Hybrid Matrix Completion Model for improved Images Recovery and Recommendation Systems

KAI XU1,3, YING ZHANG2, ZHURONG DONG3, ZHANYU LI3 and BOPENG FANG3
1School of Software Engineering, South China University of Technology, Guangzhou, 510006, China
2The Second Affiliated Hospital of Shantou University Medical College, Shantou 515041, China
3School of Transportation Engineering, Shenzhen Polytechnic, Shenzhen 518055, China
Corresponding author: Z. Dong (e-mail: arondong@szpt.edu.cn) and Z. Li (e-mail: autolzy@szpt.edu.cn).

ABSTRACT Matrix completion methods have been widely applied in images recovery and recommendation systems. Most of them are only based on the low-rank characteristics of matrices to predict the missing entries. However, these methods lack consideration of local information. To further improve the performance of matrix completion. In this paper, we propose a novel model based on matrix decompositions and matrix local information. Specifically, we update a number of rank-one matrices, which circumvented the rank estimation in matrix decomposition. And a penalty function is designed to punish singular values without introducing additional parameters. The local information component extracts similar information by an adaptive filter via convolution operation which kernel is obtained by the minimum variance. Finally, we integrate matrix decomposition and local information components via different weights. We apply the proposed method to real-world image datasets and recommendation system datasets. The experimental results demonstrate the proposed model has a lower error and better robustness than several competing matrix completion methods.

INDEX TERMS Matrix completion, images recovery, recommendation systems, adaptive local filtering.

I. INTRODUCTION

Matrix completion is to restore the missing entries in a sparse matrix based on partial observation entries, and has been enjoyed widespread applications in recent years, such as recommendation systems [1], [2], image processing [3]–[5], social networks [6], [7], large scale classification [8] and clustering [9]. In these applications, authors obtain the completed matrix by assuming the observed matrix has a low-rank or approximately low-rank structure. Thus, they use these methods, such as rank minimization (RM), nuclear norm minimization (NNM) and matrix factorization (MF), to predict the missing entries. The NNM is a relaxation to RM, and the MF decomposes a matrix into several factor matrices. Let \( \Omega \) denotes the set of observed indices in an observation matrix, \( \mathcal{P}_\Omega \) is a projection operator onto \( \Omega \), the rank minimization method is

\[
\min_X \text{rank}(X) \quad \text{s.t.} \quad \mathcal{P}_\Omega (X) = \mathcal{P}_\Omega (A). \tag{1}
\]

where \( A \in \mathbb{R}^{m \times n} \) is a sparse observed matrix, \( X \) is a completion matrix, i.e. deserved matrix. \( \text{rank}(X) \) is the rank of \( X \). Model (1) completes the observed matrix by minimizing \( \text{rank}(X) \). However, model (1) was often circumvented as an NNM due to the nonconvexity of rank function [10]. Candès et al. [11] proved that the RM problem can be surrogated by NNM. Thus, model (1) can be rewritten as follows,

\[
\min_X \|X\|_\ast \quad \text{s.t.} \quad \mathcal{P}_\Omega (X) = \mathcal{P}_\Omega (A). \tag{2}
\]

where \( \|X\|_\ast \) is the nuclear norm of \( X \). At present, NNM has achieved satisfactory performance on matrix completion problems. For example, Cai et al. [12] proposed a singular value thresholding algorithm (SVT) that using a soft-thresholding rule to punish singular values, constrain the nuclear norm and limit the rank size. However, the SVT suffers from high computational costs because of singular value decomposition (SVD) in each iteration. To enhance the efficiency of SVD in matrix completion, Oh et al. [13] only compute the partial singular values from a small factored matrix to avoid computation of SVD, which can improve the speed of SVT. The nuclear norm is the \( \ell_1 \)-norm of the singular vector according to the definition. And the nuclear norm treats every singular value as equally important. Therefore, some studies have proposed the weighted nuclear norm minimization (WNNM) [14], [15], which weights the singular...
values and provides a large advantage over the nuclear norm. But WNNM introduces some additional parameters in the regularizer. Nie et al. [16] proposed a new matrix completion method based on non-convex relaxation, using a non-convex function to punish the singular values without additional parameters. Besides, the NNM problems can be surrogated by the truncated nuclear norm minimization [17] and scaled latent nuclear norm minimization [18]. However, these NNM-based methods lack consideration of the local information when matrix completion.

MF is another matrix completion method that treats the desired matrix as a product of multiple factor matrices. For example, the desired matrix \( Y \) is decomposed into \( U \in \mathbb{R}^{m \times r} \) and \( V \in \mathbb{R}^{n \times r} \), satisfying \( Y = UV^T \), where \( r \) is the rank of \( Y \). The optimal rank will be determined by varying \( r \). Besides, \( Y \) can also be decomposed into \( U \in \mathbb{R}^{m \times r} \), \( \Sigma \in \mathbb{R}^{r \times r} \) and \( V \in \mathbb{R}^{n \times r} \). That is, \( Y = U \Sigma V^T \). Where \( \Sigma \) is a diagonal matrix, the values on its diagonal are singular values, the size of \( r \) can be restricted by changing the number of elements on the diagonal of \( \Sigma \). Since the sum of the values on the diagonal of \( \Sigma \) is the nuclear norm, the WNNM can also be weighted to the diagonal of \( \Sigma \). The MF-based methods explore the top \( r \) singular values and require fewer parameters tuning, which is simpler to implement and use.

In the MF, the low-rank matrix fitting algorithm (LMaFit) is one classic matrix decomposition model that estimates the rank by two factor matrices [19]. Inspired by LMaFit method, Tanner et al. [20] proposed the alternating steepest descent algorithm (ASD) which can replace the least square subproblem solution in LMaFit. However, the ASD is a linear method for matrix completion. In order to handle nonlinear structure data, Fan et al. [21] proposed a deep matrix factorization (DFM) which uses a deep-structure neural network to complete a sparse observation matrix. Matrix completion in a non-linear way also includes Riemannian optimization [5], [22]. SVD is often involved in matrix factorization, and the computation of SVD is burdensome work [24]. Yao et al. [25] proposed an accelerated and inexact soft-impute algorithm (AIS) for large-scale matrix and tensor completion. They use the power method [26] to accelerate calculation, and has faster convergence without SVD.

Currently, side information is used in matrix completion because it exploits the internal information of the observation matrix. Xu et al. [27] assume the column and row vectors in sparse matrix \( A \) lie in the subspace spanned by the column and row vectors in \( B_1 \in \mathbb{R}^{m \times r_1} \) and \( B_2 \in \mathbb{R}^{n \times r_2} \), respectively. \( B_1 \) and \( B_2 \) are side information matrices. The side information is introduced into the matrix completion and the completed problem is transformed into solving \( \min_Q \| Q \|_F \), where \( Q \in \mathbb{R}^{m \times n} \). and \( P_0(X) = P_0(B_1QB_2^T) = P_0(A) \). Chiang et al. [28] consider the robustness of matrix completion model, they assumed that the most given side information has noisy, so the residual matrix \( N \) is introduced to satisfy \( A = B_1QB_2^T + N \), where \( N \) can capture the information that side information fails to describe. Besides, other interests integrate the prior knowledge or data attributes to the factorization model to improve the performance of matrix completion [29], [30]. Different from side information, local information learns local characteristics from the perspective of local parts, which extracting information has more explanatory and more specific.

Although the aforementioned methods are successful in many areas, most of them simply ignore local information. Therefore, we propose a novel matrix completion method which based on the low-rank matrix factorization and local information filtering, named LAMC. First, we construct a low-rank matrix factorization model, which completes the observation matrix by iterating the rank-one matrix. The power method is used to speed up and a non-convex function is used to penalize the maximum singular value in the iterative process. Second, we construct an adaptive filter to complete missing entries. The optimal filter kernels are obtained by the minimum variances of the missing entries area. A convolution operation is rented to obtain approximations of missing entries. Finally, we integrate the results of low-rank matrix factorization and the convolution operation to complete the missing entries. Our contributions are summarized as follows:

1) The proposed model combining low-rank matrix decomposition and adaptive local filtering, which can capture more features of the observation matrix. Because the low-rank matrix decomposition and adaptive local filtering are based on the row-column relationship and local information to complete the missing entries, respectively.

2) The low-rank matrix factorization component updates the desired matrix by rank-one matrices. And a non-convex function that punishes the singular values in each iteration. Different from WNNM, the non-convex function was designed does not introduce additional parameters.

3) The convolution operations are executed with different kernel matrices, which cleverly solved the problem of hyperparameter learning (best kernel size), and fully considered the correlation of local information in observation matrices.

4) A heuristic algorithm is used to better blend our components. And experiments verify that the proposed method has a better performance when applied to image recovery and recommendation systems.

The outline of this paper is organized as follows. Section 2 describes the related work; Section 3 first presents the low-rank factorization component and then introduces the convolution operation using adaptive filters, after that, the ensemble module is presented. Experimental results on real-world image datasets and recommendation system datasets are performed in section 4. Section 5 ends with some conclusions and future research lines.

II. RELATED WORK

In this section we will introduce one of the matrix factorization methods, rank-one matrix completion (R1MC), as well as some common filters, which are related to our next work.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/
A. RANK-ONE MATRIX COMPLETION

R1MC is one of matrix factorizations which estimates a matrix into a sum of rank-one or rank-R matrices. The sparse observation matrix $A$ can be completed by the following model:

$$\min_{X, Y} \frac{1}{2} \|X - Y\|_F^2,$$

s. t. $Y = \sum_{r=1}^{R} \sigma_r u_r v_r^T$,

(3)

where $X$ is the completed matrix, $Y$ is the sum of $R$ rank-one matrices. $R$ is the rank with the best matrix completion result. The missing entries are updated by $P_\Omega(Y)$, $\Omega$ is the set outside $\Omega$. $u_r$ and $v_r$ are column vectors. We summarized Algorithm 1 to solve Model (3).

In Algorithm 1, $U$ consists of $R$ $u_r$, subject to $r = 1, 2, ..., R$. $\Sigma$ is a diagonal matrix, $\Sigma = \text{diag}(\sigma_1, \sigma_2, ..., \sigma_R)$ and $V$ consists of $R$ $v_r$, $\|v_r\|_2 = 1$ for $r = 1, ..., R$.

The proposed model is summarized in Fig.1. The completion objects are incomplete images and recommendation systems datasets. The sparse observation matrix $A$ can be obtained from these datasets. Then a low-rank matrix factorization component updates the matrix $Y$ to obtain the one recovery result via $\min_{Y} \|P_\Omega(A) - P_\Omega(Y)\|_F^2$, and $Y$ is a sum of multiple rank-one matrices. The adaptive filtering component obtains the best filter for each missing entry, the number of the missing entries is $|\Omega|$, so we need $|\Omega|$ filters. After that, the convolution operation is performed on the best filters and associated missing entry areas to obtain $Z$. The final completion matrix $X$ will be obtained by weighting $Z$ and $Y$. In application, we can directly use $X$ to display the recovered images or sort the rows of $X$ to obtain the recommendation list. The details are shown in the subsequent sections.

B. COMMON FILTERS

Filters are usually used in image processing for noise suppression, smoothing and filtering [33]. Commonly filters include: mean filter, Gaussian filter and median filter. The mean filter performs average smoothing on an image (matrix or tensor), assuming that the relative importance of local information is consistent. This is unreasonable because in the real world, the size of areas with locally relevant information is not fixed. In addition, the filter matrix size is a hyperparameter, and adjustment of hyperparameters is not easy in machine learning. However, the adaptive filters can handle this problem well.

Matrix completion can be regarded as a signal processing of observation entries to obtain the missing entries. Therefore, filtering the observation entries can complete a sparse matrix. Unlike filters that fill in missing entries from the perspective of local information, low-rank matrix completion predicts missing entries from the low-rank characteristics. Therefore, combining two components can help to achieve a better result.

III. PROPOSED METHOD

A. MODEL ARCHITECTURE

The proposed model is based on two perspectives. (i) Some sparse matrices satisfy the low-rank characteristics, and the missing entries can be predicted by the matrix factorization. (ii) There is a certain relationship between the local entries in the sparse matrix, and the missing entries can be completed by the convolution operation. Therefore, we construct an ensemble model to predict the missing entries.

The proposed model is summarized in Fig.1. The completion objects are incomplete images and recommendation systems datasets. The sparse observation matrix $A$ can be obtained from these datasets. Then a low-rank matrix factorization component updates the matrix $Y$ to obtain the one recovery result via $\min_{Y} \|P_\Omega(A) - P_\Omega(Y)\|_F^2$, and $Y$ is a sum of multiple rank-one matrices. The adaptive filtering component obtains the best filter for each missing entry, the number of the missing entries is $|\Omega|$, so we need $|\Omega|$ filters. After that, the convolution operation is performed on the best filters and associated missing entry areas to obtain $Z$. The final completion matrix $X$ will be obtained by weighting $Z$ and $Y$. In application, we can directly use $X$ to display the recovered images or sort the rows of $X$ to obtain the recommendation list. The details are shown in the subsequent sections.

B. RANK-ONE MATRIX COMPLETION

According to the SVD, any matrix can be represented as a sum of a number of rank-one matrices and its singular vectors are orthogonal. However, a matrix can also be the sum of more rank-one matrices if the singular vectors are not enforced to be orthogonal. So we construct a new matrix completion model based on R1MC as follows,
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3125152, IEEE Access

Author Name: Preparation of Papers for IEEE Access (February 2021)

Figure 1. The overall architecture of the proposed LAMC.

Algorithm 2: Low-rank matrix completion

1: input: incomplete matrix A, set Ω, and tolerance ε;
2: initialization: Y₀ = 0, current iteration i = 0, maximum iteration N;
3: while i < N do
4:    σ₀, p₀, q₀ = the largest singular value and leading left and right singular vectors of \( P(Ω) - P(Ω) \);
5:    \( F(σ) \), 0 ≤ σ ≤ σ_{i} ≤ σ;
6:    \( Y_{i+1} = Y_i + α_i p_i q_i^T \);
7:    if \( \|P(Ω) - P(Ω(Y_{i+1}))\|/\|P(Ω)\| < ε \) or i > N then break;
8: end while
9: output: \( Y_{i+1} \);

In step 4 of Algorithm 2, The power method is used to calculate \( σ_0, p_0 \) and \( q_0 \), because the largest singular value and corresponding leading left and right singular vectors can be computed efficiently by the power method [26]. We chose of largest singular value to update \( Y_i \) because A consists of \( P(Ω)Σ_{i=1}^N F(σ_i)p_i q_i^T \), and we can optimize it step by step. The function \( F(σ) \) in step 5 of algorithm 2 is a penalty function that is vital for solving mode (4) because it governs the low-rank constraint or suppresses singular values. The traditional methods are to treat \( F(σ) \) as a soft or hard threshold operation [34]. These approaches extremely suppress small singular values because they penalize smaller singular values to zero. However, considering that large singular values usually represent the signal itself, small singular values are likely to be composed of noise, therefore, (i) large singular values should be treated similarly, while (ii) small singular values should be suppressed excessively. In addition, \( 0 ≤ F(σ_i) ≤ σ_i \). Therefore, we designed the following parameterless penalty function to suppress singular values:

\[ F(σ_i) = \log(σ_i + 1). \]

C. PROOF ITERATIVE CONVERGENCE OF LOW RANK MATRIX FACTORIZATION

To verify the convergence of Algorithm 2 when solving model (4), the convergence analysis is given below. We transform the proof of iterative convergence into the following question:

\[ \|P(Ω) - P(Ω(Y_i))\|^2 < \|P(Ω) - P(Ω(Y_{i-1}))\|^2. \]

\[ C. PROOF ITERATIVE CONVERGENCE OF LOW RANK MATRIX FACTORIZATION \]

To verify the convergence of Algorithm 2 when solving model (4), the convergence analysis is given below. We transform the proof of iterative convergence into the following question:

\[ \|P(Ω) - P(Ω(Y_i))\|^2 < \|P(Ω) - P(Ω(Y_{i-1}))\|^2. \]
Proof: For algorithm 2, let we the SVD of $P_{\Omega}(A) - P_{\Omega}(Y_i) = \sum_{i=1}^{r} \sigma_i u_i v_i^T$, where $r$ is the rank of $P_{\Omega}(A) - P_{\Omega}(Y_i)$, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r \geq 0 > \sigma_r$. Then,

$$
\|P_{\Omega}(A) - P_{\Omega}(Y_i)\|^2_F = \|\sum_{i=1}^{r} \sigma_i u_i v_i^T\|^2_F = \sum_{i=1}^{r} \sigma_i^2.
$$

In step 4 of algorithm 2, we select the maximum singular value and leading left and right singular vectors to update $Y_{i+1} = Y_i + \alpha_i p_i q_i^T$. Thus, we obtain

$$
\|P_{\Omega}(A) - P_{\Omega}(Y_{i+1})\|^2_F = \|P_{\Omega}(A) - P_{\Omega}(Y_i) - P_{\Omega}(\alpha_i p_i q_i^T)\|^2_F \\
= \|P_{\Omega}(A) - P_{\Omega}(Y_i)\|^2_F - \|P_{\Omega}(\alpha_i p_i q_i^T)\|^2_F \\
\leq \|P_{\Omega}(A) - P_{\Omega}(Y_i)\|^2_F - \alpha_i^2 \|\sigma_i u_i v_i^T\|^2_F.
$$

According to the step 6 of algorithm 2, we have $\alpha_{i+1} = F(\sigma_i)$, $p_{i+1} = u_i$ and $q_{i+1} = v_i$. The projection operation $P_{\Omega}$ is equal to 0, thus $\|P_{\Omega}(A) - Y_i\|^2 = A - Y_i$. Then formula (8) can be rewritten into the following formula:

$$
\|\sum_{i=1}^{r} \sigma_i u_i v_i^T - \alpha_i p_i q_i^T\|^2_F = \|\sum_{i=1}^{r} \sigma_i u_i v_i^T - (\sigma_i - \log(\sigma_i + 1)) u_i v_i^T\|^2_F \\
= \|\sum_{i=1}^{r} \sigma_i u_i v_i^T + \log(\sigma_i + 1) u_i v_i^T\|^2_F \\
\leq \|\sum_{i=2}^{r} \sigma_i u_i v_i^T\|^2_F + \|\log(\sigma_i + 1) u_i v_i^T\|^2_F \\
= \sum_{i=2}^{r} \sigma_i^2 + \log(\sigma_i + 1)^2.
$$

Obviously $\log(\sigma_i + 1)^2 < \sigma_i^2$, thus $\|P_{\Omega}(A) - P_{\Omega}(Y_i)\|^2_F < \|P_{\Omega}(A) - P_{\Omega}(Y_{i-1})\|^2_F$. □

### D. ADAPTIVE LOCAL FILTERING MODULE

In this section, we introduce the convolution operation to extract local features and how to construct an adaptive filter.

For the sparse observation matrix $A$, the kernel matrix is $H \in \mathbb{R}^{(2k+1) \times (2k+1)}$ where $k$ is a positive integer and $2k + 1 \leq \min(m + \text{padding}, n + \text{padding})$, where padding is the extended size that makes one missing entry at the center of the filter. The value of padding is usually taken as missing, i.e. 0. The missing entry $A_{ij} = 0$ can be completed to obtain $Z_{ij}$. The mean and Gaussian filter formula for obtain $Z_{ij}$ is

$$
Z_{ij} = \text{sum}(H_{ij} \circ A(i - k; i + k, j - k; j + k)).
$$

Where $H_{ij}$ is the best filter for the missing entry $A_{ij}$, $\text{sum}(\cdot)$ is a function that sums all the entries in a matrix, $A'(a; b; c; d)$ is to select the area where the abscissa is $a$ to $b$, and the ordinate is $c$ to $d$. $A'$ is the padding matrix of $A$. The value in the kernel matrix $H_{ij}$ is set to $\frac{1}{\sum_{(i-k; i+k,j-k;j+k)\in\Omega}}$ in the mean filter, where $\Omega$ is the indicator function. For Gaussian filter, $H_{ij}$ is normalized according to the Gaussian distribution. However, for the median filter, the values of $H_{ij}$ are all set to 1, and then the Hadamard product is performed with $A(i - k; i + k, j - k; j + k)$ and the median $\text{median}(\cdot)$ is a function that selects the median in a sparse matrix, the formula is as follows.

$$
Z_{ij} = \text{median}(H_{ij} \circ A'(i - k; i + k, j - k; j + k)).
$$

In the next, we will obtain the best filter kernel size for all missing entries. Traditional filtering methods often set a fixed-size kernel matrix [35], [36]. However, the fixed-size kernel has some shortcomings. (i) It does not take into account local information consistency. Because the fixed-size kernel sees all areas equally. (ii) It requires learning hyperparameters. The kernel size of the filter $(2k+1)$ is a hyperparameter that requires expert experience to set. This is a tedious job and does not guarantee absolute reliability.

For formulas (10) and (11), it can be known that the essence of filtering is to obtain new values based on the local correlations in an observation matrix. To determine the relevance of a specific region in it, we calculate the variance of different ranges in each region, and the most relevant area corresponds to the smallest variance. Then we obtain

$$
k_{ij} = \min_k \sum_{i-k}^{i+k} \sum_{j-k}^{j+k} \frac{(A(i,u,j,v) - E)^2}{\sum_{(i+k,j+k,k; j+k)\in\Omega}}
$$

such that $A(i,j) = 0$. Where $E$ is the mean of $A'(i-k; i+k, j-k; j+k)$. The value of formula (12) can reflect the correlation degree of region in $A'(i-k; i+k, j-k; j+k)$. Fig.2(a) randomly removes 50% of the pixels and calculates the variance corresponding to different $k_{ij}$ (see Fig.2(b)) while $A(i,j) = 0$. The smallest $k_{ij}$ can be determined when $k = 1, 2, ..., \min(\frac{m}{2}, \frac{n}{2})$. That is, since $A$ has a certain dimension, we can definitely find a minimum variance corresponding to $k_{ij}$.

![A](image)

**FIGURE 2.** The variance of different kernel matrices with the center point $A(i,j) = 0$. The size of the kernel matrix is $2k+1$, so the variance is the smallest when the kernel is $7$.

### E. Ensemble learning module

In the above part, we obtain two completed matrices $Y$ and $Z$ by low-rank matrix factorization and adaptive local filtering. We hope that the completed matrix can be optimal as a whole, so we circumvent this ensemble problem by parameter learning and define the optimal weights $w_r$ and $w_z$ for $Y$ and $Z$. The optimal weights are tuned through the differential evolution (DE) algorithm [37] because the DE algorithm has the advantage of quickly approaching the global optimal solution. DE algorithm includes three basic steps: mutation, crossover, and selection. The individual differences in the population are large when the algorithm starts to iterate, and this mutation operation will make the algorithm have strong global search ability; the individual differences in the population are small, the algorithm to have a strong local search ability. The main advantages of DE are: (i). fewer parameters to be tuned, (ii). Not easy to fall into the local optimum, (iii). And faster convergence speed. The optimal weight vector is $w = (w_r, w_z)$. Algorithm 3 shows the
process of selecting the optimal weights. The fitness function is defined as the smallest root square error.

**Algorithm 3** Select the optimal weight through the DM

1: input: matrix $X$, $Y$
2: initialization: population size $NP$, mutation factor $F$, crossover probability $CR$, maximum number of functions evaluations $G$, current number of iteration $g = 0$
3: GENERATE random weight matrix $w_i$, $i = 1, 2, ..., NP$
4: while $g < G$ do
5: MUTATION, generate mutation weight matrix $v_i = w_i + F(w_{i2} - w_{i3}), i \neq r1 \neq r2 \neq r3, r1, r2, r3 \in \{1, NP\}$
6: CROSSOVER, The mutation weight matrix and the original weight matrix are crossed to produce the trial weight: $u_{ij} = \begin{cases} v_{ij}, & \text{random(0.1)} \leq CR \\ w_{ij}, & \text{otherwise} \end{cases}$, where $j$ is the index of the $j$th weight, $j = 1, 2$
7: Selection, choose $u_i$ and $v_i$ with the smallest fitness value as the new weight $w_i$, and $g = g+1$
8: output: $w$ is the weight corresponding to the minimum fitness function value of $w_i$

IV. DISCUSSION

In this part, we use some real-world datasets to assess the proposed model. The datasets are described in each subsection. We compare with some matrix completion methods, including AIS [25], DMF [21], LDMM [22], ASD [20], LMaFit [19] and SVT [12]. All experiments were performed on a PC with Intel Xeon(R) Bronze 1.70 GHz CPU and 32 GB memory. The software environment is MATLAB R2017a. The performance indicators of experiments include Root Mean Square Error (RMSE) and Mean Absolute Error (MAE).

$$RMSE = \frac{1}{|\Omega_T|} \sum_{(i,j) \in \Omega_T} (A_{ij} - X_{ij})^2,$$

$$MAE = \frac{1}{|\Omega_T|} \sum_{(i,j) \in \Omega_T} |A_{ij} - X_{ij}|,$$

where $\Omega_T$ is the set of testing data set indices in sparse matrices. $|\Omega_T|$ denotes the number of entries in a testing set. All datasets are standardized so that the results of the low-rank matrix factorization component and the adaptive filtering component are in the same dimension.

A. GRAYSCALE IMAGES RECOVERY

In this part, we obtain three grayscale images from references [16] and [32], to compare the performance of our algorithm with other algorithms. The original images (see Fig.3) are named Lenna (512×512 pixels), Rice (600×960 pixels), and Male (1,024×1,024 pixels). 50% of each image data matrix entries are randomly sampled as the training set, another 5% as the validation set for tuning parameters, and the rest as the resting set.

In our method, we apply the traditional filters to our adaptive filtering component. Table I shows the results of various filters applied to grayscale images. We can conclude from Table I: The adaptive mean filter has the best recovery performance for grayscale images. The adaptive median filter is the second most effective. The adaptive Gaussian filter has the worst effect on image recovery.

Therefore, a combination of adaptive mean filter and low-rank matrix completion is selected to restore the missing entries in images. Table II shows the comparison between our proposed method and other matrix completion methods. According to table II, the LAMC algorithm performs best on each image data set (bold in the table). The overall performance of the AIS is second only to our method. The recovery performance of LMaFit and SVT are slightly similar, but the SVT is better than the LMaFit in general, for example, SVT performs better on the Rice and Lenna datasets. The performance of LDMM is unsatisfactory, especially on the Rice data set, which results are poor due to the limitation of the validation set. ASD has a poor performance on the three image datasets, and the error on the Male data set is larger than other algorithms. This group of experiments shows that our method has the best effect on grayscale image recovery.

**TABLE I**

| Filters    | Lenna | Rice | Male |
|------------|-------|------|------|
| RMSE       | MAE   | RMSE | MAE   | RMSE  | MAE  |
| Gaussian   | 0.292 | 0.137| 0.351| 0.186 | 0.318| 0.157|
| Median     | 0.195 | 0.103| 0.302| 0.170 | 0.213| 0.116|
| Mean       | 0.184 | 0.098| 0.293| 0.165 | 0.204| 0.112|

**TABLE II**

| Methods | Lenna | Rice | Male |
|---------|-------|------|------|
| RMSE    | MAE   | RMSE | MAE  | RMSE           | MAE           |
| AIS     | 0.238 | 0.157| 0.296| 0.191          | 0.338         | 0.168         |
| DMF     | 0.428 | 0.294| 0.375| 0.263          | 0.469         | 0.343         |
| LDMM    | 0.433 | 0.271| 1.892| 0.722          | 0.452         | 0.309         |
| ASD     | 0.489 | 0.365| 0.459| 0.337          | 0.731         | 0.570         |
| LMaFit  | 0.280 | 0.186| 0.342| 0.234          | 0.273         | 0.195         |
| SVT     | 0.260 | 0.174| 0.321| 0.220          | 0.298         | 0.212         |
| LAMC    | 0.180 | 0.098| 0.2721| 0.158          | 0.196         | 0.112         |

In order to verify the performance of different scale validation sets on the experiments, we keep the proportion of the training sets unchanged, set the proportion of the verification sets to 25%, and the test sets to 25%. Table III shows the recovery of the images through adaptive mean filtering and low-rank matrix completion. From the experimental results, it can be seen that the RMSE and MAE performance of LAMC are the best. The AIS algorithm has the second best performance in terms of image recovery, with errors that are only higher overall than those of our algorithm.

**FIGURE 3.** Original Grayscale images.
The SVT can also achieve better results on grayscale image recovery, although it is inferior to the ALS algorithm. The ASD has poor results on this proportion of datasets. The results obtained by the LDMM on the Rice image are more reliable due to the proportion increase of the validation sets. This experiment shows that our proposed algorithm has strong robustness and better performance.

### Table III

| Methods | Lenna | Rice | Male |
|---------|-------|------|------|
|         | RMSE  | MAE  | RMSE | MAE  | RMSE | MAE  |
| AIS     | 0.235 | 0.156| 0.295| 0.191| 0.236| 0.168|
| DMF     | 0.425 | 0.294| 0.375| 0.262| 0.469| 0.344|
| LDMM    | 0.423 | 0.267| 0.420| 0.306| 0.425| 0.301|
| ASD     | 0.567 | 0.408| 0.498| 0.373| 0.648| 0.494|
| LMaFit  | 0.278 | 0.187| 0.344| 0.234| 0.263| 0.199|
| SVT     | 0.261 | 0.173| 0.310| 0.208| 0.275| 0.203|
| LAMC    | 0.179 | 0.098| 0.273| 0.159| 0.196| 0.112|

To fully demonstrate the completion results on images, we choose image datasets with a verification set ratio of 25% to show the recovery effect. Fig.4 shows the original sparse images, and Fig.5 shows the recovery images with different algorithms. It can be seen from Fig.5 that the LAMC has the best image restoration performance visually. The restored images obtained by the ASD are almost indistinguishable, and its restoration performance is the worst. The AIS and SVT can restore images better, and the restoration performance is only inferior to our proposed method from the visual comparison. The visual display of the image restoration matches with the RMSE of Table III.

### Figure 4. Sparse observation Grayscale images

### Figure 5. Visual comparison for different Grayscale images

**B. RECOMMENDATION SYSTEMS**

The low-rank matrix completion has been widely used in recommendation systems. However, the recommendation systems based on the matrix completion only consider the low-rank characteristic (i.e. the rows or columns of sparse observation matrix), in order to further improve the recommendation efficiency. We use common datasets, MovieLens 100K, MovieLens 1M [38] and Book-Crossing [39]. The MovieLens 100K contains 100,000 ratings issued by 943 users for 1,682 movies. The MovieLens 1M dataset consists of 1,000,209 ratings from 6,040 users on 3,703 movies. The Book-Crossing contains 13,123 users and 7,774 books, for a total of 1,048,574 user-book ratings.

We preprocess the datasets of the recommendation systems before conducting experiments. Because the random construction of the user-item rating matrices did not consider the local relations. Therefore, we defined a processing rule to put the most similar rows and the most similar columns.
together, that is, the upper row and the next row (the upper column and the next column) of each row (column) are the most similar rows (columns). The matrix constructed in this way helps to capture some local information and does not affect the row-column relations. All experiments select 60% of the rating data as the training sets, the proportion of the validation sets include 5% and 25% respectively, and the remaining parts are the test sets.

Table IV shows the experimental results of different adaptive filter methods on the recommendation systems. It can be seen that the adaptive Gaussian filter performs best on the three recommendation system datasets, especially on the Book-Crossing data set. Furthermore, the adaptive Gaussian filter has the best RMSE on MovieLens 100K and Book-Crossing. However, the adaptive median filter has the best MAE on MovieLens 1M. The adaptive mean filter performs slightly inferior to the Gaussian filter on the MovieLens datasets, but the performance on the Book-Crossing dataset is not satisfactory. Experiments show that the adaptive Gaussian filter is more efficient for MovieLens datasets.

| Filters   | MovieLens 100K | MovieLens 1M | Book-Crossing |
|-----------|---------------|--------------|---------------|
| Gaussian | 0.985         | 0.716        | 0.921         |
| Median    | 1.001         | 0.728        | 0.927         |
| Mean      | 0.985         | 0.730        | 0.921         |

Table V shows the experimental results of adaptive Gaussian filter and low-rank matrix completion on the recommendation system with 5% validation sets. The RMSE and MAE of the proposed LAMC algorithm are smaller than other methods in general, especially for MovieLens 1M and Book-Crossing datasets. The SVT algorithm performs well on the MovieLens 100K, but it did perform well on MovieLens 1M. AIS and DMF perform better on the recommendation systems, and DMF has the best MAE on the MovieLens 1M. The LDMM cannot be performed on the Book-Crossing data set due to computer memory. ASD does not converge well on the Book-Crossing data set.

| Methods  | MovieLens 100K | MovieLens 1M | Book-Crossing |
|----------|----------------|--------------|---------------|
| AIS      | 0.889          | 0.713        | 0.823         |
| DMF      | 0.921          | 0.711        | 0.823         |
| LDMM     | 0.909          | 0.721        | 0.902         |
| ASD      | 0.892          | 0.716        | 0.791         |
| LMaFit   | 0.991          | 0.821        | 0.972         |
| SVT      | 0.866          | 0.683        | 0.928         |
| LAMC     | 0.885          | 0.680        | 0.782         |

We integrate the adaptive Gaussian filter and low-rank matrix factorization to 25% of validation sets. Table VI displays the experimental results of the hybrid. The LAMC has the best RMSE on all recommendation system datasets, and its MAE is better in general. The performance of the DMF is second only to our method, and the MAE of DMF is the best on MovieLens 1M. The performance of the AIS is relatively stable, and will not cause large deviations with different proportions of validation sets. It demonstrates that AIS has better adaptability just like our method. The LDMM cannot be run on the Book-Crossing dataset due to computer memory. ASD does not converge well on the Book-Crossing data set. Experiments show the superiority of our method.

| Methods  | MovieLens 100K | MovieLens 1M | Book-Crossing |
|----------|----------------|--------------|---------------|
| AIS      | 0.886          | 0.702        | 0.824         |
| DMF      | 0.857          | 0.669        | 0.787         |
| LDMM     | 0.905          | 0.719        | 0.922         |
| ASD      | 0.959          | 0.727        | 0.857         |
| LMaFit   | 0.993          | 0.822        | 0.973         |
| SVT      | 0.875          | 0.692        | 0.870         |
| LAMC     | 0.814          | 0.648        | 0.783         |

To further verify the completion preference of our method, we show the RMSE versus time obtained from the low-rank matrix factorization. The MovieLens and Book-Crossing datasets with validation sets ratio of 25% are selected for this experiment. The experimental results are shown in Fig.6. It can be seen from fig.6 that the LAMC has better convergence and lower RMSE. The AIS has the fastest convergence on MovieLens 100K and Book-crossing datasets, but its RMSE is not better than ours. The DMF has the same performance as our method when we only perform matrix low-rank factorization, especially in the Book-Crossing data set, and its convergence speed is similar to ours. Because the LDMM cannot run on our machine on the Book-Crossing data set. Therefore, Fig.6(c) does not show the performance of the LDMM. The RMSE of the LMaFit fluctuates on the MovieLens 1M and Book-Crossing datasets, and fluctuates more obviously on the MovieLens 1M dataset. The ASD cannot achieve a good result within the specified time under the Book-Crossing data set, and its RMSE is more than 2.0. These experimental results show that our proposed method keeps a better convergence and lower RMSE without adaptive filtering.

| Methods  | MovieLens 100K | MovieLens 1M | Book-Crossing |
|----------|----------------|--------------|---------------|
| AIS      | 0.886          | 0.702        | 0.824         |
| DMF      | 0.857          | 0.669        | 0.787         |
| LDMM     | 0.905          | 0.719        | 0.922         |
| ASD      | 0.959          | 0.727        | 0.857         |
| LMaFit   | 0.993          | 0.822        | 0.973         |
| SVT      | 0.875          | 0.692        | 0.870         |
| LAMC     | 0.814          | 0.648        | 0.783         |
In this article, we propose a new matrix completion method that is a hybrid of low-rank matrix factorization and adaptive filtering. In the component of low-rank matrix factorization, the desired matrix is decomposed as several rank-one matrices and then solved iteratively. The non-convex function of adaptive filtering which borrowing the idea of convolution to extract local information. We design a method to obtain the best kernel size for each missing entry. A heuristic algorithm is used to integrate the two components. We use real-world image datasets and recommendation system datasets to perform the experiments. And the results demonstrate that the proposed method is more accurate, much faster and more robust than the competing methods.

Since tensor is a high-dimensional matrix, we are interested in using an ensemble idea to high-dimensional tensor in future work. The tensor has a regular internal structure, and how to complete it without destroying the structure is also a matter we need to study. Besides, exploring the performance of different tensor norms to apply to high-dimensional, for example, color images and videos, etc., is one of our future works.

Acknowledgments: Ying Zhang is the co-first author.

**REFERENCES**

[1] X. Guan, C. T. Li, and Y. Guan, “Matrix factorization with rating completion: An enhanced SVD model for collaborative filtering recommender systems,” IEEE Access, vol. 5, pp. 27668–27678, 2017, doi: 10.1109/ACCESS.2017.2772226.

[2] Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” Computer, vol. 42, no. 8, pp. 30–37, 2009, doi: 10.1109/MC.2009.263.

[3] Z. Jia, M. K. Ng, and G. J. Song, “Robust quaternion matrix completion with applications to image inpainting,” Numerical Linear Algebra with Applications, vol. 26, no. 4, p. Article e2245:1-35, 2019, doi: 10.1002/nla.2245.

[4] K. Xu, Y. Zhang and Z. Xiong, “Iterative rank-one matrix completion via singular value decomposition and nuclear norm regularization,” Information Sciences, vol. 578, pp. 574-591, 2021, doi: 10.1016/j.ins.2021.07.035.

[5] S. Osher, Z. Shi, and W. Zhu, “Low dimensional manifold model for image processing,” SIAM Journal on Imaging Sciences, vol. 10, no. 4, pp. 1669–1690, 2017, doi: 10.1137/16M1058868.

[6] Z. Zhao, L. Zhang, X. He, and W. Ng, “Expert finding for question answering via graph regularized matrix completion,” IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 4, pp. 993–1004, 2015, doi: 10.1109/TKDE.2014.2356461.

[7] T. Wu Zhang, W. Ping Li, L. Wang, and J. Yang, “Social recommendation algorithm based on stochastic gradient matrix decomposition in social network,” Journal of Ambient Intelligence and Humanized Computing, vol. 11, pp. 601–608, 2020, doi: 10.1007/s12652-018-1167-7.

[8] J. Fan, Z. Tian, M. Zhao, and T. W. S. Chow, “Accelerated low-rank representation for subspace clustering and semi-supervised classification on large-scale data,” Neural Networks, vol. 100, pp. 39–48, 2018, doi: 10.1016/j.neunet.2018.01.014.

[9] J. Fan and T. W. S. Chow, “Sparse subspace clustering for data with missing entries and high-rank matrix completion,” Neural Networks, vol. 93, pp. 36–44, 2017, doi: 10.1016/j.neunet.2017.04.005.

[10] N. J. A. Harvey, D. R. Karger, and S. Yekhanin, “The complexity of matrix completion,” in Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, 2006, pp. 1103–1111, doi: 10.1145/1109557.1109679.

[11] E. Candès and B. Recht, “Exact matrix completion via convex optimization,” Association for Computing Machinery, vol. 55, no. 6, pp. 111–119, 2012, doi: 10.1145/2184319.2184343.

[12] J. F. Cai, E. J. Candès, and Z. Shen, “A singular value thresholding algorithm for matrix completion,” SIAM Journal on Optimization, vol. 20, no. 4, pp. 1956–1982, 2010, doi: 10.1137/080734970.

[13] T.-H. Oh, Y. Matsushita, Y. W. Tai, and I. S. Kweon, “Fast randomized singular value thresholding for nuclear norm minimization,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 4484–4493, doi: 10.1109/CVPR.2015.7299078.

[14] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang, “Weighted nuclear norm minimization and its applications to low level vision,”

---

**FIGURE 6.** Testing RMSE vs time (in seconds) on the MovieLens datasets and Book-Crossing data set.

**V. CONCLUSIONS**

In this article, we propose a new matrix completion method that is a hybrid of low-rank matrix factorization and adaptive filtering. In the component of low-rank matrix factorization, the desired matrix is decomposed as several rank-one matrices and then solved iteratively. The non-convex function and power method are used to punish singular values, reduce parameters and speed up computation. The convergence of the proposed low-rank matrix factorization is strictly proved mathematically. The second component of adaptive filtering which borrowing the idea of convolution to extract local information. We design a method to obtain the best kernel size for each missing entry. A heuristic algorithm is used to
International Journal of Computer Vision, vol. 121, no. 2, pp. 183–208, 2017, doi: 10.1007/s11263-016-0930-5.
[15] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang, “Weighted nuclear norm minimization and its applications to low level vision,” International Journal of Computer Vision, vol. 121, no. 2, pp. 183–208, 2017, doi: 10.1007/s11263-016-0930-5.
[16] F. Nie, Z. Hu, and X. Li, “Matrix completion based on non-convex low-rank approximation,” IEEE Transactions on Image Processing, vol. 28, no. 5, pp. 2378–2388, 2019, doi: 10.1109/TIP.2018.2886712.
[17] X. Su, Y. Wang, X. Kang, R. Tao, and S. Member, “Nonconvex truncated nuclear norm minimization based on adaptive bisection method,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 11, pp. 3159–3172, 2019, doi: 10.1109/TCSVT.2018.2878803.
[18] Q. Yao and J. T. Kwok, “Accelerated and inexact soft-impute for large-scale matrix and tensor completion,” IEEE Transactions on Knowledge and Data Engineering, vol. 31, no. 9, pp. 1665–1679, 2019, doi: 10.1109/TKDE.2018.2867553.
[19] Z. Wen, W. Yin, and Y. Zhang, “Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm,” Mathematical Programming Computation, vol. 4, pp. 333–361, 2012, doi: 10.1007/s12532-012-0044-1.
[20] J. Tanner and K. Wei, “Low rank matrix completion by alternating steepest descent methods,” Applied and Computational Harmonic Analysis, vol. 40, no. 2, pp. 417–429, 2016, doi: 10.1016/j.acha.2015.08.003.
[21] J. Fan and J. Cheng, “Matrix completion by deep matrix factorization,” Neural Networks, vol. 98, pp. 34–41, 2018, doi: 10.1016/j.neunet.2017.10.007.
[22] S. Oshery, Z. Shiz, and W. Zhuy, “Low dimensional manifold model for image processing,” SIAM Journal on Imaging Sciences, vol. 10, no. 4, pp. 1669–1690, 2017, doi: 10.1137/16M1058680.
[23] B. Vandereycken, “Low-rank matrix completion by Riemannian optimization,” SIAM Journal on Optimization, vol. 23, no. 2, pp. 1214–1236, 2013, doi: 10.1137/110845768.
[24] R. Mazumder, T. Hastie, and R. Tibshirani, “Spectral regularization algorithms for learning large incomplete matrices,” Journal of Machine Learning Research, vol. 11, pp. 2287–2322, 2010, doi: 10.1016/j.cortex.2009.08.003.Predictive.
[25] Q. Yao and J. T. Kwok, “Accelerated and inexact soft-impute for large-scale matrix and tensor completion,” IEEE Transactions on Knowledge and Data Engineering, vol. 31, no. 9, pp. 1665–1679, 2019, doi: 10.1109/TKDE.2018.2876533.
[26] N. Halko, P. G. Martinsson, and J. A. Tropp, “Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions,” SIAM REVIEW, vol. 53, no. 2, pp. 217–288, 2011, doi: 10.1137/0907711806.
[27] M. Xu, R. Jin, and Z. H. Zhou, “Speedup matrix completion with side information: Application to multi-label learning,” in Proceedings of the 26th International Conference on Neural Information Processing Systems, 2013, pp. 2301–2309, doi: 10.5555/2999792.2999869.
[28] K. Y. Chiang, C. J. Hsieh, and I. S. Dhillon, “Matrix completion with noisy side information,” in Proceedings of the 28th International Conference on Neural Information Processing Systems, 2015, vol. 2, pp. 3447–3455, doi: 10.5555/2969442.2969624.
[29] L. Huang et al., “Matrix completion with side information and its applications in predicting the antigenicity of influenza viruses,” Bioinformatics, vol. 33, no. 20, pp. 3195–3201, 2017, doi: 10.1093/bioinformatics/btx390.
[30] W. Fithian and R. Mazumder, “Flexible low-rank statistical modeling with missing data and side information,” Statistical Science, vol. 33, no. 2, pp. 238–260, 2018, doi: 10.1214/18-STSA642.
[31] J. Tanner and K. Wei, “Normalized iterative hard thresholding for matrix completion,” SIAM Journal on Scientific Computing, vol. 35, no. 5, pp. S104–S125, 2013, doi: 10.1137/120876459.
[32] Q. Shi, H. Lu, and Y. M. Cheung, “Rank-one matrix completion with automatic rank estimation via L1-norm regularization,” IEEE Transactions on Neural Networks and Learning Systems, vol. 29, no. 10, pp. 4744–4757, 2018, doi: 10.1109/TNNLS.2017.2766160.
[33] I. Selesnick, “Sparsity-assisted signal smoothing (revisited),” in International Conference on Acoustics, Speech, and Signal Processing, 2017, pp. 4546–4550, doi: 10.1109/ICASSP.2017.7953017.
[34] E. C. Chi and T. Li, “Matrix completion from a computational statistics perspective,” Wiley Interdisciplinary Reviews: Computational Stats, vol. 11, no. 5, p. article: e1469, 2019, doi: 10.1002/wics.1469.
[35] S.-W.Hong and P.Bao, “An edge-preserving subband coding model based on non-adaptive and adaptive regularization,” Image and Vision Computing, vol. 18, no. 8, pp. 573–582, 2000, doi: 10.1016/S0262-8856(99)00020-7.
[36] L. D. Paarmann, Design and analysis of analog filters: A signal processing perspective. 2001.
[37] R. Storn and K. Price, “Differential evolution - A simple and efficient heuristic for global optimization over continuous spaces,” Journal of Global Optimization, vol. 11, no. 4, pp. 341–359, 1997, doi: 10.1023/A:1008020821328.
[38] F. M. Harper and J. A. Konstan, “The movielens datasets : history and context,” ACM TRANSACTIONS ON INTERACTIVE INTELLIGENT SYSTEMS, vol. 5, no. 4, p. Article No.: 19, 2015, doi: 10.1145/2827872.
[39] F. Zhang, “Improving recommendation lists through neighbor diversification,” in International Conference on Intelligent Computing and Intelligent Systems, 2009, vol. 3, pp. 222–225, doi: 10.1109/ICIISYS.2009.5358201.
Kai Xu received the M.S. degree in computer technology from Shantou University, China, in 2020. He is currently pursuing the Ph.D. degree with the School of Software Engineering, South China University of Technology, China. His research interests include convex optimization, recommendation systems, information security, dialogue systems and data mining.

Ying Zhang received the B.A degree in information management and information system from South China Normal University, Guangzhou, China in 2009. She is currently a senior engineer with the information department, Second Affiliated Hospital of Shantou University Medical College. Her research interests include machine learning and data mining.

Zhurong Dong received the M.S. degree from the South China University of Technology, Guangzhou, China, in 1998, and the Ph.D. degree in vehicle operation engineering from the Beijing Jiaotong University, Beijing, China, in 2020. He is currently a Full Professor with the Shenzhen Polytechnic, Shenzhen, China. His current research interests include control, detection techniques and data mining.

Zhanyu Li received the M.S. degree from the South China Agricultural University Chengdu, China. He is currently a Full Professor with the Shenzhen Polytechnic, Shenzhen, China. His current research interests include control, detection techniques and data mining.

Bopeng Fang received the Ph.D. degree in computer science and technology from the Southwest Jiaotong University, Chengdu, China, in 2019. He is currently a postdoctoral fellow with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China. His current research interests include computer vision and deep learning.