ASSIGNING AUTOMATIC REGULARIZATION PARAMETERS IN IMAGE RESTORATION

Ignazio Gallo and Elisabetta Binaghi
Università degli Studi dell'Insubria, via Ravasi 2, Varese, Italy

Keywords: Regularization profile, Image restoration, Adaptive regularization, Neural networks.

Abstract: This work aims to define and experimentally evaluate an adaptive strategy based on neural learning to select an appropriate regularization parameter within a regularized restoration process. The appropriate setting of the regularization parameter within the restoration process is a difficult task attempting to achieve an optimal balance between removing edge ringing effects and suppressing additive noise. In this context,in an attempt to overcome the limitations of trial and error and curve fitting procedures we propose the construction of the regularization parameter function through a training concept using a Multilayer Perceptron neural network. The proposed solution is conceived independent from a specific restoration algorithm and can be included within a general local restoration procedure. The proposed algorithm was experimentally evaluated and compared using test images with different levels of degradation. Results obtained proven the generalization capability of the method that can be applied successfully on heterogeneous images never seen during training.

1 INTRODUCTION

Restoration of blurred and noisy images requires the adoption of a regularization approach based on the specification of a cost function consisting of a least square term and a regularization term (Lagendijk and Biemond, 2001; Andrews and Hunt, 1977). The role of the two terms is controlled by the regularization parameter. The appropriate setting of the regularization parameter within the restoration process achieves an optimal balance between removing edge ringing effects and suppressing additive noise.

The critical problem of optimally estimating the regularization parameter has been investigated in depth in literature.

Previous works addressed the problem by proposing a regularization profile where the local parameter value is expressed as a monotonically decreasing function of the local variance (Qian and Clarke, 1996; Lagendijk et al., 1988; Katsaggelos and Kang, 1995). In particular Perry and Guang (Perry and Guan, 2000) proposed a perceptually motivated solution in which the constraint values decrease linearly as the logarithm of the local regional variance increases. Proceeding from these results, in a previous work we defined a statistics-based procedure assigning a separate parameter to each image pixel according to local variance computed in the neighborhood of the pixel to be examined.

The regularization parameter is specified for each pixel as $\lambda(x,y) = Y(S(x,y))$ where $S(x,y)$ is the local variance of the degraded input image g varying from S_{min} to S_{max}, while Y corresponds to the log-linear function:

$$Y(S;\lambda_{min},\lambda_{max}) = \quad (1)$$
$$= \frac{\lambda_{min}-\lambda_{max}}{S_{max}-S_{min}}(log(S)-S_{min})+\lambda_{max}$$

To determine the function Y univocally and then build a specific *regularization profile*, we need to fix values λ_{min} and λ_{max} corresponding to S_{max} and S_{min} respectively.

The present work proposes a novel approach to regularization profile estimation based on the approximation capability of the supervised neural learning technique based on Multilayer Perceptron Network (MLP). The interest in this novel strategy mainly lies in the possibility of inducing the regularization function from a set of training images, directly mapping local variance values and/or other image features to regularization parameters without requiring trial and error and curve fitting procedures.

Gallo I. and Binaghi E. (2009).
ASSIGNING AUTOMATIC REGULARIZATION PARAMETERS IN IMAGE RESTORATION.
In *Proceedings of the Fourth International Conference on Computer Vision Theory and Applications*, pages 74-77
DOI: 10.5220/0001785500740077

2 THE PROPOSED METHOD

The proposed method for regularization parameter assignment is conceived as a pre-processing phase within a general restoration strategy. To make the paper self-contained and to exploit all the ingredients of the overall strategy adopted in the experimental part of the work, we briefly outline the salient aspects of a restoration strategy developed and presented in a previous study. It consists of a neural iterative method which uses a gradient descent algorithm to minimize a local cost function derived from a traditional global constrained least square measure (Gallo et al., 2008).

In particular, the degradation measure to be minimized is a local cost function $E(x,y)$ defined at any point (x,y) in an $M \times N$ image:

$$E(x,y) = \frac{1}{2}\left[g(x,y) - h * \hat{f}(x,y)\right]^2 + \frac{1}{2}\lambda(x,y)\left[d * \hat{f}(x,y)\right]^2 \quad (2)$$

where $h * \hat{f}(x,y)$ denotes the convolution between a blur filter h centered in a point (x,y) of the restored image $\hat{f}$ and the restored image $\hat{f}$ itself; $d * \hat{f}(x,y)$ denotes the convolution between a high-pass filter d centered in a point (x,y) of the restored image $\hat{f}$ and the restored image $\hat{f}$ itself.

A multilayer perceptron model, trained with the supervised back propagation learning algorithm (Rumelhart et al., 1986), was adopted to compute the regularization parameter based on specific local information extracted from the degraded image $g(x,y)$ previously scaled in a range $[0,1]$. The neural learning task accomplished within the neural training phase can be formulated as a search for the best approximation of the function $\lambda(x,y) = Y(\mathbf{S}^m)$ where $\mathbf{S}^m$ represents a set of statistical measures extracted directly from the degraded image. The present work uses $\mathbf{S}^m = (S_1(x,y), S_2(x,y), S_3)$ where S_1 is the local variance computed directly on the degraded image and S_2 is the local variance computed on the degraded image smoothed with a Gaussian low-pass filter. In particular we use the variance calculated in a window measuring 3×3 as statistical measure S_1 and the variance calculated in a window measuring 5×5 as statistical measure S_2.

The joint use of S_1 and S_2 is motivated by the need to preserve image features during restoration. S_3 is a constant value derived from the histogram of S_1. In particular S_3 is the value of variance corresponding to the peak value in the histogram. This is an important feature because it is directly correlated to the amount of noise in the degraded image and we know that λ should be proportional to the amount of noise in the data (Inoue et al., 2003).

The training set presented to the neural network for the supervised learning task is constituted by N pairs of elements $((S_1,S_2,S_3),\hat{\lambda}_j)^n$ where $n = 1,\ldots,N$. The second component of training examples $\hat{\lambda}_j$ are the expected outputs for the corresponding input components and are constituted by regularization values obtained from successful restoration processes as explained in section 2.1.

The trained network is expected to be able to generalise, i.e. to associate adequate regularization values with degraded input images never seen during training.

2.1 Regularization Profile Construction

Representative samples of the function $\lambda(x,y) = Y(\mathbf{S}^m)$ are necessary if we want to train a neural network that represents it. Algorithm 1 describes in details the method used to compute a sampling of this function while Figure 1 shows an example of tabular data obtained from the same algorithm. Representative samples of the function $\lambda(x,y) = Y(\mathbf{S}^m)$ must be presented to the network during the training phase for learning. Algorithm 1 describes the procedure used to build the sample pairs $((S_1,S_2,S_3),\hat{\lambda}_j)^n$.

λ	0.0001	0.0002	0.0004	0.0008	0.0016	0.0032	0.0065	0.0130	0.0262	0.0525
0.2	2.25	**7.06**	7.54	3.18	0.28	-0.59	-0.29	-1.16	-2.39	-1.5
0.18	**2.26**	7.04	7.63	3.31	0.45	-0.42	-0.17	-0.97	-2.15	-1.4
0.15	2.25	7	**7.67**	3.44	0.61	-0.25	-0.03	-0.76	-1.9	-1.26
0.13	2.24	6.92	7.66	3.56	0.78	-0.07	0.13	-0.52	-1.62	-1.07
0.1	2.21	6.8	7.59	**3.67**	0.97	0.12	0.31	-0.23	-1.27	-0.81
0.01	1.28	3.04	3.44	2.48	**1.46**	**1.07**	**1.19**	**1.25**	**1**	**1.11**
0.001	-0.03	-0.16	-0.08	0	0.16	0.25	0.28	0.47	0.78	0.67
0.0001	-0.24	-0.59	-0.55	-0.38	-0.08	0.08	0.09	0.3	0.69	0.55
0.00001	-0.27	-0.63	-0.6	-0.42	-0.11	0.06	0.07	0.28	0.68	0.54
0.000001	-0.27	-0.63	-0.6	-0.42	-0.11	0.06	0.07	0.28	0.68	0.54

Ranges of variance

Figure 1: The columns list the ISNR values obtained restoring the image with a set of different constant λ values. Each column identifies a group of pixels with variance included in a prefixed range. The best λ value for a given range of variance corresponds to the highest ISNR values obtained (in bold) .

Our approach compares the improvement in signal to noise ratio (ISNR) measures calculated on a set of restored pixels $\hat{f}(x,y)$, all having a statistical measure included in an interval $I_i \leq S(x,y) < I_{i+1}$. Then we choose the best $\hat{\lambda}$ corresponding to the best ISNR. The result of this approach is an approximation of the function $\lambda(x,y) = Y(\mathbf{S}^m)$ representing the regularization profile with which to compute the regularization parameter.

The training set is built applying Algorithm 1 to a set of images representative of a given domain. To be exhaustive, each image in turn must be degraded with different levels of noise and different kinds of blur.

Algorithm 1 - function $\lambda(x,y) = Y(\mathbf{S}^m)$ sampling.

Require: to select a degraded image $g(x,y)$ and the corresponding undistorted image $f(x,y)$;
Require: to break in R regular intervals $I_1,\ldots,I_R$ the range $[log(S_{1,min}), log(S_{1,max})]$;
Require: to define a set of L regularization parameters $\Lambda = \{\lambda_1,\ldots,\lambda_L\}$;
1: **for** $j = 1$ to L **do**
2: **for** $s = 1$ to R **do**
3: restore all the pixels belonging to the interval I_s using λ_j as regularization parameter;
4: select the best parameter $\hat{\lambda}_j$, for all the pixels belonging to the interval I_s, choosing what has maximized the ISNR measure;
5: **end for**
6: **end for**
7: Pattern set extraction

3 EXPERIMENTS

The proposed algorithm was experimentally evaluated and compared using the six test images shown in Figure 2. Images (a-c) were used to generate the training set while images (d-f) were used as a test.

In the experiments all the test images were degraded by a Gaussian filter having standard deviation $\sigma_x = \sigma_y = 1.0$ and corrupted by Gaussian noise having standard deviation $\sigma = 5, 15, 25$. During the training set construction, the blurred images (a-c) of Figure 2 without added noise, were also used.

Referring to Algorithm 1, the parameters used in the experiments were:

- $R = 15$: the range of variance of each image used in training was split into 15 intervals;
- for each interval, up to 300 patterns were selected;
- $L = 15$: the constant regularization parameters used in each interval $\Lambda = \{0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.028, 0.046, 0.064, 0.082, 0.1, 0.12, 0.14, 0.16, 0.18, 0.2\}$;
- the restoration algorithm was run for 20 iterations for each different λ_j applied to each range I_i considered.

The input pattern was created by reading pixel values in a window 3×3 centered on a particular pixel in the two images $S_1(x,y)$ and $S_2(x,y)$. To these values we added S_3 which is a constant value for each image. In this way the network used has 19 input neurons, a single output neuron and a hidden layer with 38 neurons. The network was trained for 1000 epochs over all the training examples.

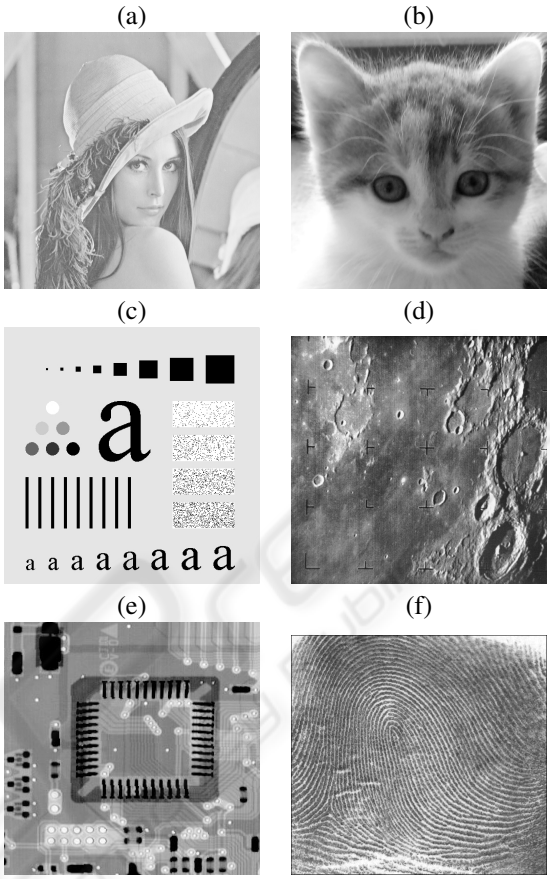

Figure 2: Undistorted images used in training: Lena (a), Cat (b), A (c); undistorted test images: Moon (d), Board (e) and Fingerprint (f).

To evaluate the restoration performances of our approach quantitatively, the well-known Improvement in Signal-to-Noise Ratio (ISNR) measure (Banhom and k. Katsaggelos, 1997) was adopted. This can be estimated as follows:

$$ISNR = 10\log_{10}\left(\frac{\sum_{x,y}(f(x,y)-g(x,y))^2}{\sum_{x,y}(f(x,y)-\hat{f}(x,y))^2}\right) \quad (3)$$

where $g(x,y)$ is the given degraded image and $\hat{f}(x,y)$ is the restored image. This measure can only be evaluated for controlled experiments in which the blur and noise have been synthetically introduced. The maximally achievable ISNR depends strongly on the content of the image, the type of blur considered and the signal-to-noise ratio of the blurred image.

Table 1 summarizes the ISNR values obtained restoring all the images shown in Figure 2 with different levels of degradation. The ISNR values are always positive except for one case where the image to be restored was corrupted by a smaller amount of noise. However, comparing the ISNR values obtained with

our algorithm (Table 1) and the ISNR values obtained with the algorithm proposed in (Gallo et al., 2007) and showed in Table 2, the results are very similar.

Table 1: ISNR measure obtained from the restoration of all the pictures shown in Figure 2. The neural network was trained for 1000 epochs on a subset of pixels extracted from images Lena, A, and Cat. Finally the same network was tested on images Moon, Fingerprint and Board.

Training			Test		
Image	noise	isnr	Image	noise	isnr
Lena	5	2.05	Moon	5	0.42
Lena	15	7.35	Moon	15	4.16
Lena	25	9.40	Moon	25	6.45
A	5	0.24	Fingerpr.	5	-2.51
A	15	0.87	Fingerpr.	15	1.66
A	25	1.53	Fingerpr.	25	3.02
Cat	5	2.07	Board	5	1.72
Cat	15	7.19	Board	15	3.85
Cat	25	9.09	Board	25	5.41

Table 2: Best restoration results obtained by a different algorithm using a trial and error approach.

Image	noise (σ)	λ_{min}	λ_{max}	ISNR
Cat	5	0.000001	0.1	2.33
Cat	15	0.000001	0.2	7.31
Cat	25	0.15	0.2	9.13

4 CONCLUSIONS

As seen in our experimental context, the proposed method for automatically assigning regularization parameters during restoration produces successful results and can be conceived as a general model for adaptive regularization assignment within a restoration procedures. The generalization capability of the network used for estimating the regularization profile was proven using a different set of images for training and test phases. Results obtained ensure that the solution proposed can be conceived for operational tools addressing collections of heterogeneous images without the need for retraining.

Future works will attempt to improve the feature extraction/selection task to capture essential representative image features and investigate the generalization capacity of the neural model in depth in relation to different imagery.

REFERENCES

Andrews, H. C. and Hunt, B. R. (1977). *Digital Image Restoration*. Prentice-Hall, New Jersey.

Banhom, M. R. and k. Katsaggelos, A. (1997). Digital image restoration. *IEEE Signal Processing Mag.*

Gallo, I., Binaghi, E., and Macchi, A. (2007). Adaptive image restoration using a local neural approach. In *2nd International Conference on Computer Vision Theory and Applications*.

Gallo, I., Binaghi, E., and Raspanti, M. (2008). Semi-blind image restoration using a local neural approach. In *Signal Processing, Pattern Recognition, and Applications*, pages 227–231.

Inoue, K., Iiguni, Y., and Maeda, H. (2003). Image restoration using the rbf network with variable regularization parameters. *Neurocomputing*, 50:177–191.

Katsaggelos, A. K. and Kang, M. G. (1995). Spatially adaptive iterative algorithm for the restoration of astronomical images. *Int. J. Imaging Syst. Technol.*, 6:305–313.

Lagendijk, R. L. and Biemond, J. (2001). *Iterative Identification and Restoration of Images*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Lagendijk, R. L., Biemond, J., and Boekee, D. E. (1988). Regularized iterative image restoration with ringing reduction. *Acoustics, Speech, and Signal Processing*, 36(12):1874–1888.

Perry, S. W. and Guan, L. (2000). Weight assignment for adaptive image restoration by neural networks. *IEEE Trans. on Neural Networks*, 11:156–170.

Qian, W. and Clarke, L. P. (1996). Wavelet-based neural network with fuzzy-logic adaptivity for nuclear image restoration. In *Proceedings of the IEEE*, volume 84, pages 1458–1473.

Rumelhart, H., Hinton, G. E., and Williams, R. J. (1986). Learning internal representation by error propagation. *Parallel Distributed Processing*, pages 318–362.