Neural Network based on Local Dimensionality Reduction for Matrix Completion

1st Ryohei Sasaki
School of Computer Science
Tokyo University of Technology
Tokyo, Japan
sasakirh@stf.teu.ac.jp

2nd Guo Bin
Faculty of Computer and Information Science
Hosei University
Tokyo, Japan
bin.guo.8r@stu.hosei.ac.jp

3rd Katsumi Konishi
Faculty of Computer and Information Science
Hosei University
Tokyo, Japan
konishi@hosei.ac.jp

***Abstract*—This paper deals with a matrix completion problem where each column vector belongs to a low-dimensional manifold. A lot of methods for this problem have been proposed, and most of them are based on the matrix rank minimization problem. Though they work well when column vectors belong to a low dimensional linear subspace, the estimation accuracy deteriorates when vectors belong to a low-dimensional manifold. For matrix completion problems with a low-dimensional manifold, recently, neural network based methods have been proposed, however, they do not achieve high completion accuracy for the problem with high dimension or complex manifold. This paper proposes an architecture for matrix completion based on clustering and local projection, taking advantage of the fact that any manifold can be locally approximated as a low-dimensional linear space. Numerical simulations demonstrate the effectiveness of the proposed method.**

***Index Terms*—nonlinear matrix completion, dimensionality reduction, autoencoder**

I. Introduction

This paper deals with the matrix completion problem when each column vector belongs to a low-dimensional manifold (called nonlinear matrix completion). Matrix completion is the problem of completing the missing elements in a given matrix. The low-rank matrix completion problem has various applications in the field of signal processing, including collaborative filtering in recommendation systems [1], missing completion in image processing [2], channel estimation in wireless communication [3], low-order model fitting and system identification [4], and human-motion recovery [5]. Several methods of estimating missing entries have been studied, and most of them solve matrix completion problems by assuming that column or row vectors of a matrix belong to a low dimensional linear subspace and formulating them as matrix rank minimization problems [6]–[9]. However, in most practical applications, the column vectors of a matrix belong to a low dimensional manifold, these classical methods do not achieve high performance.

To achieve higher recovery performance for real applications, some methods using neural networks for the nonlinear matrix completion problem have proposed [10], [11]. However, these architectures consist of a simple three-layer neural network, which cannot provide high accuracy when the manifold is complex or its dimension is high. For nonlinear matrix completion problems, authors have proposed matrix completion methods based on the assumption that a local neighborhood of each vector on the manifold can be approximated as a low dimensional linear subspace [12]–[14]. These approaches iteratively solve low-rank matrix completion problems for submatrices consisting of neighbor column vectors of the matrix and achieves higher recovery performance. Based on the ideas of these methods, this paper proposes a new neural network architecture with local dimensionality reduction. This architecture is based on the optimization problem in [14], and this paper describes the relationship between the optimization problem and an autoencoder used in the proposed method. Numerical examples show the effectiveness of the proposed method.

II. Matrix Completion and Low-Rank Approach

A. Linear Matrix Completion

This section introduces the nonlinear matrix completion and its solution.

Firstly, this paper describes a standard linear matrix completion problem. The matrix completion problem is a problem which estimate missing entries of a matrix $\boldsymbol{X} \in \mathbb{R}^{M \times N}$. Many classical methods assume that the column (or row) vectors of the matrix belong to a low dimensional linear subspace and solve the following matrix rank minimization problem,

$$\begin{aligned} &\text{Minimize} && \text{rank}(\boldsymbol{X}) \\ &\text{subject to} && x_{m,n} = x^*_{m,n} \text{ for } (m,n) \in \Omega \end{aligned}, \tag{1}$$

where Ω denotes a given index set and $x^*_{m,n}$ denotes a true entry of the matrix to be recovered. Instead of ranks, a variety of objective functions are used since the problem (1) is known as NP-hard. This paper introduces a matrix rank minimization approach [9], which uses an auxiliary variable matrix $\boldsymbol{W} \in \mathbb{R}^{M \times M}$. If the rank of $\boldsymbol{X}$ is low, there exists a low rank matrix $\boldsymbol{W} \in \mathbb{R}^{M \times M}$ such that $\boldsymbol{X} = \boldsymbol{W}\boldsymbol{X}$ is satisfied. Based on this fact, [9] has formulated the low-rank matrix completion

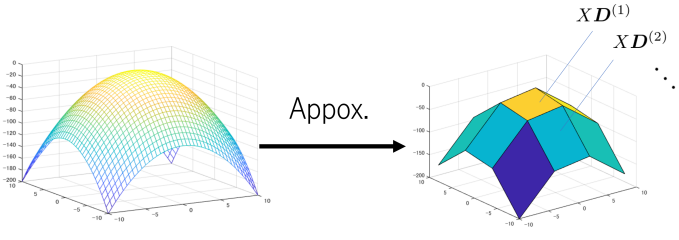

Fig. 1. Approximation idea for a manifold with k-means clustering

problem as follows,

$$\begin{array}{ll} \text{Minimize} & \|\boldsymbol{W}\|_F^2 \\ \text{subject to} & \boldsymbol{X} = \boldsymbol{W}\boldsymbol{X} \\ & x_{m,n} = x^*_{m,n} \text{ for } (m,n) \in \Omega \end{array}, \tag{2}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. To provide an approximate solution of the above problem, (2) is relaxed as follows,

$$\begin{array}{ll} \text{Minimize} & \|(\boldsymbol{I}-\boldsymbol{W})\boldsymbol{X}\|_F^2 + \gamma\|\boldsymbol{W}\|_F^2 \\ \text{subject to} & x_{m,n} = x^*_{m,n} \text{ for } (m,n) \in \Omega \end{array}, \tag{3}$$

where $\boldsymbol{I}$ denotes the identity matrix and $\gamma > 0$ is given constant. Since this problem is the bi-convex quadratic problem for $\boldsymbol{X}$ and $\boldsymbol{W}$, a solution of this problem can be obtained by an alternating optimization for $\boldsymbol{X}$ and $\boldsymbol{W}$ as follows,

$$\begin{array}{lllll} \text{Step 1} & \boldsymbol{W} & \leftarrow & \boldsymbol{W} - \mu\left\{\boldsymbol{W}\boldsymbol{X}\boldsymbol{X}^T - \boldsymbol{X}\boldsymbol{X}^T + \gamma\boldsymbol{W}\right\} \\ \text{Step 2} & \boldsymbol{X} & \leftarrow & \boldsymbol{X} - \tau(\boldsymbol{I}-\boldsymbol{W})^T(\boldsymbol{I}-\boldsymbol{W})\boldsymbol{X} \\ \text{Step 3} & x_{m,n} & \leftarrow & x^*_{m,n} \text{ for } (m,n) \in \Omega \end{array}, \tag{4}$$

where μ and τ denote step size parameters.

However, in most practical applications, the column vectors of a matrix do not belong to a low dimensional linear subspace. Therefore, the classical methods which estimate the low-rank matrix do not achieve high performance.

B. NONLINEAR MATRIX COMPLETION

In recent years, several methods which assume that the column vector of the matrix belong to a low dimensional manifold have been proposed. This paper introduces our approach based on the idea which local area of each vector on the manifold can be approximated by a low dimensional linear subspace.

In [13], authors proposed a new algorithm, which applies the k-means clustering method to $\boldsymbol{x}_i$ w.r.t. Euclidean distance and gives K submatrix completion problems of K clusters. [13] defines diagonal matrix $\boldsymbol{D}^{(k)}$ whose n-th diagonal entries

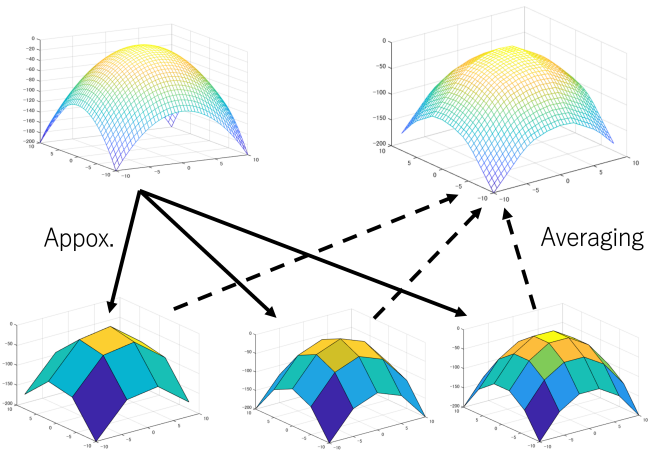

Fig. 2. Approximation idea for a manifold by dividing by multiple patterns and averaging

$d^{(k)}_{n,n}$ is defined by

$$d^{(k)}_{n,n} = \begin{cases} 1 & \text{if } \boldsymbol{x}_n \text{ is a member of the } k\text{-th cluster} \\ 0 & \text{otherwise} \end{cases}, \tag{5}$$

and then considers the following submatrix rank minimization problem with (5) as follows,

$$\begin{array}{ll} \text{Minimize} & \displaystyle\sum_{k=1}^{K} \text{rank}\left(\boldsymbol{X}\boldsymbol{D}^{(k)}\right) \\ \text{subject to} & x_{m,n} = x^*_{m,n} \text{ for } (m,n) \in \Omega \end{array}. \tag{6}$$

Figure 1 shows the idea to solve the nonlinear matrix completion problem based on eq. (6). Based on this idea, we can solve the problem by minimizing each submatrix rank.

However, the algorithm corresponds with a low-rank approach based on the assumption that column vectors belong to multiple low dimensional linear subspaces, that is, a manifold is approximated by piece-wise linear subspaces, and the recovery accuracy decreases. To achieve high recovery accuracy, [13] has proposed multiple k-means clustering based algorithm, which uses k-means clustering for $K \in \{K_1, K_2, ..., K_c\}$, where K_c denotes the number of members in the c-th clustering. Based on the idea, the submatrix completion problem is formulated as follows,

$$\begin{array}{ll} \text{Minimize} & \displaystyle\sum_{c=1}^{C}\sum_{k=1}^{K_c} \text{rank}\left(\boldsymbol{X}\boldsymbol{D}^{(c,k)}\right) \\ \text{subject to} & x_{m,n} = x^*_{m,n} \text{ for } (m,n) \in \Omega \end{array}, \tag{7}$$

where $\boldsymbol{D}^{(c,k)}$ denotes a diagonal matrix whose n-th diagonal

element is defined by

$$d_{n,n}^{(c,k)} = \begin{cases} 1 & \text{if } \boldsymbol{x}_n \text{ is a member of the } k\text{-th cluster} \\ & \text{of the } c\text{-th clustering } (K = K_c) \\ 0 & \text{otherwise} \end{cases} . \tag{8}$$

Figure 2 shows the idea of the problem (7). Based on this idea, [13] proposed an algorithm which obtains each cluster and minimizes its submatrix alternately. Furthermore, [14] proposed a technique to solve the problem (7) substituting the quadratic form (3) for the matrix rank. [14] reformulated the problem (7) as follows,

$$\begin{aligned} \text{Minimize} \quad & \sum_{c=1}^{C} \sum_{k=1}^{K_c} f_\gamma \left(\boldsymbol{W}^{(c,k)}, \boldsymbol{X}\boldsymbol{D}^{(c,k)} \right) \\ \text{subject to} \quad & x_{m,n} = x^*_{m,n} \text{ for } (m,n) \in \Omega \end{aligned} , \tag{9}$$

where f_γ is defined by

$$\begin{aligned} & f_\gamma \left(\boldsymbol{W}^{(c,k)}, \boldsymbol{X}\boldsymbol{D}^{(c,k)} \right) \\ & \quad = \left\| (\boldsymbol{I} - \boldsymbol{W}^{(c,k)}) \boldsymbol{X}\boldsymbol{D}^{(c,k)} \right\|_F^2 + \gamma \left\| \boldsymbol{W}^{(c,k)} \right\|_F^2 \end{aligned} \tag{10}$$

Since this problem is the bi-convex quadratic problem for $\boldsymbol{X}$ and $\boldsymbol{W}^{(c,k)}$, the solution of this problem can be obtained by an alternating optimization for $\boldsymbol{X}$ and $\boldsymbol{W}^{(c,k)}$ for each (c,k).

However, the approximation accuracy of these methods deteriorates when the manifold is complex or its dimension is high. Therefore, this paper proposes a new neural network architecture based on eq. (9) to improve the estimation accuracy.

III. NEURAL NETWORK ARCHITECTURE BASED ON LOCAL DIMENSIONALITY REDUCTION

In this chapter, we consider eq. (9) as an optimization problem for multiple autoencoders, and propose a new neural network architecture for matrix completion.

First, we focus on the objective function in eq. (3). This objective function can be rewritten as follows,

$$\|(\boldsymbol{I} - \boldsymbol{W})\boldsymbol{X}\|_F^2 + \gamma \|\boldsymbol{W}\|_F^2 = \sum_{n=1}^{N} \|\boldsymbol{x}_n - \boldsymbol{W}\boldsymbol{x}_n\|_2^2 + \gamma \|\boldsymbol{W}\|_F^2, \tag{11}$$

where $\boldsymbol{x}_n$ denotes n-th column vector of $\boldsymbol{X}$. If all entries of $\boldsymbol{X}$ are known, the minimization problem of eq. (11) can be considered as an optimization problem for autoencoders using the linear operator $\boldsymbol{W}$. Hence, eq. (3) can be interpreted as the problem of optimizing the autoencoder parameters $\boldsymbol{W}$ and the unknown elements of $\boldsymbol{X}$. Therefore, in this chapter, eq. (9) is reformulated as follows using a neural network $\boldsymbol{G}^{(c)}(\boldsymbol{x}_n)$,

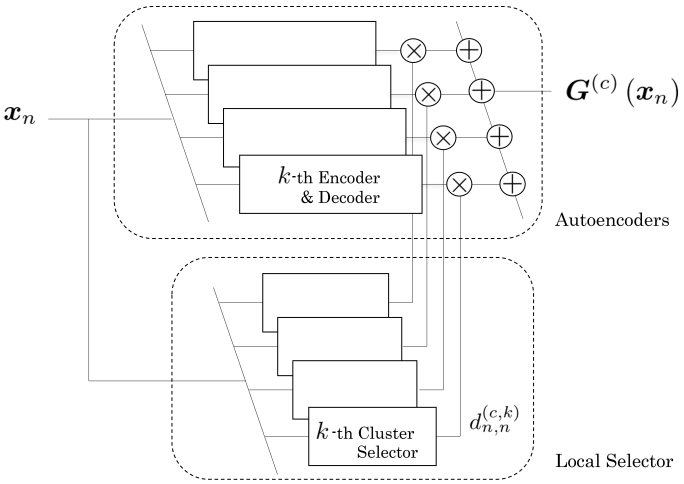

Fig. 3. Local dimensionality reduction net $\boldsymbol{G}^{(c)}$

$$\begin{aligned} \text{Minimize} \quad & g_\gamma(\theta) \\ \text{subject to} \quad & x_{m,n} = x^*_{m,n} \text{ for } (m,n) \in \Omega \end{aligned} , \tag{12}$$

where $g_\gamma(\theta)$ is defined as

$$g_\gamma(\theta) = \sum_{c=1}^{C} \sum_{n=1}^{N} \|\boldsymbol{x}_n - \boldsymbol{G}^{(c)}(\boldsymbol{x}_n)\|_2^2 + \gamma \sum_{c=1}^{C} \mathcal{L}\left(\boldsymbol{G}^{(c)}\right) \tag{13}$$

with $\theta = \left\{ \{x_{m,n}\}_{(m,n)\in\Omega^c}, \left\{ \left\{ \left\{ \boldsymbol{W}_l^{(c,k)} \right\}_{l=1}^{L+1} \right\}_{k=1}^{K_c} \right\}_{c=1}^{C} \right\}$, the neural network $\boldsymbol{G}^{(c)}(\boldsymbol{x}_n)$ is defined as follows,

$$\begin{aligned} & \boldsymbol{G}^{(c)}(\boldsymbol{x}_n) = \\ & \sum_{k=1}^{K_c} d_{n,n}^{(c,k)} \boldsymbol{W}_{L+1}^{(c,k)} \text{ReLU}\left(\boldsymbol{W}_L^{(c,k)} \text{ReLU}\left(\cdots \text{ReLU}\left(\boldsymbol{W}_1^{(c,k)} \boldsymbol{x}_n \right) \right) \right) \end{aligned} \tag{14}$$

with the number of hidden layers L and the activation function ReLU, and the regularization term $\mathcal{L}$ is defined as

$$\mathcal{L}\left(\boldsymbol{G}^{(c)}\right) = \sum_{k=1}^{K_c} \sum_{l=1}^{L+1} \left\| \boldsymbol{W}_l^{(c,k)} \right\|_F^2 . \tag{15}$$

Note that eq. (12) is equivalent to eq. (9) when the number of hidden layers L is 0. Figure 3 shows the overview of the proposed architecture $\boldsymbol{G}^{(c)}$. In Fig. 3, the upper row shows the autoencoders, and the lower row shows the selector that chooses which cluster $\boldsymbol{x}_n$ belongs to.

Finally, this paper shows the algorithm for the minimization problem (12) using k-means clustering and Adam optimizer [15] as shown in Algorithm 1. In Algorithm 1, this paper utilizes the same heuristic technique used in [14] to

give the regularization parameter γ which achieves the best performance.

Algorithm 1 Matrix completion using k-means clustering and autoencoders

Require: $\boldsymbol{X}^{(init)}, \{K_c\}_{c=1}^{C}, \gamma_{max}, \gamma_{min}, \eta_\gamma, t_{max}$

$\boldsymbol{X} \leftarrow \boldsymbol{X}^{(init)}, t \leftarrow 0, \gamma \leftarrow \gamma_{max}$

$\boldsymbol{W}_l^{(c,k)} \leftarrow I$ for all (c, k, l)

repeat

  $t \leftarrow t+1, \gamma \leftarrow \max(\gamma/\eta_\gamma, \gamma_{min})$

  for $c = 1$ to C **do**

    Apply the k-means clustering to $\{\boldsymbol{x}_n\}_{n=1}^{N}$, obtain K_c clusters, and construct $\boldsymbol{D}^{(c,k)}$

  end for

  Update θ with $\boldsymbol{D}^{(c,k)}$ using Adam to minimize $g_\gamma(\theta)$

until $t_{max} < t$

Ensure: $\boldsymbol{X}$.

IV. NUMERICAL EXAMPLES

This section presents several numerical examples for the nonlinear matrix completion. In this section, each n-th column vector of $\boldsymbol{X}^* \in \mathbb{R}^{M\times N}$ is generated by the following mapping function $\boldsymbol{\psi}_p : \mathbb{R}^r \mapsto \mathbb{R}^{\binom{r+p}{p}}$ defined as

$$\boldsymbol{\psi}_p(\boldsymbol{y}) = (\boldsymbol{y}^{\boldsymbol{\alpha}})_{|\boldsymbol{\alpha}|\le p} \in \mathbb{R}^{\binom{r+p}{p}}, \tag{16}$$

where $\boldsymbol{\alpha} = [\alpha_1 \ \cdots \ \alpha_r]$ denotes a multi-index of non-negative integers, $\boldsymbol{y}^{\boldsymbol{\alpha}}$ is defined as $\boldsymbol{y}^{\boldsymbol{\alpha}} = y_1^{\alpha_1}\cdots y_r^{\alpha_r}$, $|\boldsymbol{\alpha}| = \alpha_1 + \cdots + \alpha_r$. The matrix $\boldsymbol{X}^*$ is generated by $\boldsymbol{U}_p\left[\boldsymbol{\psi}_p(\boldsymbol{y}_1), \boldsymbol{\psi}_p(\boldsymbol{y}_2), \cdots, \boldsymbol{\psi}_p(\boldsymbol{y}_N)\right]$ using $\boldsymbol{U}_p \in \mathbb{R}^{M\times\binom{r+p}{p}}$ and $\boldsymbol{Y} = [\boldsymbol{y}_1\ \boldsymbol{y}_2\ \cdots\ \boldsymbol{y}_N] \in \mathbb{R}^{r\times N}$ generated by an i.i.d. continuous uniform distribution whose supports are $[-0.5, 0.5]$ and $[-1, 1]$, the elements of $\boldsymbol{Y}$ are normalized as $\max|(Y)_{i,j}| = 1$. The index set Ω is generated using the Bernoulli distribution with the given probability $q = 0.3$, for which an index (m, n) belongs to Ω. This paper uses root mean square error as

$$\text{RMSE} = \sqrt{\frac{\sum_{(m,n)\in\Omega^c}\left(x_{m,n} - x^*_{m,n}\right)^2}{|\Omega^c|}}$$

to evaluate each algorithm. All numerical experiments were run on the PyTorch framework and used CUDA.

This paper applies the low-rank matrix-completion algorithm IPMS [8], the nonlinear matrix completion method using neural network (GAIN) [11], the locally low-rank approach (LLRA) [14], and Algorithm 1 (proposed method) to several matrix completion problems with $M = 100$, $N = 10,000$, and $p = 3, 5, 7$ for (16). A maximum iteration number of $t_{max} = 1,000$ is used for all algorithms. The parameters for GAIN are the same in [11], and the parameters for IPMS, LLRA, and proposed method are the same in [14]. In the proposed method, the number of the clustering C and each number of clusters K_c are $\{K_c\}_{c=1}^{C} = \{20, 30, 40\}$ which are used in [14].

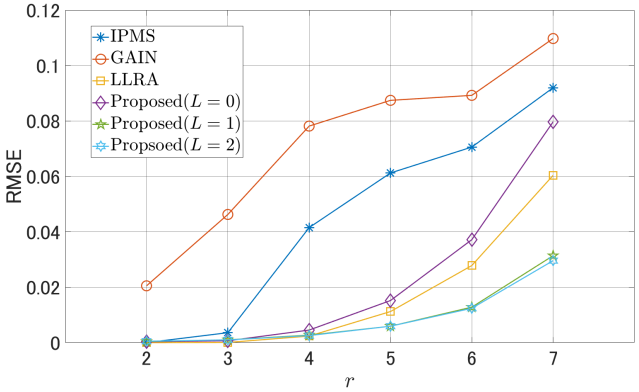

Fig. 4. Average RMSE for 5 trials with $p = 3$ for (16)

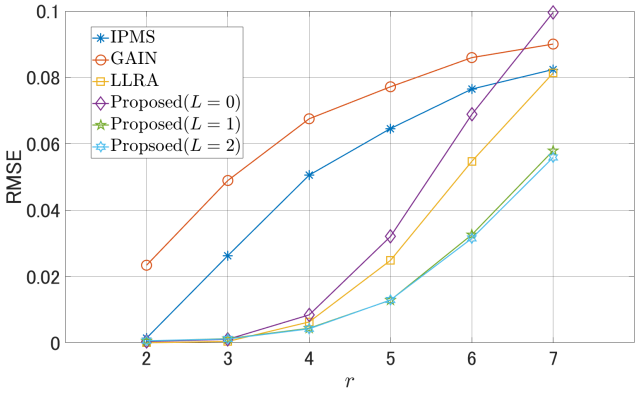

Fig. 5. Average RMSE for 5 trials with $p = 5$ for (16)

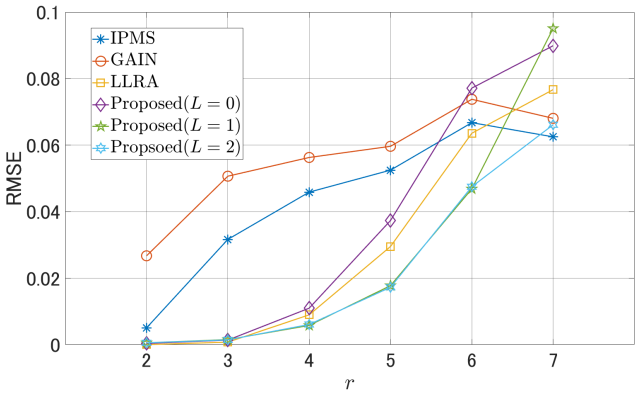

Fig. 6. Average RMSE for 5 trials with $p = 7$ for (16)

The average RMSE of each algorithm is shown in Fig. 4-6 for $r \in \{2, 3, 4, 5, 6\}$. As can be seen, the proposed method ($L = 1, 2$) has higher completion accuracy than the

TABLE I
THE AVERAGE COMPUTATIONAL TIME COST (SECOND) OF THE ALGORITHMS

IPMS [8]	GAIN [11]	LLRA [14]
7.9 [s]	362.7[s]	167.0[s]

Proposed($L = 0$)	Proposed($L = 1$)	Proposed($L = 2$)
115.9 [s]	142.1[s]	167.8[s]

other methods, except for the case where $p = 7, r = 7$. The optimization problems of the proposed method with $L = 0$ are equivalent to LLRA, but their accuracy is different (the accuracy of the proposed method is inferior) because those optimization methods are different. Therefore, it is possible that the proposed method with $L = 1, 2$ can achieve better accuracy by using better optimization method. The average computation time of each algorithm is shown in Table I. It can be seen that the computational time of LLRA is almost equal to that of the proposed method with $L = 2$.

V. CONCLUSION

This paper deals with a nonlinear matrix completion problem, which is a problem of restoring missing entries in a given matrix, where its column vectors belong to a low dimensional manifold. This paper focuses on the fact that a matrix completion problem can be regarded as the optimization problem using autoencoders and proposes a new neural network architecture to achieve higher estimation accuracy for the matrix completion problem. Numerical examples show that the estimation accuracy of the proposed method is higher than the conventional method. As our future work, we formulate the problem so that the clustering is treated as one part of the optimization problem because the technique with clustering and optimizing parameters are heuristic in the proposed algorithm.

ACKNOWLEDGMENT

This work was supported by JSPS KAKENHI Grant Number JP19H02163.

REFERENCES

[1] X. Guan, C. T. Li, Y. Guan, "Matrix factorization with rating completion: An enhanced SVD model for collaborative filtering recommender systems", IEEE Access, vol. 5, pp. 27668–27678, 2017.

[2] K. H. Jin, J. C. Ye, "Annihilating filter-based low-rank Hankel matrix approach for image inpainting," IEEE Trans. Image Processing., vol. 24, no. 11, pp. 3498–3511, 2015.

[3] N. Suga, R. Sasaki, and T. Furukawa, "Channel estimation using matrix factorization based interpolation for OFDM systems," Proc. IEEE Vehicular Technology Conf., 2019.

[4] M. Verhaegen, A. Hansson, "N2SID: Nuclear norm subspace identification of innovation models," Automatica, vol. 72, pp. 57–63, 2016.

[5] Q. Zhao, D. Meng, Z. Xu, W. Zuo, Y. Yan, "L_1-norm low-rank matrix factorization by variational Bayesian method," IEEE Trans. Neural Networks and Learning Systems, vol. 26, no. 4, pp. 825–839, 2015.

[6] J. F. Cai, E. J. Candés, Z. Shen, "A singular value thresholding algorithm for matrix completion," SIAM J. Optimization, vol. 20, no. 4, pp. 1956–1982, 2010.

[7] K. Mohan, M. Fazel, "Iterative reweighted algorithms for matrix rank minimization," J. Machine Learning Research (JMLR), vol. 13, no. 1, pp. 3441–3473, 2012.

[8] K. Konishi, K. Uruma, T. Takahashi, T. Furukawa, "Iterative partial matrix shrinkage algorithm for matrix rank minimization," Signal Processing, vol 100, pp. 124–131, 2014.

[9] K. Konishi, "Parallel GPU implementation of null space based alternating optimization algorithm for large-scale matrix rank minimization," Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing., pp. 3689-3692, 2012.

[10] S. Sedhain, A. K. Menon, S. Sanner, and L. Xie, "Autorec: Autoencoders meet collaborative filtering," Proc. 24th Int. Conf. World Wide Web, pp. 111–112, 2015.

[11] J. Yoon, J. Jordon, and M. van der Schaar, "GAIN: Missing Data Imputation using Generative Adversarial Nets," Proc. 35th Int. Conf. Machine Learning, pp. 5689–5698, 2018.

[12] R. Sasaki, K. Konishi, T. Takahashi, and T. Furukawa, "Local low-rank approach to nonlinear-matrix completion," EURASIP Journal on Advances in Signal Processing, 2021.

[13] K. Konishi, T. Shise, R. Sasaki, and T. Furukawa, "Multiple k-Means clustering based locally low-rank approach to nonlinear matrix completion," Proc. European Signal Processing Conf., 2019.

[14] R. Sasaki and K. Konishi, "Acceleration technique for multiple k-means clustering based locally low-rank approach to nonlinear matrix completion," Proc. European Signal Processing Conf., 2021.

[15] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," Proc. 3rd Int. Conf. Learning Representations (ICLR), 2015.