

# Scale Space Smoothing, Image Feature Extraction and Bessel Filters

Sasan Mahmoodi and Steve Gunn

School of Electronics and Computer Science, Building 1, Southampton University,  
Southampton, SO17 1BJ, UK  
{sm3, srg}@ecs.soton.ac.uk

**Abstract.** The Green function of Mumford-Shah functional in the absence of discontinuities is known to be a modified Bessel function of the second kind and zero degree. Such a Bessel function is regularized here and used as a filter for feature extraction. It is demonstrated in this paper that a Bessel filter does not follow the scale space smoothing property of bounded linear filters such as Gaussian filters. The features extracted by the Bessel filter are therefore scale invariant. Edges, blobs, and junctions are features considered here to show that the extracted features remain unchanged by varying the scale of a Bessel filter. The scale invariance property of Bessel filters for edges is analytically proved here. We conjecture that Bessel filters also enjoy this scale invariance property for other kinds of features. The experimental results presented here confirm our conjecture of the scale invariance property of the Bessel filters.

**Keywords:** Scale Space, Linear Filtering, Bessel Filter, Feature Extraction.

## 1 Introduction

Scale space theory has been established based on the application of a series of bounded linear filters such as a Gaussian filter on images. In a scale space setting, it is observed that the features extracted from an image may change if the scale of a Gaussian filter varies. This phenomenon is known as scale space smoothing. Such an observation has led to the scale space theory to propose a framework to select the “most important” scale in which a feature should be extracted [1],[2]. Bounded linear filters usually demonstrate scale space smoothing. However, in this paper, we demonstrate that a Bessel filter which is unbounded at the center does not show any property associated with the scale space smoothing when it is used to extract features such as edges, ridges, blobs, and corners. Such an unbounded filter is numerically intractable. We therefore propose a method here to regularize the filter. The rest of the paper is structured as follows. Section 2 introduces the Bessel filter and the theory behind it and a regularization method for implementations is presented in section 3. Section 4 deals with the numerical results. The paper concludes in section 5.

## 2 Theory

The Green function associated with the Mumford-Shah functional for the whole plane, in the absence of boundaries, is a modified Bessel function of the second kind and zero degree [3]. A closed form of this Green function is written as:

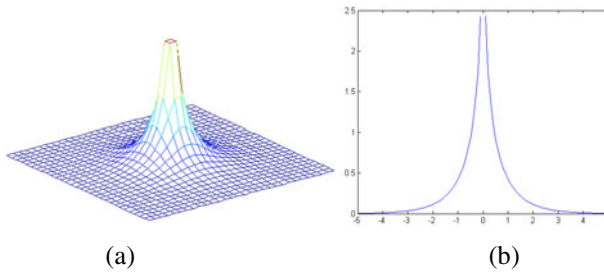
$$h(x, y; v, w) = K_0 \left( \sqrt{\frac{(x-v)^2 + (y-w)^2}{\mu}} \right) \tag{1}$$

where  $K_0(\cdot)$  is the modified Bessel function of the second kind and zero degree and  $(x, y), (v, w) \in \mathbb{R}^2$ .

A three dimensional view of Green function (1) for  $(u, w) = (0,0)$  is shown in figure (1-a). The cross-section of this Green function is also depicted in figure (1-b). As shown in figure (1), Bessel function (1) is singular at the centre which is not numerically tractable for implementation. It is therefore important for numerical purposes to regularize function  $h$ .

In this paper, we suggest the following regularized and normalized function named as *Bessel filter*.

$$h_\epsilon(r) = \begin{cases} \frac{K_0(\frac{|r|}{\sqrt{\mu}})}{K_0(\frac{\epsilon}{2\sqrt{\mu}})} & |r| > \frac{\epsilon}{2} \\ 1 & |r| \leq \frac{\epsilon}{2} \end{cases} \tag{2}$$



**Fig. 1.** The M-S Green function for  $\mu = 10$ : a) 3D view b) a cross-section

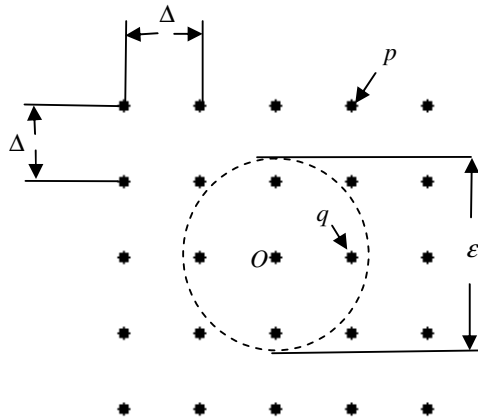
The construction of the filter proposed in equation (2) is explained in more details in section 3 and figure (3). We are here inspired by the following theorem proved for an edge detection algorithm based on the Bessel filter [4].

**Theorem:** *The gradient magnitude of the convolved image  $u: \mathbb{R}^2 \rightarrow \mathbb{R}^+$  calculated in  $u = h_\epsilon * g$  has local maxima on discontinuities of a given piecewise-constant image  $g: \Omega \rightarrow \mathbb{R}^+$  as  $\epsilon \rightarrow 0$ .*

The above theorem implies that scale-space smoothing is not applicable to Green function (2) when it is used for edge detection. In fact, there is a family of filters which are scale invariant for edge detection. The Bessel filter investigated in this paper is a member of this family. Another example in such a family is also designed and analyzed in [5]. We conjecture in this paper that Green function (1) does not demonstrate any property associated with the scale space smoothing for the detection of any feature, i.e. features extracted by the Bessel filter investigated here are scale invariant. The numerical results presented in section 4 of this paper support such a conjecture.

### 3 Implementation Issues

The convolution of the function  $h$  in (1) with the original input image  $g$  is calculated as the first step for feature extraction to calculate the convolved image  $u$ . Then an appropriate differential entity of  $u$  is examined to find regions for which this differential entity is maximum.



**Fig. 2.** A 5 x 5 grid used for the construction of a Bessel filter proposed in equation (2) for the case of  $2 < \frac{\epsilon}{\Delta} < 3$

The algorithm proposed here is implemented in Matlab 7.3 environment. We have exploited the built in *besselk* function in Matlab to construct the Bessel filter. Figure (2) helps us understand how the regularized Bessel filter proposed in (2) can be constructed. A 5 x 5 window forming the window grids of the filter is shown in this figure. It is noted that  $\Delta$  and  $\epsilon$  in the figure are the sampling distance and regularizing parameter respectively. The dashed line circle in figure (2) represents the circle with a radius given in (3).

$$|r| = \frac{\epsilon}{2} \quad (3)$$

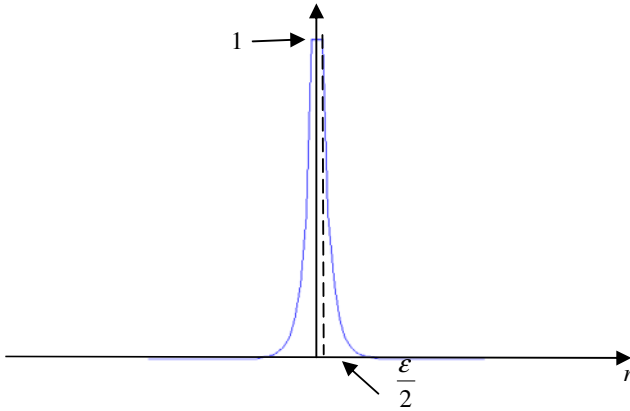


Fig. 3. The cross section of the regularized filter proposed in (2)

According to equation (2), the values of the filter in all points inside this circle should be unity and the filter values in all points outside of this circle should be set to the following value:

$$h_{\epsilon}(r) = \frac{K_0\left(\frac{|r|}{\sqrt{\mu}}\right)}{K_0\left(\frac{\epsilon}{2\sqrt{\mu}}\right)} \tag{4}$$

where  $|r|$  is the distance between the central point  $O$  and the point outside of the circle. If the point is the outside of the dashed line circle (such as point  $p$  shown in the figure), then the value given in equation (4) is therefore assigned to the filter at point  $p$ . For a point inside the circle (such as points  $q$  and  $O$ ), unity is assigned to the filter at this point. For the case when  $\epsilon < \Delta$ , the dashed circle contains only the central point  $O$  and therefore the value of the filter at only the central point  $O$  is set to unity.

We consider  $\Delta$  being unity throughout this paper. Having constructed the filter, the derivatives of the filter with respect to  $x$  and  $y$  are convolved with the original image to compute the derivatives of the image. Such derivatives can then be used in various differentials entities for image feature extraction. A cross section of constructed regularized Bessel filter is also depicted in figure (3). Finally we need to determine the window size according to which the Bessel filter defined in (2) is truncated, since this filter in spatial domain is not band limited. Therefore we define the truncated filter as:

$$h_{\epsilon}(r, d) = \begin{cases} 1 & 0 \leq r < \frac{\epsilon}{2} \\ \frac{K_0\left(\frac{r}{\sqrt{\mu}}\right)}{K_0\left(\frac{\epsilon}{\sqrt{\mu}}\right)} & \frac{\epsilon}{2} < r \leq d \\ 0 & r > d \end{cases}$$

We use the term defined in (5) to determine a filter size for the Bessel filter with respect to  $\mu$ .

$$E(d) = \sum_r |h_\varepsilon(r) - h_\varepsilon(r, d)| \quad (5)$$

We notice that for,  $d = 5\sqrt{\mu}$   $E$  becomes negligible ( $E = 0.0045$ ) in comparison with the total area under the filter  $h_\varepsilon(r)$ , i.e.,

$$E(d = 5\sqrt{\mu}) \ll \sum_r h_\varepsilon(r)$$

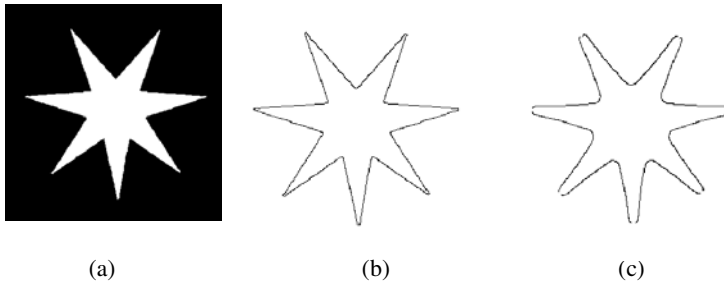
Therefore, in this paper the window size of the filter proposed here is set to  $W = (10\sqrt{\mu} + 1) \times (10\sqrt{\mu} + 1)$ . It is clear that if we choose larger window sizes than this, the algorithm becomes numerically more expensive with almost the same accuracy. In summary, once the Bessel filter is constructed, its derivatives with respect to  $x$  and  $y$  directions are numerically calculated and then these derivatives are convolved with an input image to compute the derivatives of the image in  $x$  and  $y$  directions. The various differential entities investigated in this paper are finally computed to extract appropriate features for image analysis by detecting the local maxima of the corresponding differential entities.

## 4 Numerical Results

Let us start this section by presenting our numerical results on edge detection. It is customary in literature (see e.g. [6]) to convolve an image with a filter and then find the maxima of the absolute value of the gradient of the convolved image to detect edges, i.e., the maxima of the following differential entity correspond to the edges of image  $I$ .

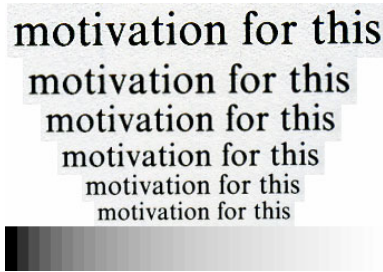
$$Y = |\nabla u| \quad (6)$$

where  $u = h * I$  and  $h$  is the filter. The image of figure (4-a) is examined for edge detection. Gaussian and Bessel filters with size 41 are employed for edge detection to produce the results in figures (4-b and c).

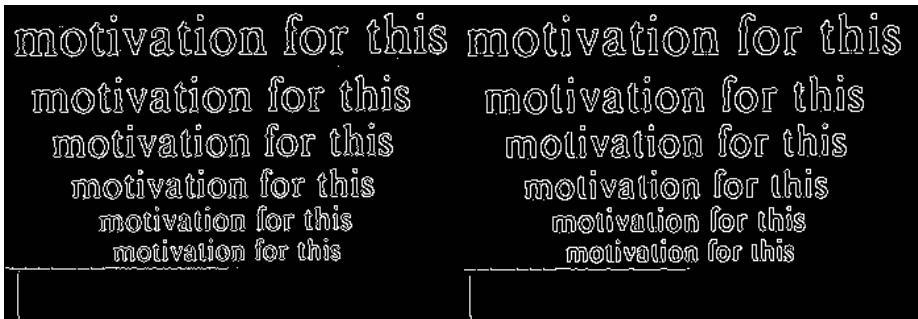


**Fig. 4.** Edge detection with two filters with the size 41, a) Original image b) Edge map produced by Bessel filter ( $\varepsilon = 0.1$ ) c) Edge map produced by Gaussian filter

As shown in figure (4), the corners in the edge map produced by a Gaussian filter, become distorted when the filter size (and hence scale) is considerably high. This phenomenon is known as scale space smoothing. The corners in the edge map produced by the Bessel filter however do not show any distortion as the filter size (scale) increases. We note that the filter parameters such as the standard deviation ( $\sigma = \frac{\text{filter size}}{6}$ ) of the Gaussian filter and  $\mu$  for Bessel filter are calculated according to the filter size. Figure (5-a) shows a real world image whose edge maps are detected by the Bessel and Gaussian filters. This figure depicts another example for edge detection showing that the Bessel filter can extract same features regardless of the scale. However some features are distorted in some scales when a Gaussian filter is used for edge detection. Same filter size (size =9) is used for both filters. As shown in this figure, some letters (such as “t”, “f”, “a”, “m” and so on) even in lower scales are detected with better accuracy by the Bessel filter. A noisy synthetic image is also shown in figure (6-a). In order to remove the noise, the filter size 13 is chosen for both filters. More details are extracted in the edge map produced by the Bessel filter. According to the theorem presented in section 2, as  $\epsilon \rightarrow 0$  higher accuracy in edge detection is achieved by the Bessel filter. The parameter  $\epsilon$  however is set to unity for the purpose of noise removal in this example. As can be seen from this figure, a better accuracy in edge detection is achieved by the Bessel filter.



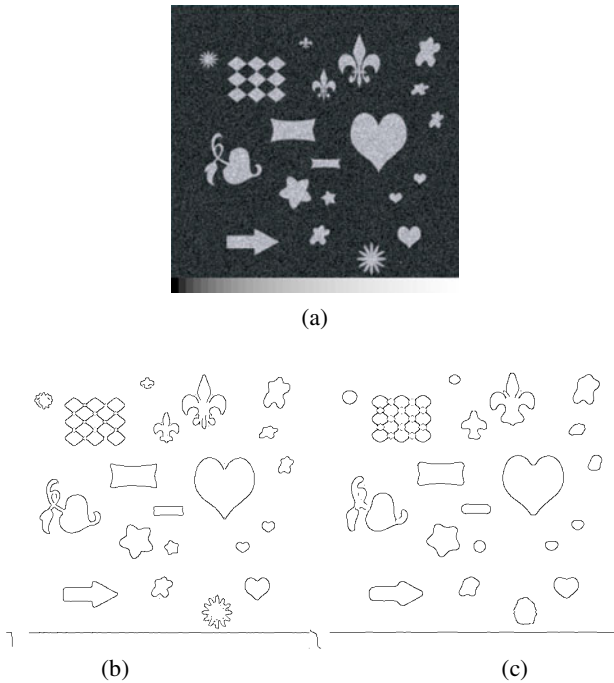
(a)



(b)

(c)

**Fig. 5.** Edge maps in various scales a) Original image b) Edge map produced by the Bessel filter (size=9,  $\epsilon = 0.1$ ), c) Edge map produced by the Gaussian filter(size=9)



**Fig. 6.** Edge maps of a noisy image produced by the Gaussian and Bessel filters (size=13) a) Original noisy image b) Edge map produced by the Bessel filter ( $\varepsilon = 1$ ) c) Edge map produced by the Gaussian filter

In both figures (5) and (6), a stair case strip of logarithmically increasing grey scales is added at the bottom of the original images to ensure that the equivalent values for thresholds are used in both algorithms.

Figure (7-a) shows a Gaussian circle. Bessel filters with size 9 and 301 are applied to this Gaussian circle to detect edges as shown in figures (7-b and c). Gaussian filters with the same sizes are applied to the Gaussian circle to produce edge maps shown in figures (7-d and e).

As shown in figure (7), the edge map of the Gaussian circle significantly changes when the filter size (and consequently the scale) of the Gaussian filter increases from 9 to 301. However a slight change in edge maps is observed when the Bessel filter with two different sizes are used. The reason for the slight change in the edge maps is that it is not numerically tractable to set  $\varepsilon$  to zero. Analytically it can be proved, as  $\varepsilon \rightarrow 0$  the edge maps produced by the Bessel filter will be unchanged, regardless of the filter size (the proof will be similar to the proofs of the theorems presented in [4]).

For blob detection, the maxima of the following differential entity are detected.

$$\Gamma = -\nabla^2 u$$

where  $u = h * I$  in which  $h$  is either a Bessel or a Gaussian filter and  $I$  is the input image. Figure (8) shows the first 11 maxima of  $\Gamma$  for blob detection using the Bessel

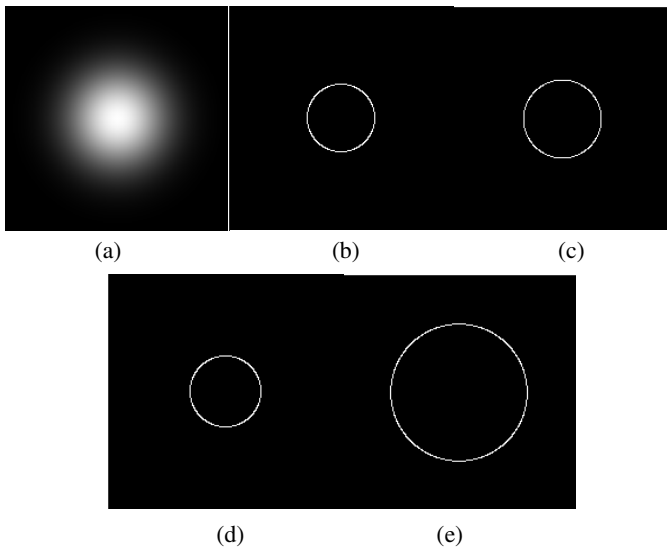
and Gaussian filters with two different sizes. As shown in figures (8-a and b), lower scale blobs are detected by Bessel filters with both sizes 21 and 61. However, by increasing the size of the Gaussian filter, the detected blobs change significantly (see figures (8-c and d)). As can be seen from figure (8), Bessel filters produce scale invariant features when they are used for blob detection. It is noted that for Bessel filters the values of the local maxima of  $\Gamma$  always decrease from the top to the bottom of the blob pattern image shown in figure (8) in all scales (small and large scales).

However the more bottom the blob is in the image of figure (8), the higher the value of the local maxima of  $\Gamma$  is, when a large scale is used for a Gaussian filter. For a Gaussian filter with a smaller scale, on the other hand, the values of the local maxima of  $\Gamma$  decrease from the top of the blob pattern image to its bottom.

The corners of the image shown in figure (9) are detected by convolving the original image with both Bessel and Gaussian filters and finding the maxima of the following differential entity:

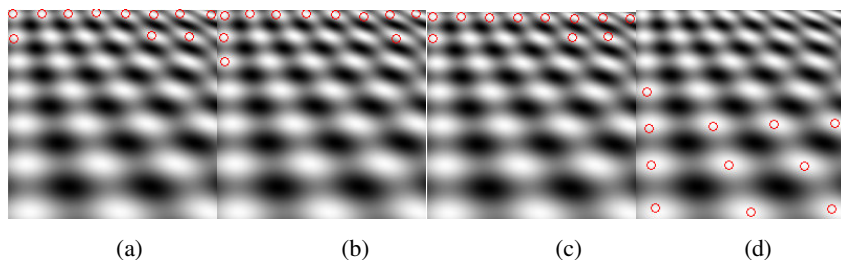
$$\Xi = u_y^2 u_{xx} - 2u_x u_y u_{xy} + u_x^2 u_{yy}$$

where  $u = h * I$  in which  $h$  is either a Bessel or a Gaussian filter and  $I$  is the input image. The first seven maxima of the differential entity  $\Xi$  are plotted in figure (9). As can be seen from figure (9), the locations of the detected corners change, when Gaussian filters with higher scales are employed. This is due to the scale space smoothing property associated with Gaussian filters. However the corner locations detected by the Bessel filters with various sizes remain unchanged.

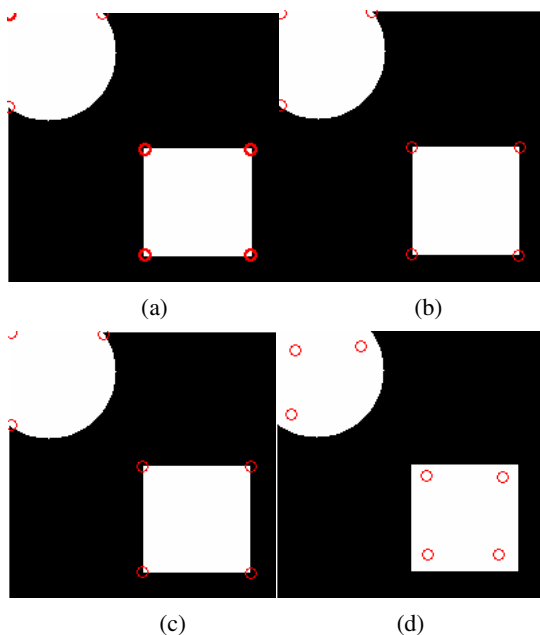


**Fig. 7.** The effect of scale space smoothing on the edge maps of a Gaussian circle a) Original image b) edge map produced by the Bessel filter with size=9 c) edge map produced by the Bessel filter with size=301 d) edge map produced by the Gaussian filter with size=9 e) edge map produced by the Gaussian filter with size=301





**Fig. 8.** The first 11 blobs detected by using Bessel and Gaussian filters with two different filter sizes a) Bessel filter with size=21 ( $\varepsilon = 0.1$ ) b) Bessel filter with size=61 ( $\varepsilon = 0.1$ ) c) Gaussian filter with size 21 and d) Gaussian filter with size 61



**Fig. 9.** The first seven most important corners detected by Bessel and Gaussian filters a) Bessel filter with size=9 ( $\varepsilon = 1$ ) b) Bessel filter with size=201 ( $\varepsilon = 1$ ) c) Gaussian filter with size=9, d) Gaussian filter with size=201

## 5 Conclusion

The Green function associated with the Mumford-Shah functional in the absence of discontinuities is considered in this paper for feature extraction. A regularization method is presented here to introduce a Bessel filter. It is analytically proved in this work that Bessel filters produce scale invariant features for edge detection. It is therefore conjectured here that Bessel filters always produce scale invariant features for the detection of other features. Numerical results for Gaussian edges, blobs, corners as

well as edges presented in this paper support such a conjecture. The features extracted by Bessel filters are invariant to the scale (size) of the filter.

**Acknowledgments.** This work was supported in part by the IST program of the European Community, under the PASCAL2 Network of Excellence, the IST-2007-216886 and PinView project with grant number 216529.

## References

1. Lindeberg, T.: Feature Detection with Automatic Scale Selection. *International Journal of Computer Vision* 30(2), 79–116 (1998)
2. Lindeberg, T.: Edge Detection and Ridge Detection with Automatic Scale Selection. *International Journal of Computer Vision* 30(2), 117–154 (1998)
3. Mumford, D., Shah, J.: Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics* 42(4), 577–688 (1989)
4. Mahmoodi, S.: Technical Report 1: Bessel Filter Analysis, University of Southampton, School of Electronic and Computer Science (December 2010), <http://users.ecs.soton.ac.uk/sm3/BesselFilterTheorems.pdf>
5. Mahmoodi, S.: Scale invariant filtering design and analysis for edge detection. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* (in press)
6. Canny, J.F.: A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 8(6), 679–698 (1986)