A Novel Riemannian Metric Based on Riemannian Structure and Scaling Information for Fixed Low-Rank Matrix Completion

Shasha Mao, *Member, IEEE*, Lin Xiong, Licheng Jiao, *Member, IEEE*, Tian Feng, and Sai-Kit Yeung, *Member, IEEE*

***Abstract*—Riemannian optimization has been widely used to deal with the fixed low-rank matrix completion problem, and Riemannian metric is a crucial factor of obtaining the search direction in Riemannian optimization. This paper proposes a new Riemannian metric via simultaneously considering the Riemannian geometry structure and the scaling information, which is smoothly varying and invariant along the equivalence class. The proposed metric can make a tradeoff between the Riemannian geometry structure and the scaling information effectively. Essentially, it can be viewed as a generalization of some existing metrics. Based on the proposed Riemanian metric, we also design a Riemannian nonlinear conjugate gradient algorithm, which can efficiently solve the fixed low-rank matrix completion problem. By experimenting on the fixed low-rank matrix completion, collaborative filtering, and image and video recovery, it illustrates that the proposed method is superior to the state-of-the-art methods on the convergence efficiency and the numerical performance.**

***Index Terms*—Collaborative filtering, fixed low-rank matrix completion, image recovery, Riemannian metric, Riemannian optimization.**

I. Introduction

THE PROBLEM of low-rank matrix completion is viewed as an interesting work in which the missing elements of a low-rank matrix need to be estimated and completed just based on the limited known elements of this matrix. Its pivotal issue is that the matrix must be low-rank or approximatively low-rank. Otherwise, it is impossible to figure out without some additional information [1]. In [2], this problem is traced to the early rank minimization problem (RMP), shown as

$$\min_{\mathbf{X}} \ \operatorname{rank}(\mathbf{X}) \quad \text{s.t.} \ \left\|\mathbf{X}-\hat{\mathbf{X}}\right\|_F \le \varepsilon \tag{1}$$

where $\hat{\mathbf{X}} \in \mathbb{R}^{m\times n}$ is the original matrix, $\mathbf{X} \in \mathbb{R}^{m\times n}$ is the variable, and $\operatorname{rank}(\mathbf{X})$ denotes the rank of $\mathbf{X}$. By (1), a matrix $\mathbf{X}$ with the lowest rank is to approximate $\hat{\mathbf{X}}$ in Frobenius norm with a tolerance of ε. Thereafter, RMP is extended to the general affine rank minimization problem (ARMP) [3]–[7], defined as

$$\min_{\mathbf{X}} \ \operatorname{rank}(\mathbf{X}) \quad \text{s.t.} \ \mathcal{A}(\mathbf{X}) = \mathbf{b} \tag{2}$$

where $\mathcal{A}$ expresses the affine transformation: $\mathbb{R}^{m\times n} \to \mathbb{R}^{d\times 1}$ and $\mathbf{b} \in \mathbb{R}^{d\times 1}$. In general, ARMP is a challenging nonconvex optimization problem which cannot be directly solved within an acceptable time whether in theory or in practice.

Although both ARMP and its approximation [8] are NP-hard, the convex relaxation method with the trace norm minimization has been introduced early in [2], by which the rank constraint is relaxed to a convex function, such as the trace norm. Thus, ARMP can be relaxed as follows:

$$\min_{\mathbf{X}} \ \|\mathbf{X}\|_* \quad \text{s.t.} \ \mathcal{A}(\mathbf{X}) = \mathbf{b} \tag{3}$$

where $\|\mathbf{X}\|_*$ indicates the trace norm of $\mathbf{X}$. Recently, convex relaxation methods [9]–[14] based on the trace norm have attracted a lot of attentions, because its theoretical performance is guaranteed under some conditions and consistency analysis [1], [3], [15]–[17]. However, there is an important bottleneck that the rank of the minimizers will be large in intermediate iterations while the final minimizer of the trace norm minimization is a low-rank matrix. Coinciding with the requirement of solving large-scale problem and minimum memory, the rank estimation of the low-rank matrix is needed. At present, some simple rank estimation strategies of the low-rank matrix have been provided by some researchers [11], [18], [19], but it is still an open problem.

Fortunately, an alternative approach mentioned in [4] and [5] recasts RMP into the low-rank matrix approximation problem

Manuscript received December 11, 2015; revised April 16, 2016; accepted June 25, 2016. This work was supported in part by the National Basic Research Program (973 Program) of China under Grant 2013CB329402, and in part by the National Natural Science Foundation of China under Grant 61473215 and Grant 61573267. This paper was recommended by Associate Editor L. Wang. *(Corresponding author: Lin Xiong.)*

S. Mao, L. Xiong, and L. Jiao are with the Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, International Research Center for Intelligent Perception and Computation, Joint International Research Laboratory of Intelligent Perception and Computation, Xidian University, Xi'an 710071, China (e-mail: skymss0828@gmail.com; bruinxiong@me.com; lchjiao@mail.xidian.edu.cn).

T. Feng and S.-K. Yeung are with the Singapore University of Technology and Design, Singapore 487372 (e-mail: feng_tian@sutd.edu.sg; saikit@sutd.edu.sg).

This paper has supplementary downloadable multimedia material available at http://ieeexplore.ieee.org provided by the authors.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2016.2587825

with the rank inequality constraint, and several algorithms [6], [20] have been addressed to solve this problem recently. Actually, the low-rank matrix completion problem is a special case of ARMP, shown as

$$\begin{aligned} &\min_{\mathbf{X}} \ \|\mathbf{X}\|_* \\ &\text{s.t.} \ \ \mathcal{P}_\Omega(\mathbf{X}) = \mathcal{P}_\Omega\left(\hat{\mathbf{X}}\right) \end{aligned} \tag{4}$$

where Ω is a subset of the complete set of indices $\{1, \ldots, m\} \times \{1, \ldots, n\}$, $\mathcal{P}_\Omega(\mathbf{X})_{ij} = \mathbf{X}_{ij}$ when $(i, j) \in \Omega$, and $\mathcal{P}_\Omega(\mathbf{X})_{ij} = 0$ when $(i, j) \notin \Omega$. In this paper, we focus on its another formulation

$$\begin{aligned} &\min_{\mathbf{X}} \ \frac{1}{|\Omega|}\left\|\mathcal{P}_\Omega(\mathbf{X}) - \mathcal{P}_\Omega(\hat{\mathbf{X}})\right\|_F^2 \\ &\text{s.t.} \ \ \mathbf{X} \in \left\{\mathbf{X} \in \mathbb{R}^{m\times n} \middle| \text{rank}(\mathbf{X}) = r\right\} \end{aligned} \tag{5}$$

where $|\Omega|$ denotes the cardinality of the set Ω. Unlike the problem proposed in [4]–[6], the rank constraint of the problem (5) is the equality constraint, and it is named as the fixed low-rank matrix completion problem. It is well-known that the matrix factorization idea as a popular way has been utilized to solve [19, eq. (5)] and [21, eq. (4)]. Even in the computer vision field, the matrix factorization model is also widely used [22]–[30]. However, the nonuniqueness of the matrix factorization easily makes the solving procedure to fall into the local minimum.

In order to effectively overcome the nonuniqueness of the matrix factorization, many algorithms [18], [31]–[38] are proposed based on the Riemannian quotient manifold by virtue of the introduction of the equivalence class to solve the problem (5). Specially, it is a prominent feature of some algorithms that the underlying Riemannian geometry structure derived from the matrix factorization model of the search space is considered, for example, the Riemannian submanifold is adopted on the tangent space of the product space or total space in [32]. Moreover, both Ngo and Saad [34] and Mishra *et al.* [35] have introduced how to use the scaling information in solving the problem (5). Because of considering the scaling information, the scaled gradient direction outperforms the canonical gradient direction under the gradient-based optimization.

Inspired by Mishra *et al.* [35] and Meyer *et al.* [36], it is known that both the Riemannian quotient manifold structure and the scaling information play indispensable roles in constructing a proper Riemannian metric. Actually, a Riemannian metric defines the search space of tangent vectors in tangent space. Riemannian geometry structure and the scale information constrain this search space from different angles, respectively. In well-conditioned problems, Riemannian geometry structure guarantees the global optimal solution. However, most practical problems in system recommendation and computer vision are not well-conditioned. When only Riemannian geometry structure behind the specific matrix factorization model is considered, it can not gain an acceptable solution for the ill-conditioned problems. By virtue of the adaptive preconditioning character, the scale information assures an acceptable solution in ill-conditioned problems. But, while considering only the scale information of the cost function of the matrix completion problem, an insufficient performance may be gained because of the lack of the intrinsic structure of data. It is a worth considering problem that how to construct a more general Riemannian metric.

Based on the above analyses, we propose a new Riemannian metric which can simultaneously consider the Riemannian geometry structure behind the full-rank matrix factorization model and the scaling information derived from the hessian information of the cost function of the low-rank matrix approximation problem in this paper, and we prove the proposed metric is smoothly varying and invariant along the equivalent class. With the proposed metric, we also propose an efficient algorithm under the framework of the Riemannian manifold optimization to solve a more general matrix completion problem, such as the collaborative filtering, image and video recovery. Simply speaking, the proposed Riemannian metric is equivalent to an additive metric characterized with a linear combination of the Riemannian geometry and the scaling information, where a weight coefficient is used to balance their contributions, but it can bring a wonderful result when it is applied to the gradient-based Riemannian optimization algorithm. More specifically, the proposed metric can make tradeoff between the Riemannian geometry structure and the scaling information by adjusting the weight coefficient.

Compared with some previous works [34]–[36] the research contributions of this paper are summarily listed as follows.

1) A novel Riemannian metric is proposed, which is the first work of simultaneously considering the structure of the Riemannian quotient manifold and the scaling information, and it can effectively realize mutual compensation of advantages attributed to the Riemannian structure and the scaling information.
2) The full-rank matrix factorization model can be extended by our proposed Riemannian metric, because its corresponding product space can be equipped with not only other Riemannian metrics but also our metric.
3) A Riemannian nonlinear conjugate gradient algorithm with our proposed metric is designed to verify the effectiveness of our metric, which can gain better performance compared with state-of-the-art algorithms.
4) Our Riemannian metric is more general for the fixed low-rank matrix completion, and some metrics are equivalent to special cases[1] of our metric, such as qGeomMC, Right inv. $\mathbf{GH}^T$, LMaFit.

This paper is organized as follows. We review the Riemannian structure and the scaling information in Section II. Sections III and IV show the details of constructing the proposed Riemannian metric and recalculating the components of the optimization on Riemannian quotient manifold, respectively. In Section V, a nonlinear conjugate gradient algorithm (RiemSScRNCG) is introduced. Section VI shows the experimental results. Finally, the conclusion remarks and the future work are presented in Section VII.

[1]The detailed discussions of the connection between them are shown in the supplementary material.

II. Riemannian Structure of Quotient Manifold and Scaling Information of the Problem

The equivalence class is the basis of the structure of the Riemannian quotient manifold. By introducing the concept of equivalence class, the nonuniqueness of the matrix factorization is avoided so that the global minima can be gained. Moreover, the scaling information of the specific problem [such as (5)] derives from the hessian information of the corresponding objective function according to [35] which is used in this paper.

A. Geometry Structure of Riemannian Quotient Manifold

Before the discussion of Riemannian manifold geometry structure, we will review the two-factor factorization of the fixed low-rank nonsymmetric matrix, also called matrix bi-factorization [39]. Note that this paper focuses on the two-factor factorization. Given a fixed low-rank nonsymmetric real matrix $\mathbf{X} \in \mathbb{R}^{m\times n}$ which satisfies $\text{rank}(\mathbf{X}) = r$ and $r \ll \min(m, n)$, a well known factorization is derived from the thin singular value decomposition (SVD): $\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T$, where $\Sigma \in \mathbb{R}^{r\times r}$ is a diagonal matrix, and columns of $\mathbf{U} \in \mathbb{R}^{m\times r}$ and $\mathbf{V} \in \mathbb{R}^{n\times r}$ satisfy the orthogonality, respectively. Furthermore, it is easily rearranged as follows:

$$\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T = (\mathbf{U}\Sigma^{\frac{1}{2}})(\Sigma^{\frac{1}{2}}\mathbf{V}^T) = \mathbf{G}\mathbf{H}^T \tag{6}$$

where $\mathbf{G} = \mathbf{U}\Sigma^{(1/2)} \in \mathbb{R}^{m\times r}$, $\mathbf{H} = \mathbf{V}\Sigma^{(1/2)} \in \mathbb{R}^{n\times r}$ and $\text{rank}(\mathbf{G}) = \text{rank}(\mathbf{H}) = r$. Let $\mathbb{R}_*^{m\times r}$ denote the set of all matrices with $m \times r$ whose columns are linearly independent, and then we have $\mathbf{G} \in \mathbb{R}_*^{m\times r}$. Similarly, we get $\mathbf{H} \in \mathbb{R}_*^{n\times r}$. Therefore, (6) is also called as full-rank matrix factorization. However, the factorization of (6) is not unique, because there is an equation for any invertible matrix $\mathbf{D} \in \mathbb{R}_*^{r\times r}$

$$\mathbf{G}\mathbf{H}^T = \mathbf{G}\mathbf{D}^{-1}\mathbf{D}\mathbf{H}^T = \left(\mathbf{G}\mathbf{D}^{-1}\right)\left(\mathbf{H}\mathbf{D}^T\right)^T. \tag{7}$$

A more general mathematical description of (7) is a mapping

$$(\mathbf{G}, \mathbf{H}) \mapsto \left(\mathbf{G}\mathbf{D}^{-1}, \mathbf{H}\mathbf{D}^T\right), \forall \mathbf{D} \in \mathbb{R}_*^{r\times r}. \tag{8}$$

In view of $\mathbf{G}$ and $\mathbf{H}$, the subspace $\text{span}(\mathbf{G})$ spanned by columns of $\mathbf{G}$ is equivalent to the subspace $\text{span}(\mathbf{G}\mathbf{D}^{-1})$ spanned by columns of $\mathbf{G}\mathbf{D}^{-1}$ for $\mathbf{D}$. The matrix $\mathbf{X}$ leaves unchanged via the mapping (8). This invariance takes root in the fact that the column spaces of $\mathbf{G}$ and $\mathbf{H}$ are invariant to the transformation of coordinates (e.g., the rotation of coordinates). Thus, the equivalence class of $\mathbf{X}$ is defined as

$$\begin{aligned}[\mathbf{X}] &= [(\mathbf{G}, \mathbf{H})] \\ &= \left\{ \left(\mathbf{G}\mathbf{D}^{-1}, \mathbf{H}\mathbf{D}^T\right) \middle| \mathbf{G} \sim \mathbf{G}\mathbf{D}^{-1}, \mathbf{H} \sim \mathbf{H}\mathbf{D}^T, \mathbf{D} \in \mathbb{R}_*^{r\times r} \right\}\end{aligned} \tag{9}$$

where $\sim$ denotes the equivalence relation on $\mathbb{R}_*^{m\times r}$ or $\mathbb{R}_*^{n\times r}$, $(\mathbf{G}, \mathbf{H})$ represents the full-rank factorization of $\mathbf{X}$, $\mathbf{D}$ is any invertible matrix, and $\mathbb{R}_*^{r\times r}$ is also called as the general linear group GL_r in [40]. Moreover, let $\mathbb{R}_*^{m\times r} \times \mathbb{R}_*^{n\times r}$ be a product space denoted by $\bar{\mathcal{X}}$, also called as the total space [40]. By virtue of the equivalence class and the product space, it is natural to introduce the quotient manifold, and we have

$$\mathcal{X} = \bar{\mathcal{X}} / \text{GL}_r \tag{10}$$

where $\mathcal{X}$ is called the quotient space or quotient manifold.[2] A mapping $\pi : \bar{\mathcal{X}} \to \mathcal{X}$ constructs the relationship between the total space $\bar{\mathcal{X}}$ and the quotient manifold $\mathcal{X}$, named as the quotient map or canonical projection map [33].

An element x of $\mathcal{X}$ is denoted by $x = \pi([\bar{x}])$, where $\bar{x} \in \bar{\mathcal{X}}$ is the matrix representative of $[\bar{x}]$ in $\bar{\mathcal{X}}$. Furthermore, the tangent vector ζ_x in $T_x\mathcal{X}$ needs a matrix representative in $\bar{\mathcal{X}}$ to calculate effectively, where $T_x\mathcal{X}$ expresses the tangent space of $\mathcal{X}$ at x. Specifically, there is a unique horizontal lifted representative of ζ_x, denoted by $\bar{\zeta}_{\bar{x}}$[3] in $T_{\bar{x}}\bar{\mathcal{X}}$, where $T_{\bar{x}}\bar{\mathcal{X}}$ expresses the tangent space of $\bar{\mathcal{X}}$ at $\bar{x}$. In other words, the matrix representative $(\bar{\zeta}_{\mathbf{G}}, \bar{\zeta}_{\mathbf{H}})$ of ζ_x is only restricted to the horizontal direction in $T_{\bar{x}}\bar{\mathcal{X}}$, which means it remains unchanged along the equivalence class $[\bar{x}]$ in that direction. The set of those directions constructs a subspace of $T_{\bar{x}}\bar{\mathcal{X}}$, named as the horizontal subspace $\mathcal{H}_{\bar{x}}\bar{\mathcal{X}}$, and the complementary space of $\mathcal{H}_{\bar{x}}\bar{\mathcal{X}}$ is tangent to $[\bar{x}]$, named as the vertical subspace $\mathcal{V}_{\bar{x}}\bar{\mathcal{X}}$. Then, we have $\mathcal{V}_{\bar{x}}\bar{\mathcal{X}} \oplus \mathcal{H}_{\bar{x}}\bar{\mathcal{X}} = T_{\bar{x}}\bar{\mathcal{X}}$.

Furthermore, an inner product varied smoothly on the horizontal subspace of $T_{\bar{x}}\bar{\mathcal{X}}$ is presented by the way of a metric $\bar{g}_{\bar{x}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$ defined on $\bar{\mathcal{X}}$. Since $\bar{g}_{\bar{x}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$ is independent of $\bar{x} \in [\bar{x}]$, it defines a valid Riemannian metric $g_x(\zeta_x, \eta_x)$ in $T_x\mathcal{X}$ of $\mathcal{X}$, shown as follows:

$$g_x(\zeta_x, \eta_x) = \bar{g}_{\bar{x}}\left(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}}\right) \tag{11}$$

where ζ_x and η_x are the tangent vectors in $T_x\mathcal{X}$, and their horizontal lifted representatives are $\bar{\zeta}_{\bar{x}}$ and $\bar{\eta}_{\bar{x}}$ in $\mathcal{H}_{\bar{x}}\bar{\mathcal{X}}$. Additionally, the product structure of $\bar{\mathcal{X}}$ naturally inherits the metric from each factor (such as $\mathbb{R}_*^{m\times r}$ and $\mathbb{R}_*^{n\times r}$) in [40]. Therefore, $\mathcal{X}$ equipped with $g_x(\zeta_x, \eta_x)$ is named as a Riemannian quotient manifold of $\bar{\mathcal{X}}$. Moreover, more details can be referenced in [33] and [41].

B. Scaling Information of the Matrix Approximation

It is well-known that the Newton algorithm of the second-order optimization method is a classical scaling gradient descent algorithm. For $\mathbf{X} = \mathbf{G}\mathbf{H}^T$, let $\mathbf{G} \in \mathbb{R}_*^{m\times r}$ vectorize, and a vector $\mathbf{g} \in \mathbb{R}^{mr\times 1}$ is obtained by stacking the columns of $\mathbf{G}$ below on another. Similarly, a vector $\mathbf{h} \in \mathbb{R}^{nr\times 1}$ is gained for $\mathbf{H}$. Then they are stacked as $[\,\mathbf{g}^T \quad \mathbf{h}^T\,]^T \in \mathbb{R}^{(m+n)r\times 1}$ whose full hessian matrix is a symmetric matrix with $(m+n)r \times (m+n)r$. Although the objective function of the matrix approximation problem

$$\bar{f}(\mathbf{G}, \mathbf{H}) = \frac{1}{2}\left\|\hat{\mathbf{X}} - \mathbf{G}\mathbf{H}^T\right\|_F^2 \tag{12}$$

is nonconvex because of including the product of two variables $\mathbf{G}$ and $\mathbf{H}$, it is strictly convex on $\mathbf{G}$ when $\mathbf{H}$ is fixed

[2] It resembles Euclidean space locally and defines the differential structure globally. It is equipped with an inner product varied smoothly on the horizontal subspace of the tangent space.

[3] Here, for convenience, we abuse the notation by denoting $\bar{\zeta}_{\bar{x}}$ as the tangent vector in the tangent space $T_{\bar{x}}\bar{\mathcal{X}}$ of the total space $\bar{\mathcal{X}}$ at $\bar{x}$ as well as the horizontal lifted representative of the tangent vector in $T_{\bar{x}}\bar{\mathcal{X}}$.

and vice versa. In order to reduce the complexity, a diagonal approximation matrix is adopted instead of the full hessian information, whose diagonal elements are chosen from the full hessian matrix of (12). Moreover, the diagonal approximation hessian matrix has the form $\begin{bmatrix} \mathrm{Hess}_{\mathbf{g}} & \mathbf{0}^{mr\times nr} \\ \mathbf{0}^{nr\times mr} & \mathrm{Hess}_{\mathbf{h}} \end{bmatrix} \in \mathbb{R}^{(m+n)r\times(m+n)r}$, where $\mathrm{Hess}_{\mathbf{g}} \in \mathbb{R}^{mr\times mr}$ and $\mathrm{Hess}_{\mathbf{h}} \in \mathbb{R}^{nr\times nr}$ are two diagonal matrices. According to [22], the scaling gradients with the diagonal elements of the diagonal approximation hessian matrix have the forms $\mathrm{Hess}_{\mathbf{g}}^{-1}(\partial\bar{f}/\partial\mathbf{g})$ and $\mathrm{Hess}_{\mathbf{h}}^{-1}(\partial\bar{f}/\partial\mathbf{h})$, where $(\partial\bar{f}/\partial\mathbf{g})$ and $(\partial\bar{f}/\partial\mathbf{h})$ are the Euclidean gradients in vectorized form. Therefore, a particular metric equipped with scaling information on $T_{\bar{x}}\bar{\mathcal{X}}$ is addressed in [35], shown as

$$\bar{g}_{\bar{x}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}}) = \mathrm{Tr}\big(\big(\mathbf{H}^T\mathbf{H}\big)\,\bar{\zeta}_{\mathbf{G}}^T\bar{\eta}_{\mathbf{G}}\big) + \mathrm{Tr}\big(\big(\mathbf{G}^T\mathbf{G}\big)\,\bar{\zeta}_{\mathbf{H}}^T\bar{\eta}_{\mathbf{H}}\big) \quad (13)$$

where $\bar{x} = (\mathbf{G}, \mathbf{H})$ and $\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}} \in T_{\bar{x}}\bar{\mathcal{X}}$.

III. New Riemannian Metric on the Horizontal Subspace of the Riemannian Quotient Manifold

Most researches construct a metric based on either the Riemannian geometry structure or the scale information of a problem. In [33], a metric is constructed based on the geometry structure of Riemannian quotient manifold, and the metric [35] shown by (13) is designed based on the scale information of the cost function without the geometry structure. In order to distinguish three metrics in the following part, we use $\bar{g}_{\bar{x}}^{\mathrm{geom}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$, $\bar{g}_{\bar{x}}^{\mathrm{scal}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$, and $\bar{g}_{\bar{x}}^{\mathrm{eucl}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$ to denote them, respectively, where $\bar{g}_{\bar{x}}^{\mathrm{eucl}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$ is an Euclidean metric equipped in the product space $\mathbb{R}_*^{m\times r} \times \mathbb{R}_*^{n\times r}$ in [36]:

$$\bar{g}_{\bar{x}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}}) = \mathrm{Tr}\big(\bar{\zeta}_{\mathbf{G}}^T\bar{\eta}_{\mathbf{G}}\big) + \mathrm{Tr}\big(\bar{\zeta}_{\mathbf{H}}^T\bar{\eta}_{\mathbf{H}}\big) \quad (14)$$

where $\bar{x} = (\mathbf{G}, \mathbf{H})$ and $\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}} \in T_{\bar{x}}\bar{\mathcal{X}}$.

Although the good performance obtained by the constructed metrics has demonstrated the significant of the Riemannian geometry and the scale information, there are still some disadvantages in them. For example, for the ill-conditional problems, an acceptable solution cannot be obtained while only considering the Riemannian geometry structure; when only using the scale information, an insufficient performance may be gained. Hence, we designed one novel Riemannian metric on the tangent space[4] $T_{\bar{x}}\bar{\mathcal{X}}$ of $\bar{\mathcal{X}}$ based on both the Riemannian geometry structure and the scaling information. By considering them simultaneously, a matrix Ψ_1 is constructed

$$\Psi_1 = \alpha\begin{pmatrix} \big(\mathbf{G}^T\mathbf{G}\big)^{-1} & \mathbf{0}^{r\times r} \\ \mathbf{0}^{r\times r} & \big(\mathbf{H}^T\mathbf{H}\big)^{-1} \end{pmatrix} + (1-\alpha)\begin{pmatrix} \mathbf{H}^T\mathbf{H} & \mathbf{0}^{r\times r} \\ \mathbf{0}^{r\times r} & \mathbf{G}^T\mathbf{G} \end{pmatrix}$$

where α is a weight coefficient and $\Psi_1 \in \mathbb{R}^{2r\times 2r}$. Interestingly, it is found that Ψ_1 is equivalent to a linear combination of the Riemannian geometry and the scaling information, where α is used to adjust the significance of the Riemannian geometry and the scaling information. For Ψ_1, Lemma 1 is given and its proof is shown in the supplementary material.

Lemma 1: Given two matrices $\mathbf{G} \in \mathbb{R}_*^{m\times r}$ and $\mathbf{H} \in \mathbb{R}_*^{n\times r}$ and $\alpha \in [0, 1]$, the matrix Ψ_1 is a positive definite matrix.

Then, by using the block matrix and the properties of matrix trace [42], we obtain a new Riemannian metric $\bar{g}_{\bar{x}}^{\mathrm{addi}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$ on $T_{\bar{x}}\bar{\mathcal{X}}$, shown as

$$\begin{aligned}\bar{g}_{\bar{x}}^{\mathrm{addi}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}}) = {} & \mathrm{Tr}\Big(\Big(\alpha\big(\mathbf{G}^T\mathbf{G}\big)^{-1} + (1-\alpha)\mathbf{H}^T\mathbf{H}\Big)\bar{\zeta}_{\mathbf{G}}^T\bar{\eta}_{\mathbf{G}}\Big) \\ & + \mathrm{Tr}\Big(\Big(\alpha\Big(\mathbf{H}^{\mathbf{T}}\mathbf{H}\Big)^{-1} + (1-\alpha)\mathbf{G}^T\mathbf{G}\Big)\bar{\zeta}_{\mathbf{H}}^T\bar{\eta}_{\mathbf{H}}\Big).\end{aligned} \quad (15)$$

From (15), it is obvious that $\bar{g}_{\bar{x}}^{\mathrm{addi}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}}) = \bar{g}_{\bar{x}}^{\mathrm{scal}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$ if $\alpha = 0$ and $\bar{g}_{\bar{x}}^{\mathrm{addi}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}}) = \bar{g}_{\bar{x}}^{\mathrm{geom}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$ if $\alpha = 1$, which indicates that $\bar{g}_{\bar{x}}^{\mathrm{geom}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$ and $\bar{g}_{\bar{x}}^{\mathrm{scal}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$ are equivalent to two special cases of the proposed new metric in fact.

According to the analysis in Section II, given a tangent vector ζ_x in $T_x\mathcal{X}$ at $x \in \mathcal{X}$, it is uniquely represented by the horizontal lifted representative $\bar{\zeta}_{\bar{x}} \in \mathcal{H}_{\bar{x}}\bar{\mathcal{X}}$ in $T_{\bar{x}}\bar{\mathcal{X}}$ endowed with a horizontal subspace $\mathcal{H}_{\bar{x}}\bar{\mathcal{X}}$, satisfied with $\zeta_x = \mathrm{D}\pi(\bar{x})[\bar{\zeta}_{\bar{x}}]$, where $\mathrm{D}\pi(\bar{x})[\bar{\zeta}_{\bar{x}}]$ means the directional derivative of $\pi(\bar{x})$ along the direction $\bar{\zeta}_{\bar{x}}$. For the proposed Riemannian metric (15), we can gain Proposition 1.

Proposition 1: Let $\bar{x} \in \bar{\mathcal{X}}$ be represented by $(\mathbf{G}, \mathbf{H}) \in \mathbb{R}_*^{m\times r} \times \mathbb{R}_*^{n\times r}$ and ζ_x be a tangent vector in $T_x\mathcal{X}$ of $\mathcal{X} = \mathbb{R}_*^{m\times r} \times \mathbb{R}_*^{n\times r}/\mathrm{GL}_r$ at $x = \pi([(\mathbf{G}, \mathbf{H})])$, where π is a quotient map[5] $\pi : \bar{\mathcal{X}} \to \mathcal{X}$. The horizontal lifts of $\zeta_{\pi([(\mathbf{G},\mathbf{H})])}$ at points $(\mathbf{G}, \mathbf{H})$ and $(\mathbf{G}\mathbf{D}^{-1}, \mathbf{H}\mathbf{D}^T)$ have a relationship with each other, shown as

$$\big(\bar{\zeta}_{\mathbf{G}\mathbf{D}^{-1}}, \bar{\zeta}_{\mathbf{H}\mathbf{D}^T}\big) = \Big(\bar{\zeta}_{\mathbf{G}}\mathbf{D}^{-1}, \bar{\zeta}_{\mathbf{H}}\mathbf{D}^T\Big),\ \forall\mathbf{D} \in \mathrm{GL}_r.$$

Then, the metric (15) is smoothly varying on the horizontal subspace and invariant along the equivalence class. Thus, we have

$$\begin{aligned} g_x(\zeta_x, \eta_x) &= \bar{g}_{(\mathbf{G},\mathbf{H})}^{\mathrm{addi}}\big((\bar{\zeta}_{\mathbf{G}}, \bar{\zeta}_{\mathbf{H}}), (\bar{\eta}_{\mathbf{G}}, \bar{\eta}_{\mathbf{H}})\big) \\ &= \bar{g}_{(\mathbf{G}\mathbf{D}^{-1},\mathbf{H}\mathbf{D}^{\mathbf{T}})}^{\mathrm{addi}}\big((\bar{\zeta}_{\mathbf{G}\mathbf{D}^{-1}}, \bar{\zeta}_{\mathbf{H}\mathbf{D}^T}), \big(\bar{\eta}_{\mathbf{G}\mathbf{D}^{-1}}, \bar{\eta}_{\mathbf{H}\mathbf{D}^T}\big)\big). \end{aligned}$$

The proof of Proposition 1 is shown in the Appendix.

IV. Components of the Optimization on Riemannian Quotient Manifold

To the best of our knowledge, the domain of numerical optimization on smooth manifolds has been advanced significantly in the past few years [31]–[33], [35], [40], [41]. In this section, we need reconsider and recalculate the components of the optimization algorithm for the proposed new metric, e.g., projections onto subspaces, retraction, and vector transport.

A. Projections Onto the Tangent Space and the Horizontal Subspace

Since we focus on the total space $\bar{\mathcal{X}}$, the tangent space $T_{\bar{x}}\bar{\mathcal{X}}$ has the product structure: $T_{\bar{x}}\bar{\mathcal{X}} = T_{\mathbf{G}}\mathbb{R}_*^{m\times r} \times T_{\mathbf{H}}\mathbb{R}_*^{n\times r}$, where

[4]Indeed, the constructed Riemannian metric is on the horizontal subspace $\mathcal{H}_{\bar{x}}\bar{\mathcal{X}}$ of the tangent space $T_x\mathcal{X}$ for the Riemannian quotient manifold.

[5]The mapping π is the continuous surjective, called as Riemannian submersion. In other words, a Riemannian submersion is a submersion of Riemannian manifolds such that $\mathrm{D}\pi$ maintains the inner products of vectors which are orthogonal to equivalence classes.

$\bar{x} = (\mathbf{G}, \mathbf{H}) \in \bar{\mathcal{X}}$ and $\bar{\mathcal{X}} = \mathbb{R}_*^{m\times r} \times \mathbb{R}_*^{n\times r}$. Additionally, as $\mathbb{R}_*^{m\times r}$ and $\mathbb{R}_*^{n\times r}$ are an open subset[6] of $\mathbb{R}^{m\times r}$ and $\mathbb{R}^{n\times r}$, respectively, we have $T_{\bar{x}}\bar{\mathcal{X}} = \mathbb{R}^{m\times r} \times \mathbb{R}^{n\times r}$. Hence, for an arbitrary point $\hat{x} = (\Xi_{\mathbf{G}}, \Xi_{\mathbf{H}}) \in \mathbb{R}^{m\times r} \times \mathbb{R}^{n\times r}$ in Euclidean product space, the projection operator $\Upsilon_{\bar{x}}$ onto $T_{\bar{x}}\bar{\mathcal{X}}$ is the identity map, shown as

$$\Upsilon_{\bar{x}}(\Xi_{\mathbf{G}}, \Xi_{\mathbf{H}}) = (\Xi_{\mathbf{G}}, \Xi_{\mathbf{H}}). \tag{16}$$

Based on the previous analyses, $T_{\bar{x}}\bar{\mathcal{X}}$ is decomposed into the vertical and horizontal subspaces which are denoted as $\mathcal{V}_{\bar{x}}\bar{\mathcal{X}}$ and $\mathcal{H}_{\bar{x}}\bar{\mathcal{X}}$, respectively. In order to obtain the projection operator $\Pi_{\bar{x}}$ onto $\mathcal{H}_{\bar{x}}\bar{\mathcal{X}}$, we need Lemma 2.

Lemma 2: The Riemannian quotient manifold $\mathbb{R}_*^{m\times r} \times \mathbb{R}_*^{n\times r}/\mathrm{GL}_r$ equipped with the Riemannian metric (15) admits a set of horizontal tangent vectors $(\bar{\xi}_{\mathbf{G}}, \bar{\xi}_{\mathbf{H}}) \in \mathbb{R}^{m\times r} \times \mathbb{R}^{n\times r}$ which meets the condition

$$\begin{aligned}\mathbf{G}^T\bar{\xi}_{\mathbf{G}}\Big(\alpha\big(\mathbf{G}^T\mathbf{G}\big)^{-1} + (1-\alpha)\mathbf{H}^T\mathbf{H}\Big)\\ = -\Big((1-\alpha)\mathbf{G}^T\mathbf{G} + \alpha\big(\mathbf{H}^T\mathbf{H}\big)^{-1}\Big)\bar{\xi}_{\mathbf{H}}^T\mathbf{H}.\end{aligned} \tag{17}$$

The proof of Lemma 2 is shown in the supplementary material. Thus, the vertical subspace has the expression $\mathcal{V}_{\bar{x}}\bar{\mathcal{X}} = \mathcal{V}_{(\mathbf{G},\mathbf{H})}\bar{\mathcal{X}} = \{(\mathbf{G}\Phi, \mathbf{H}\Phi^T) | \Phi \in \mathbb{R}^{r\times r}\}$, and the horizontal subspace has the expression

$$\begin{aligned}\mathcal{H}_{\bar{x}}\bar{\mathcal{X}} = \Big\{ (\bar{\xi}_{\mathbf{G}}, \bar{\xi}_{\mathbf{H}}) \Big| \mathbf{G}^T\bar{\xi}_{\mathbf{G}}\Big(\alpha\big(\mathbf{G}^T\mathbf{G}\big)^{-1} + (1-\alpha)\mathbf{H}^T\mathbf{H}\Big)\\ = -\Big((1-\alpha)\mathbf{G}^T\mathbf{G} + \alpha\big(\mathbf{H}^T\mathbf{H}\big)^{-1}\Big)\bar{\xi}_{\mathbf{H}}^T\mathbf{H}\\ \bar{\xi}_{\mathbf{G}} \in \mathbb{R}^{m\times r}, \quad \bar{\xi}_{\mathbf{H}} \in \mathbb{R}^{n\times r}\Big\}.\end{aligned}$$

Projecting any tangent vector $\bar{\zeta}_{\bar{x}} \in T_{\bar{x}}\bar{\mathcal{X}}$ onto the horizontal subspace is performed by the projection operator

$$\Pi_{\bar{x}}(\bar{\zeta}_{\bar{x}}) = \big(\bar{\zeta}_{\mathbf{G}} - \mathbf{G}\Phi, \bar{\zeta}_{\mathbf{H}} - \mathbf{H}\Phi^T\big) \tag{18}$$

where $\Phi \in \mathbb{R}^{r\times r}$ can be uniquely obtained by solving a Lyapunov equation. The calculation of Φ is detailed in the supplementary material.

B. Retraction and Vector Transport

In the iterative optimization method, it always involves computing a search direction (such as the gradient) and then moving along this direction with the step size. In the search process, the next iteration point is still in the search space. In the geometric sense, the exponential mapping is the most natural retraction on the Riemannian quotient manifold. But its calculation cost [43] is too expensive to implement in many practical applications. Fortunately, a mapping $R_{\bar{x}} : \mathcal{H}_{\bar{x}}\bar{\mathcal{X}} \to \bar{\mathcal{X}}$ is adopted to make a retraction from horizontal subspace to the total space in [33] and [40], which locally approximates the exponential mapping $\mathrm{Exp}_{\bar{x}} : \mathcal{H}_{\bar{x}}\bar{\mathcal{X}} \to \bar{\mathcal{X}}$, and it can be numerically calculated corresponding to an abstract retraction $R_x : T_x\mathcal{X} \to \mathcal{X}$ on the Riemannian quotient manifold. Thus, we have the retraction of the product space corresponding to the full-rank matrix factorization here

$$\begin{aligned}\bar{x}_{\mathrm{new}} &= R_{\bar{x}_{\mathrm{old}}}\big(s_{\mathrm{old}}\bar{\xi}_{\bar{x}_{\mathrm{old}}}\big)\\ &= \big(\mathbf{G} + s_{\mathrm{old}}\bar{\xi}_{\mathbf{G}}, \mathbf{H} + s_{\mathrm{old}}\bar{\xi}_{\mathbf{H}}\big)\end{aligned} \tag{19}$$

[6]Since its complement $\{\bar{\mathbf{G}} \in \mathbb{R}^{m\times r} | \det(\bar{\mathbf{G}}^T\bar{\mathbf{G}}) = 0\}$ is closed.

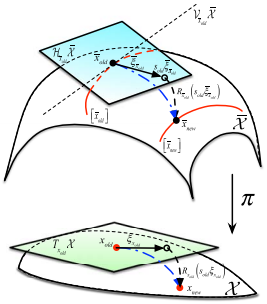

Fig. 1. Visualization of the retraction. The horizontal subspace $\mathcal{H}_{\bar{x}_{\mathrm{old}}}\bar{\mathcal{X}}$ defines an unique horizontal lifted space for the abstract tangent space $T_{x_{\mathrm{old}}}\mathcal{X}$. The retraction $R_{\bar{x}_{\mathrm{old}}}$ maps a horizontal tangent vector $s_{\mathrm{old}}\bar{\xi}_{\bar{x}_{\mathrm{old}}}$ with the step size s_{old} onto the point $\bar{x}_{\mathrm{new}}$. $\bar{\xi}_{\bar{x}_{\mathrm{old}}}$ denotes the horizontal lift of a tangent vector $\xi_{x_{\mathrm{old}}}$ on the Riemannian quotient manifold $\mathcal{X}$. The red lines denote the equivalence classes on the total space $\bar{\mathcal{X}}$. The red solid points are denoted as the points on $\mathcal{X}$. The blue dotted lines are the geodesics on $\mathcal{X}$ and $\bar{\mathcal{X}}$.

where s_{old} is the step size, $\bar{\xi}_{\bar{x}_{\mathrm{old}}} \in \mathcal{H}_{\bar{x}_{\mathrm{old}}}\bar{\mathcal{X}}$ denotes a search direction, and $\bar{x}_{\mathrm{new}} \in \bar{\mathcal{X}}$ is a gained new iteration corresponding to $\bar{x}_{\mathrm{old}} = (\mathbf{G}, \mathbf{H})$. As follows, a visualization of the retraction is given in Fig. 1. In brief, the retraction is a mapping from the horizontal subspace to the total space of the Riemannian quotient manifold. Actually, the purpose of introducing the operation of retraction is to ensure that the next iteration point is still on the quotient manifold.

As the Euclidean space is the flat space, the parallel transport of a vector does not change its own direction. But, on the Riemannian quotient manifold, if a tangent vector is moved from one point to the next iteration point by the parallel transport, it will not be a tangent vector generally. Thus, we transport the tangent vector from one point to the next iteration point along the path on the Riemannian quotient manifold by removing the component of the transported tangent vector in the space [40]. Specifically, this space is the orthogonal complement space of the tangent space. The vector transport $\mathcal{T}_{x\to z}$ from x to z on $\mathcal{X}$ can be viewed as a mapping $\mathcal{T}_{x\to z} : T_x\mathcal{X} \to T_z\mathcal{X}$, where $z = R_x(s\eta_x)$ and s is a step size. By virtue of the horizontal lifted representatives of $\bar{\zeta}_{\bar{x}}$ and the search direction $\bar{\eta}_{\bar{x}}$ in $\bar{\mathcal{X}}$, the horizontal lift $\overline{\mathcal{T}_{s\eta_x}(\zeta_x)}$ of the vector transport $\mathcal{T}_{s\eta_x}(\zeta_x)$ can be given in terms of the operator (18) as

$$\overline{\mathcal{T}_{s\eta_x}(\zeta_x)} = \Pi_{\bar{x}+s\bar{\eta}_{\bar{x}}}\big(\bar{\zeta}_{\bar{x}}\big). \tag{20}$$

In fact, the horizontal lift $\overline{\mathcal{T}_{x\to x+s\eta_x}}$ of the vector transport $\mathcal{T}_{x\to x+s\eta_x}$ can be viewed as a mapping $\overline{\mathcal{T}_{x\to x+s\eta_x}} : \mathcal{H}_{\bar{x}}\bar{\mathcal{X}} \to \mathcal{H}_{\bar{x}+s\bar{\eta}_{\bar{x}}}\bar{\mathcal{X}}$, and Fig. 2 is a visualization of the vector transport.

C. Complexity Analysis

Supposed $m \gg r$, $n \gg r$, we have $\mathbf{G} \in \mathbb{R}_*^{m\times r}$, $\mathbf{H} \in \mathbb{R}_*^{n\times r}$, $\bar{\zeta}_{\mathbf{G}}, \bar{\eta}_{\mathbf{G}} \in \mathbb{R}^{m\times r}$ and $\bar{\zeta}_{\mathbf{H}}, \bar{\eta}_{\mathbf{H}} \in \mathbb{R}^{n\times r}$ according to the previous analyses. For the proposed Riemannian metric $\bar{g}_{\bar{x}}^{\mathrm{addi}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$, its numerical complexity is $\mathcal{O}(mr^2 + nr^2)$ based on (15). Given $\bar{\zeta}_{\mathbf{G}} \in \mathbb{R}^{m\times r}$and $\bar{\zeta}_{\mathbf{H}} \in \mathbb{R}^{n\times r}$, we have $\Upsilon_{\bar{x}}(\bar{\zeta}_{\mathbf{G}}, \bar{\zeta}_{\mathbf{H}}) = (\bar{\zeta}_{\mathbf{G}}, \bar{\zeta}_{\mathbf{H}})$, which is the identity map, and thus there is no computational cost in practical. For the projection operator $\Pi_{\bar{x}}(\bar{\zeta}_{\bar{x}})$

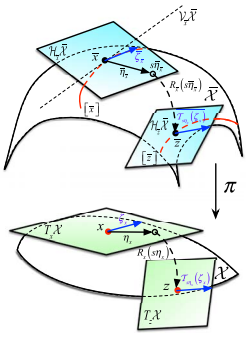

Fig. 2. Visualization of the vector transport. The vector transport $\mathcal{T}_{s\eta_x}(\zeta_x)$ on the Riemannian quotient manifold $\mathcal{X}$ is performed actually on the total space $\bar{\mathcal{X}}$ by the horizontal lifts, and it is denoted by $\overline{\mathcal{T}_{s\eta_x}(\zeta_x)}$. The tangent vector $\bar{\zeta}_{\bar{x}}$ is transported "parallelly" from the point $\bar{x} \in \bar{\mathcal{X}}$ to the point $\bar{z} = R_{\bar{x}}(s\bar{\eta}_{\bar{x}}) \in \bar{\mathcal{X}}$ and we obtain $\overline{\mathcal{T}_{s\eta_x}(\zeta_x)}$ on the horizontal subspace $\mathcal{H}_{\bar{z}}\bar{\mathcal{X}}$ eventually.

onto the horizontal subspace $\mathcal{H}_{\bar{x}}\bar{\mathcal{X}}$ shown in (18), the numerical complexity is $\mathcal{O}(mr^2 + nr^2)$. The computational cost of $R_{\bar{x}}$ shown in (19) is $\mathcal{O}(mr + nr)$. The numerical complexity of $\Pi_{\bar{x}+s\bar{\eta}_{\bar{x}}}(\bar{\zeta}_{\bar{x}})$ in (20) includes the complexity $\mathcal{O}(mr^2 + nr^2)$ of forming the Lyapunov equation and the complexity $\mathcal{O}(r^3)$ of solving it. The detailed analyses are shown in the supplementary material.

V. RiemSScRNCG Algorithm

With the proposed metric, we design a Riemannian nonlinear conjugate gradient algorithm based on the additive metric of Riemannian structure and scaling information to solve the fixed low-rank matrix completion problem in this paper, named as RiemSScRNCG algorithm, and it is outlined in Algorithm 1. As follows, we introduce some details of RiemSScRNCG.

On the Riemannian quotient manifold $\mathcal{X}$, a smooth function $f : \mathcal{X} \to \mathbb{R}$ is induced by the corresponding smooth cost function $\bar{f} : \bar{\mathcal{X}} \to \mathbb{R}$ which is invariant along the GL_r. The Riemannian gradient $\mathrm{grad}_x f$ of f at a point x can be uniquely represented by its horizontal lift $\overline{\mathrm{grad}_x f}$. Since $\bar{f}$ is constant on each equivalence class, we get $\mathrm{grad}_{\bar{x}}\bar{f}$ belongs to the horizontal subspace. Actually, the projection operator $\Pi_{\bar{x}}$ mentioned in (18) is implicitly included. Naturally, we have

$$\overline{\mathrm{grad}_x f} = \mathrm{grad}_{\bar{x}}\bar{f}. \tag{21}$$

Specifically, the matrix representative of $\mathrm{grad}_{\bar{x}}\bar{f}$ can be obtained from the definition of the Riemannian gradient. In other words, the Riemannian gradient $\mathrm{grad}_{\bar{x}}\bar{f}$ is the unique tangent vector of $T_{\bar{x}}\bar{\mathcal{X}}$, which meets $\mathrm{D}\bar{f}[\bar{\eta}_{\bar{x}}] = \bar{g}_{\bar{x}}(\mathrm{grad}_{\bar{x}}\bar{f}, \bar{\eta}_{\bar{x}})$ for all $\bar{\eta}_{\bar{x}} \in T_{\bar{x}}\bar{\mathcal{X}}$, where $\mathrm{D}\bar{f}[\bar{\eta}_{\bar{x}}]$ denotes the standard Euclidean directional derivative of $\bar{f}$ along the direction $\bar{\eta}_{\bar{x}}$, and $\bar{g}_{\bar{x}}(\bullet, \bullet)$ is the Riemannian metric defined on $T_{\bar{x}}\bar{\mathcal{X}}$ of $\bar{\mathcal{X}}$.

For the fixed low-rank matrix completion problem (5), it can be rewritten as

$$\min_{(\mathbf{G},\mathbf{H})\in\mathbb{R}_*^{m\times r}\times\mathbb{R}_*^{n\times r}} \frac{1}{|\Omega|}\left\|\mathcal{P}_\Omega(\mathbf{G}\mathbf{H}^T) - \mathcal{P}_\Omega(\hat{\mathbf{X}})\right\|_F^2. \tag{22}$$

Thus, we have the cost function

$$\bar{f}(\mathbf{G},\mathbf{H}) = \frac{1}{|\Omega|}\left\|\mathcal{P}_\Omega\left(\mathbf{G}\mathbf{H}^{\mathbf{T}}\right) - \mathcal{P}_\Omega\left(\hat{\mathbf{X}}\right)\right\|_F^2.$$

Algorithm 1 RiemSScRNCG Algorithm

Input: Matrix $\mathcal{P}_\Omega(\hat{\mathbf{X}})$, Ω, fixed rank r, α.
Output: Matrices $\mathbf{G} \in \mathbb{R}_*^{m\times r}$ and $\mathbf{H} \in \mathbb{R}_*^{n\times r}$ gained via (22).

1: Initialize two matrices $\mathbf{G}_0 \in \mathbb{R}_*^{m\times r}$ and $\mathbf{H}_0 \in \mathbb{R}_*^{n\times r}$, $\beta_0 := 0$, $i := 0$;
2: Compute $\mathrm{grad}_{\bar{x}_i}\bar{f} = \left(\mathrm{grad}_{\mathbf{G}_i}\bar{f}, \mathrm{grad}_{\mathbf{H}_i}\bar{f}\right)$;
3: Let the search direction $\bar{\zeta}_{\bar{x}_i}$ be $\left(\bar{\zeta}_{\mathbf{G}_i}, \bar{\zeta}_{\mathbf{H}_i}\right)$ and have
$$\left(\bar{\zeta}_{\mathbf{G}_i}, \bar{\zeta}_{\mathbf{H}_i}\right) = -\left(\mathrm{grad}_{\mathbf{G}_i}\bar{f}, \mathrm{grad}_{\mathbf{H}_i}\bar{f}\right);$$
4: **repeat**
5: **if** $i == 0$
6: **then** Guess the initial step size s_i° via exact line-search;
7: **else** Adopt the adaptive step size strategy mentioned in [20] to get s_i°.
8: **end if**
9: Obtain the Armijo step size s_i by satisfying the Armijo condition with s_i°;
10: Compute $\mathbf{G}_{i+1}$ and $\mathbf{H}_{i+1}$ according to the retraction (19), and the update steps are shown as
$$\mathbf{G}_{i+1} = \mathbf{G}_i + s_i\bar{\zeta}_{\mathbf{G}_i}, \mathbf{H}_{i+1} = \mathbf{H}_i + s_i\bar{\zeta}_{\mathbf{H}_i};$$
11: Compute $\mathrm{grad}_{\bar{x}_{i+1}}\bar{f} = \left(\mathrm{grad}_{\mathbf{G}_{i+1}}\bar{f}, \mathrm{grad}_{\mathbf{H}_{i+1}}\bar{f}\right)$;
12: Compute the coefficient β_{i+1} by (23);
13: Compute the search direction $\bar{\zeta}_{\bar{x}_{i+1}} = \left(\bar{\zeta}_{\mathbf{G}_{i+1}}, \bar{\zeta}_{\mathbf{H}_{i+1}}\right)$, where
$$\bar{\zeta}_{\mathbf{G}_{i+1}} = -\mathrm{grad}_{\mathbf{G}_i}\bar{f} + \beta_{i+1}\Pi_{\mathbf{G}_i+s_i\bar{\zeta}_{\mathbf{G}_i}}\left(\bar{\zeta}_{\mathbf{G}_i}\right),$$
$$\bar{\zeta}_{\mathbf{H}_{i+1}} = -\mathrm{grad}_{\mathbf{H}_i}\bar{f} + \beta_{i+1}\Pi_{\mathbf{H}_i+s_i\bar{\zeta}_{\mathbf{H}_i}}\left(\bar{\zeta}_{\mathbf{H}_i}\right);$$
14: **if** $\bar{g}^{\mathrm{addi}}_{\bar{x}_{i+1}}\left(\bar{\zeta}_{\bar{x}_{i+1}}, \mathrm{grad}_{\bar{x}_{i+1}}\bar{f}\right) > 0$
15: **then**
$$\bar{\zeta}_{\mathbf{G}_{i+1}} = -\mathrm{grad}_{\mathbf{G}_{i+1}}\bar{f}, \bar{\zeta}_{\mathbf{H}_{i+1}} = -\mathrm{grad}_{\mathbf{H}_{i+1}}\bar{f};$$
16: **end if**
17: $i := i + 1$;
18: **until** convergence.

For the cost function $\bar{f}(\mathbf{G},\mathbf{H})$, its numerical complexity mainly comes from the term $\mathcal{P}_\Omega(\mathbf{G}\mathbf{H}^T)$. By the ingenious use of the sparse structure, the computational cost of $\mathcal{P}_\Omega(\mathbf{G}\mathbf{H}^T)$ is $|\Omega|r$. Thus, the numerical complexity of $\bar{f}(\mathbf{G},\mathbf{H})$ is $\mathcal{O}(|\Omega|r)$. Additionally, the partial derivative $(\partial\bar{f}/\partial\bar{x})$ of $\bar{f}$ with respect to $\bar{x}$ is obtained by $(\partial\bar{f}/\partial\bar{x}) = ((\partial\bar{f}/\partial\mathbf{G}), (\partial\bar{f}/\partial\mathbf{H})) \in \mathbb{R}^{m\times r} \times \mathbb{R}^{n\times r}$, where $(\partial\bar{f}/\partial\mathbf{G}) = (2/|\Omega|)(\mathcal{P}_\Omega(\mathbf{G}\mathbf{H}^T) - \mathcal{P}_\Omega(\hat{\mathbf{X}}))\mathbf{H}$ and $(\partial\bar{f}/\partial\mathbf{H}) = ((2/|\Omega|)(\mathcal{P}_\Omega(\mathbf{G}\mathbf{H}^T) - \mathcal{P}_\Omega(\hat{\mathbf{X}})))^T\mathbf{G}$. Therefore, the matrix representative of $\mathrm{grad}_{\bar{x}}\bar{f}$ is calculated on the horizontal subspace equipped with the proposed Riemannian metrics $\bar{g}^{\mathrm{addi}}_{\bar{x}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$, and we have

$$\left(\mathrm{grad}_{\mathbf{G}}\bar{f}, \mathrm{grad}_{\mathbf{H}}\bar{f}\right) = \left(\mathbf{Q}\mathbf{H}\mathbf{A}^{-1}, \mathbf{Q}^T\mathbf{G}\mathbf{B}^{-1}\right)$$

where $\mathbf{A} = \alpha(\mathbf{G}^T\mathbf{G})^{-1} + (1-\alpha)\mathbf{H}^T\mathbf{H}$, $\mathbf{B} = \alpha(\mathbf{H}^T\mathbf{H})^{-1} + (1-\alpha)\mathbf{G}^T\mathbf{G}$, and $\mathbf{Q} = (2/|\Omega|)(\mathcal{P}_\Omega(\mathbf{G}\mathbf{H}^T) - \mathcal{P}_\Omega(\hat{\mathbf{X}}))$. By the computational deduction, the numerical complexity of $\mathrm{grad}_{\mathbf{G}}\bar{f}$ and $\mathrm{grad}_{\mathbf{H}}\bar{f}$ are all $\mathcal{O}(|\Omega|r + mr^2 + nr^2)$.

The introduction of the vector transport $\mathcal{T}_{s\eta_x}(\zeta_x)$ is ready for the nonlinear conjugate gradient algorithm on the Riemannian quotient manifold. The search direction $\bar{\zeta}_{\bar{x}_{i+1}}$ can be computed by the variant of the classical Polak-Ribière-Polyak (PRP) which is generalized to the Riemannian manifold in [40].

For the Riemannian metric $\bar{g}^{\text{addi}}_{\bar{x}}(\bar{\zeta}_{\bar{x}}, \bar{\eta}_{\bar{x}})$, the calculation of the coefficient β_{i+1} is done as follows:

$$\beta_{i+1} = \frac{\bar{g}^{\text{addi}}_{\bar{x}_{i+1}}\left(\text{grad}_{\bar{x}_{i+1}}\bar{f}, \text{grad}_{\bar{x}_{i+1}}\bar{f} - \overline{\mathcal{T}_{x_{i+1}}(\zeta_{x_i})}\right)}{\bar{g}^{\text{addi}}_{\bar{x}_i}\left(\text{grad}_{\bar{x}_i}\bar{f}, \text{grad}_{\bar{x}_i}\bar{f}\right)}. \quad (23)$$

VI. Experiment and Analysis

In order to validate the performance of our Riemannian metric, three experiments are performed on synthetic datasets, real-world recommendation system datasets, and image and video recovery, respectively, and their details are shown in the following parts. From (22), the fixed-rank constraint is embedded into the full-rank factorization model, and this optimization problem has attracted a lot of attention in recent years. Thus, some state-of-the-art algorithms are compared with the proposed method in our experiments. The compared algorithms are roughly summarized into three categories based on optimization methods, shown as follows.

1) *Trace Norm Minimization:* The representative algorithms are SVT [9], FPCA [10], APGL [11], SOFT-IMP [44], and IALM [45].
   a) *SVT*[7]*:* The step size δ is set to $1.2(mn/|\Omega|)$ and the parameter τ is set to $\tau = 5n$ as suggested by Cai *et al.* [9].
   b) *FPCA*[8]*:* The parameter η_μ is set to 0.25 and τ is set to 1 as suggested in [10]. We set the parameter $\bar{\mu} = 10^{-30}\mu^0$, where $\mu^0 = \|\mathcal{A}^*(b)\|_2$ and $\mathcal{A}^*$ denotes the adjoint operator of $\mathcal{A}$.
   c) *APGL*[9]*:* The parameter $\bar{\mu}$ is set to $10^{-30}\mu^0$, where the definition of μ^0 is the same as in FPCA. For APGL, a quadratic approximation function of the objective function is constructed.
   d) *IALM*[10]*:* The parameter μ_0 is set to $(1/\|\hat{\mathbf{X}}\|_2)$, and the parameter has $\rho = 1.2172 + 1.8588\rho_s$ mentioned in [45], where $\rho_s = (|\Omega|/mn)$. IALM is the representative of the alternating direction method (ADM) [46].
   e) *SVP*[11]*:* The step size η_t is set to $(1/(1+\delta)p)$ as suggested in [4], where p denotes the density of sampled entries, $p = (|\Omega|/mn)$, and $\delta = (1/3)$.

   The PROPACK [47] is used to compute a partial SVD in all above mentioned algorithms except FPCA. For FPCA, a fast Monte Carlo algorithm [48] is applied to compute an approximate SVD. Although SVP [4] does not belong to the trace norm minimization algorithm, the partial SVD needs to be computed by PROPACK. Thus, we list SVP above.
2) *Matrix Factorization Model:* The representative algorithm is LMaFit [19].
   a) *LMaFit*[12]*:* The parameter γ_1 is set to 0.7 and $\tilde{\omega} = 50$ suggested in [19]. LMaFit is an efficient nonlinear successive over-relaxation algorithm.
3) *Manifold Optimization:* The representative algorithms are OptSpace [18], ScGrassMC [34], LRGeomCG [32], qGeomMC [35], Right inv. $\mathbf{GH}^T$ [33], $\mathbf{UBV}^T$ [33], and $\mathbf{UY}^T$ [33].
   a) *OptSpace*[13]*:* We set the parameter $\tau = 10^{-3}$ as suggested by authors. The MATLAB implementation without regularized term is adopted in our experiments.
   b) *ScGrassMC*[14]*:* It is a gradient-based method equipped with scaled metric on the bi-Grassmannian manifold for low-rank matrix completion.
   c) *LRGeomCG*[15]*:* The iterative points move on the embedded submanifold conceptually, and the numerical implementation is done by using the matrix factorization model, such as SVD. The Euclidean metric is adopted.
   d) *qGeomMC*[16]*:* The scaling information is considered to construct a proper Riemannian metric on the Riemannian quotient manifold.
   e) *Right inv.* $\mathbf{GH}^T$[17]*:* Its metric is constructed by the structure of Riemannian quotient manifold and is invariant along the equivalence class in the product space.
   f) $\mathbf{UBV}^T$*:* It comes from the polar factorization model, where $\mathbf{B}$ is a $r \times r$ symmetric positive definite matrix.
   g) $\mathbf{UY}^T$*:* It comes from the subspace-projection factorization model. The total space is defined as $\text{St}(m, r) \times \mathbb{R}^{n\times r}_*$.

In experiments, all reported times are wall-clock time that contains the setup phase of the solvers but excludes the setup phase of the problems. For the fair comparison, all other manifold optimization methods (except OptSpace) adopt the Armijo line-search to compute an Armijo-optimal step size on nonlinear conjugate gradient algorithm. For the initial guess of the step size, the exact line-search is employed in all methods, and the adaptive step size strategy mentioned in [33] is also applied. Additionally, the maximum number of iterations is set to 250 for all methods unless otherwise specified, the parameter α is set to 0.1 or 0.2 for RiemSScRNCG, and the parameters are set to the optimal values mentioned in relevant works for other methods.

A. Experiments on Synthetic Datasets

In this experiment, we have two settings for the fixed low-rank matrix completion problem: 1) well-conditioned random low-rank matrices whose singular values change smoothly, and

[7] http://statweb.stanford.edu/∼candes/svt/
[8] http://www1.se.cuhk.edu.hk/∼sqma/FPCA.html
[9] http://www.math.nus.edu.sg/∼mattohkc/NNLS.html
[10] http://perception.csl.illinois.edu/matrix-rank/sample_code.html
[11] http://www.cs.utexas.edu/∼pjain/svp/
[12] http://lmafit.blogs.rice.edu/
[13] http://web.engr.illinois.edu/∼swoh/software/optspace/code.html
[14] http://www-users.cs.umn.edu/∼thango/
[15] http://web.math.princeton.edu/∼bartv/matrix_completion.html
[16] http://www.montefiore.ulg.ac.be/∼mishra/softwares/qGeomMC.html
[17] http://www.montefiore.ulg.ac.be/∼mishra/codes.html

TABLE I
MEAN RESULTS FOR FIVE RANDOM INSTANCES FOR LOW-RANK MATRIX COMPLETION WITH A FIXED RANK

Fixed rank = 10, OS = 5, $\alpha = 0.2$, $tol = 10^{-20}$, Maxiter = 250					
Size of problem	2000, CN=1.138	4000, CN=1.093	8000, CN=1.063	16000, CN=1.044	32000, CN=1.031
RiemSScRNCG, time(s.), #its.	**1.68**, *87.2*	**4.74**, *83.6*	**10.93**, 93	**24.85**, *98.2*	**60.37**, *110.8*
qGeomMC, time(s.), #its.	1.94, 105.6	5.87, 107.8	*11.44*, 99.8	27.77, 107	83.79, 144.6
Right inv. $\mathbf{GH}^T$, time(s.), #its.	2.26, 138.6	7.22, 125.6	15.02, 120	30.71, 114.8	71.47, 128.2
LRGeomCG, time(s.), #its.	2.29, 101	8.24, 132.6	20.36, 153.8 (1)	50.38, 170.4	128.45, 201.8 (2)
LMaFit, time(s.), #its.	1.75, 124.6	6.19, 138.8	15.96, 173.4	51.88, 250 (5)	107.4, 250 (5)
ScGrassMC, time(s.), #its.	2.91, 109.6 (3)	1.59, 18.2 (5)	6.83, 46.8 (5)	20.81, 60 (3)	32.24, 44.2 (5)
ScGrassMC (EL), time(s.), #its.	*1.71*, **54.4**	*4.91*, **60.2**	12.53, **63.8**	*25.39*, **65**	*62.83*, **64**
$\mathbf{UBV}^T$, time(s.), #its.	2.13, 89.2	6.07, 84.3	11.72, *79.2*	22.79, 71.2 (1)	88.1, 131.6 (1)
$\mathbf{UY}^T$, time(s.), #its.	2.32, 106	7.98, 125.4	21.43, 162.4 (1)	41.58, 140.6	103.39, 160.4 (1)

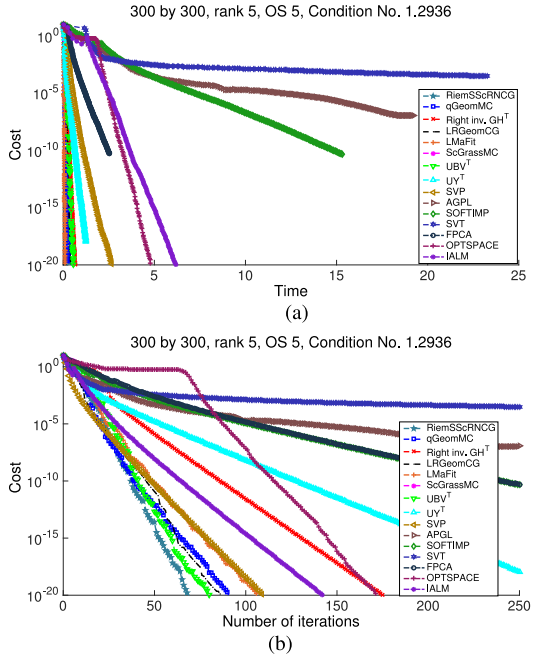

Fig. 3. Convergence behaviors for all algorithms for the small well-conditioned fixed low-rank matrix completion. (a) Convergence time curves of all algorithms. (b) Convergence iteration curves of all algorithms.

the condition number (CN) is 1.4 at most and 2) ill-conditioned random low-rank matrices whose singular values are exponential decay, and the CN reaches 200 at most. Each low-rank matrix is set as an $m \times n$ random matrix with rank r generated like [9], and two matrices $\mathbf{M}_L \in \mathbb{R}^{m\times r}$ and $\mathbf{M}_R \in \mathbb{R}^{n\times r}$ are generated with independent and identically distributed standard Gaussian entries. Then, the well-conditioned matrix $\hat{\mathbf{X}}$ with rank r is gained by $\hat{\mathbf{X}} = \mathbf{M}_L\mathbf{M}_R^T$. The set of observations Ω is uniformly sampled from all sets of cardinality $|\Omega|$ at random. The degree of freedom (DOF) of $m \times n$ matrix with rank r is $r(m+n-r)$. The over-sampling (OS) coefficient is defined as $\text{OS} = |\Omega|/r(m+n-r)$. Thus, sampling rate (SR) is determined by $\text{SR} = \text{OS} * \text{DOF}/mn$. Additionally, because the synthetic data are in noiseless case, we set that each algorithm is terminated when the cost function $\bar{f}$ is below $\text{tol} = 10^{-20}$ or the number of iterations exceeds the given maximum.

First, we perform the experiment on the small well-conditioned problem which is a 300×300 matrix with rank $= 5$, OS $= 5$ and CN $= 1.2936$. Actually, this experiment can effectively help choose some appropriate algorithms for the comparison in follow-up experiments. The convergence behavior is illustrated over one random independent instance in Fig. 3. From Fig. 3, it is seen that the convergence performance of the trace norm minimization algorithms is roughly inferior to the manifold optimization algorithms except OptSpace. Particularly, our proposed RiemSScRNCG method outperforms other compared manifold optimization methods in aspect of time and number of iterations. Additionally, like the block-coordinate descent algorithm, LMaFit has a smaller computational cost at each iteration and it has a better convergence than the trace norm minimization algorithms. Thus, LMaFit is always preserved in the follow-up experiments.

Second, the experiment of the large well-conditioned problems is implemented, and its results are shown in Table I. In Table I, we report on the mean running time and the number of iterations[18] over five random independent instances, where the best results are bolded and the second best ones are italicized. Since the numerical computational problem emerges when the adaptive step size strategy is applied in ScGrassMC, we also adopt the exact linear search during the whole algorithm procedure for ScGrassMC, denoted as ScGrassMC(EL). Note that OptSpace and trace norm minimization algorithms are not included, because their computational costs are unacceptable on large matrix completion compared with other methods.

From Table I, it is obvious that RiemSScRNCG spends the least time compared with other methods, even if its number of iteration (the second best) is more than ScGrassMC(EL) (the best of all). Moreover, we also find that Right inv. $\mathbf{GH}^T$ gains the worst results in methods of Riemannian structure from small-scale to large-scale problems, but it is still superior to LRGeomCG. It indicates the metric with Riemannian structure is better than the Euclidean metric. In addition, the convergence performance of qGeomMC and ScGrassMC(EL) is better than Right inv. $\mathbf{GH}^T$, LMaFit, $\mathbf{UBV}^T$, and $\mathbf{UY}^T$ except RiemSScRNCG, which indicates the scaling information is also of superiority and indispensability. In short, RiemSScRNCG obtains the least computational

[18] In Tables I and II, the number in the parenthesis shows the number of stopping the algorithm when the maximum of the iterative number is arrived or the local minimum is obtained, and its higher value indicates the worse global convergence performance.

TABLE II
MEAN RESULTS FOR FIVE RANDOM INSTANCES WITH DIFFERENT CONDITION NUMBERS

Fixed size n = 2000, Fixed rank = 10, OS = 5, $\alpha = 0.1$, $tol = 10^{-20}$, Maxiter = 250					
Condition number	10	20	30	40	50
RiemSScRNCG, time(s.), #its.	*1.61*, *75.4*	**2.07**, **97**	**2.11**, **82.4**	**2.65**, **124.6** (1)	**2.66**, **120.4** (1)
qGeomMC, time(s.), #its.	**1.26**, **63.2**	**2.07**, *102.8*	*2.53*, **113**	3.61, *183.6* (3)	4.11, *192.2* (3)
Right inv.$\mathbf{GH}^T$, time(s.), #its.	5.11, 250 (5)	5.08, 250 (5)	6.11, 250 (5)	5.07, 250 (5)	5.67, 250 (5)
LRGeomCG, time(s.), #its.	5.95, 250 (5)	5.92, 250 (5)	6.88, 250 (5)	5.87, 250 (5)	6.3, 250 (5)
LMaFit, time(s.), #its.	3.61, 250 (5)	3.56, 250 (5)	3.71, 250 (5)	*3.55*, 250 (5)	*3.74*, 250 (5)
ScGrassMC, time(s.), #its.	0.21, 6.8 (5)	0.27, 7 (5)	0.27, 6.2 (5)	0.31, 7.4 (5)	0.35, 7.6 (5)
ScGrassMC (EL), time(s.), #its.	8.26, 249 (5)	10.01, 249 (5)	8.33, 249 (5)	9.19, 249 (5)	8.37, 249 (5)
$\mathbf{UBV}^T$, time(s.), #its.	6.44, 250 (5)	6.51, 250 (5)	7.58, 250 (5)	6.47, 250 (5)	6.87, 250 (5)
$\mathbf{UY}^T$, time(s.), #its.	5.94, 250 (5)	5.97, 250 (5)	7.63, 250 (5)	5.87, 250 (5)	6.29, 250 (5)

cost on well-conditioned low-rank matrix completion problems (rank $=$ 10) from small-scale to large-scale, which verifies that RiemSScRNCG is more efficient than other state-of-the-art algorithms, and it is attribute to consider the Riemannian geometry structure behind the full-rank matrix factorization model and the scaling information derived from the hessian information of the cost function of fixed low-rank matrix completion problem simultaneously in RiemSScRNCG.

Third, for the ill-conditioned random low-rank matrices, let $\hat{\mathbf{U}} = \text{qf}(\mathbf{M}_L)$ and $\hat{\mathbf{V}} = \text{qf}(\mathbf{M}_R)$ be semi-orthogonal matrices, where qf($\bullet$) denotes the Q factor in the QR factorization, and $\hat{\Sigma}$ is a $r \times r$ diagonal matrix whose diagonal elements are the singular values sorted by descending order. Specifically, the singular values obey exponential decay and are obtained by $\hat{\Sigma} = 100 * \text{diag}(\text{logspace}(-\log 10(\text{CN}), 0, r))$ in MATLAB, where CN is given a large number. Thus, the matrix $\hat{\mathbf{X}} = \hat{\mathbf{U}}\hat{\Sigma}\hat{\mathbf{V}}^T$ is a random ill-conditioned matrix with rank r. In this experiment, CN is 200, and the detailed numerical results are reported in Table II.

From Table II, we find that most algorithms except RiemSScRNCG and qGeomMC are unable to obtain the global optimum when the maximum number of iterations is arrived. It illustrates that the scaling information of the cost function of the specific problem has played an important role in obtaining an optimal solution and the good convergence for ill-conditioned problems, because it can be interpreted as an adaptive preconditioning step. Thus, RiemSScRNCG and qGeomMC which consider the scale information can obtain the global optimum for most problems. But, we can still see that RiemSScRNCG outperforms qGeomMC and obtains the best performance, which indicates combining Riemannian structure with scaling information is better than only using scaling information. Although ScGrassMC and ScGrassMC(EL) also consider preconditioning step, they are unable to get the optimum solution of problems due to use the Euclidean metric, which also indicates that the Riemannian metric is more suitable and effective than the Euclidean metric. Additionally, the huge-scale well-conditioned fixed low-rank matrix completion problem and the large-scale ill-conditioned fixed low-rank matrix completion problem are also experimented, and the experimental results illustrates that the proposed method can obtain the best performance compared with other methods, detailedly shown in the supplementary material.

For our proposed Riemannian metric, it is constructed by combining the Riemannian geometry structure and the scaling information based on a parameter α, and we also find using a proper parameter enables RiemSScRNCG to obtain a better convergence performance in experiments. Thus, finally, we make an analysis for the parameter α ($\alpha \in [0, 1]$) in this part, where α is given $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. In addition, qGeomMC and Right inv. $\mathbf{GH}^T$ are also implemented in this experiment, because they are essentially equivalent to two special cases of RiemSScRNCG, corresponding to $\alpha = 0$ and $\alpha = 1$, respectively. Fig. 4 shows experimental results of the large-scale well-conditioned problem and the small-scale ill-conditioned problem, respectively, and a clearer version is shown in the supplementary material.

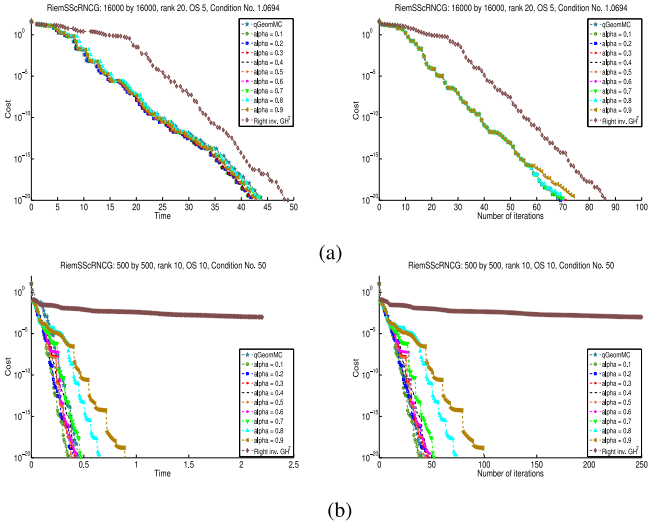

Fig. 4. Convergence behaviors of RiemSScRNCG with the different parameters for fixed low-rank matrix completion. (a) Large-scale well-conditioned problem. (b) Small-scale ill-conditioned problem.

From Fig. 4(a), it is obviously seen that the convergence behavior of RiemSScRNCG is superior to qGeomMC and Right inv. $\mathbf{GH}^T$, when $\alpha = \{0.2, 0.3, 0.4\}$. In Fig. 4(b), when $\alpha = \{0.1, 0.2, 0.3\}$, RiemSScRNCG is superior to other two methods. In experiments, we find that the higher the weight value of scaling information is than the Riemannian geometry structure, the more it will improve convergence efficiency for ill-conditioned problems, which means a small α is better. Meanwhile, the Riemannian geometry structure is also not ignored. For well-conditioned problems, the requirement of the scaling information is not more urgent than the former. Hence, in order to simplify the parameter choice, $\alpha = 0.1$ is adopted

TABLE III
MEAN RESULTS FOR TEN RANDOM INDEPENDENT PARTITIONS ON FOUR DATASETS

Dataset	Epinions			MovieLens-10M		
	RMSE	MAE	Time(s)	RMSE	MAE	Time(s)
RiemSScRNCG	**0.326±0.001**	**0.170±0.001**	*67.9±5.4*	**0.971±0.013**	**0.716±0.011**	**59.6±8.2**
qGeomMC	0.332±0.002	0.174±0.002	94.3±7.0	*0.984±0.012*	*0.728±0.007*	*66.9±11.8*
Right inv.$\mathbf{GH}^T$	0.357±0.004	0.197±0.003	93.4±7.0	1.079±0.009	0.788±0.004	83.8±8.7
LRGeomCG	*0.331±0.003*	*0.172±0.002*	108.1±7.6	1.011±0.006	0.743±0.005	87.7±13.9
LMaFit	0.778±0.007	0.539±0.006	**52.7±13.3**	1.028±0.002	0.756±0.002	111.4±2.0
ScGrassMC	0.454±0.042	0.276±0.037	95.1±17.5	1.049±0.023	0.765±0.013	93.4±12.7
ScGrassMC(EL)	0.383±0.041	0.205±0.024	186.8±21.9	1.011±0.006	0.743±0.005	251.8±8.8
$\mathbf{UBV}^T$	0.342±0.003	0.184±0.002	155.9±12.3	1.111±0.003	0.810±0.002	112.6±5.3
$\mathbf{UY}^T$	0.353±0.003	0.194±0.002	126.3±10.3	1.086±0.009	0.792±0.005	82.3±12.2
Dataset	Jester-All			Netflix		
	RMSE	MAE	Time(s)	RMSE	MAE	Time(s)
RiemSScRNCG	**4.319±0.025**	**2.875±0.010**	**72.6±6.0**	**0.950±0.034**	**0.716±0.019**	*1788.3±50.3*
qGeomMC	*4.353±0.001*	2.918±0.003	152.1±18.8	1.065±0.061	0.785±0.035	2668.1±72.3
Right inv.$\mathbf{GH}^T$	*4.353±0.001*	*2.917±0.001*	*107.4±21.5*	1.170±0.086	0.855±0.048	3225.3±134.2
LRGeomCG	4.556±0.003	3.021±0.003	168.1±19.1	1.086±0.126	0.756±0.038	3448.4±131.8
LMaFit	4.754±0.002	3.119±0.002	117.1±0.5	1.141±0.022	0.829±0.009	3195.1±17.3
ScGrassMC	4.660±0.003	3.084±0.003	151.4±21.3	1.484±0.188	1.024±0.092	**1219.4±33.8**
ScGrassMC(EL)	4.655±0.003	3.072±0.003	221.2±41.4	*0.994±0.065*	*0.736±0.042*	6961.6±184.1
$\mathbf{UBV}^T$	4.685±0.004	3.093±0.004	143.2±27.8	1.099±0.043	0.813±0.023	3209.9±79.1
$\mathbf{UY}^T$	4.683±0.002	3.089±0.002	135.9±33.1	1.128±0.099	0.825±0.051	2692.2±65.7

for ill-conditioned synthetic datasets, real-word recommendation datasets, image and video datasets, and $\alpha = 0.2$ is adopted for well-conditioned synthetic datasets in our experiments.

B. Experiments on Real-World Recommendation System

In this part, four real-world large-scale collaborative filtering datasets are applied to verify the performance of RiemSScRNCG.

1) The Epinions Dataset[19] [49], it comes from the Epinions.com website and contains 664 824 reviews from 49 290 users on 139 738 different items.
2) The MovieLens-10M Dataset,[20] it comes from the online movie recommender service MovieLens and contains 10 000 054 ratings from 71 567 users on 10 681 movies.
3) The Jester-All Dataset,[21] it derives from the Jester Joke Recommender System and contains 4.1 million continuous ratings of 100 jokes from 73 496 users.
4) The Netflix Prize Dataset,[22] it contains 100 480 507 ratings from 480 189 users on 17 770 movies.

In real-world problems, the datasets are very noisy. Unlike noiseless cases in which the value of the cost function is as the stopping criterion, we terminate the iteration via a small relative residual $\|\mathcal{P}_\Omega(\mathbf{X}) - \mathcal{P}_\Omega(\hat{\mathbf{X}})\|_F \leq \varepsilon_1 \|\mathcal{P}_\Omega(\hat{\mathbf{X}})\|_F$ here. However, while the matrix that will be recovered is not exact low-rank, the small relative residual is not adequate. According to the Armijo convergence theorem [50], Armijo condition [51], and the convergent analysis of Riemannian conjugate gradient method [52], it is known that RiemSScRNCG monotonically decreases in the value of cost function (22). Thus, we also provide a difference in cost function values as the stopping condition, $|\Omega|(f(\mathbf{X}_i) - f(\mathbf{X}_{i+1})) \leq \varepsilon_2 \|\mathcal{P}_\Omega(\hat{\mathbf{X}})\|_F^2$, where $f(\mathbf{X}_i) = (1/|\Omega|)\|\mathcal{P}_\Omega(\mathbf{X}_i) - \mathcal{P}_\Omega(\hat{\mathbf{X}})\|_F^2$, and ε_1 and ε_2 are the predefined tolerances. In the performance comparison, the root-mean-square error (RMSE) and the mean absolute error (MAE) of the testing dataset are used as the criterions in real-world applications, defined as $\text{RMSE} = (1/\sqrt{|\Omega_{\text{test}}|})\|\mathcal{P}_{\Omega_{\text{test}}}(\mathbf{X}^*) - \mathcal{P}_{\Omega_{\text{test}}}(\hat{\mathbf{X}})\|_F$ and $\text{MAE} = (1/|\Omega_{\text{test}}|)\sum |\mathcal{P}_{\Omega_{\text{test}}}(\mathbf{X}^*_{ij}) - \mathcal{P}_{\Omega_{\text{test}}}(\hat{\mathbf{X}}_{ij})|$, where $\hat{\mathbf{X}}$ denotes the original matrix, $\mathcal{P}_{\Omega_{\text{test}}}(\hat{\mathbf{X}})$ indicates the independent observed entries for testing, $\mathbf{X}^*$ is the recovered matrix, and Ω_{test} expresses the index set of the testing partition of observed entries, $i, j \in \Omega_{\text{test}}$. The training and testing partitions of the datasets are divided similarly to [21], [53], and [54]. In this experiment, the rank of the recovered matrix $\mathbf{X}^*$ is fixed at 10, ε_1 is set to 10^{-5}, ε_2 is set to 10^{-6}, the maximum number of iterations is 500, and $\alpha = 0.1$ for RiemSScRNCG.

In Table III, we report the RMSE and MAE of nine algorithms over ten random independent 80/20 training/testing partitions. Obviously, it is seen that RiemSScRNCG performs best in terms of RMSE, MAE, and the computational time compared with other methods, and it is inferior to LMaFit and ScGrassMC only on the computational time of Epinions and Netflix, respectively. We also find that RMSE and MAE for Jester-All are higher than other three datasets, which indicates that the Jester-All problem may not be a good test problem for low-rank matrix completion, because the matrix to be recovered has only 100 columns to begin with. However, Right inv. $\mathbf{GH}^T$ gains a better performance than other methods except RiemSScRNCG for Jester-All, which illustrates the metric with Riemannian structure plays an important role in optimization on this dataset. Thereby, RiemSScRNCG can gain the best performance for Jester-All. Additionally, we find that LMaFit has the least computational cost on the Epinions, but its performance is worse than others, and this phenomenon also emerges on the Netflix for ScGrassMC, which is caused because they may fall into the local minimum.

C. Experiments on Image and Video Recovery

In this part, we apply nine algorithms for grayscale image and color video recovery (similar to [10] and [19]) to verify

[19] http://www.trustlet.org/wiki/Downloaded_Epinions_dataset
[20] http://www.grouplens.org/datasets/movielens/
[21] http://www.ieor.berkeley.edu/~goldberg/jester-data/
[22] http://archive.ics.uci.edu/ml/noteNetflix.txt

TABLE IV
IMAGE RECOVERY RESULTS [PSNR (TIME COST/S)] MEASURED BY THE PEAK SIGNAL-TO-NOISE RATIO (dB)

Dataset	Barbara	Boat	Cameraman	Lena	Man	Mandrill	Peppers
RiemSScRNCG	**29.4760** (**9.99**)	**32.3515** (*18.58*)	*29.1849* (**15.24**)	**32.4936** (**16.27**)	**31.7361** (*14.05*)	*26.1723* (**14.70**)	**31.1888** (**15.23**)
qGeomMC	28.5695 (15.61)	*32.3180* (22.21)	**29.3723** (17.84)	31.8507 (*17.59*)	31.6382 (18.33)	**26.1740** (19.92)	30.5238 (17.43)
Right inv.$\mathbf{GH}^T$	19.4972 (18.44)	18.6133 (21.71)	18.6733 (20.19)	20.9321 (21.71)	21.0616 (21.70)	18.4265 (20.28)	20.1787 (21.17)
SVP	23.1875 (116.4)	23.5383 (187.5)	20.8690 (112.1)	25.3524 (111.7)	26.7258 (115.4)	20.3902 (130.1)	23.7362 (112.2)
LRGeomCG	29.1203 (24.38)	30.5971 (28.84)	23.3726 (27.19)	26.7596 (27.85)	30.1432 (27.80)	25.9460 (28.04)	29.9280 (27.15)
LMaFit	*29.2987* (*12.23*)	29.3577 (**17.94**)	24.3598 (*16.56*)	*32.2437* (18.35)	31.1765 (**12.27**)	26.1668 (*16.52*)	*30.6955* (*16.45*)
ScGrassMC	28.5813 (18.45)	26.2918 (20.12)	23.0996 (21.57)	29.6883 (22.76)	28.7220 (24.12)	25.8415 (20.65)	28.3067 (21.75)
ScGrassMC(EL)	28.7927 (28.51)	27.9456 (36.76)	27.7354 (31.28)	30.1075 (36.11)	*31.7355* (33.28)	26.1132 (31.43)	27.4211 (33.28)
$\mathbf{UY}^T$	19.3221 (23.66)	20.2206 (29.29)	20.3063 (27.83)	20.5302 (28.32)	20.7380 (28.59)	18.9592 (27.15)	18.7470 (27.43)

TABLE V
IMAGE INPAINTING RESULTS [RERR (TIME COST/S)] MEASURED BY RELATIVE ERROR

Dataset	Barbara	Boat	Cameraman	Lena	Man	Mandrill	Peppers
RiemSScRNCG	**1.89e-04** (**13.36**)	**5.12e-03** (*21.01*)	**5.90e-03** (**22.65**)	**1.79e-03** (**22.20**)	*7.14e-04* (*20.66*)	**3.11e-04** (**17.71**)	**3.72e-03** (**19.98**)
qGeomMC	*2.89e-04* (20.43)	1.32e-02 (24.31)	*2.96e-02* (28.78)	9.85e-03 (29.09)	2.34e-03 (26.98)	3.35e-03 (27.70)	3.09e-02 (25.98)
Right inv.$\mathbf{GH}^T$	1.98e-01 (29.91)	1.81e-01 (29.83)	2.12e-01 (29.27)	1.62e-01 (29.01)	1.97e-01 (30.27)	1.83e-01 (35.97)	2.58e-01 (30.69)
SVP	1.15e-02 (172.6)	1.26e-01 (163.5)	1.28e-01 (166.6)	1.75e-01 (165.3)	9.80e-02 (171.4)	2.89e-02 (262.3)	1.41e-01 (243.2)
LRGeomCG	2.93e-04 (17.19)	*8.66e-03* (35.59)	6.19e-02 (34.96)	2.43e-02 (27.85)	**6.86e-04** (31.08)	*4.72e-04* (25.94)	*4.88e-03* (36.21)
LMaFit	2.92e-04 (*15.24*)	2.31e-02 (**18.97**)	6.60e-02 (*23.10*)	8.32e-02 (*23.35*)	3.52e-03 (**18.61**)	7.52e-04 (*20.76*)	3.22e-02 (*22.28*)
ScGrassMC	1.34e-01 (26.84)	1.14e-01 (28.30)	1.12e-01 (27.23)	1.77e-01 (28.57)	1.71e-02 (24.50)	4.12e-02 (23.22)	1.12e-01 (25.67)
ScGrassMC(EL)	4.02e-04 (48.68)	5.42e-02 (40.64)	7.23e-02 (41.52)	8.82e-02 (48.93)	4.23e-03 (35.55)	3.66e-03 (36.12)	3.31e-02 (38.23)
$\mathbf{UY}^T$	1.86e-01 (36.91)	2.05e-01 (37.49)	1.91e-01 (34.54)	1.70e-01 (34.43)	2.10e-01 (34.22)	1.87e-01 (33.38)	1.92e-01 (38.05)

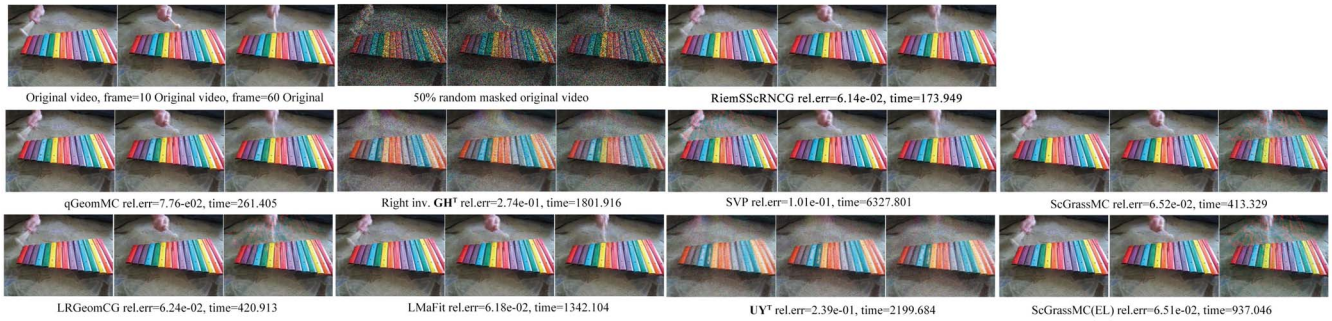

Fig. 5. Results of videos recovered by different algorithms on the *Xylophone* (measured by rel.err and time).

the performance. The task is to complete the missing pixel values of an image or a video at given pixel values whose positions have been determined. Specifically, when the missing pixel positions are not randomly distributed, it is also called inpainting. If the image or video is of low-rank or numerical low-rank, that problem can be solved as a matrix completion problem. The relative error of recovered image or video is given by rel.err $= \|\mathbf{X}^* - \hat{\mathbf{X}}\|_F / \|\hat{\mathbf{X}}\|_F$, and the stopping tolerances are the same as in Section VI-B. In experiments, we set $\alpha = 0.1$ for RiemSScRNCG. For the image recovery, the maximum number of iterations is set to 300, and it is set to 250 for the video recovery. Note that SVP is added instead of $\mathbf{UBV}^T$ in this experiment, because $\mathbf{UBV}^T$ falls into a serious numerical problem on image and video recovery.

For the image recovery, we use the benchmark images[23] which include *Barbara*, *Boat*, *Cameraman*, *Lena*, *Man*, *Mandrill*, and *Peppers*, and the size of each image is 512×512. In experiments, two situations are considered: one is the general image recovery, where the image is randomly masked on 50% pixels and the remaining ones are applied as the observations; the other is inpainting, where the benchmark image is masked nonrandomly and the number of affected pixels is seldom large. We perform ten random independent instances and obtain the mean of the peak signal-to-noise ratio (PSNR) and the relative error (RERR). Table IV presents the numerical results of image recovery in terms of the PSNR and the mean costs of time, and Table V presents the numerical results of image inpainting in terms of the relative error and the mean costs of time.

From Tables IV and V, it is obviously seen that RiemSScRNCG can obtain better performance for most test images than other compared methods, while only qGeomMC gains the best results for *Cameraman* and *Man* images in Table IV and only LRGeomCG gains the best results for *Man* images in Table V. Actually, RiemSScRNCG also gains the second lowest error which is only slightly worse than qGeomMC and LRGeomCG for *Cameraman* and *Man* images. Moreover, the results of time cost also illustrate that RiemSScRNCG can obtain lower computational cost on most images than others. In compared methods, only LMaFit has a better computational efficiency in *Boat* and *Man* images than RiemSScRNCG.

For the color video recovery, we use the benchmark video sample which comes from the MATLAB Image Processing Toolbox, shortened from the *Xylophone* in the following parts. In fact, each frame of the video is represented by an image

[23]http://www.utdallas.edu/%7Ecxc123730/mh_bcs_spl.html

stored in the RGB format, and thus the *Xylophone* video is reshaped into a $76\,800 \times 423$ matrix in our experiment. In our experiment, 50% of the entries of the *Xylophone* are randomly marked, and other entries are used for the observations. In addition, the rank of the recovered matrix $\mathbf{X}^*$ is fixed at 80. Fig. 5 shows the results of the color video frames recovered by nine algorithms on the *Xylophone* video, where the numerical performance of each algorithm is shown at the bottom of its picture. Visually, it is seen that RiemSScRNCG and other algorithms (except Right inv. $\mathbf{G}\mathbf{H}^T$ and $\mathbf{U}\mathbf{Y}^T$) can not only restore the low-rank part (static part, such as background) of the video but also recognize the detailed part (moving part, such as hands) while recovering the low-rank part of video quite successfully. But, RiemSScRNCG obtains the best numerical performance (the lowest error $6.14e^{-2}$ and the least time 173.949 s) and computational efficiency compared with others, when it gains the roughly same visual effect as qGeomMC and LMaFit.

VII. Conclusion

Inspired by the deficiency derived from considering only the Riemannian geometry structure or the scaling information, we propose a new Riemannian metric by simultaneously considering the Riemannian geometry structure and the scaling information on the horizontal subspace of the quotient manifold for fixed low-rank matrix completion in this paper. Meanwhile, based on this metric, we also propose an efficient algorithm under the framework of Riemannian nonlinear conjugate gradient method. Experimental results illustrates the proposed method outperforms the compared state-of-the-art methods, which demonstrates that simultaneously considering them is more effective than only considering one. By using our proposed method, more general practical problems can be solved to obtain better performance with more efficient way. Moreover, our method can be expanded to Riemannian similarity learning.

Although it is validated that the proposed metric can obtain better performance, there is still several worthy studied issues starting from the proposed method in the future. For examples, how to adaptively choose a proper parameter α, how to develop the second order optimization algorithm on the Riemannian quotient manifold equipped with the proposed Riemannian metric, how to construct a new metric with both the Riemannian geometry and the scaling information based on a nonlinear way, and how to combine discriminative information.

Appendix

Proof of Proposition 1

Proof: Given an arbitrary smooth function $f : \mathbb{R}_*^{m\times r} \times \mathbb{R}_*^{n\times r}/\mathrm{GL}_r \to \mathbb{R}$ on the Riemannian quotient manifold $\mathcal{X}$, by using the quotient map π, a new function $\bar{f} = f \circ \pi : \mathbb{R}_*^{m\times r} \times \mathbb{R}_*^{n\times r} \to \mathbb{R}$ on the total space $\bar{\mathcal{X}}$ is also smooth. Reviewing the mapping (8) mentioned in Section II, we have $l : (\mathbf{G}, \mathbf{H}) \to (\mathbf{G}\mathbf{D}^{-1}, \mathbf{H}\mathbf{D}^T), \forall \mathbf{D} \in \mathrm{GL}_r$. By virtual of $\pi(l(\mathbf{G}, \mathbf{H})) = \pi(\mathbf{G}, \mathbf{H})$ for any $(\mathbf{G}, \mathbf{H})$, we get $\bar{f}(l(\mathbf{G}, \mathbf{H})) = \bar{f}(\mathbf{G}, \mathbf{H})$ for any $(\mathbf{G}, \mathbf{H})$. Let us take the differential of both sides

$$\mathrm{D}\bar{f}(l(\mathbf{G}, \mathbf{H}))\big[\mathrm{D}l(\mathbf{G}, \mathbf{H})\big[\big(\bar{\zeta}_{\mathbf{G}}, \bar{\zeta}_{\mathbf{H}}\big)\big]\big] = \mathrm{D}\bar{f}(\mathbf{G}, \mathbf{H})\big[\big(\bar{\zeta}_{\mathbf{G}}, \bar{\zeta}_{\mathbf{H}}\big)\big]. \tag{24}$$

From the definition of $(\bar{\zeta}_{\mathbf{G}}, \bar{\zeta}_{\mathbf{H}})$, we know

$$\mathrm{D}\bar{f}(\mathbf{G}, \mathbf{H})\big[\big(\bar{\zeta}_{\mathbf{G}}, \bar{\zeta}_{\mathbf{H}}\big)\big] = \mathrm{D}f(\pi(\mathbf{G}, \mathbf{H}))\big[\zeta_{\pi([\mathbf{G},\mathbf{H}])}\big].$$

Furthermore, under the known condition of Proposition 1, we have

$$\mathrm{D}l(\mathbf{G}, \mathbf{H})\big[\big(\bar{\zeta}_{\mathbf{G}}, \bar{\zeta}_{\mathbf{H}}\big)\big] = \Big(\bar{\zeta}_{\mathbf{G}}\mathbf{D}^{-1}, \bar{\zeta}_{\mathbf{H}}\mathbf{D}^T\Big), \ \forall \mathbf{D} \in \mathrm{GL}_r.$$

Then, we are going to plug them into (24) and obtain

$$\begin{aligned}&\mathrm{D}\bar{f}\Big(\mathbf{G}\mathbf{D}^{-1}, \mathbf{H}\mathbf{D}^T\Big)\Big[\Big(\bar{\zeta}_{\mathbf{G}}\mathbf{D}^{-1}, \bar{\zeta}_{\mathbf{H}}\mathbf{D}^T\Big)\Big]\\ &\qquad = \mathrm{D}f\Big(\pi\Big(\mathbf{G}\mathbf{D}^{-1}, \mathbf{H}\mathbf{D}^T\Big)\Big)\big[\zeta_{\pi([\mathbf{G},\mathbf{H}])}\big].\end{aligned}$$

This equation is hold for any smooth function f, we educe that

$$\mathrm{D}\pi\Big(\mathbf{G}\mathbf{D}^{-1}, \mathbf{H}\mathbf{D}^T\Big)\Big[\Big(\bar{\zeta}_{\mathbf{G}}\mathbf{D}^{-1}, \bar{\zeta}_{\mathbf{H}}\mathbf{D}^T\Big)\Big] = \zeta_{\pi([\mathbf{G},\mathbf{H}])}.$$

By the definition of horizontal lift, we find that $(\bar{\zeta}_{\mathbf{G}}\mathbf{D}^{-1}, \bar{\zeta}_{\mathbf{H}}\mathbf{D}^T)$ is the unique horizontal lift of $\zeta_{\pi([\mathbf{G},\mathbf{H}])}$ at $(\mathbf{G}\mathbf{D}^{-1}, \mathbf{H}\mathbf{D}^T)$. Specifically, by applying the Lemma 2 at point $(\mathbf{G}\mathbf{D}^{-1}, \mathbf{H}\mathbf{D}^T)$ and the horizontal tangent vector $(\bar{\zeta}_{\mathbf{G}}\mathbf{D}^{-1}, \bar{\zeta}_{\mathbf{H}}\mathbf{D}^T)$, we have

$$\begin{aligned}&\Big(\mathbf{G}\mathbf{D}^{-1}\Big)^T\bar{\zeta}_{\mathbf{G}}\mathbf{D}^{-1}\Bigg(\alpha\Bigg(\Big(\mathbf{G}\mathbf{D}^{-1}\Big)^T\Big(\mathbf{G}\mathbf{D}^{-1}\Big)\Bigg)^{-1}\Bigg)\\ &\qquad + \Big(\mathbf{G}\mathbf{D}^{-1}\Big)^T\bar{\zeta}_{\mathbf{G}}\mathbf{D}^{-1}\Big((1-\alpha)\Big(\big(\mathbf{H}\mathbf{D}^T\big)^T\Big)\big(\mathbf{H}\mathbf{D}^T\big)\Big)\\ &\quad = \mathbf{D}^{-T}\mathbf{G}^T\bar{\zeta}_{\mathbf{G}}\Big(\alpha\big(\mathbf{G}^T\mathbf{G}\big)^{-1} + (1-\alpha)\mathbf{H}^T\mathbf{H}\Big)\mathbf{D}^T\\ &\qquad - \Big((1-\alpha)\Big(\mathbf{G}\mathbf{D}^{-1}\Big)^T\Big(\mathbf{G}\mathbf{D}^{-1}\Big)\Big)\Big(\bar{\zeta}_{\mathbf{H}}\mathbf{D}^{\mathbf{T}}\Big)^T\mathbf{H}\mathbf{D}^T\\ &\qquad -\Bigg(\alpha\Big(\big(\mathbf{H}\mathbf{D}^T\big)^T\mathbf{H}\mathbf{D}^T\Big)^{-1}\Bigg)\big(\bar{\zeta}_{\mathbf{H}}\mathbf{D}^T\big)^T\mathbf{H}\mathbf{D}^T\\ &\quad = -\mathbf{D}^{-T}\Big((1-\alpha)\mathbf{G}^T\mathbf{G} + \alpha\big(\mathbf{H}^T\mathbf{H}\big)^{-1}\Big)\bar{\zeta}_{\mathbf{H}}^T\mathbf{H}\mathbf{D}^T.\end{aligned}$$

Since

$$\begin{aligned}&\mathbf{D}^{-T}\mathbf{G}^T\bar{\zeta}_{\mathbf{G}}\Big(\alpha\big(\mathbf{G}^T\mathbf{G}\big)^{-1} + (1-\alpha)\mathbf{H}^T\mathbf{H}\Big)\mathbf{D}^T\\ &\qquad = -\mathbf{D}^{-T}\Big((1-\alpha)\mathbf{G}^T\mathbf{G} + \alpha\big(\mathbf{H}^T\mathbf{H}\big)^{-1}\Big)\bar{\zeta}_{\mathbf{H}}^T\mathbf{H}\mathbf{D}^T\end{aligned}$$

then we get

$$\begin{aligned}&\mathbf{G}^T\bar{\zeta}_{\mathbf{G}}\Big(\alpha\big(\mathbf{G}^T\mathbf{G}\big)^{-1} + (1-\alpha)\mathbf{H}^T\mathbf{H}\Big)\\ &\qquad = -\Big((1-\alpha)\mathbf{G}^T\mathbf{G} + \alpha\big(\mathbf{H}^T\mathbf{H}\big)^{-1}\Big)\bar{\zeta}_{\mathbf{H}}^T\mathbf{H}.\end{aligned}$$

Obviously, $(\bar{\zeta}_{\mathbf{G}}\mathbf{D}^{-1}, \bar{\zeta}_{\mathbf{H}}\mathbf{D}^T)$ is the unique horizontal lift of $\zeta_{\pi([\mathbf{G},\mathbf{H}])}$ at $(\mathbf{G}\mathbf{D}^{-1}, \mathbf{H}\mathbf{D}^T)$. Using the known condition

$$\big(\bar{\zeta}_{\mathbf{G}\mathbf{D}^{-1}}, \bar{\zeta}_{\mathbf{H}\mathbf{D}^T}\big) = \Big(\bar{\zeta}_{\mathbf{G}}\mathbf{D}^{-1}, \bar{\zeta}_{\mathbf{H}}\mathbf{D}^T\Big), \ \forall \mathbf{D} \in \mathrm{GL}_r$$

and the above results, we have

$$\begin{aligned}&\bar{g}^{\mathrm{addi}}_{(\mathbf{G},\mathbf{H})}\big(\big(\bar{\zeta}_{\mathbf{G}}, \bar{\zeta}_{\mathbf{H}}\big), \big(\bar{\eta}_{\mathbf{G}}, \bar{\eta}_{\mathbf{H}}\big)\big)\\ &\qquad = \bar{g}^{\mathrm{addi}}_{(\mathbf{G}\mathbf{D}^{-1},\mathbf{H}\mathbf{D}^T)}\big(\big(\bar{\zeta}_{\mathbf{G}\mathbf{D}^{-1}}, \bar{\zeta}_{\mathbf{H}\mathbf{D}^T}\big), \big(\bar{\eta}_{\mathbf{G}\mathbf{D}^{-1}}, \bar{\eta}_{\mathbf{H}\mathbf{D}^T}\big)\big).\end{aligned}$$

■

REFERENCES

[1] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, 2009.

[2] M. Fazel, "Matrix rank minimization with applications," Ph.D. dissertation, Dept. Elect. Eng., Stanford Univ., Stanford, CA, USA, 2002.

[3] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," *SIAM Rev.*, vol. 52, no. 3, pp. 471–501, 2010.

[4] R. Meka, P. Jain, and I. S. Dhillon, "Guaranteed rank minimization via singular value projection," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2010, pp. 937–945.

[5] K. Lee and Y. Bresler, "ADMiRA: Atomic decomposition for minimum rank approximation," *IEEE Trans. Inf. Theory*, vol. 56, no. 9, pp. 4402–4416, Sep. 2010.

[6] J. P. Haldar and D. Hernando, "Rank-constrained solutions to linear matrix equations using powerfactorization," *IEEE Signal Process. Lett.*, vol. 16, no. 7, pp. 584–587, Jul. 2009.

[7] X. Lu, T. Gong, P. Yan, Y. Yuan, and X. Li, "Robust alternative minimization for matrix completion," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 3, pp. 939–949, Jun. 2012.

[8] R. Meka, P. Jain, C. Caramanis, and I. S. Dhillon, "Rank minimization via online learning," in *Proc. 25th Int. Conf. Mach. Learn.*, Helsinki, Finland, 2008, pp. 656–663.

[9] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM J. Optim.*, vol. 20, no. 4, pp. 1956–1982, 2010.

[10] S. Ma, D. Goldfarb, and L. Chen, "Fixed point and Bregman iterative methods for matrix rank minimization," *Math. Program.*, vol. 128, nos. 1–2, pp. 321–353, 2011.

[11] K.-C. Toh and S. Yun, "An accelerated proximal gradient algorithm for nuclear norm regularized linear least squares problems," *Pac. J. Optim.*, vol. 6, no. 3, pp. 615–640, 2010.

[12] Y. Hu, D. Zhang, J. Ye, X. Li, and X. He, "Fast and accurate matrix completion via truncated nuclear norm regularization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 9, pp. 2117–2130, Sep. 2013.

[13] R. Cabral, F. De la Torre, J. P. Costeira, and A. Bernardino, "Matrix completion for weakly-supervised multi-label image classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 1, pp. 121–135, Jan. 2015.

[14] K. Li, J. Yang, and J. Jiang, "Nonrigid structure from motion via sparse representation," *IEEE Trans. Cybern.*, vol. 45, no. 8, pp. 1401–1413, Aug. 2015.

[15] M. Fazel, E. Candes, B. Recht, and P. Parrilo, "Compressed sensing and robust recovery of low rank matrices," in *Proc. 42nd Asilomar Conf. Signals Syst. Comput.*, Pacific Grove, CA, USA, 2008, pp. 1043–1047.

[16] Y. Liu, F. Shang, L. Jiao, J. Cheng, and H. Cheng, "Trace norm regularized CANDECOMP/PARAFAC decomposition with missing data," *IEEE Trans. Cybern.*, vol. 45, no. 11, pp. 2437–2448, Nov. 2015.

[17] F. R. Bach, "Consistency of trace norm minimization," *J. Mach. Learn. Res.*, vol. 9, pp. 1019–1048, Jun. 2008.

[18] R. H. Keshavan and S. Oh, "OPTSPACE: A gradient descent algorithm on the Grassman manifold for matrix completion," *arXiv:0910.5260*, 2009.

[19] Z. Wen, W. Yin, and Y. Zhang, "Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm," *Math. Program. Comput.*, vol. 4, no. 4, pp. 333–361, 2012.

[20] J. D. M. Rennie and N. Srebro, "Fast maximum margin matrix factorization for collaborative prediction," in *Proc. 22nd Int. Conf. Mach. Learn.*, Bonn, Germany, 2005, pp. 713–719.

[21] Y. Liu, L. C. Jiao, and F. Shang, "A fast tri-factorization method for low-rank matrix recovery and completion," *Pattern Recognit.*, vol. 46, no. 1, pp. 163–173, 2013.

[22] A. M. Buchanan and A. W. Fitzgibbon, "Damped Newton algorithms for matrix factorization with missing data," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2. San Diego, CA, USA, 2005, pp. 316–322.

[23] P. Chen, "Optimization algorithms on subspaces: Revisiting missing data problem in low-rank matrix," *Int. J. Comput. Vis.*, vol. 80, no. 1, pp. 125–142, 2008.

[24] R. Hartley and F. Schaffalitzky, "Powerfactorization: 3D reconstruction with missing or uncertain data," in *Proc. Aust. Japan Adv. Workshop Comput. Vis.*, vol. 74. 2003, pp. 76–85.

[25] R. Vidal and R. Hartley, "Motion segmentation with missing data using powerfactorization and GPCA," in *Proc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2. Washington, DC, USA, 2004, pp. II-310–II-316.

[26] R. Vidal, R. Tron, and R. Hartley, "Multiframe motion segmentation with missing data using powerfactorization and GPCA," *Int. J. Comput. Vis.*, vol. 79, no. 1, pp. 85–105, 2008.

[27] K. Li, Q. Dai, W. Xu, J. Yang, and J. Jiang, "Three-dimensional motion estimation via matrix completion," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 2, pp. 539–551, Apr. 2012.

[28] R. Ma, N. Barzigar, A. Roozgard, and S. Cheng, "Decomposition approach for low-rank matrix completion and its applications," *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 1671–1683, Apr. 2014.

[29] J.-H. Kim, J.-Y. Sim, and C.-S. Kim, "Video deraining and desnowing using temporal correlation and low-rank matrix completion," *IEEE Trans. Image Process.*, vol. 24, no. 9, pp. 2658–2670, Sep. 2015.

[30] Q. Liu, Z. Lai, Z. Zhou, F. Kuang, and Z. Jin, "A truncated nuclear norm regularization method based on weighted residual error for matrix completion," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 316–330, Jan. 2016.

[31] U. Shalit, D. Weinshall, and G. Chechik, "Online learning in the embedded manifold of low-rank matrices," *J. Mach. Learn. Res.*, vol. 13, pp. 429–458, Feb. 2012.

[32] B. Vandereycken, "Low-rank matrix completion by Riemannian optimization," *SIAM J. Optim.*, vol. 23, no. 2, pp. 1214–1236, 2013.

[33] B. Mishra, G. Meyer, S. Bonnabel, and R. Sepulchre, "Fixed-rank matrix factorizations and Riemannian low-rank optimization," *Comput. Stat.*, vol. 29, nos. 3–4, pp. 591–621, 2014.

[34] T. T. Ngo and Y. Saad, "Scaled gradients on Grassmann manifolds for matrix completion," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1412–1420.

[35] B. Mishra, K. A. Apuroop, and R. Sepulchre, "A Riemannian geometry for low-rank matrix completion," *arXiv:1211.1550*, 2012.

[36] G. Meyer, S. Bonnabel, and R. Sepulchre, "Linear regression under fixed-rank constraints: A Riemannian approach," in *Proc. 28th Int. Conf. Mach. Learn.*, Bellevue, WA, USA, 2011, pp. 545–552.

[37] N. Boumal and P.-A. Absil, "RTRMC: A riemannian trust-region method for low-rank matrix completion," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 406–414.

[38] R. H. Keshavan, A. Montanari, and S. Oh, "Matrix completion from a few entries," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2980–2998, Jun. 2010.

[39] Y. Liu, L. C. Jiao, F. Shang, F. Yin, and F. Liu, "An efficient matrix bi-factorization alternative optimization method for low-rank matrix recovery and completion," *Neural Netw.*, vol. 48, pp. 8–18, Dec. 2013.

[40] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton, NJ, USA: Princeton Univ., 2009.

[41] G. Meyer, "Geometric optimization algorithms for linear regression on fixed-rank matrices," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Univ. Liège, Liège, Belgium, 2011.

[42] K. B. Petersen and M. S. Pedersen, *The Matrix Cookbook*, vol. 7, Tech. Univ. Denmark, Kongens Lyngby, Denmark, 2008, p. 15.

[43] J. M. Lee, *Introduction to Smooth Manifolds*. New York, NY, USA: Springer-Verlag, 2001.

[44] R. Mazumder, T. Hastie, and R. Tibshirani, "Spectral regularization algorithms for learning large incomplete matrices," *J. Mach. Learn. Res.*, vol. 11, pp. 2287–2322, Aug. 2010.

[45] Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *arXiv preprint arXiv:1009.5055*, 2010.

[46] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[47] R. M. Larsen. (2004). *Propack-Software for Large and Sparse SVD Calculations*. [Online]. Available: http://sun.stanford.edu/rmunk/PROPACK

[48] P. Drineas, R. Kannan, and M. W. Mahoney, "Fast Monte Carlo algorithms for matrices II: Computing a low-rank approximation to a matrix," *SIAM J. Comput.*, vol. 36, no. 1, pp. 158–183, 2006.

[49] P. Massa and P. Avesani, "Trust-aware bootstrapping of recommender systems," in *Proc. ECAI Workshop Recommender Syst.*, vol. 28. Trento, Italy, 2006, pp. 29–33.

[50] L. Armijo, "Minimization of functions having Lipschitz continuous first partial derivatives," *Pac. J. Math.*, vol. 16, no. 1, pp. 1–3, 1966.

[51] J. Nocedal and S. J. Wright, *Numerical Optimization*. New York, NY, USA: Springer, 2006.
[52] H. Sato and T. Iwai, "A new, globally convergent Riemannian conjugate gradient method," *Optim. J. Math. Program. Oper. Res.*, vol. 64, no. 4, pp. 1011–1031, 2015.
[53] S. Laue, "A hybrid algorithm for convex semidefinite optimization," in *Proc. 29th Int. Conf. Mach. Learn. (ICML)*, Edinburgh, U.K., 2012, pp. 177–184.
[54] M. Tan, I. W. Tsang, L. Wang, B. Vandereycken, and S. J. Pan, "Riemannian pursuit for big matrix recovery," in *Proc. 31st Int. Conf. Mach. Learn. (ICML)*, Beijing, China, 2014, pp. 1539–1547.

Shasha Mao (M'14) received the Ph.D. degree from Xidian University, Xi'an, China, in 2014.

She was a Post-Doctoral Fellow with the Singapore University of Technology and Design, Singapore, from 2014 to 2016. She is currently a Research Fellow with Nanyang Technological University, Singapore. Her current research interests include deep learning, ensemble learning, low-rank and sparse matrix factorization, object detection, and face recognition.

Lin Xiong received the B.S. degree from the Shaanxi University of Science and Technology, Xi'an, China, in 2003, and the Ph.D. degree from the School of Electronic Engineering, Xidian University, Xi'an, in 2015.

He is currently an Engineer of Learning and Vision, Core Technology Group, Panasonic Research and Development Center Singapore, Singapore. His current research interests include Riemannian manifold optimization, low-rank and sparse matrix factorization, background modeling, ensemble learning, and active learning.

Licheng Jiao (M'89) received the B.S. degree from Shanghai Jiaotong University, Shanghai, China, in 1982, and the M.S. and Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 1984 and 1990, respectively.

Since 1992, he has been a Professor with the School of Electronic Engineering, Xidian University, Xi'an, where he is currently the Director of the Key Laboratory of Intelligent Perception and Image Understanding of the Ministry of Education of China. He is in charge of about 40 important scientific research projects and has published over 20 monographs and 100 papers in international journals and conferences at top venues, including the IEEE TRANSACTIONS ON ENERGY CONVERSION, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON SIGNAL PROCESSING, the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING, the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS, and *Pattern Recognition*. His current research interests include image processing, natural computation, nonlinear circuits and systems theory, learning theory and algorithms, optimization problems, wavelet theory, machine learning, and data mining.

Tian Feng received the B.S. degree in geographical information systems from Zhejiang University, Hangzhou, China. He is currently pursuing the Ph.D. degree with the Singapore University of Technology and Design, Singapore.

His current research interests include computer graphics and computer vision in urban modeling and understanding.

Sai-Kit Yeung (M'05) received the Ph.D. degree in electronic and computer engineering from the Hong Kong University of Science and Technology, Hong Kong, in 2009.

He is currently an Assistant Professor with the Pillar of ISTD, Singapore University of Technology and Design, Singapore. His current research interests include computer vision, computer graphics, and image processing.