Banouar *et al. EURASIP Journal on Advances in Signal Processing* (2018) 2018:72
https://doi.org/10.1186/s13634-018-0591-3

EURASIP Journal on Advances in Signal Processing

RESEARCH

Open Access

Collaborating filtering using unsupervised learning for image reconstruction from missing data

Oumayma Banouar[1,2*], Souad Mohaoui[1] and Said Raghay[1]

Abstract

In the image acquisition process, important information in an image can be lost due to noise, occlusion, or even faulty image sensors. Therefore, we often have images with missing and/or corrupted pixels. In this work, we address the problem of image completion using a matrix completion approach that minimizes the nuclear norm to recover missing pixels in the image. The image matrix has a low rank. The proposed approach uses the nuclear norm function as a surrogate of the rank function in the aim to resolve the problem of rank minimization that is known as an NP-hard problem. It is an adaptation of the collaborating filtering approach used for users' profile construction. The main advantage of this approach is that it uses a learning process to classify pixels into clusters and exploits them to run a predictive method in the aim to recover the missing or unknown data. For performance evaluation, the proposed approach and the existing matrix completion methods are compared for image reconstruction according to the PSNR measure. These methods are applied on a dataset composed of standard images used for image processing. All the recovered images obtained during experimentation are also dressed to compare them visually. Simulation results verify that the proposed approach achieves better performances than the existing matrix completion methods used for image reconstruction from missing data.

Keywords: Image reconstruction, Bi-clustering, Matrix completion, Unsupervised learning, Prediction, Rank function, Nuclear norm function, Surrogate model

1 Introduction

The reconstruction of missing pixels from an incomplete image is a very active research area in image processing. A simple model for such problem can be defined as follows: given an image which is incomplete, i.e., it has missing pixels, the purpose is to fill its missing pixels based on some observed pixels. In analogy with matrix completion problem, the problem of recovering missing pixels in an image can be referred to as image completion problem.

In this work, we are interested in recovering missing pixels from an incomplete image using a matrix completion method based on the minimization of the nuclear norm of a matrix. The nuclear norm minimization is a category of low-rank matrix approximation methods. Mathematically speaking, given an incomplete image X, missing values are estimated given observed pixels: $\{D_{ij}/i,j \in \Omega\}$ where Ω denotes the set of observed entries. The common assumption is that the matrix should be low-rank (most images have low rank). Then, a direct approach is to minimize the rank of the matrix with certain constraints. This problem is NP-hard; a convex relaxation is often used to make the minimization tractable. As the rank function is simply the number of nonvanishing singular values, the most appropriate choice is to replace the rank function with the nuclear norm. Therefore, the proposed approach is based on the nuclear norm minimization that is the surrogate model of the rank minimization.

The approach used in this work is proposed in [1] for users' profile construction. It uses a matrix completion method based on nuclear norm optimization of the matrix to predict users' preferences about items. A bi-clustering process is adopted to detect users' clusters and items' clusters in the aim to promote the personal relevancy

*Correspondence: o.banouar@edu.uca.ma
[1]Department of Applied Mathematics and Computer Science, Faculty of Science and Technics, Cadi Ayyad University, Marrakesh, Morocco
[2]Higher School of Technology, Sidi Mohamed Ben Abdellah University, Fez, Morocco

concept [2]. It applies the prediction process on the ratings given by users that share almost the same preferences.

The main problem with recovering missing data in images is the sparcity of the matrix that modelize them. With the same principle, we adapt the users' profile construction method to recover the missing pixels. The obtained experimental results proved the efficiency of the proposed prediction process. The proposed approach is applied on a benchmark that contains standard images for image processing. They are gray-level images that have different histograms. The obtained results are compared visually to those obtained by applying different nuclear norm optimization algorithms. The peak signal-to-noise ratio (PSNR) measure is also calculated for each recovered image.

The remaining of this article is organized as the follows. Section 2 presents the role of nuclear norm minimization in the optimization of low-rank matrices. It exposes then the problem statement and explains the proposed approach. It also reviews the related works. Section 3 addresses the experimental protocol and discusses the obtained results. The conclusion closes the paper.

2 Methods

2.1 Minimization of low-rank matrices using nuclear norm minimization

In the area of engineering and applied science such as machine learning and computer vision, a wide range of problems can be or have been represented under low-rank minimization framework, since the low-rank formulation seems to be able to capture the low-order structure of the underlying problems.

In many practical problems, one would like to guess the missing entries of an $n_1 \times n_2$ matrix from a sampling Ω of its entries. This problem is known as the matrix completion problem. It comes up in a great number of applications including those of collaborating filtering. The collaborating filtering is the task of automatic predicting of the entries in an unknown data matrix. A popular example is the movie recommendation case where the task is to make automatic predictions about the interests of a user by collecting taste information from its formal interests or by collecting them from other users.

In mathematical terms, this problem is posed as follows:

A data matrix $X \in \mathbb{R}^{n_1 \times n_2}$ is the matrix to be known as much as possible. The only information available about it is a sampling set of entries $M_{ij}, (i,j) \in \Omega$, where Ω is a subset of the complete set of entries $\{1, .., n_1\} \times \{1, .., n_2\}$.

Very few factors contribute to an individual's tastes. Therefore, the problem of matrix completion is an optimization problem of a low-rank r matrix from a sample of its entries. The matrix rank satisfies $r \leq \min(n_1, n_2)$. Such a matrix is represented by counting $n_1 \times n_2$ numbers but has only $r \times (n_1 \times n_2 - r)$ degrees of freedom. When the matrix rank is small and its dimension is large, then the data matrix carries much less information than its ambient dimension suggests. In the case of collaborative prediction movie recommendation system, users—rows of the matrix—are given the opportunity to rate items—columns of the data matrix. However, they usually rate very few ones so there are very few scattered observed entries of this data matrix. In this case, the users-ratings matrix is approximately low-rank, because as mentioned, it is commonly believed that only very few factors contribute to an individual's tastes or preferences. These preferences are stored in a user profile [1]. In the same analogy, matrix completion can be used to restore images with missing data. From limited information, we aim to recover the image, i.e., infer the many missing pixels.

2.2 Problem statement

Given $\Omega \subset [n_1] \times [n_2]$ a set of elements of an unknown rank-r matrix, $X \in \mathbb{R}^{n_1 \times n_2}$. The values of elements $M_{ij}, (i,j) \in \Omega$ are known. The task is to recover incomplete matrix X. Formally, the low-rank matrix completion problem is given by:

$$\begin{cases} \text{minimize} & \text{rank}(X) \\ \text{subject to} & P_\Omega(X) = P_\Omega(M) \end{cases} \tag{1}$$

where $P_\Omega : \mathbb{R}^{n_1 \times n_2} \longrightarrow \mathbb{R}^{n_1 \times n_2}$ is the orthogonal projection onto the subspace of matrices that vanish outside of Ω, $(i,j) \in \Omega$ if and only if M_{ij} is observed. $P_\Omega(X)$ is defined by:

$$P_\Omega(X) = \begin{cases} X_{ij} & \text{if } (i,j) \in \Omega \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

The data known in M is given by $P_\Omega(M)$. The matrix X is recovered then from $P_\Omega(X)$ if it is the unique matrix of rank less or equal to r and consistent with the data.

In a practical point of view, the rank minimization problem is an NP-hard problem. Algorithms are not capable to resolve it in time once the matrices have an important dimension. They require time doubly exponential in the dimension of the matrix to find the exact solution. Authors in [3] proposed the nuclear norm minimization method. Replacing the rank of a matrix by its nuclear norm can be justified as a convex relaxation (the nuclear norm $\|X\|_* = \sum_i \sigma_i(X)$ is the largest convex lower bound of $rank(X)$ on the ball $\{X / \|X\|_* = \sigma(X) \leq 1\}$ [3]). Consequently, the problem (2) is then replaced by the following:

$$\begin{cases} \text{minimize } \nu & \|X\|_* \\ \text{subject to} & P_\Omega(X) = P_\Omega(M) \end{cases} \tag{3}$$

where the nuclear norm $\|X\|_*$ is defined as the sum of its singular values: $\|X\|_* = \sum_i \sigma_i(X)$.

Since the nuclear norm ball $\{X : \|X\|_* \leq 1\}$ is the convex hull of the set of rank-one matrices with spectral norm bounded by one, authors in [4] interpret that under suitable conditions, the rank minimization program (2) and the convex program (3) are formally equivalent in the sense that they have exactly the same unique solution.

Matrix completion problem is not as ill posed as thought. It is possible to resolve it by convex programming. The rank function counts the number of nonvanishing singular values when the nuclear norm sums their amplitude. The nuclear norm is a convex function. It can be optimized efficiently via semidefinite programming.

The following theorem is demonstrated by authors in [4].

Theorem 1 *Let M be an $n_1 \times n_2$ matrix of a rank r sampled from the random orthogonal model, and put $n = \max(n_1, n_2)$. Suppose we observe m entries of M with locations sampled uniformly at random. Then, they are numerical constants C and c such that if:*

$$m \geq Cn^{5/4} r \log n \tag{4}$$

The minimizer to the problem (3) is unique and equal to M with probability at least $1 - cn^{-3}$; that is to say, the semidefinite program (3) recovers all the entries of M with no error. In addition, if $r \leq n^{1/5}$, then the recovery is exact with probability at least $1 - cn^{-3}$ provided that:

$$m \geq Cn^{(6/5)} r \log n \tag{5}$$

Under the hypothesis of Theorem 1, there is a unique low-rank matrix, which is consistent with the observed entries. This matrix can be recovered by the convex optimization (3). For most problems, the nuclear norm relaxation is formally equivalent to the combinatorial hard rank minimization problem.

If the coherence is low, few samples are required to recover M. As an example, matrices with incoherent column and row space matrices with random orthogonal model or those with small components of the singular vectors of M.

Conventional semidefinite programming solvers such as SDPT3 [5] and SeDeMi [6] solve the problem (3). However, such solvers are usually based on interior-point methods and cannot deal with large matrices. They can only solve problems of size at most hundreds by hundreds on a moderate computer. These solvers are problematic when the size of the matrix is large. They need to solve huge systems of linear equations to compute the Newton direction. To be precise, SDTP handles only square matrices with the size less than 100. Another alternative is to think of using iterative solvers such as the method of conjugate gradients to solve the Newton system. However, it is still problematic as well since it is well known that the condition number of the Newton system increases rapidly as one gets closer to the solution. Furthermore, none of these general-purpose solvers use the fact that the solution may have low rank.

Therefore, the first-order methods are used to complete large low-rank matrices by solving (3).

In the special matrix completion setting presented in (3), $P_\Omega(X)$ is the orthogonal projector onto the span of matrices vanishing outside of Ω. Therefore, the (i,j)th component of $P_\Omega(X)$ is equal to X_{ij} if $(i,j) \in \Omega$ and 0 otherwise. $X \in \mathbb{R}^{n_1 \times n_2}$ is then the optimization variable. Fix $\tau > 0$ and a sequence δ_k of scalar step sizes. Starting with $Y_0 = 0 (\in \mathbb{R}^{n_1 \times n_2})$, the algorithm defines until a stopping criterion is reached:

$$\begin{cases} X_k = \text{shrink}(Y_{k-1}, \tau) \\ Y_k = Y_{k+1} + \delta_k P_\Omega(M - X_k) \end{cases} \tag{6}$$

shrink(x, λ) is a nonlinear function that applies a soft-thresholding rule at level λ to the singular values of the input matrix. The key property here is that for large values of τ, the sequence X_k converges to a solution which very nearly minimizes (3). Hence, at each step, one only needs to compute at most one singular value decomposition and perform a few elementary matrix additions.

2.3 The singular value thresholding algorithm

The most popular approaches to matrix completion in literature are the thresholding methods that can be divided into two groups: one-step thresholding methods and iterative thresholding methods. Despite the strong theoretical guaranties which have been obtained for one-step thresholding procedures, they show poor behavior in practice and only work under the uniform sampling distribution which is not realistic in many practical situations [7]. On the other hand, iterative thresholding methods are well adapted for general nonuniform distribution as well as they show practical performances as in [4]. Authors in [8] proposed a first-order singular value thresholding algorithm SVT which is a key subroutine in many numerical schemes for solving nuclear norm minimization. The conventional approach for SVT is to find the singular value decomposition SVD of the matrix, then to shrink its singular values.

The singular value decomposition step

The singular value shrinkage operator is the key building block of the SVT algorithm. Consider the singular value decomposition SVD of a matrix $X \in \mathbb{R}^{n_1 \times n_2}$ of rank r

$$X = U \Sigma V^* \quad \text{Where} \quad \Sigma = diag(\{\sigma_i\}_{1 \leq i \leq \tau}) \tag{7}$$

where U and V are respectively $n_1 \times r$ and $n_2 \times r$ matrices with orthonormal columns, and the singular values σ_i are positive. For each $\tau \geq 0$, the soft-thresholding operator D_τ is defined as follows:

$$D_\tau(X) = UD_\tau(\Sigma)V^* = U.\begin{pmatrix} (\sigma_1-\tau)_+ & & \\ & \ddots & \\ & & (\sigma_r-\tau)_+ \end{pmatrix}.V^* \tag{8}$$

where t_+ is the positive part of t defined by:

$$(\sigma_i-\tau)_+ = \max(0, \sigma_i-\tau) = \begin{cases} \sigma_i - \tau & \text{if} \quad \sigma_i - \tau > 0 \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

In other words, in $D_\tau(X)$, the singular vectors of X are kept and the singular values are shrinked by the soft-thresholding.

Even though the SVD may not be unique, it is easy to see that the singular value shrinkage operators are well defined. In some sense, this shrinkage operator is a straightforward extension of the soft-thresholding rule for scalars and vectors. In particular, note that if many of the singular values of X are below the threshold τ, the rank of D_τ may be considerably lower than that of X, just like the soft-thresholding rule applied to vectors leads to sparser outputs whenever some entries of the input are below threshold.

The singular value thresholding operator is the proximal operator associated with the nuclear norm. The proximal operator has its origins in convex optimization theory, and it has been widely used for non-smooth convex optimization problems, such as the l_1-norm minimization problems arising from compressed sensing [9] and related areas. It is well known that the proximal operator of the l_1-norm is the soft-thresholding operator, and soft-thresholding-based algorithms are proposed to solve l_1-norm minimization problems [10].

Shrinkage iteration step

The singular value thresholding SVT algorithm approximates the minimization (3) by:

$$\begin{cases} \min_X \tau\|X\|_* + \frac{1}{2}\|X\|_F^2 \\ \text{subject to} \quad X_{ij} = M_{ij} \end{cases} \tag{10}$$

with a large parameter τ . $\|.\|_F$ denotes the matrix Frobenius norm or the square root of the summation of squares of all entries. Then, it applies a gradient ascent algorithm to its dual problem. The iteration is:

$$\begin{cases} X_k = D_\tau(Y_{k-1}, \tau) \\ Y_k = Y_{k+1} + \delta_k P_\Omega(M - X_k) \end{cases} \tag{11}$$

where D_τ is the SVT operator defined as:

$$D_\tau = \arg\min \frac{1}{2}\|Y - X\|_F + \tau\|X\|_* \quad, X \in \mathbb{R}^{n_1 \times n_2} \tag{12}$$

The iteration is called the SVT algorithm, and it was shown in [11] to be an efficient algorithm for huge low-rank matrix completion. Two crucial properties make the SVT algorithm suitable for matrix completion.

- Low-rank property: The matrices X_k turn out to have low rank, and hence, the algorithm has minimum storage requirement since it only needs to keep principal factors in memory.
- Sparsity: For each $k \geq 0$, Y_k vanishes outside of Ω and is, therefore, sparse, a fact, which can be used to evaluate the shrink function rapidly.

The SVT algorithm

The initial step of the SVT algorithm is to start with the following:

- $Y_0 = 0$;
- Choosing a large τ to make sure that the solution of (11) is close enough to the solution of (3).
- Defining k_0 as the integer that obeys to: $\frac{\tau}{\delta\|P_\Omega(M)\|} \in (k_0 - 1, k_0)$
- Since $Y_0 = 0$, $X_k = 0$, $Y_k = k\delta P_\Omega(M), k = 1, ..., k_0$

The stopping criteria of the SVT algorithm is motivated by the first-order optimality conditions for the minimization of the problem (10). The solution X_τ^* to (11) must verify:

$$\begin{cases} X = D_\tau(Y) \\ P_\Omega(X - M) = 0 \end{cases} \tag{13}$$

where Y is a matrix vanishing outside of Ω^c. Therefore, to make sure that X_k is close to X_τ^*, it is sufficient to check how close (X_k, Y_{k-1}) is obeying (13). By definition, the first equation in (13) is always true. Therefore, it is natural to stop (12) when the error in the second equation is below a specified tolerance:

$$\frac{\|P_\Omega(X - M)\|_F}{\|P_\Omega(M)\|_F} \leq \epsilon \tag{14}$$

The matrix completion problem can be viewed as a special case of the matrix recovery matrix, where one has to recover the missing entries of a matrix, given limited number of known entries.

2.4 Literature review

Other works dressed other algorithms in the attempt to minimize the nuclear norm of low-rank sparse matrix. Authors in [12] presented the fixed-point continuation (FPC) algorithm. It combines the fixed-point continuation [13] with Bregman iteration [14]. The iteration is as follows:

$$\begin{cases} \text{Iterate on } i \text{ to get } X_k \begin{cases} X_i = D_\tau(Y_{i-1}) \\ Y_i = X_{i-1} + \delta_i P_\Omega(M + Z_{k-1} - X_i) \end{cases} \\ Z_k = Z_{k-1} + P_\Omega(M * X_k) \end{cases} \tag{15}$$

In fact, the FPC algorithm is a gradient ascent algorithm applied to an augmented Lagrangian of (3). The augmented Lagrangian multiplier (ALM) method in [15] reformulates the problem into:

$$\begin{cases} \min_X \|X\|_* \\ \text{subject to } \; X + E = P_\Omega(M), \;\; P_\Omega(E) = 0, \end{cases} \tag{16}$$

where E is an auxiliary variable. The corresponding (partial) ALM function is:

$$\Gamma(X, E, Y, \mu) = \|X\|_* + < Y, P_\Omega(M) - X - E > + \frac{\mu}{2} \|P_\Omega(M) - X - E\|_F^2 \text{ with } P_\Omega(E) = 0 \tag{17}$$

An inexact gradient ascent is applied to the ALM and leads to the following algorithm:

$$\begin{cases} X_k = D_{\mu_k^{-1}}\left(P_\Omega(M) - E_k + \mu_k^{-1} Y_{k-1}\right) \\ E_k = P_{\Omega^c}(X_k) \\ Y_k = Y_{k-1} + \mu_k P_\Omega(M - X_k) \end{cases} \tag{18}$$

For all these algorithms, the SVT operator is the key to make them converge to low-rank matrices.

Just like the FPC and SVT algorithms, the proximal gradient (PG) [16] algorithm for matrix completion needs to compute the SVD at each iteration. It is as simple as the cited algorithms.

There are two main advantages of the SVT algorithm over the FPC and the PG algorithms when the former is applied to solve the problem of matrix completion.

First, in some cases, we dispose a sequence of low-rank iterates; in contrast, so many iterates at the initial phase of the FPC or PG algorithms may not have low rank even though the optimal solution itself has low rank. We observed this behavior when we applied them to solve the problem of matrix completion.

Second, the intermediate matrices generated during the resolution of our problem are sparse due to the sparcity of Ω, the set of observation. This makes the SVT algorithm computationally more attractive. Indeed, the generated matrices by FPC and PG algorithms may not be sparse and specially for the last one.

The first-order methods presented above are the basis of a number of recent works that minimize the nuclear norm of a matrix to recover an image with missing data.

In [17], authors proposed a two-step proximal gradient algorithm to solve nuclear norm regularized least squares for the purpose of recovering low-rank data matrix from sampling of its entries. Each iteration generated by the proposed algorithm is a combination of the latest three points, namely, the previous point, the current iterate, and its proximal gradient point. This algorithm preserves the computational simplicity of classical proximal gradient algorithm [16] where a singular value decomposition in proximal operator is involved. Global convergence is followed directly in the literature.

Authors in [18, 19] adopted the SVT algorithm to achieve the completed matrix but by using the power method [20] instead of using PROPACK [21] for computing the singular value decomposition of large and sparse matrix. They showed that accelerating Soft-Impute is indeed possible while still preserving the "sparse plus

Fig. 1 Benchmark of images. Set of images used in the tests. This is a set of benchmark used as an original image in the tests

Table 1 PSNR values of the reconstructed images using different ratios

Image	Method	15% missing data	25% missing data	35% missing data
Lena	FPC	46.2950	45.8888	45.6461
	ALM	48.2790	48.2248	45.5295
	IALM	48.2673	48.2278	46.0819
	PPG	45.0964	41.9579	40.2865
	PG	45.0966	41.9582	40.2863
	SVT with power method	48.3221	48.2745	48.2295
	Proposed approach	*48.9848*	*48.7817*	*48.5952*
Cameraman	FPC	28.2855	32.3433	31.7894
	ALM	36.8075	33.9694	29.5169
	IALM	36.7101	33.9849	29.8495
	PPG	31.3222	27.9425	24.3797
	PG	31.3199	27.9458	24.3757
	SVT with power method	40.9528	40.5182	40.0158
	Proposed approach	*43.3525*	*42.4823*	*40.4596*
Flinstones	FPC	28.9102	37.4826	37.2896
	ALM	42.9923	37.3583	34.3512
	IALM	42.8769	37.4573	34.6774
	PPG	31.5781	29.5989	24.3693
	PG	31.5936	29.5887	24.3708
	SVT with power method	43.6614	43.5614	43.2075
	Proposed approach	*47.8234*	*47.3814*	*44.3920*
House	FPC	28.6407	27.7999	27.4826
	ALM	37.0989	36.4579	36.3583
	IALM	36.8952	36.5262	36.4573
	PPG	31.5992	29.8729	29.5998
	PG	31.6005	29.8769	29.5701
	SVT with power method	38.1457	38.0827	37.6494
	Proposed approach	*39.0707*	*38.1743*	*37.8234*
Man	FPC	39.2493	38.5864	38.1565
	ALM	41.3640	41.2133	40.4455
	IALM	41.2952	41.2191	40.8616
	PPG	39.4997	39.1895	39.3266
	PG	39.5008	39.1889	39.3264
	SVT with power method	42.9572	42.4158	42.0357
	Proposed approach	*44.3212*	*43.9939*	*43.6724*

For each test setting, six results are provided: FPC, ALM, IAM, PPG, PG, SVT with power method, and our proposed model. The best value of the PSNR is the number written in italics on each row

low rank" structure. To further reduce the iteration time complexity, instead of computing SVT exactly using PROPACK, they proposed an approximate SVT scheme based on the power method. Though the SVT obtained in each iteration is only approximate, they demonstrated that convergence can still be as fast as performing exact SVT. Hence, the resultant algorithm has low iteration complexity and fast convergence rate. Our objective is to increase the accuracy and the precision of image completion results by adopting unsupervised learning process that takes into account the characteristics of image pixels.

2.5 Nuclear norm minimization-based collaborating filtering for image reconstruction

In the problem of collaborating filtering based on nuclear norm minimization, the goal is to predict entries of an unknown matrix based on a subset of its observed entries. For example in a collaborative prediction movie recommendation system, where the rows of the matrix represent users and columns represent movies, the task is to predict ratings that users gave to movies based on their preferences. The prediction of users' preferences over movies—they have not yet seen—are then based on patterns in the partially observed rating matrix. The setting can be formalized as a matrix completion problem completing entries in a partially observed data matrix.

Algorithm 1 Collaborating filtering based algorithm

Input: Image with missing data X, the smoothing degree m_u
$b = \emptyset$ set of observed pixels values
$\Omega = \emptyset$ set of observed pixels indices
Find the clusters C_u and C_i
m=size(C_u)
n=size(C_i)
for u from 1 to m do
for i from 1 to n do
if exist(X_{ui}) then
b.append(X_{ui})
Ω.append((u,i))
end if
end for
end for
$\widehat{X} = SVT([m,n], \Omega, b, m_u)$
if notexist($\widehat{X}_{u_k,i_{k'}}$) then
Apply median filtering on 3-by-3 neighbours
end if
Output: Recovered matrix $\widehat{X}$

In the same analogy for image completion problem, the collaborating filtering setting aims to predict the pixels missing in the image based on the partially observed entries, i.e., pixels in the image. The proposed approach then is based on two main steps:

- Clustering step: uses a learning process to identify pixels' clusters.
- Prediction step: uses a predictive method based on clusters found in the first step to predict the unknown pixels.

Clustering defines the optimal partitioning of a given set of N data points into K subgroups. The points belonging to the same group are as similar as much as possible. However, data points from two different groups share the maximum difference.

The first step of our approach is to perform a data filtering. The learning process starts by applying a principal component analysis (PCA) in the attempt to reduce the number of variables and make the information less redundant. As a result, our data are centered. To detect the pixels' clusters, the process adopts a bi-clustering step founded on prototype-based clustering by using the K-means algorithm on the principal component scores, that is, the representation of the data matrix in the principal component space and its correlation matrix.

The second process takes place to predict the missing pixels using the clusters, which performs a new framework for predicting the missing pixels. The clustering phase regroups automatically the pixels of an image into different homogeneous regions. These homogeneous regions usually contain similar objects or part of them. As a result, interesting performance will be achieved in the prediction step.

Fig. 2 Results obtained for image "man" using the different methods. From left to right, the order of the images is as follows: simulated image with missing data, recovered image using FPC algorithm, recovered image using ALM algorithm, recovered image using IALM algorithm, recovered image using PPG algorithm, recovered image using PG algorithm, recovered image using SVT with power method, and finally the recovered image using our proposed approach

For a given point in the image, we identify clusters in which the selected pixel row index, respectively the column index, belongs. The predicted value is the result of singular value thresholding (SVT) algorithm applied on the matrix containing values of pixels existing in the intersection between the two clusters found in step 1. The adopted algorithm takes as parameters three mandatory elements:

- Ω the set of locations corresponding to the observed entries.
- b the linear vector which contains the observed elements.
- m_u the smoothing degree.

The set of locations corresponding to the observed entries Ω might be defined in three forms:

- The first one as a sparse matrix where only the elements different of 0 are to take into account.
- The second one as a linear vector that contains the position of the observed elements.
- The third one where Ω is specified as indices (i, j) with $(i, j) \in \mathbb{N}$.

The application of the proposed algorithm in image completion procures in some cases certain results that are out of range. In this case, we propose to use a median filtering on the predicted pixels. The median filter is often used as a typical pre-processing step to improve the result of later process in signal processing (for example, edge detection on an image). The idea is to use it as a final process to replace each entry (here, entries are the predicted pixels) with the median of neighboring entries, which performs a good result in image reconstruction as shown in the experimental results.

Fig. 3 Results obtained for image "cameraman" using the different methods. From left to right, the order of the images is as follows: simulated image with missing data, recovered image using FPC algorithm, recovered image using ALM algorithm, recovered image using IALM algorithm, recovered image using PPG algorithm, recovered image using PG algorithm, recovered image using SVT with power method, and finally the recovered image using our proposed approach

The result of our proposed approach is a completed data matrix that contains all the pixels' values. The goal of the proposed approach is to predict the missing pixels in the image matrix. Our learning process detects the partitions of pixels' indices where the predicting process exploits the clusters found to predict the missing value. It works on the assumption that pixels in the same cluster share almost the same characteristics in the image.

3 Results and discussion

The proposed approach is compared with several state-of-the-art matrix completion methods including the following: fixed point continuation (FPC) algorithm [22], proximal gradient (PG) algorithm [16, 18, 19], partial proximal gradient (PPG) algorithm [16], augmented Lagrange multiplier (ALM) algorithm [15], and inexact augmented Lagrange multiplier (IALM) algorithm [15]. All these methods need the PROPACK Package [21] for computing the SVD for large and sparse matrix. Our approach was also compared to the method presented in [18, 19] that used the power method [20].

The images used are the standards for image processing. We chose a benchmark of five images (Fig. 1) with different gray-level histogram. The computed results are the peak signal-to-noise ratio (PSNR).

We constructed images with arbitrary missing data from the specified benchmark.

In Table 1, the PSNR values are shown for the five different images in Fig. 1 with different choices of the percentage of missing data. The best value of the PSNR is the number written in italics on each row. We can deduce then that our proposed approach is always better than the others, which assures the efficiency of our algorithm.

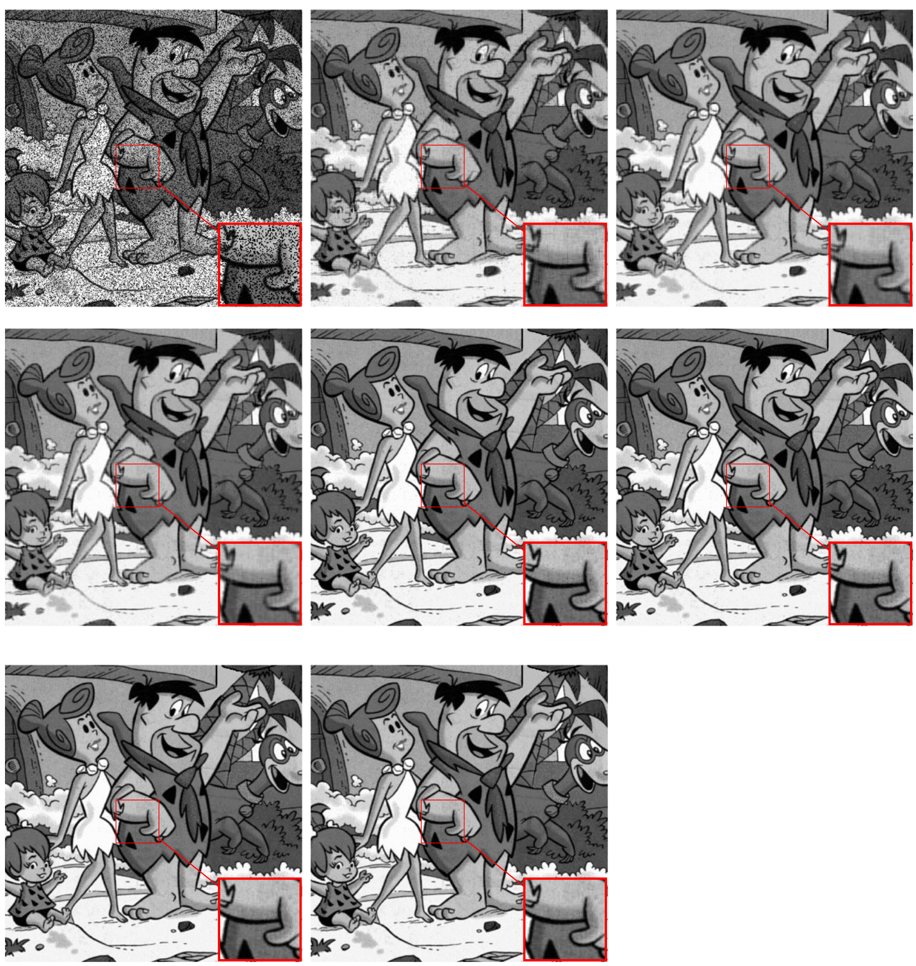

Fig. 4 Results obtained for image "flinstones" using the different methods. From left to right, the order of the images is as follows: simulated image with missing data, recovered image using FPC algorithm, recovered image using ALM algorithm, recovered image using IALM algorithm, recovered image using PPG algorithm, recovered image using PG algorithm, recovered image using SVT with power method, and finally the recovered image using our proposed approach

In Figs. 2, 3, 4, 5, and 6, we have shown the simulated images compared with the recovered images using our proposed approach, FPC, ALM, IALM, PPG, and PG algorithms. Visually, we can ensure that our approach predicts the missing pixels effectively.

The execution of the main proposed algorithm requires an average of 2 to 10 min on 2.60 GHz Intel i7 core computer for 256 × 256 grayscale images.

The fact that our approach adopts a clustering step to detect the regions with similar pixels allowed us to augment the relevancy and the precision of our SVT-based prediction process. Indeed, when the SVT algorithm is applied on the sub-matrix that contains the pixels of the same cluster, the predicted values procured better PSNR and reconstructed images that are visually consistent more than the SVT algorithm using the power method presented by [18–20]. In addition, the sparcity of the observation matrix made the SVT algorithm the most suitable resolution method for matrix completion problem. Indeed, when recovering the missing image pixels, the FPC, PG, and ALM algorithms procured at their initial phase many iterates that have not a low rank though the optimal solution itself has low rank.

4 Conclusions

We propose in this work a new method for image reconstruction from missing data. It is based on two main steps. The first one is a bi-clustering process using K-means algorithm to identify pixels' clusters. It is applied on the matrix of PCA scores and its correlation. The second step predicts the missing pixels by applying a matrix completion algorithm on the observations' matrices obtained using the clusters found in step 1. In each iteration, a matrix of observations is constructed. It contains the

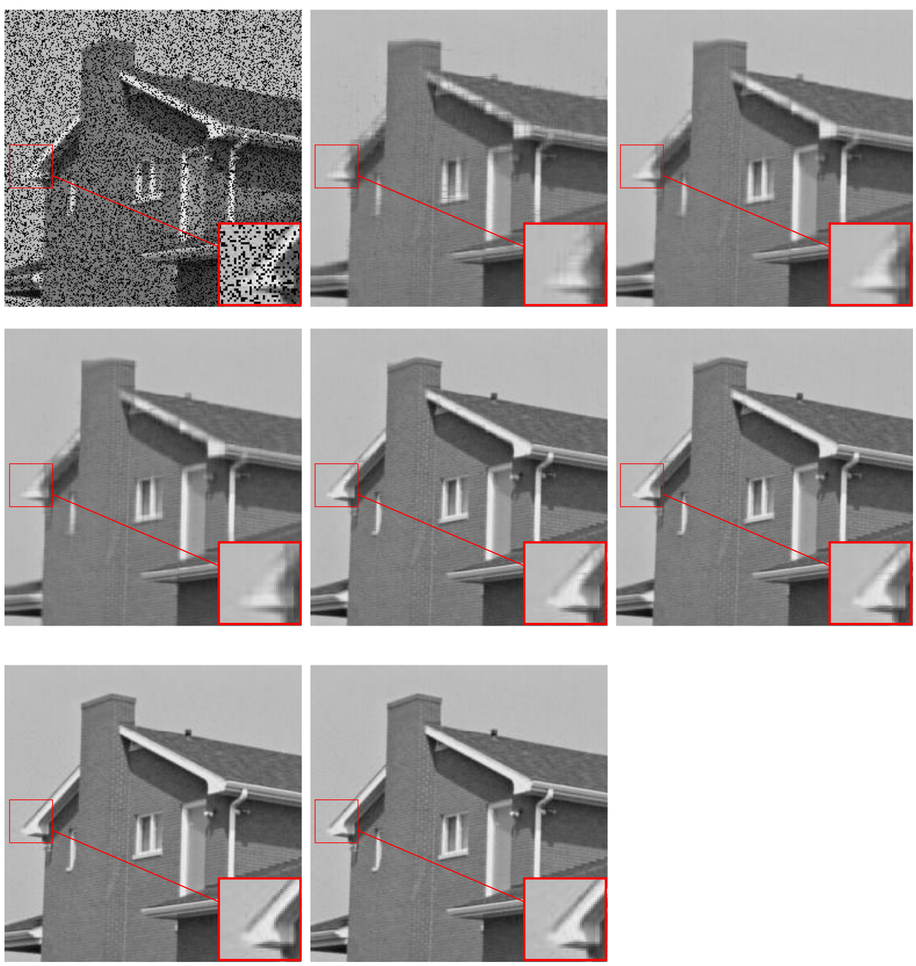

Fig. 5 Results obtained for image "house" using the different methods. From left to right, the order of the images is as follows: simulated image with missing data, recovered image using FPC algorithm, recovered image using ALM algorithm, recovered image using IALM algorithm, recovered image using PPG algoritm, recovered image using PG algorithm, recovered image using SVT with power method, and finally the recovered image using our proposed approach

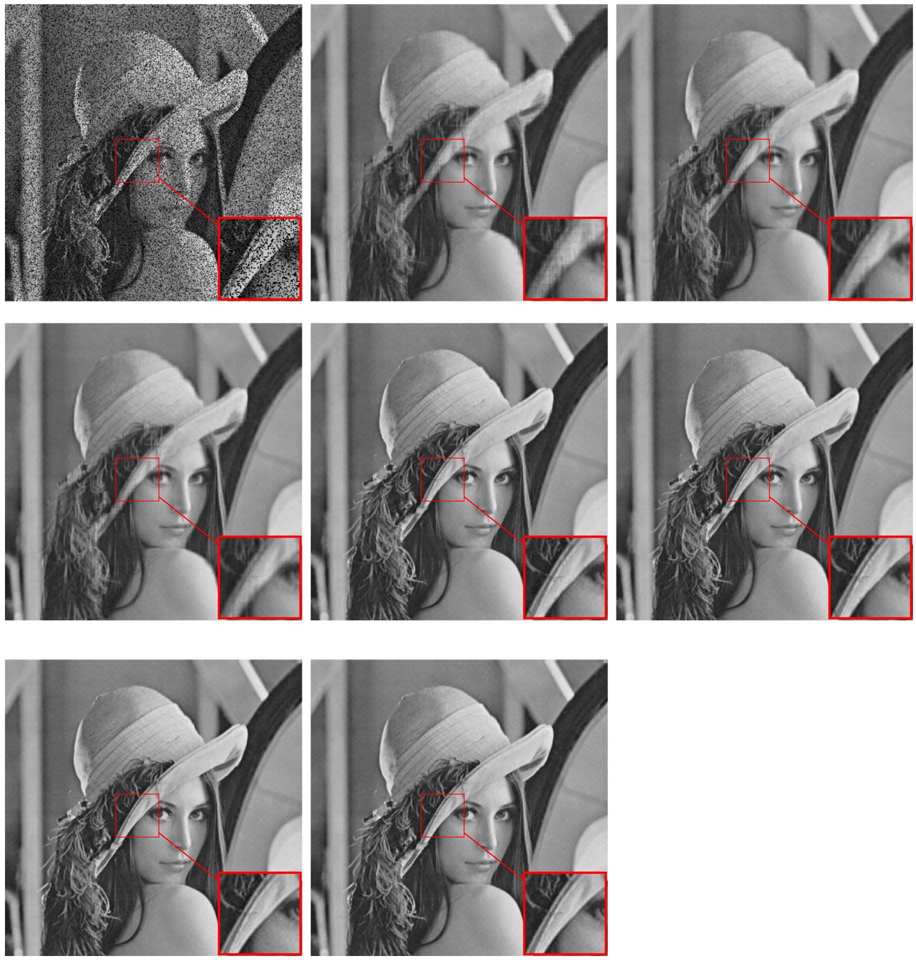

Fig. 6 Results obtained for image "Lena" using the different methods. From left to right, the order of the images is as follows: simulated image with missing data, recovered image using FPC algorithm, recovered image using ALM algorithm, recovered image using IALM algorithm, recovered image using PPG algorithm, recovered image using PG algorithm, recovered image using SVT with power method, and finally the recovered image using our proposed approach

values of pixels that are in the same cluster of the selected missing pixel.

The experimental process is conducted on a benchmark of five standard gray-level images in image processing. The proposed approach is compared visually to different nuclear norm minimization methods for matrix completion and also by measuring the PSNR for different percentages of missing data. Indeed, the proposed approach augments the PSNR of the completion by exploiting the fact that the SVT algorithm is applied per blocks, i.e., on matrices that contain pixels regrouped in the same cluster. A cluster contains eventually pixels that share almost the same characteristics.

Abbreviations

ALM: Augmented Lagrange multiplier; FPCA: Fixed point continuation algorithm; IALM: Inexact augmented Lagrange multiplier; PCA: Principal component analysis; PG: Proximal gradient; PPG: Partial proximal gradient; PSNR: Peak signal-to-noise ratio; SVT: Singular value thresholding; SVD: Singular value decomposition

Acknowledgements

Not applicable.

Funding

Not applicable.

Availability of data and materials

The benchmark used to demonstrate the effectiveness of the proposed approach is composed of five standard images used for image processing. These images are frequently found in literature and available on the following site: http://www.imageprocessingplace.com/root_files_V3/image_databases.htm

Authors' contributions

OB realized the experimental process. For the remaining work, OB, SM, and SR contributed equally to the rest of the work. All authors read and approved the final manuscript.

Authors' information

Oumayma Banouar, first author, received a PhD degree from the Laboratory of Applied Mathematics and Computer Science, Faculty of Science and Techniques, Cadi Ayyad University, Marrakesh, Morocco.
Souad Mohaoui is a PhD student of the Laboratory of Applied Mathematics and Computer Science, Faculty of Science and Techniques, Cadi Ayyad University, Marrakesh, Morocco.

Said Raghay received a PhD degree from the Laboratory of Applied Mathematics and Computer Science, Faculty of Science and Techniques, Cadi Ayyad University, Marrakesh, Morocco.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 17 January 2018 Accepted: 25 October 2018
Published online: 29 November 2018

References

1. O. Banouar, S. Raghay, Novel method for users profiles construction through collaborative filtering. IJCSNS. **17**, 170-176 (2017)
2. G. Koutrika, Y. Ioannidis, Personalizing queries based on networks of composite preferences. ACM Trans. Database Syst. **35**, 1-50 (2010)
3. Fazel, H. Hindi, S.P. Boyd, in *proceedings of the American Control Conference*. A rank minimization heuristic with application to minimum order system approximation (IEEE, Arlington, 2001)
4. J. Cai, J.E. Cands, C. Zuowei, A singular value thresholding algorithm for matrix completion. SIAM J. Optim. **20**, 1956–1982 (2010)
5. K. Toh, M. Todd, R. Tutuncu, SDPT3, a Matlab software package for semidefinite programming. Optim. Methods Softw. **11**, 545–581 (1999)
6. J. Sturm, Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. Optim. Methods Softw. **11**, 625–653 (1999)
7. O. Klopp, Noisy low-rank matrix completion with general sampling distribution. Bernoulli. **20**, 282–303 (2014)
8. J.E. Candès, Y. Plan, in *Proceedings of the IEEE, 98*. Matrix completion with noise, vol. 98, (2010), pp. 925–936. https://doi.org/10.1109/JPROC.2009.2035722
9. J.E. Candès, B. Recht, Exact matrix completion via convex optimization. Found. Comput. Math. **9**, 717-772 (2009)
10. J.E. Candès, X. Li, Y. Ma, J. Wright, Robust principal component analysis. JACM. **58**, 37 (2011). https://doi.org/10.1145/1970392.1970395
11. O. Banouar, S. Raghay, User profile construction for personalized access to multiple data sources through matrix completion method. IJCSNS. **16**, 51-57 (2016)
12. S. Ma, D. Goldfarb, L. Chen, Fixed point and Bregman iterative methods for matrix rank minimization. Optim. Control. **128**, 321-353 (2011)
13. P.L. Combettes, V.R. Wajs, Signal recovery by proximal forward-backward splitting. Multiscale Model. Simul. **4**, 1168–1200 (2005)
14. S. Osher, M. Burger, D. Goldfarb, J. Xu, W. Yin, An iterative regularization method for total variation-based image restoration. Multiscale Model. Simul. **4**, 460–489 (2005)
15. Z. Lin, M. Chen, L. Wu, Y. Ma, The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices (2010). UIUC Technical Report UILU ENG-09-2215 arXiv:1009.5055 [math.OC]
16. Z. Lin, A. Ganesh, J. Wright, L. Wu, M. Chen, Y. Ma, in *Intl. Workshop on Comp. Adv. in Multi-Sensor Adapt. Processing, Aruba, Dutch Antilles*. Fast convex optimization algorithms for exact recovery of a corrupted low-rank matrix. UIUC Technical Report UILU-ENG-09-2214, (2009)
17. Q. Wang, W. Cao, Z. Jin, Two-step proximal gradient algorithm for low-rank matrix completion. Stat. Optim. Inf. Comput. **4**(2), 201-210 (2016)
18. Q. Yao, J.T. Kwok, in *Proc. of the Int. Joint Conf. on Art. Intel*. Accelerated inexact soft impute for fast large scale matrix completion and tensor completion (AAAI Press, Buenos Aires, 2015)
19. Q. Yao, J.T. Kwok, Accelerated inexact soft impute for fast large scale matrix completion and Tensor completion. IEEE Trans. Knowl. Data Eng. (2017). arXiv:1703.05487v2 [cs.NA]
20. N. Halko, P.G. Martinsson, J.A. Tropp, Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions. SIAM. Rev. **53**(2), 217288 (2011)
21. K.C. Toh, S. Yun, An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems. Pac. J. Optim. **6**, 615–640 (2010)
22. E.T. Hale, W. Yin, Y. Zhang, Fixed-point continuation for l1-minimization: methodology and convergence. SIAM J. Optim. **19**, 1107–1130 (2008)