

# A Novel Method for Generating Scale Space Kernels Based on Wavelet Theory

Dionísio Doering<sup>1</sup>

Adalberto Schuck Júnior<sup>2</sup>

**Abstract:** The linear scale-space kernel is a Gaussian or Poisson function. These functions were chosen based on several axioms. This representation creates a good base for visualization when there is no information (in advanced) about which scales are more important. These kernels have some deficiencies, as an example, its support region goes from minus to plus infinite. In order to solve these issues several others scale-space kernels have been proposed. In this paper we present a novel method to create scale-space kernels from one-dimensional wavelet functions. In order to do so, we show the scale-space and wavelet fundamental equations and then the relationship between them. We also describe three different methods to generate two-dimensional functions from one-dimensional functions. Then we show results got from scale-space blob detector using the original and two new scale-space bases (Haar and Bi-ortogonal 4.4), and a comparison between the edges detected using the Gaussian kernel and Haar kernel for a noisy image. Finally we show a comparison between the scale space Haar edge detector and the Canny edge detector for an image with one known square in it, for that case we show the Mean Square Error (MSE) of the edges detected with both algorithms.

Index Terms—Kernel, Scale space, Wavelets.

---

<sup>1</sup> Lawrence Berkeley National Laboratory. One Cyclotron Road, MS50A6134, Berkeley, CA, USA.  
{[ddoering@lbl.gov](mailto:ddoering@lbl.gov)}

<sup>2</sup> DELET – EE – UFRGS. Av. Osvaldo Aranha, 103, sala 206. Centro Porto Alegre, RS. Brasil.  
{[schuck@ufrgs.br](mailto:schuck@ufrgs.br)}

## 1 Introduction

There are several different multi-scale signal processing techniques such as quad-tree [1], pyramid [1], wavelets [2][3] and scale-space [4]. The linear scale-space, for example, defined through its axioms has the Gaussian (and Poisson) as its kernel function, although several authors have been discussing if the Gaussian function is the best one for the scale-space. In order to answer this question it is necessary to compare the results from the Gaussian scale-space with other scale-spaces.

The scale-space theory was proposed by Witking [4] and expanded to the bi-dimensional case by Koenderink [5]. This theory has been largely used to extract information from images. Some examples are edge detection [6], ridge detection [7], structure extraction [8, 9] and movement detection [10]. The scale space created from the convolution of an image with a Gaussian or Poisson function have the following properties [6][7][11]: (i) causality, (ii) isotropy, (iii) homogeneity, (iv) non creation of local extrema, (v) non enhancement of local extrema, (vi) semi-group structure and (vii) infinitesimal generator. Florack et al. [12] say that these are good properties for image representation where there is no previous knowledge about the image because they work equally in any direction of the image. But in applications where one wants to emphasize a specific direction the Gaussian (or Poisson) function may not be the best choice. Another issue of the Gaussian scale-space, as mentioned by Remaki and Cheriet [13], is the length of its kernel that goes from minus to plus infinite and this implies the necessity of the use of large masks. Based on these limitations several authors have been searching for new scale-space kernels.

Another multi scale theory widely used is the Wavelet Transform [14]. In this work the wavelet transform of a signal is defined as being the internal product between the signal and the mother wavelet function in a specific scale and shifted by some factor. In the definition of the wavelet transform it is not defined a mother wavelet function allowing this transform to be very flexible. The most important conditions for these functions (wavelets) are admissibility and regularity. Due to the former condition the frequency component of the function at frequency zero must be zero. And due to the latter condition the wavelet function must be local in both time and frequency domains. These two conditions together imply that a wavelet function must be a band pass filter. Due to the enormous flexibility of the wavelet transform it became a general case for several others transforms, such as the Canny edge detector [15] and the scale-space theory [4]. The scale-space is a particular case of the wavelet transform when the derivatives of the Gaussian functions are used to extract information [16].

The relationship between the wavelets and the scale-space that were demonstrated by Mallat [16] haven't been used to solve digital image processing tasks up to now. One of the challenges dealt by the scale-space community is the definition of new kernels that are able to overcome the Gaussian limitations. In this paper (i) we will use the scale-space theory, (ii)

the Wavelet theory and (iii) the relationship between them in order to create a novel method to generate scale-space kernels from the wavelet mother functions. With this we want to enlarge the discussions about the scale space kernels.

## 2 Theoretical Fundamentals

### 2.1 The Scale Space

The scale-space representation of an image is a set of images that represent the original image in different scales. Mathematically, given a bi-dimensional function  $f(x, y) \in \mathfrak{R}^2 \rightarrow \mathfrak{R}$ , its scale space representation  $L(x, y, t) \in \mathfrak{R}^2 \times \mathfrak{R}_+ \rightarrow \mathfrak{R}$  is:

$$L(x, y, t) = f(x, y) * g(x, y, t) \tag{1}$$

where  $t$  is the scale,  $*$  is the convolution operator and  $g(x, y, t) \in \mathfrak{R}^2 \times \mathfrak{R}_+ \rightarrow \mathfrak{R}$  is the Gaussian kernel defined as:

$$g(x, y, t) = \frac{1}{(2\pi)} e^{-\frac{(x^2+y^2)}{2t}} \tag{2}$$

where the scale ( $t$ ) is the variance ( $\sigma^2$ ) of the Gaussian function.

#### 2.1.1 Scale-space blob detector

When a gray scale image is represented as a tri-dimensional surface its objects will become blobs [8]. There are two types of blob detectors, (1) one that has clear object in a dark background and (2) one that has dark objects in a clear background. The blob detector [17, 18] is defined using the image Laplacian:

$$\nabla^2 L = L_{xx} + L_{yy} \tag{3}$$

where  $\nabla^2 L$  is the image Laplacian,  $L_{xx}$  and  $L_{yy}$  are the second derivative of  $L$  in the  $x$  and  $y$  directions respectively. Images with clear regions and dark background, for example, have their blob detector operator defined as follows:

$$\begin{aligned} I(x, y) &= 0, & \nabla^2 L &> 0 \\ I(x, y) &= 1, & \nabla^2 L &\leq 0 \end{aligned} \tag{4}$$

where  $I$  is a binary image, where  $0$  = black= background and  $1$  = white = blob.

#### 2.1.2 Scale space edge detector

Lindeberg [19] proposed the detection of edges based on the second and third derivatives of the scale space image using the  $(u, v)$  coordinate system. In this system  $v$  is the direction parallel to the gradient direction and the  $u$  direction is perpendicular to the gradient direction. Then, a pixel will be an edge pixel if the next two conditions are met:

$$L_{vv} = L_x^2 L_{xx} + 2L_x L_y L_{xy} + L_y^2 L_{yy} = 0 \tag{5}$$

$$L_{vvv} = L_x^3 L_{xxx} + 3L_x^2 L_y L_{xxy} + 3L_x L_y^2 L_{xyy} + L_y^3 L_{yyy} < 0 \tag{6}$$

where  $L_{vv}$  and  $L_{vvv}$  are the second and third derivatives of  $L$  in the  $v$  direction.

## 2.2 Wavelet Theory

The wavelet transform is an efficient tool for local analysis of non-stationary and transient signals [20]. The wavelet functions are created by scaling and shifting the mother wavelet. For a function to be a wavelet function it must have the admissibility and regularity conditions. This means that these functions must be local in the time (or space) and frequency (or scale) domains.

The wavelet transform can be implemented in two different ways, the Continuous Wavelet Transform (CWT) or the Discrete Wavelet Transform (DWT). In this paper we are going to use the CWT definitions because the scale space is also continuous and then it is possible to relate these two theories.

### 2.2.1 Bi-dimensional Continuous Wavelet Transform (CWT2D)

The CWT can be extended to two or more dimensions and keep the same properties of the 1D case [21]. Given a bi-dimensional signal (2D), for example an image  $f(x,y)$ , its CWT2D for a wavelet function  $\psi$  is:

$$w2D(s, \theta, b_x, b_y) = \frac{1}{s} \left\langle \psi \left( \frac{r_\theta(x - b_x, y - b_y)}{s} \right), f(x, y) \right\rangle = \frac{1}{s} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \psi \left( \frac{r_\theta(x - b_x, y - b_y)}{s} \right)^* f(x, y) dx dy \tag{7}$$

where  $\langle \rangle$  is the internal product,  $\psi$  is a bi-dimensional wavelet, shifted by  $b_x$  in the  $x$  direction and  $b_y$  in the  $y$  direction, scaled by  $s$  ( $s > 0$ ) and rotated by  $\theta$  ( $0 \leq \theta < 2\pi$ ) where  $r_\theta$  is a rotation matrix. For (7) to hold it is necessary that the image  $f(x,y)$  needs to be squared integrable and defined in the plane  $\mathfrak{R}^2$ , or  $f(x, y) \in L^2$  (or in other words, have finite energy). For  $\psi$  be a mother wavelet, it must satisfy the admissibility and regularity conditions as mentioned before. The CWT2D defined by (7) has four parameters ( $s, \theta, b_x, b_y$ ) [22]. In order to be possible to visualize this transform it is necessary to lock some of them. There are several different combinations but the combinations where  $(b_x, b_y)$  or  $(s, \theta)$  are kept locked have been more used because it's physical interpretation.

### 2.2.2 The mother wavelet families

The mother wavelets can be defined in different ways. One option is using its equation in the time (space) domain. Another option is to use the function properties in the frequency domain. As examples of mother wavelets, there are those that are defined in the

Matlab version 5.3: Morlet; Mexican hat; Meyer; Haar; Daubechies; Symlets; Coiflets; Splines; bi-orthogonal.

### 2.3 Relationships between the Scale Space and the Continuous Wavelet Transform

It was shown by Mallat [16] that the scale space is a special case of the wavelet transform. In that work, besides proving that these two transforms are related, the author also defined a method to generate new wavelets from a smoothing function. This method is described as follows:

A smoothing function is defined as:

$$\theta_s(x) = \frac{1}{s} \theta\left(\frac{x}{s}\right) \quad (14)$$

The wavelet functions generated from the smoothing function are defined as:

$$\psi^1(x) = \frac{d\theta(x)}{dx} \quad (15)$$

$$\psi^2(x) = \frac{d^2\theta(x)}{dx^2} \quad (16)$$

In that work Mallat used the bi-dimensional Gaussian function as the smoothing function and he had generated two wavelets. In [23] Sheng claims that any derivative of order 'n' of the Gaussian function can be a wavelet. This allows us to expand the Malat's method to the 'n' order (for n > 0) and then the wavelets are defined as:

$$\psi^n(x) = \frac{d^n\theta(x)}{dx^n} \quad (17)$$

The scale space is a special case of the wavelet theory when the function  $\theta(s)$  is the Gaussian function.

## 3 The Method Proposed

In this section the method proposed to generate new kernels for the scale space from wavelet functions is defined, as well the results obtained. The functions needed to realize this task were developed using the software Matlab 5.3. The image processing was done in a Athlon XP 1.2 GHz with 256 Mb of RAM memory.

### 3.1 Method to generalize the scale space from different wavelet families

One of the objectives of this work is to generate the scale space and its operators from different wavelet families. The scale space operators are defined using the Gaussian derivatives, and this function works as a smoothing filter. The wavelet functions are band pass filters. So it is necessary to associate each wavelet function to a smoothing function and from this smoothing function it is possible to apply the scale space operator definition. In order to associate a smoothing function to each wavelet function it will be used the Mallat's method described in section 2.3 backwards. This means that we are going to start with the wavelet function ( $\psi^n(x)$ ) looking for a smoothing function ( $\theta_s(x)$ ), as will be shown later.

It is assumed that the wavelet function under test is a derivative of some unknown smoothing function, but it is unknown which derivative order is the wavelet from this smoothing function. Based on wavelet theory, it is known that the wavelet function is a band pass filter and the smoothing function is a low pass filter. Using these characteristics it is going to be determined which is the derivative order of the wavelet function (if exist such a scaling or smoothing function). To do that the following algorithm is used:

1. The wavelet is considered the n-order derivative of some smoothing function.
2. The wavelet is integrated.
3. The frequency response of the new function is obtained by means of FFT.
4. If the frequency response shows that the new function is a band bass function then steps 2 and 3 are repeated for the new function.
5. If the new function is a low pass filter then it is defined as the smoothing function that is associated to the wavelet function under test.
6. The derivative order is defined as being equal to the number of times that steps 2 and 3 were executed, or the number of times that the function was integrated.

To exemplify the use of this algorithm we will apply it to the Haar wavelet function. Figure 1(a) shows the Haar wavelet and Figure 1(b) shows its frequency response. From the frequency response one can see that this function is a band pass filter as expected because this is a wavelet function. Following the algorithm, on step 1 this function is considered the n-order of a smoothing function. On step 2 we integrate this function and the resulting function is shown in Figure 1(c). On step 3 we generate its frequency response and show it in Figure 1(d). Step 4 we evaluate the frequency response and conclude that this is not a band pass filter and then we proceed to step 5. In 5 we define this new function as being the smoothing function of the wavelet Haar because its frequency response is a low pass filter. Step 6 determines the order of the wavelet function as being 1 since the wavelet function was integrated once to get to the smoothing function.

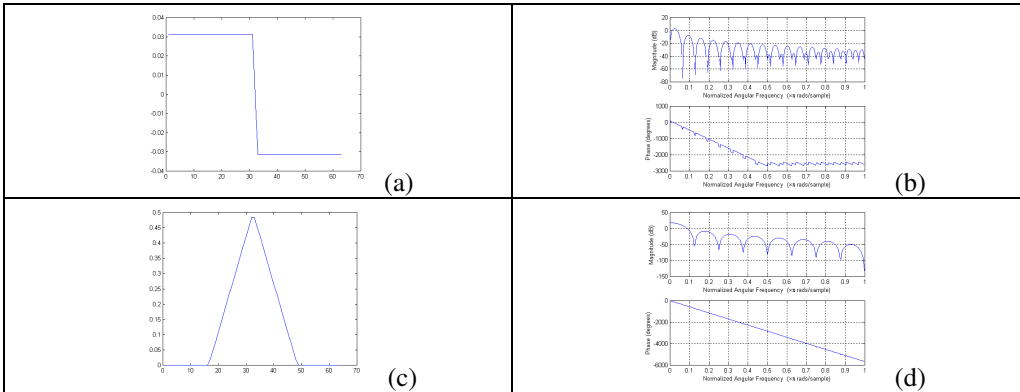


Figure 1- (a) wavelet Haar on scale 32 and (b) first integral of the wavelet Haar on scale 32, (c) magnitude and phase of the frequency response of the function wavelet Haar and (d) first integral of the wavelet Haar.

Once the smoothing functions are known and which is the derivative order of the wavelet function, it is possible then to apply scale space operator definition to them. As seen in the previous section the scale space operators are defined from the derivatives ( $L_x, L_{xx}, \dots$ ) of some smoothing function.

### 3.2 Choosing the mother wavelet function

The choice of the mother wavelet is of fundamental importance to the analyses of a given signal when this transformation is used. This is independent of the signal dimension (1D, 2D, ...). Antoine [21] defines some options to solve this task. The first of them is to choose the mother wavelet that gives the best result for a given problem. The second option is to create a new mother wavelet for the application. The third option is to use some wavelet families that have their behavior and definition known, at least for the 1D case. This work uses the third option to choose the mother wavelet added of some requirements necessary to allow the comparison between the wavelet and the scale space. These conditions are postulated as follows: (1) the existence of a smoothing function created by integrating the wavelet function  $n$  times; (2) the smoothing function is real and symmetric; (3) the smoothing functions derivatives must follow the properties of the wavelet transform (admissibility and the regularity conditions).

It is important to remark that this work does not have the intention to prove the necessity or sufficiency of the postulates stated above because it is out of its scope. The wavelet functions that don't have such characteristics will not be used in this work. The list of functions that fulfill the postulated above are: the Gaussian derivatives (Mexican Hat); Haar; Bi-ortogonal 1.3; Bi-ortogonal 1.5; Bi-ortogonal 2.2; Bi-ortogonal 2.4; Bi-ortogonal 2.6; Bi-ortogonal 2.8; Bi-ortogonal 4.4; Bi-ortogonal 5.5; Bi-ortogonal 6.8.

The results got for each one of these functions using the algorithm explained in section 3.1 are summarized in Table 1.

Table 1: Summary of the features that the wavelet functions have when they were tested with the algorithm defined in section 4.3.

Function name	Function Order*
Bi-orthogonal 1.3	1
Bi-orthogonal 1.5	1
Bi-orthogonal 2.2	2
Bi-orthogonal 2.4	2
Bi-orthogonal 2.6	2
Bi-orthogonal 2.8	2
Bi-orthogonal 4.4	4
Bi-orthogonal 5.5	6
Bi-orthogonal 6.8	6
Haar	1
Mexican Hat	2

\*number of times that the function was integrated in order to get a smoothing function.

### 3.3 Generation of bi-dimensional functions from one-dimensional functions

Some of the functions mentioned on the last section don't have a mathematical definition in the time (space) domain for the bi-dimensional case. It was used in this work three different ways to solve this task.

The first method to generate the bi-dimensional functions uses a given function  $y = f(x)$ , symmetric and the bi-dimensional function is generated by rotating the uni-dimensional function. This rotation is done numerically and the function is rotated in relation to the z axis. Figure 2 shows the one-dimensional and bi-dimensional Gaussian function generated using this method.

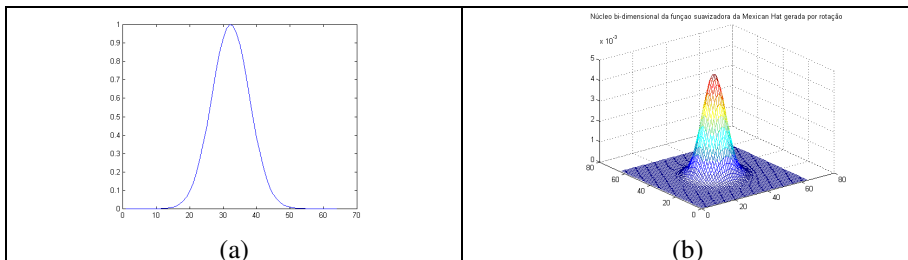


Figure 2 – (a) one-dimensional and (b) bi-dimensional gaussian function generated by rotation of (a).



The second method used to generate a bi-dimensional function from a one-dimensional function receives one function  $f(x)$  and one function  $f(y)$ , both symmetric to some point and then generate a bi-dimensional function using tensor product. The tensor product between these two functions  $f(x)$  and  $f(y)$  is defined as:

$$f(x,y) = f(x) \cdot f(y) \quad (18)$$

The third method used to generate a bi-dimensional function from a one-dimensional function receives a given function  $y = f(x)$ , symmetric to some point and generates a bi-dimensional function by convolution. The convolution between these two functions is defined as follows:

$$F(x,y) = f(x) * f(y) \quad (19)$$

where  $*$  is the convolution operator.

All the three methods shown before generate a bi-dimensional Gaussian function identical to the function generated by the mathematical bi-dimensional function definition. This demonstrates that all of them could be used. Although, when these methods are used for other functions other than the Gaussian the bi-dimensional function that results may vary for each method. In this work it is going to be shown only the results got using the bi-dimensional functions generated by rotation because the operators that result from this method are symmetric and this makes them rotational invariant. This is a feature that exists in the Gaussian scale space and it will be preserved in this work.

### 3.4 Examples of the new scale-space kernels use.

The scale space representation of an image can be done for the different scale spaces using different smoothing function as well as for different methods of generating bi-dimensional functions. One example is the Lena image representation. Figure 3 shows a scale space representation using the Haar's smoothing function for scales 2, 8 and 16. Figure 4 shows the blobs of the Lena image using the Haar's smoothing function for scales 2, 8 and 16. Figure 5 shows the edges of the Lena image using the scale space edge detector and the Haar's smoothing function for the scales 2, 8, and 16. These images provide good examples to show the different scale space features in a qualitative way.



Figure 3 – Scale space representation of the Lena image using the Haar smoothing function, created by convolution on scales 2 (a), 8 (b) and 16 (c).

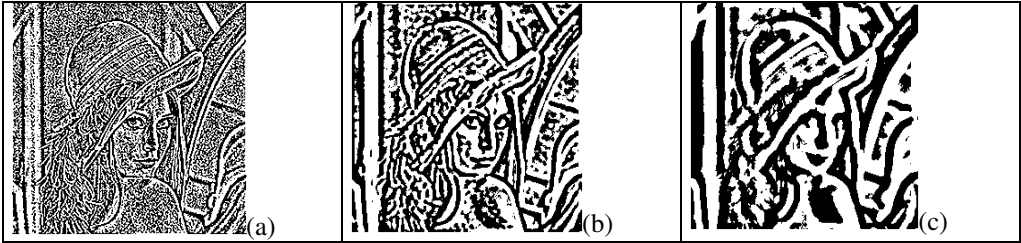


Figure 4 – Blobs from the Lena image, when using the scale space blob detector and the Haar smoothing function, created by rotation, on scales 2 (a), 8 (b) and 16 (c).

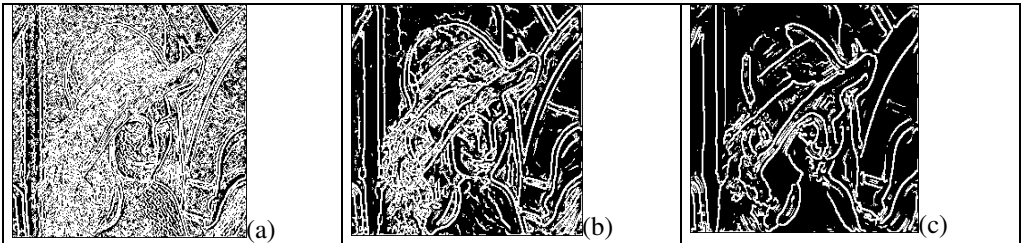


Figure 5 – Edges from the Lena image, when using the scale space blob detector and the Haar smoothing function, created by rotation, on scales 2 (a), 8 (b) and 16 (c).

In order to analyze some of the new scale spaces properties independently it was used an artificial image called phantom1 (Figure 6a). This image has objects with different shapes and sizes but the same contrast. This allows us to observe the behavior of the scale space operator in different scales. To understand the behavior of the different kernels with respect to noise a second phantom image was created (Figure 6b). This phantom besides objects of different intensities has noise added to it. The noisy image is defined as:

$$\hat{I}(x, y) = f(x, y) + c.n(x, y) \tag{20}$$

where  $\hat{I}(x,y)$  is the noisy image,  $f(x,y)$  is the original image,  $n(x,y)$  is the normal noise with standard deviation set to one and  $c$  is a constant that multiplies the noise. This constant was arbitrarily chosen to 8. In a future work, it will be presented the robustness of detection algorithms with noise level in images using the technique here presented.

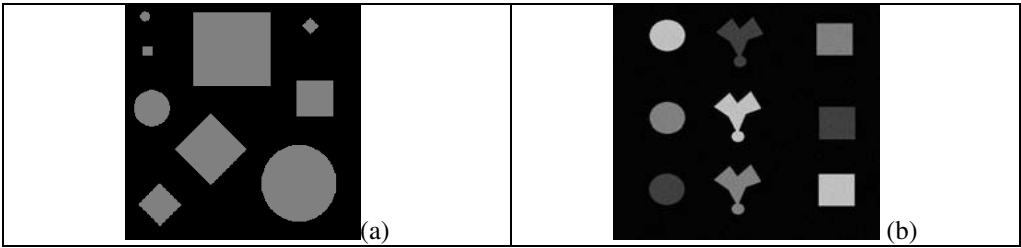


Figure 6 – Artificial images called (a) *phantom1* and (b) *phantom2* (b).

Figure 7 shows the *phantom1* image's scale space blobs using the Haar smoothing function for scales 2, 54 and 80. Figure 8 shows the edges of the *phantom1* image using the scale space edge detector for the Haar smoothing function for scales 2, 8 and 16. Figure 9 shows the blobs of the *phantom1* image using the smoothing function of the wavelet Bi-orthogonal 4.4 for scales 2, 26 and 72. Figure 10 shows the edges of the *phantom1* image using the scale space edge detector and the smoothing function of the wavelet Bi-orthogonal for scales 1, 4 and 16.

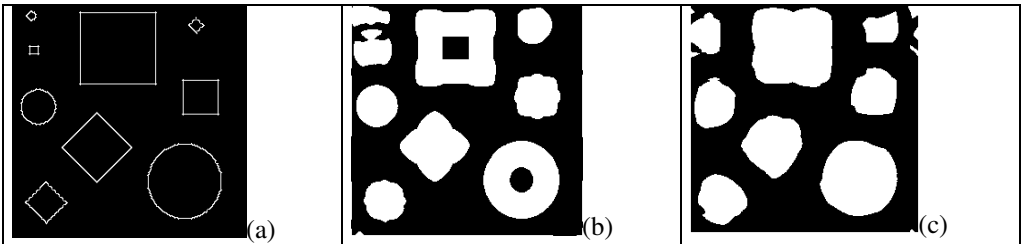


Figure 7 – Blobs from the *phantom1* image, when using the scale space blob detector and the Haar smoothing function, created by rotation, on scales 2 (a), 54 (b) and 80 (c).

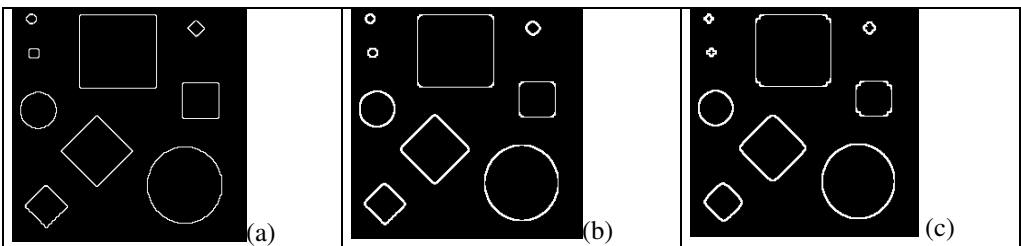


Figure 8 – Edges from the *phantom1* image, when using the scale space blob detector and the Haar smoothing function, created by rotation, on scales 2 (a), 8 (b), and 16 (c).

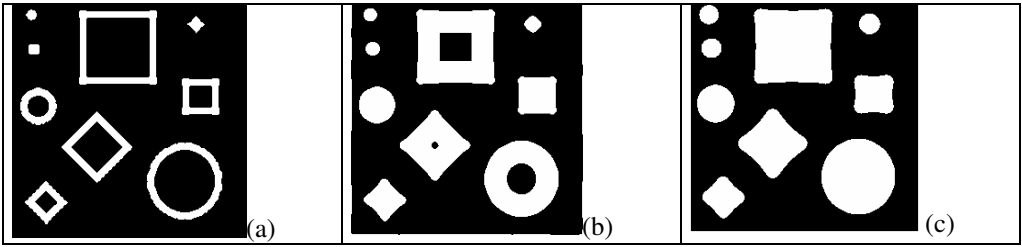


Figure 9 – Blobs from the *phantom1* image, when using the scale space blob detector and the Bi-orthogonal 4.4 smoothing functions, created by rotation, on scales 2 (a), 26 (b) and 72 (c).

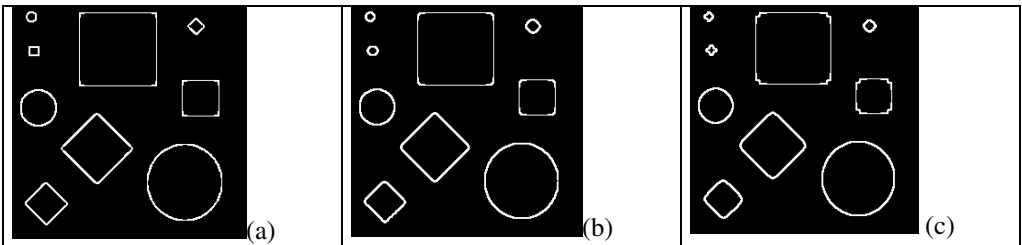


Figure 10 – Edges from the *phantom1* image, when using the scale space blob detector and the Bi-orthogonal 4.4 smoothing function, created by rotation, on scales 1 (a), 4 (b) and 16 (c).

Figure 11 shows the detected edges of the noisy *phantom2* image (Figure 6(a)) using the Gaussian smoothing function, for scales 2 (a), 16 (b) and 32 (c). Figure 12 shows the detected edges of the noisy *phantom2* image using the smoothing function of the Haar wavelet, for scales 2 (a), 16 (b) and 32 (c).

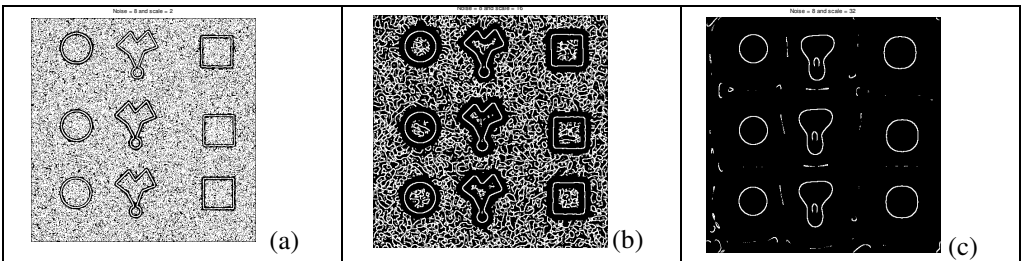


Figure 11 – Edges detected from the *phantom2* image when using the Gaussian scale space on scales 2 (a) 16 (b), e 32 (c).

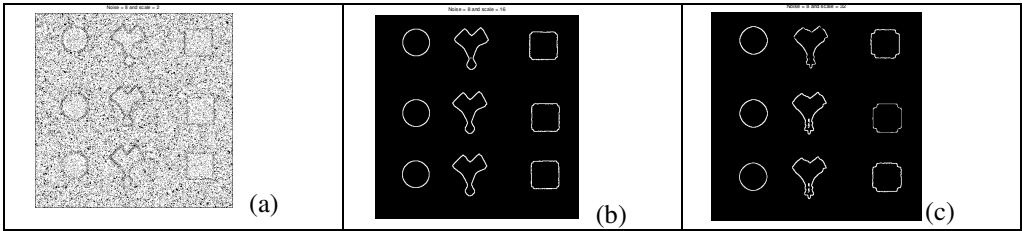


Figure 12 – Edges detected from the *phantom2* image when using the Haar scale space on scales 2 (a), 16 (b), e 32 (c).

In order to create a quantitative comparison between the scale space edge detector using the Haar kernel and the Canny edge detector we create a phantom with a white box in a black background (phantom 3). The edges of this phantom are known and this allowed us to calculate the mean square error (MSE) for the edges detected with both algorithms while adding different random noise levels. Figure 13 (a) shows the MSE for the Canny edge detector and Scale Space Haar edge detector (at scale set to 12) for the phantom 3 using different noise levels. Figure 13 (b) shows the number of edge points detected using the Canny edge detector and Scale Space Haar edge detector (at scale set to 12) for a phantom3 using different noise levels. These examples show that the scale space algorithm has a superior performance over the Canny edge detector for noisy images.

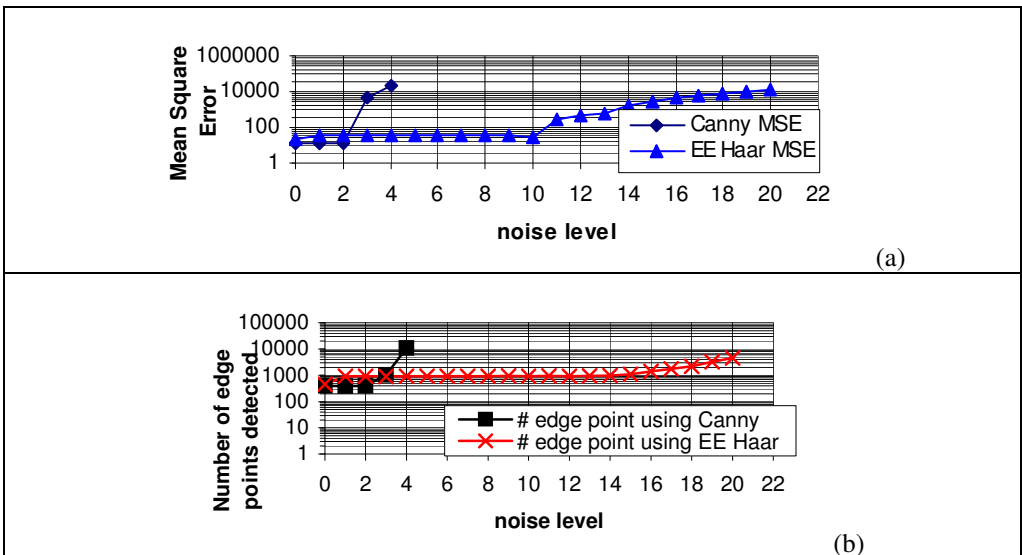


Figure 13 – Mean square error for the Canny edge detector and Scale Space Haar edge detector (at scale set to 12) for a phantom versus noise level (a). Number of points detected using the Canny edge detector and Scale Space Haar edge detector (at scale set to 12) for a phantom versus noise level (b).

## 4 Discussion and Conclusions

One of the scale space aspects that have been widely discussed by several researchers is its kernel (originally the Gaussian function). The objective of these discussions is to know if the Gaussian is the best function or in which cases it is the best one. Besides that the researches are developing new kernels with new or different features. Once these new kernels are known it is possible to compare the scale space generated with them against the Gaussian scale space and then find out which one gets the best results for different cases of study. In this work, it was used (1) the scale space theory, (2) the wavelet theory and (3) the relationships between them to develop a new method to create new scale space kernels from wavelet functions.

In the proposed method, we start from a known wavelet function. Then we find the smoothing function that is associated to this wavelet through numerical integration and analyses of its frequency domain response. Once the smoothing function is available it is used the original definition of the scale space operators with this new function. This work shows the results from the scale space blob and edge detectors. The only thing that was changed was the kernel.

According with the wavelet transform theory the wavelet function is band pass function ( $\psi$ ). Some of these functions have a low pass function associated ( $\phi$ ). It is important to note that the smoothing function is NOT this low pass function ( $\phi$ ) and it actually is a function generated through the  $n$  times integration of the wavelet function. For example, the Mexican Hat function does have a smoothing function (the Gaussian) but it does not have a  $\phi$  function.

The proposed method limits the set of wavelet functions that are used with it based on some postulates. These postulates were defined in such a way that once we use a wavelet function with the algorithm it must be possible to get a new function that has a low pass filter behavior. Besides this only symmetric wavelet functions were used.

The wavelet functions that were studied in this work are the functions that already exist in Matlab version 5.3. Among them the wavelet functions Bi-orthogonal 1.3, Bi-orthogonal 1.5, Bi-orthogonal 2.2, Bi-orthogonal 2.4, Bi-orthogonal 2.6, Bi-orthogonal 2.8, Bi-orthogonal 4.4, Bi-orthogonal 5.5, Bi-orthogonal 6.8, Haar and Mexican Hat are the ones that have the features required by the postulated defined in this work. Those functions were used with the proposed method to generate new scale space kernels.

Some of the wavelets that were used in this work don't have definition for the 2D case. Due to that it was used some methods to create 2D wavelets from 1D wavelets. Three different methods to solve this problem were used and they are rotation, tensor product and convolution. When these methods were used with the 1D Gaussian function the results match the 2D Gaussian function defined analytically. This indicates that any one of them could be

used. But the rotation method generates 2D functions that are always symmetric and because of it this was the chosen method.

After we have the wavelet function, a method to create scale space kernels and 2D functions, it was possible to generate the scale space associated with each function. All kernels produced are local and rotational invariant.

The Haar wavelet function used in this work was defined from minus 0.5 to plus 0.5 because this makes it symmetric in relation to zero. As this function is defined inside a limited range it is correct to say that it has compact support. The Gaussian function, for example, is defined from minus to plus infinite. In order to apply the latter filter in a signal it is necessary to truncate it at some point. Values that are based on the standard deviation, such as one, two or three standard deviations are usually chosen. Because this filter is truncate, the result of filtering an image with it is an approximation of the continuous case. As bigger the point where the function is truncated the better the approximation is. On the other hand the bigger this value is then the bigger is the processing cost to solve this task. An important advantage for the Haar kernel in relation to the Gaussian is that it is completely defined in a given range therefore it doesn't need to be truncated. Besides that the total computational cost to use this kernel will also be lower than the Gaussian. Kernels with compact support have been developed as shown in [16].

The Haar wavelet function is usually used to detect discontinuities. For images these discontinuities are probably edges of the objects. When the smoothing function of the Haar function was used with the scale space edge detector this behavior was preserved, even for large scale as shown in Figure 5. For this case the biggest distortions had happened at the object's corners. It was also shown the edges detected with the scale space edge detector using the smoothing function of the wavelet Haar and Gaussian. The noise degradation of the edges was far less when using the smoothing function of the Haar wavelet than when using the Gaussian smoothing function. These results are examples that each kernel has its own features, and then the discussion about which is the best kernel for each application is extremely valid.

The wavelet transform has several kernels and it is necessary to choose which kernel will be used in a given application. The scale space also has several kernels and it is also necessary to choose which kernel to use in a given application. Antoine [21] defines some options to solve this task, as described in section 3.2. The authors suggest this method to be used for the scale space theory too. The main contribution of this work for the task of choosing the best scale space kernel is the method to generate new kernels and the ten new kernels described earlier. The Mexican hat wavelet generated the Gaussian smoothing function and it was used to validate the algorithm that generates new scale space kernels from wavelet functions and it is not considered a new kernel. With that there is now a bigger set of kernels that can be used to find out which one is the best for each application.

We also compared the results got using the Canny edge detector with the scale space Haar edge detector. The result of the MSE showed us that for low noise images the Canny algorithm has a better performance. But when we added higher noise levels the scale space Haar edge detector outperform the Canny algorithm. This shows that for such noisy images the scale space Haar edge detector is a better option. The number of edge points detected also show us that the Canny edge detector has a better performance when the noise level on the image is low while the scale space edge detector has a better performance for noisy images.

In this work it was proposed a new method to create new scale space kernels from wavelet functions. Some of the challenges in this field were solved, but there are several others that are still open and they are suggested for future work. Some suggestions are:

- Implement ridge detector;
- Implement scale-spaces with non-symmetric bases;
- Verify for the new kernels which axioms they satisfy;
- Test for detection robustness of noisy images.

## 6 Acknowledgment

Dionísio Doering thanks CAPES for his scholarship.

## 7 References

- [1] Lindeberg, T., "Linear scale-space: Basic theory", Kluwer Academic Publishers, 1994.
- [2] Daubechies, I., "Ten Lectures on Wavelets", Society for Industrial and Applied Mathematics (SIAM), 1992.
- [3] A Wavelet Tour of Signal Processing, Second Edition: Wavelet Analysis & Its Applications (Hardcover); Academic Press; 2nd edition September 15, 1999.
- [4] Witkin, A. P., Scale-space Filtering. In Proc. 8<sup>th</sup> Int. Joint. Conf. Art. Intel. Pages 1019-1022. Karlsruhe, West Germany, 1983.
- [5] Koendering, J.J., "The Structure of Images", Biol. Cybern., 1984.
- [6] Lindeberg, T., "Scale-Space: A Framework for handling image structures at multiple scales", Proc. CERN School of Computing Netherlands, September, 1996.
- [7] Cunha, A. M., "Espaço de Escala e Detecção de Arestas", outubro de 2000.
- [8] Lindeberg, T., Eklundh, J. "Scale-space primal sketch: construction and experiments", Image and Vision Computing, Volume 10, Issue 1, January-February 1992, Pages 3-18.
- [9] ZUCCOLOTTO, M., OLABARRIAGA, S., SCHUCK JR., A., "Segmentação de cavidades em imagens cardiográficas utilizando Espaço de escalas", IICLAEB, 2002.
- [10] Laptév, I., Lindeberg, T., "Velocity adaptation of spatio-temporal receptive fields for direct recognition of activities: an experimental study", Image and Vision Computing, 22, pp.:105-116, 2004.
- [11] Bosworth, J. H., Acton, S. T., "Morphological scale-space in image processing", Digital Image processing, 2003, 338-367.
- [12] Florack, L M J, Salden, A H, ter Haar Romeny, B M, Koenderink, J J, and Viergever, M A, "Nonlinear scale-space", Image and Vision Computing Volume 13 Number 4 May 1995.
- [13] Remaki, L., Cheriet, M., "Visual Data Extraction From Bi-Level Document Images Using a Generalized Kernel Family with Compact Support, in Scale-Space", Proceedings of the 5<sup>th</sup> International Conference on Document Analysis and Recognition - IEEE Conference, pp. 609-612, 1999.
- [14] Grossman, A, Morlet, J. , Decomposition of Hard Functions into square integrable wavelets of constant shape, SIAM, 1984.



- [15] Canny, J., "A Computational approach to edge detection", IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 8, pp. 679-698, 1986.
- [16] Mallat, S., Hwang, W. L., "Singularity Detection and Processing with Wavelets", IEEE Transactions on Information Theory, Vol.38, N°2, 1992.
- [17] Lindeberg: "Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention", International Journal of Computer Vision, vol. 11, pp. 283--318, Dec. 1993.
- [18] Gonzales, R. C., Woods, R. E., *Processamento de Imagens digitais*, 1992.
- [19] Lindeberg, T., "Edge Detection and Ridge Detection with Automatic Scale Selection", IEEE, 1996
- [20] Zhang, L., Lenders, P., Locating the Head Boundary with 2D Continuous Wavelet, International Symposium on Intelligent Multimedia, 2001.
- [21] Antoine, J. P., "Image analysis with two-dimensional continuous wavelet transform", Signal Processing, 1993.
- [22] Antoine, J. P., "Image analysis with two-dimensional continuous wavelet transform", Proc. of Int. Conf., France, 1989.
- [23] Sheng, Y., "Wavelet Transform", The Transform and Applications Handbook, IEEE Press, 1996.