Constrained mutual convex cone method for image set based recognition

Naoya Sogi[a,*], Rui Zhu[b], Jing-Hao Xue[c], Kazuhiro Fukui[a]

[a] *Graduate School of Systems and Information Engineering, University of Tsukuba, Japan.*
[b] *Faculty of Actuarial Science and Insurance, City, University of London, UK.*
[c] *Department of Statistical Science, University College London, UK*

Abstract

In this paper, we propose convex cone-based frameworks for image-set classification. Image-set classification aims to classify a set of images, usually obtained from video frames or multi-view cameras, into a target object. To accurately and stably classify a set, it is essential to accurately represent structural information of the set. There are various image features, such as histogram-based features and convolutional neural network features. We should note that most of them have non-negativity and thus can be effectively represented by a convex cone. This leads us to introduce the convex cone representation to image-set classification. To establish a convex cone-based framework, we mathematically define multiple angles between two convex cones, and then use the angles to define the geometric similarity between them. Moreover, to enhance the framework, we introduce two discriminant spaces. We first propose a discriminant space that maximizes gaps between cones and minimizes the within-class variance. We then extend it to a weighted discriminant space by introducing weights on the gaps to deal with complicated data distribution. In addition, to reduce the computational cost of the proposed methods, we develop a novel strategy for fast implementation. The effectiveness of the proposed methods is demonstrated experimentally by using five databases.

Keywords: Image-set based method, Convex cone representation, Multiple angles

*Corresponding author
Email address: sogi@cvlab.cs.tsukuba.ac.jp (Naoya Sogi)

1. Introduction

In this paper, we propose a method for image-set classification based on convex cone models, which can exactly represent the geometrical structure of an image set. For the last decade, image-set based classification methods have gained substantial attention in various applications of multi-view images or videos, such as 3D object recognition and motion analysis. The essence of image-set based classification is on how to effectively and low-costly measure the similarity between two image sets. To this end, several types of methods using different models have been proposed [1–9].

Among the conventional methods, subspace-based methods have been well known as effective methods due to the compactness of a subspace model, simple geometrical relationship of class subspaces, and practical and efficient computation. In this type of methods, a set of images is compactly modelled by a subspace in a high-dimensional vector space, where the subspace is generated by applying the principal component analysis (PCA) to the image set without data centering. After converting each image set to a subspace, the similarity between two sets to be compared can be calculated by using the canonical angles between their subspaces [10, 11]. Typical subspace-based methods include the mutual subspace method (MSM) [1] and its extension, the constrained mutual subspace method (CMSM) [12].

The validity of the subspace representation is also supported by the evidence based on physical characteristics. For example, a low-dimensional subspace (with at most nine dimensions) can represent a set of images of a convex object with Lambertian reflectance under a fixed camera view and various illumination conditions. Such a subspace is called the illumination subspace [13–15]. It has been empirically shown that the subspace representation works effectively, even when the above assumptions are not strictly satisfied. In fact, many studies have supported the effectiveness of the subspace representation in various problems [1, 16, 17]. Our representation by using convex cones is an enhanced extension of the subspace representation.

Various image features, such as LBP, HoG and CNN features, have only non-negative values. This characteristic induces the additivity of feature vectors [18]. Furthermore, the additivity allows only the linear combination with non-negative coeffi-

cients of feature vectors. Accordingly, a set of features forms a convex cone instead of a subspace in a high-dimensional vector space, where a convex cone is mathematically defined as a subset of a subspace that is closed under the linear combination with non-negative coefficients. It is well known that a set of front-facing images under various illumination conditions forms a convex cone, referred to as an illumination cone [13–15]. The illumination cone has an advantage over the illumination subspace as it has a more accurate representation ability. Several studies have demonstrated the effectiveness of the convex cone representation compared with that of the subspace representation [18–20]. These advantages naturally motivated us to propose a framework, through replacing a subspace with a convex cone, to model a set of image features with non-negative values.

To incorporate the convex cone model into the framework of image-set recognition, we need to consider how to calculate the structural similarity between two convex cones. To this end, we define multiple angles between two convex cones to capture exactly the geometrical relationship between them, like the canonical angles between two subspaces [10, 11]. We then propose a new method for obtaining the angles in turn from the smallest to the largest by applying the alternating least squares method (ALS) [21] to the convex cones sequentially. Finally, we define the geometric similarity between two convex cones based on the obtained angles. We call the classification method using this similarity the *mutual convex cone method* (MCM).

Furthermore, to improve the performance of the MCM, we introduce the projection of convex cones onto a discriminant space $\mathcal{D}$, which minimizes the within-class variance and enlarges the gaps (between-class variance) between convex cones. The gaps between convex cones precisely capture the difference component between the cones, i.e., difference information, such as shape difference, among various objects. Since such information is essential for classification, the projection onto $\mathcal{D}$ enhances the classification ability of MCM, similarly to that of the projection of class subspaces onto a generalized difference subspace (GDS) in CMSM [22]. Finally, we classify an input image set by using the cone similarity between the projected input and class convex cones $\hat{\mathcal{C}}_{in}, \{\hat{\mathcal{C}}_j\}$, as shown in Fig. 1. We call this method the *constrained mutual convex cone method* (CMCM).

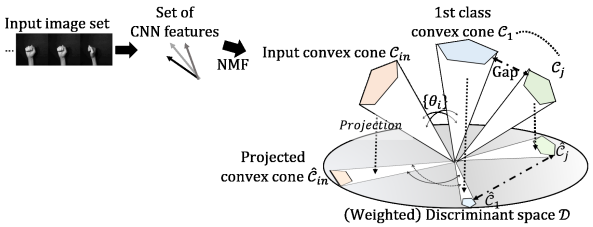


Figure 1: Conceptual diagram of the basic idea of the proposed methods. First, feature vectors are extracted from an image set. Then, each set of features is represented by a convex cone. The classification is performed by calculating the similarity based on the angles $\{\theta_i\}$ between the convex cones $\mathcal{C}_{in}$ and $\{\mathcal{C}_j\}$. To enhance the classification ability of this approach, the projection of the convex cones onto the discriminant space $\mathcal{D}$ or the weighted discriminant space is introduced before calculating the angles.

Then, we further extend the proposed MCM and CMCM, considering more practical cases. So far, MCM and CMCM assume that an image set of a class can be well represented by a single convex cone. However, it is not necessarily reasonable in many practical applications, e.g., in the case that there are multiple videos collected under different situations for a class. In such cases, a single cone is insufficient to represent the complicated structure information. To address this issue, we represent a set of images by multiple convex cones instead of a single convex cone.

Moreover, for the above representation, we redesign how to generate a discriminant space for CMCM. We re-define the between-class variance (gaps) to extract more complicated gaps between multiple convex cones. We calculate the gaps for every pair of convex cones from different classes and then generate a discriminant space from these gaps. In this process, we also introduce weights on gaps to further enhance the discriminant ability. The basic idea is that the weight on a gap is set to be larger when the gap is small. From this strategy, we expect the effect that small gaps between two cones can be enlarged after the projection onto the discriminant space.

However, the extension method requires high computational cost in compensation for its high discriminant ability. To alleviate this high computational cost, we introduce a new strategy for fast implementation. The key idea is to divide the similarity calculation into two steps, where we first use the subspace-based method and then use the cone-based method.

In preparation for this fast implementation strategy, we generate the subspaces con-

taining each convex cone by applying the Gram-Schmidt orthogonalization to the basis of the cone in advance. Then, in the first step, we calculate the similarities between the input and reference subspaces corresponding to their original cones, and select several neighbourhood cones of the input cone by using the similarities. The calculation using subspaces is much faster than that directly using cones. In the second step, we precisely calculate the similarities between the input cone and the selected neighbourhood cones by the proposed method. We name the CMCM with this selection process "fast CMCM". As shown later, the fast CMCM can achieve about ten times speedup in comparison with the original method.

The main contributions of this paper are summarized as follows.

1. To enhance the subspace-based methods, we introduce a convex cone representation to accurately and compactly represent a set of features with the non-negative constraint as typified by CNN features.
2. We introduce two novel mechanisms in our cone-based classification: a) multiple angles between two convex cones to measure the similarity between the cones; b) projection of convex cones onto a discriminant space to enlarge the class separability.
3. To enhance the classification performance of the cone-based framework, we propose a weighted discriminant space to further enlarge the class separability by reflecting the local relationship between multiple convex cones of a class.
4. To reduce the computational cost induced by the convex cone representation, we develop a fast implementation of CMCM by switching the cone-based and subspace-based methods.
5. With the valid combination of the contributions 2) and 3), we build three types of novel image-set based classification methods, called MCM, CMCM and extended CMCM, based on the convex cone representation, the discriminant space, and the multiple cone representation.

Some preliminary ideas have appeared in our earlier work [23]. However, the present paper has been significantly enhanced in the following aspects: 1) this paper developed two new technical extensions, as described in our contributions 3 and 4

as mentioned above, which led to remarkable improvement in the classification performance and reduction in computational cost; 2) this paper provided much more detailed formulation and in-depth analysis of the proposed methods from two pivotal perspectives: the effectiveness of using multiple angles for the classification performance and the essence of the gaps among cones; and 3) this paper also redesigned all the experiments on object recognition and face recognition, to extensively verify the effectiveness of the proposed methods.

2. Related work

This section first describes the algorithms for two standard image-set classification methods, MSM and CMSM. Then, we provide an overview of the concept of convex cones and the generation of a convex cone by non-negative matrix factorization.

2.1. Mutual subspace method based on canonical angles

Mutual subspace method (MSM) [1] is a classification method of an image set based on its subspace representation. The essence of MSM is to use the structural similarity between subspaces as the similarity between input and reference image sets. The subspace similarity is defined by the canonical angles between two subspaces to precisely compare the whole structures of them.

Let $\mathcal{S}_1$ and $\mathcal{S}_2$ be N_1 and $N_2(\leq N_1)$-dimensional subspaces in a d-dimensional vector space, respectively. The canonical angles $\{0 \leq \theta_1, \cdots, \theta_{N_2} \leq \frac{\pi}{2}\}$ between $\mathcal{S}_1$ and $\mathcal{S}_2$ are recursively defined as follows [10, 11]:

$$\begin{aligned} \cos\theta_i = \max_{\mathbf{u}\in\mathcal{S}_1}\max_{\mathbf{v}\in\mathcal{S}_2} \mathbf{u}^{\mathrm{T}}\mathbf{v} = \mathbf{u}_i^{\mathrm{T}}\mathbf{v}_i, \\ s.t.\ \|\mathbf{u}_i\|_2 = \|\mathbf{v}_i\|_2 = 1, \mathbf{u}_i^{\mathrm{T}}\mathbf{u}_j = \mathbf{v}_i^{\mathrm{T}}\mathbf{v}_j = 0, i \neq j, \end{aligned} \tag{1}$$

where θ_i is the i-th canonical angle between $\mathcal{S}_1$ and $\mathcal{S}_2$, and θ_i is formed by two canonical vectors $\mathbf{u}_i$ and $\mathbf{v}_i$. The j-th canonical angle θ_j is the smallest angle in the direction orthogonal to the canonical angles $\{\theta_k\}_{k=1}^{j-1}$.

The canonical angles can be calculated from the orthogonal projection matrices onto subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$. Let $\{\phi_i\}_{i=1}^{N_1}$ be orthonormal basis vectors of $\mathcal{S}_1$ and $\{\psi_i\}_{i=1}^{N_2}$

be orthonormal basis vectors of $\mathcal{S}_2$. The projection matrices $\mathbf{P}_1$ and $\mathbf{P}_2$ are calculated as $\sum_{i=1}^{N_1} \phi_i {\phi_i}^{\mathrm{T}}$ and $\sum_{i=1}^{N_2} \psi_i {\psi_i}^{\mathrm{T}}$, respectively; and $\cos^2 \theta_i$ can be obtained as the i-th largest eigenvalue λ_i of $\mathbf{P}_1\mathbf{P}_2$ (or $\mathbf{P}_2\mathbf{P}_1$) [10, 11].

The structural similarity between two subspaces $\mathcal{S}_1$ and $\mathcal{S}_2$ is defined by using the canonical angles, as follows:

$$sim(\mathcal{S}_1, \mathcal{S}_2) = \frac{1}{N_2} \sum_{i=1}^{N_2} \cos^2 \theta_i. \tag{2}$$

In MSM, an input subspace $\mathcal{S}_{in}$ is classified by comparison with class subspaces $\{\mathcal{S}_c\}_{c=1}^{C}$ by measuring their similarity using this similarity.

2.2. Constrained MSM

MSM was extended to Constrained MSM (CMSM) [12, 22] by introducing projection of subspaces onto a constraint space. As a constraint space, generalized difference subspace (GDS) [22] is typically used. GDS consists of only difference components among subspaces $\{\mathcal{S}_c\}_{c=1}^{C}$. Thus, the projection of class subspaces onto GDS can enlarge the separability among the class subspaces, substantially enhancing the classification performance of MSM [22].

2.3. Convex cone representation

In this subsection, we describe the definition of a convex cone and the projection of a vector onto a convex cone. A convex cone $\mathcal{C}$ in the d-dimensional vector space $\mathbb{R}^d$ is defined by a finite number of generators (basis vectors) $\{\mathbf{b}_i \in \mathbb{R}^d\}_{i=1}^{r}$:

$$\mathcal{C} = \{\mathbf{a} | \mathbf{a} = \sum_{i=1}^{r} w_i \mathbf{b}_i, w_i \geq 0\}. \tag{3}$$

As indicated by (3), a convex cone has non-negative constraints on the combination coefficients, unlike a subspace.

Given a set of feature vectors $\{\mathbf{f}_i \in \mathbb{R}^d\}_{i=1}^{N}$. We obtain the basis vectors $\{\mathbf{b}_i\}_{i=1}^{r}$ of a convex cone by non-negative matrix factorization (NMF) [24, 25] to suppress noise and remove redundant bases of a cone model.

Let $\mathbf{F} = [\mathbf{f}_1 \mathbf{f}_2 \dots \mathbf{f}_N] \in \mathbb{R}^{d \times N}$ and $\mathbf{B} = [\mathbf{b}_1 \mathbf{b}_2 \dots \mathbf{b}_r] \in \mathbb{R}^{d \times r (< N)}$. NMF generates the basis vectors $\mathbf{B}$ by solving the following optimization problem:

$$\arg\min_{\mathbf{B},\mathbf{W}} \|\mathbf{F} - \mathbf{B}\mathbf{W}\|_F \;\; s.t. \;\; (\mathbf{B})_{i,j}, (\mathbf{W})_{i,j} \geq 0, \tag{4}$$

where $\|\cdot\|_F$ denotes the Frobenius norm, and the number of basis vectors r is a hyperparameter. We use the alternating non-negativity-constrained least squares-based method [25] to solve this problem.

Although the basis vectors can be computed by NMF, the projection of a vector $\mathbf{x}$ onto the convex cone is slightly complicated by the non-negative constraints. In [18], the projection is defined with the non-negative least squares (NNLS) method [26] as follows:

$$\arg\min_{\{w_i\}} \|\mathbf{x} - \sum_{i=1}^{r} w_i \mathbf{b}_i\|_2 \;\; s.t. \;\; w_i \geq 0. \tag{5}$$

The projected vector $\hat{\mathbf{x}}$ is obtained as $\hat{\mathbf{x}} = \sum_{i=1}^{r} w_i \mathbf{b}_i$.

In the end, the angle θ between a vector $\mathbf{x}$ and the convex cone can be calculated as follows:

$$\cos\theta = \frac{\hat{\mathbf{x}}^{\mathrm{T}} \mathbf{x}}{\|\hat{\mathbf{x}}\|_2 \|\mathbf{x}\|_2}. \tag{6}$$

3. Mutual convex cone method

In this section, we describe the algorithm of MCM, after establishing the definition of geometric similarity between two convex cones.

3.1. Basic idea

The canonical angles between subspaces can be analytically calculated from the projection matrices in closed form. In contrast, the calculation of the angles between cones is not trivial, as the projection onto a cone includes the process of NNLS (Eq. 5). Hence, we propose a new method for obtaining the angles in turn from the smallest to the largest by applying the alternating least squares method (ALS) [21] to the convex cones sequentially. The key idea here is to project convex cones onto the orthonormal complement space of the subspace spanned by two vectors forming the angle obtained

Algorithm 1: Algorithm to search for the pair $\mathbf{p}_1$ and $\mathbf{q}_1$.

Input: Basis vectors $\{\mathbf{b}_i^1\}, \{\mathbf{b}_i^2\}$ of two convex cones, $\mathcal{C}_1$ and $\mathcal{C}_2$.

Let $\mathcal{P}_j(\mathbf{y})$ be the projection operator of a vector $\mathbf{y}$ onto a convex cone $\mathcal{C}_j$, explained in Section 2.3.

1. Randomly initialize $\mathbf{y} \in \mathbb{R}^d$.
2. $\mathbf{p}_1 = \mathcal{P}_1(\mathbf{y})/\|\mathcal{P}_1(\mathbf{y})\|_2$.
3. $\mathbf{q}_1 = \mathcal{P}_2(\mathbf{y})/\|\mathcal{P}_2(\mathbf{y})\|_2$.
4. $\hat{\mathbf{y}} = (\mathbf{p}_1 + \mathbf{q}_1)/2$.
5. If $\|\hat{\mathbf{y}} - \mathbf{y}\|_2$ is sufficiently small, the procedure is completed. Otherwise, return to step 2 after setting $\mathbf{y} = \hat{\mathbf{y}}$.

return $\cos^2\theta_1 = (\frac{\mathbf{p}_1^{\mathrm{T}}\mathbf{q}_1}{\|\mathbf{p}_1\|_2\|\mathbf{q}_1\|_2})^2$.

in the previous step and then to apply ALS to the projected cones again. This sequential projection works effectively like the orthogonal decomposition of a convex cone in high dimensional vector space.

The following subsection describes the detailed definition of the multiple angles and the similarity between convex cones.

3.2. Multiple angles and geometric similarity between two convex cones

To define the geometric similarity between two convex cones, we consider how to define multiple angles between two convex cones like canonical angles. Let two convex cones $\mathcal{C}_1$, $\mathcal{C}_2$ be formed by basis vectors $\{\mathbf{b}_i^1 \in \mathbb{R}^d\}_{i=1}^{N_1}$ and $\{\mathbf{b}_i^2 \in \mathbb{R}^d\}_{i=1}^{N_2}$, respectively. Assume that $N_1 \leq N_2$ for convenience. As we need to consider the non-negative constraint, the angles between two convex cones cannot be obtained analytically, unlike the canonical angles. Instead, we find two vectors, $\mathbf{p} \in \mathcal{C}_1$ and $\mathbf{q} \in \mathcal{C}_2$, which form the smallest angles between the convex cones. In this way, we sequentially define multiple angles from the smallest to the largest, in order.

First, we search for a pair of d-dimensional vectors $\mathbf{p}_1 \in \mathcal{C}_1$ and $\mathbf{q}_1 \in \mathcal{C}_2$, which have the maximum correlation, by solving the following optimization problem:

$$\cos\theta_1 = \max_{\mathbf{p}_1 \in \mathcal{C}_1} \max_{\mathbf{q}_1 \in \mathcal{C}_2} \mathbf{p}_1^{\mathrm{T}}\mathbf{q}_1, \quad s.t.\ \|\mathbf{p}_1\|_2 = \|\mathbf{q}_1\|_2 = 1. \tag{7}$$

This problem can be solved by the alternating least squares method (ALS) [21]. Thus, the first angle θ_1 can be obtained as the angle formed by $\mathbf{p}_1$ and $\mathbf{q}_1$. The algorithm of

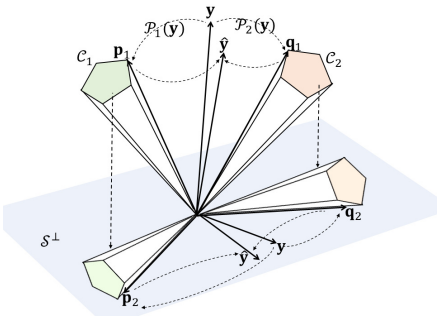


Figure 2: Conceptual diagram of the procedure searching for pairs of vectors $\{\mathbf{p}_i, \mathbf{q}_i\}$. The first pair of $\mathbf{p}_1$ and $\mathbf{q}_1$ can be found by the alternating least squares method. The second pair of $\mathbf{p}_2$ and $\mathbf{q}_2$ is obtained by searching the orthogonal complement space $\mathcal{S}^\perp$ of $\mathcal{S} = \text{Span}\{\mathbf{p}_1, \mathbf{q}_1\}$ [23].

the ALS is summarized in Algorithm 1.

For the second angle θ_2, we find a pair of vectors $\mathbf{p}_2$ and $\mathbf{q}_2$ with the maximum correlation, but with the minimum correlation with $\mathbf{p}_1$ and $\mathbf{q}_1$. Such a pair can be found by applying ALS to the projected convex cones $\mathcal{C}_1$ and $\mathcal{C}_2$ on the orthogonal complement space $\mathcal{S}^\perp$ of the subspace $\mathcal{S}$ spanned by the vectors $\mathbf{p}_1$ and $\mathbf{q}_1$ as shown in Fig. 2. Then θ_2 is formed by $\mathbf{p}_2$ and $\mathbf{q}_2$. In this way, we can obtain all of the pairs of vectors $\mathbf{p}_i, \mathbf{q}_i$ forming the i-th angle θ_i, $i = 1, \ldots, N_1$.

With the resulting angles $\{\theta_i\}_{i=1}^{N_1}$, we define the geometric similarity sim between two convex cones $\mathcal{C}_1$ and $\mathcal{C}_2$ as

$$sim(\mathcal{C}_1, \mathcal{C}_2) = \frac{1}{N_1} \sum_{i=1}^{N_1} \cos^2 \theta_i. \tag{8}$$

3.3. Algorithm of mutual convex cone method

The mutual convex cone method (MCM) classifies an input convex cone, using the similarities defined by Eq. (8) between the input and reference convex cones. MCM consists of two phases, a training phase and a recognition phase.

Given C class sets with L images $\{\mathbf{x}_i^c\}_{i=1}^{L}$.

Training Phase

1. Feature vectors $\{\mathbf{f}_i^c\}$ are extracted from the images $\{\mathbf{x}_i^c\}$ of class c.
2. The basis vectors of class-c convex cone, $\{\mathbf{b}_j^c\}$, are generated by applying NMF to the set of feature vectors $\{\mathbf{f}_i^c\}$.

3. $\{\mathbf{b}_j^c\}$ are recorded as the class convex cone of class c.
4. The above process is conducted for all C classes.

Recognition Phase

1. A set of images $\{\mathbf{x}_i^{in}\}$ is input.
2. Feature vectors $\{\mathbf{f}_i^{in}\}$ are extracted from images $\{\mathbf{x}_i^{in}\}$.
3. The basis vectors of the input convex cone, $\{\mathbf{b}_j^{in}\}$, are generated by applying NMF to the input set of feature vectors $\{\mathbf{f}_i^{in}\}$.
4. The input image set $\{\mathbf{x}_i^{in}\}$ is classified based on the similarity (Eq. (8)) between the input convex cone $\{\mathbf{b}_j^{in}\}$ and the class-c convex cone $\{\mathbf{b}_j^c\}$.

4. Constrained mutual convex cone method

In this section, we extend MCM by introducing the projection onto a discriminant space. We first describe the basic idea of the introduction of a discriminant space, and then define the discriminant space based on the gaps among convex cones. After that, we detail the algorithm of the extended MCM.

4.1. Basic idea

As convex cones capture essential information of each image set, the gaps between them precisely capture the difference components between corresponding objects, such as the shape difference. The performance of MCM can be enhanced by extracting the gap information, since such information is essential for classification. To this end, we design a discriminant space based on the gaps.

4.2. Generation of discriminant space

To enhance the performance of MCM, we introduce a discriminant space $\mathcal{D}$, which enlarge the gaps (the between-class variance $\mathbf{S_b}$) and minimizes the within-class variance $\mathbf{S_w}$ for the convex cones projected on $\mathcal{D}$, similarly to the Fisher discriminant analysis (FDA). In our method, the within-class variance $\mathbf{S_w}$ is calculated from basis vectors of convex cones, and the between-class variance $\mathbf{S_b}$ is calculated from gaps among convex cones for effectively utilizing the information formed by convex cones.

We define these gaps as follows. Let $\mathcal{C}_c$ be the c-th class convex cone with N_c basis vectors $\{\mathbf{b}_i^c\}_{i=1}^{N_c}$, $\mathcal{P}_c$ be the projection operation of a vector onto $\mathcal{C}_c$ defined by Eq. (5),

Algorithm 2: Procedure to search for a set of first vectors $\{\mathbf{p}_1^c\}_{c=1}^C$

Input: Basis vectors $\{\mathbf{b}_i^c\}$ of convex cones $\{\mathcal{C}_c\}_c$

Let $\mathcal{P}_j(\mathbf{y})$ be the projection operator of a vector $\mathbf{y}$ onto a convex cone $\mathcal{C}_j$.
1. Randomly initialize $\mathbf{y}_1$.
2. Project $\mathbf{y}_1$ onto each convex cone, and then normalize the projection as $\mathbf{p}_1^c = \mathcal{P}_c(\mathbf{y}_1)/\|\mathcal{P}_c(\mathbf{y}_1)\|_2$.
3. $\hat{\mathbf{y}}_1 = \sum_{c=1}^C \mathbf{p}_1^c/C$.
4. If $\|\mathbf{y}_1 - \hat{\mathbf{y}}_1\|_2$ is sufficiently small, the procedure is completed. Otherwise, return to step 2 after setting $\mathbf{y}_1 = \hat{\mathbf{y}}_1$.

return $\{\mathcal{P}_c(\mathbf{y})\}_c$

and C be the number of the classes. We consider C vectors $\{\mathbf{p}_1^c\}$, $c = 1, 2, \ldots, C$, such that the sum of the correlation $\sum_{i \neq j} (\mathbf{p}_1^i)^{\mathrm{T}} \mathbf{p}_1^j / (\|\mathbf{p}_1^i\|_2 \|\mathbf{p}_1^j\|_2)$ is maximum. Such vectors can be obtained by using the concept of generalized canonical correlation analysis [27, 28]. The detailed procedure is shown in Algorithm 2, which is almost the same as the original algorithm, except that the non-negative least squares (LS) method is used instead of the standard LS method.

Next, we search for a set of second vectors $\{\mathbf{p}_2^c\}$ with the maximum sum of the correlations under the constraint that they have the minimum correlation with the previously found $\{\mathbf{p}_1^c\}$. The second vectors $\{\mathbf{p}_2^c\}$ can be obtained by applying the above procedure to the convex cones projected onto the orthogonal complement space of the vector $\mathbf{y}_1$. In the same way, a set of the j-th vectors $\{\mathbf{p}_j^c\}$ can be computed by applying the same procedure to the convex cones projected onto the orthogonal complement space of $\{\mathbf{y}_k\}_{k=1}^{j-1}$. In this way, we finally obtain the sets of $\{\mathbf{p}_j^c\}$. With the sets of $\{\mathbf{p}_j^c\}$, we define a difference vector $\mathbf{d}_j^{c_1 c_2}$ as

$$\mathbf{d}_j^{c_1 c_2} = \mathbf{p}_j^{c_1} - \mathbf{p}_j^{c_2}. \tag{9}$$

Considering that each difference vector represents the gap between the two convex cones, we use these vectors to define $\mathbf{S}_\mathbf{b}$ as

$$\mathbf{S}_\mathbf{b} = \sum_{j=1}^{N_g} \sum_{c_1=1}^{C-1} \sum_{c_2=c_1+1}^{C} \mathbf{d}_j^{c_1 c_2} (\mathbf{d}_j^{c_1 c_2})^{\mathrm{T}}, \tag{10}$$

where N_g is the minimum number of basis vectors of class convex cones, i.e., $\min(\{N_c\})$.

Next, we define the within-class variance $\mathbf{S_w}$ by using the basis vectors $\{\mathbf{b}_i^c\}$ for all classes of convex cones:

$$\mathbf{S_w} = \sum_{c=1}^{C} \sum_{i=1}^{N_c} (\mathbf{b}_i^c - \mu_c)(\mathbf{b}_i^c - \mu_c)^{\mathrm{T}}, \tag{11}$$

where $\mu_c = \sum_{i=1}^{N_c} \mathbf{b}_i^c / N_c$. Finally, the N_d-dimensional discriminant space $\mathcal{D}$ is spanned by N_d eigenvectors $\{\boldsymbol{\phi}_i\}_{i=1}^{N_d}$ corresponding to the N_d largest eigenvalues $\{\gamma_i\}_{i=1}^{N_d}$ of the following eigenvalue problem:

$$\mathbf{S_b}\boldsymbol{\phi}_i = \gamma_i \mathbf{S_w}\boldsymbol{\phi}_i. \tag{12}$$

4.3. Algorithm of constrained mutual convex cone method

We construct the constrained MCM (CMCM) by incorporating the projection onto the discriminant space $\mathcal{D}$ into the MCM. CMCM consists of a training phase and a recognition phase. In the following, we explain each phase for the case in which C classes have L images $\{\mathbf{x}_i^c\}_{i=1}^{L}$ each and the discriminant space $\mathcal{D}$ is utilized.

Training Phase

1. Feature vectors $\{\mathbf{f}_i^c\}$ are extracted from images $\{\mathbf{x}_i^c\}$.
2. The basis vectors of the c-th class convex cone, $\{\mathbf{b}_j^c\}$, are generated by applying NMF to each class set of feature vectors.
3. Difference vectors $\{\mathbf{d}_j^{c_1 c_2}\}$ are computed according to the method explained in section 4.2.
4. The discriminant space $\mathcal{D}$ is generated by solving Eq. (12) with $\{\mathbf{b}_j^c\}$ and $\{\mathbf{d}_j^{c_1 c_2}\}$.
5. The basis vectors $\{\mathbf{b}_j^c\}$ are projected onto $\mathcal{D}$.
6. A convex cone formed by a set of the projected basis vectors $\{\hat{\mathbf{b}}_j^c\}_j$ is registered as the class convex cones of class c.

Recognition Phase

1. A set of images $\{\mathbf{x}_i^{in}\}$ is input.
2. Feature vectors $\{\mathbf{f}_i^{in}\}$ are extracted from images $\{\mathbf{x}_i^{in}\}$.
3. The basis vectors of a convex cone, $\{\mathbf{b}_j^{in}\}$, are generated by applying NMF to the set of feature vectors.

4. The basis vectors $\{\mathbf{b}_j^{in}\}$ are projected onto the discriminant space $\mathcal{D}$ and then the lengths of the projected basis vectors are normalized to 1. The normalized projections are represented by $\{\hat{\mathbf{b}}_j^{in}\}$.
5. The input set $\{\mathbf{x}_i^{in}\}$ is classified based on the similarity (Eq. (8)) between the input convex cone $\{\hat{\mathbf{b}}_j^{in}\}$ and each class convex cone $\{\hat{\mathbf{b}}_j^{c}\}$.

5. Extension of constrained mutual convex cone method

In this section, we further enhance the CMCM by incorporating the information of the fine local structure between different classes into the generation of an enhanced discriminant space, considering the case that an image set of a class has the complex structure. We further improve the ability of the enhanced discriminant space by introducing weights on gaps. Finally, we propose a fast implementation of the enhanced CMCM.

5.1. Basic idea

In practical applications, a class distribution are often complicated, for example, in the case that an image set of a class contains multiple videos collected under different situations. In these cases, it is reasonable to represent each class by multiple cones instead of a single cone. In our method using multiple reference cones, the classification of an input cone $\mathcal{C}_{in}$ is performed by the nearest neighbor classifier in a very similar procedure to the original CMCM using a single convex cone, except that this enhanced CMCM uses a newly designed discriminant space.

In the following sections, we redesign the method for generating an enhanced discriminant space in response to the multiple convex cone representation of a class. Then, we introduce weights on gaps to incorporate local structure into the generation of an enhanced discriminant space.

5.2. CMCM with enhanced discriminant space

Consider n^c training image sets $\{\mathbf{X}_i^c\}_{i=1}^{n^c}$ for the c-th class, where each image set $\mathbf{X}_i^c$ has n_i^c images $\{\mathbf{x}_{i,j}^c\}_{j=1}^{n_i^c}$. Let $\mathbf{F}_i^c \in \mathbb{R}^{d\times n_i^c}$ be a matrix whose j-th column vector is a feature vector such as pixel intensities and CNN feature extracted from the image $\mathbf{x}_{i,j}^c$,

and $\{\mathbf{b}_{i,j}^c \in \mathbb{R}^d\}_{j=1}^{N_i^c}$ be the basis vectors of each reference convex cone $\mathcal{C}_i^c$ generated from each feature set $\mathbf{F}_i^c$.

We first reformulate the between-class variance $\mathbf{S_b}$ by using multiple gaps (difference vectors) $\{\mathbf{d}_j^{ik,hl}\}$ between every pair of convex cones $\mathcal{C}_i^k$ and $\mathcal{C}_h^l$ from different classes as follows:

$$\mathbf{S}'_{\mathbf{b}} = \sum_{j=1}^{N_g} \sum_{k=1}^{C-1} \sum_{l=k+1}^{C} \sum_{i=1}^{n^k} \sum_{h=1}^{n^l} \mathbf{d}_j^{ik,hl} (\mathbf{d}_j^{ik,hl})^{\mathrm{T}}, \tag{13}$$

where N_g is the minimum number of basis vectors of references convex cones, min($\{N_i^c\}$), and $\{\mathbf{d}_j^{ik,hl}\}$ are the difference vectors between vector pairs $\{\mathbf{p}_j^{ik}, \mathbf{p}_j^{hl}\}$ of two convex cones $\mathcal{C}_i^k$ and $\mathcal{C}_h^l$, which are obtained by applying the method described in Sec. 4.2, and C is the number of classes.

Subsequently, we reformulate the within-class variance $\mathbf{S_w}$ as follows:

$$\mathbf{S}'_{\mathbf{w}} = \sum_{c=1}^{C} \sum_{i=1}^{n^c} \sum_{j=1}^{N_i^c} (\mathbf{b}_{i,j}^c - \mu_i^c)(\mathbf{b}_{i,j}^c - \mu_i^c)^{\mathrm{T}}, \tag{14}$$

where μ_i^c is the mean vector of the basis vectors $\{\mathbf{b}_{i,j}^c\}$ of the i-th reference convex cone of the c-th class, $\{\mathcal{C}_i^c\}$, which is calculated by $\sum_{j=1}^{N_i^c} \mathbf{b}_{i,j}^c / N_i^c$.

We obtain an enhanced discriminant space $\mathcal{D}_e$ as the subspace spanned by N_d eigenvectors corresponding to the N_d largest eigenvalues of the following eigenvalue problem:

$$\mathbf{S}'_{\mathbf{b}} \boldsymbol{\phi}_i = \gamma_i \mathbf{S}'_{\mathbf{w}} \boldsymbol{\phi}_i. \tag{15}$$

With the multiple cone representation, the projection of cones onto $\mathcal{D}_e$ can more accurately and finely maximize between-class variance $\mathbf{S}'_{\mathbf{b}}$ while minimizing within-class variance $\mathbf{S}'_{\mathbf{w}}$, in comparison with a naive discriminant space generated based on a single cone representation.

5.3. Enhanced discriminant space with weights

So far, all the gaps (difference vectors) are treated with the same contribution to generating the enhanced discriminant space $\mathcal{D}_e$. However, the smallest gaps between

convex cones are more important for discrimination than the largest gaps. In fact, several studies reported the validity of this idea [29–31]. Motivated by these studies, we expect that adding different weights can ensure that the smallest gaps between two cones are selectively enlarged after the projection onto the discriminant space, which leads to a better discriminative ability. Concretely, we enhance the discriminant space by introducing the weighted between-class variance with weights $\{w^{ikhl}\}$ as follows:

$$\mathbf{S}''_{\mathrm{b}} = \sum_{j=1}^{N_g} \sum_{k=1}^{C-1} \sum_{l=k+1}^{C} \sum_{i=1}^{n^k} \sum_{h=1}^{n^l} w^{ikhl} \mathbf{d}_j^{ik,hl} (\mathbf{d}_j^{ik,hl})^{\mathrm{T}}, \tag{16}$$

where the weights $\{w^{ikhl}\}$ are defined as

$$w^{ikhl} = \sum_{j=1}^{N_g} ((\mathbf{p}_j^{ik})^{\mathrm{T}} \mathbf{p}_j^{hl})^2 / N_g. \tag{17}$$

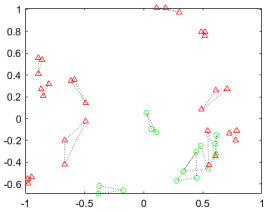
This formulation means that the value of w^{ikhl} increases as the corresponding gap between convex cones becomes smaller. The enhanced discriminant space with weights maximizes more effectively the between-class variance, while minimizing the within-class variance. We can obtain the enhanced discriminant space as the subspace spanned by N_d eigenvectors corresponding to the N_d largest eigenvalues of the following eigenvalue problem:

$$\mathbf{S}''_{\mathbf{b}} \boldsymbol{\phi}_i = \gamma_i \mathbf{S}'_{\mathbf{w}} \boldsymbol{\phi}_i. \tag{18}$$

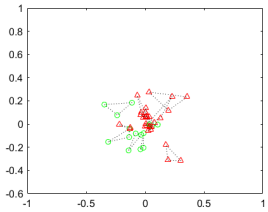
In the following, we refer this further enhanced discriminant space as the weighted discriminant space $\mathcal{D}_{ew}$.

5.4. Validity of enhanced discriminant space

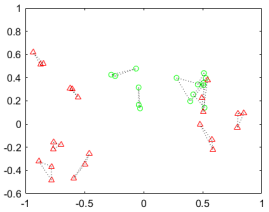
To see the high discriminative ability of the enhanced discriminant spaces, we visualize the projections of convex cones onto $\mathcal{D}$, $\mathcal{D}_e$, and $\mathcal{D}_{ew}$ as 2D maps by using multi-dimensional scaling (MDS) [32]. For the visualization, we synthesized ten cones $\{\mathcal{C}_i^1\}_{i=1}^{10}$ for class-1 and five cones $\{\mathcal{C}_i^2\}_{i=1}^{5}$ for class-2. Each convex cone $\mathcal{C}_i^c$ is spanned by three 100-dimensional basis vectors $\{\mathbf{b}_{i,j}^c \in \mathbb{R}^{100}\}_{j=1}^{3}$. Figure 3(a) shows the 2D visualization map of the synthesized cones without any projection.



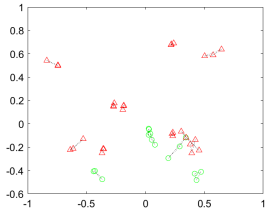
(a) Original data.



(b) Distribution on the discriminant space $\mathcal{D}$ generated from two class convex cones.



(c) Distribution on $\mathcal{D}_e$.



(d) Distribution on $\mathcal{D}_{ew}$.

Figure 3: Results of the projections onto discriminant spaces $\mathcal{D}$, $\mathcal{D}_e$, and $\mathcal{D}_{ew}$. Each plot is generated by MDS. The shapes of each point indicate the corresponding class. The dotted lines are plotted between basis vectors of a cone.

Figs. 3 (b), (c) and (d) show the maps of the cones projected onto $\mathcal{D}$, $\mathcal{D}_e$ and $\mathcal{D}_{ew}$, respectively. Through the comparison between the two maps in (b) and (c), we can see that the cones projected onto $\mathcal{D}_e$ are better separated than the cones projected onto $\mathcal{D}$. This indicates a clear advantage of the enhanced discriminant space $\mathcal{D}_e$ using multiple cones for each class over the naive discriminant space $\mathcal{D}$ using a single cone for each class.

Next, we evaluate the validity of introducing weights to the enhanced discriminant space $\mathcal{D}_{ew}$. We cannot see a large visual difference between the two projection maps of (c) and (d), since a 2D projection map cannot capture completely high-dimensional structures in the 100-dimensional vector space. However, by comparing them carefully, we can observe that there is no overlap in (d), while there are partial overlaps in (c).

To further verify the advantage of $\mathcal{D}_{ew}$ over $\mathcal{D}_e$ quantitatively, we calculated the class separability of the projected cones in the original 100-dimensional vector space.

The class separability is defined as $\mathrm{tr}(\mathbf{S_b}')/\mathrm{tr}(\mathbf{S'_w})$. This index was used for both $\mathcal{D}_e$ and $\mathcal{D}_{ew}$ for fair comparison. The class separabilities of the projections on D_e and D_{ew} are 259.4 and 910.6, respectively. This large difference supports clearly the validity of the introduction of weights to the enhanced discriminant space.

5.5. Fast implementation of CMCM

CMCM is much more computationally costly than the subspace-based methods, MSM and CMSM. This is because the calculation of similarity between cones needs heavy computation due to ALS. Moreover, the cost of the extended CMCM with N reference cones is N times higher than that of the original CMCM.

To alleviate this high cost, we divide the similarity calculation into two steps. The first step is based on the subspace similarity in Eq.(2) and the second step is based on the cone similarity in Eq.(8).

We generate in advance the subspaces containing each cone by applying the Gram-Schmidt orthogonalization to the bases of the cone. Then, in the first step, we generate the input subspace from an input cone and calculate the similarities between the input and reference subspaces. After that, we select several neighborhood reference subspaces according to the subspace similarities obtained above. In the second step, we calculate the similarities of the input cone and the reference cones, which correspond to the reference subspaces selected above. Finally, the input cone is classified into the class with the maximum cone similarity. This two-step process can reduce the computational cost largely, while maintaining the high discriminative ability of the extended CMCM, as clearly shown in experiments in sec.6.

6. Evaluation experiments

In this section, we conduct four experiments to evaluate the effectiveness of the proposed methods. In the first two experiments, we mainly demonstrate the effectiveness of the convex cone representation by comparing the performances of the proposed methods (MCM and the CMCM with $\mathcal{D}$) with the fundamental subspace-based methods (MSM and CMSM). More concretely, the first experiment verifies the effectiveness

of using multiple angles between convex cones to measure the structural similarity between them, using the multi-view objects dataset [33]. The second experiment reveals how efficiently a convex cone captures essential information of an image set for classification by observing the transitions of performances while varying the number of training data, using the multi-view hand shape dataset [34].

The third experiment evaluates the validity of the convex cone model and the representation ability of multiple convex cone models for image-set classification on the YouTube Celebrities dataset [35], using four types of typical image features.

The fourth experiment thoroughly evaluates the classification performance of the proposed methods using three datasets: 1) YouTube Celebrities (YTC) [35], 2) RGBD Object [36], and 3) YouTube Faces (YTF) [37].

6.1. Effectiveness of using multiple angles

In this experiment, we verify the effectiveness of using multiple angles for calculating the structural similarity between convex cones, through a classification experiment using the ETH-80 dataset [33].

6.1.1. Experimental protocol

The ETH-80 dataset consists of object images in eight different classes. Each class has ten types of objects. Thus, this dataset consists of images taken from 80 objects (= 8 classes $\times$ 10 objects). As each object is captured from 41 viewpoints, the total number of images is 3280 (= 80 objects $\times$ 41 viewpoints). One object randomly sampled from each class set was used for training, and the remaining nine objects were used for testing. As an input image set, we used 41 multi-view images for each object. Thus, we have eight image sets for training and 72 (= 8 classes $\times$ 9 objects) image sets for testing. We used images scaled to 32 $\times$ 32 pixels and converted to grayscale. Vectorized features of the grayscale images were used as input, i.e. the dimension of the feature vector is 1024.

We evaluated the classification performance of the mutual convex cone method (MCM) and the constrained MCM (CMCM) with the discriminant space $\mathcal{D}$, while varying the number of angles used for calculating the similarity. As baselines, the mu-

tual subspace method (MSM) and the constrained MSM (CMSM) were also evaluated. Dimensions of reference subspaces and convex cones were set to 20, and dimensions of input subspaces and convex cones were set to 10.

6.1.2. Results and discussion

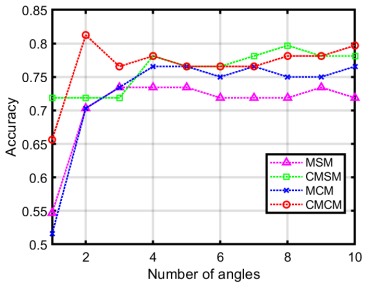


Figure 4: Results of classification experiment. The vertical axis is the accuracy, and the horizontal axis is the number of angles used for calculating the similarity.

Figure 4 shows the accuracy changes of the different methods against the number of angles. The horizontal axis denotes the number of angles used for calculating the similarity. We can confirm that the accuracy of MCM and CMCM increases, as the number of angles increases. This result shows clearly the importance of comparing the whole structures of convex cones by using multiple angles rather than using only the minimum angle for accurate classification.

In the case of using one or two angles, the accuracy of CMCM is lower than CMSM. However, with an increase in the number of angles, CMCM outperforms the methods MSM and CMSM which are based on the subspace representation. This supports the effectiveness of the convex cone representation and indicates that using multiple angles is essential to compare the structures of two convex cones.

6.2. Representation ability of a convex cone

This experiment aims to reveal how efficiently a convex cone captures essential structural information of an image set. To this end, we evaluate our methods while changing the number of training data, using the multi-view hand shape dataset [34]. As detailed later, we show the results from the soft voting with a CNN, in addition

to our methods and MSM/CMSM, to verify the importance of considering structural information of image sets.

6.2.1. Experimental protocol

Figure 5: Sample images of the multi-view hand shape dataset used in the experiments. Each row shows a hand shape from various viewpoints.

The multi-view hand shape dataset [34] consists of 30 classes of hand shapes. Each class data was collected from 100 subjects at a speed of 1 fps for 4 seconds using a multi-camera system equipped with seven synchronized cameras at intervals of 10 degrees. During data collection, the subjects were asked to rotate their hands at a constant speed to increase the number of viewpoints. Figure 5 shows sample images in the dataset. The total number of images collected was 84000 (= 30 classes$\times$4 frames$\times$7 cameras $\times$100 subjects).

We randomly divided the subjects into two sets. One set was used for training, and the other was used for testing. To evaluate the efficiency of convex cone representation, we conducted the experiment by setting the numbers of subjects used for training to 1, 2, 3, 4, 5, 10, and 15. Hence, the number of training images was 840N (= 30 classes$\times$7 cameras$\times$4 frames$\times$ N subjects). We set the number of subjects used for testing to 50. We treated 28 images of a subject as an input image set. Thus, the total number of convex cones for testing was 1500 (=30 classes$\times$50 subjects).

In this experiment, we used CNN features. To extract effective CNN features, we fine-tuned the ResNet-50 [38] pre-trained on ImageNet [39]. To this end, we slightly modified the architecture of the ResNet-50 for our experimental setting. First, we replaced the final layer of the ResNet-50 with a 1024-way fully connected (FC) layer with the ReLU function. Next, we added a $class\ number$-way FC layer with softmax behind the replaced FC layer. Then, we trained the modified ResNet using the training images.

We extracted a CNN feature of each image from the replaced 1024-way FC layer. Thus, the dimensionality d of a CNN feature vector was 1024.

Besides, we utilized this fine-tuned network as a baseline of the methods, which do not consider the structure of an image set, with the following procedure; we classified an input image set based on the average value of the output conviction degrees class from the last FC layer with softmax. In the following, we call this method as "softmax".

For subspace and cone-based methods, the parameters were tuned by grid search on the training set with the following ranges: the dimension of class subspaces and convex cones varied from 10 to 50 in increments of 10; the dimension of input subspace and convex cone varied from 5 to 20 in increments of 5. For CMCM, the dimensions of the discriminant space $\mathcal{D}$ was set to the matrix rank of the between-class variance $\mathbf{S_b}$. For CMSM, the maximum dimension d_{max} of GDS is $(\text{the dimension of class subspace}) \times (\text{the number of classes})$. The dimension of GDS was tuned by the same strategy while varying the reduction dimension d_{red} with the range from 5 to 30 in increments of 5. The dimension of GDS is set to $d_{max} - d_{red}$. This strategy was also conducted on the experiments in the later subsections.

6.2.2. Results and discussion

Table 1 shows the accuracies versus the number N of training subjects. The overall performances of the subspace and convex cone methods achieved competitive results compared with that of softmax. In particular, the improvements are significant when the number of training subjects N is small. From this result, we can see the importance of considering structural information of image sets.

Table 1: Change in the accuracies (%) against the number of training subjects.

N	1	2	3	4	5	10	15
softmax	36.07	71.41	83.87	86.60	91.60	95.73	96.53
MSM	62.27	73.47	85.27	87.60	91.13	95.27	96.20
CMSM	65.87	74.73	87.40	91.00	92.87	95.73	96.27
MCM	63.07	74.60	85.67	88.27	92.07	95.40	96.67
CMCM	**67.87**	**75.33**	**87.47**	**91.33**	**93.53**	**96.27**	**97.00**

Our methods outperformed the subspace-based methods, MSM and CMSM. This

supports the effectiveness of our core ideas: the utilizing of convex cone representation and the cone similarity with multiple angles. Besides, the result suggests that a convex cone can extract meaningful information of image-sets efficiently and stably, even if the number of training data is small.

Moreover, CMCM showed superior performance to MCM in all cases. This improvement indicates the effectiveness of the projection onto $\mathcal{D}$, which is designed to extract discriminative features based on differences among the cones. This insight also means the efficient representation ability of a convex cone model, since the difference among them works well.

6.3. Representation ability of multiple convex cones

In this subsection, we evaluate the representation ability of multiple convex cones in addition to a single convex cone, on the video-based face recognition dataset, YouTube Celebrities (YTC) [35], using four representative image features: Local Binary Pattern (LBP) [40], Histogram of Gradient (HoG) [41] and two types of CNN features, which are extracted from ResNets trained on ImageNet [39] and VGGFace2 [42] datasets, respectively. We show the classification performances of our methods, including the CMCM with $\mathcal{D}_{ew}$ and its fast implementation. For convenience, we use wCMCM to denote the CMCM with $\mathcal{D}_{ew}$.

6.3.1. Experimental protocol

The YTC dataset contains 1910 videos of 47 people. We used a set of face images extracted from a video by the Incremental Learning Tracker (ILT) [43], as an image set. Six videos per each person were randomly selected as training data, and nine videos per each person were randomly selected as test data. We repeated the evaluation five times with different random selections.

For extracting LBP and HoG features, all the extracted face images were scaled to 30 $\times$ 30 pixels and converted to grayscale. For extracting CNN features, all face images were scaled to 224 $\times$ 224 pixels and then inputted to the networks.

The parameters were tuned by the grid search algorithm on the training set, with the ranges shown in the left column of Table 2. The dimensions of the discriminant

Table 2: Parameter ranges used in the grid search algorithm. d_{is} and d_{cs} are the dimensions of input and class subspaces, respectively. N^c and N^{in} are the number of basis vectors of input and class cones, respectively. d_{red} is a parameter for the dimension of GDS, and N_i^c is the number of basis vectors of each cone used for wCMCM. Each element [x–y/z] in the table means that the corresponding parameter varied from x to y in increments of z.

	YTC	RGBD	YTF
d_{is},N^{in}	[3–15/3]	[4–20/4]	[3–15/3]
d_{cs},N^c	[6–30/6]	[8–40/8]	[6–30/6]
d_{red}	[3–15/3]	[4–20/4]	[3–15/3]
N_i^c	[3–15/3]	[4–20/4]	[3–15/3]

space $\mathcal{D}$ and the weighted discriminant space $\mathcal{D}_{ew}$ were set to the matrix rank of the between-class variance $\mathbf{S_b}$ and $\mathbf{S_b''}$, respectively. Besides, for the fast wCMCM, the number of nearest convex cones to be selected by subspace similarities is set to 5.

Table 3: Experimental results (recognition rate (%), standard deviation) for the YTC dataset.

	LBP	HoG	ImageNet	VGG face2
#Clusters[min,max/mode]	[1,3/2]	[1,5/2]	[1,3/2]	[1,3/1]
MSM	32.91±1.29	60.90±1.57	55.51±2.04	89.69±0.96
CMSM	48.75±3.26	72.39±1.76	71.02±1.50	91.49±1.00
MCM	37.40±2.81	62.55±1.84	56.12±1.71	90.54±1.16
CMCM	53.33±2.26	72.86±1.92	71.39±1.92	92.34±1.00
wCMCM	**69.27±2.17**	**77.21±2.52**	**81.47±1.32**	**92.96±0.72**
fast wCMCM	69.08±2.18	76.97±2.87	81.42±1.28	92.91±1.16

6.3.2. Results and discussion

Table 3 shows the classification results of the baselines and the proposed methods using the four kinds of features. The experimental results support the effectiveness of our main ideas as well as the previous two experiments since MCM and CMCM showed competitive results compared with the baselines regardless of the features we used.

Furthermore, the performance of CMCM increased by introducing multiple convex cones and the weighted discriminant space $\mathcal{D}_{ew}$ in wCMCM. This further enhancement shows that 1) the cones have superior representation ability, and 2) the weights work effectively to obtain local fine structural information between cones of different classes

as we expected. Moreover, we notice that the fast version of wCMCM achieved almost the same recognition rate as the original wCMCM, while speeding up more than ten times compared with the original wCMCM as shown in Table 4. This result concludes that we can compare the similarity between a pair of cones, which are faraway to each other, by using the subspace similarity instead of the cone similarity.

Although wCMCM significantly outperforms CMCM in all features, amounts of improvements are different. To analyze this difference, we automatically estimated the number of clusters in each class by applying DBSCAN clustering [44]. The second row of Table 3 shows the minimum, maximum, and mode numbers of clusters. It can be seen that the more clusters there are in a class on average, the more significantly wCMCM improves the classification performance. This indicates that we can efficiently represent the complex structure of each class by using multiple convex cones and extract meaningful information for classification by using the weighted differences between them.

Table 4: Average classification times (millisecond). The numbers of angles for the similarity are 10. This experiment is conducted by Matlab 2018b on Intel CPU i7-7700.

wCMCM	fast wCMCM
507.2	30.0

6.4. Comparison of classification performance with conventional methods

In this subsection, we thoroughly evaluate the classification performance of the proposed methods using three public datasets, YTC, RGBD and YTF. As comparison methods, in addition to the baselines, we show the results of the various fundamental and recent subspace-based methods (DCC [45], GDA [16], GGDA [29], MMD [2], PML [17], RMML-GM [46]), as references. In particular, PML and RMML-GM have been known as powerful classification methods for image-set based recognition. They learn a transformation matrix of subspaces by solving an optimization problem on a Grassmann manifold like Fisher discriminant analysis. Besides, we show the results of other types of methods: covariance-based methods (LEML [47], RMML-SPD [46]), an affine hull-based method (AHISD [4]) and a sparse-representation method (SANP [48]).

In the following, details of each dataset and experimental protocols are described. After that, experiment results are shown.

6.4.1. Datasets and experimental protocols

The **RGBD Object dataset** [36] consists of object images in 51 different classes. There are 3 to 14 objects in each class, and each object is captured from over 200 viewpoints. As an image set, we used a set of multi-view images of each object. The half image sets per each class were randomly selected as training data, and the remaining image sets were used as test data. We repeated the evaluation five times with different random selections. For this dataset, we used CNN features extracted from the ResNet-18 trained on ImageNet [39].

The **YouTube Faces (YTF) database** [37] contains 3425 videos of 1595 people. We cropped face regions with the annotated data [37] and used the cropped face images of each video as an image set. As we removed the classes with only one or two videos, the number of classes used for this experiment is 226. As with the experiment for the RGBD dataset, the image sets of each class were randomly split in half into training and test data. We repeated the evaluation five times with different random splits.

In addition to the above two datasets, we show the additional results of the conventional methods on **YTC** with the same setting in the previous experiment.

For the YTC and YTF datasets, we used CNN features extracted from the ResNet-50 trained on the VGGFace2 dataset [42].

As with the previous experiment, the parameters were tuned by the grid search algorithm on the training set, with the ranges shown in Table 2. Besides, we carefully selected the parameters of the other reference methods by the grid search algorithm based on the suggested ranges in each paper.

6.4.2. Results and discussion

Table 5 shows the classification results of the proposed methods and various conventional methods. The proposed methods showed consistent results with the previous experiments for all datasets, i.e., the results support the effectiveness of our key ideas: the convex cone representation, the cone similarity with multiple angles, and the

Table 5: Experimental results (recognition rate (%), standard deviation) for the three public datasets.

		YTC	RGBD	YTF
Conventional methods	AHISD [4]	90.02±1.17	80.14±1.73	88.99±0.44
	SANP [48]	89.97±1.08	79.28±2.85	70.09±1.86
	LEML [47]	90.83±2.00	88.06±3.12	78.68±0.42
	RMML-SPD [46]	89.93±1.40	88.20±3.58	74.76±0.65
	DCC [45]	92.34±0.81	89.78±2.12	91.63±0.93
	GDA [16]	90.36±1.55	88.06±1.55	81.19±2.88
	GGDA [29]	92.48±1.45	88.78±0.64	81.94±2.46
	MMD [2]	90.30±1.18	82.45±2.72	86.39±1.37
	PML [17]	91.25±0.10	90.22±0.03	89.60±0.06
	RMML-GM [46]	91.30±0.80	90.93±1.48	89.74±0.70
Base lines	MSM	89.69±0.96	89.78±1.06	89.96±0.53
	CMSM	91.49±1.00	90.22±1.85	91.85±0.57
Proposed methods	MCM	90.54±1.16	91.74±0.84	92.60±0.92
	CMCM	92.34±1.00	91.94±0.94	92.82±0.92
	wCMCM	**92.96±0.72**	**92.23±0.94**	**93.17±0.41**
	fast wCMCM	92.91±1.16	91.94±0.60	**93.17±0.31**

discriminant spaces. For instance, CMCM and wCMCM showed better results than CMSM by more than 1% on the RGBD and YTF datasets.

Furthermore, the proposed method achieved competitive results compared with more powerful subspace-based methods. This result also supports the validity of the proposed methods, and indicates that our cone-based frameworks can be further enhanced by utilizing the progress of the subspace-methods in the future.

7. Conclusion

In this paper, we established a novel framework for image set classification, which is based on the convex cone representation, referred to as the constrained mutual convex cone method (CMCM).

The key idea of our framework is to represent an image set by a convex cone and then measure the similarity between two image sets as the geometric similarity between two corresponding convex cones. The geometric similarity of two convex cones is measured with the angles between them, which we defined newly in this paper, by using the alternating least squares method. To derive higher performance from our cone

representation, we designed a new type of discriminant space that increases the class separability between sets of cones from different classes. Moreover, we enhanced the ability of this discriminant space by introducing weights to enlarge the gaps between a pair of close convex cones. As CMCM has high computational cost, we constructed its fast implementation by combining our cone-based method with the conventional subspace-based method.

In the evaluation experiments, we first verified that using multiple angles is essential to compare two convex cones. Then, we demonstrated that the difference between convex cones could capture more useful information for image-set classification. The classification performance of the proposed frameworks was evaluated through extensive experiments, showing that it can achieve competitive results compared with various conventional methods.

In the future, we will further explore the development of 1) a novel discriminant space for convex cones by incorporating recent advances in image feature extraction methods, such as [49–51], to enhance the classification performance, and 2) a fast calculation method of the smallest angles between convex cones.

Acknowledgements

Part of this work was supported by JSPS KAKENHI Grant Number JP16H02842 and a Grant-in-Aid for JSPS Fellows.

References

[1] O. Yamaguchi, K. Fukui, K. Maeda, Face recognition using temporal image sequence, in: IEEE International Conference on Automatic Face and Gesture Recognition, 1998, pp. 318–323.

[2] R. Wang, S. Shan, X. Chen, W. Gao, Manifold-manifold distance with application to face recognition based on image set, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2008, pp. 1–8.

[3] R. Wang, H. Guo, L. S. Davis, Q. Dai, Covariance discriminative learning: A natural and efficient approach to image set classification, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 2496–2503.

[4] H. Cevikalp, B. Triggs, Face recognition based on image sets, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2010, pp. 2567–2573.

[5] Z.-Q. Zhao, S.-T. Xu, D. Liu, W.-D. Tian, Z.-D. Jiang, A review of image set classification, Neurocomputing 335 (2019) 251–260.

[6] H. Cevikalp, G. G. Dordinejad, Discriminatively learned convex models for set based face recognition, in: International Conference on Computer Vision, 2019, pp. 10123–10132.

[7] D. Wei, X. Shen, Q. Sun, X. Gao, W. Yan, Prototype learning and collaborative representation using grassmann manifolds for image set classification, Pattern Recognition 100 (2020) 107123.

[8] G. Zhang, J. Yang, Y. Zheng, Z. Luo, J. Zhang, Optimal discriminative feature and dictionary learning for image set classification, Information Sciences 547 (2021) 498–513.

[9] T.-Y. Hu, A. G. Hauptmann, Statistical Distance Metric Learning for Image Set Retrieval, in: IEEE International Conference on Acoustics, Speech and Signal Processing, 2021, pp. 1765–1769.

[10] S. N. Afriat, Orthogonal and oblique projectors and the characteristics of pairs of vector spaces, in: Mathematical Proceedings of the Cambridge Philosophical Society, Vol. 53, 1957, pp. 800–816.

[11] H. Hotelling, Relations between two sets of variates, Biometrika 28 (3/4) (1936) 321–377.

[12] K. Fukui, O. Yamaguchi, Face recognition using multi-viewpoint patterns for robot vision, in: International Symposium of Robotics Research, 2005, pp. 192–201.

[13] A. S. Georghiades, P. N. Belhumeur, D. J. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (6) (2001) 643–660.

[14] P. N. Belhumeur, D. J. Kriegman, What is the set of images of an object under all possible illumination conditions?, International Journal of Computer Vision 28 (3) (1998) 245–260.

[15] K.-C. Lee, J. Ho, D. J. Kriegman, Acquiring linear subspaces for face recognition under variable lighting, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (5) (2005) 684–698.

[16] J. Hamm, D. D. Lee, Grassmann discriminant analysis: a unifying view on subspace-based learning, in: International Conference on Machine Learning, ACM, 2008, pp. 376–383.

[17] Z. Huang, R. Wang, S. Shan, X. Chen, Projection metric learning on Grassmann manifold with application to video based face recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 140–149.

[18] T. Kobayashi, N. Otsu, Cone-restricted subspace methods, in: International Conference on Pattern Recognition, 2008, pp. 1–4.

[19] Z. Wang, R. Zhu, K. Fukui, J.-H. Xue, Matched shrunken cone detector (MSCD): Bayesian derivations and case studies for hyperspectral target detection, IEEE Transactions on Image Processing 26 (11) (2017) 5447–5461.

[20] R. Zhu, Z. Wang, N. Sogi, K. Fukui, J.-H. Xue, A novel separating hyperplane classification framework to unify nearest-class-model methods for high-dimensional data, IEEE Transactions on Neural Networks and Learning Systems (2019) 1–11.

[21] M. Tenenhaus, Canonical analysis of two convex polyhedral cones and applications, Psychometrika 53 (4) (1988) 503–524.

[22] K. Fukui, A. Maki, Difference subspace and its generalization for subspace-based methods, IEEE Transactions on Pattern Analysis and Machine Intelligence 37 (11) (2015) 2164–2177.

[23] N. Sogi, T. Nakayama, K. Fukui, A method based on convex cone model for image-set classification with CNN features, in: International Joint Conference on Neural Networks (IJCNN), 2018, pp. 1–8.

[24] D. D. Lee, H. S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (6755) (1999) 788.

[25] H. Kim, H. Park, Nonnegative matrix factorization based on alternating nonnegativity constrained least squares and active set method, SIAM Journal on Matrix Analysis and Applications 30 (2) (2008) 713–730.

[26] R. Bro, S. De Jong, A fast non-negativity-constrained least squares algorithm, Journal of Chemometrics 11 (5) (1997) 393–401.

[27] J. Vía, I. Santamaría, J. Pérez, Canonical correlation analysis (CCA) algorithms for multiple data sets: Application to blind SIMO equalization, in: European Signal Processing Conference, 2005, pp. 1–4.

[28] J. Vía, I. Santamaría, J. Pérez, A learning algorithm for adaptive canonical correlation analysis of several data sets, Neural Networks 20 (1) (2007) 139–152.

[29] M. T. Harandi, C. Sanderson, S. Shirazi, B. C. Lovell, Graph embedding discriminant analysis on Grassmannian manifolds for improved image set matching, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2011, pp. 2705–2712.

[30] M. Sugiyama, Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis, Journal of Machine Learning Research 8 (May) (2007) 1027–1061.

[31] M. Sugiyama, Local Fisher discriminant analysis for supervised dimensionality reduction, in: International Conference on Machine Learning, ACM, 2006, pp. 905–912.

[32] I. Borg, P. Groenen, Modern multidimensional scaling: Theory and applications, Journal of Educational Measurement 40 (3) (2003) 277–280.

[33] B. Leibe, B. Schiele, Analyzing appearance and contour based methods for object categorization, in: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 2, 2003, pp. 409–415.

[34] Y. Ohkawa, K. Fukui, Hand-shape recognition using the distributions of multi-viewpoint image sets, IEICE Transactions on Information and Systems 95 (6) (2012) 1619–1627.

[35] M. Kim, S. Kumar, V. Pavlovic, H. Rowley, Face tracking and recognition with visual constraints in real-world videos, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2008, pp. 1–8.

[36] K. Lai, L. Bo, X. Ren, D. Fox, A large-scale hierarchical multi-view rgb-d object dataset, in: IEEE International Conference on Robotics and Automation, IEEE, 2011, pp. 1817–1824.

[37] L. Wolf, T. Hassner, I. Maoz, Face recognition in unconstrained videos with matched background similarity, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2011, pp. 529–534.

[38] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[39] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., ImageNet large scale visual recognition challenge, International Journal of Computer Vision 115 (3) (2015) 211–252.

[40] T. Ojala, M. Pietikäinen, D. Harwood, A comparative study of texture measures with classification based on featured distributions, Pattern Recognition 29 (1) (1996) 51–59.

[41] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Conference on Computer Vision and Pattern Recognition, Vol. 1, IEEE, 2005, pp. 886–893.

[42] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, A. Zisserman, VGGFace2: A dataset for recognising faces across pose and age, in: IEEE International Conference on Automatic Face & Gesture recognition, IEEE, 2018, pp. 67–74.

[43] D. A. Ross, J. Lim, R.-S. Lin, M.-H. Yang, Incremental learning for robust visual tracking, International Journal of Computer Vision 77 (1-3) (2008) 125–141.

[44] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al., A density-based algorithm for discovering clusters in large spatial databases with noise., in: KDD, Vol. 96, 1996, pp. 226–231.

[45] T.-K. Kim, J. Kittler, R. Cipolla, Discriminative learning and recognition of image set classes using canonical correlations, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (6) (2007) 1005–1018.

[46] P. Zhu, H. Cheng, Q. Hu, Q. Wang, C. Zhang, Towards generalized and efficient metric learning on Riemannian manifold, in: International Joint Conference on Artificial Intelligence, 2018, pp. 3235–3241.

[47] Z. Huang, R. Wang, S. Shan, X. Li, X. Chen, Log-Euclidean metric learning on symmetric positive definite manifold with application to image set classification, in: International Conference on Machine Learning, 2015, pp. 720–729.

[48] Y. Hu, A. S. Mian, R. Owens, Sparse approximated nearest points for image set classification, in: IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2011, pp. 121–128.

[49] Z. Ren, Q. Sun, B. Wu, X. Zhang, W. Yan, Learning Latent Low-Rank and Sparse Embedding for Robust Image Feature Extraction, IEEE Transactions on Image Processing 29 (2020) 2094–2107.

[50] H. Du, Y. Wang, F. Zhang, Y. Zhou, Low-Rank Discriminative Adaptive Graph Preserving Subspace Learning, Neural Processing Letters 52 (3) (2020) 2127–2149.

[51] Z. Wang, F. Nie, L. Tian, R. Wang, X. Li, Discriminative feature selection via a structured sparse subspace learning module, in: International Joint Conference on Artificial Intelligence, 2020, pp. 3009–3015.