Image Denoising and Edge Detection Method Using Least Squares Support Vector Machine

Jo Sok Chol *, Ryang Chol Sik, Choe Yu Song, Kim Hyok Jin, U Hyok Chol

Faculty of Mathematics, **Kim Il Sung** University,
Taesong District, Pyongyang, DPR Korea
* Corresponding author Email addresses : SC.Jo@star-co.net.kp

Abstract: This paper proposes a novel denoising and edge detection algorithms for image using least squares support vector machine (LS-SVM) with Gaussian radial basis functions (RBF) kernel. The new filter, called least squares support vector machine filter (LS-SVMF) for image denoising, is based on the general concept of binary filters and machine learning theory. Using the LS-SVM, a set of the new gradient operators and the corresponding second derivative operators are obtained. Computer experiments are carried out for denoising and extracting edge information from real images. The results obtained for the applications show that the proposed algorithms outperform many other existing methods in the image denoising task and the traditional edge detectors. The proposed algorithms can be successfully applied for the processing of images corrupted with impulsive noise while maintaining the visual quality and a low reconstruction error.

Keywords: *least squares support vector machine (LS-SVM), Gaussian radial basis functions (RBF), image denoising, edge detection, learning machine*

1. Introduction

Denoising and edge detection are a key issue in image processing, computer vision, and pattern recognition. Many filters showed a great performance in signal, image and video processing tasks when the underlying noise follows a heavy-tail distribution. In particular, Boolean filters are a class of nonlinear filters widely used that encompass stack filters, weighted median filters, among other filters [1]. The success of these filters depends on two intrinsic properties: edge preservation and efficient noise attenuation, being robust against impulsive noise [2]. In practice, human perception effects play an important role in determining whether an edge exists or not. A variety of algorithms have been proposed to analyze image intensity variation, including statistical methods, difference methods[3-7]. Recently the support vector machine(SVM), based on statistical learning theory, as a powerful new tool for data classification and function estimation, was developed [8]. SVM maps input data into a high dimensional feature space where it may become linearly separable. Recently SVM has been applied to various fields such as pattern recognition and object detection, function estimation etc. Especially, Suykens and Vandewalle, Suykens et al. proposed a modified version of SVM called least squares SVM (LS-SVM), which resulted in a set of linear equations instead of a quadratic programming problem, which can extend the application of the SVM to time series prediction, etc.[9]. In this paper, we take advantage of the classification capability of LS- SVM in a binary filtering process, replacing the Boolean function that characterizes the Boolean filter by a LS-SVM, defines a new family of nonlinear filters. This paper aims at exploring a new edge detection algorithm based on the LS-SVM. The proposed algorithms are compared to the others. Experimental results prove that the new approach is effective for image edge detection. The rest of this paper consists of the followings. Section 2 describes the SVM and LS-SVM. Section 3 represents the theory and algorithm for image denoising using the LS-SVM. Section 4 gives image edge detection method using the LS-SVM. Section 5 deals with the experimental results. Section 6 describes the conclusion.

2. SVM And LS-SVM

2.1 *SVM*

SVM principles were developed by Vapnik and presented in several works as in [8-11]. Given l training pairs (X_1, y_1), $(X_2, y_2), \cdots, (X_l, y_l)$ where $X_i \in \chi$ (χ is the subset of R^n) is an input vector labeled by $y_i \in \{+1, -1\}$ for $i = 1, 2, \cdots, l$, the SVM calculates a separating hyperplane, i.e., $W^T X + b = 0$, with the largest margin. This hyperplane for the linear SVM can be found by solving the following quadratic programming problem:

$$\min \ J(w, \xi) = \frac{1}{2}\|W\|^2 + c \cdot \sum_{i=1}^{l} \xi_i$$

$$\text{s.t.} \quad y_i(W^T X_i + b) \ge 1 - \xi_i, \quad \xi_i \ge 0 \quad \forall i = 1, 2, \cdots, l \tag{1}$$

where c is the parameter denoting the trade-off between the margin width and the training data error and $\xi_i \ge 0$ are slack variables. Problem (1) is usually posed in its Wolfe dual form with respect to Lagrange multipliers $\alpha_i \in [0, c]$, $i = 1, \cdots, l$, which can be solved by standard quadratic optimization packages. The bias b can easily be calculated from any margin support vector X_i satisfying $0 < \alpha_i < c$. The decision function is, therefore, given by

$$f(X) = \text{sgn}(W^T X + b) = \text{sgn}(\sum_{i=1}^{l} a_i y_i X_i^T X + b) \tag{2}$$

Those training vectors corresponding to Lagrange multipliers α_i greater than zero are called SVs, as $f(X)$ depends on them exclusively. For some problems, improved classification can be achieved using nonlinear SVMs. The basic idea of nonlinear SVMs is to map data vectors from the input space to a high-dimensional kernel space using a nonlinear mapping ϕ, and then perform pattern classification using linear SVMs in the kernel space. Most of the commonly used kernel functions in SVMs have the properties of pd or cpd, which leads to the global optimum of the solution of SVMs being fully guaranteed from a theoretical point of view [8]. For an unknown input pattern X, we have the following decision function:

$$f(X)=\mathrm{sgn}(\sum_{i=1}^{l} a_i y_i K(X_i\ ,\ X)+b) \tag{3}$$

2.2 *LS-SVM*

Based on statistical learning and the structural risk minimization principle, SVM is closely related to both expectation risk and generalization performance. On the basis of classical SVM, Sukens and Vandewalle presented LS-SVM approach, where the constraints of inequalities in the classical SVM approach are replaced by equality constraints. In addition, the LS-SVM solution can be found directly from solving a set of linear equations instead of quadratic programming [12-15].

Given a training set $\{(X_i\ ,\ y_l)\}_{i=1}^{N}$, with input data $X_i \in R^N$ and classification labels $y_i \in \{+1,\ -1\}$, according to Suykens and vandewalle and Suykens et al., a classifier in the primal space can be written as follows

$$f(X)=W^T\varphi(X)+b \tag{4}$$

The LS-SVM can be expressed as follows

$$\min_{\mathrm{w,b,e}} J(w,e)=\frac{1}{2}\|W\|^2+\gamma\cdot\frac{1}{2}\cdot\sum_{i=1}^{N}e_i^2 \tag{5}$$

Constraints

$$W^T\varphi(X_i)+b=y_i-e_i\ , \qquad i=1,2,\cdots,N \tag{6}$$

where $W \in R^N$ and the nonlinear function $\varphi(\cdot) : R^M \rightarrow R^N$ is usually introduced by a kernel function that maps the input space to a high-dimensional feature space.

However, it needs not to evaluate W and $\varphi(\cdot)$ explicitly in the LS-SVM framework. By using the Lagrangian multiplier optimization method, the solution to the minimization problem in the primal space can be obtained by solving the following linear system:

$$\begin{bmatrix} 0 & 1_N^T \\ 1_N & K+\frac{1}{\gamma}I_N \end{bmatrix}\begin{bmatrix} b \\ a \end{bmatrix}=\begin{bmatrix} 0 \\ y \end{bmatrix} \tag{7}$$

where $y=[y_1,\ y_2,\cdots,y_N]$, $1_N=[1,1,\cdots,1]^T$, $a=(a_1,a_2,\cdots,a_N)^T$, γ denotes regularization parameter and

$K=\varphi(X_i)^T\varphi(X_j)=K(X_i\ ,X_j)$.

The LS-SVM classifier is given as

$$f(X)=\mathrm{sgn}(\sum_{i=1}^{N} a_i y_i K(X_i\ ,\ X)+b) \tag{8}$$

Where a,b are the solutions to Eq. (7).

3. IMAGE DENOISING USING LS-SVM

3.1 *Least Squares Support Vector Machine Filter*

Let $Z=[Z_1,\ Z_2,\cdots,Z_N]$ be the observation vector to be filter ed, furthermore, let $Z^m=[\ z_1^m,\ z_2^m,\cdots,z_N^m]$ be its correspondent threshold decomposition at threshold level m. The output of the Support Vector Machine filter, denoted by LS-SVMF (Z), is defined as:

$$LS-SVMF(\ Z_1,\cdots,\ Z_N\ )=\frac{1}{2}\sum_{m=-M+1}^{M}LS-SVMF_f(z_1^m,z_2^m,\cdots,z_N^m) \tag{9}$$

where $LS-SVMF_f(\bullet):\{-1,\ 1\}^N \rightarrow \{-1,\ 1\}$, is a decision function corresponding to a LS-SVM as shown in Eq.(8) and $Z^m=T^m(X)$, is the threshold decomposition of input vector Z.

Substituting $LS-SVMF_f(\bullet)$ for the decision structure from Eq. (8), Eq.(9) can be reduced as follows

$$LS-SVMF(Z)=\frac{1}{2}\sum_{m=-M+1}^{M}(\mathrm{sgn}(\sum_{i=1}^{N}y_i a_i\cdot K(Z^m,\ Z_i)+b)) \tag{10}$$

It seems that the computational cost of Eq.(10) increases. This, however, can be remarkably reduced if for any

$m\in(-\infty,\ Z(1)]$, $m\in(Z(i-1),\ Z(i)]$, $i=2,\cdots,N$, $m\in(Z(N),+\infty)$,

threshold decomposition outputs the same binary vectors.

Therefore, there are at least N+1 of different binary vectors, Z^m. From the above, Eq. (10) can be reduced as follows

$$LS-SVMF(Z)= \\ =\frac{Z(1)+Z(N)}{2}+\frac{1}{2}\sum_{j=2}^{N}(Z(j)-Z(j-1))(\mathrm{sgn}(\sum_{i=1}^{N}a_i y_i K(Z^m,Z_i)+b)) \tag{11}$$

where $Z(i)$ is the i -th smallest sample of the set $\{Z_1,\ Z_2,\cdots,Z_N\}$, with $Z(1)\le Z(2)\le\cdots\le Z(N)$. The filter representation in Eq. (11) is an interpretation of the LS-SVM filter.

3.2 *Least Squares Support Vector Machine Filter algorithm*

We designed several 3x3 masks for generalizing some particular cases that may appear in a filtering process. To do this, we made a 9-component vector and used to train the LS-SVM defining the filtering characteristic function.

Figure 1 shows the designed masks. These masks made it possible for the LS-SVM to get a good prediction model to generalize the other possible cases that may appear in filtering. On the top of each mask is the assigned label that represents the desired output for that particular mask. $[X:y]=[\ 1,\ 1,-1,1,1,-1,1,1,-1]$ is the training vector for the first mask of Fig.1.

The assigned label to this particular mask is "+1" , which means that the white zone is generalized.

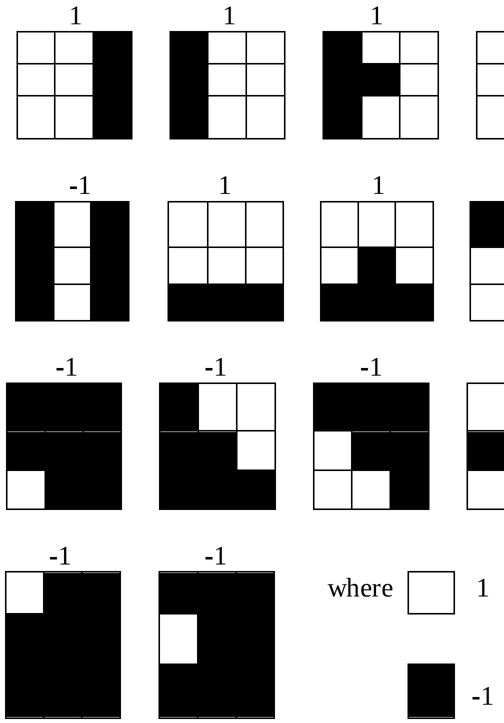


Figure 1. Example of LS-SVM training mask.

Procedures to implement the Least Squares Support Vector Machine Filter can be summarized as the following steps:

3.2.1 *Calculation of the coefficient* $a_i\ (i=1,2,\cdots,14)$ *and* b *in the LS-SVM classifier.*

<step1> input the given 14 training pairs $\{(X_i, y_i)\},\ i=1,2,\cdots 14$

<step2> setting the regularization parameter γ and the spread parameter σ of the RBF.

<step3> calculating the value of RBF $K(X_i, X_j)$.

$$K(X_i, X_j) = \varphi(X_i)^T \cdot \varphi(X_j) = \exp[-\frac{(X_i - X_j)^T (X_i - X_j)}{2\sigma^2}]$$

$$i, j = 1,2,\cdots 14$$

<step4> solving the following set of linear equations.

$$\begin{bmatrix} 0 & 1 & \cdots & 1 \\ 1 & \varphi(X_1)^T \varphi(X_1) + 1/\gamma & \cdots & \varphi(X_1)^T \varphi(X_{14}) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \varphi(X_{14})^T \varphi(X_1) & \cdots & \varphi(X_{14})^T \varphi(X_l) + 1/\gamma \end{bmatrix} \begin{bmatrix} b \\ a_1 \\ \vdots \\ a_{14} \end{bmatrix} = \begin{bmatrix} 0 \\ y_1 \\ \vdots \\ y_{14} \end{bmatrix}$$

<step5> output the coefficients $a_i\ (i=1,2,\cdots,14)$ and b.

3.2.2 *filtering operation process.*

< step1>input the observation image vector

$$Z = [\ z_1,\ z_2,\ \cdots\ z_9\ ].$$

< step2> arranging the image vector $Z^* = [\ z(1),\ z(2),\ \cdots\ z(9)\ ]$

where, $z(1) \le z(2) \le \cdots \le z(9)$.

< step3> threshold operation of the image vector

$$z_i^{m(j)} = T^{m(j)}(z_i) = \begin{cases} +1, & z_i \ge m(j) \\ -1, & z_i < m(j) \end{cases}$$

where,

$$m(j) = z(j),\quad j = 2, \cdots 9$$

$$i = 1, 2, \cdots 9$$

<step4> filtering operation.

$$LS-SVMF(Z) = \frac{z(1)+z(9)}{2} + \frac{1}{2}\sum_{j=2}^{9}(z(j)-z(j-1))(\mathrm{sgn}(\sum_{i=1}^{14} a_i K(Z^{m(j)}, X_i) + b))$$

<step5> output the filtering result.

4. Image Edge Detection Method Using LS-SVM

4.1 *Derivatives Of The Image Intensity Surface and Derivative Operators*

Many characteristics of the gray level intensity function in Eq. (8) depend on the type of kernel function $K(X_i, X_j)$. Although, there are many kernel functions available for calcul ation, in practice the Gaussian radial basis function (RBF) kernel is one of the most frequently studied kernels. There are two parameters γ and σ in RBF kernel-based LS-SVM. The parameter σ is very important, which can be constant in the image processing within the limited error tolerance. Parameter γ in LS-SVM controls the solution insensitivity to the error. Let S be a symmetric 2-D neighborhood defined on $R \times C$ and $f(r,c)$ be the observed intensity value at $(r,c) \in S$. The intensity estimation function of the LS-SVM with RBF kernel over the S can be written as follows

$$f(r,c) = \sum_{k=1}^{N} a_k \exp\left\{-(|r-r_k|^2 + |c-c_k|^2)/\sigma^2\right\} + b \tag{12}$$

where the a_k and b are the solution to Eq.(7). Evaluating the first and second row and column partial derivatives at point (r,c) yields the one- and second-order directional derivatives

$$\begin{aligned} \frac{\partial f}{\partial r} &= -\sum_{k=1}^{N} \frac{2a_k}{\sigma^2}(r-r_k)\exp\left\{-(|r-r_k|^2 + |c-c_k|^2)/\sigma^2\right\} \\ \frac{\partial f}{\partial c} &= -\sum_{k=1}^{N} \frac{2a_k}{\sigma^2}(c-c_k)\exp\left\{-(|r-r_k|^2 + |c-c_k|^2)/\sigma^2\right\} \\ \frac{\partial^2 f}{\partial r^2} &= -\sum_{k=1}^{N} \frac{2a_k}{\sigma^2}(1-\frac{2}{\sigma^2}(r-r_k)^2)\exp\left\{-(|r-r_k|^2 + |c-c_k|^2)/\sigma^2\right\} \\ \frac{\partial^2 f}{\partial c^2} &= -\sum_{k=1}^{N} \frac{2a_k}{\sigma^2}(1-\frac{2}{\sigma^2}(c-c_k)^2)\exp\left\{-(|r-r_k|^2 + |c-c_k|^2)/\sigma^2\right\} \end{aligned} \tag{13}$$

with the derivatives of the point (r, c) on the intensity

surface, the gradient vector at points (r, c) can be defined as

$$\nabla f(r,c) = [G_r, G_c]^T = [\partial f / \partial r, \partial f / \partial c]^T,$$

the gradient and directional angle of the vector is

$$mag(\nabla f) = [G_r^2 + G_c^2]^{1/2}, \quad \phi(r,c) = \arctan(G_c / G_r).$$

The second order derivative value of the point (r, c) can be defined as $\nabla^2 f = \partial^2 f / \partial r^2, \partial^2 f / \partial c^2$.

4.2 *Algorithm for the Image Edge Detection*

The procedures to create the second order derivative operator L for the image edge detection can be summarized as the following steps:

<step1> input initial data.

$$R = [-1\ 0\ 1],\ C = [-1\ 0\ 1]$$
$$X = R \times C$$

<step2> setting the regularization parameter γ and the parameter σ of the RBF.

<step3> calculating the value of RBF $K(X_i, X_j)$.

$$K(X_i, X_j) = \varphi(X_i)^T \cdot \varphi(X_j) = \exp[-\frac{(X_i - X_j)^T (X_i - X_j)}{2\sigma^2}]$$
$$i, j = 1, 2, \cdots 9$$

<step4> calculating matrices B^T, Q, F_1 and F_2

$$B^T = \frac{1^T \Omega^{-1}}{1^T \Omega^{-1} 1}, \quad Q = A(I - 1B^T)$$
$$F_1 = [f_{11}, \cdots, f_{19}],\ F_2 = [f_{21}, \cdots, f_{29}]$$

where ,

$$f_{1k} = -\frac{2}{\sigma^2}(1 - \frac{2}{\sigma^2}(r - r_k)^2)\exp\{-(|r - r_k|^2 + |c - c_k|^2)/\sigma^2\},$$
$$f_{2k} = -\frac{2}{\sigma^2}(1 - \frac{2}{\sigma^2}(c - c_k)^2)\exp\{-(|r - r_k|^2 + |c - c_k|^2)/\sigma^2\},$$
$$k = 1, 2, \cdots, 9$$

<step5> constructing the matrix l.

$$l = (F_1 + F_2) \times Q$$

<step6> rearranging central row of the matrix l.

$$L : 3 \times 3 \text{ matrix}$$

<step7> output the second order derivative operator L.

5. Experimental Results

First of all, the proposed LS-SVMF is used in an image denoising, i.e. to mitigate the impulsive noise in images. LS-SVMF is trained using masks like ones shown in Fig.1, for the RBF. Using these training masks, the result obtained shows the following coefficients , $a_i\ (i = 1, 2, \cdots, 14)$ and b.

$a_1 = 0.8490$, $a_2 = 0.7900$, $a_3 = 0.8717$, $a_4 = 0.9321$, $a_5 = -0.9751$, $a_6 = 0.7775$, $a_7 = 0.8603$, $a_8 = 0.8551$, $a_9 = -0.6947$, $a_{10} = -0.9755$, $a_{11} = -0.8813$, $a_{12} = -0.8730$, $a_{13} = -0.7539$, $a_{14} = -0.7822$, $b = 0.0335$

The performance of the proposed filter is compared to the performance obtained using SVM filter, ANN (artificial neural network) filter, Smoothing filter, Min filter. These filters with Mean Square Error (MSE) and Mean Absolute Error (MAE) are chosen and used in the comparison. Thus the performance of the LS-SVMF is compared to the performance of these filters.Figure 2 shows the performance of the LS-SVMF with RBF kernel. As can be seen, the LS-SVMF has the best performance in eliminating impulsive noise efficiently, while preserving details and features of the original image.

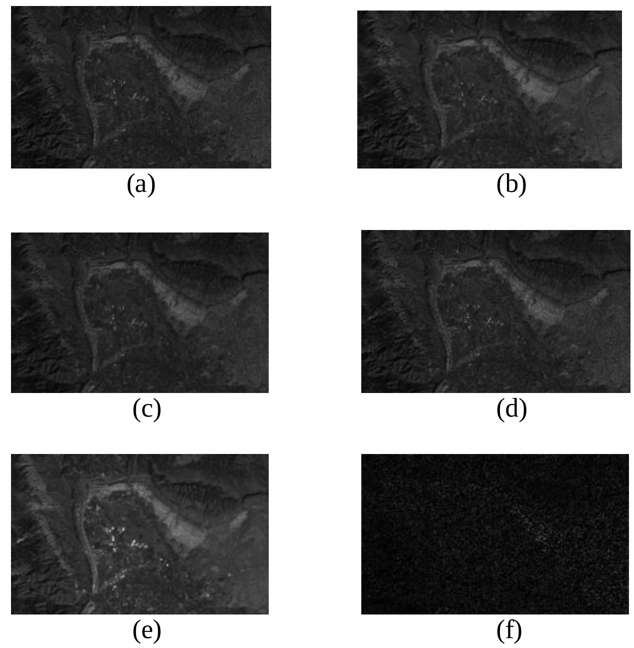


Figure 2. Performances of LS-SVMF and various filters (a) original image with noise density 20%, (b) LS-SVMF, (c) SVM filter, (d) ANN filter, (e) Smoothing filter, (f) Min filter.

Table 1 depicts the selected results of the MSE and MAE for the filtering process for a noise density of 20%. It can be observed by comparing the images and the errors values. As a result, the LS-SVMF shows the best performance compared to that of the other filters. The LS-SVMF not only removes impulsive noise effectively, but also keeps details well, acting as a competitive filter for the elimination of impulsive noise in images.

Table1. Mean Square Error and Mean Absolute Error.

Error / filters	MSE	MAE
LS-SVMF	65.57003	4.33437
SVM	101.07010	5.20913
ANN	124.46556	6.53673
Smoothing	172.11790	9.42107
Min	1198.36805	28.86915

Second of all, this paper proposed an approach for edge detection using a different point of view from the traditional. In this study, let R be defined as R= [-1 0 1] and C as C= [-1 0 1], the $\sigma = 0.5$, these operators defined over the symmetric 3×3 neighborhood can be represented as follows

$$g_r = \begin{bmatrix} 0.0977 & 0.5413 & 0.0488 \\ 0.5413 & 0.0000 & -0.5413 \\ -0.0488 & -0.5413 & -0.0977 \end{bmatrix}, \quad g_c = g_r^T$$

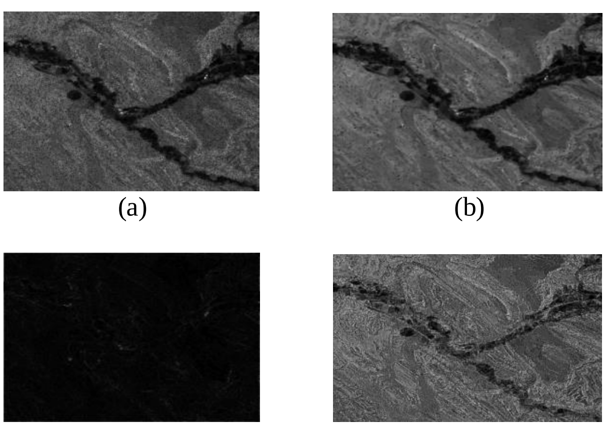
$$L = \begin{bmatrix} 0.1747 & 1.6768 & 0.1747 \\ 1.6768 & -7.4059 & 1.6768 \\ 0.1747 & 1.6768 & 0.1747 \end{bmatrix}.$$

For other example, let R be defined as R = [-2 -1 0 1 2], C as C= [-2 -1 0 1 2] and the coefficient m=2, the $\sigma = 1$, the operators over the symmetric 5×5 neighborhood defined over $R \times C$ can be represented as follows

$$g_r = \begin{bmatrix} -0.0168 & -0.0237 & -0.0105 & -0.0237 & -0.0168 \\ -0.0237 & -0.0230 & -0.0503 & -0.0230 & -0.0237 \\ -0.0105 & -0.0503 & 0.4150 & -0.0503 & -0.0105 \\ -0.0237 & -0.0230 & -0.0503 & -0.0230 & -0.0237 \\ -0.0168 & -0.0237 & -0.0105 & -0.0237 & -0.0168 \end{bmatrix}, \quad g_c = g_r^T$$

$$L = \begin{bmatrix} -0.0068 & 0.0071 & 0.0046 & 0.0071 & -0.0068 \\ 0.0071 & -0.0458 & -0.1360 & -0.0458 & 0.0071 \\ 0.0046 & -0.1360 & 0.6797 & -0.1360 & 0.0046 \\ 0.0071 & -0.0458 & -0.1360 & -0.0458 & 0.0071 \\ -0.0068 & 0.0071 & 0.0046 & 0.0071 & -0.0068 \end{bmatrix}.$$

Figure 3 shows the performance of the proposed edge detection method in noise image. The proposed edge detection using LS-SVM shows the better performance than the rest of the methods.



(a) (b) (c) (d)

Figure 3. Edge detection in images.
(a)original image with noise density 20%, (b)LS-SVMF image, (c)second order derivative operator, (d)gradient operator.

CONCLUSION

This paper proposes image denoising and edge detection method using LS-SVM. The proposed LS-SVMF is based on the general concept of binary filters where the characteristic function of the Boolean filter is replaced by the LS-SVM. Also a number of gradient operators and the second order derivative operators are obtained from the LS-SVM with RBF kernel. Experiments on a variety of images were carried out by using LS-SVM with RBF kernel and some selected parameters. Experimental results indicate that this new metho d gives a better performance than the traditional methods either in the noise elimination or the edge detection.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Ryang Chol-Sik: Methodology,Writing-original draft.Choe Yu Song: Jo Sok Chol: U Hyok-Chol: Writing–review-editing.

DECLARATION OF COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous referees for their very strict comments and valuable suggestions to imp rove the quality of this paper and to make more readable.

REFERENCES

1. Jose L. Paredes, Gonzalo R. Arce. Stack Filters, Stack Smoothers, and Mirrored Theshold Decomposition, IEEE Transactions on Signal Processing. 47 (10) (1999) 2757-2767.
2. L. Yin, R. Yang, M. Gabbouj, Y. Neuvo. Weighted Median Filters: ATutorial , IEEETrans. on Circuits and Systems. 43 (3) (1996)157-192.
3. Hou, Z.J., Koh, T.S.,. Robust edge detection. Pattern Recognit. 36 (9) (2003) 2083–2091.
4. J. Lladós, M. Rusiñol, A. Fornés, D. Fernández, A. Dutta, On the influence of word representations for handwritten word spotting in historical documents, Int. J. Pattern Recognit. Artif. Intell. 26 (2012).
5. M. Kalera, B. Zhang, S. Srihari, Offline signature verification and identification using distance statistics, Int. J. Pattern Recognit. Artif. Intell. 18 (7) (2004) 1339–1360.
6. U.-V. Marti, H. Bunke, Using a statistical language model to improve the performance of an HMM-based cursive handwriting recognition systems, Int. J. Pattern Recognit. Artif. Intell. (2001) 65–90.
7. F. Leclerc, R. Plamondon, Automatic signature verification: the state of the art-1989– , Int. J. Pattern Recognit. Artif. Intell. 8 (3) (1994) 643–660.
8. N. Cristianini and J. Shawe-Taylor. An Introduction to Support Vector Machines,(Cambridge University Press, Cambridge, 2000)
9. Hu, Z.Y.; Liu, Q.Y.; Tian, Y.X.; Liao, Y.F. A short-term wind speed forecasting model based on improved QPSO optimizing LSSVM. In Proceedings of the International Conference on Power System Technology (Chengdu, China, 2014) , pp.20–22.
10. Vapnik. V, The Nature of Statistical Learning Theory (Springer-Verlag, New York , 1995)
11. X. Yang, Q. Song, Y. Wang, A weighted support vector machine for data classification, Int. J. Pattern Recognit. Artif. Intell. 21 (05) (2007) 961–976.
12. Mahjoob, M.J.; Abdollahzade, M.; Zarringhalam, R. Ga based optimized LSSVM forecasting of short term electricity price in competitive power markets. In Proceedings of 2008 3rd IEEE Conference on Industrial Electronics and Applications (Siem Reap, Cambodia, 2017), pp.73–78.
13. Yuan, X.; Chen, C.; Yuan, Y.; Huang, Y.; Tan, Q. Short-term wind power prediction based on LSSVM–GSA model , Energy Convers. Manag. 101 (2015) 393–401.
14. Quanxue Gaoa ; Xinbo Gaoa ; Feiping Nie b . SVM based multi-label learning with missing labels for image annotation. Pattern Recognition 78 (2018) 307–317.

15. Xinying Panga ; Xianli Panb ; Yitian Xub . Multi-parameter safe sample elimination rule for accelerating nonlinear multi-class support vector machines. Pattern Recognition 95 (2019) 1–11.