Application of HS-LMBP Hybrid Neural Network Algorithm in Image Denoising

Haijun Wang, Menke Neimule, Jin Tao
Ordos Institute Of Technology, Inner Mongolia Ordos, China

Abstract. In order to overcome the problems such as poor global search ability, slow convergence rate, and easy to fall into local minimum values in the image denoising process of traditional BP neural networks, the HS-LMBP hybrid neural network image denoising algorithm is proposed which combines the harmony search algorithm and the LMBP algorithm. The HS-LMBP hybrid neural network algorithm combines the high speed of the LMBP algorithm and the global nature of the HS algorithm, which can be a good improvement to the existing problems of the BP algorithm model. Compared with the Wiener filtering, BP, LMBP and PSO-LMBP model image denoising effects, the denoising model using HS-LMBP neural network algorithm has a better denoising effect.

1 Introduction

With the rapid development of modern information technology, images, as an important information carrier for people to understand the objective world, are exerting more and more important influence on people's lives. Image denoising is one of the classical problems in image processing research. The research of denoising methods is attracting more and more attention of scholars. Most of the non-blind image denoising methods of conventional image are inversely resolved by point spread function, such as median filter, Wiener filter, mean filter, etc. The denoising effect of these methods is good, but the point spread function is difficult to solve, so this type of method the practical application range is limited. The opposite is the image blind denoising method, which can estimate the effective point spread function without any prior information and then solve the problem. This kind of blind denoising method is widely used at present, but there is also a large amount of calculation, local convergence, and the solution is not only problem[1]. Neural networks have the advantages of good nonlinear mapping, parallel computing, and self-adaptive capabilities. In recent years, they have been widely applied to research areas such as image feature extraction, image denoising, image segmentation, and image recognition.

In the study of neural network image denoising algorithms, Back Propagation (BP) algorithm is currently the most commonly used neural network algorithm. However, because the BP algorithm is based on gradient descent principle, there is a weak global search capability and slow convergence speed, easy to fall into local minimums and other issues. In order to overcome these problems, the improved BP algorithm (Levenberg Marquardt, LM) algorithm is often used to build model. This algorithm combines the local convergence of the Gauss-Newton method and the global convergence of the gradient descent method, the efficiency of the LMBP model for image denoising is significantly better than the gradient descent BP algorithm. However, because it can't fundamentally solve the problem of local minima easily caused by the randomness of the initial weight and threshold of the BP algorithm, the model optimization effect is limited. Harmony search (HS) algorithm is a novel intelligent optimization algorithm proposed by Korean scholar Geem Z W et al. in 2001[2], the existing research results show that this algorithm can well solve the problem that the BP algorithm is easy to fall into the local minimum value caused by improper choice of initial and weight threshold [3]. Therefore, a HS-LMBP mixed-image denoising algorithm based on the harmony search algorithm and LMBP algorithm is proposed, experimental results show that compared with Wiener filtering, BP, LMBP, PSO-LMBP algorithm model image denoising effect, using HS-LMBP algorithm modeling image denoising can achieve better results.

2 HS-LMBP algorithm principle

2.1 LMBP algorithm principle

The LMBP algorithm is different from the basic BP algorithm using the local sample information to modify the weight value. The LMBP algorithm uses all the sample information to modify the weight and threshold, and it can adjust weight and threshold of the model adaptively between the Gauss-Newton method and the gradient descent method, while maintaining a fast
descending speed. Each iteration is no longer along a single negative gradient direction, so the LMBP algorithm is more efficient than the basic BP algorithm.

The LMBP algorithm steps are briefly described as follows [4]:

Step 1. Set \( t = (t_1, t_2, \cdots, t_n) \) be the expected output vector, and \( y = (y_1, y_2, \cdots, y_n) \) is the actual output vector, initialize each training parameter, including error target \( \varepsilon \), scale factor \( \mu \), and so on.

Step 2. Set a vector with \( x^{(i)} \) as the weight and threshold for the k iteration, calculation of the error function value \( E(x^{(i)}) \) shown in formula (1), if \( E(x^{(i)}) < \varepsilon \) or reach the maximum number of times, end; else calculate the new weight and threshold \( x^{(i+1)} \).

\[
E(x^{(i)}) = \frac{1}{2} \sum_{i=1}^{n} e_i^2 (2) = \frac{1}{2} \sum_{i=1}^{n} (t_i - y_i)^2
\]

Step 3. According to the formula (2), the \( x^{(i+1)} \) is calculated, and the \( \nabla^2 E(x^{(i)}) \) in the formula is the Hessian matrix of the error function \( E(x^{(i)}) \), and the \( \nabla E(x^{(i)}) \) is the gradient. Calculation of Jacobian matrix \( J(x) \) by formula (5), among \( S(x^{(i)}) = \sum_{i=1}^{n} e_i(x^{(i)}) \nabla e_i(x^{(i)}) \).

At the point near the extreme point, \( S(x^{(i)}) \approx 0 \), the \( x^{(i+1)} \) is calculated by formula (6), the ratio coefficient and the I is a unit matrix.

\[
x^{(i+1)} = x^{(i)} - [\nabla^2 E(x^{(i)})]^{-1} \nabla E(x^{(i)})
\]

\[
J(x^{(i)}) = \begin{bmatrix}
\frac{\partial e_1(x^{(i)})}{\partial x_1} & \ldots & \frac{\partial e_1(x^{(i)})}{\partial x_n} \\
\frac{\partial e_2(x^{(i)})}{\partial x_1} & \ldots & \frac{\partial e_2(x^{(i)})}{\partial x_n} \\
\vdots & \ddots & \vdots \\
\frac{\partial e_n(x^{(i)})}{\partial x_1} & \ldots & \frac{\partial e_n(x^{(i)})}{\partial x_n}
\end{bmatrix}
\]

\[
x^{(i+1)} = x^{(i)} - [J^T(x^{(i)})J(x^{(i)}) + \mu I]^{-1} J(x^{(i)}) e(x^{(i)})
\]

Step 4. If \( E(x^{(i+1)}) < E(x^{(i)}) \), then go to Step 2, otherwise the weight and threshold is not updated to Step 3.

2.2 HS algorithm principle

The idea of the harmony search algorithm comes from the band’s creative process. Figure 1 is a comparison chart of music creation and harmony optimization process, in the picture, a band of 7 people each plays a musical instrument. Their performances correspond to a group of harmony \( x = [x_1, x_2, x_3, x_4, x_5, x_6, x_7] \). They will continue to cooperate and rehearse to get the best harmonization effect, the whole process uses an \( f(x) \) function to measure the harmonic effect, this \( f(x) \) is equivalent to the commander and weighs each set of harmonies played by them. If they do not meet the requirements, they continue playing until they get a satisfactory set of harmonies, this is the basic principle of the harmony algorithm [5].

![Figure 1](https://doi.org/10.1051/matecconf/201817303065)

**Fig.1.** Music creation and harmony optimization process

According to the characteristics of the harmony search algorithm, the specific steps are described as follows [6]:

Step 1. Harmonic memory initialization—Randomly generate \( HMS \) initial harmony values and store it in the harmony memory bank (HM), as follows:

\[
HM = \begin{bmatrix}
\begin{array}{cccc}
x_1 & x_2 & \ldots & x_n \\
x_2 & x_2 & \ldots & x_n \\
\vdots & \vdots & \ddots & \vdots \\
x_1^{HMS} & x_2^{HMS} & \ldots & x_n^{HMS}
\end{array}
\end{bmatrix}
\]

Among them, \( x_i' \) is the i-th component of the j-th solution vector; \( n \) is the dimension of the solution, ie the number of parameters that need to be optimized, and \( f(x) \) is the objective function.

Step 2. New harmony generated randomly—In the harmony optimization process, the generation of new harmony needs to follow three rules: ① Select from the original harmonic memory bank; ② Random creation of new harmonies (that is, a random generation of new solutions); ③ Fine-tuning the new solution resulting from compliance with rules ① and ②. The specific process is as follows:

For New Harmony \( x^{new} = (x_1^{new}, x_2^{new}, \cdots, x_n^{new}) \), each component \( x_i^{new} \) is selected from the original harmony memory bank HM and is generated according to equation (8):

\[
x_i^{new} = \begin{cases} x_i^{new} \in (x_1', x_2', \cdots, x_n^{HMS}), & \text{if } r1 < HMCR \\
x_i, & \text{otherwise}
\end{cases}
\]

Where \( r1 \) is a random number in the (0, 1) interval, HMCR is the probability value of the harmony memory bank; \( x_i \) is the value range of the i-th solution. If the new harmonic generated is from the original harmonic memory bank, it will be fine-tuned per equation (9).

\[
x_i^{new} = x_i(k + m), m \in [-1, 1], & \text{if } r3 < PAR(\text{Discrete space})
\]

Among them, \( bw \) is the fine tuning step, PAR is the fine tuning probability, and \( r2 \) and \( r3 \) are the random numbers of (0, 1) interval, \( k = 1, 2, \cdots, > HMS \).

Step 3. Original memory update—Using the objective function \( f(x) \) to evaluate the new solution generated in step (2), if the new harmony \( f(x^{new}) \) is better than the
worst harmonized \( f(x_{worst}) \) of the original memory bank, then \( x_{worst} = x^{n \text{worst}} \); otherwise the original memory bank does not change.

Step4: Iterative termination judgment----If the algorithm iterates to the maximum algebra or satisfies the termination condition, the algorithm ends and the optimal harmonic is output; otherwise, Steps 2 and Step 3 are repeated until the algorithm ends.

2.3 HS-LMBP algorithm step design

The mathematical problem of the optimization of the HS-LMBP neural network is described as follows:

\[
\min E(w,v,\theta, r) = \frac{1}{N} \sum_{i=1}^{N} (y_i - y'_i)^2
\]

\[s.t. \ w \in \mathbb{R}^{m-p-n}, \ v \in \mathbb{R}^{p-n}, \ \theta \in \mathbb{R}^p, \ r \in \mathbb{R}^n\] \hspace{1cm} (10)

Where \( w, v, \theta, r \) is the weight threshold for the desired structure of the \( m-p-n \) neural network, \( y_i \) is the \( k \)-th sample actual value, and \( y'_i \) is the \( k \)-th sample model output value. The harmony search algorithm is used to solve the above quadratic nonlinear optimization problem, that is, the optimal parameter of weight and threshold combination of the model is obtained.

The specific implementation steps are as follows:

Step1. Parameter initialization----Set the harmony memory size \( HMS \), memory bank probability \( HMCR \), tuning probability \( PAR \), fine-tuning step \( bw \), and maximum iterations \( T_{\text{max}} \). The parameter of weight and threshold \( w, v, \theta, r \) of the BP neural network as a whole, encoded according to the HS algorithm Step1, determining the search space dimension \( d \) of the harmony search algorithm, each harmony voice contains the full weight and threshold of the BP neural network. The code length is calculated as follows:

\[ d = mp + pn + p + n \] \hspace{1cm} (11)

Step2. Fitness function definition----Since the harmony search algorithm uses the minimum value of the objective function as its fitness function in the optimization process, the fitness function is defined as:

\[
F(w,v,\theta, r) = \frac{1}{N} \sum_{i=1}^{N} (y_i - y'_i)^2
\]

Step3. Adaptive parameter settings [7]----In this study, equations (13) and (14) are applied to the HS algorithm

\[
PAR(t) = PAR_{\text{min}} + (PAR_{\text{max}} - PAR_{\text{min}}) \times t / T_{\text{max}} \] \hspace{1cm} (13)

\[
bw(t) = bw_{\text{max}} \times \exp(t / T_{\text{max}} \times \log(bw_{\text{max}} / bw_{\text{min}})) \] \hspace{1cm} (14)

Step4. HS optimized weight and threshold----HS optimization BP neural network initial weight and threshold, global extreme iterative optimization based on Step2 - Step4 in the above HS algorithm.

Step5. The algorithm terminates the judgment by iteration----It determines whether the fitness satisfies the termination condition. If it satisfies, it outputs the best harmony, that is, the optimal solution of the problem; otherwise, it performs Step4 (HS algorithm optimization operation).

Step6. Optimal harmony decoding----The BP neural network initial weight threshold \( w, v, \theta, r \) are obtained by decoding the HS-optimized solution values corresponding to the optimal harmony, and they are given to the neural network image compression model based on the LMBP algorithm for secondary training, perform secondary training until the error function \( E(x^{(t)}) < \text{goal} \) calculated according to (1) or the maximum number of operations \( epochs \).

3 sample structure and definition of hidden layer nodes

3.1 sample structure

For the extraction of image feature information, a nine-grid sliding window structure with blocking idea is used to traverse. The gray value of one point and its surrounding eight points in the degraded image is taken as an input sample, and the whole blur is used in the 3x3 sliding window. After the image is traversed, all input data of the BP model is obtained. For the output layer, because the output of the model is the noise-free pixel corresponding to the center of the sliding window, the number of output layer nodes is 1, this constructs a model input/output sample pair of <9 input, 1 output>. Figure 2 shows an unnoise Lenna image with a size of 150x150, Figure 3 shows a blurred image with Gaussian noise added, and Figure 4 shows the sample data that can be modeled after the image feature is extracted.

3.2 Definition of hidden layer nodes

For the determination of hidden layer nodes, this paper adopts the golden section method proposed by Professor Xie Kewen to determine. The specific steps are as follows:

Step1. Determine the initial search interval endpoints \( a \) and \( b \) as shown in equation (15).

\[ a = (n + m) / 2 \leq r \leq (n + m) + 10 = b \] \hspace{1cm} (15)

Step2. In the interval [a, b], the errors of 0.618 points and 0.382 points are calculated continuously, and the error is compared and the interval is preserved. After comparing the advantages and disadvantages of multiple rounds. Get the error function set corresponding to the hidden layer node, as shown in equation (16).
4.1 parameter settings

Using Matlab2012b platform for experimental simulation, related parameter settings:BP learning rate \(l=0.05\), Training times \(epochs=50\), Training goal \(goal=0.0001\), Training algorithm is used trainlm; HS algorithm memory bank size \(HMS=30\), Memory probability \(HMCR=0.95\), fine-tuning probability \(PAR_{\text{max}}=0.1\), \(PAR_{\text{min}}=0.5\), fine-tuning step \(bw_{\text{max}}=d/1000\), \(bw_{\text{min}}=d/10000\), Number of iterations \(T_{\text{max}}=100\).

4.2 Result analysis

Using Gaussian Noise Image with 150×150 Brain and Cameraman as Test Samples, Wiener filtering, BP, LMBP, PSO-LMBP, and HS-LMBP algorithms are used to model denoising experiments respectively. Figure 6-12 shows the results of the relevant denoising experiments on the Brain chart. It can be seen from the figure that the Winner filtering does not remove the noise well and the image quality is not significantly improved compared to the noise-containing graph. The BP algorithm model is not only unable to eliminate the noise but also makes the image details more blurred due to the limited number of operations into the local optimum, while the LMBP, PSO-LMBP and HS-LMBP have achieved a good denoising effect. Similar results were obtained from the related experiments of Cameraman diagrams in Figure 13-19. In order to further quantitatively evaluate several algorithms, structural similarity (SSIM) functions and peak signal-to-noise ratio (PSNR) functions are used as evaluation criteria. Table 1 shows the evaluation results based on these two evaluation functions. It can be seen from the table that the HS-LMBP denoising algorithm proposed in this paper is the best in both SSIM and PSNR.

5 Conclusion

Aiming at image denoising, an HS-LMBP image denoising algorithm based on LMBP algorithm and harmony search algorithm is proposed. The algorithm firstly uses the HS algorithm to obtain the approximate optimal solution of the weight threshold of the BP algorithm, and then uses the LMBP to perform the second optimization to determine the final algorithm model parameters. The algorithm thus constructed can dynamically control the mutual conversion process between the global search and the local search, and has better convergence. Experimental results show that the image denoising model based on HS-LMBP algorithm is feasible and the denoising effect is better than Wiener filtering, BP, LMBP, and PSO-BP algorithm models. Therefore, it has in-depth research value. At the same time, the model-related parameters are modified, and the proposed HS-BP algorithm can also be applied to other image processing research fields. The algorithm also has the value of promotion and research.
Acknowledgements

This work has been supported by China National Natural Science Foundation Project(61741509), Scientific research project of Inner Mongolia higher education institutions(NJZZ07007, NJZY16382).

Reference