Defocused Image Restoration Using RBF Network and Kalman Filter

Yugang Jiang, Qing Wu and Ping Guo
Department of Computer Science
Beijing Normal University
Beijing, 100875, P. R. China
pguo@ieee.org

Abstract – A novel defocused image restoration technique is proposed, which is based on radial basis function (RBF) neural network and Kalman filter. In this technique, firstly a RBF neural network is trained in wavelet domain to estimate defocus parameter. After obtaining the point spread function (PSF) parameter, Kalman filter is adopted to complete the restoration. We experimentally illustrate its performance on simulated data and compare it with other methods. Results show that the proposed PSF parameter estimation technique is more robust to noise.

Keywords: Wavelet Transform, Neural Network, Kalman Filter, Defocused Image Restoration.

1 Introduction

Defocus blur is caused by inaccurate focus in imaging systems like cameras, microscopes or other optical equipments. These situations exist in remote sensing, cosmos detecting, traffic control, medical treatment, analysis of micro phenomenon and many other important fields of research or applications. Therefore restoration of defocused image has a theoretical significance as well as a prospect of real applications.

In general, discrete model for a linear degradation caused by blurring can be given by the following equation [1],

\[ y(i, j) = h(i, j) \ast \ast f(i, j) + n(i, j), \]

where \( \ast \ast \) indicates two-dimensional convolution, \( f(i, j) \) represents an original image, \( y(i, j) \) is the degraded image, \( h(i, j) \) represents the two-dimensional point spread function (PSF), and \( n(i, j) \) is the additive noise. As for defocus blur, PSF is modeled as a uniform intensity distribution within a circular disk [1],

\[ h(i, j) = \begin{cases} \frac{1}{\pi R^2} & \sqrt{i^2 + j^2} \leq R \\ 0 & \text{otherwise} \end{cases} \]

where disk radius \( R \) is the only unknown parameter for this type of blur.

Many existing image restoration algorithms assume that the PSF is known, but in practical it is not always the case. The restoration without knowing of the PSF is called blind image restoration. Fourier methods can be used to estimate the defocus parameter \( R \) through calculating a ratio of power of high frequencies portion to that of low frequencies portion [2]. However, a main drawback of the method is its bad noise immunity. To solve this problem, an efficient method for estimating parameter \( R \) is proposed in this paper. Firstly we construct feature vectors of several blurred images with known defocus radius \( R \) in wavelet domain, then a radial basis function (RBF) neural network is trained using the vectors as inputs and defocus parameters as outputs. After the network is trained, for a blind defocused image, \( R \) can be estimated through calculating the feature vector and using it as input of the trained neural network.

With known radius \( R \), many traditional algorithm could be applied to restore the degraded image. Among them, Kalman filter is an optimal recursive filter algorithm to the discrete-data linear filtering problem. Because this algorithm does not require all previous data to be kept in storage and reprocessed every time a new measurement is taken, it can save computation and storage requirements in a large amount. So Kalman filter is adopted to complete the restoration.

The organization of this paper is as follows: In Section 2, we briefly introduce the proposed parameter estimation technique. Section 3 introduces Kalman filter and its application to image restoration. The restoration results and comparison are given in Section 4. Conclusions are presented in the last Section.

2 Estimating the defocus parameter

We propose and implement a parameter estimation technique in this section. Fig. 1 shows the description of this technique. In the first phase a RBF neural network is designed and trained. In the second phase \( R \) can be estimated using the trained neural network. A brief description of this technique is given in the following paragraphs.

2.1 Relationship between wavelet coefficients and \( R \)

The wavelet transform provides a powerful and versatile framework for image processing. It is widely used in the fields of image de-noising, compression, fusion, etc [3].

The two-dimensional discrete wavelet transform (DWT) hierarchically decompose an input image into a series of successively lower resolution images and their associated de-
Defocused image with unknown \( R \)

Parameter Estimating Phase

2D Wavelet Transform

Extract Features in Wavelet Domain

Training a RBF net

Parameter \( R \)

Images with known \( R \) for training

2D Wavelet Transform

Extract Features in Wavelet Domain

Training Phase

Defocused image with unknown \( R \)

Parameter Estimating Phase

Figure 1: Defocus parameter estimation process

tail images. DWT is implemented by a set of filters, which are convolved with the image rows and columns. An image is convolved with low-pass and high-pass filters and the odd samples of the filtered outputs are discarded resulting in down sampling the image by a factor of 2. The \( l \) level wavelet decomposition of an image \( I \) results in an approximation image \( X_l \) and three detail images \( H_l, V_l, \) and \( D_l \) in horizontal, vertical, and diagonal directions respectively. Decomposition into \( l \) levels of an original image results in a down sampled image of resolution \( 2^l \) with respect to the image as well as detail images \([4, 5]\).

When an image is defocused, edges in it are smoothed and widened. The amount of high frequency band decreases, and that corresponding to low frequency band increases \([2]\).

In order to denote the relationship between wavelet coefficients and defocused radius \( R \), we define five variables named \( p_1, p_2, p_3, p_4, \) and \( p_5 \) as:

\[
\begin{align*}
\begin{cases}
    p_1 &= |H_1|/|H_2| \\
    p_2 &= |H_2|/|X_2| \\
    p_3 &= |H_1|/\text{num}_H, \\
    p_4 &= |H_2|/\text{num}_H \\
    p_5 &= |X_2|/\text{num}_X,
\end{cases}
\end{align*}
\]

(3)

where \(|H_1|\) and \(|X_1|\) represents the summation of all coefficients’ absolute value in \( H_l \) and \( X_l \) respectively, \( \text{num}_H \) is total number of coefficients in \( H_l \), \( \text{num}_X \) is total number of coefficients in \( X_l \).

An original image is blurred artificially by a uniform defocus PSF with \( R \) whose value ranging from 1 to 10. The relationship between \( p_1, p_2, p_3 \) and \( R \) are shown in Fig. 2 (a), where the curves are normalized in \([0, 1]\) interval. When \( R \) increases, \( p_2 \) and \( p_3 \) decrease monotonously and \( p_1 \) decreases first and increases after reaching a bottom point. This shows that wavelet transform has characters of a band-pass filter.

In order to estimate defocus parameter \( R \), only known the relationship is not enough. As shown in Fig. 2 (b), every image has the monotonous curve between \( p_2 \) and \( R \), but they are not superposition. For a degraded unknown PSF image, \( R \) can not be calculated because the curve of the given image is not known. For example, if \( p_2 \) of image “rice” has been calculated, and then we estimate \( R \) according curve of “mri” in Fig. 2, wrong results are obtained obviously. To solve this problem, one of the methods is to choose neural networks. Computational artificial neural networks are known to have the capability for performing complex mappings between input and output data \([6, 7]\). Here we propose a neural network approach to estimate \( R \). The variables \( p_1 \sim 5 \) are chosen to train the neural network.

![Figure 2](image_url)

(a)

(b)

Figure 2: (a) The relationship between \( p_1 \sim 3 \) and \( R \). (b) Curve \( p_2 \) of different images

2.2 Training a RBF neural network

RBF neural network is a most commonly-used feed-forward network. It usually has one hidden layer, and the basis function is radial symmetry. The output of the network looks like \([8]\):

\[
y_k(\chi) = \sum_{j=1}^{\infty} w_{kj}\varphi_j(\chi) + w_{k0} \Leftrightarrow y(\chi) = W\varphi(\chi).
\]

(4)
where $\chi$ is a input vector, $w_{k0}$ is a set of bias constants, $\phi_i(||\chi - \mu_j||) \equiv 1$, $\alpha$ is the number of RBF hidden neurons and $W$ holds the weights and bias. In the experiments, the radial basis functions are chosen as Gaussian type:

$$\phi_i(||\chi - \mu_j||) = \exp[-\frac{1}{2\gamma_i^2}||\chi - \mu_j||^2],$$

(5)

where $\mu_j$ is the center and $\gamma_i$ is the standard deviation of the Gaussian function, respectively.

Sixteen original images are chosen to train the RBF network. The images are defocused artificially with $R$ whose values ranging from 2 to 7. So the total number of training samples is $96$. Then feature vectors are constructed using variables $p_{1-5}$ of each image:

$$\chi = (p_1, p_2, p_3, p_4, p_5).$$

(6)

For the network output vector, we use one-of-$k$ encoding method, that is, for $R = 2$, $t = (0, 0, 0, 0, 1)^T$; for $R = 3$, $t = (0, 0, 0, 1, 0)^T$ and so on.

When training samples $\chi, t \in [1, 96]$, are given, the weights matrix $W$ can be obtained as $W = \Phi^T$, $\Phi$ is pseudo-inverse of $\Phi$, where $\Phi$ is a matrix:

$$\Phi = \begin{bmatrix}
1 & \cdots & \phi(||\chi_1 - \mu_1||) & \cdots & \phi(||\chi_{96} - \mu_1||) \\
\vdots & \ddots & \vdots & \ddots & \vdots \\
\phi(||\chi_1 - \mu_0||) & \cdots & \phi(||\chi_{96} - \mu_0||)
\end{bmatrix}. $$

(7)

After obtaining weights matrix $W$, the defocused parameter $R$ can be calculated using the trained RBF network.

### 3 Kalman filter and its application

#### 3.1 Kalman filter theory

The Kalman filter was first created by R.E. Kalman in 1960 for problems in satellite orbit mechanics [9]. In essence, it is a recursive linear parameter estimation filter, or a least-squares fitting technique. It has been used in the area of automatic control, navigation, image restoration and so on.

The Kalman filter addresses the general problem of trying to estimate the state $x \in \mathbb{R}^n$ of a discrete-time controlled process that is governed by the linear stochastic difference equation [10]:

$$x_k = Ax_{k-1} + Bu_k + w_{k-1},$$

(8)

with a measurement $z \in \mathbb{R}^m$ that is:

$$z_k = Cx_k + v_k.$$  

(9)

The $n \times n$ matrix $A$ in the difference Eq. 8 relates the state at the previous time step $k - 1$ to the state at the current step $k$, in the absence of either a driving function or process noise. The $n \times l$ matrix $B$ relates the optional control input $u \in \mathbb{R}^l$ to the state $x$. The $m \times n$ matrix $C$ in the measurement Eq. 9 relates the state to the measurement $z_k$. The random variables $w_k$ and $v_k$ represent the process and measurement noise.

The equations for the Kalman filter are described in the following two groups:

**Time update equations:**

$$\hat{x}_k = A\hat{x}_{k-1} + Bu_k,$$  

(10)

$$P_k = AP_{k-1}A^T + Q.$$  

(11)

**Measurement update equations:**

$$K_k = P_kC^T(CP_kC^T + E)^{-1},$$  

(12)

$$\hat{x}_k = \hat{x}_k + K_k(z_k - C\hat{x}_k);$$  

(13)

$$P_k = (I - K_kC)P_k.$$  

(14)

where $\hat{x}_k$ is the a priori state estimate, namely the estimate without the $k$th measurement and $\hat{x}_k$ is a posteriori state estimate, namely the estimate given the $k$th measurement; $P_k$ and $P_k$ are the a priori and a posteriori estimate error covariance, respectively; $Q$ is the process noise covariance matrix; $E$ is the measurement noise covariance matrix and $K$ is the filter gain matrix.

The time update equations are responsible for projecting forward the current state and error covariance estimates to obtain the a priori estimates for the next time step. The measurement update equations are responsible for the feedback. The time update equations can also be thought of as predictor equations, while the measurement update equations can be thought of as corrector equations.

After each time and measurement update pair, the process is repeated with the previous a posteriori estimates used to project or predict the new a priori estimates. This recursive nature is one of the very appealing features of the Kalman filter: it makes practical implementations much more feasible than an implementation of a Wiener filter [11] which is designed to operate on all of the data directly for each estimate.

#### 3.2 Application to image restoration

It is assumed that the original image can be presented by a zero-mean homogeneous $N \times N$ discrete random field. The image can then be modelled by the semicausal type of model [12]:

$$x(m, n) = \sum_{s, t \in \Theta} a(s, t)s(m - s, n - t) + u(m, n),$$  

(15)

where

$$\Theta = \{s, t: 0 \leq s \leq s_2, -t_1 \leq t \leq t_2 \land (s, t) \neq (0, 0)\}.$$  

(16)

$x(m, n)$ is the intensity value of the pixel $(m, n)$ in the original image, $u(m, n)$ is the noise input term, and $\Theta$ is the random field of the chosen model. All points on a line with index $m$ may be combined into a vector $x(m) = [x(m, 1), \ldots, x(m, N)]^T$. By rewriting Eq. 15 and 16, we obtain a matrix-vector equation:

$$A_0 x(m) = -\sum_{s=1}^{s_2} A_s x(m - s) + u(m).$$  

(17)
Similarly, we could also rewrite the discrete two-dimensional convolution Eq. 1 of the image model into the following matrix-vector form:

\[ y(m) = \sum_{k=-k_1}^{k_2} C_k x(m-k) + n(m), \quad (18) \]

where, \( x(m) \) is the image vector \([x(m,1), \ldots, x(m,N)]^T\), and \( y(m) \) is the observation vector \([y(m,1), \ldots, y(m,N)]^T\), \( m = 1, \ldots, N \), and \( A \) and \( C \) are \( N \times N \) matrices of band-Toeplitz structure.

Firstly the band-Toeplitz matrices are approximated by circulant matrices by inserting some elements to the upper-right and lower-left. Then the following equations are obtained:

\[ A_k^c x(m) = -\sum_{s=1}^{s_2} A_k^c x(m-s) + u(m), \quad (19) \]
\[ y(m) = \sum_{k=-k_1}^{k_2} C_k^c x(m-k) + n(m), \quad (20) \]

where matrices \( A_k^c \) and \( C_k^c \) are the circulant approximation of matrices \( A \) and \( C \). Then they are diagonalized by means of the discrete Fourier transform (DFT), and Eq. 19 and 20 are rewritten as:

\[ \Lambda_{A_k} \bar{x}(m) = -\sum_{s=1}^{s_2} \Lambda_{A_k} \bar{x}(m-s) + \bar{u}(m), \quad (21) \]
\[ \bar{y}(m) = \sum_{k=-k_1}^{k_2} \Lambda_{C_k} \bar{x}(m-k) + \bar{n}(m), \quad (22) \]

where \( \Lambda_{A_k} \) and \( \Lambda_{C_k} \) are the matrices of eigenvalues of \( A_k^c \) and \( C_k^c \); \( \bar{x}(m), \bar{y}(m), \bar{u}(m), \bar{n}(m) \) are the DFTs of the vector \( x(m) \), \( y(m) \), \( u(m) \), \( n(m) \), respectively.

Based on the two equations above, a set of low-order Kalman filters suitable for parallel processing of the data in the transform domain can be derived. In this way, the computational requirement is reduced from an order of \( O(N^4) \) to an order of \( O(N^2 \log_2 N) \).

4 Restoration results

The experiments are carried out by using the MATLAB image processing toolbox. The original image "aerial sight" shown in Fig. 3 (a) was blurred by a uniform defocus blur with \( R = 5 \). Fig. 3 (b) and (c) show the degraded images without and with additive noise, respectively.

Wavelet transform is performed to the image in Fig. 3 (c) and the threshold technique proposed by Donoho [13] is adopted to identify and zero out the wavelet coefficients which are likely to arise form noise. Then variables \( p_{1-5} \) of the de-noised image are calculated and used as input of the neural network trained in Section 2.2. The output of the network is given in Eq. 23. According to the encoding strategy of output vector \( \mathbf{t} \), it is obviously that \( R \) should be equal to 5. Finally Kalman filter is employed for restoring Fig. 3 (c) using the estimated PSF. Restoration result is shown in Fig. 3 (d).

\[ \mathbf{t} = (0.041, -0.074, 0.936, 0.318, 0.246, -0.087). \quad (23) \]

As a comparison, we estimate the blur parameter with Fourier method. For the defocused image without noise shown in Fig. 3 (b), firstly two dimensional DFT is adopted to obtain its Fourier spectrum. The defocus parameter can be estimated through calculating a ratio \( f \) of power of high frequencies portion to that of low frequencies portion [2]. In order to evaluate the capability of noise immunity of the Fourier method, ratio \( f \) of the image with noise shown in Fig. 3 (c) is also calculated. As shown in Fig. 4 (a), the noise effect on \( p_2 \) is insignificant. In Fig. 4 (b), the noise influences the value of ratio \( f \) so heavily that it is impossible to estimate the defocused parameter. This result shows that the proposed technique can estimate the defocused parameter credibly and is more robust to noise than the Fourier method.

5 Conclusions

In this paper, we present a new defocused image restoration technique. Defocused parameter was estimated by a RBF neural network trained in wavelet domain. Unlike the other parameter estimating method such as Fourier method,
the main advantage of the proposed technique is that it is robust to noise because wavelet transform has an excellent de-noising ability.

Due to the limitation of training samples, a drawback of the trained network is that the generalization ability is not very satisfying. In the further work, we will try to find solutions to this problem. For example, the possible solutions may be finding more stable representation of defocused parameter in wavelet domain or constructing a training set that covers a wide frequency band. Despite having the drawback, this technique is very suitable for the specific domain such as face images, astronomical images and so on, because the frequency characters of the images in specific domain are similar. In experiments we have tested it using face images, the results are quite satisfied.

Acknowledgement
This work was fully supported by a grant from the National Natural Science Foundation of China (Project No. 60275002). The authors would like to thank the anonymous reviewers for their constructive comments on this paper.

References