Image and Vision Computing 27 (2009) 623–627

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Support vector regression based image denoising

ARTICLE INFO

Article history:
Received 23 July 2007
Received in revised form 17 December 2007
Accepted 9 June 2008

ABSTRACT

Support vector regression (SVR) has been applied for blind image deconvolution. In this correspondence, it is applied in the problem of image denoising. After training on noisy images with ground-truth, support vectors (SVs) are identified and their weights are computed. Then the SVs and their weights are used in denoising different images corrupted by random noise at different levels on a pixel-by-pixel basis. The proposed SVR based image denoising algorithm is an example-based approach since it uses SVs in denoising. The SVR denoising is compared with a multiple wavelet domain method (Besov ball projection). Some initial experiments indicate that SVR based image denoising outperforms Besov ball projection method on non-natural images (e.g. document images) in terms of both peak signal-to-noise ratio (PSNR) and visual inspection.

1. Introduction

Image denoising is an important image processing task, both as itself, and as a preprocessing in image processing pipeline. Considerable research has been performed [1–4]. Automatic image enhancement technique driven by an evolutionary optimization process [5] can be applied in the problem of denoising. Fuzzy logic filter [6] was proposed for image enhancement tasks that include image denoising. The wavelet transform-based approaches are one of those most successful methods for image denoising [7–11]. As a matter of fact, denoising is one of the most important applications of wavelets. Promising results have been achieved by wavelet based denoising.

For typical natural images, most of the wavelet coefficients have very small magnitudes, except for a few large ones that represent important high frequency features of the image such as edges. Since white noise is distributed evenly among all wavelet coefficients, removing small wavelet coefficients decreases most of the noise energy while preserving most of the image energy. This explains why as simple an operation as thresholding in the wavelet domain can effectively reduce noise while preserving image edges. Thresholding is just a very simple way to take advantage of the sparseness property of wavelet transform. Recently, multiple wavelet basis image denoising methods [11–13] have been proposed. Generally, these algorithms produce better denoising results than the conventional wavelet thresholding. In this correspondence, one of the multiple wavelet basis algorithms, namely, Besov ball projections (MWBBP) method [11], is compared with the proposed SVR denoising algorithm.

The support vector regression denoising algorithm is a machine learning based approach. We had applied it in blind image deconvolution [14]. It is extended in image denoising in this work. We formulate image denoising as a regression problem and use support vector regression in solving the problem. In this regression 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 original noise-free image. During the training phase, support vectors (SVs) and their weights are computed. Then, in the test phase, the SVs can perform denoising on images that were not in the training set and the noise level in the test image can be different from that in the training images. Though the test images are globally different from those in the training images, locally, they are very similar. This local similarity is the underlying assumptions of SVR based denoising and other example-based approaches. This local similarity is becoming more and more recognized in the literature [15–17]. Image coding using Vector Quantization (VQ) is also based on this local similarity property of images. An example of image coding by VQ is shown in Fig. 1 where (a) is the LENA image, (c) is the CAMERAMAN image and (b) is the LENA image coded by the CAMERAMAN image. In this case, the CAMERAMAN image is the codebook, the size of the vector is 16 (4×4 neighborhood). The local similarity of images allows one image be represented by another image. Therefore, the examples in the training images can be used in test images. The examples are not the entire image, rather, it refers to the patches (local neighborhood). SVR learning is local, not global. The localness guarantees the generalization. Since SVR is very powerful in generalizing to unseen data, the local similarity requirement is relaxed in practice. In some cases, even if the training image and test image are quite different such as document image and nature image, SVR can still efficiently remove noise as shown in the experiments in Section 3.

The wavelet characteristics of certain types of images, such as document images, are very different from those of natural images that have a sparse representation. Therefore, wavelet-domain denoising on these images that do not have spare representation is not as efficient as it is on natural images on which the spareness assumption holds. On the other hand, SVR based image denoising can easily overcome such a limitation simply by training a model

0262-8856/$ - see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2008.06.006

Fig. 1. (a) The LENA image. (b) The LENA image that is coded by the CAMERAMAN image, patch size is 4 by 4. (c) The CAMERAMAN image.

from the corresponding examples of the non-natural images (e.g. document image).

This correspondence is organized as follows. Section 2 presents the proposed SVR based image denoising algorithm. Comparative experiments with Besov ball projections method on a variety of images at different noise levels are shown in Section 3. In Section 4, a concluding remark is made.

2. Support vector regression based image denoising

Given training data $(X_1;y_1),\ldots,(X_l;y_l)$, where X_i are input attribute (feature) vectors (in noisy image) and y_i are the associated target values (in the original image), traditional linear regression looks for a linear function W^TX+b that minimizes the mean square error:

$$\min_{w,b}\sum_{i=1}^{l}(y_i-(W^TX_i+b))^2. \tag{1}$$

where W is the corresponding weight vector for the feature vector X and b is the intercept (a constant term). Intuitively, for 1D case, linear regression is to find a line that crosses all the sample points. A linear regression is not sufficient if the input data is not linearly distributed. Support vector machines use a mapping function $\phi(x)$ to map the data into a higher dimensional space, where a linear function is adequate. Once data is mapped into the high-dimensional space, overfitting can occur. Consequentially, the trained function has poor generalization on test data that usually will never be identical to the training data. To limit overfitting thus to make the trained function more applicable on new data, a soft margin and a regularization term are added into the objective function. Support vector regression [18] solves the following optimization problem in order to find a linear function W^TX+b:

$$\min_{W,b,\xi,\xi^*}\frac{1}{2}W^TW+C\sum_{i=1}^{l}(\xi_i+\xi_i^*) \tag{2}$$

$$\text{subject to } y_i-(W^T\phi(X_i)+b)\leqslant\epsilon+\xi_i,$$
$$(W^T\phi(X_i)+b)-y_i\leqslant\epsilon+\xi_i^*,$$
$$\xi_i,\xi_i^*\geqslant 0,\ i=1,\ldots,l.$$

where W is the corresponding weight vector for the feature vector X, b is the intercept (a constant term), ξ_i is the upper training error, (ξ_i^* is the lower training error) subject to the ϵ – insensitive tube $|\,y-(W^T\phi(X)+b)\,|\leqslant\epsilon$ and ϵ is a threshold that specifies the width of the tube. The cost function ignores any training data that lies within the ϵ – insensitive tube. This soft margin technique has the advantage of tolerating mislabeled samples in the training set, thus increases the robustness of learning. In this objective function, $\frac{1}{2}W^TW$ can be viewed as a regularization term that effectively smoothes the function $W^T\phi(X_i)+b$ to limit overfitting. Effectively, within the ϵ – insensitive tube, the regularization term constrains the line to be as flat as possible. The norm W^TW measures the flatness of the line.

Similar to support vector classification, W may be a huge vector variable because $\phi(X)$ maps data to a higher dimensional space, the dual problem is solved instead:

$$\min_{\alpha,\alpha^*}\frac{1}{2}(\alpha-\alpha^*)^TQ(\alpha-\alpha^*)+\epsilon\sum_{i=1}^{l}(\alpha_i+\alpha_i^*)+\sum_{i=1}^{l}y_i(\alpha-\alpha^*) \tag{3}$$

$$\text{subject to}\sum_{i=1}^{l}(\alpha-\alpha^*)=0,0\leqslant\alpha,\alpha^*\leqslant C,i=1,\ldots,l$$

where $Q_{ij}=K(X_i,X_j)\equiv\phi(X_i)^T\phi(X_j)$. The derivation of the dual is the same as in support vector classification. The primal-dual relation shows that

$$W=\sum_{i=1}^{l}(-\alpha_i+\alpha_i^*)\phi(X_i) \tag{4}$$

Thus, the approximate function is

$$\sum_{i=1}^{l}(-\alpha_i+\alpha_i^*)K(X_i,X)+b \tag{5}$$

From Eq. (5), it is clear that the direct mapping $\phi(X)$ is not used. Since support vector algorithms only depend on dot products between input patterns, the mapping is actually implicit through kernels $K(X_i,X)$. Commonly used kernel functions are linear, polynomial, Gaussian, sigmoid etc. Eq. (5) also shows that SVR is significantly different from traditional linear regression/filtering. The W is not computed explicitly. Instead, its dual is solved. The complexity of SVR is much higher than traditional linear regression. The complexity is determined by the number of SVs. For traditional filtering, it is determined by the PSF support of the filter, such as 5 by 5 or 7 by 7. Moreover, SVR is adaptive, those SVs (representative examples) that are similar to a test pattern dominate the output since they are closer and $K(X_i,X)$ is larger. $K(X_i,X)$ can be interpreted as weights. For other test patterns that are different, the dominant SVs are also different. For each test pattern, the contribution from each SV is different, it is determined by the similarity between the test pattern and SV. On the other hand, a traditional linear regression/filter does not have such adaptiveness. The weights/coefficients in those filters are fixed.

To apply SVR in image denoising is straightforward. The input feature vector X for a pixel in a noisy image is formed by its neighborhood pixels in the noisy image. The target value y is the corresponding pixel value in the noise-free image. Therefore, for any pixel except for those near the boundaries of the image, a $X-y$ pair is made and the training set is made from the $X-y$ pairs. The size of the window may be interpreted as the image denoising filter point spread function (PSF) support. Usually it is small such as a 3 by 3 window. A large value is unnecessary and inappropriate since the correlation between pixels decreases as the pixels are separated further apart. Thus the pixels on the boundary of a large window have little information about the central pixel in the window. Moreover, a larger window increases the dimensionality of the feature vector which in turn increases the time for training/

testing. Moreover, similar to adaptive Wiener filter, a large denoising filter PSF support tends to oversmooth the image. As a result, the edge in the denoised image might appear blurry.

3. Experiments

In the following experiments, the images are taken from the USC-SIPI Database [19], which is widely used in image processing research. In order to demonstrate the generalization ability of SVR, the training set is intentionally limited to very few images i.e. one or two images. All the test images are different from the training images. Since there are no document image in the USC-SIPI database, a number of scanned document images were used in the experiments. One of those images is shown in Fig. 3. In all the experiments, the size of the neighborhood window is 3×3. The training images are LENA, HOUSE, a document image, and combinations of these images. Random Gaussian noise is added to the training images with a variance of 0.01 except for the document image, where the noise variance was 0.05 so that the texts become illegible. In the test images, the noise variance is from 0.01 to 0.04 in steps of 0.01 for the non-document images and the noise variances in the document images varies from 0.05 to 0.08 in steps of 0.01. LibSVM [20] is an implementation of SVR and it is used in the experiments.

The peak signal-to-noise ratio (PSNR) is often used as an objective quality measurement of the denoised image. Given the original image, PSNR is defined as:

$$\mathrm{PSNR} = 10\log_{10}\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}255^2}{\sum_{i=1}^{M}\sum_{j=1}^{N}(f(i,j)-\hat{f}(i,j))^2} \quad (6)$$

where $\hat{f}(i,j)$ is the denoised image, and $f(i,j)$ is the original image. $M \times N$ is the size of the image.

Table 1 summarizes the result on some of the test images. In order to compare the performance of the algorithms on different types of images, the test images are divided into two classes: natural image and non-natural image (i.e. document image and texture image). For each test image, four noise levels are applied. PSNRs of the denoised images in each class at each noise level is averaged. There are two SVR models used: one is trained on the LENA image, the other is trained on both the LENA image and the HOUSE image. It can be observed that MWBBP works better on the natural images. On the other hand, SVR works better on the document image and the texture image, whose wavelet domain characteristics are significantly different from those of natural images. Fig. 2 shows the comparative results on the CAMERAMAN image. There are no significant visual differences in this results.

Table 1 also shows that an expanded training set usually results in a better SVR model for denoising since the PSNR of the denoised images by the LenaHouse SVR model are generally higher than those by the SVR trained on the LENA alone. This is common for supervised learning: better performance is achieved as more examples are available.

Table 1
PSNR (dB) comparisons of MWBBP and SVR (using two models) based image denoising

Noise variance	Noisy	MWBBP	SVR	
			Lena	Lenahouse
Natural images (Aerial, Boat, Cameraman, Baboon, Peppers)				
0.01	20.14	26.04	25.13	25.81
0.02	17.27	24.52	23.63	24.02
0.03	15.62	23.70	22.53	22.79
0.04	14.48	23.13	21.69	21.89
Non-natural images (Texture, Document)				
0.05	17.70	20.27	20.75	21.20
0.06	15.88	19.27	20.34	20.59
0.07	14.80	18.76	19.93	20.09
0.08	13.93	18.31	19.63	19.65

To further demonstrate that a specific model improves the denoising result, another SVR model is trained from a document image that resembles the test document images. For this group of experiments, the natural images are made up of TIFFANY and ELAINE; the non-natural image is made up of two document images. As shown in Table 2, the specific SVR model resulted in a larger performance difference both in terms of PSNR and visual inspection on the document images. On the other hand, the document SVR does not work well on natural images. However, the performance on the natural images is improved by using a third SVR model that was trained on both the document image and a photographic image (LENA image). This highlights the advantage of SVR based image denoising: it can handle different types of images by including the corresponding images in the training set. Wavelet based method does not share this property. It only works well for natural images. For doc-

Table 2
PSNR (dB) comparisons of two SVR models and MWBBP on natural images and document images

Noise variance	Noisy	MWBBP	SVR		
			Doc	Lena	Lenahouse
Natural images (Tiffany, Elaine)					
0.01	20.45	26.72	23.40	24.72	25.51
0.02	17.66	25.28	21.77	23.67	23.43
0.03	16.11	24.32	20.59	22.95	21.99
0.04	14.99	23.50	19.58	22.31	20.82
Non-natural images (Doc1, Doc2)					
0.05	15.57	17.50	20.82	16.65	20.56
0.06	14.87	16.86	20.40	16.63	20.15
0.07	14.18	16.27	19.83	16.49	19.61
0.08	13.63	15.75	19.31	16.34	19.11

Fig. 2. (a) The noisy CAMERAMAN image, Gaussian noise variance is 0.04, PSNR = 14.71 dB. (b) The denoised image by MWBBP, PSNR = 22.61 dB. (c) The denoised image by Lena Doc SVR model, PSNR = 21.30 dB.

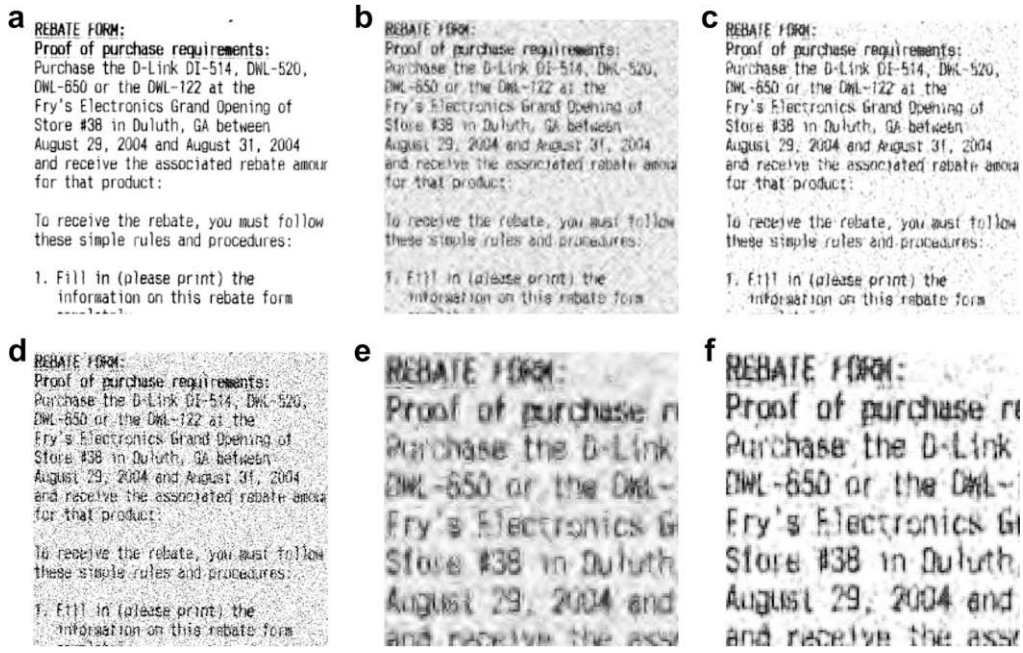

Fig. 3. (a) The test document image. (b) The denoised image by MWBBP, PSNR = 15.53 dB. (c) The denoised image by doc SVR model, PSNR = 18.89 dB. (d) The noisy Document image, Gaussian noise variance is 0.08, PSNR = 13.60 dB. (e) A zoomed-in region of (b). (f) A zoomed-in region of (c).

uments images and other types of images that do not have sparseness property, wavelet based approach is not efficient.

Fig. 3 shows the comparative results on one of the test document images. Significant visual differences in the results are observed. The result by the SVR trained on the document image clearly achieved a better result. In the MWBBP result, there are lot of visible distortions around the text. Such distortions are not seen in the SVR result. To better illustrate the difference, Fig. 3 (e) and (f) show the zoomed-in regions of the MWBBP and SVR results. In the region where there is no text, the SVR result is much cleaner.

4. Conclusion

In this correspondence, SVR is applied to a new application in image processing, namely, image denoising. Experiments show that SVR can learn a generally applicable model, even one trained on a very small data set. The learned SVR models are those identified support vectors (SVs) and their weights. The algorithm has been tested on a variety of images (texture, aerial, natural, document). Initial experiments already suggest that SVR based image denoising achieves better performance on non-natural images such as document images than wavelet-domain based approaches e.g. the multiple wavelet basis Besov ball projections method. The performance of SVR based image denoising is completely determined by the training since SVR is one of those supervised learning algorithms. It is an example-based approach since SVs are examples. When the model is trained on a larger data set, as with other machine learning approaches, the result is expected to be improved since there are more examples available. A more specific training set can usually generate a better model that produces better denoising on the images that are similar to those in the training set since the example patches in the training images match better with those patches in the test images. In real applications, in order to get a SVR model that can work for a variety of images (e.g. document images and photographic images), one can train a mixed model with both document images and photographic images. That is to say, SVR based image denoising can adapt to different types of images simply by including corresponding examples in the training set. Wavelet based approaches do not share such property. However, wavelet based approaches require no training and have significant lower complexity. The study in this correspondence indicates that a machine learning approach can outperform DSP based approach for image denoising on certain types of images. This is because that SVR or other example-based approaches do not make explicit assumption about the characteristics of images such as the sparseness property of images in wavelet transform, which is required for wavelet based image denoising.

References

[1] O.M. Lysaker, S. Osher, X. Tai, Noise removal using smoothed normals and surface fitting, IEEE Trans. Image Process. 13 (10) (2004) 1345–1357.
[2] M. Ghazel, G.H. Freeman, E.R. Vrscay, Fractal image denoising, IEEE Trans. Image Process. 12 (12) (2003) 1560–1578.
[3] S. Kuo, J.D. Johnston, Spatial noise shaping based on human visual sensitivity and its application to image coding, IEEE Trans. Image Process. 11 (5) (2002) 509–517.
[4] J. Xie, D. Zhang, W. Xu, Spatially adaptive wavelet denoising using the minimum description length principle, IEEE Trans. Image Process. 13 (2) (2004) 179–187.
[5] C. Munteanu, A. Rosa, Gray-scale image enhancement as an automatic process driven by evolution, IEEE Trans. Syst. Man Cybern. B 34 (2) (2004).
[6] F. Farbiz, M.B. Menhaj, S.A. Motamedi, M.T. Hagan, A new fuzzy logic filter for image enhancement, IEEE Trans. Syst. Man Cybern. B 30 (1) (2000).
[7] A.A. Bharath, J. Ng, A steerable complex wavelet construction and its application to image denoising, IEEE Trans. Image Process. 14 (7) (2005) 948–959.
[8] J. Zhong, R. Ning, Image denoising based on wavelets and multifractals for singularity detection, IEEE Trans. Image Process. 14 (10) (2005) 1435–1447.
[9] J. Portilla, V. Strela, M.J. Wainwright, E.P. Simoncelli, Image denoising using scale mixtures of gaussians in the wavelet domain, IEEE Trans. Image Process. 12 (11) (2003) 1338–1351.
[10] S.G. Chang, B. Yu, M. Vetterli, Spatially adaptive wavelet thresholding with context modeling for image denoising, IEEE Trans. Image Process. 9 (9) (2000) 1522–1531.
[11] H. Choi, R.G. Baraniuk, Multiple wavelet basis image denoising using Besov ball projections, IEEE Signal Process. Lett. 11 (9) (2004) 717–720.
[12] P. Ishwar, K. Ratakonda, P. Moulin, N. Ahuja, Image denoising using multiple compaction domains, in: Proc. IEEE Int. Conf. Acoust., Speech, Signal Proc. – ICASSP'98, Seattle, WA, 1998, pp. 1889–1892.

[13] S.P. Ghael, A.M. Sayeed, R.G. Baraniuk, Improved wavelet denoising via empirical Wiener filtering, in: Proc. SPIE, vol. 3169, Wavelet Appl. Signal Image Process, 1997.
[14] D. Li, R. Mersereau, S. Simske, Blind image deconvolution through support vector regression, IEEE Trans. Neural Netw. 18 (3) (2007) 931–935.
[15] S.K. Alexander, S. Kovacic, E.R. Vrscay, A model for image self-similarity and the possible use of mutual information, in: Proc. the 2007 European Signal Process. Conf. (EUSIPCO-2007), Poznan, Poland, 2007, pp. 975–979.
[16] A. Buades, B. Coll, J.-M. Morel, A review of image denoising algorithms, with a new one, SIAM J. Multiscale Model. Simulation 4 (2) (2005) 490–530.
[17] A. Buades, B. Coll, J.-M. Morel, A non-local algorithm for image denoising, in: Proc. IEEE Comp. Society Conf. Comp. Vision Pattern Recognit. (CVPR'05), vol. 2, 2005, pp. 60–65.
[18] V. Vapnik, The Nature of Statistical Learning Theory, Springer Verlag, New York, 1995.
[19] University of Southern California – Signal & Image Processing Institute - The USC-SIPI Image Database. Available from: <http://sipi.usc.edu/services/database/Database.html>.
[20] C.C. Chang, C.J. Lin, LIBSVM: a library for support vector machines. Software available at: <http://www.csie.ntu.edu.tw/cjlin/libsvm> (2001).

Dalong Li
Pegasus Imaging Corporation,
4001 Riverside Drive Tampa, FL, 33603,
United States
E-mail address: dalongli@gmail.com