ADAPTIVE WEIGHTED LEAST SQUARES SVM BASED SNOWING MODEL FOR IMAGE DENOISING

SHENG ZHENG
Institute of Intelligent Vision and Image Information
College of Science, China Three Gorges University
Yichang 443002, P. R. China
zsh@ctgu.edu.cn

CHANGCAI YANG*
School of Automation, Science and Technology
on Multi-Spectral Information Processing Laboratory
Huazhong University of Science and Technology
Department of Electronics and Information Engineering
Huazhong University of Science and Technology
Wuhan 430074, P. R. China
changcaiyang@gmail.com

EMILE A. HENDRIKS
Information and Communication Theory Group
Mathematics and Computer Science
Delft University of Technology
2628CD Delft, The Netherlands
e.a.hendriks@tudelft.nl

XIAOJUN WANG
School of Electronic Engineering
Dublin City University, Dublin 9, Ireland
xiaojun.wang@dcu.ie

Received 8 November 2012
Revised 12 June 2013
Accepted 12 June 2013
Published 5 November 2013

We propose a snowing model to iteratively smoothe the various image noises while preserving the important image structures such as edges and lines. Considering the gray image as a digital terrain model, we develop an adaptive weighted least squares support vector machine (LS-SVM) to iteratively estimate the optimal gray surface underlying the noisy image. The LS-SVM works on Gaussian noise while the weighted LS-SVM works

*Corresponding author.
on the outliers and non-Gaussian noise. To improve its performance in preserving the
directional signal while suppressing the noise, the dominant orientation information of
the gradients is integrated into the weighting scheme. The contribution of each attribute
to the final LS-SVM model is adaptively determined in a fitness modulated, elongated,
elliptical contour spread along the direction of the local edge structure. The farther away
from the hyperplane in kernel space the point is, the less weight it gets, while the point
on the direction of the local structure gets more weight. With adaptive weighting scheme,
the robust LS-SVM smoothes most strongly along the edges, rather than across them,
while it minimizes the effects of outliers, and results in strong preservation of details
in the final output. The iteratively adaptive reweighted LS-SVM simulated the snowing
process. The investigation on real images contaminated by mixture Gaussian noise has
demonstrated that the performance of the present method is stable and reliable under
noise distributions varying from Gaussian to impulsive.

**Keywords**: Snowing model; image denoising; weighted least squares SVM; directional
support value transform (SVT); support value matrix.

AMS Subject Classification: 22E46, 53C35, 57S20

1. Introduction

One of nature’s greatest beauties is the way that fresh snow covers the world in
a perfect blanket of crystalline white. It replaces sharp angles with gentle curves,
and clings to surfaces to form ghostly silhouettes. Snow transforms commonplace
scenes into fantastic wonderlands, greatly changing the appearance and mood of
the landscape, allowing us to see familiar sights in a fresh, exciting way. Unlike
existing research to set up a snow pack modeling of visual snow accumulation
for computer graphics, we are primarily concerned with simulating fallen snow
for image denoising. In practice, the images are often contaminated by noise due
to various reasons. A common problem in diverse domains of applied science and
engineering is to restore the original image. So far, various techniques have been pro-
duced to deal with it. Currently, the most popular methodologies include, for exam-
ple, wavelet-based image denoising methods, image denoising methods based on
Partial Differential Equations, neighborhood filters, dictionary-based image
denoising methods, some methods for impulse detection, and methods of
nonlinear modeling using kernel regression and/or local expansion approximation
techniques. In many cases, the denoising techniques are focused on a particular
noise model (Gaussian, impulse, etc.). Thus, they cannot treat more complex
models, which are often met in practical applications, effectively.

This paper proposes a different image denoising approach, a snowing model,
which can be stable and reliable over a wider range of noise distributions. In the
natural snowing process, the snow deposits adaptively on the ground surface, and
g radually cover the small details, such as the grasses, while the contours of the
grass fields are gradually manifested. Finally, the digital terrain model without the
tiny details such as the grasses and trees is formed. The gray image is regarded
as a digital terrain model, and the image denoising problem is related to the optimal
g gray surface estimation from noisy samples, the pixels. An iteratively adaptive
Adaptive Weighted LS-SVM Based Snowing Model for Image Denoising

weighted least squares support vector machine (LS-SVM) is proposed to realize the snowing model. To solve a nonlinear problem in the original space, the support vector machine (SVM) is applied to map the data into a higher dimensional space (possibly of infinite dimension) and the task is more tractable, by employing a linear model in the feature space. The use of the kernel function only needs the calculation of the scalar product of the intermediate space, avoids the computation of the nonlinear mapping function, and never needs to explicitly even know what the mapping function is. It has proved to be a powerful tool for pattern recognition and function estimation. For the image denoising problem, the feature vector for a pixel in a noisy image is formed by its neighborhood pixels (local patch) and the target is the corresponding pixel in the noisy image. The objective is to obtain an optimal estimation of a pixel from the noisy information of its neighborhood pixels. The dominant orientation information of the gradients and the error of each data from the fitted hyperplane in kernel space are used to adaptively determine the contribution of each attribute to the final regression function. This results in a weighting scheme in fitness modulated, elongated, elliptical contours spread along the direction of the local edge structure. The farther away from the fitted hyperplane in kernel space the point is, the less weight it gets, while the point in the direction of the local structure gets more weight. Though there has been some work exploring the use of kernels in the denoising problem, the methodology presented here is fundamentally different. In steering kernel regression (SKR), the notion of kernel regression has been adopted. The original image is formulated as a local (e.g., Taylor) approximation series around a center and data adaptive kernels are used, as weighted factors, to penalize distances away from the center. In a relatively similar context, kernels have been employed by other well-known denoising methods (such as in Refs. 20–23). In Ref. 21, a support vector regression approach is considered for the Gaussian noise case and in Refs. 22, 23 the kernel principal components of local pixel grouping are extracted and this expansion is truncated to produce the denoising effect. Reference 20 assumes that the original (noise free) image lies in a Reproducing Kernel Hilbert Space (RKHS) and thus it is formulated as a linear combination of specific kernel functions that justify the existence of a RKHS. The SVM approach we have used gives us the power to model and work in infinite-dimensional Hilbert spaces, instead of a simple family of kernels. Hence, our approach lies closer to the RKHS methods and support vector regression and kernel principal component analysis derived in Refs. 20–22, than to the kernel regression employed in Ref. 19 and elsewhere. Similar to RKHS and SKR, the proposed denoising procedure is performed inside a pixel-centered region that moves from one pixel to the next and the parameters of the representation model are controlled adaptively at each region to preserve the fine details and local statistics of the image. In the proposed method, the LS-SVM-model is adaptive to the local structure and statistical information, and the contribution of each attribute to the final regression function is adaptively determined by the local directional information and the fitness of the LS-SVM model. This is completely different from
the method in the support vector regression (SVR) approach in Ref. 21, where the very few images are used as the training image of the SVR, and the image denoising is completed by the generalization of the SVR model. In the proposed LS-SVM model, the kernels are used in order to explicitly model the denoised image, in contrast to the SKR method of Ref. 19, where kernels are used as weighted factors to a Taylor approximation. And it is also different from the RHKS method in Ref. 20, where the image denoising is completed by optimal semi-parametric representation. Here, the regularized risk minimization problem is a $l_1$ norm cost function, and the image gray surface is optimally fitted by the hyperplane in kernel space with a least squares object function.

The paper is structured as follows. In Sec. 2, we briefly describe the LS-SVM. In Sec. 3, we present the proposed approach to the image denoising problem. The framework of the snowing model, the details of the implementation as well as the algorithmic scheme can be found there. Experiments on images corrupted by various types of synthetic noise models (impulse, gaussian, uniform, mixed) are detailed in Sec. 4, and Sec. 5 concludes the paper.

2. Least Squares Support Vector Machine (LS-SVM)

Assume a scattered data set of $N$ points $\{(x_k, y_k)\}_{k=1}^{N}$ with input data $x_k \in \mathbb{R}^n$ and output data $y_k \in \mathbb{R}$. For smooth fitting of image, the input vector $x_k$ represents coordinates of the pixels. In SVM, the smooth fitting function is represented by

$$ f(x) = w^T \phi(x) + b, \quad k = 1, \ldots, N, \quad (2.1) $$

where the nonlinear mapping $\phi : \mathbb{R}^n \rightarrow \mathbb{R}^m$ maps the input data into a so-called high dimensional feature space (which can be infinite dimensional) and vector $w \in \mathbb{R}^m$ and bias $b \in \mathbb{R}$.

For smooth fitting and efficient computational purposes, in LS-SVM, SVM has been modified into the following object function

$$ J(w, e) = \frac{1}{2} w^T w + \frac{\gamma}{2} \sum_{k=1}^{N} e_k^2 \quad (2.2) $$

subject to equality constraints

$$ y_k = w^T \phi(x_k) + b + e_k, \quad k = 1, \ldots, N. \quad (2.3) $$

The (2.2) stands for the minimization of the Vapnik Chervonenkis (VC) dimension, while (2.3) minimizes the training errors ($e_k$) ($\gamma$ is a trade-off constant between smoother solutions and residuals).

The weight vector $w$ can be infinite-dimensional, which makes it impossible to calculate $w$ from (2.2) in general. Therefore, one normally computes the model in a dual space instead of the primal space. Define the following Lagrangian
equation:

\[
L(w, b, e, \alpha) = J(w, e) - \sum_{k=1}^{N} \alpha_k \{ w^T \phi(x_k) + b + e_k - y_k \},
\]

with Lagrange multipliers \( \alpha_k \in \mathbb{R} \). The conditions for optimality are given by the linear Karush–Kuhn–Tucker (KKT) system. That is to say, the optimality condition leads to the following \((N + 1) \times (N + 1)\) linear system

\[
\begin{bmatrix}
0 & 1_v^T \\
1_v & \Omega + \frac{1}{\gamma} I
\end{bmatrix}
\begin{bmatrix}
b \\
\alpha
\end{bmatrix} =
\begin{bmatrix}
0 \\
Y
\end{bmatrix},
\]

where \( Y = [y_1, \ldots, y_N]^T \), \( 1_v = [1, \ldots, 1]^T \), \( \alpha = [\alpha_1, \ldots, \alpha_N]^T \), \( \Omega_{i,j} = K(x_i, x_j) \), for \( i, j = 1, 2, \ldots, N \). The \( K(x_i, x_j) = \phi(x_i)^T \phi(x_j) \), \( i, j = 1, 2, \ldots, N \), is a kernel function satisfying the Mercer’s condition. The use of kernel function only requires the calculation of the scalar product of the intermediate space, avoiding the computation of the nonlinear \( \phi(x_i) \) function, and never even needs to know explicitly what \( \phi(x_i) \) is. The second row of (2.5) gives \( 1_v b + (\Omega + \gamma^{-1} I) \alpha = Y \) and together with the first row gives the explicit solution

\[
b = \frac{1_v^T (\Omega + \gamma^{-1} I)^{-1} Y}{1_v^T (\Omega + \gamma^{-1} I)^{-1} 1_v},
\]

\[
\alpha = (\Omega + \gamma^{-1} I)^{-1} (Y - 1_v b)
\]

\[
= (\Omega + \gamma^{-1} I)^{-1} \left( I - \frac{1_v 1_v^T (\Omega + \gamma^{-1} I)^{-1}}{1_v^T (\Omega + \gamma^{-1} I)^{-1} 1_v} \right) Y = QY,
\]

where \( Q \) is an \( N \times N \) matrix. The support values in a pixel-centered region are calculated by multiplying the column vector \( Y \) by the corresponding row vectors of \( Q \). That is to say, for the pixel in a pixel-centered region, the corresponding support value can be computed individually as a linear combination of the pixels. The weight associated with each pixel is, respectively, determined by the elements of the corresponding row vector of \( Q \). For a rectangular region, reshaping the corresponding row vectors of matrix \( Q \), the weight kernels become the support value filters. If the support value of the discrete image in the pixel is approximated by the corresponding support value of the input vector in region center, the support values of the whole image can be computed by convolving the image with the support value filter deduced from the central row vector of \( Q \).

Then we can get the final LS-SVM regression model for smooth fitting function as the following:

\[
y(x) = \sum_{k=1}^{N} \alpha_k K(x, x_k) + b,
\]

where \( \alpha, b \) are the solutions of (2.5).
3. Snowing Model for Image Denoising

3.1. Natural snowing on the ground

As snow falls from the sky, it lands on the ground, forming a snow layer and eventually accumulating with other layers to form a snow pack. Usually, each snow layer represents a different type of snow, with its own physical properties, because individual snow layers are often deposited under varying weather conditions. In between snowfalls, all snow layers are affected by a wide number of environmental factors that can accentuate or remove the differences between layers, and greatly change the physical properties of the entire snow pack. Snow layers continually change until they melt, evaporate, or become ice. Snowing transforms commonplace scenes into fantastic wonderlands, greatly changing the appearance and mood of the landscape. As illustrated in Fig. 1, the snow pack covering on the roof of the house removes the detailed textures of different tiles, and merges the tiles into a whole region with manifested contour of the building roof. The snow deposited on the grasses removes the differences of grasses themselves and manifests the contour of object underlying the grasses. The length of the snowing time significantly affects the appearance of the landscape. The longer the snowing time, the thicker the snow pack becomes, which smoothes more details of the landscape, such as the grasses and manifests the contour of the larger object, such as the grass field.

From the perspective of image processing, the image can be regarded as a digital terrain model, the landscape, and the noise correspond to the tiny details of the landscape. For image processing, the purpose of the snowing model is that the snowing process would remove the noise and restore the image from the noisy data for post image processing such as segmentation etc.

3.2. Framework of snowing model for image denoising

The overall proposed snowing model-based image denoising approach is illustrated in Fig. 2. The problem of image denoising is transferred as a problem of digital terrain analysis, where the gray value of the image is regarded as the elevation

![Fig. 1. The effect of snowing. (a) Before snowing and (b) covered with snow pack.](image-url)
Adaptive Weighted LS-SVM Based Snowing Model for Image Denoising

Fig. 2. The framework of the snowing model.

value of a digital terrain, and the evolution of terrain surface is modeled by the snowing.

The snowing model is developed to simulate the evolution of the terrain surface when snows continuously accumulate on the digital terrain model. The simulation model of the snow depositing process is based on the observation of snowing in the real world. In the real world, it has been observed that if snow particles drop from a height above a terrain surface, they move towards the terrain, with a downward speed. The particles interact with the terrain surface and surrounding snow particles. The snow classically follows the steepest inclination of the terrain surface and finally finds a natural stopping point. It has been found that snow accumulates faster on steeper slopes, an action which leads to a thicker snow canopy at the “stop” location, while snow accumulates more slowly on less steep slopes, presenting a thinner snow canopy at the “stop” location. Another observed feature is that snow fills small pits in the terrain surface very quickly, enabling the larger elements of the terrain to be more obvious. In other words, the smaller terrain indentations and peaks disappear leaving the outline of the larger terrain elements more obvious. The final “stop” location of the snow particles is determined by several factors, such as the speed and direction of the snow movement, distribution of neighboring snow particles and the attributes of the terrain surface, etc. When the overall movement of snowing terminates, the snow pack on the terrain surface is then formed.

It is complex and difficult to accurately simulate the real snowing process. In the proposed snowing model, the process of snow accumulation in the space of digital terrain model will be primarily studied based on a simplified solution so
as to integrate more factors in modeling snow deposition and improve the computation efficiency. Here, digital terrain will be modeled by the object oriented model with the basic geometric elements of point, line and patch. In modeling snow deposition process, the object oriented model is applied to describe the geometry of spatial objects on the terrain surface as points (such as for isolated terrain feature point), lines (such as for mountain ridges, or contours of buildings) and surfaces (such as buildings and lakes), which respectively correspond to impulse noises or non-Gaussian noises, edges or lines, plate areas in image. The depositing speed and pattern of snow are related to the geometry of the spatial objects, and topologic relationships between the objects on the terrain surface. These are modeled in the simulation of the snow accumulation process. In real snowing process, the snow pack gradually becomes thick when snowing continues, the evolution of the terrain surface increases the elevation value of a digital terrain, which corresponds to increasing the gray value of image and makes the image become more and more bright. However, image processing requires the snowing model not to change the brightness of the image. For this purpose, the iteratively adaptive weighted LS-SVM is proposed to automatically integrate these into our simulated snowing model.

The evolution of terrain surface in the snowing process corresponds to restore the gray surface of image from the noisy data in snowing model. The SVM is a powerful tool for function estimation, the feature vector for a pixel in a noisy image is formed by its neighborhood pixels (local patch) and the target is the corresponding pixel in the noisy image. The LS-SVM model is powerful to obtain an optimal estimation of a pixel from the Gaussian noisy information of its neighborhood pixels. Figure 3(b) shows the results obtained by the application of the unweighted LS-SVM on Lena. As observed in the kernel-based denoising approaches, we can immediately see that the result of the denoising process is a blurry image. The noise has been removed successively, but in the process most of the fine details have been lost. At the same time, the outliers or non-Gaussian noises cannot be effectively removed out. To improve the performance of the LS-SVM in case of outliers and noise with non-Gaussian distribution, a weighted version of the LS-SVM algorithm can be used. Figure 3(c) shows the results obtained by the application of the weighted LS-SVM. The weighted LS-SVM effectively smoothes the Gaussian and non-Gaussian noises, but it still blurs the image, because the weighted LS-SVM model just only automatically identify and smoothes the noises, it cannot discriminate the different geometry of spatial objects on the terrain surface as point, line or edge, patch, and automatically adjust the snow depositing speed or control the smoothing process of the image noises in different areas. If the structural information of the image, the geometrical information of spatial objects on the terrain surface, can be automatically identified and integrated into the weighting scheme to adaptively determine the contribution of each attribute to the final regression function of the LS-SVM, the performance of the LS-SVM based snowing model will be improved, as demonstrated in Fig. 3(d).
Adaptive Weighted LS-SVM Based Snowing Model for Image Denoising

Fig. 3. (a) Lena image corrupted by 20% of uniform distributed impulse noise (Type II, α = 128) and Gaussian noise with $s = 10$ (PSNR = 17.98 dB), (b) denoised image by unweighted LS-SVM (PSNR = 27.61 dB), (c) denoised image by the standard weighted LS-SVM (PSNR = 29.34 dB) and (d) denoised image by the proposed adaptive weighted LS-SVM (PSNR = 32.30 dB). The snowing model significantly increased the quality of denoised image.

3.3. Snowing model-adaptive weighted LS-SVM

In case of outliers and noise with non-Gaussian distribution the performance of the LS-SVM is affected. In order to reduce the impact of these outliers a weighted version of the LS-SVM algorithm is used.\cite{27} The weighted LS-SVM algorithm first computes an unweighted LS-SVM and calculates the errors $e_s$. Then the standard deviation of the errors $\hat{s}$ is computed by $\hat{s} = 1.483 \times \text{MAD}(e_s)$, a robust estimate of the standard deviation, in order to identify the outliers that affect the model performance.

To ensure that salient features of the image are non-symmetrically enhanced, the dominant orientation information of the gradients is used to further adaptively determine the contribution $w_k$ of each attribute to the final regression function. Its performance depends on the estimation of the oriented pattern direction and the
maximum support radius of the kernel. Obviously, the more robust the estimation
of the underlying pattern direction is, the more enhanced the salient features are.
Meanwhile, to further effectively reduce noise effects while preserving images’ sharp-
ness, for the weighting kernel $u$, large support radii should be preferred in texture-
less areas and smaller ones in areas with rich textures. Following this consideration,
in this paper, we propose to make all aspects of the weighting kernel, including the
effective size, shape and direction, adaptive locally to image features such as edges
and the outliers or non-Gaussian noise. This results in a weighting scheme in fitness
or errors modulated, elongated, elliptical contours spread along the direction of the
local edge structure. We finally define the data-adaptive weighting kernel $u$ applied
at each pixel $x_k$ inside a pixel-centered $(x_0)$ region as follows:

$$u_k = \nu_k \exp \left( -\frac{(x_k - x_0)^T C_0 (x_k - x_0)}{2h^2} \right), \quad (3.1)$$

where $h$ is a global smoothing factor, $C_0$ is (symmetric) covariance matrix based
on differences in the local gray values. A good choice for $C_0$ will effectively spread
the kernel function along the local edges. Here we take a convenient form of the
covariance matrix used in Ref. 19. The modulated factor $\nu_k$ can be defined based
on estimated standard deviation $\hat{s}$ as follows:

$$\nu_k = \begin{cases} 
1, & \text{if } |e_s/\hat{s}| \leq c_1 \\
\frac{c_2 - |e_s/\hat{s}|}{c_2 - c_1}, & \text{if } c_1 \leq |e_s/\hat{s}| \leq c_2 \\
10^{-4}, & \text{otherwise}
\end{cases} \quad (3.2)$$

where $c_1$ and $c_2$ are constant with typical values of 2.5 and 3, respectively.

With the locally adapted weighting scheme in (3.1), a new adaptive weighted LS-
SVM problem where the errors and directional information and the contributions
are weighted is formulated. The formulation of the problem is given by:

$$\min_{w, b, e} J(w, e) = \frac{1}{2} w^T w + \frac{\gamma}{2} \sum_{k=1}^{N} \frac{1}{u_k} e_k^2 \quad (3.3)$$

subject to equality constrains $w^T \phi(x_k) + b = y_k - e_k, k = 1, \ldots, N$. Applying
the Lagrangian and conditions for optimality the problem simplifies to:

$$\begin{bmatrix} 0 & I^T \\ I & \Omega + M_\gamma \end{bmatrix} \begin{bmatrix} b \\ \alpha \end{bmatrix} = \begin{bmatrix} 0 \\ Y \end{bmatrix}, \quad (3.4)$$

where $M_\gamma$ is a diagonal matrix given by:

$$M_\gamma = \text{diag} \left( \frac{1}{\gamma u_1}, \ldots, \frac{1}{\gamma u_N} \right). \quad (3.5)$$
With this model, the denoising is most effective along the edges, rather than across them, resulting in strong preservation of details, while effectively reduces the impact of the outliers or non-Gaussian noise.

### 3.4. Algorithm of snowing model — Iteratively adaptive reweighted LS-SVM

Just as the snow depositing is terrain surface dependent, the proposed adaptive weighted LS-SVM image denoising model is data dependent. Consequently, it is sensitive to the noise in the input image. We can obtain a robust estimate based upon the proposed LS-SVM solutions using an iteratively reweighting approach. In the $i$th iteration one can weight the error variables $\epsilon_k^{(i)} = \alpha_k^{(i)}/\gamma$ for $k = 1, \ldots, N$ by weighting factors $u_k^{(i)} = (u_1^{(i)}, \ldots, u_N^{(i)})$ determined by the weighting function in (3.1). An iterative algorithm to simulate the snowing process for image denoising is formulated in Algorithm 1.

While we do not provide an analysis of the convergence of this iterative procedure, it can be noted that increasing the number of iterations reduces the variance of the estimate, and also leads to increased bias (which manifests as blurriness). A future line of work will analyze the derivation of an effective stopping rule from first principles.

**Algorithm 1 Iteratively adaptive reweighted LS-SVM.**

**Input:** the noisy image

**Output:** The denoised image

1. repeat
2. Estimate the gradients of the input image, and compute the covariance matrix $C_0$;
3. Given optimal learning parameters $(\gamma, \delta)$, compute the $N \times N$ matrix $Q$ in (2.7);
4. For each pixel $(x)$ do
   a. Compute the residuals $\epsilon_k^{(i)} = \alpha_k^{(i)}/\gamma$ from the unweighted LS-SVM in (2.7);
   b. Compute $\hat{s} = 1.483 \text{MAD}(\epsilon_s)$ from the $\epsilon_s$ distribution;
   c. Determine the weights $\nu_k$ based upon the weight function in (3.2);
   d. Determine the weights $u_k^{(i)}$ based upon $C_0(x)$ and the weight function in (3.1);
   e. Solve the weighted LS-SVM (3.4) with a diagonal matrix given by:
      $$M_\gamma = \text{diag}(\frac{1}{\gamma u_1}, \ldots, \frac{1}{\gamma u_N})$$
      resulting the model $I^{(i)}(x) = \sum_{k=1}^N \alpha_k^{(i)} K(x, x_k) + b^{(i)}$.
5. until some stopping criterion is satisfied.
4. Experiment Results

To evaluate the performance of various image denoising algorithms, the provided test images corrupted with various types of synthetic noise, which are based on the test images contained in the Waterloo Image Repository. To verify the validation of the proposed snowing model based image denoising method, we tested the data sets provided in Ref. 20. The results were compared with those obtained using several state of the art models (BiShrink, BLS-GSM, K-SVD, SKR, BM3D, and the RKHS method), where in all cases the input parameters were carefully adjusted to obtain the best possible results with respect to PSNR. For the proposed snowing model, the global smoothing factor of the structural weighting kernel is fixed as 1.2, the size of pixel-centered window for the LS-SVM is set to 9 × 9, the gradients of the reconstructed images are estimated by the first order derivative Gaussian filter with standard deviation of 0.85 for computing covariance matrix $C_0$, the kernel parameter $\sigma$ equals to the standard deviation of the estimated errors by the unweighted LS-SVM ($1.483 \text{MAD}(e_k)$), and the $\gamma$ is set to $10^{(4.0-0.1\sigma)}$, and the iteration number is adaptive determined to obtain the best possible with respect to PSNR.

All experiments are implemented in Matlab and run on a Pentium(R) TM 1.7 GHz machine with 1 G RAM.

4.1. Gaussian noise

The results of the adaptive weighted LS-SVM based snowing method on various images corrupted by Gaussian noise are reported in Table 1 and the visual comparison is shown in Fig. 4. The BM3D algorithm is reported to be one of the best approaches in Gaussian noise removal. Its performance shows an average improvement of approximately 1–2 dB (sometimes more) over the proposed LS-SVM and RKHS approaches. In many cases, our method seems to have similar behavior (in terms of PSNR and visual comparison) with another very well-known wavelet-based algorithm called BiShrink. Compared with RKHS, it provides a little higher PSNR values in Table 1 and demonstrates some better visual quality on lines or edges in Figs. 4(b) and 4(c).

4.2. Impulse noise

The results of the adaptive weighted LS-SVM-based snowing method on the Lena, Peppers, Barbara, and Boat images corrupted by bipolar impulse noise (Type I)
Adaptive Weighted LS-SVM Based Snowing Model for Image Denoising

<table>
<thead>
<tr>
<th>Image</th>
<th>Noise</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Noisy</td>
</tr>
<tr>
<td>Lena</td>
<td>s = 10</td>
<td>28.07</td>
</tr>
<tr>
<td></td>
<td>s = 20</td>
<td>22.13</td>
</tr>
<tr>
<td></td>
<td>s = 30</td>
<td>18.71</td>
</tr>
<tr>
<td>Peppers</td>
<td>s = 10</td>
<td>28.13</td>
</tr>
<tr>
<td></td>
<td>s = 20</td>
<td>22.32</td>
</tr>
<tr>
<td></td>
<td>s = 30</td>
<td>18.93</td>
</tr>
<tr>
<td>Barbara</td>
<td>s = 10</td>
<td>28.11</td>
</tr>
<tr>
<td></td>
<td>s = 20</td>
<td>22.16</td>
</tr>
<tr>
<td></td>
<td>s = 30</td>
<td>18.73</td>
</tr>
<tr>
<td>Boat</td>
<td>s = 10</td>
<td>28.26</td>
</tr>
<tr>
<td></td>
<td>s = 20</td>
<td>22.19</td>
</tr>
<tr>
<td></td>
<td>s = 30</td>
<td>18.76</td>
</tr>
</tbody>
</table>

and uniformly distributed impulse noise (Type II) are reported in Tables 2 and 3, respectively, and the visual comparison is illustrated in Figs. 5 and 6. In both types of impulse noise the RKHS algorithm gives excellent results (both visually and in terms of PSNR). The wavelet based techniques are known not to be able to deal with impulse noise effectively. The proposed snowing method performs well and is close to the RKHS method for uniformly distributed impulse noises (Type II). Results show that the snowing noise removal algorithm can achieve an improvement of more than 5 dB (some times up to 10 dB) in terms of PSNR, over most

(a) (b)

Fig. 4. The denoising results of Boat with Gaussian noise by different models. (a) Boat corrupted by Gaussian noise with s = 30, (b) denoised image by the snowing approach (PSNR = 28.55 dB), (c) denoised image by RKHS (PSNR = 27.35 dB), (d) denoised image by BiShrink (PSNR = 27.77 dB), (e) denoised image by SKR (PSNR = 28.21 dB) and (f) denoised image by BM3D (PSNR = 29.72 dB).
S. Zheng et al.

Table 2. Results of the adaptive weighted LS-SVM based snowing method on various images corrupted by bipolar impulse noise (Type I) ($a = 100$).

<table>
<thead>
<tr>
<th>Image</th>
<th>Noise (%)</th>
<th>Noisy (dB)</th>
<th>Noisy Snowing (dB)</th>
<th>RKHS (dB)</th>
<th>BiShrink (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>20</td>
<td>15.77</td>
<td>34.56</td>
<td>33.62</td>
<td>24.56</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>14.01</td>
<td>30.11</td>
<td>31.48</td>
<td>26.16</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>12.76</td>
<td>27.21</td>
<td>30.22</td>
<td>25.27</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>11.78</td>
<td>23.65</td>
<td>28.01</td>
<td>24.51</td>
</tr>
<tr>
<td>Peppers</td>
<td>20</td>
<td>16.10</td>
<td>31.85</td>
<td>32.22</td>
<td>23.24</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>14.31</td>
<td>29.49</td>
<td>30.49</td>
<td>25.33</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>13.08</td>
<td>27.00</td>
<td>29.21</td>
<td>24.29</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>12.13</td>
<td>23.92</td>
<td>27.92</td>
<td>23.35</td>
</tr>
<tr>
<td>Barbara</td>
<td>20</td>
<td>15.81</td>
<td>26.18</td>
<td>26.21</td>
<td>23.45</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>14.07</td>
<td>24.63</td>
<td>24.91</td>
<td>23.47</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>12.78</td>
<td>23.01</td>
<td>23.80</td>
<td>22.75</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>11.81</td>
<td>20.25</td>
<td>22.88</td>
<td>22.25</td>
</tr>
<tr>
<td>Boat</td>
<td>20</td>
<td>15.92</td>
<td>31.67</td>
<td>31.01</td>
<td>24.04</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>14.14</td>
<td>27.79</td>
<td>28.84</td>
<td>24.71</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>12.90</td>
<td>25.08</td>
<td>27.57</td>
<td>23.91</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>11.92</td>
<td>21.97</td>
<td>25.88</td>
<td>23.30</td>
</tr>
</tbody>
</table>

Fig. 4. (Continued)
### Adaptive Weighted LS-SVM Based Snowing Model for Image Denoising

Table 3. Results of the adaptive weighted LS-SVM based snowing method on various images corrupted by uniformly distributed impulse noise (Type II) ($\alpha = 128$).

<table>
<thead>
<tr>
<th>Image</th>
<th>Noise (%)</th>
<th>Noisy</th>
<th>Snowing</th>
<th>RKHS</th>
<th>BiShrink</th>
<th>KSVD</th>
<th>SKR</th>
<th>BM3D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>20</td>
<td>18.42</td>
<td>35.31</td>
<td>35.03</td>
<td>22.82</td>
<td>28.96</td>
<td>29.78</td>
<td>30.94</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>16.63</td>
<td>32.43</td>
<td>32.90</td>
<td>25.88</td>
<td>27.97</td>
<td>28.01</td>
<td>29.82</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>15.37</td>
<td>30.48</td>
<td>31.52</td>
<td>26.74</td>
<td>27.33</td>
<td>27.56</td>
<td>29.06</td>
</tr>
<tr>
<td>Peppers</td>
<td>20</td>
<td>18.68</td>
<td>32.84</td>
<td>32.73</td>
<td>22.94</td>
<td>29.08</td>
<td>29.56</td>
<td>30.85</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>16.93</td>
<td>31.21</td>
<td>31.76</td>
<td>25.21</td>
<td>27.94</td>
<td>28.12</td>
<td>29.70</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>15.68</td>
<td>29.68</td>
<td>30.50</td>
<td>25.89</td>
<td>27.10</td>
<td>26.52</td>
<td>28.61</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>14.70</td>
<td>28.17</td>
<td>29.72</td>
<td>25.50</td>
<td>26.06</td>
<td>25.06</td>
<td>27.36</td>
</tr>
<tr>
<td>Barbara</td>
<td>20</td>
<td>18.42</td>
<td>28.26</td>
<td>27.25</td>
<td>22.86</td>
<td>27.11</td>
<td>27.28</td>
<td>29.46</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>15.43</td>
<td>25.29</td>
<td>24.80</td>
<td>24.28</td>
<td>25.53</td>
<td>25.02</td>
<td>27.46</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>14.45</td>
<td>24.32</td>
<td>24.29</td>
<td>23.81</td>
<td>24.88</td>
<td>23.67</td>
<td>26.74</td>
</tr>
<tr>
<td>Boat</td>
<td>20</td>
<td>18.56</td>
<td>33.06</td>
<td>32.12</td>
<td>22.59</td>
<td>27.71</td>
<td>28.11</td>
<td>29.42</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>16.77</td>
<td>30.28</td>
<td>30.50</td>
<td>25.07</td>
<td>26.98</td>
<td>26.85</td>
<td>28.32</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>15.52</td>
<td>28.32</td>
<td>29.13</td>
<td>25.40</td>
<td>26.11</td>
<td>25.67</td>
<td>27.45</td>
</tr>
</tbody>
</table>

Fig. 5. The denoising results of Lena with 20% of bipolar impulse noise by different models. (a) Lena corrupted by 20% of bipolar impulse noise (Type I) (PSNR = 15.81 dB), (b) denoised image by snowing approach (PSNR = 26.18 dB) and (c) denoised image by RKHS (PSNR = 26.21 dB).

Wavelet-based methods and much better visual quality (even in cases where the difference of PSNRs is relatively small). The same is true for most other methods such as KSVD and SKR. The snowing approach performs relatively well in the presence of bipolar impulse noise (Type I), but it provides much better visual quality than the BiShrink approach even in case of high noise level, as shown in Fig. 6. The RKHS method performs better than the snowing method in case of bipolar impulse noise (Type I) with high level noise. The possible reason is that the proposed LS-SVM in snowing method uses the least squares cost function while the RKHS approach uses the $l_1$ norm cost function, and the $l_1$ norm cost...
Fig. 6. The denoising results of Barbara with uniformly distributed impulse noise by different models. (a) Barbara corrupted by 40% of uniformly distributed impulse noise (Type II) (PSNR = 15.43 dB), (b) denoised image by the snowing approach (PSNR = 25.29 dB), (c) denoised image by RKHS (PSNR = 29.13 dB), (d) denoised image by BiShrink (PSNR = 24.26 dB), (e) denoised image by KSVD (PSNR = 25.53 dB) and (f) denoised image by BM3D (PSNR = 27.45 dB).
function works better against bipolar outliers. The denoised images obtained by the BM3D algorithm, which are superior to the ones obtained by other kernel based or wavelet based methods, yield significant loss in details and a difference of 1–3 dB in comparison to the proposed methodology. Although BM3D results in higher PSNR than the proposed snowing method, there is a significant loss in fine details as it can be seen, for example, in Fig. 6. In Fig. 6(f), it can be observed that the face of Barbara is distorted and the texture of the chair (behind her), as well the texture of the table cloth, are blurred compared to the image obtained by the proposed snowing method, as it becomes evident in Fig. 6(b). There are, of course, some other regions (patterns in the pants and scarf) that BM3D restores better.

Table 4. Results of the adaptive weighted LS-SVM based snowing method on various images corrupted by uniform noise.

<table>
<thead>
<tr>
<th>Image</th>
<th>Noise</th>
<th>Noisy</th>
<th>Snowing</th>
<th>RKHS</th>
<th>BiShrink</th>
<th>KSVD</th>
<th>SKR</th>
<th>BM3D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>±20</td>
<td>26.85</td>
<td>33.91</td>
<td>33.00</td>
<td>33.56</td>
<td>34.33</td>
<td>34.73</td>
<td>35.04</td>
</tr>
<tr>
<td></td>
<td>±30</td>
<td>23.34</td>
<td>32.44</td>
<td>30.76</td>
<td>31.80</td>
<td>32.34</td>
<td>33.30</td>
<td>33.43</td>
</tr>
<tr>
<td></td>
<td>±40</td>
<td>20.85</td>
<td>31.47</td>
<td>29.36</td>
<td>30.48</td>
<td>31.00</td>
<td>31.03</td>
<td>32.21</td>
</tr>
<tr>
<td>Peppers</td>
<td>±20</td>
<td>27.00</td>
<td>32.55</td>
<td>31.78</td>
<td>33.05</td>
<td>33.89</td>
<td>33.96</td>
<td>34.35</td>
</tr>
<tr>
<td></td>
<td>±30</td>
<td>23.50</td>
<td>31.16</td>
<td>30.25</td>
<td>31.31</td>
<td>32.10</td>
<td>32.38</td>
<td>32.97</td>
</tr>
<tr>
<td></td>
<td>±40</td>
<td>21.05</td>
<td>30.06</td>
<td>28.81</td>
<td>29.96</td>
<td>30.68</td>
<td>31.17</td>
<td>31.78</td>
</tr>
<tr>
<td>Barbara</td>
<td>±20</td>
<td>26.87</td>
<td>30.96</td>
<td>27.16</td>
<td>31.60</td>
<td>33.43</td>
<td>32.70</td>
<td>34.62</td>
</tr>
<tr>
<td></td>
<td>±30</td>
<td>23.35</td>
<td>28.68</td>
<td>25.44</td>
<td>29.24</td>
<td>31.02</td>
<td>30.75</td>
<td>32.41</td>
</tr>
<tr>
<td></td>
<td>±40</td>
<td>20.87</td>
<td>27.04</td>
<td>24.18</td>
<td>27.72</td>
<td>29.76</td>
<td>29.35</td>
<td>31.05</td>
</tr>
<tr>
<td>Boat</td>
<td>±20</td>
<td>26.88</td>
<td>32.53</td>
<td>31.15</td>
<td>32.48</td>
<td>33.64</td>
<td>33.37</td>
<td>33.84</td>
</tr>
<tr>
<td></td>
<td>±30</td>
<td>23.39</td>
<td>30.87</td>
<td>29.13</td>
<td>30.40</td>
<td>30.94</td>
<td>31.49</td>
<td>32.04</td>
</tr>
<tr>
<td></td>
<td>±40</td>
<td>20.92</td>
<td>29.75</td>
<td>27.46</td>
<td>29.02</td>
<td>30.11</td>
<td>30.17</td>
<td>30.72</td>
</tr>
</tbody>
</table>

Fig. 7. The denoising results of Boat with uniform noise by different models. (a) Boat corrupted by 40% of uniform noise (PSNR = 20.92 dB), (b) denoised image by the snowing approach (PSNR = 29.75 dB), (c) denoised image by RKHS (PSNR = 27.46 dB), (d) denoised image by BiShrink (PSNR = 29.02 dB), (e) denoised image by KSVD (PSNR = 30.11 dB) and (f) denoised image by BM3D (PSNR = 30.72 dB).
4.3. Uniform noise

Table 4 and Fig. 7 report the results of the snowing denoising algorithm on the Lena, Peppers, Barbara, and Boat images corrupted by various types of uniform noise. The BM3D gives best results in terms of PSNR, followed by SKR, KSVD,
Fig. 8. The denoising results of Lena with mixed noise by different models. (a) Lena corrupted by 20% of uniformly distributed impulse noise (Type II) and Gaussian noise with $s = 10$ and $\alpha = 128$ (PSNR = 17.98 dB), (b) denoised image by the snowing approach (PSNR = 32.30 dB), (c) denoised image by RKHS (PSNR = 31.87 dB), (d) denoised image by BiShrink (PSNR = 25.28 dB), (e) denoised image by KSVD (PSNR = 28.61 dB) and (f) denoised image by BM3D (PSNR = 30.66 dB).
Fig. 9. The denoising results of Boat with mixed noise by different models. (a) Boat corrupted by 30\% of uniformly distributed impulse noise (Type II) \((a = 128)\) and Gaussian noise with \(s = 20\) (PSNR = 15.95 dB), (b) denoised image by the snowing approach (PSNR = 28.37 dB), (c) denoised image by RKHS (PSNR = 28.73 dB), (d) denoised image by BiShrink (PSNR = 26.24 dB), (e) denoised image by KSVD (PSNR = 26.57 dB) and (f) denoised image by BM3D (PSNR = 27.79 dB).
Adaptive Weighted LS-SVM Based Snowing Model for Image Denoising

Fig. 10. The denoising results of Boat with mixed 4 noise by different models. (a) Boat corrupted by uniform noise in the interval \([-10, +10]\) and 10% of uniformly distributed impulse noise (Type II) \((a = 128)\) (PSNR = 21.20 dB), (b) denoised image by the snowing approach (PSNR = 33.90 dB), (c) denoised image by RKHS (PSNR = 32.24 dB), (d) denoised image by BiShrink (PSNR = 23.55 dB), (e) denoised image by KSVD (PSNR = 28.58 dB) and (f) denoised image by BM3D (PSNR = 30.72 dB).
Fig. 11. The denoising results of Lena with mixed 5 noise by different models. (a) Lena corrupted by uniform noise in the interval \([-10, +10]\), 10% of uniformly distributed impulse noise (Type II), and \(a = 128\) Gaussian noise with \(s = 10\) (PSNR = 20.34 dB), (b) denoised image by the snowing approach (PSNR = 32.99 dB), (c) denoised image by RKHS (PSNR = 32.41 dB), (d) denoised image by BiShrink (PSNR = 25.98 dB), (e) denoised image by KSVD (PSNR = 29.57 dB) and (f) denoised image by BM3D (PSNR = 31.94 dB).
Adaptive Weighted LS-SVM Based Snowing Model for Image Denoising

and the proposed snowing method. We note in Fig. 7 that the proposed snowing denoising method clearly give better results both visually and in terms of PSNR than the RKHS denoising algorithm.

4.4. Mixed noise

We included in the simulated experiments several images corrupted by mixed noise of various types as specified below:

- Mixed 1: 20% of uniform distributed impulse noise (Type II, $a = 128$) + Gaussian noise with $s = 10$.
- Mixed 2: 30% of uniform distributed impulse noise (Type II, $a = 128$) + Gaussian noise with $s = 20$.
- Mixed 4: uniform noise in the interval $[-10, +10]$ + 10% uniform distributed impulse noise (Type II, $a = 128$).
- Mixed 5: uniform noise in the interval $[-10, +10]$ + 10% uniform distributed impulse noise (Type II, $a = 128$) + Gaussian noise with $s = 10$.

The results are reported in Table 5 and Figs. 8–11. The proposed snowing method outperform the state of the art in three out five different types mixed noise (Mixed 1, Mixed 4, Mixed 5) and achieve comparable results for others (Mixed 2, Mixed 3) in terms of PSNR. Compared with BM3D and RKHS, the snowing method provides similar performance in terms of PSNR and gives some better visually quality, as shown in Figs. 8–11. This confirms the high performance of the adaptive weighted LS-SVM applied in the proposed snowing method.

5. Conclusion

A novel denoising algorithm was presented based on the use of adaptive weighted LS-SVM. The weighting scheme in fitness or errors modulated, elongated, elliptical contours spread along the direction of the local edge structure was developed for coping with the problems associated with the smoothing around edges, which is a common problem in almost all denoising algorithms. The iteratively adaptive reweighted LS-SVM is used to simulate the snowing model for image denoising. The performance of the proposed snowing denoising method was compared to the state of the art on the test images corrupted with various types of synthetic noise. For most experiments the proposed approach has performance comparable to the state of the art including RKHS, Shrink, BLS-GSM, KSVD, SKR, and BM3D. In case of the mixed noise, the snowing method seems to outperform the other denoising methods in terms of the PSNR and visual quality.
S. Zheng et al.

Acknowledgments

This paper is partially supported by the fund from China National Natural Science Fund (Project Nos. 60875009 and 60972162), and the State Scholarship Fund awarded by China Scholarship Council.

References

Adaptive Weighted LS-SVM Based Snowing Model for Image Denoising