment. As a comparison, Canny's procedure can not detect edges well in this case. We tried many combinations of its three parameters, including all combinations listed in Table 5 of Heath *et al.* (1997). Figure 4(f)–4(i) display its detected edges when (σ_G, T_l, T_h) equals $(0.6, 0.3, 0.970)$, $(1.2, 0.3, 0.974)$, $(1.8, 0.5, 0.900)$, and $(5.0, 0.3, 0.900)$, respectively. The four σ_G values considered here represent different levels of pre-smoothing. From the plots, it can be seen that the two edge segments $\{(x, y) : x = 0.5, y \in [0, 0.5]\}$ and $\{(x, y) : y = 0.5, x \in (0.5, 1]\}$ with small jump magnitudes in the image intensities are completely missed by Canny's procedure while some false edges are detected in all four situations. This result is consistent with some of Fleck's (1992a, Figures 8–14) findings that Canny's procedure is hard to detect edges at the top or bottom of a ramp, especially when the ramp slope is relatively large and the jump magnitudes in image intensities at the edge pixels are relatively small.

Next, we consider two real images: the cameraman image and a log-transformed C-band, HH-polarization, synthetic aperture radar (SAR) image of an area near Thetford forest, England. The first one is a standard test image in the computer science literature, and the second one is a standard test image in the statistical literature (cf., e.g., Glasbey and Horgan 1995, Qiu 1992). The original cameraman image is shown in Figure 5(a), which has 256×256 pixels with gray levels in the range $[0, 255]$, and the original SAR image is shown in Figure 5(g), which has 250×250 pixels with gray levels in the range $[0, 255]$. Their noisy versions with i.i.d. noise from $N(0, 30^2)$ distribution

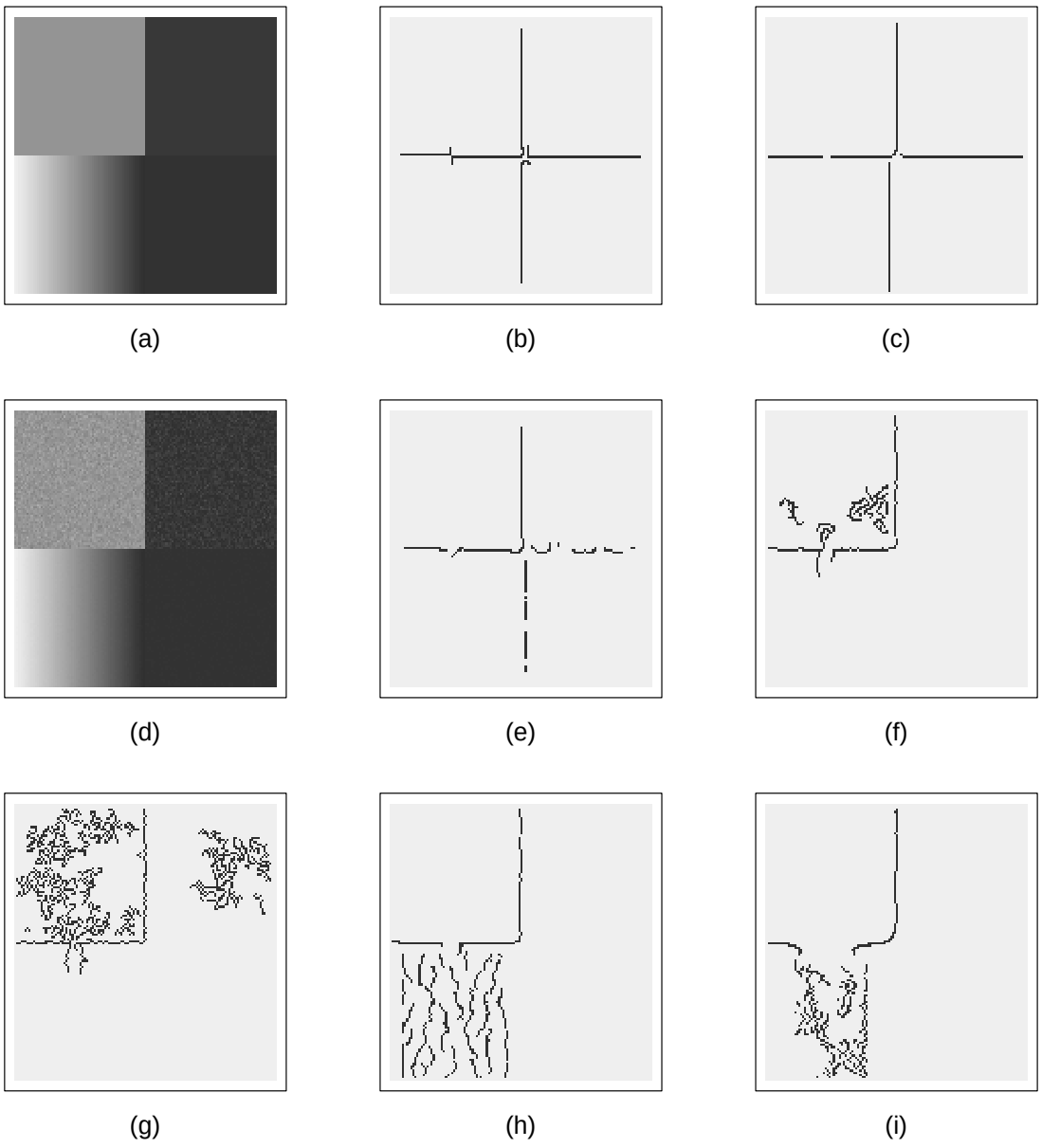

Figure 4: (a): True image when $n = 128$. (b): Detected edges from (a) by the proposed procedure. (c): Detected edges from (a) by Canny's procedure. (d): Noisy image with i.i.d. random errors from $N(0, 0.5^2)$ in all pixels with $j \leq n/2$ and i.i.d. random errors from $N(0, 10^2)$ in all pixels with $j > n/2$. (e) Detected edges from (d) by the proposed procedure. (f)–(i): Detected edges from (d) by Canny's procedure in various cases.

are shown in Figure 5(d) and Figure 5(j), respectively. The proposed edge detector and Canny's procedure are then applied to both original images and their noisy versions. For the proposed procedure, α_n is fixed at 0.01 and h_n is respectively chosen to be 7, 7, 9, and 11 in cases with the original cameraman image, its noisy version, the original SAR image, and the noisy SAR image. In Canny's procedure, its three parameters (σ_G, T_l, T_h) are searched as in the previous example, to be respectively (0.6,0.3,0.850), (1.2,0.2,0.800), (1.2,0.4,0.950), and (1.8,0.3,0.950) in the four cases mentioned above. The detected edges of the proposed procedure are shown in Figures 5(b), 5(e), 5(h), and 5(k), and detected edges of Canny's procedure are shown in Figures 5(c), 5(f), 5(i), and 5(l). From the plots, it can be seen that, in the cameraman example, Canny's procedure detects the camera and the cameraman reasonably well, and it fails detecting some other edges (e.g., the tower in the right part of the image), especially when the noise level is high (i.e., in plot (f)). As a comparison, the proposed procedure detects the major edges reasonably well, but its detected edges have some breaks, which is partially due to the fact that we did not make any post-processing (e.g., edge linking in the "hysteresis" step of Canny's procedure) in our procedure. In the SAR image example, we can see that Canny's procedure detects some unnecessary details, especially when noise level is high (i.e., in plot (l)). The proposed procedure, on the other hand, detects the major edges well, but its detected edges seem not straight enough in some parts of the image. Again, some post-processing might be helpful.

5 Conclusions

We have presented a jump detection procedure based on local quadratic kernel smoothing. It makes use of the information about jumps contained in both the first-order and second-order derivatives of the regression surface. A data-driven bandwidth selection procedure via bootstrap is also suggested. Numerical examples show that the proposed jump detection procedure works well in practice.

Acknowledgments: The authors thank three referees for many constructive comments and suggestions which greatly improved the quality of the paper. Peihua Qiu's research was partially supported by an National Security Agency grant and an National Science Foundation grant.

References

Canny, J. (1986), "A Computational Approach to Edge Detection," *IEEE Transaction on Pattern*

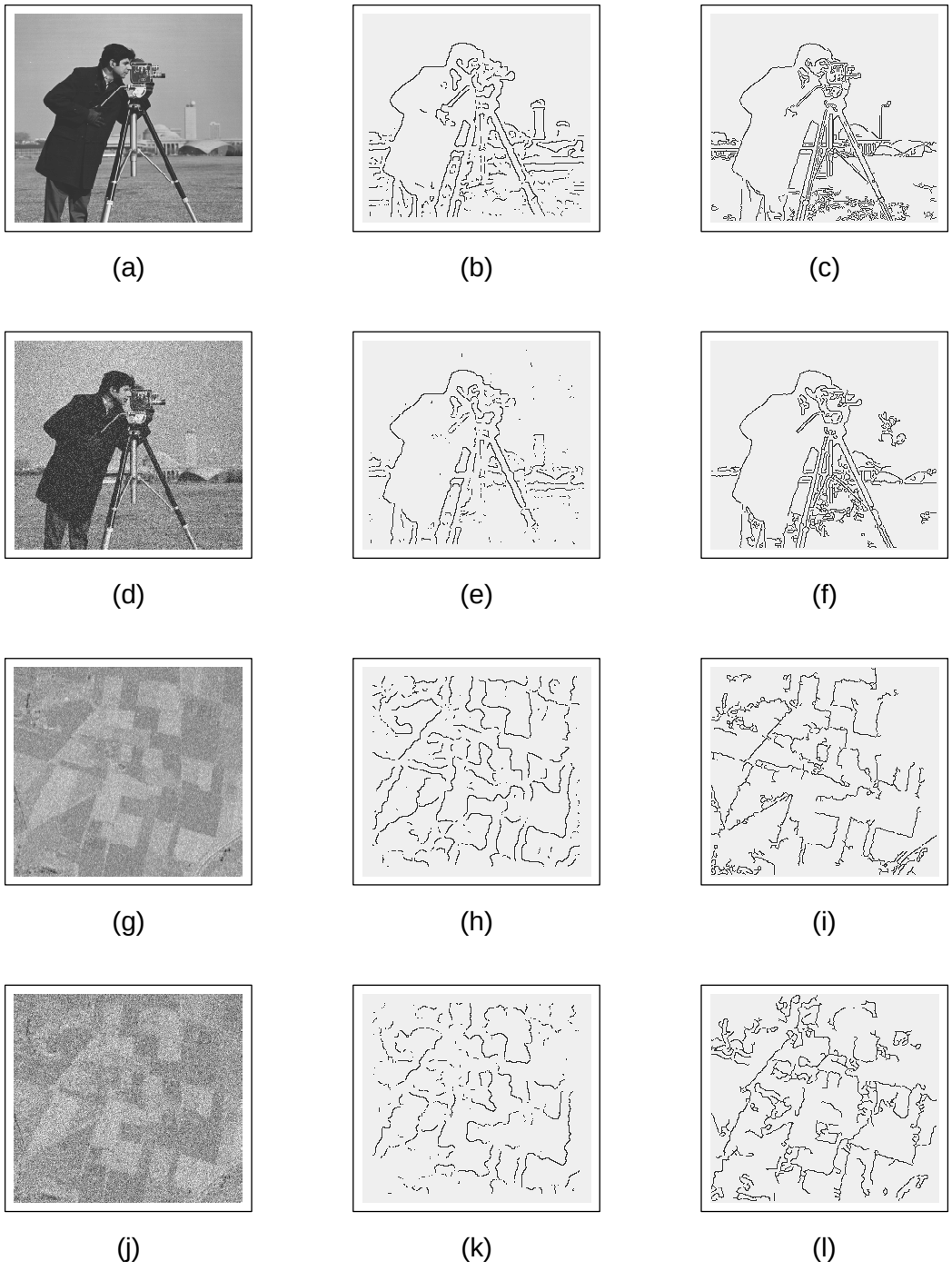

Figure 5: (a): Original cameraman image. (b): Detected edges by the proposed procedure from (a). (c): Detected edges by Canny's procedure from (a). (d)-(f): Noisy version of the cameraman image and the corresponding detected edges. (g)-(i): Original SAR image and the corresponding detected edges. (j)-(l): Noisy version of the SAR image and the corresponding detected edges.

Analysis and Machine Intelligence, **8**, 679–698.

Clark, J.J. (1988), "Singularity theory and phantom edges in scale space," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **10**, 720–727.

Clark, J.J. (1989), "Authenticating edges produced by zero-crossing algorithms," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **11**, 43–57.

Fleck, M.M. (1992a), "Some defects in finite-difference edge finders," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **14**, 337–345.

Fleck, M.M. (1992b), "Multiple widths yield reliable finite differences," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **14**, 412–428.

Gijbels, I., and Goderniaux, A.C. (2004), "Bandwidth selection for changepoint estimation in nonparametric regression," *Technometrics*, **46**, 76–86.

Glasbey, C., and Horgan, G. (1995), *Image Analysis for the Biological Sciences*, John Wiley & Sons: New York.

Gonzalez, R.C., and Woods, R.E. (2002), *Digital Image Processing (2nd ed.)*, Prentice Hall: New York.

Haralick, R.M. (1984), "Digital step edges from zero crossing of second directional derivatives," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **6**, 58–68.

Hall, P., Peng, L., and Rau, C. (2001), "Local likelihood tracking of fault lines and boundaries," *Journal of the Royal Statistical Society (Series B)*, **63**, 569–582.

Hall, P., and Rau, C. (2000), "Tracking a smooth fault line in a response surface," *The Annals of Statistics*, **28**, 713–733.

Hall, P., and Rau, C. (2002), "Likelihood-based confidence bands for fault lines in response surfaces," *Probability Theory and Related Fields*, **124**, 26–49.

Heath, M.D., Sarkar, S., Sanocki, T., and Bowyer, K.W. (1997), "A robust visual method for assessing the relative performance of edge-detection algorithms," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **19**, 1338–1359.

Heath, M.D., Sarkar, S., Sanocki, T., and Bowyer, K.W. (1998), "Comparison of edge detectors - a methodology and initial study," *Computer Vision and Image Understanding,* **69**, 38–54.

Huertas, A., and Medioni, G. (1986), "Detection of intensity changes using Laplacian-Gaussian masks," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **8**, 651–664.

Korostelev, A.P., and Tsybakov, A.B. (1993), *Minimax Theory of Image Reconstruction,* Lecture Notes in Statistics, **82**, Springer: New York.

Liu, G., and Haralick,R. (2002), "Optimal matching problem in detection and recognition performance evaluation," *Pattern Recognition*, **35**, 2125–2139.

Marr, D., and Hildreth, E. (1980), "Theory of edge detection," *Proceedings of the Royal Society of London. Series B*, **207**, 187–217.

Müller, H.G., and Song, K.S. (1994), "Maximin estimation of multidimensional boundaries," *Journal of Multivariate Analysis*, **50**, 265-281.

Nalwa, V.S., and Binford, T.O. (1986), "On detecting edges," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **8**, 699–714.

O'Sullivan, F., and Qian, M. (1994), "A regularized contrast statistic for object boundary estimation - implementation and statistical evaluation," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **16**, 561–570.

Pratt, W.K. (1991), *Digital Image Processing (2nd ed.)*, John Wiley & Sons: New York.

Qiu, P. (1997), "Nonparametric estimation of jump surface," *Sankhya (Series A),* **59**, 268–294.

Qiu, P. (1998), "Discontinuous regression surfaces fitting," *The Annals of Statistics,* **26**, 2218–2245.

Qiu, P. (2002), "A nonparametric procedure to detect jumps in regression surfaces," *Journal of Computational and Graphical Statistics*, **11**, 799–822.

Qiu, P. (2004), "The local piecewisely linear kernel smoothing procedure for fitting jump regression surfaces," *Technometrics*, **46**, 87–98.

Qiu, P. (2005), *Image Processing and Jump Regression Analysis,* John Wiley & Sons: New York.

Qiu, P., and Bhandarkar, S.M. (1996), "An edge detection technique using local smoothing and statistical hypothesis testing," *Pattern Recognition Letters,* **17**, 849–872.

Qiu, P., and Yandell, B.(1997), "jump detection in regression surfaces," *Journal of Computational and Graphical Statistics*, **6**, 332–354.

Rakesh, R.R., Chaudhuri, P., and Murthy, C.A. (2004), "Thresholding in edge detection: a statistical approach," *IEEE Transactions in Image Processing,* **13**, 927–935.

Rosenfeld, A., and Kak, A.C. (1982), *Digital Picture Processing (2nd ed.),* Academic Press: New York.

Torre, V., and Poggio, T.O. (1986), "On edge detection," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **8**, 147–163.

Wang, Y. (1998), "Change curve estimation via wavelets," *Journal of the American Statistical Association*, **93**, 163–172.

Appendix

A Two Lemmas

Lemma A.1 *Suppose that the regression function g has continuous fourth-order derivatives in each closed subset of $[0,1]\times[0,1]$ where g is continuous, $E(\varepsilon_{11}^2)<\infty$, the kernel function K is a Lipschitz-1 continuous, radially symmetric, bivariate density function defined in a unit circle centered at the origin, the bandwidth h_n satisfies the conditions that $h_n=o(1)$ and $\frac{\sqrt{\log(n)}}{nh_n}=o(1)$, then*

$$\begin{aligned}
\|\widehat{a}-g\|_{\Omega_{h_n}} &= O(h_n^4)+O\left(\frac{\sqrt{\log(n)}}{nh_n}\right), \quad a.s.,\\
\|\widehat{b}-g'_x\|_{\Omega_{h_n}} &= O(h_n^2)+O\left(\frac{\sqrt{\log(n)}}{nh_n^2}\right), \quad a.s.,\\
\|\widehat{c}-g'_y\|_{\Omega_{h_n}} &= O(h_n^2)+O\left(\frac{\sqrt{\log(n)}}{nh_n^2}\right), \quad a.s.,\\
\|\widehat{d}-g''_{xy}\|_{\Omega_{h_n}} &= O(h_n^2)+O\left(\frac{\sqrt{\log(n)}}{nh_n^3}\right), \quad a.s.,\\
\|\widehat{e}-g''_{xx}/2\|_{\Omega_{h_n}} &= O(h_n^2)+O\left(\frac{\sqrt{\log(n)}}{nh_n^3}\right), \quad a.s.,
\end{aligned}$$

$$\|\widehat{f} - g''_{yy}/2\|_{\Omega_{h_n}} \quad = \quad O(h_n^2) + O\left(\frac{\sqrt{\log(n)}}{nh_n^3}\right), \quad a.s.,$$

where $\Omega_{h_n} = \{(x,y) : (x,y) \in [h_n, 1-h_n] \times [h_n, 1-h_n], \sqrt{(x-x')^2 + (y-y')^2} \geq h_n$, *where* (x', y') *is any point on the JLC}, and* $\|g\|_{\Omega_{h_n}}$ *is defined to be* $\max_{(x,y)\in\Omega_{h_n}} |g(x,y)|$.

Proof: It can be checked that the solution vector $\widehat{\beta}$ to $\beta = (a, b, c, d, e, f)^T$ of the minimization problem (2) has the matrix expression

$$\widehat{\beta} = (\widetilde{X}^T \widetilde{K} \widetilde{X})^- \widetilde{X}^T \widetilde{K} Z,$$

where $\widetilde{X}$ is the $n^2 \times 6$ design matrix, $Z = (Z_{11}, Z_{12}, \ldots, Z_{nn})^T$, $\widehat{\beta} = (\widehat{a}, \widehat{b}, \widehat{c}, \widehat{d}, \widehat{e}, \widehat{f})^T$, and $\widetilde{K}$ is the $n^2 \times n^2$ diagonal weight matrix with its $[(i-1)n+j]$-th element being $K\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n}\right)$. Because the design points $\{(x_i, y_j), \quad i, j = 1, 2, ..., n\}$ are equally spaced and the kernel function K is radially symmetric, it can be checked that

$$(\widetilde{X}^T \widetilde{K} \widetilde{X})^- = \frac{1}{\Delta} \begin{pmatrix} M_{11}^* & 0 & 0 & 0 & M_{51}^* & M_{61}^* \\ 0 & M_{22}^* & 0 & 0 & 0 & 0 \\ 0 & 0 & M_{33}^* & 0 & 0 & 0 \\ 0 & 0 & 0 & M_{44}^* & 0 & 0 \\ M_{15}^* & 0 & 0 & 0 & M_{55}^* & M_{65}^* \\ M_{16}^* & 0 & 0 & 0 & M_{56}^* & M_{66}^* \end{pmatrix}, \tag{3}$$

where M_{ij}^* is (i,j)-th cofactor of $M = \widetilde{X}^T \widetilde{K} \widetilde{X}$ and Δ is the determinant of M. By expression (3), we have

$$\widehat{a}(x,y) \quad = \quad \frac{1}{\Delta} \sum_{i=1}^{n} \sum_{j=1}^{n} \left\{ M_{11}^* + M_{51}^* (x_i - x)^2 + M_{61}^* (y_j - y)^2 \right\} K\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n}\right) Z_{ij}.$$

For any point $(x,y) \in \Omega_{h_n}$, by using the relationship between M_{ij}^* and M_{ij}, it can be checked that

$$\begin{aligned} E(\widehat{a}(x,y)) \quad = \quad & g(x,y) + \frac{1}{\Delta} \sum_{i=1}^{n} \sum_{j=1}^{n} \left\{ M_{11}^* + M_{51}^* (x_i - x)^2 + M_{61}^* (y_j - y)^2 \right\} \left\{ \frac{1}{24} [g_x^{(4)}(x,y)(x_i - x)^4 + \right. \\ & \left. \frac{1}{6} g_x^{(2)}(x,y) g_y^{(2)}(x,y)(x_i - x)^2 (y_j - y)^2 + g_y^{(4)}(x,y)(y_j - y)^4] \right\} K\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n}\right) \\ & + o(h_n^4) \end{aligned}$$

and

$$\frac{1}{\Delta} \sum_{i=1}^{n} \sum_{j=1}^{n} \left\{ M_{11}^* + M_{51}^* (x_i - x)^2 + M_{61}^* (y_j - y)^2 \right\} (x_i - x)^4 K\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n}\right)$$

$$
\begin{aligned}
&= \frac{1}{\Delta}\{K_{40}M_{11}^* + K_{60}M_{51}^* + K_{42}M_{61}^*\} \\
&= \frac{1}{\Delta}\begin{vmatrix} K_{40} & 0 & 0 & 0 & K_{20} & K_{02} \\ 0 & K_{20} & 0 & 0 & 0 & 0 \\ 0 & 0 & K_{02} & 0 & 0 & 0 \\ 0 & 0 & 0 & K_{22} & 0 & 0 \\ K_{60} & 0 & 0 & 0 & K_{40} & K_{22} \\ K_{42} & 0 & 0 & 0 & K_{22} & K_{04} \end{vmatrix} = O(h_n^4).
\end{aligned}
$$

In the last equation above, we have used the results that $K_{ij} = O\left(\frac{1}{n^2 h_n^{i+j+2}}\right)$, for $i, j = 0, 1, 2, 3, 4$. So $E(\widehat{a}(x,y)) = g(x,y) + O(h_n^4)$. It can be checked that this result is uniformly true for $(x,y) \in \Omega_{h_n}$. Therefore, we have

$$
\|E(\widehat{a}) - g\|_{\Omega_{h_n}} = O(h_n^4). \tag{4}
$$

On the other hand, for any $(x,y) \in \Omega_{h_n}$,

$$
\begin{aligned}
\widehat{a}(x,y) - E(\widehat{a}(x,y)) &= \frac{1}{\Delta}\sum_{i=1}^{n}\sum_{j=1}^{n}\{M_{11}^* + M_{51}^*(x_i - x)^2 + M_{61}^*(y_j - y)^2\}\epsilon_{ij}K\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n}\right) \\
&= \frac{O(1)}{n^2 h_n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\epsilon_{ij}K\left(\frac{x_i - x}{h_n}, \frac{y_j - y}{h_n}\right) \\
&= O\left(\frac{\sqrt{\log(n)}}{nh_n}\right), \text{ a.s.}
\end{aligned} \tag{5}
$$

Then, by (4) and (5),

$$
\|\widehat{a} - g\|_{\Omega_{h_n}} = O(h_n^4) + O\left(\frac{\sqrt{\log(n)}}{nh_n}\right), \text{ a.s.}
$$

By similar arguments, the asymptotic properties of $\widehat{b}, \widehat{c}, \widehat{d}, \widehat{e}$, and $\widehat{f}$ stated in Theorem 3.1 can be proved. □

From Lemma A.1, it is easy to derive the following results:

$$
\left\|M - \sqrt{(g_x')^2 + (g_y')^2}\right\|_{\Omega_{h_n}} = O(h_n^2) + O\left(\frac{\sqrt{\log(n)}}{nh_n^2}\right), \text{a.s.},
$$

$$
\left\|L - \nabla^2 g\right\|_{\Omega_{h_n}} = O(h_n^2) + O\left(\frac{\sqrt{\log(n)}}{nh_n^3}\right), \text{a.s.},
$$

where M and L are the estimated surface gradient and Laplacian defined in Section 2.

Lemma A.2 *Given the conditions stated in Lemma A.1, we further assume that the surface gradient has one-sided limits at the JLCs and that the JLCs have one-sided tangent lines at each*

point on them. Let Γ *be a given JLC and* (x, y) *be a given point in the design space. If* (x_τ, y_τ) *is the closest point on* Γ *to* (x, y) *and the Euclidean distance between the two points is* τh_n*, where* $0 \leq \tau < 1$ *is a constant;* $C_\tau > 0$ *is the jump size of* g *at* (x_τ, y_τ)*;* $Q_n^{(1)}(x, y)$ *and* $Q_n^{(2)}(x, y)$ *are two different parts of* $N_n(x, y)$ *separated by* Γ *with a positive jump at* (x_τ, y_τ) *from* $Q_n^{(1)}(x, y)$ *to* $Q_n^{(2)}(x, y)$*;* $Q^{(1)}$ *and* $Q^{(2)}$ *are the two corresponding parts of the support of* K*, then,*

(i)

$$\widehat{b}(x, y) = \frac{C_\tau}{\widetilde{K}_{02} h_n} \int\int_{Q^{(2)}} uK(u, v)\, dudv + o\left(\frac{1}{h_n}\right), \ a.s.,$$

(ii)

$$\widehat{c}(x, y) = \frac{C_\tau}{\widetilde{K}_{20} h_n} \int\int_{Q^{(2)}} vK(u, v)\, dudv + o\left(\frac{1}{h_n}\right), \ a.s.,$$

(iii) if $\tau = 0$*, that is, if* (x, y) *is on* Γ*, then*

$$L(x, y) = o\left(\frac{1}{h_n^2}\right),$$

(iv) if $0 < \tau < 1$ *and* $(x, y) \in Q_n^{(1)}(x, y)$*, then*

$$L(x, y) = \frac{C_1}{h_n^2} + o\left(\frac{1}{h_n^2}\right),$$

where

$$C_1 = \frac{2C_\tau}{\widetilde{\Delta}} \left(2\widetilde{M}_{15}^* \int\int_{Q^{(2)}} K(u, v)\, dudv + (\widetilde{M}_{55}^* + \widetilde{M}_{56}^*) \int\int_{Q^{(2)}} (u^2 + v^2) K(u, v)\, dudv\right) > 0,$$

(v) if $0 < \tau < 1$ *and* $(x, y) \in Q_n^{(2)}(x, y)$*, then*

$$L(x, y) = -\frac{C_2}{h_n^2} + o\left(\frac{1}{h_n^2}\right),$$

where

$$C_2 = \frac{2C_\tau}{\widetilde{\Delta}} \left(2\widetilde{M}_{15}^* \int\int_{Q^{(1)}} K(u, v)\, dudv + (\widetilde{M}_{55}^* + \widetilde{M}_{56}^*) \int\int_{Q^{(1)}} (u^2 + v^2) K(u, v)\, dudv\right) > 0.$$

In the above expressions,

$$\begin{aligned}
\widetilde{K}_{ij} &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} u^i v^j K(u, v)\, dudv; \ \ \widetilde{M}_{15}^* = \widetilde{K}_{20}^3 \widetilde{K}_{22} (\widetilde{K}_{22} - \widetilde{K}_{40}); \\
\widetilde{M}_{55}^* &= \widetilde{K}_{20}^2 \widetilde{K}_{22} (\widetilde{K}_{00} \widetilde{K}_{40} - \widetilde{K}_{20}^2); \ \ \widetilde{M}_{56}^* = \widetilde{K}_{20}^2 \widetilde{K}_{22} (\widetilde{K}_{20}^2 - \widetilde{K}_{00} \widetilde{K}_{22}); \\
\widetilde{\Delta} &= \widetilde{K}_{20} \widetilde{M}_{15}^* + \widetilde{K}_{40} \widetilde{M}_{55}^* + \widetilde{K}_{22} \widetilde{M}_{56}^*.
\end{aligned}$$

Proof: By expression (3), we have

$$\begin{aligned}
\widehat{b}(x,y) &= \frac{1}{K_{20}}\sum_{i=1}^{n}\sum_{j=1}^{n}(x_i-x)K\left(\frac{x_i-x}{h_n},\frac{y_j-y}{h_n}\right)Z_{ij}\\
\widehat{c}(x,y) &= \frac{1}{K_{02}}\sum_{i=1}^{n}\sum_{j=1}^{n}(y_j-y)K\left(\frac{x_i-x}{h_n},\frac{y_j-y}{h_n}\right)Z_{ij}\\
\widehat{e}(x,y) &= \frac{1}{\Delta}\sum_{i=1}^{n}\sum_{j=1}^{n}\{M_{15}^*+M_{55}^*(x_i-x)^2+M_{65}^*(y_j-y)^2\}K\left(\frac{x_i-x}{h_n},\frac{y_j-y}{h_n}\right)Z_{ij} \qquad (6)\\
\widehat{f}(x,y) &= \frac{1}{\Delta}\sum_{i=1}^{n}\sum_{j=1}^{n}\{M_{16}^*+M_{56}^*(x_i-x)^2+M_{66}^*(y_j-y)^2\}K\left(\frac{x_i-x}{h_n},\frac{y_j-y}{h_n}\right)Z_{ij}, \qquad (7)
\end{aligned}$$

where $M_{15}^*=M_{16}^*=K_{20}^3K_{22}(K_{22}-K_{40})$, $M_{55}^*=M_{66}^*=K_{20}^2K_{22}(K_{00}K_{40}-K_{20}^2)$, and $M_{56}^*=M_{65}^*=K_{20}^2K_{22}(K_{20}^2-K_{00}K_{22})$. For a given point $(x,y)\in\Omega$, if (x_τ,y_τ) is the closest point on Γ from (x,y) and their Euclidean distance is τh_n, where $0\le\tau<1$ a constant, then, by some similar arguments to those in the proof of Lemma A.1, we have

$$\begin{aligned}
E(\widehat{b}(x,y)) &= \frac{1}{\widetilde{K}_{02}}\frac{1}{n^2h_n^4}\left(\sum_{(x_i,y_j)\in Q_n^{(1)}(x,y)}+\sum_{(x_i,y_j)\in Q_n^{(2)}(x,y)}\right)(x_i-x)g(x_i,y_j)K\left(\frac{x_i-x}{h_n},\frac{y_j-y}{h_n}\right)\\
&\quad +o\left(\frac{1}{h_n}\right)\\
&= \frac{1}{\widetilde{K}_{02}h_n}g_-(x_\tau,y_\tau)\int\int_{Q^{(1)}}uK(u,v)\,dudv\\
&\quad +\frac{1}{\widetilde{K}_{02}h_n}g_+(x_\tau,y_\tau)\int\int_{Q^{(2)}}uK(u,v)\,dudv+o\left(\frac{1}{h_n}\right)\\
&= \frac{C_\tau}{\widetilde{K}_{02}h_n}\int\int_{Q^{(2)}}uK(u,v)\,dudv+o\left(\frac{1}{h_n}\right). \qquad (8)
\end{aligned}$$

Similar to (5), it can be shown that $\widehat{b}(x,y)-E(\widehat{b}(x,y))=O\left(\frac{\sqrt{\log(n)}}{nh_n^2}\right)=o\left(\frac{1}{h_n}\right)$, a.s. Then, result (i) in Lemma A.2 can be obtained by combining this result with equation (8). Result (ii) in Lemma A.2 can be obtained similarly.

By equations (6) and (7) and the relationships that M_{15}^*=M_{16}^*, M_{55}^*=M_{66}^*, and M_{56}^*=M_{65}^*, we have

$$\begin{aligned}
E(L(x,y)) &= \frac{1}{\Delta}\sum_{i=1}^{n}\sum_{j=1}^{n}\{2M_{15}^*+(M_{55}^*+M_{65}^*)((x_i-x)^2+(y_j-y)^2)\}g(x_i,y_j)K\left(\frac{x_i-x}{h_n},\frac{y_j-y}{h_n}\right)\\
&= \frac{C_\tau}{\Delta}\sum_{(x_i,y_j)\in Q_n^{(2)}(x,y)}\{2M_{15}^*+(M_{55}^*+M_{65}^*)((x_i-x)^2+(y_j-y)^2)\}K\left(\frac{x_i-x}{h_n},\frac{y_j-y}{h_n}\right)\\
&\quad +o\left(\frac{1}{h_n^2}\right).
\end{aligned}$$

It can be checked that

$$\widetilde{M}_{15}^{*}\widetilde{K}_{00} + \widetilde{M}_{55}^{*}\widetilde{K}_{20} + \widetilde{M}_{56}^{*}\widetilde{K}_{02} = 0 \tag{9}$$

After applying result (9) to the above expression of $E(L(x,y))$, when $\tau = 0$, we have

$$E(L(x,y)) = \frac{2C_\tau}{\widetilde{\Delta} h_n^2}\left(\widetilde{M}_{15}^{*}\widetilde{K}_{00} + \widetilde{M}_{55}^{*}\widetilde{K}_{20} + \widetilde{M}_{56}^{*}\widetilde{K}_{02}\right) + o\left(\frac{1}{h_n^2}\right) = o\left(\frac{1}{h_n^2}\right). \tag{10}$$

Similar to equation (5), it can be shown that

$$L(x,y) - E(L(x,y)) = O\left(\frac{\sqrt{\log(n)}}{nh_n^3}\right) = o\left(\frac{1}{h_n^2}\right),\ a.s. \tag{11}$$

Result (iii) in Lemma A.2 can be obtained by combining equations (10) and (11).

When $0 < \tau < 1$ and $(x,y) \in Q^{(1)}(x,y)$, we have

$$\begin{aligned} E(L(x,y)) &= \frac{2C_\tau}{\widetilde{\Delta} h_n^2}\left(2\widetilde{M}_{15}^{*}\int\int_{Q^{(2)}} K(u,v)\,dudv + (\widetilde{M}_{55}^{*} + \widetilde{M}_{56}^{*})\int\int_{Q^{(2)}} (u^2+v^2)K(u,v)\,dudv\right) \\ &\quad + o(\frac{1}{h_n^2}) \\ &= \frac{C_1}{h_n^2} + o\left(\frac{1}{h_n^2}\right) \end{aligned} \tag{12}$$

It can be checked that,

$$\frac{\int\int_{Q^{(2)}}(u^2+v^2)K(u,v)\,dudv}{2\int\int_{Q^{(2)}} K(u,v)\,dudv} > \widetilde{K}_{20} \tag{13}$$

and

$$\widetilde{M}_{15}^{*} = \widetilde{K}_{20}^{3}\widetilde{K}_{22}(\widetilde{K}_{22} - \widetilde{K}_{40}) < 0 \tag{14}$$

So, by equations (9), (13) and (14), the quantity C_1 defined in Lemma A.2 is positive. Then, result (iv) in Lemma A.2 can be obtained by combining this result with results (11) and (12). Result (v) in Lemma A.2 can be obtained similarly. □

From results (i) and (ii) of Lemma A.2, it can be concluded that, if the point (x,y) is close to a JLC, then the quantities $\widehat{b}(x,y)$ and $\widehat{c}(x,y)$ are large; thus, they contain jump information around (x,y). From results (iii)–(v), the estimated Laplacian $L(x,y)$ changes its sign from positive to negative when the underlying regression surface g jumps from a lower level to a higher level; thus, it has the zero-crossing properties around the JLC.

B Proof Of Theorem 3.1

Suppose that X and Y are two independent random variables, $X \sim N(0,1)$, and $Y \sim N(0,1)$. Then, it is easy to check that

$$\widehat{\sigma}_n^2 \chi_2^2(\delta_n) \stackrel{D}{=} \widehat{\sigma}_n^2 X^2 + \left(\widehat{\sigma}_n Y + \sqrt{(g_x')^2 + (g_y')^2}\right)^2,$$

where $\stackrel{D}{=}$ denotes equality in distribution. Using the fact that $\frac{Z_{1-\alpha_n}}{nh_n^2} = O\left(\frac{1}{h_n(\log(n))^\eta}\right)$, $\widehat{\sigma}_n = O(\frac{1}{nh_n^2})$, a.s., and $\sqrt{(g_x')^2 + (g_y')^2}$ is bounded in Ω_{h_n}, we have

$$\left\| T_n - \sqrt{(g_x')^2 + (g_y')^2} \right\|_{\Omega_{h_n}} = O\left(\frac{1}{h_n(\log(n))^\eta}\right), \text{ a.s.} \tag{15}$$

From Lemma A.1, we also have

$$\left\| M - \sqrt{(g_x')^2 + (g_y')^2} \right\|_{\Omega_{h_n}} = O(h_n^2) + O\left(\frac{\sqrt{\log(n)}}{nh_n^2}\right) = o\left(\frac{1}{h_n(\log(n))^\eta}\right), \text{ a.s.} \tag{16}$$

By equations (15) and (16), we can conclude that, when n is large enough, the following expression is true for any $\rho > 0$:

$$\widehat{D}_n \cap \Omega_\rho \subset J_n \cap \Omega_\rho, \text{ a.s.,} \tag{17}$$

where $J_n = H_n \backslash \Omega_{h_n}$ and $H_n = [h_n, 1-h_n] \times [h_n, 1-h_n]$.

For a point $(x,y) \in J_n$, if (x_τ, y_τ) is the closest point on Γ from (x,y) and their Euclidean distance is τh_n, where $0 \leq \tau < 1$ a constant, then from results (i) and (ii) in Lemma A.2, we have $M(x,y) = O(\frac{1}{h_n})$. After combining this result with equation (15), for large enough n, we have

$$\min_{(x,y) \in J_{n,\tau}} M(x,y) > T_n \tag{18}$$

where $J_{n,\tau} = \{(x,y) : (x,y) \in H_n, \sqrt{(x-x')^2 + (y-y')^2} \leq \tau h_n$, where (x',y') is any point on the JLC$\}$.

For any point $(x,y) \in D \cap \Omega_\rho$, we can find two points $(x_1,y_1) \in Q_n^{(1)}(x,y)$ and $(x_2,y_2) \in Q_n^{(2)}(x,y)$, each of which has an Euclidean distance of τh_n from (x,y). By result (iv) of Lemma A.2, we have

$$L(x_1,y_1) = \frac{C_1}{h_n^2} + o\left(\frac{1}{h_n^2}\right)$$

and

$$L(x_2,y_2) = -\frac{C_2}{h_n^2} + o\left(\frac{1}{h_n^2}\right),$$

where C_1 and C_2 are two positive constants defined in Lemma A.2.

When n is large enough, we have

$$L(x_1, y_1) > 0 \text{ and } L(x_2, y_2) < 0. \tag{19}$$

Now consider the circular neighborhood $P_{\tau,h_n}(x,y) = \{(x',y') : (x',y') \in H_n, \sqrt{(x-x')^2 + (y-y')^2} \leq \tau h_n\}$. By combining expressions (18) and (19), we have, when n is large enough,

$$P_{\tau,h_n}(x,y) \cap \widehat{D}_n \neq \phi. \tag{20}$$

By expressions (17) and (20), we have

$$d_H(\widehat{D}_n \cap \Omega_\rho, D \cap \Omega_\rho) \leq h_n, \text{ a.s. },$$

from which the conclusion of Theorem 3.1 can be obtained.