

# BOX SPLINE WAVELET FRAMES FOR IMAGE EDGE ANALYSIS

WEIHONG GUO\* AND MING-JUN LAI†

**Abstract.** We present a new box spline wavelet frame and apply it for image edge analysis. The wavelet frame is tight and constructed based on a box spline of eight directions which is seldom used for applications before. Due to the eight different directions, it can find edges of various types in detail quite well. In addition to step edges (local discontinuities in intensity), it is able to locate Dirac edges (momentary changes of intensity) and hidden edges (local discontinuity in intensity derivatives). The method is simple and robust to noise. Many numerical examples are presented to demonstrate the effectiveness of this method. Quantitative and qualitative comparison with other edge detection techniques are given to show the advantage of this wavelet frame. Our test images include synthetic ones with known ground truth and natural, medical images with rich geometric information.

**Key words.** Edge detection, box spline, wavelet frames, step edge, Dirac edge

**1. Introduction.** Edge detection, consistent with human perception is the first step for image interpretation and understanding. People are interested in identifying the constituent parts of an image when viewing it. Edges provide the topology and structure information of essential objects in an image. That is, edges present a skeleton of an image. These edge information can be used directly for feature extraction, object identification, region segmentation etc. It can also be used as a prior information to help improve other tasks such as image denoising, image restoration, image reconstruction, pattern recognition etc. In pattern recognition analysis for instance, it saves tremendous computational time to work on a 1-bit 0-1 edge map than on a 8-bit gray scale image. General edge detection task recovers step edges (local discontinuities in intensity), Dirac edges (momentary changes of intensity), fractal edges (which are caused by noise present in the image), as well as other edges such as hidden edges (discontinuous locations of some directional derivatives of images). Certainly, edge detectors produce edges with compromise among accuracy, completeness, and in some cases, smoothness. The goal of the research is to find the details of an image as accurate as possible.

In this paper, we present an edge detector based on box spline wavelet frames. We shall show that it is able to detect the features of an image excellently. Box splines are compactly supported piecewise polynomial functions. See [12] and [26] for detail in theory and see [22] for explicit representations of bivariate box splines on three and four direction meshes. Box splines are smooth refinable functions and hence are often used to construct various wavelet functions such as bi-orthogonal wavelets, pre-wavelets, tight wavelet frames in multivariate setting (cf. see [18, 23, 27] for explicit formulas of them with any regularity). Although it is known for several years that wavelet functions can be used for image edge detection, the performance and effectiveness of box spline wavelet frames for such application are not well understood. In particular, it is not known which box spline wavelet frame works best for edge extraction. Following the construction method in [27] and based on the work in [33], we have obtained box spline wavelet framelets based on various box splines  $B_{111}, B_{221}, B_{222}, B_{1111}, B_{2111}, B_{2211}$  on three or four direction mesh (cf. [12] or [26]) and  $B_8$  which is a box spline based on 8 different directions to be explained in section 3. Mainly we have experimented these wavelet frames as well as those based on the tensor product of univariate linear, quadratic, cubic and quadratic B-splines for edge detection. It turns out that  $B_8$ , the one based on eight different directions, is most effective in catching all details of images. Qualitative and quantitative comparison with some other types of edge detectors shows that the proposed box spline edge detector produces more accurate edges under similar completeness and smoothness. For the sake of fair comparison, we start with a synthetic image with ground truth edges, and apply four edge detectors under various parameter settings to this image. We then compare the best results of the four methods using measure Pratt's Figure of Merit (Figure 4.1) and compare results under all parameter settings using measure probability of detection (Figure 4.2). The results show that the proposed edge detector leads to higher Figure of Merit and consistently higher probability of detection. Visual qualitative comparison (Figures 4.3 - 4.11) implies that the proposed edge detector is able to catch fine details and is robust to noise. We explain the detail in section 3 and demonstrate results in section 4 with various applications.

---

\*Department of Mathematics, Case Western Reserve University, Cleveland, OH, 44106, USA. (wxg49@case.edu).

†Department of Mathematics, The University of Georgia, Athens, GA 30602, USA. (mj1ai@math.uga.edu)

The paper is organized as follows. We start with a review on existing edge detectors in section 2, and explain the construction of box spline wavelet frames, and image noise removal and edge detection based on the box spline frames in section 3. In particular, we explain a new box spline function based on a 8 direction mesh and use it to construct wavelet frames. In section 4, we compare the proposed box spline edge detector with several others, and demonstrate that the box spline wavelet frame always performs the best in capturing the details of images. In addition we present examples to find the hidden edges (the discontinuity of second order derivatives) of images in section 4.5. Finally, conclusion is drawn in section 5.

**2. Review on edge detectors.** In this section we review some existing edge detectors, some of which will be compared with the proposed edge detector.

**2.1. Partial derivative based edge detectors.** An important class of existing edge detectors is based on partial derivatives of the input image. Locations with maximal gradient or zero Laplacian are classified as edges [3, 13, 19, 28, 31, 20]. Those gradient based edge detectors typically include three steps. First, to remove noise if the input image is not clean enough; usually Gaussian convolution is applied. Second, to compute approximated partial derivatives which can be done by convolving with some kernels. Various kernels have been developed for this purpose with different accuracy along different directions [48, 46]. Prewit and Canny edge detectors for instance differ in specific forms of the kernels used to approximate partial derivatives. Third, to locate edges through thresholding the norm of gradients. The standard thresholding treats pixels with gradient magnitude greater than one threshold as edges. Hysteresis thresholding uses two threshold values. Any pixel above the upper threshold is characterized as edge, so are those pixels that are in the neighborhood of an edge pixel and with gradients higher than the lower threshold. Hysteresis thresholding leads to connected edges and is sometimes referred to as linking. Non-maximum suppression is sometimes used to thin edges in method like Canny.

Edge detection methods of this type are robust to low-level noise, but tend to mistakenly detect fake edges created by excessive noise/artifacts. To enhance the robustness to noise/artifacts, one can combine them with local mutual information [16].

**2.2. Continuous wavelet based and shearlet edge detectors.** Mallat’s wavelet based method [30] can not only detect the location of edges but can also characterize the types (step, Dirac, fractal, smooth) of edges through analyzing changes in *Lipschitz regularity* across different scales. We first review definition of Lipschitz regularity and wavelet transform, followed by presenting a theorem that helps relate the two of them.

**DEFINITION 2.1** (Lipschitz regularity). *Let  $0 \leq \alpha \leq 1$ . A function  $f(x)$  is uniformly Lipschitz  $\alpha$  over an interval  $(a, b)$  if there exists a constant  $K$  such that for any  $x_0, x_1 \in (a, b)$ ,  $|f(x_0) - f(x_1)| \leq K|x_0 - x_1|^\alpha$ .*

Lipschitz regularity is difficult to verify directly and is thus not practically useful. However, the following theorem relates local Lipschitz regularity with the dyadic wavelet transform in an explicit form. Let  $\theta(x)$  be a smoothing function that satisfies  $\int_{-\infty}^{\infty} \theta(x) dx = 1$ ,  $\lim_{x \rightarrow -\infty} \theta(x) = 0$ , and  $\lim_{x \rightarrow \infty} \theta(x) = 0$ . Let  $\omega(x) := \theta'(x)$ , which is a wavelet since  $\int_{-\infty}^{\infty} \omega(x) dx = 0$ . The continuous wavelet transform of  $f$  at scale  $s$  is defined as  $W_s f(x) := f * \omega_s(x)$ , where  $\omega_s(x) = \frac{\omega(x/s)}{s}$  is the scaled wavelet. For dyadic wavelet transforms, the scale  $s$  is chosen as  $s = 2^j, j = 1, 2, \dots$ .

**THEOREM 2.2.** [30, 10] *Let  $0 < \alpha < 1$ . A function  $f(x)$  is uniformly Lipschitz  $\alpha$  over  $(a, b)$  if and only if there exists a constant  $K > 0$  such that the wavelet transform satisfies  $|W_{2^j} f(x)| \leq K(2^j)^\alpha$ , for all  $x \in (a, b)$ , and  $j = 1, 2, \dots$ .*

According to this theorem, if  $\sup_{x \in (a, b)} |W_{2^j} f(x)|$  strictly decreases as  $j$  increases, then  $-1 \leq \alpha < 0$ , and there is an impulse (Dirac edge) at  $x$ ; if the maximum increases with scale, then  $0 < \alpha \leq 1$ , and the signal/image intensity changes gradually and thus is smooth at  $x$ . When the maximum does not change much across scales,  $\alpha = 0$ , and there is a step edge at  $x$ . These form a base for using continuous wavelet for edge detection.

However, continuous wavelet based edge detector [30] has difficulty in distinguishing close edges and has poor angular accuracy. It is due to the well known fact that wavelets are perfect in describing *isotropic* structures, but not so well dealing with *anisotropic* phenomena. A more recent work in this direction of research is to use shearlet [14, 21] for edge detection and analysis [45]. It claims to be effective in detecting the location, orientation of edges and the number of edges at each point. Shearlet transform decomposes an image with respect to scale, location and orientation. The orientation information at each scale is directly available.

Computationally, shearlet transform applies kernels with trapezoidal support along various directions to the high frequency component of the input image using multiple scales. The locations of the edges of the image are then characterized by observing the change of shearlet transform coefficients across scales. The orientations of edges are extracted from the directions along which the shearlet transform coefficients are significant while the number of edges at each point is determined by the number of peaks in absolute shearlet transform coefficients.

**2.3. Segmentation based edge detectors.** One can also extract edges from segmentation results. Image segmentation [32, 2, 4] partitions image domain into different subregions, each of which is homogeneous with respect to some characteristics such as intensity. The borders of those subregions form the edges. To give more mathematical details, we take 2-phase Mumford-Shah method as an example. Let  $g$  be the intensity function of an input image, it separates the image domain into two parts  $\Omega_1$  and  $\Omega_2$ , one inside the edge contour  $\Gamma$  and one outside, such that  $g$  can be approximated by functions  $f_1, f_2$  that are  $C^1$  smooth in  $\Omega_1, \Omega_2$  respectively. This is done by minimizing

$$E(f_1, f_2, \Gamma) = \frac{1}{2} \left[ \int_{\Omega_1} (g - f_1)^2 + \int_{\Omega_2} (g - f_2)^2 \right] + \alpha \left[ \int_{\Omega_1} |\nabla f_1|^2 + \int_{\Omega_2} |\nabla f_2|^2 \right] + \beta \cdot \text{Length}(\Gamma)$$

with respect to functions  $f_1, f_2$  and contour  $\Gamma$ . The implementation of Mumford-Shah model is made easy in Chan-Vese method [4] using level set [34]. More specifically, in the special case when  $f_1, f_2$  are constant values  $c_1, c_2$ , it represents the edge contour  $\Gamma$  by the zero level set of a Lipschitz function  $\Phi : \Omega \rightarrow \mathbb{R}$  and the regions inside and outside of the contour  $\Gamma$  by the regions with positive and negative  $\Phi$  values respectively. Let  $H(\cdot)$  be Heaviside function defined as  $H(z) = 1$  for positive  $z$  and 0 elsewhere. Then the  $\Phi$  minimizes

$$\frac{1}{2} \int_{\Omega} [H(\Phi)(g - c_1)^2 + (1 - H(\Phi))(g - c_2)^2] + \beta \int_{\Omega} |\nabla H(\Phi)|$$

gives the segmentation of the domain and the edge contours. Especially, edges are detected from 0 level set of  $\Phi$ .

Though segmentation results can be used to derive edges, it is not a common practice to extract edges from segmentation algorithms.

**2.4. Other methods.** Other edge detectors include Mumford-Shah green function [29], morphological gradient [40, 35, 37], fractal geometry [47, 43], high order and variable order total variation [41] based methods.

**3. Box spline tight wavelet frames.** Tight wavelet frames are generalizations of discrete orthonormal wavelets [38, 39, 7, 8, 9, 11, 33, 27, 25]. Box splines are refinable functions and one can easily choose various directions to have a box spline function with a desired order of smoothness. Naturally, they have been used to construct various wavelet functions including wavelet frames. An advantage of using a frame is that it consists of many redundant functions which can approximate various edges and features better than using only linearly independent functions. Hence, a tight wavelet frame based on box spline with more redundant directions is able to find more edges/features from an image than a frame based on a box spline with less redundant directions, e.g. tensor product of univariate B-spline wavelet frames. In this section, we first review box spline tight wavelet frames and then present our derivation of the new box spline wavelet frame based on  $B_8$ .

**3.1. Review on general box spline tight wavelet frames.** We begin with the definition of tight frames based on multi-resolution approximation of  $L_2(\mathbb{R}^2)$ . Given a function  $\psi \in L_2(\mathbb{R}^2)$ , we set

$$\psi_{j,k}(y) = 2^j \psi(2^j y - k).$$

Let  $\Psi$  be a finite subset of  $L_2(\mathbb{R}^2)$  and

$$\Lambda(\Psi) := \{\psi_{j,k}, \psi \in \Psi, j \in Z, k \in Z^2\}.$$

DEFINITION 3.1. We say that  $\Lambda(\Psi)$  is a frame if there exist two positive numbers  $A$  and  $B$  such that

$$A\|f\|_{L_2(\mathbb{R}^2)}^2 \leq \sum_{g \in \Lambda(\Psi)} |\langle f, g \rangle|^2 \leq B\|f\|_{L_2(\mathbb{R}^2)}^2$$

for all  $f \in L_2(\mathbb{R}^2)$ .  $\Lambda(\Psi)$  is a tight frame if it is a frame with  $A = B$ . In this case, after a renormalization of the  $g$ 's in  $\Psi$ , we have

$$\sum_{g \in \Lambda(\Psi)} |\langle f, g \rangle|^2 = \|f\|_{L_2(\mathbb{R}^2)}^2$$

for all  $f \in L_2(\mathbb{R}^2)$ .

It is known (cf. [10]) that when  $\Lambda(\Psi)$  is a tight frame, any  $f \in L_2(\mathbb{R}^2)$  can be represented by  $g \in \Lambda(\Psi)$ , i.e.

$$f = \sum_{g \in \Lambda(\Psi)} \langle f, g \rangle g, \quad \forall f \in L_2(\mathbb{R}^2).$$

Let  $\phi \in L_2(\mathbb{R}^2)$  be a compactly supported refinable function, i.e.,

$$\hat{\phi}(\omega) = P(\omega/2)\hat{\phi}(\omega/2)$$

where  $P(\omega)$  is a trigonometric polynomial in  $e^{i\omega}$ .  $P$  is often called the mask of refinable function  $\phi$ . We look for  $Q_i$  (trigonometric polynomial) such that

$$P(\omega)\overline{P(\omega + \ell)} + \sum_{i=0}^r Q_i(\omega) \overline{Q_i(\omega + \ell)} = \begin{cases} 1, & \text{if } \ell = 0, \\ 0, & \ell \in \{0, 1\}^2 \pi \setminus \{0\}. \end{cases} \quad (3.1)$$

The conditions (3.1) are called the *Unitary Extension Principle (UEP)* in [38], [39] and [11]. With these  $Q_i$ 's we can define wavelet frame generators or framelets  $\psi^{(i)}$  defined in terms of the Fourier transform by

$$\hat{\psi}^{(i)}(\omega) = Q_i(\omega/2)\hat{\phi}(\omega/2), \quad i = 1, \dots, r. \quad (3.2)$$

Then, if  $\phi$  belongs to Lip  $\alpha$  with  $\alpha > 0$ ,  $\Psi = \{\psi^{(i)}, i = 1, \dots, r\}$  generates a tight frame, i.e.,  $\Lambda(\Psi)$  is a tight wavelet frame (cf. [27]).

Furthermore, letting  $\mathcal{Q}$  be a rectangular matrix defined by

$$\mathcal{Q} = \begin{bmatrix} Q_1(\xi, \eta) & Q_1(\xi + \pi, \eta) & Q_1(\xi, \eta + \pi) & Q_1(\xi + \pi, \eta + \pi) \\ Q_2(\xi, \eta) & Q_2(\xi + \pi, \eta) & Q_2(\xi, \eta + \pi) & Q_2(\xi + \pi, \eta + \pi) \\ Q_3(\xi, \eta) & Q_3(\xi + \pi, \eta) & Q_3(\xi, \eta + \pi) & Q_3(\xi + \pi, \eta + \pi) \\ Q_4(\xi, \eta) & Q_4(\xi + \pi, \eta) & Q_4(\xi, \eta + \pi) & Q_4(\xi + \pi, \eta + \pi) \\ \vdots & \vdots & \vdots & \vdots \\ Q_r(\xi, \eta) & Q_r(\xi + \pi, \eta) & Q_r(\xi, \eta + \pi) & Q_r(\xi + \pi, \eta + \pi) \end{bmatrix}, \quad (3.3)$$

and  $\mathcal{P} = (P(\xi, \eta), P(\xi + \pi, \eta), P(\xi, \eta + \pi), P(\xi + \pi, \eta + \pi))^\top$ , (3.1) is simply

$$\mathcal{Q}^* \mathcal{Q} = I_{4 \times 4} - \overline{\mathcal{P}} \mathcal{P}^\top. \quad (3.4)$$

The construction of tight wavelet frames is to find  $\mathcal{Q}$  satisfying (3.4) which is an exact reconstruction condition. It was observed in [27] that  $\mathcal{Q}$  can be easily found if  $\mathcal{P}$  satisfies the QMF condition, i.e.,  $\mathcal{P}^T \mathcal{P} = 1$ . In this case,  $\mathcal{Q}$  has a very simple expression. However, the mask  $P$  of a refinable function  $\phi$  usually satisfies the following *sub-QMF condition*

$$\sum_{\ell \in \{0, 1\}^2 \pi} |P(\omega + \ell)|^2 \leq 1. \quad (3.5)$$

For example, bivariate box splines on three or four direction mesh or eight direction mesh to be explained below, the mask will satisfy (3.5). Although there is no Riesz-Féjer theorem in the multivariate setting, we are able to find additional Laurent polynomials  $\tilde{P}_i$  to have (3.6) for bivariate box spline functions (cf. [27]). Then we can use the following result (mainly its constructive proof in [27]) to find tight wavelet frame.

**THEOREM 3.2** (Lai and Stöckler, 2006). *Suppose that  $P$  satisfies the sub-QMF condition (3.5). Suppose that there exists Laurent polynomials  $\tilde{P}_1, \dots, \tilde{P}_N$  such that*

$$\sum_{m \in \{0,1\}^2} |P_m(\omega)|^2 + \sum_{i=1}^N |\tilde{P}_i(2\omega)|^2 = 1, \quad (3.6)$$

where  $P_m, m \in \{0,1\}^2$  are polyphases of  $P$ . Then there exist  $4+N$  compactly supported tight frame generators with wavelet masks  $Q_m, m = 1, \dots, 4+N$  such that  $P, Q_m, m = 1, \dots, 4+N$  satisfy (3.4).

We next recall the definition of bivariate box spline functions on direction set  $D$ . For example, writing  $e_1 = (1,0), e_2 = (0,1), e_3 = e_1 + e_2, e_4 = e_1 - e_2, e_5 = (2,1), e_6 = (2,-1), e_7 = (1,2), e_8 = (1,-2)$  to be direction vectors, let  $D$  be a set of these vectors with some repetitions. Such  $D$  is called a direction set. Then the box spline  $\phi_D$  associated with direction set  $D$  may be defined in terms of Fourier transform by

$$\hat{\phi}_D(\omega) = P_D\left(\frac{\omega}{2}\right)\hat{\phi}_D\left(\frac{\omega}{2}\right), \quad (3.7)$$

where  $P_D$  is

$$P_D(\omega) = \prod_{\xi \in D} \frac{1 + e^{-i\xi \cdot \omega}}{2}.$$

We refer the interested reader to [5] and [12] for many properties of box splines. For explicit polynomial representation of bivariate box splines, see [22] and [26] which is enable us to evaluate these box splines easily. Note that it is easy to show that the mask  $P_D$  satisfies (3.5). To construct the associated tight framelets, we mainly find additional Laurent polynomials to satisfy (3.6). However, finding these additional Laurent polynomials may not be an easy task as there is no existence theory nor a constructive procedure except for box splines on three and four direction mesh.

**3.2. The eight direction box spline.** When using a box spline wavelet frame, we have the flexibility to choose a direction set. We can choose a box spline function with as many directions as possible to increase the redundancy. However, the more directions, the smoother the box spline function is, the longer the length of the low-pass and high-pass filters is, and hence, the more difficult to find tiny detail of the image. Empirical results show that the wavelet frame based on box spline with eight directions is the ideal one for edge/feature/detail detection. In this paper we shall present a tight framelet based on box spline  $\phi_8 := \phi_{D_8}$  with

$$D_8 = \{e_1, e_2, e_1 + e_2, e_1 - e_2, 2e_1 + e_2, 2e_1 - e_2, e_1 + 2e_2, e_1 - 2e_2\}. \quad (3.8)$$

Since  $D_8$  contains eight directions, we call the  $\phi_8$  eight direction box spline. The eight direction box spline  $\phi_8$  is new and has not been studied in the literature. Let us explain it a little bit more. It is a bivariate spline function of total degree  $\leq 6$  and is in  $C^5$ , that is, it is fifth continuously differentiable. Also  $\phi_8$  is compactly supported and is nonnegative. All integer translates of  $\phi_8$  are linearly dependent. Thus they are redundant. But they form a partition of unity after a scale. The mask  $P_8 := P_{D_8}$  can be found easily and is

$$P_8(\xi, \eta) = \left(\frac{1 + e^{i\xi}}{2}\right) \left(\frac{1 + e^{i\eta}}{2}\right) \left(\frac{1 + e^{i(\xi+\eta)}}{2}\right) \left(\frac{1 + e^{i(\xi-\eta)}}{2}\right) \\ \left(\frac{1 + e^{i(\xi+2\eta)}}{2}\right) \left(\frac{1 + e^{i(2\xi+\eta)}}{2}\right) \left(\frac{1 + e^{i(\xi-2\eta)}}{2}\right) \left(\frac{1 + e^{i(2\xi-\eta)}}{2}\right). \quad (3.9)$$

For convenience, we shall write  $P_8(\xi, \eta) = \sum_{j,k} p_{j,k} e^{-ij\xi} e^{-ik\eta}$ . To use Theorem 3.2, we need to solve the polynomial equation in (3.6) which is generally not guaranteed to have a solution. However, for box splines

on three and four direction mesh, it is known that the polynomial equation (3.6) has a solution (cf. [27] and [24]). Along this general direction, one result is worth mentioning. In [15], (3.6) has a solution if the inequality in (3.5) is an strict inequality. For box spline  $\phi_8$ , we solve (3.6) by using a brute force method. Indeed, we do not know how many extra Laurent polynomials  $\tilde{P}_j$  we need at the first place. That is, we do not know  $N$  in (3.6). Also we do not know what the degrees of these extra Laurent polynomials  $\tilde{P}_j$  are although one intuition is that the degrees should be less than or equal to 6, the degree of the 8 direction box spline  $\phi_8$ . We started with one extra Laurent polynomial and (3.6) leads to a system of several multivariate quadratic equations. We use Maple to set up these equations. However, the system has no solution. We then try two extra polynomials, three extra polynomials, until ten extra Laurent polynomials which seem to have a good chance to get the system solved. The system of many multivariate quadratic equations with more than 50 variables arises from these ten extra polynomials by Maple. These are quadratic equations and hard to solve. We use Maple to simplify these multivariate quadratic equations and then use our intuition to decouple these equations and break the system into smaller subsystems to solve. Detail is not shown here. Eventually we are able to solve the system within a very small tolerance (e.g.  $10^{-10}$ ) and find ten Laurent polynomials  $\tilde{P}_j, j = 1, \dots, 10$  to have (3.6), i.e.

$$1 - \sum_{\nu \in \{0, \pi\}^2} |P_8(\omega + \nu)|^2 \approx \sum_{j=1}^{10} |\tilde{P}_j(2\omega, 2\eta)|^2, \quad \omega = (\xi, \eta) \in [0, 2\pi]^2,$$

where these ten trigonometric polynomials are given below:

$$\begin{aligned} \tilde{P}_1(\xi, \eta) &= \frac{42}{14561} - \frac{542}{4269} e^{i(\xi+4\eta)} + \frac{42}{14561} e^{2i\xi} + \frac{191}{1576} e^{i\xi}, \\ \tilde{P}_2(\xi, \eta) &= \frac{281}{1476} - \frac{51}{26513} e^{i(4\xi+\eta)} + \frac{281}{1476} e^{2i\eta} - \frac{605}{1597} e^{i(\xi+\eta)}, \\ \tilde{P}_3(\xi, \eta) &= \frac{1}{192} - \frac{9}{32} e^{i(2\xi+3\eta)} + \frac{1}{192} e^{4i\xi} + \frac{283}{1152} e^{2i(\xi+\eta)} + \frac{29}{1152} e^{2i\xi}, \\ \tilde{P}_4(\xi, \eta) &= \frac{192}{15731} - \frac{233}{19415} e^{i(3\xi+2\eta)} + \frac{192}{1573} e^{4i\eta} - \frac{172}{741} e^{2i\eta}, \\ \tilde{P}_5(\xi, \eta) &= \frac{139}{2849} - \frac{278}{2849} e^{i(\xi+3\eta)} + \frac{139}{2849} e^{2i\xi}, \\ \tilde{P}_6(\xi, \eta) &= \frac{76}{4195} - \frac{843}{3208} e^{i(3\xi+\eta)} + \frac{76}{4195} e^{2i\eta} + \frac{227}{1002} e^{i(\xi+\eta)}, \\ \tilde{P}_7(\xi, \eta) &= \frac{412}{2807} - \frac{211}{1364} e^{i(\xi+2\eta)} + \frac{412}{2807} e^{2i\xi} - \frac{263}{3788} e^{2i(\xi+\eta)} - \frac{263}{3788} e^{2i\eta}, \\ \tilde{P}_8(\xi, \eta) &= \frac{152}{2475} - \frac{288}{779} e^{2i(\xi+\eta)} + \frac{152}{2475} e^{2i\eta} + \frac{494}{2001} e^{i(\xi+\eta)}, \\ \tilde{P}_9(\xi, \eta) &= \frac{19}{15834} - \frac{100}{983} e^{3i(\xi+\eta)} - \frac{19}{15834} e^{3i\xi} + \frac{100}{983} e^{3i\eta}, \\ \tilde{P}_{10}(\xi, \eta) &= \frac{230}{10131} - \frac{230}{10131} e^{3i\xi}. \end{aligned}$$

By Theorem 3.2, we will have 14 tight wavelet frame generators using the constructive steps in [27]. These 14 tight frames  $\psi^\ell$  in terms of Fourier transform can be expressed by

$$\widehat{\psi}_8^\ell(\xi, \eta) = Q^\ell(\xi/2, \eta/2) \widehat{\phi}_8(\xi/2, \eta/2), \quad \ell = 1, \dots, 14, \quad (3.10)$$

where  $Q^\ell(\xi, \eta) = \sum_j \sum_k q_{jk}^\ell e^{-ij\xi} e^{-ik\eta}$ . Due to the limited space in this article, we list the low pass filter and one of the high pass filters  $q_{jk}^1$  in the Appendix. Others are available upon request. With these filters, we are able to describe how to use them to find edges of images.

**3.3. Image decomposition and reconstruction.** Our box spline frames based edge detector involves three steps: 1) image decomposition based on the tight wavelet framelets; 2) zero out low pass parts and threshold high pass parts; and 3) reconstruction of edge map based on tight wavelet framelets. Specifically,

we first use the box spline wavelet frame to decompose an image into many levels of subimages which consist of a low-pass part and several high-pass parts of the image; secondly, we set the low-pass part to be zero and thresholding high-pass parts, and thirdly, we use the resulting high-pass parts to get the edge map. The motivation is that edges of an image are represented in high frequencies, while the smoothing parts of an image are represented in terms of translates and dilations of the refinable box spline function. Thus, we set the coefficients of the low-pass part in terms of translates and dilations of the box spline to be zero. Meanwhile, since noise is also represented in high frequencies, we apply a percentage thresholding technique (see Algorithm 1) to remove noise.

As explained above, using box spline wavelet frame for edge detection is easy. All we need to do is to explain how to do image decomposition and reconstruction. For convenience, we use the 14 tight wavelet frame functions:  $\{\psi^1, \dots, \psi^{14}\}$  constructed in the previous section based on the box spline function  $\phi_8$  to illustrate the decomposition and reconstruction. For an image  $f$  with finite energy, i.e.  $f \in L^2(\mathbb{R}^2)$ , let  $a_{j,\mathbf{k}}$  be the value of the inner product of  $f$  with the  $\phi_{j,\mathbf{k}}(\cdot) := 2^{2j}\phi_8(2^j \cdot -\mathbf{k})$  for  $\mathbf{k} \in \mathbb{Z}^2$  and  $j \in \mathbb{Z}$ . Note that since  $\phi_8$  is compactly supported and  $\int \phi(x)dx = 1$ ,  $a_{j,\mathbf{k}}$  can approximate the grayscale value at  $\mathbf{k}$  when  $j$  is large enough. Similarly, let  $b_{j,\mathbf{k}}^\ell$  be the value of the inner product of  $f$  with box spline wavelet framelets  $\psi_{j,\mathbf{k}}^\ell(\cdot) := \psi^\ell(2^j \cdot -\mathbf{k})$ 's for all  $j \in \mathbb{Z}, \mathbf{k} \in \mathbb{Z}^2$  and  $\ell = 1, \dots, 14$ .

Recall we have

$$\phi_{j,\mathbf{m}}(\cdot) = \sum_{\mathbf{k} \in \mathbb{Z}^2} p_{\mathbf{k}-2\mathbf{m}} \phi_{j+1,\mathbf{k}}(\cdot) \quad \text{and} \quad \psi_{j,\mathbf{m}}^\ell(\cdot) = \sum_{\mathbf{k} \in \mathbb{Z}^2} q_{\mathbf{k}-2\mathbf{m}}^\ell \phi_{j+1,\mathbf{k}}(\cdot)$$

by using the refinable property (3.9) and wavelet frame construction (3.10) for all integers. Note that the refinable function  $\phi$  and tight wavelet frames  $\psi^1, \dots, \psi^N$  are locally supported, the coefficients  $\{p_{\mathbf{k}-2\mathbf{m}}\}$  and  $\{q_{\mathbf{k}-2\mathbf{m}}^\ell\}$  are finite sequences for all  $\ell = 1, \dots, 14$ . Here  $p_{\mathbf{k}-2\mathbf{m}}$ 's are associated with the coefficients of  $P_8$  and  $q_{\mathbf{k}-2\mathbf{m}}^\ell$  are the coefficients of  $Q_\ell$  as explained in the Appendix. Thus we obtain the following tight wavelet frame decomposition algorithm by taking the inner products on both sides of the above two equations with image function  $f$ :

$$a_{j,\mathbf{m}} = \sum_{\mathbf{k} \in \mathbb{Z}^2} p_{\mathbf{k}-2\mathbf{m}} a_{j+1,\mathbf{k}} \quad \text{and} \quad b_{j,\mathbf{m}}^\ell = \sum_{\mathbf{k} \in \mathbb{Z}^2} q_{\mathbf{k}-2\mathbf{m}}^\ell a_{j+1,\mathbf{k}} \quad (3.11)$$

for all  $j \in \mathbb{Z}, \mathbf{m} \in \mathbb{Z}^2$  and  $\ell = 1, \dots, 14$ . Let  $X_j$  be the matrix associated with the  $(j)^{th}$  level image containing  $a_{j,k} = \langle f, \phi_{j,k} \rangle, \forall k \in \mathbb{Z}_+^2$  with  $|k| \leq N$  (e.g.,  $N = 512$ ) for all  $j \in \mathbb{Z}$ . Suppose that  $j$  is an integer large enough such that the given image  $f$  is approximately  $X_{j+1}$ , i.e. the pixel value  $f_k \approx a_{j+1,k}$ . Then the image decomposition procedure is to compute 2D convolution of  $P$  and each  $Q_\ell, \ell = 1, \dots, 14$  with the matrix  $X_{j+1}$ , i.e., find

$$P * X_{j+1}, \quad Q_\ell * X_{j+1}, \ell = 1, \dots, 14,$$

where  $*$  stands for 2D convolution and then downsample by deleting all the odd number of rows and columns of resulted matrices to obtain matrices  $X_j$  and  $Y_{j,\ell}$  for  $\ell = 1, \dots, 14$  due to the indices in (3.11).

Next let us describe the reconstruction procedure. Due to the exact reconstruction, we have

$$\phi_{j+1,\mathbf{m}}(\cdot) = \sum_{\mathbf{k} \in \mathbb{Z}^2} \{p_{\mathbf{m}-2\mathbf{k}} \phi_{j,\mathbf{k}}(\cdot) + \sum_{\ell=1}^8 q_{\mathbf{m}-2\mathbf{k}}^\ell \psi_{j,\mathbf{k}}^\ell(\cdot)\} \quad (3.12)$$

by using the perfection condition (3.4). By taking inner products on both sides of the above equation we have the tight wavelet frame reconstruction algorithm

$$a_{j+1,\mathbf{m}} = \sum_{\mathbf{k} \in \mathbb{Z}^2} \{p_{\mathbf{m}-2\mathbf{k}} a_{j,\mathbf{k}} + \sum_{\ell=1}^N q_{\mathbf{m}-2\mathbf{k}}^\ell a_{j,\mathbf{k}}^\ell\}. \quad (3.13)$$

Again these computation can be done by upsampling the image  $X_j, Y_{j,\ell}$  by 2, i.e., adding zero columns in between columns of  $X_j$  and  $Y_{j,\ell}, \ell = 1, \dots, 14$  and then add zero rows in between rows of the resulting matrices and then convolve with  $P$  and  $Q_\ell, \ell = 1, \dots, 14$ .

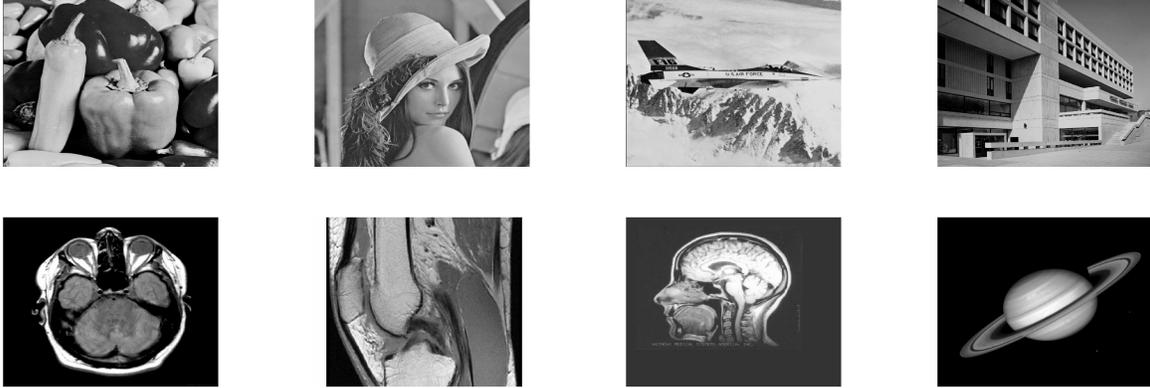


FIG. 3.1. *Eight testing images.*

Usually,  $X_j$ 's are called low-pass part of the image  $X_{j+1}$  and  $Y_{j,i}, i = 1, \dots, \ell$  are high-pass parts of the image since the low-pass sub-image has low-pass (smooth) parts of the image and the high-pass sub-images have high frequency (detail) parts of the image  $X_{j+1}$ . If necessary, we can repeat the decomposition process several times by convoluting low-pass filter  $P$  and high-pass filters  $Q_\ell$ 's with the low-pass image  $X_j$  to get  $X_{j-1}$  and  $Y_{j-1,\ell}, \ell = 1, \dots, 14$ . In order to detect a fractal edge, we need to apply such a decomposition two or more times.

**3.4. Noise removal before edge detection.** Images are usually contaminated by noise. It is sometimes necessary to remove noise from images before computing the edges. A classic method for image denoising is the wavelet shrinkage method which consists of using a hard or soft thresholding algorithm to trim the wavelet coefficients. In the following, we propose another approach based on the so-called weak orthogonal greedy algorithm (cf. [42]) to further remove the noise.

The main idea is to look for a sparse representation of a noisy image in a redundant wavelet frame system. We emphasize that a box spline wavelet frame has one significant advantage: the efficiency in computing inner products of an image with wavelet framelet functions as well as their integer translations and dilations as the decomposition and reconstruction explained in the previous subsection. Let  $\Phi = [\phi_1, \dots, \phi_n]$  be a wavelet frame matrix consisting of the values of framelet functions over pixel locations, i.e.  $\phi_1, \dots, \phi_n$  consist of refinable functions  $\phi(\cdot + (j, k)), \psi^{(\ell)}(j, k), \ell = 1, \dots, 14, (j, k) \in [1, N] \times [1, N]$  for an integer  $N > 1$  as described in the previous section. Let  $f$  be the image,  $G_k(f)$  be the  $k^{\text{th}}$  approximation of  $f$ , and  $R_k(f)$  be the residual of the  $k^{\text{th}}$  iteration.

ALGORITHM 1 (Percentage Thresholding Algorithm). *We begin with  $\Lambda_0 = \emptyset, R_0(f) = f, G_0(f) = 0$ . Choose a thresholding sequence  $\{t_1, t_2, \dots\}$  with all  $t_k \in (0, 1]$ .*

- *Step 1. For  $k \geq 1$ , find  $M_k = \max_{i \notin \Lambda_{k-1}} |\langle R_{k-1}(f), \phi_i \rangle|$ ;*
- *Step 2. Let  $\Lambda_k = \Lambda_{k-1} \cup \{i, |\langle R_{k-1}(f), \phi_i \rangle| \geq t_k M_k\}$ ;*
- *Step 3. Let  $L_{\Lambda_k}(f)$  the the best approximation (least squares approximation) of  $R_{k-1}(f)$  in subspace  $S_{\Lambda_k} = \text{span}\{\phi_i, i \in \Lambda_k\}$ .*
- *Step 4. Update  $G_k = G_{k-1}(f) + L_{\Lambda_k}(f)$  and  $R_k(f) = R_{k-1}(f) - L_{\Lambda_k}(f)$ .*
- *Step 5. If  $\|L_{\Lambda_k}(f)\|$  is small enough, we stop the algorithm. Otherwise we advance  $k$  to  $k + 1$  and go to Step 1.*

This algorithm differs from the weak orthogonal greedy algorithm by choosing more than one component per iteration. The name for this algorithm will be explained later. In this paper, we report some numerical results using the wavelet frame based on tensor product of symmetric quartic B-splines constructed in [6].

In our experiment, we first use the classic method of hard thresholding to remove some noise from a noisy image based on wavelet frame decomposition and reconstruction. Although we can use multi-levels, for simplicity, we choose to have one level decomposition and reconstruction to demonstrate our algorithm. That is, we decompose an image into one low-pass part with several high-pass parts and apply the hard thresholding technique to remove noise from high-pass parts and then reconstruct from the low-pass part

and resulting high-pass parts to yield a denoised image.

Next we use the denoised image as  $f$  and the associated tight wavelet frame to form a wavelet frame matrix  $\Phi$  and then apply Algorithm 1 for further noise removal. In this experiment, we use a thresholding sequence  $\mathbf{t} = \{t_0, t_1, t_2, t_3, \dots\}$  with  $t_i = rt_0$  with e.g.  $r = 0.78$  and  $t_0 = 0.9$  for  $i = 1, 2, \dots, 5$ . That is, we do five iterations in Algorithm 1. Note that a wavelet frame provides an efficiency in computing the inner products of the denoised image  $f$  with column vectors in  $\Phi$ . Since  $\phi_i, i = 1, \dots, n$  are just various wavelet framelets and their integer translates, the inner products  $\langle R_k, \phi_i \rangle$  are just convolution of  $\phi_i$  with the image or the  $(k - 1)^{th}$  residual. In each iteration of Algorithm 1, we use  $t_k$  to form a thresholding  $\epsilon_j = t_k M_j$  with  $M_j$  being the largest inner product in absolute value in the  $j^{th}$  kind of framelets for  $j = 1, \dots, 14$ . Choosing those inner products which are larger than  $\epsilon_j$  is equivalent to using the hard thresholding in percentage. This is the reason we call this algorithm the percentage thresholding algorithm. Computing the best approximation in Algorithm 1 is the same as reconstructing image using the resulting wavelet framelet coefficients. In short, using Algorithm 1 for image denoising is similar to the classic image denoising using wavelet framelets with this percentage thresholding technique.

The performance of Algorithm 1 is tested on the eight images shown in Figure 3.1. All images have intensity range  $[0, 255]$  and are added white Gaussian noise with  $\sigma = 20$ . We use the eight images in Figure 3.1 and apply the hard-thresholding method based on the wavelet frame mentioned above to denoise and find the best denoised image in terms of the standard Peak-Signal-to-Noise Ratio (PSNR). Then we use Algorithm 1 to further remove noise. In the following table, we report the PSNR before and after using Algorithm 1.

Table 1. PSNR before and after applying Algorithm 1

	Peppers	Lena512	F16	Bank
Before	30.25	30.66	30.38	29.47
After	30.37	31.14	30.53	29.53
NZC	15.21%	5.43%	21.67%	27.88%
	Brain	Knee	MRI	Saturn
Before	32.08	30.12	32.48	35.28
After	32.50	30.23	32.80	35.65
NZC	7.62%	23.73%	4.99%	0.523%

Certainly, when using a multi-level decomposition and reconstruction of wavelet frames, one may get slightly better PSNR values than those in Table 1. We leave them to the interested reader. An advantage of using this approach for image denoising is that one only needs about 25% or less nonzero coefficients of a tight wavelet frame to represent a denoised image. In Table 1, we list the percentage of nonzero coefficients (NZC) in a wavelet frame representation for each image. Here the NZC is a percentage of the number of nonzero coefficients over the size of images after denoising.

**3.5. Image edge detection.** The classic image edge detection is to set the low-pass part subimage into zero and reconstruct from the zero low-pass part together with high-pass parts. What is new in this paper is that we reconstruct the edges of an image based on the zero low-pass part and the high-pass parts after the percentage thresholding technique. That is, for each high-pass part, we use only the largest, say 50% of the all coefficients. Here is the outline of image edge detection procedure.

ALGORITHM 2 (Box Spline Edge Detector). *We apply the following steps for an input image  $f$ :*

- *Remove noise to get a cleaner image  $\tilde{f}$  by using Algorithm 1.*
- *Apply the tight wavelet frame to decompose the image  $\tilde{f}$  into one low-pass and various high-pass subimages.*
- *Set the low-pass part to be zero matrix and choose the lower 50% of the high-pass parts.*
- *Reconstruct image  $\hat{f}$  from zero low pass and thresholded high pass components.*
- *Using  $k$ -means classification with two classes to automatically classify  $\hat{f}$  into two categories: either 0 or 1.*
- *Clean up by removing all isolated edge, i.e., a location with value 0 which is surrounded by 1 or a location with value 1 surround by 0.*

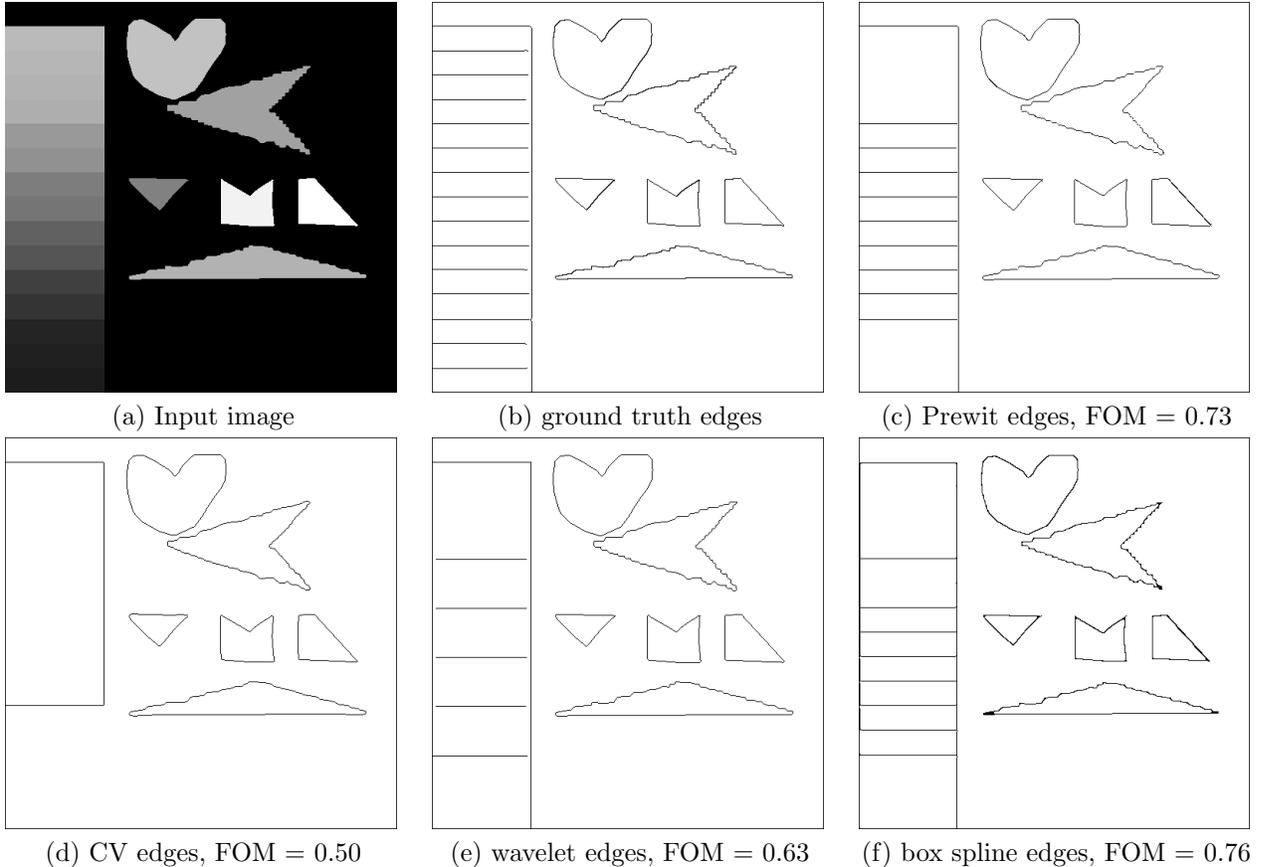


FIG. 4.1. Comparison of four edge detectors on a synthetic image.

The only parameter in the proposed Algorithm is the 50%. We understand that fine tuning of this threshold might lead to better results, but to make it simple, we just use 50%, and it turns out the numerical results are consistently satisfying already.

Note noise exists in both high frequency and low frequency parts. Though cutting off portion of the high frequency part as we do in the third step of the above edge detection algorithm does remove some high frequency noise, there is still some high frequency noise left. To avoid noise and also faked edges in edge results due to noise, we apply box spline wavelet frames based denoising to more sophisticatedly remove noise in the first step before edge detection.

**4. Numerical experiments.** This section consists of three sets of experimental results: one is a comparison of box spline edge detector with some selected edge detectors: Prewit, Canny, Chan-Vese, wavelet [30] and shearlet [45]; one is to test the scale invariance of the proposed method; one is the numerical results of edges/features/details from noisy images by using the proposed edge detector, and the other is the application of box spline based edge detection to object identification. Note, the wavelet [30] and shearlet [45] methods detect edges by analyzing the change of transform coefficients across different scales. These methods are more sophisticated than inserting kernels in discrete wavelet and shearlet into the proposed edge detection methods. We have tested discrete wavelets such as Haar, D4, D6 and biorthogonal 9/7 in the proposed algorithm to compute edges. The performance is not so good and we thus did not show them in this paper. We have also inserted discrete shearlet kernels from fast finite shearlet transform (FFST) toolbox [17] into the proposed edge detection, but the results are not as good as that in [45].

**4.1. Comparison of several edge detection methods.** We demonstrate the advantage of the proposed box spline edge detection method through comparing with some methods mentioned in Section 2. While there has been some attempts on proposing quantitative measures for the performance of an edge

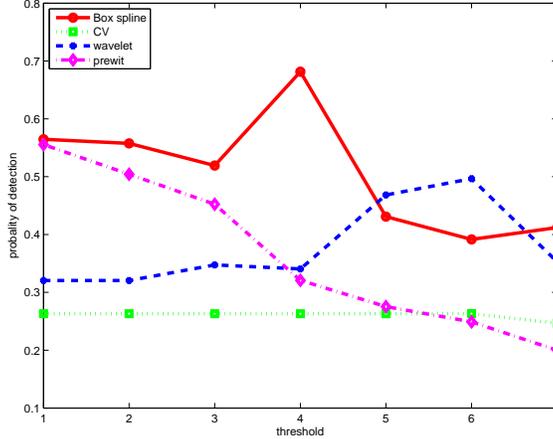


FIG. 4.2. Compare probability of detection of the four edge detectors on various parameters. The test image is the one shown in Figure 4.1(a).

detector, there seems to be no consensus on which one is the best. We select two popular ones: Pratt’s figure of merit ( $FOM$ ) [36] and probability of detection ( $P_D$ ) [1], the probability for an edge detector to find true edges. Computation of both of these measures require ground truth edges. Thus, we create one image with ground truth edges by hand as shown in Figure 4.1.

Pratt’s figure of merit is basically a measure describing the distance from the detected edges to the ground truth edges. The larger  $FOM$  the better. It is related to mis-detection of true edges, detection of false edges and is defined as

$$FOM = \frac{1}{\max(n_d, n_g)} \sum_{k=1}^{n_g} \frac{1}{1 + \alpha d(k)}$$

where  $n_d, n_g$  are number points on detected edges and ground truth edges respectively,  $d(k)$  is the distance from the  $k$ th detected edge point to the actual ground truth edges.  $\alpha$  is a scaling constant that we set as  $1/9$ .

Casting edge detection as a hypothesis testing problem to determine if an image point is on edge or not, the edge detection process can be described by the probability of correct edge detection  $P_D$  and probability of false edge detection  $P_F$ . Probability of detection describes the ability of an edge detection method to locate actual edges. Let  $M_d, M_g$  be the detected and ground truth binary edge maps with 1,0 intensity implying edge, and no edge. Then  $P_D$  is defined as

$$P_D = Prob(M_d = 1 | M_g = 1) = \frac{Prob(M_d = 1, M_g = 1)}{Prob(M_g = 1)}.$$

To avoid the intervention of denoising process to the performance evaluation of edge detection methods, we start with a simulated clean image (shown in Figure 4.1 (a)) with ground truth edges (Figure 4.1 (b)). It contains objects of various shapes and rectangular boxes with gradually changing intensities. We apply Prewit, Chan-Vese, wavelet and box spline edge detection methods to (a) under each method’s best parameter setting and obtain results as shown in (c)-(f). It can be seen that box spline result has the highest  $FOM$  and is thus closest to the ground truth edges shown in (b).

The comparison in Figure 4.1 is done on edges detected under each edge detection method’s best possible parameter selection. Note that more horizontal lines on the left can be detected by lowering the threshold value, but due to the fact that those areas have low intensities, and low contrast, one has to set the threshold value to be extremely low to detect those horizontal lines. As a result, we pay the price of having broken and/or non-smooth edges almost everywhere at the geometries on the right hand side. We thus presented the results that have the best overall appearance. In practice and in general, however one might not be able



FIG. 4.3. Comparison of five edge detectors on lena image.

to select the best parameter. In Figure 4.2, we compare  $P_D$  of the four edge detection methods under seven different parameter setting. One can see that box spline edge detection method has significantly higher  $P_D$  than the other four methods except for two out of seven of the wavelet edge detection results.

Next, we compare those edge detectors on natural images with more details and show in Figure 4.3 to Figure 4.8. No ground truth edges are available for quantitative comparison, but visual comparison shows that box spline method performs better in catching fine edges. See for instance the edges of Lena’s hair and eyes in Figure 4.4 and Barbara image’s table cloth, pants and the face as well as the edges of the textures and plants in Figure 4.8.

From computation time perspective, the proposed algorithm is indeed slower than Canny and Prewitt algorithms. Canny and Prewitt uses two convolution (x-direction and y-direction) while our algorithm uses 15 convolutions with one low-pass and 14 high-pass box spline filters. And the sizes of our filters are bigger. The current non-optimized implementation of the proposed algorithm is about 8 times as expensive as Canny and Prewitt algorithms on average. It is about 8 times and 5 times as expensive as wavelet and shearlet based method respectively. However, it is comparable with that of Chan-Vese.

**4.2. Scale invariance.** We have also tested the scale invariance of the proposed edge detector, i.e., the invariance of the edge detection results when the object of interest is observed at different scales. The test image is Lena. Suppose we are interested in edge detection of Lena’s hat decoration, we simulate views of the hat decoration at three different scales as shown in the top row of Fig. 4.9. The proposed edge detector is applied to get edges as shown in the bottom row of 4.9. One can observe that the edges of the hat decoration part are similar across scales. Therefore, the proposed edge detector has the potential of being scale invariant.

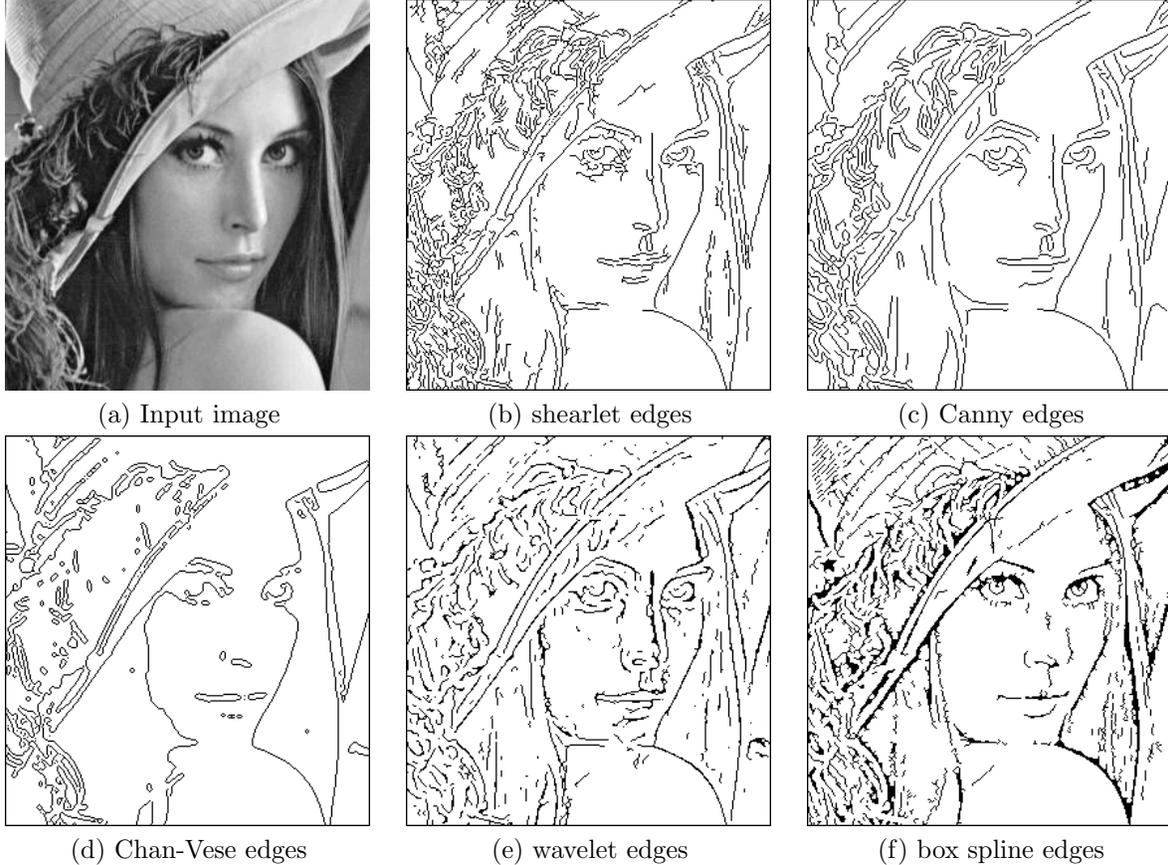


FIG. 4.4. Zoom in comparison of five edge detectors on the Lena image shown in Figure 4.3.

**4.3. Edge detection for noisy images.** Next we show the robustness of the proposed edge detector to noise by exhibiting edges detected from noisy images in Figure 4.10. We first add Gaussian noise with mean zero and variant  $\sigma^2 = 20$  to each image. The PSNR is around 22. Then we apply a standard wavelet frame denoising method (hard thresholding) to obtain an denoised image which is fed into Algorithm 1 and then apply Algorithm 2. Numerical results are shown in Figures 4.10 and 4.11.

It is clear that the edges/features/details of images do not loss much. The faces of Lena and Barbara are still clearly shown. The letters on the body of airplane F16 can be seen without any mistakes. This demonstrates the effectiveness of box spline wavelet frames for edge detection.

**4.4. Dirac edges.** It is well-known that the gradient based edge detectors such as Canny and Prewit fail to detect Dirac edges, locations with momentary intensity changes, accurately. They mistakenly treat locations to the left and the right side of Dirac edges as discontinuity in intensity, and thus detect double step edges instead of Dirac edges. Our box spline wavelet frame detector can find such Dirac edges exactly. Here is an example. The left panel of Figure 4.12 is a testing image. Our spline edge detector finds the exact edges appear exactly the same as the left panel of Figure 4.12.

**4.5. Application to hidden edges.** We now apply the box spline wavelet frame edge detector to find hidden edges, i.e., locations with some kind of discontinuous derivatives of image intensity. Detecting such edges has an important application for aircraft surface manufacture. As to reduce the turbulence, the surface of the body of an aircraft needs to be  $C^2$  smooth to avoid generating singularities of airflows. One way to find defects in the surface of an aircraft body is to find locations with discontinuous first or second order derivatives, i.e., hidden edges. In this subsection, we artificially create a surface with discontinuous



FIG. 4.5. Comparison of five edge detectors on Barbara image.

first order derivatives. For example,

$$z_1(x, y) = \begin{cases} (x-1)^2 + (y-1)^2 - 0.5, & \text{if } (x-1)^2 + (y-1)^2 > 0.5 \\ 0 & \text{otherwise} \end{cases}$$

over  $(x, y) \in [0, 2.55]^2$ . The 3D graph and the 2D intensity image of the function  $z_1(x, y)$  are shown in the left and the right panel of Figure 4.13 respectively.

From the 2D intensity image, we hardly see any edges. However, from the 3D surface, we can easily see the places where the first order derivatives are discontinuous. For another example, let

$$z_2(x, y) = \begin{cases} (x+y-1.5)^2 - 0.5, & \text{if } (x+y-1.5)^2 > 0.5 \\ 0 & \text{otherwise} \end{cases}$$

over  $(x, y) \in [0, 2.55]^2$ . The graph and the image of the function  $z_2(x, y)$  are shown in Figure 4.14.

To detect these hidden edges in Figures 4.13 and 4.14, we apply our box spline edge detector and numerical results are shown in Figure 4.15.

We next present an example to detect hidden edges with discontinuous second order derivatives. Our artificial example is as follows.

$$z_3(x, y) = \begin{cases} ((x-1)^2 + (y-1)^2 - 0.35)^2, & \text{if } (x-1)^2 + (y-1)^2 \geq 0.35 \\ 0.125 - ((x-1.05)^2 + (y-1)^2)^2 & \text{if } 0.125 \geq ((x-1.05)^2 + (y-1)^2)^2 \\ 0 & \text{otherwise} \end{cases}$$

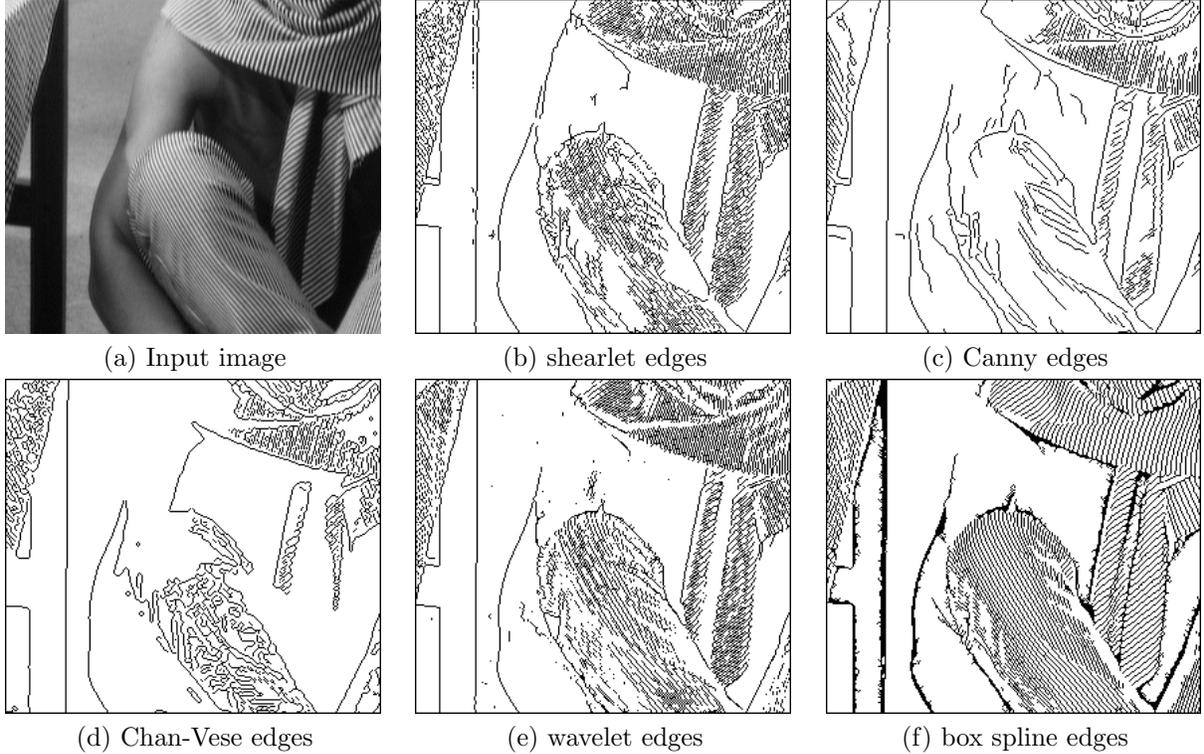


FIG. 4.6. Zoom in comparison of five edge detectors on the Barbara image shown in Figure 4.5.

over  $(x, y) \in [0, 2.55]^2$ . The graph and the image of the function  $z_3(x, y)$  are shown in Figure 4.16. From the image as well as the surface shown in Figure 4.16, we can hardly see by naked eyes any irregularities or defects of the surface. However, our edge detector reveals hidden edges where possible defects may be located (Figure 4.17).

**4.6. Application on object identification.** The results of box spline edge detector can be used for feature extraction, object identification, region segmentation etc. In this subsection, we provide one examples of practical applications: to segment object of interest from medical images.

As shown in Figure 4.18, starting with a 8-bit gray scale (valued in between 0 and 255) cardiac image (top left), we first use the proposed box spline edge detector to create a 1-bit (valued 0 or 1) edge mask (top right), followed by a cleaning up process to remove isolated edges (bottom left). To get the two objects of interest in the center of the image, we drop a small box inside each of the two objects, and grow the regions [44] until it touches the borders of the objects. The results are shown in the bottom right panel. Applying region growing segmentation to the 1-bit 0-1 edge mask is better than to the 8-bit gray scale image as borders of the objects of interest are more reliable and ready to be used in the 1-bit edge map. One may calculate the area or the volume of the region of interests afterwards.

**5. Conclusion.** We present an edge detection algorithm based on a new eight direction box spline tight frame constructed using theory in [27]. Construction of tight wavelet frames based on this box spline is nontrivial as there is theory guarantee the existence of a solution to the polynomial equation (3.6) associated with this eight direction box spline. Also, the mathematical problem for the solution of a general polynomial equation (3.6) is still open. We use a brute force method to solve it. Once the framelets are found, computing edges based on these framelets is very simple and does not need complicated optimization criteria. It applies box splines based wavelet transform to decompose a given image into one low pass and several high pass components. When there is no noise, edges are only the inverse wavelet transform of the high-pass components. When noise exists, one needs some threshold values to separate true edges from noise, both of which belong to high pass components. Quantitative and qualitative comparison with several other existing edge detectors demonstrate the effectiveness and efficiency of the proposed box spline edge detector in

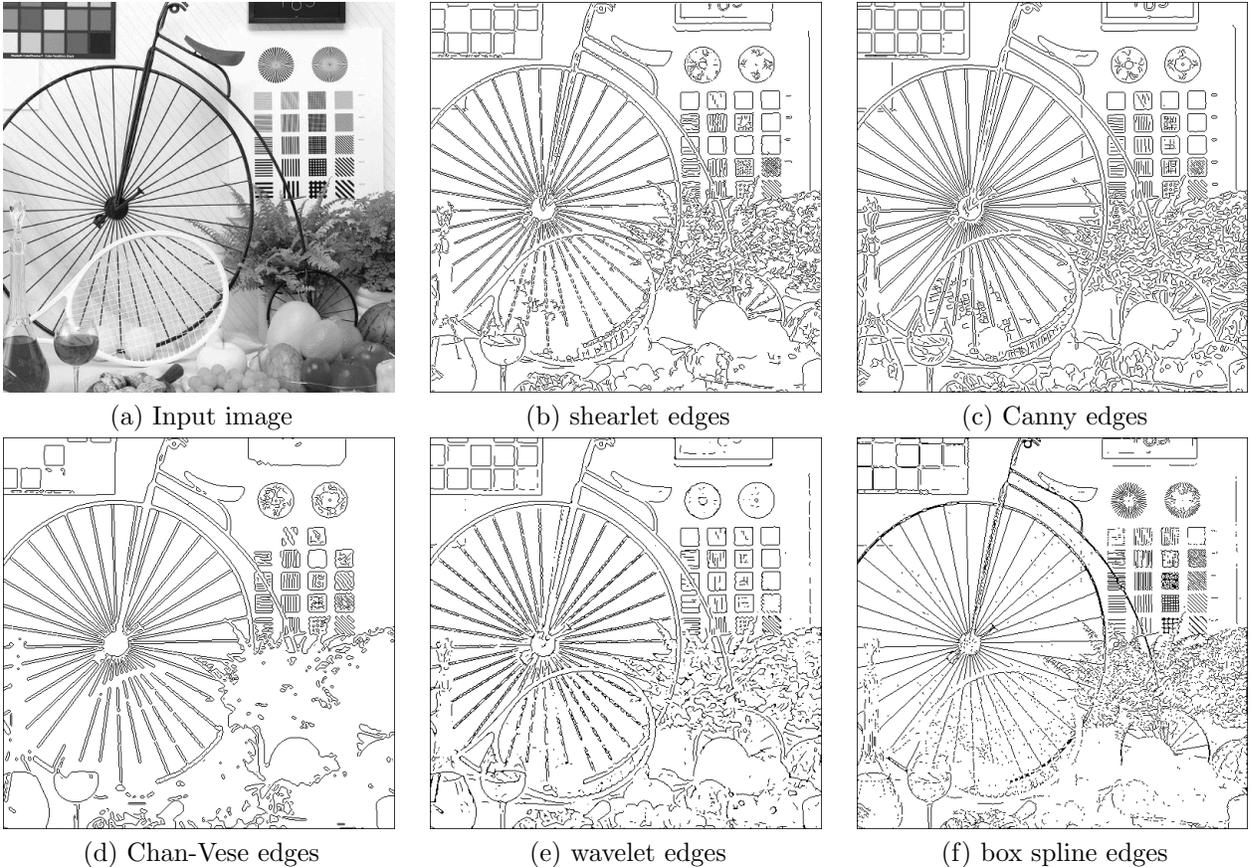


FIG. 4.7. Comparison of five edge detectors on a bike image.

detecting step edges. Our edge detector is also able to detect Dirac edges and hidden edges with discontinuous intensity derivatives. In addition, we show the proposed method is robust to noise. Finally, we apply the edge detector to object identification. One example is given to show its performance.

**6. Acknowledgments.** The second author would like to thank Dr. Kyunglim Nam for her diligent work in help him computing wavelet frame coefficients and implementation of many box spline and B-spline wavelet frames in MATLAB. In addition, the second author also thanks Dr. Alexander Petukhov for discussing weak orthogonal greedy algorithms with him. In addition, we thanks Dr. Sung Ha Kang, George Biros and Wei Zhu for providing us with the medical image studied in this paper.

#### REFERENCES

- [1] K. E. Abdou and K.K. Pratt. Quantitative design and evaluation of enhancement/thresholding edge detectors. *Proc. of IEEE*, 67(5):753–763, 1979.
- [2] L. Ambrosio and V. M. Tortorelli. On the approximation of free discontinuity problems. *boll. Un. Mat. Ital. B*, 7(6):105–123, 1992.
- [3] J. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 8:679–714, 1986.
- [4] T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Trans. Image Processing*, 10(2):266–277, 2001.
- [5] C. K. Chui. *Multivariate splines*, volume 54 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1988. With an appendix by Harvey Diamond.
- [6] C. K. Chui and W. He. Compactly supported tight frames associated with refinable functions. *Appl. Comput. Harmon. Anal.*, 8(3):293–319, 2000.
- [7] C. K. Chui and W. He. Construction of multivariate tight frames via Kronecker products. *Appl. Comput. Harmon. Anal.*, 11(2):305–312, 2001.
- [8] C. K. Chui, W. He, and J. Stöckler. Compactly supported tight and sibling frames with maximum vanishing moments. *Appl. Comput. Harmon. Anal.*, 13(3):224–262, 2002.

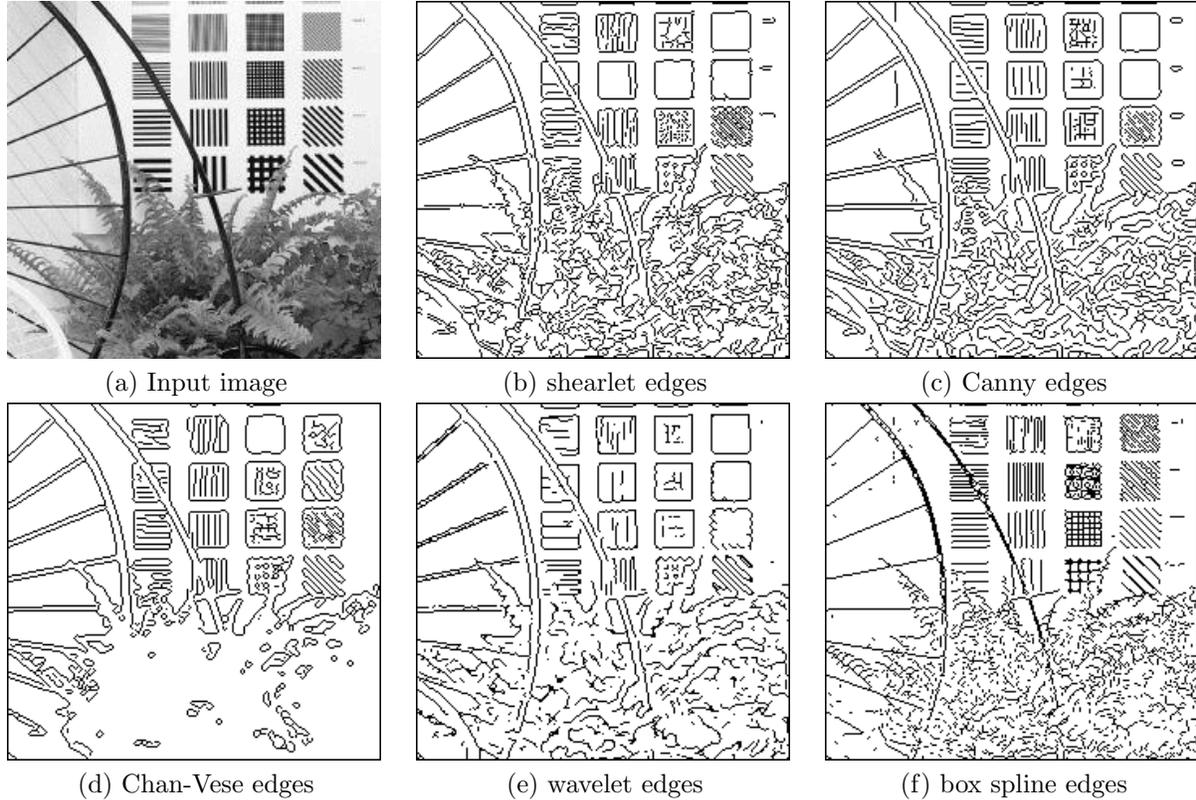


FIG. 4.8. Zoom in comparison of five edge detectors on the bike image shown in Figure 4.7.

- [9] C. K. Chui, W. He, and J. Stöckler. Compactly supported tight and sibling frames with maximum vanishing moments., *Appl. Comput. Harmon. Anal.*, 13:224–262, 2002.
- [10] I. Daubechies. *Ten lectures on wavelets*, volume 61 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1992.
- [11] I. Daubechies, B. Han, A. Ron, and Z. Shen. Framelets: Mra-based constructions of wavelet frames. *Appl. Comput. Harmon. Anal.*, 14(1):1–46, 2003.
- [12] C. de Boor, K. Hölig, and S. Riemenschneider. *Box Splines*. Springer Verlag, 2003.
- [13] R. Deriche. Fast algorithms for low-level vision. *IEEE Tran. Pattern Anal. Mach. Intell.*, 12(A):78–87, 1990.
- [14] G. Easley, D. Labate, and W.Q. Lim. Sparse directional image representations using the discrete shearlet transform. *Appl. Comput. Harmon. Anal.*, 25:2546, 2008.
- [15] J. Geronimo and M. J. Lai. Factorization of multivariate positive laurent polynomials. , *Journal of Approximation Theory*, 139:327–345, 2006.
- [16] W. Guo and F. Huang. A local mutual information guided denoising technique and its application to self-calibrated partially parallel imaging. *MICCAI, Part II, LNCS 5242*, pages 939–947, 2008.
- [17] S. Häuser. Fast Finite Shearlet Transform, February 2012.
- [18] W. He and M. J. Lai. Construction of bivariate compactly supported biorthogonal box spline wavelets with arbitrarily high regularities. *Applied and Computational Harmonic Analysis*, 6:53–74, 1999.
- [19] B. Jahne, H. Schar, and S. Korgel. Principles of filter design. In *B. Jahne, H. HauEecker, and P. GeiEJer, eds., Computer Vision and Applications, vol-volume 2, Signal Processing and Pattern Recognition, chapter 6, Academic Press, San Diego*, pages 125–151, 1999.
- [20] J. J. Koenderink and A. J. van Doorn. Generic neighborhood operators. *IEEE Trans. PAMI*, 14(F):597–605, 1992.
- [21] D. Labate, W.-Q Lim, G. Kutyniok, and G. Weiss. Sparse multidimensional representation using shearlets. *Proc. SPIE Wavelets XI*, 5914:254262, 2005.
- [22] M. J. Lai. Fortran subroutines for  $B$ -nets of box splines on three- and four-directional meshes. *Numer. Algorithms*, 2(1):33–38, 1992.
- [23] M. J. Lai. Construction of multivariate compactly supported prewavelets in  $l_2$  spaces and pre-riesz basis in sobolev spaces. *Journal of Approximation Theory*, 142:83–115, 2006.
- [24] M. J. Lai and K. Nam. On the number of tight wavelet framelets associated with multivariate box splines. *Journal of Approximation Theory and its Application*, 2008.
- [25] M. J. Lai and A. Petukhov. Method of virtual components in the multivariate setting. *Journal of Fourier Analysis and Its Applications*, 16:471–494, 2010.
- [26] M. J. Lai and L. L. Schumaker. *Spline Functions over Triangulations*. Cambridge University Press, 2007.

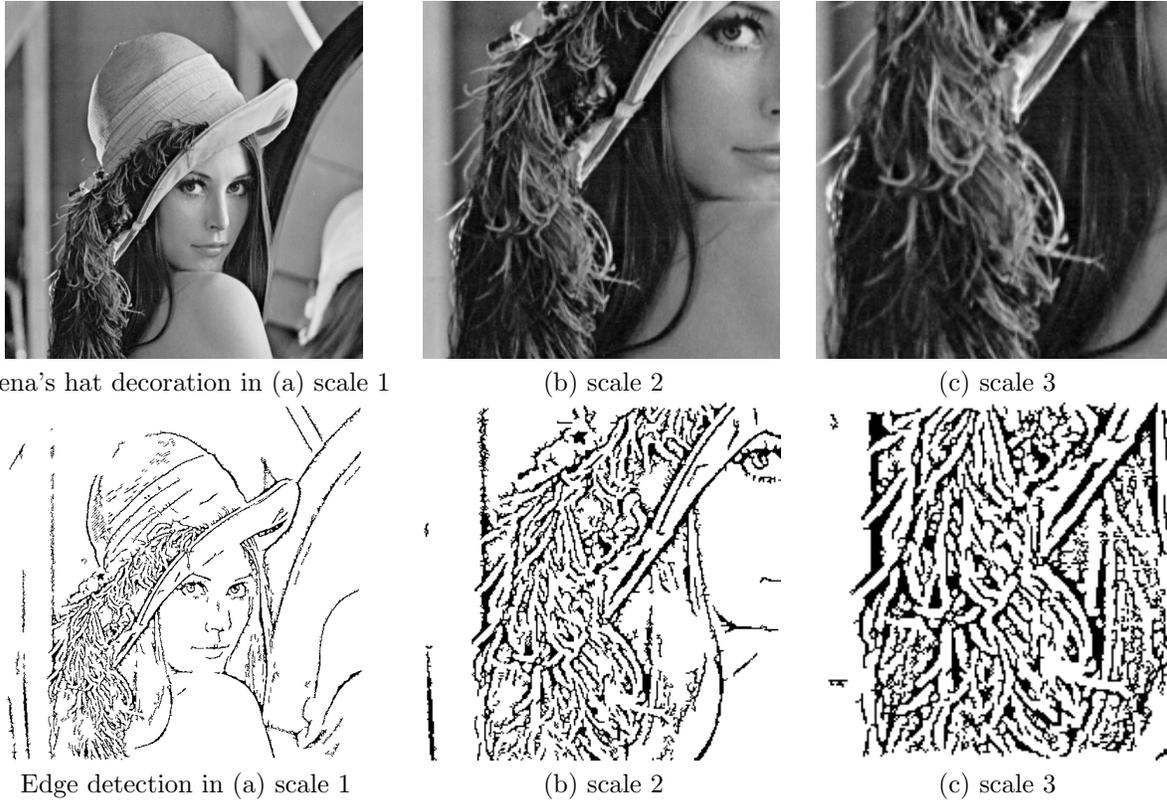


FIG. 4.9. Illustration of scale invariance of the proposed edge detection method. Detect edges at three different scales of an object of interest – the decoration on Lena’s hat.

- [27] M.-J. Lai and J. Stöckler. Construction of multivariate compactly supported tight wavelet frames. *Appl. Comput. Harmon. Anal.*, 21(3):324–348, 2006.
- [28] S. Lanser and W. Eckstein. Eine modification des deriche-verfahrens zur kantendetektion. *B. Radig, ed., Mustererkennung 1991, vol. 290 of Informatik Fachberichte, DAGM Symposium, Munchen, Springer, Berlin*, pages 151–158, 1991.
- [29] S. Mahmoodi. Edge detection filter based on mumford-shah green function. *SIAM J. Imaging Sciences*, 5(1):343–365, 2012.
- [30] S. Mallat. Characterization of signals from multiscale edges. *IEEE. Trans. Patt. Anal. Machine Intell.*, 14:710–732, 1992.
- [31] D. Marr and E. Hildreth. Theory of edge detection. *Proc. Royal Society, London*, pages 187–217, 1980.
- [32] D. Mumford and J. Shah. Optimal approximation by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.*, 42:557–685, 1989.
- [33] K. Nam. *Tight Frame Construction and Its Application for Image Processing*. PhD thesis, University of Georgia, 2005.
- [34] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. *J. Comput. Phys.*, 79:12–49, 1988.
- [35] I. Pitas and A.N. Venetsanopoulos. Nonlinear digital filters: Principles and applications. *Kluwer Academic Publishers*, 1990.
- [36] W. K. Pratt. Digital image processing. *New York: Wiley-Interscience*, 1978.
- [37] D. Qi, F. Guo, and L. Yu. Medical image edge detection based on omnidirectional multi-scale structure element of mathematical morphology. *IEEE International Conference on Automation and Logistics*, pages 2281–2286, 2007.
- [38] A. Ron and Z. Shen. Compactly supported tight affine spline frames in  $L_2(\mathbf{R}^d)$ . *Math. Comp.*, 67(221):191–207, 1998.
- [39] A. Ron and Z. Shen. Construction of compactly supported affine frames in  $l_2(r^d)$ . In *Advances in Wavelets*, 1998.
- [40] J. Serra. Image analysis and mathematical morphology. *Academic Press*, 1982.
- [41] W. Stefan, R. A. Renaut, and A. Gelb. Improved total variation-type regularization using higher-order edge detectors. *SIAM Journal on Imaging Sciences*, 3(2):232–251, June 2010.
- [42] V. N. Temlyakov. Greedy approximation. *Acta Numer*, 17:235409, 2008.
- [43] B. Tian, H. Yuan, and X. Yue. Feature extraction algorithm for space targets based on fractal theory. *Second International on Space Information Technology, SPIE-INT SOC Optical Engineering*, 6795:79518–79518, 2007.
- [44] S. Umbaugh. Computer vision and image processing: A practical approach using cviptools. 1997.
- [45] S. Yi, D. Labate, R. G. Easley, and H. Krim. A shearlet approach to edge edge analysis and detection. *IEEE trans. Imag. Proc.*, 18(5):929–941, 2009.
- [46] L. Zhai, S. Dong, and H. Ma. Recent methods and applications on image edge detection. *International Workshop on Geoscience and Remote Sensing*, pages 332 – 335, 2008.

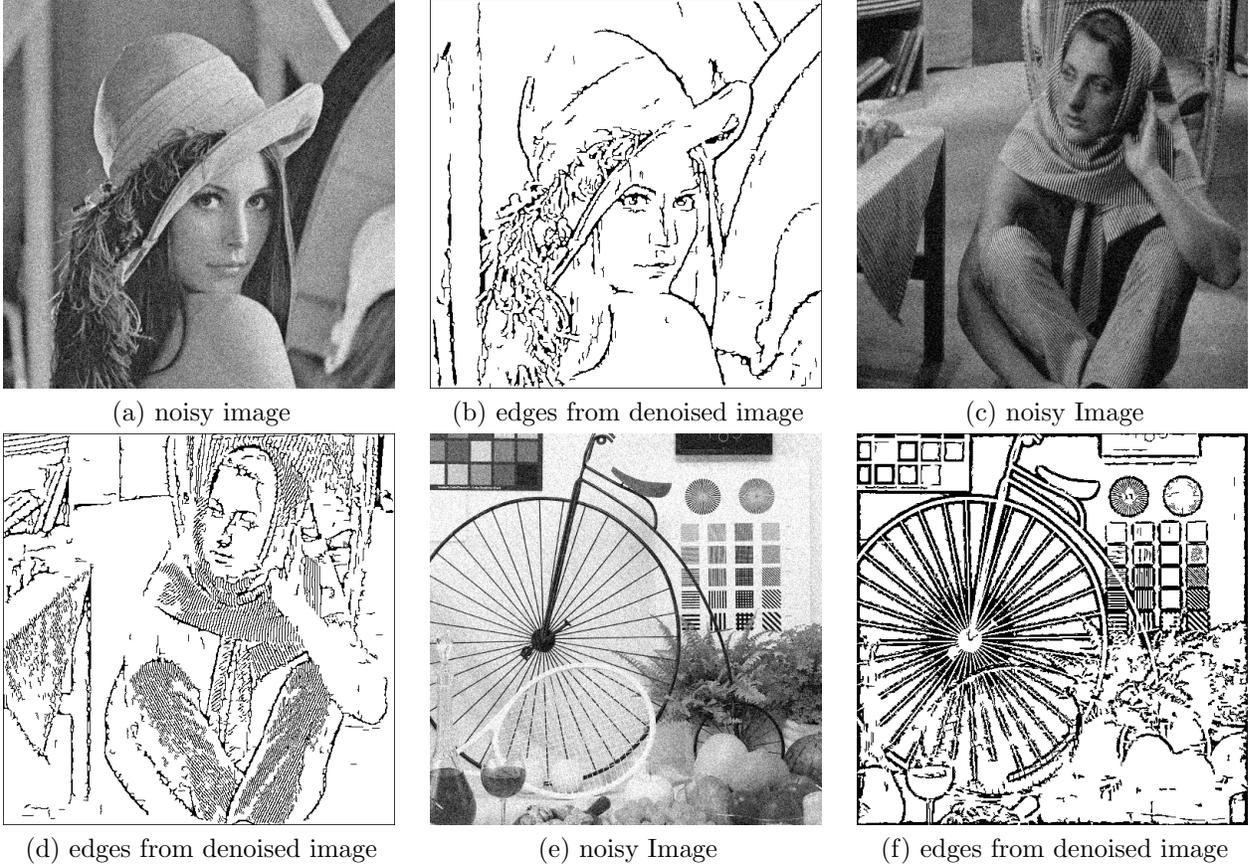


FIG. 4.10. Edges detected from noisy images.

- [47] L. Zhang, A. Butler, and C. Sun. Fractal dimension assessment of brain white matter structural complexity post stroke in relation to upper-extremity motor function. *Brain Research*, 1228(4):229–240, 2008.
- [48] D. Ziou and S. Tabbone. Edge detection techniques - an overview. *International Journal of Pattern Recognition and Image Analysis*, 8:537–559, 1998.

**7. Appendix.** In this Appendix, we mainly give some ideas about the low pass filter  $P$  and one of the 14 tight framelets  $\psi^\ell$  in terms of Fourier transform. That is,

$$\widehat{\psi}_8^\ell(\xi, \eta) = Q^\ell(\xi/2, \eta/2) \widehat{\phi}_8(\xi/2, \eta/2), \quad \ell = 1, \dots, 14, \quad (7.1)$$

where  $Q^\ell(\xi, \eta) = \sum_j \sum_k q_{jk}^\ell e^{-ij\xi} e^{-ik\eta}$ .

The low-pass filter  $P$  is

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 2 & 2 & 2 & 1 & 0 & 0 \\ 0 & 1 & 2 & 4 & 5 & 5 & 4 & 2 & 1 & 0 \\ 0 & 2 & 4 & 6 & 8 & 8 & 6 & 4 & 2 & 0 \\ 1 & 2 & 5 & 8 & 10 & 10 & 8 & 5 & 2 & 1 \\ 1 & 2 & 5 & 8 & 10 & 10 & 8 & 5 & 2 & 1 \\ 0 & 2 & 4 & 6 & 8 & 8 & 6 & 4 & 2 & 0 \\ 0 & 1 & 2 & 4 & 5 & 5 & 4 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 2 & 2 & 2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} /256.$$

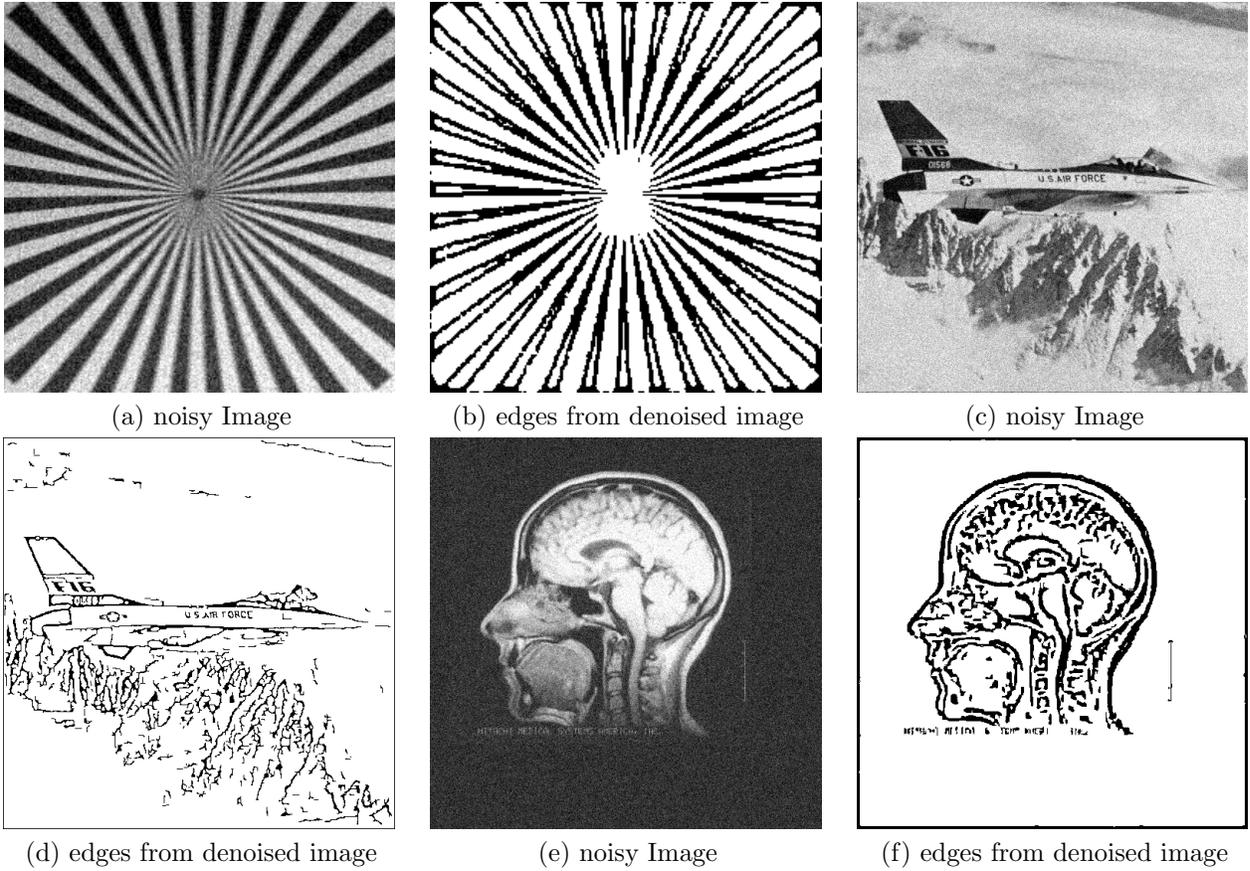


FIG. 4.11. Edges detected from noisy images.

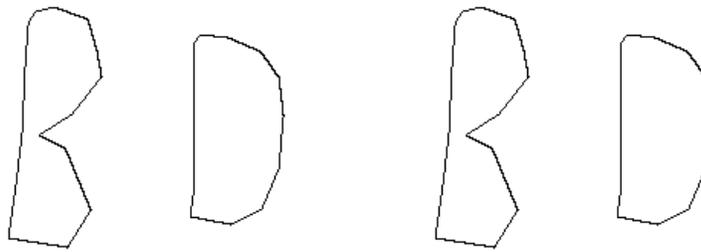


FIG. 4.12. The proposed box spline edge detector is able to detect Dirac edges.

For  $Q^{(1)}$ , the coefficient matrix  $Q_1 = [q_{jk}^{(1)}]_{1 \leq j, k \leq 18}$  is

0	0	0	0	0	0	-1/32768
0	0	0	0	-1/32768	-1/16384	-1/8192
0	0	0	-1/32768	-1/16384	-3/16384	-13/32768
0	0	0	-1/16384	-1/4096	-9/16384	-1/1024
0	0	-1/32768	-3/16384	-1/2048	-37/32768	-63/32768
0	0	-1/16384	-3/8192	-15/16384	-1/512	-27/8192
0	-1/32768	-3/16384	-19/32768	-47/32768	-32/11275	-53/11351
0	-1/16384	-1/4096	-25/32768	-15/8192	-29/8192	-95/16384
-1/32768	-1/16384	-5/16384	-7/8192	-33/16384	-125/32768	-51/8192
-1/32768	-1/16384	-5/16384	-13/16384	-61/32768	-29/8192	-95/16384
0	-1/16384	-3/16384	-5/8192	-47/32768	-32/11275	-53/11351
0	-1/32768	-1/8192	-13/32768	-15/16384	-63/32768	-53/16384
0	0	-1/32768	-3/16384	-1/2048	-9/8192	-63/32768
0	0	0	-1/16384	-3/16384	-1/2048	-15/16384
0	0	0	0	-1/16384	-3/16384	-13/32768
0	0	0	0	0	-1/32768	-1/8192
0	0	0	0	0	0	1/32768

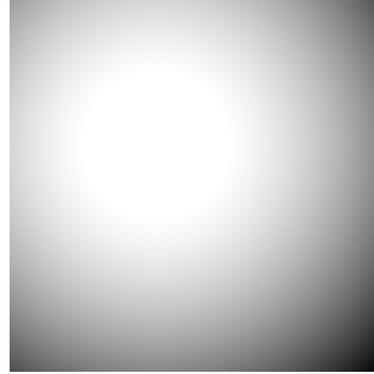
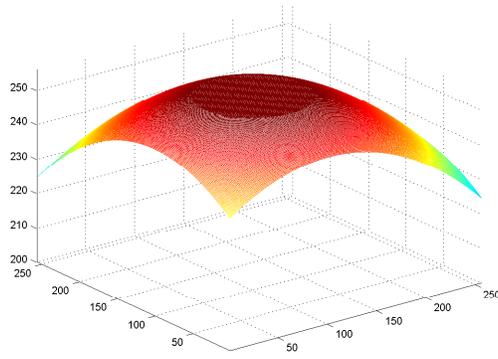


FIG. 4.13. A function with discontinuous derivatives around a circle.

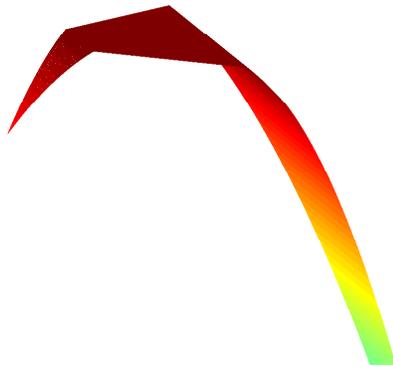


FIG. 4.14. Another function with discontinuous derivatives at two lines.

$-1/32768$	$-1/16384$	$-1/16384$	$-1/16384$	$-1/16384$	0	0
$-3/16384$	$-1/4096$	$-9/32768$	$-1/4096$	$-3/16384$	$-1/8192$	$-1/16384$
$-19/32768$	$-13/16384$	$-7/8192$	$-25/32768$	$-5/8192$	$-3/8192$	$-3/16384$
$-3/2048$	$-15/8192$	$-1/512$	$-15/8192$	$-3/2048$	$-1/1024$	$-9/16384$
$-32/11275$	$-29/8192$	$-125/32768$	$-29/8192$	$-32/11275$	$-1/512$	$-9/8192$
$-53/11351$	$-47/8192$	$-101/16384$	$-47/8192$	$-53/11351$	$-27/8192$	$-1/512$
$-217/32768$	$-133/16384$	$-33/3821$	$-72/8903$	$-27/4096$	$-53/11351$	$-32/11275$
$-72/8903$	$-81/8192$	$-43/4096$	$-72/7327$	$-72/8903$	$-47/8192$	$-29/8192$
$-33/3821$	$-78/7387$	$541/1107$	$-43/4096$	$-33/3821$	$-101/16384$	$-125/32768$
$-133/16384$	$-81/8192$	$-78/7387$	$-81/8192$	$-133/16384$	$-47/8192$	$-29/8192$
$-27/4096$	$-133/16384$	$-33/3821$	$-72/8903$	$-217/32768$	$-53/11351$	$-32/11275$
$-53/11351$	$-95/16384$	$-51/8192$	$-95/16384$	$-53/11351$	$-27/8192$	$-63/32768$
$-32/11275$	$-29/8192$	$-125/32768$	$-29/8192$	$-32/11275$	$-1/512$	$-37/32768$
$-47/32768$	$-61/32768$	$-33/16384$	$-15/8192$	$-47/32768$	$-15/16384$	$-1/2048$
$-5/8192$	$-13/16384$	$-7/8192$	$-25/32768$	$-19/32768$	$-3/8192$	$-3/16384$
$-3/16384$	$-5/16384$	$-5/16384$	$-1/4096$	$-3/16384$	$-1/16384$	$-1/32768$
$-1/16384$	$-1/16384$	$-1/16384$	$-1/16384$	$-1/32768$	0	0

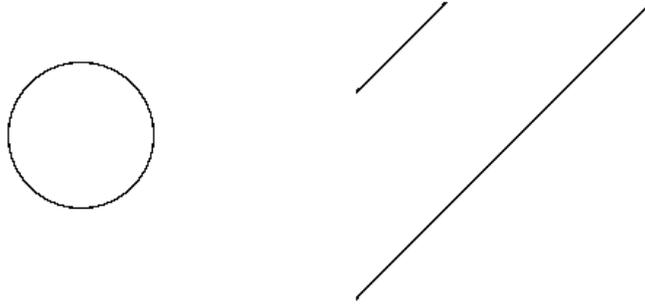


FIG. 4.15. Locations of hidden edges of Figures 4.13 (left) and 4.14(right).

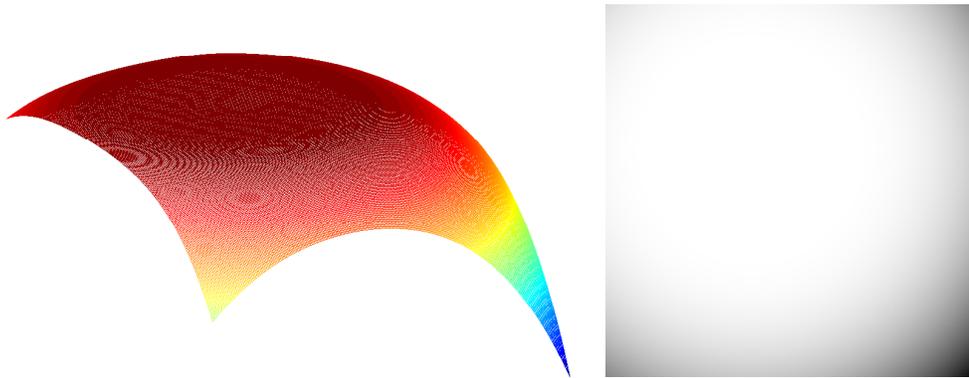


FIG. 4.16. A function with discontinuous second order derivatives at two circles.

$$\begin{bmatrix}
 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 \\
 -1/16384 & 0 & 0 & 0 \\
 -1/4096 & -1/16384 & 0 & 0 \\
 -9/16384 & -3/16384 & -1/16384 & 0 \\
 -1/1024 & -3/8192 & -1/8192 & 0 \\
 -3/2048 & -5/8192 & -3/16384 & -1/16384 \\
 -15/8192 & -25/32768 & -1/4096 & -1/16384 \\
 -1/512 & -7/8192 & -9/32768 & -1/16384 \\
 -15/8192 & -13/16384 & -1/4096 & -1/16384 \\
 -3/2048 & -19/32768 & -3/16384 & -1/32768 \\
 -1/1024 & -13/32768 & -1/8192 & -1/32768 \\
 -9/16384 & -3/16384 & -1/16384 & 0 \\
 -1/4096 & -1/16384 & -1/32768 & 0 \\
 -1/16384 & -1/32768 & 0 & 0 \\
 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0
 \end{bmatrix} .$$

The remaining high-pass filters (in fact all low and high-pass filters) can be obtained in an electronic form per request.

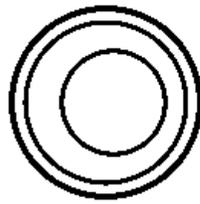


FIG. 4.17. Hidden edges found from Figure 4.16 using the proposed box spline edge detector.

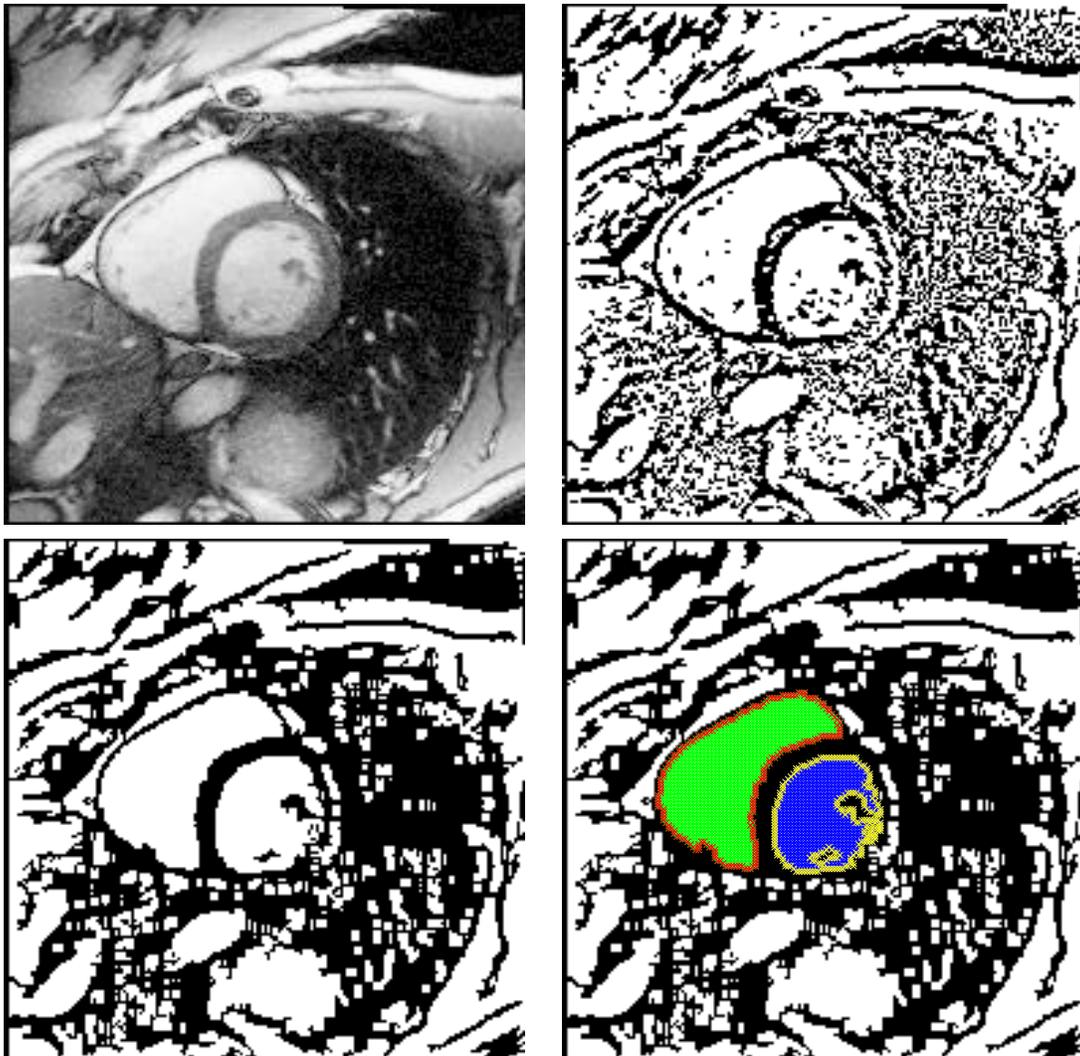


FIG. 4.18. A medical image (top left), the resulting image based on box spline tight-wavelet frame detector (top right), cleaner edges (bottom left), and the outline of two regions of interest (bottom right).