Knowledge-Based Systems xxx (2014) xxx–xxx

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Particle Swarm Optimization based dictionary learning for remote sensing big data

Lizhe Wang [a,*], Hao Geng [b], Peng Liu [a], Ke Lu [c], Joanna Kolodziej [d], Rajiv Ranjan [e], Albert Y. Zomaya [f]

[a] Institute of Remote Sensing and Digital Earth, Chinese Academy of Sciences, PR China
[b] Department of Electrical Engineering and Information Sciences, University of Science and Technology of China, PR China
[c] University of the Chinese Academy of Sciences, Chinese Academy of Sciences, PR China
[d] Institute of Computer Science, Cracow University of Technology, Poland
[e] The Commonwealth Scientific and Industrial Research Organisation, Australia
[f] School of Information Technologies, The University of Sydney, Australia

ARTICLE INFO

Article history:
Received 7 April 2014
Received in revised form 3 October 2014
Accepted 6 October 2014
Available online xxxx

Keywords:
Online dictionary learning
Particle Swarm Optimization
Sparse representation
Big data
Machine learning

ABSTRACT

Dictionary learning, which is based on sparse coding, has been frequently applied to many tasks related to remote sensing processes. Recently, many new non-analytic dictionary-learning algorithms have been proposed. Some are based on online learning. In online learning, data can be sequentially incorporated into the computation process. Therefore, these algorithms can train dictionaries using large-scale remote sensing images. However, their accuracy is decreased for two reasons. On one hand, it is a strategy of updating all atoms at once; on the other, the direction of optimization, such as the gradient, is not well estimated because of the complexity of the data and the model. In this paper, we propose a method of improved online dictionary learning based on Particle Swarm Optimization (PSO). In our iterations, we reasonably selected special atoms within the dictionary and then introduced the PSO into the atom-updating stage of the dictionary-learning model. Furthermore, to guide the direction of the optimization, the prior reference data were introduced into the PSO model. As a result, the movement dimension of the particles is reasonably limited and the accuracy and effectiveness of the dictionary are promoted, but without heavy computational burdens. Experiments confirm that our proposed algorithm improves the performance of the algorithm for large-scale remote sensing images, and our method also has a better effect on noise suppression.

1. Introduction

Recently, sparse representation has become a very popular topic in the area of remote sensing image processing. In many tasks related to remote sensing images, such as image segmentation, fusion, classification, reconstruction, and change detection, sparse representation is frequently employed to improve the performance of the algorithms. Modeling data as sparse combinations of atoms, which are the elements of a dictionary, can manifest the important intrinsic characteristics of remote sensing images.

There is a long research history on how to sparsely represent a signal or data by a set of bases. We also call this set of bases a dictionary. There are two different classes of dictionary: analytic and non-analytic. Many of the earlier studies on sparse representation focused on analytic dictionaries. Different bases, such as Fourier transformations, wavelets [1], curvelet [2], bandelet [3], directionlet [4], and grouplet [5], were proposed in different periods. The development of analytic dictionaries went through several stages, such as multi-resolution, localization, anisotropy, and adaptation. Another large class of dictionary is non-analytical. Unlike decompositions based on a predefined analytic base (such as a wavelet) and its variants, we can also learn a hyper complete dictionary without analytic form, which has neither fixed forms of atoms nor requires base vectors to be orthogonal. The basic assumption behind the learning approach is that the structure of complex incoherent characters can be more accurately extracted directly from the data than by using a mathematical description.

A non-analytic dictionary learning problem apparently can be modeled as a constraint-optimization problem. The optimization of both the dictionary and coefficients is non-convex, but alternative optimization is convex. Therefore, many algorithms consist of two stages: atom updating and sparse coding. The main differences between most methods, such as the method of optimal directions

* Corresponding author.
E-mail address: Lizhe.Wang@gmail.com (L. Wang).

http://dx.doi.org/10.1016/j.knosys.2014.10.004
0950-7051/

(MOD) [6], generalized PCA (GPCA) [7], and K-SVD [8], are their atom-updating stages. Obviously, we hope that dictionary learning is as efficient as possible. The direct optimization method (DOM) [9] denoted the algorithm as a one-step block-coordinate proximal gradient descent. It is more efficient than alternating optimization algorithms. On the other hand, the effectiveness of sparse representation is also important. The Fenchel duality method [10] solved this problem in a dual space and promoted the effectiveness of the dictionary. Another idea is to use a first-order series expansion instead of the dictionary-coefficient matrix product [11]. Doing so improves performance while adding only a small additional computational load. Apart from constraint optimization, we can also model the dictionary problem as a stochastic process. Non-parametric Bayesian dictionary learning (NBDL) [12] employs a truncated beta-Bernoulli process to infer an appropriate dictionary, and obtains significant improvements in image recovery [12]. Furthermore, multi-scale dictionary learning can also be presented as a fully Bayesian model [13].

Non-analytic dictionary learning is very efficient in data representation; however, it also introduces many new problems. First, the relationship between the over-complete atoms attracts much research attention. The atoms in the dictionary could be incoherent [14], multi-model [15,16], multi-dictionary [17,18], multi-scale [19,20], or hierarchical [21,22]. Second, the dictionary problem is also a supervised versus unsupervised issue. In the early research, most of the dictionary learning methods were unsupervised. Recently, with its wide applications to many different areas, using discriminative information [23–26] in the dictionary learning process has become popular. Supervised dictionary learning [27] makes the atoms more sophisticated and more flexible. Furthermore, the conception of task-driven dictionary learning was proposed [28].

For large groups of data, it is very hard to take all the data into the computation model at once. Therefore, in addition to the batch-based methods mentioned above, a group of online learning methods, such as recursive least squares dictionary learning (RLS) [29], online dictionary learning (ODL) [30], and the non-parametric Bayesian method (NBDL) [12], were developed in recent years. However, these online learning tools also led to some new problems and concerns, such as how to introduce the data into the training process in a smooth and orderly manner, how to perform dimension reduction [31], and how to optimize the structure of the dictionarys atoms. More importantly, experiments show that the accuracy of sparse representation of the dictionary produced by online learning is decreased because of the complexity of the large data sets of remote sensing images. The reason for this is that the strategy of updating atoms in ODL, RLS, or NBDL is unreasonable when handling large data. When taking both dictionaries, D, and coefficients, α, as variables, it is difficult to optimize them at the same time because of the non-convex character of the object function. Alternative optimization of atoms and coefficients decreases the accuracy of dictionary learning, especially when the data set is very large. It is easy to stop at a local extreme value under the influence of the continuous computation manner, noise, and complexity of the data. However, for large remote sensing data sets, there are often many other priors than the sparsity that we can utilize in the dictionary-learning process.

First, for a certain area, the remote sensing images for a terrestrial object at different times always show some similarities because the changing of land covers is usually slow. Second, for the same scene, the remote sensing images from different sensors often share similar textures to some extent. Therefore, when the location is given, we can usually use the history or multi-source data to guide the direction of the optimization in the atom-updating stage of dictionary learning. For example, in the atom-updating stage of the ODL method, the gradient direction can be easily guided by reference data or an existing reference dictionary.

In this paper, to effectively and sparsely represent the large remote sensing image set, we use reference data as priors and introduce PSO [32] into the atom-updating stage of the ODL algorithm. In the iteration, special atoms in the current dictionary are selected as the particles in the PSO model. In order to reduce the dimensions of the particles, every selected atom is represented by the linear combination of a reference image and the remaining atoms. To make the optimization more efficient, in PSO, the flying directions are limited to the few dimensions that are estimated by considering the relationship between the different subspaces of the atoms. Furthermore, for the redundant and cluster characters of the textures of the large remote sensing data set, the features of the reference data constrain and guide the ranges and directions of random particle movement in PSO. As a result, the flying of the particles in PSO is semi-random. This proposed semi-random PSO promotes the accuracy of the atom updating, and does not result in heavy computational burdens because of the guidance of the reference data. In the following sections, we first summarize the ODL algorithm and then propose our method based on the new atom-updating scheme.

2. Online dictionary learning based on gradient descent

The non-analytic sparse representation uses a hyper-complete dictionary matrix $D \in \mathbb{R}^{m\times n}$, which includes n atoms for columns to represent a signal $x \in \mathbb{R}^m$ as a sparse linear combination of these atoms. The representation of sample data x can be written as the approximate $x \approx D\alpha$, which satisfies $||x - D\alpha||_p \leqslant \varepsilon$. Here, the typical norm for sparse representation is l^p-norms, and usually is true in the case of $p = 2$. Dictionary learning is an optimization problem written as:

$$\arg\min_{D,\alpha} \frac{1}{2}\|X - D\alpha\|_2^2 + \lambda||\alpha||_1. \quad (1)$$

For convenience, $X \in \mathbb{R}^{m\times q} (m \ll q)$ is the training data set and $x_i \in \mathbb{R}^m$ is the ith column of training data matrix X. The dictionary is denoted by $D = \{d_1, \cdots, d_j, \cdots, and\ d_n\}$, and d_j stands for the jth column of D. λ is a regularization parameter. α is the coefficient of sparse representation. The Frobenius norm of a matrix X in $\mathbb{R}^{m\times p}$ here can be denoted by $||X||_F \triangleq (\sum_{i=1}^m \sum_{j=1}^q X[i,j]^2)^{1/2}$. The object function to be minimized in Eq. (1) is not jointly convex in α and D, but it becomes convex in one variable, keeping the other fixed. Thus, the ODL algorithm can be divided into two steps that alternately solve the optimization problem in an iterative loop. One is keeping D fixed and finding α, which is called the sparse coding stage. The other is keeping α fixed and finding D, which is called the atom-updating stage.

In the first stage, the ODL uses LARS [33] or orthogonal matching pursuit (OMP) [34] to find α:

$$\alpha_t = \arg\min_{\alpha} \frac{1}{2}||x_t - D_{t-1}\alpha||_2^2 + \lambda||\alpha||_1, \quad (2)$$

where the subscript t means the tth iteration of the ODL procedure.

In the second step, the original objective function is:

$$D = \arg\min_{D\in C} \frac{1}{t}\sum_{i=1}^{t}\frac{1}{2}||x_i - D\alpha_i||_2^2 + \lambda||\alpha_i||_1. \quad (3)$$

As we know, the optimization variable in the object function is the dictionary D; in the meantime, we use the matrix form of X instead of the vector α. The Eq. (3) can thus be rewritten as:

$$D_t = \arg\min_{D\in C} \frac{1}{2}||X - D\alpha||_F^2. \quad (4)$$

To some extent, the Frobenius norm of a matrix can be replaced by the trace of another matrix, and we can obtain the following equation:

$$D_t = \arg\min_{D\in C} \frac{1}{2} Tr[(X - D\alpha)(X - D\alpha)^T]. \quad (5)$$

After using the knowledge of the trace to operate Eq. (5), the object function is simplified as:

$$\begin{aligned} D_t &= \arg\min_{D\in C} \frac{1}{2t} Tr[(D\alpha)(D\alpha)^T - 2X(D\alpha)^T] \\ &= \arg\min_{D\in C} \frac{1}{2t} Tr[D^T D\alpha\alpha^T - 2D^T X\alpha^T]. \end{aligned} \quad (6)$$

Two intermediate variables, A and B, are introduced in ODL. They carry information on the sparse coefficients of the previous samples. The forms of intermediate variables come from the stochastic gradient algorithm [35]. A and B update themselves as:

$$A_t \leftarrow A_{t-1} + \alpha_t\alpha_t^T. \quad (7)$$

$$B_t \leftarrow B_{t-1} + x_t\alpha_t^T. \quad (8)$$

With the definition of A and B, the objective function is ultimately written as:

$$D = \arg\min_{D\in C} \frac{1}{t}(Tr(D^T DA_t) - Tr(D^T B_t)). \quad (9)$$

While updating atoms in ODL based on the stochastic gradient decent, calculating the gradient ∇ of Eq. (9) signifies a partial derivative with respect to matrix D:

$$\begin{aligned} \nabla &= -\frac{\partial\left(\frac{1}{2}D^T DA - D^T B\right)}{\partial D} \\ &= -\frac{1}{2}\left(\frac{\partial(D^T D)}{\partial D}A + \frac{\partial(D^T D)^T}{\partial D}A\right) + \frac{\partial(D^T)}{\partial D}B \\ &= B - DA \end{aligned} \quad (10)$$

After getting the gradient, the authors give the updating equation to update the atoms one by one, based on the stochastic gradient descent algorithm:

$$u_j \leftarrow \frac{1}{A_{jj}}(b_j - Da_j) + d_j. \quad (11)$$

Therefore, the atom d_j is:

$$d_j \leftarrow \frac{1}{max(\|u_j\|_2, 1)} u_j. \quad (12)$$

We find that current atoms updating stage for small data set is not enough for large data set. For large data set, the atoms updating stage is more easily apt to plunge into local minimum. Therefore, in this paper, PSO is introduced into the atoms updating stage of the dictionary learning process. When we take atom as the particle in PSO, it is relative easy to implement in discrete computation. Furthermore, using PSO, the atoms updating stage can search the atoms in a larger scope.

To some extent, the PSO algorithm is similar to the stochastic gradient descent algorithm. The PSO, which simulates the aggregation and migration of birds, was first presented by Drs. Kennedy and Eberhart in 1995.

PSO optimizes a problem by having a population of candidate solutions, here dubbed particles, and moving these particles around in the search space according to simple mathematical formulae over the particle's position and velocity. Each particle's movement is influenced by its local best-known position, but is also guided toward the best-known positions in the search space, which are updated as better positions are found by other particles. This is expected to move the swarm toward the best solutions. The PSO algorithm, without complex evolution operations, which works well in terms of calculation speed and accuracy, was developed very quickly in recent years. Compared with the gradient algorithm, PSO may be closer to the infinite exhaustive method. The strategy of constructing a dictionary requires ODL to update all atoms in an iterative loop, but we can optimize only one atom or part of an atom if necessary. Therefore, we propose our method based on PSO. The proposed algorithm focuses on improving the performance of ODL by using PSO [32] in the optimization portion of atoms in a dictionary.

3. Introducing PSO into the atom-updating stage

Our goal is to improve the performance of ODL in terms of the accuracy of reconstructed remote sensing images. Briefly, the algorithm we presented applies the PSO algorithm to optimize the most important atom selected by the scheme we designed in the dictionary-updating step. The selected atom is linearly represented by the other atoms in the dictionary, and the particle in PSO is the coefficient of the linear representation but not the selected atom.

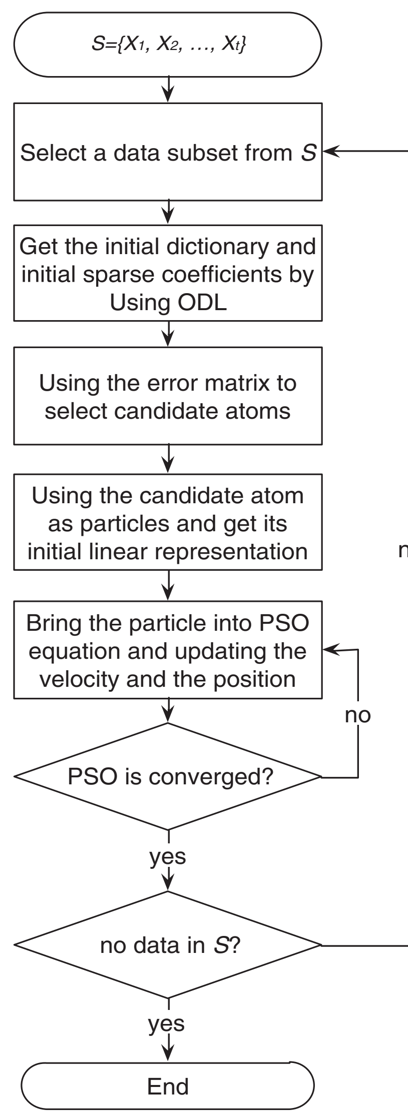


Fig. 1. The flowchart of the proposed method.

Please cite this article in press as: L. Wang et al., Particle Swarm Optimization based dictionary learning for remote sensing big data, Knowl. Based Syst. (2014), http://dx.doi.org/10.1016/j.knosys.2014.10.004

Therefore, the dimension of the particle will not be too high. Every time, a part of data are introduced into the learning process for each iteration. The flowchart of the proposed algorithm is shown in Fig. 1 in this section.

Our algorithm makes four main advances:

(1) The scheme for selecting special atoms.
We make some changes to the norm part of Eq. (1).

$$\|X - D\alpha\|_F^2 = \left\|X - \sum_{j=1}^{K} d_j \alpha_j\right\|_F^2 = \left\|\left(X - \sum_{j\neq k} d_j \alpha_j\right) - d_k \alpha_k\right\|_F^2$$

We define E_k as:

$$E_k = X - \sum_{j\neq k} d_j \alpha_j \qquad (13)$$

Here, α_j is the jth row in the sparse coefficient matrix α. E_k stands for the error of all samples when the kth atom is removed [36]. The special atom d_k here is defined as the atom that makes the biggest contribution to the estimate of a sample X. d_k is defined as:

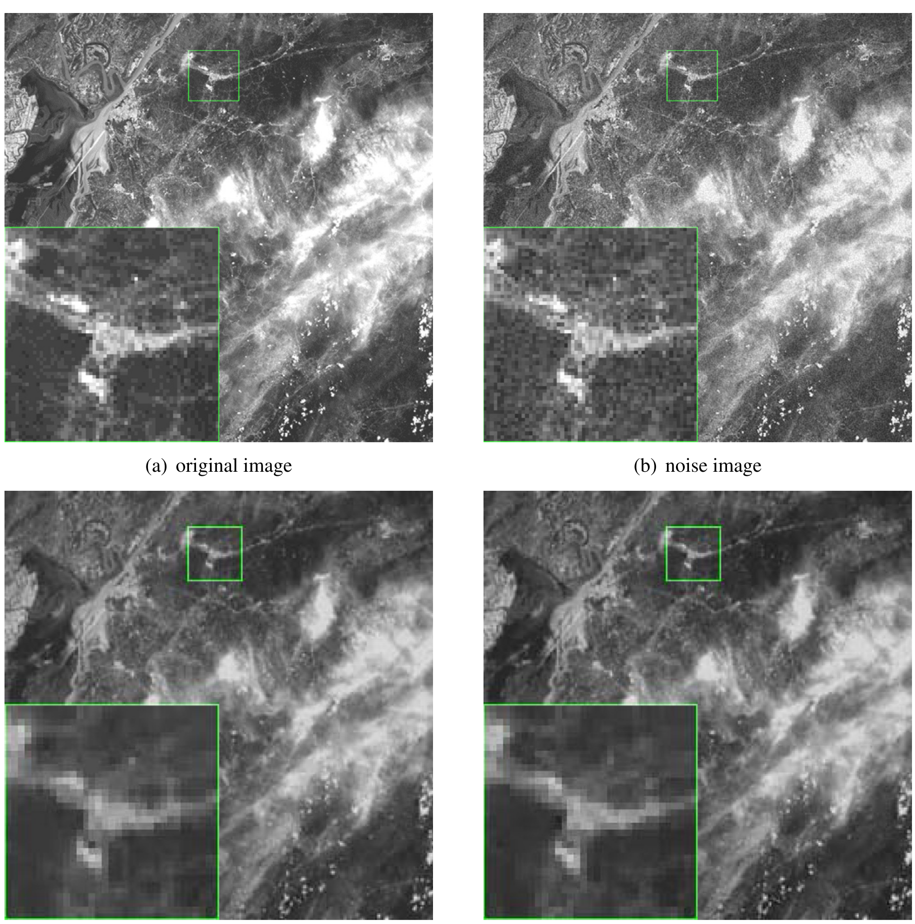
$$d_k = \arg\max_d ||E_k||_F^2.$$

(2) The scheme for selecting particles.
The flying behavior of particles is random to a certain degree. If we directly choose an atom to be the particle, the atomic pattern may not still be organized in terms of texture features and it may appear to be noisy. Because of the theorem of linear algebra, in a hyper complete dictionary matrix, $D \in \mathbb{R}^{m\times q} (m \ll q)$, we can easily find m atoms as a vector group to represent other atoms linearly. We might as well make some derivations. Assume that r_d is the rank of matrix D. The rank r_r of row vector group D satisfies $r_r \leqslant m$, and r_c, which is the rank of column vector group D, satisfies $r_c \leqslant n$. We know the theorem that, in a matrix $D, r_r = r_c = r_d$. It is not difficult to know $r_c \leqslant m$ in the case of $m \ll q$. The number of maximum linearly independent vector groups of columns must be less than or equal to m, so we have a conclusion that we can find not more than m column vectors as maximum linearly independent groups to represent the other columns. It is reasonable for us to use the coefficients, which are the linear representations of the atoms selected to be particles. Here we consider the worst case scenario: that the number of vectors in a maximum independent group is equal to m. In fact, the number is usually less than m. In a sense, it is reducing the dimensions of the particles in a PSO problem as follows:

$$d_k = y_1 v_1 + y_2 v_2 + \cdots + y_m v_m, \qquad (14)$$

where $V = \{v_1, \ldots, v_m\}$ stands for the vector group and y_i is the linear representation coefficient. The set $Y = \{y_1, \ldots, y_m\}$ is the particle in our method.

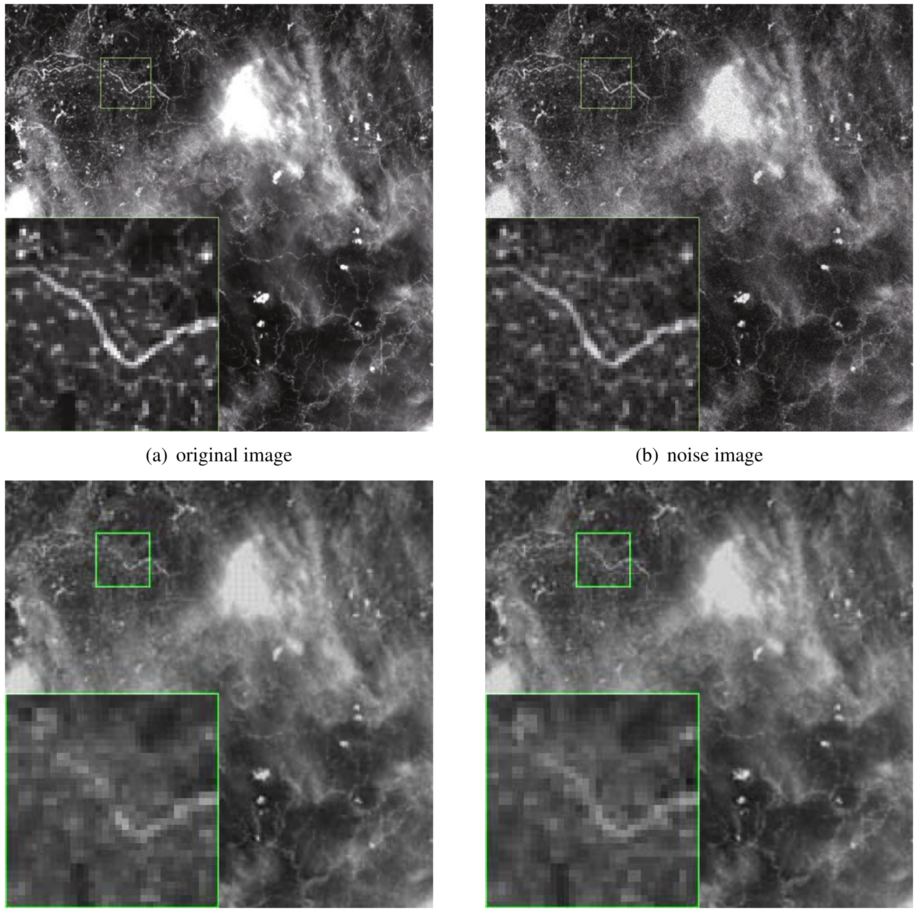
(3) The **matching** Initial velocities scheme.
Because the textures of large-scale remote sensing images are complicated, the values of the coefficients we obtain from the strategy of selecting a particle range from 10^{-4} to



(a) original image (b) noise image (c) ODL (d) PSODL

Fig. 2. Compare the reconstruction results of Landsat-8 images.

Please cite this article in press as: L. Wang et al., Particle Swarm Optimization based dictionary learning for remote sensing big data, Knowl. Based Syst. (2014), http://dx.doi.org/10.1016/j.knosys.2014.10.004



(a) original image (b) noise image

(c) ODL (d) PSODL

Fig. 3. Compare the reconstruction results of HJ-1-A images.

10^4. We thus designed a matching scheme of initial velocities, which can be dynamically adjusted according to the value of the coefficients. If the particles fly randomly in the search space, the time required to find an optimal solution will greatly increase. The scheme aims to guide the flying particles in more reasonable dimensions.

(4) Stability for the whole algorithm.
As we know, in high-dimension situations, PSO is not very effective at achieving optimal solutions. Our experiments show that, without any stability measurements, sometimes PSO is not stable enough to optimize the atoms. On the other hand, we also need a mechanism to carry past information, which includes changes to sparse coefficients due to PSO. Here, after optimizing by PSO, we use the intermediate variables A and B, but make only a small change. The formulas of A and B that were used in our method are positioned as:

$$A_t \leftarrow A_{t-1} + \beta \alpha_t \alpha_t^T \tag{15}$$

and

$$B_t \leftarrow B_{t-1} + \beta x_t \alpha_t^T, \tag{16}$$

where β is an important coefficient whose value often ranges from 0.05 to 0.1. In ODL, it accelerates the convergence speed [30], and in our algorithm, it plays a role in improving the stability of our method.

The selected atom, d_k, and the other atoms are randomly chosen from the initial atomic group. We need some other atoms to be the vector group so that we can represent the atomic group. After the atomic group is represented by the vector group, we then get the linear representation coefficients. These coefficients fly in the search space as a particle swarm at velocities that are dynamically adjusted according to each particles own flying experience and the groups flying experience.

Assume that in a m dimension objective search space (m is also the number of vectors in a maximum linearly independent group) the particle swarm is $Y = (Y_1, Y_2, \ldots, Y_n)$, which is made up by n particles, and Y_i is the coefficient set of linear representation. Y_i, which also stands for the position of the ith particle, can be presented as $(y_{i_1}, y_{i_2}, \ldots, y_{i_m})^T$. Individual extremum $P_i = (p_{i_1}, p_{i_2}, \ldots, p_{i_m})^T$ is the previous best position of the ith particle. When the global extremum $P_g = (p_{g_1}, p_{g_2}, \ldots, p_{g_m})^T$ represents the previous best position of the swarm. Flying velocity can be presented as $V_i = (v_{i_1}, v_{i_2}, \ldots, v_{i_m})^T$. The velocity update is conducted as follows:

$$V_i^{k+1} = \omega V_i^k + c_1 r_1 (P_i^k - Y_i^k) + c_2 r_2 (P_g^k - Y_i^k) \tag{17}$$

where k represents the iteration number and ω is the inertia weight [37]. r_1 and r_2 are random numbers that range from 0 to 1. c_1 and c_2 are called acceleration factors, whose values range from 0 to 4. Here, we let c_1 equal c_2. The equation for updating particle positions is given below:

$$Y_i^{k+1} = Y_i^k + V_i^{k+1} \tag{18}$$

Please cite this article in press as: L. Wang et al., Particle Swarm Optimization based dictionary learning for remote sensing big data, Knowl. Based Syst. (2014), http://dx.doi.org/10.1016/j.knosys.2014.10.004

Algorithm 1. Online dictionary learning based on PSO

Require:
A group of samples $\{X_1, X_2, \ldots, X_t\}$,
1: **for** $n = 1$ to t **do**
2: Get X_n from a set of samples and use ODL to get the dictionary D_n and the sparse coefficient matrix α_n ;
3: Make use of the atom-selecting scheme to choose the special atom d_{k_n}, which is to be optimized by PSO.

$$d_{k_n} = \arg\max_{d} ||E_{k_n}||_F^2 \tag{19}$$

4: Use Eq. (14) to make the initial representation of d_{k_n} and get the linear representation coefficients as the initial particle swarm. Using the initial atomic group as the input variable d_{in}, calculate the initial value of the objective function of PSO. Obtain the initial individual extremum and the initial global extremum.

$$f_n = \left|\left|(X_n - \sum_{j_n \neq k_n} d_{j_n}\alpha_{j_n}) - d_{in}\alpha_{k_n}\right|\right|_F^2 \tag{20}$$

5: Apply Eqs. (17) and (18) to update the velocity and the position of the particle. Multiply the new swarm by the vector group to gain the new atomic group. Calculate the value of Eq. (20) with the new atomic group. and update the individual extremum and the global extremum.
6: Repeat step 5 until convergence of the PSO algorithm occurs. Get the new atom.
7: Construct the dictionary D_n. Then, with the OMP algorithm, get the new α_n. Using (15) and (16), update intermediate variables A and B. The new dictionary is the initial D for the ODL when the next sample, X_{n+1}, arrives.
8: **end for**

After several iterative loops, the best position of the swarm is the new coefficients of the selected atom, and we now have a new dictionary. Based on the new D, we get the new α. Then we update the variables A and B using (15) and (16) shown in Algorithm 1.

4. Experiments and Results

The performance of the proposed algorithm is tested with the real large remote sensing image data set. The remote sensing images are selected from two satellite image sets. They are from Landsat-8 and HJ-1-A. For convenience, partial images at a size of 512×512 are shown in figures. The proposed algorithm (PSODL) is comprehensively compared with the ODL algorithm. In experiments, the performances of two different algorithms are measured qualitatively and quantitatively. For qualitative comparisons, the images reconstructed by two different algorithms are compared visually for accuracy of reconstruction. For quantitative comparisons, the reconstructed performs are measured in terms of the peak signal-to-noise ratio (PSNR).

In the first experiment, the original images all include the additive noise, and we use four remote sensing images, which are named Image$X$ (here, X means a number, such as 1, 2, 3, 4, etc.) to test the performance of two algorithms. The additive noise satisfies the normal distribution, with a mean of 0 and standard deviation of 0.1. A normal distribution for additive noise is the most common pattern for real satellite images. With the same control parameters, we qualitatively and quantitatively measure the reconstruction performance of the two algorithms. Here we set the number of atoms at 100 and the size of each atom is 8×8. Fig. 2 shows the reconstruction of the original Image1 with additive noise by ODL and PSODL. Image1 was selected from Landsat-8. Fig. 2(a) shows the original Image1. Fig. 2(b) is the original Image1 with additive noise. Fig. 2(c) is the reconstruction image by ODL and Fig. 2(d) is the result of PSODL. Fig. 3 shows the result of the remote sensing images from HJ-1-A. We can observe that there are more edges and details that are reconstructed by the proposed PSODL. From Figs. 2 and 3, we can see that the proposed algorithm is better than ODL at maintaining the edge texture

Table 1
The comparison of the performances of different method by different images with additive noise. σ is the standard deviation of the noise.

σ	Method	Image1	Image2	Image3	Image4
0.2	ODL	25.0645	26.7851	25.6123	24.7673
	PSODL	25.1474	26.8611	25.7102	24.9304
0.3	ODL	24.4947	25.8528	25.0626	24.2324
	PSODL	24.6190	25.9428	25.1823	24.4531
0.4	ODL	23.9255	25.3754	24.1570	23.6514
	PSODL	24.0488	25.4853	24.2707	23.9463
0.5	ODL	23.5808	24.6571	21.2242	23.1855
	PSODL	23.6989	24.7729	21.5771	23.4699

Table 2
The comparison of PSNR with sparsity of 5%.

Unit:dB	Image1	Image2	Image3	Image4
ODL	26.5514	30.9478	30.2476	26.3403
PSODL	26.5714	30.9895	30.2667	26.4140

Table 3
The comparison of PSNR with the sparsity of 10%.

Unit:dB	Image1	Image2	Image3	Image4
ODL	24.0673	27.4614	27.8027	23.4125
PSODL	24.0770	27.5017	27.8407	23.4602

Table 4
The comparison of sparsity with the controlled PSNR value of 24 dB.

	Image1 (%)	Image2 (%)	Image3 (%)	Image4 (%)
ODL	2.9	0.99	1.28	2.34
PSODL	2.89	0.99	1.29	2.35

Table 5
The comparison of sparsity with the controlled PSNR value of 20 dB.

	Image1 (%)	Image2 (%)	Image3 (%)	Image4 (%)
ODL	0.34	0.27	0.21	0.49
PSODL	0.34	0.26	0.21	0.51

Table 6
The sensitivity of PNSR with respect to the change of iteration parameter λ on Image4.

λ	5	8	10	12	14	15
PSNR	26.3764	26.3784	26.3848	26.4020	26.4064	26.4140
λ	17	30	40	50	60	
PSNR	26.4254	26.4308	26.4268	26.4242	26.4264	

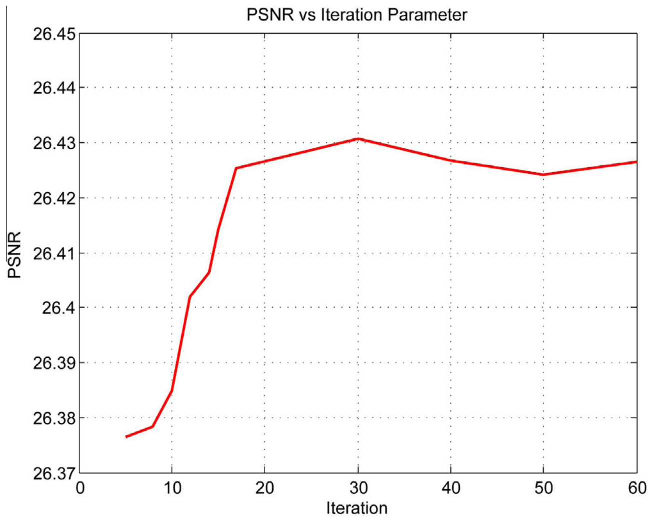


Fig. 4. PSNR vs iteration number.

characteristics of the large-scale remote sensing image, such as coastlines, mountain ridges, and rivers. Table 1 quantitatively shows the reconstruction performances of the two algorithms. We also observe that the PSNR of images reconstructed by PSODL is higher than that of ODL. The experiments in Table 1 confirm that the reconstruction performance with noise of our algorithm from remote sensing images is better than that of ODL. This also means that our algorithm has a better effect on noise suppression or image de-noising.

In the second experiment, we compare the performance of ODL and PSODL from two sides. One is the performances of two algorithms in accurate reconstruction with the same sparsity; the other is the sparsity performance with the same expected values of PSNR. We found that, if the size is too small, the sparsity of the two algorithms is not very high. The reason for this is that the scale of the texture of the remote sensing images is large and complicated. Therefore, in the experiment comparing sparsity, we set the size of the data sample and atom to 16×16. Table 2 provides the comparison of the reconstructed images in terms of PSNR, with the same sparsity control. The sparsity value is 5%. Table 3 is also a PSNR comparison, but the sparsity value of is 10%. From the results summarized by the two tables, it is clear that the PSNR of PSODL is higher than that of ODL in most cases. Table 4 is a comparison of the sparsity of the representation coefficients in matrix α with an expected PSNR of 24 dB. Table 5 is also a comparison of sparsity, with an expected PSNR value of 20 dB. We can see that the PSODL sparsity is almost the same as that of ODL in most cases.

In the third experiment, the reconstruction performance with respect to the PSO iteration number was also measured in terms of PSNR. We tested the PSODL reconstruction performance from 5 to 60 iterations. To show the details more clearly, we used the data in Table 6 to make a line chart. Fig. 4 shows the PSNR change with respect to the change of the iteration number. It is easily seen that, at the beginning, the PSNR increased with the iteration number. However, when the iterations reached a special point, the PSNR growth trend slowed and stabilized. Therefore, our method still maintains some of the PSO algorithm properties.

5. Conclusion

In this paper, we proposed an approach to represent large-scale remote sensing images based on introducing PSO into online dictionary learning. By establishing a new strategy of selecting atoms and modeling the atom-updating stage using PSO, the proposed algorithm improves the performance of ODL algorithms in terms of the accuracy of large-scale remote sensing images. We did some experiments with two satellite image sets to test the proposed algorithm. The results demonstrate that, with the same sparsity, the precision of the representation by our proposed algorithm is higher than that of ODL. On the condition of the same precision control, the sparsity of results achieved by the proposed algorithm is almost the same as the sparsity of those achieved by ODL. Furthermore, our algorithm also has a better effect on noise suppression. In the future work, we plan to add some priors from geographical information or image texture to improve a faster convergence.

Acknowledgements

This work was partially supported by the NSFC (Grant Nos. 61272314 & 61271435), Supported by Beijing Natural Science Foundation (4141003).

References

[1] Stephane G. Mallat, A theory for multiresolution signal decomposition: the wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell. 11 (7) (1989) 674–693.
[2] E.J. Candes, L. Demanet, D.L. Donoho, L. Ying, Fast discrete curvelet transforms, Multiscale Model. Simul. 5 (3) (2006) 861–899.
[3] E. LePennec, S. Mallat, Sparse geometric image representations with bandelets, IEEE Trans. Image Process. 14 (4) (2005) 423–438.
[4] V. Velisavljevic, B. Beferull-Lozano, M. Vetterli, P.L. Dragotti, Directionlets: anisotropic multidirectional representation with separable filtering, IEEE Trans. Image Process. 15 (7) (2008) 1916–1933.
[5] S. Mallat, Geometrical grouplets, Appl. Comput. Harmon. Anal. 26 (2) (2009) 161–180.
[6] K. Engan, S.O. Aase, J. Hakon Husoy, Method of optimal directions for frame design, in: 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1999 Proceedings, vol. 5, 1999, pp. 2443–2446.
[7] R. Vidal, Y. Ma, S. Sastry, Generalized principal component analysis (gpca), Trans. Pattern Anal. Mach. Intell. 27 (12) (2005) 1945–1959.
[8] M. Aharon, M. Elad, A. Bruckstein, K-svd: an algorithm for designing overcomplete dictionaries for sparse representation, Trans. Signal Process. 54 (11) (2006) 4311–4322.
[9] Alain Rakotomamonjy, Direct optimization of the dictionary learning problem, IEEE Trans. Signal Process. 61 (2013) 5495–5506.
[10] S. Wang, Y. Xia, Q. Liu, P. Dong, D. Feng, J. Luo, Fenchel duality based dictionary learning for restoration of noisy images, IEEE Trans. Image Process. 22 (12) (2013) 5214.
[11] Mostafa Sadeghi, Massoud Babaie-Zadeh, Christian Jutten, et al., Dictionary learning fro sparse decomposition: a novel approach, IEEE Signal Process. Lett. 20 (12) (2013) 1195–1198.
[12] Mingyuan Zhou, Haojun Chen, John Paisley, Lu Ren, Lingbo Li, Zhengming Xing, David Dunson, Guillermo Sapiro, Lawrence Carin, Nonparametric bayesian dictionary learning for analysis of noisy and incomplete images, Trans. Imgage Process. 21 (1) (2012) 130–144.
[13] J.M. Hughes, D.N. Rockmore, Y. Wang, Bayesian learning of sparse multiscale image representations, IEEE Trans. Image Process. 22 (12) (2013) 4972–4983.
[14] Daniele Barchies, Mark D. Plumbley, Learning incoherent dictionaries for sparse approximation using iterative projections and rotations, IEEE Trans. Signal Process. 61 (5–8) (2013) 2055–2065.
[15] Gianluca Monaci, Philippe Jost, Pierre Vandergheynst, Boris Mailhe, Sylvain Lesage, Rémi Gribonval, Learning multimodal dictionaries, IEEE Trans. Image Process. 16 (9) (2007) 2272–2283.
[16] E. Kidron, Y.Y. Schechner, M. Elad, Cross-modal localization via sparsity, IEEE Trans. Signal Process. 55 (4) (2007) 1390–1404.
[17] Jianchao Yang, Zhaowen Wang, Zhe Lin, Scott Cohen, Thomas Huang, Coupled dictionary training for image super-resolution, IEEE Trans. Image Process. 21 (8) (2012) 3467–3478.
[18] Leslie N. Smith, Michael Elad, Improving dictionary learning: multiple dictionary updates and coefficient reuse, IEEE Signal Process. Lett. 20 (1) (2013) 79–82.
[19] Boaz Ophir, Michael Lustig, Michael Elad, Multi-scale dictionary learning using wavelets, IEEE J. Sel. Top. Signal Process. 5 (5) (2011) 1014–1024.
[20] J. Mairal, G. Sapiro, M. Elad, Multiscale sparse image representation with learned dictionaries, in: IEEE International Conference on Image Processing, 2007, ICIP 2007, vol. 3, 2007, pp. III–105–III–108.
[21] R. Yan, L. Shao, Y. Liu, Nonlocal hierarchical dictionary learning using wavelets for image denoising, IEEE Trans. Image Process. 22 (12) (2013) 4689.
[22] Toshimitsu Aritake, Hideitsu Hino, Noboru Murata, Learning ancestral atom via sparse coding, IEEE J. Sel. Top. Signal Process. 7 (2013) 586–594.

Please cite this article in press as: L. Wang et al., Particle Swarm Optimization based dictionary learning for remote sensing big data, Knowl. Based Syst. (2014), http://dx.doi.org/10.1016/j.knosys.2014.10.004

[23] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, Andrew Zisserman, Discriminative learned dictionaries for local image analysis, in: IEEE Conference on Computer Vision and Pattern Recognition, 2008, CVPR 2008, IEEE, 2008, pp. 1–8.
[24] Qiang Zhang, Baoxin Li, Discriminative k-svd for dictionary learning in face recognition, in: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2010, pp. 2691–2698.
[25] Zhuolin Jiang, Zhe Lin, Larry S. Davis, Learning a discriminative dictionary for sparse coding via label consistent k-svd, in: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2011, pp. 1697–1704.
[26] Zhuolin Jiang, Zhe Lin, Larry S. Davis, Label consistent k-svd: learning a discriminative dictionary for recognition, IEEE Trans. Pattern Anal. Mach. Intell. 99 (1) (2013) 1.
[27] Mehrdad J. Gangeh, Ali Ghodsi, Mohamed S. Kamel, Kernelized supervised dictionary learning, IEEE Trans. Signal Process. 61 (17–20) (2013) 4753–4767.
[28] Julien Mairal, Francis Bach, Jean Ponce, Task-driven dictionary learning, IEEE Trans. Pattern Anal. Mach. Intell. 34 (4) (2012) 791–804.
[29] K. Skretting, K. Engan, Image compression using learned dictionaries by rls-dla and compared with k-svd, in: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2011, pp. 1517–1520.
[30] J. Mairal, F. Bach, J. Ponce, G. Sapiro, Online dictionary learning for sparse coding, in: Proceedings of the 26th Annual International Conference on Machine Learning, ICML'09, ACM, 2009, pp. 689–696.
[31] Hongkai Xiong, Zhiming Pan, Xinwei Ye, Chang Wen Chen, Sparse spatio-temporal representation with adaptive regularized dictionary learning for low bit-rate video coding, IEEE Trans. Circ. Syst. Video Technol. 23 (4) (2013) 710–728.
[32] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of IEEE International Conference on Neural Networks, vol. 4, 1995, pp. 1942–1948.
[33] L. Jacob, G. Obozinski, J.P. Vert, Group lasso with overlap and graph lasso, in: Proceedings of the International Conference on Machine Learning (ICML), 2009.
[34] Y. Pati, R. Rezaiifar, P. Krishnaprasad, Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition, in: 1993 Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers, vol. 1, 1993, pp. 40–44.
[35] L. Bottou, O. Bousquet, The trade-offs of large scale learning, vol. 20, 2008.
[36] Michal Aharon, Michael Elad, Alfred Bruckstein, K-svd: an algorithm for designing overcomplete dictionaries for sparse representation, IEEE Trans. Signal Process. 54 (11) (2006) 4311–4321.
[37] Y. Shi, R. Eberhart, A modified particle swarm optimizer, in: IEEE World Congress on Computational Intelligence, 1998, pp. 69–73.

Please cite this article in press as: L. Wang et al., Particle Swarm Optimization based dictionary learning for remote sensing big data, Knowl. Based Syst. (2014), http://dx.doi.org/10.1016/j.knosys.2014.10.004