Mean polynomial kernel for face membership authentication

Raissa Relator Yoshihiro Hirohashi Tsuyoshi Kato
Department of Computer Science, Graduate School of Engineering, Gunma University
1-5-1 Tenjin-cho, Kiryu, Gunma 376-8515, Japan
{relator-raissa@kato-lab., hirohashi-yoshihiro@kato-lab., katotsu@}cs.gunma-u.ac.jp

Abstract

Face recognition techniques have gained much attention and research interests over the recent years due to their vast applications in security and authentication systems. Some of the popular approaches involve support vector machines (SVM), which can either be a binary or a multiclass classification problem, and subspace learning, where data is assumed to lie on some low dimensional manifold, such as that employing the Grassmann kernels. Recent trends involve data in the form of image sequences, hence treating them as data points in a Grassmann manifold and performing discriminant analysis in this space has been widely used. However, this technique requires determining the reduced dimensionality which has been a critical issue for such techniques. In this paper we introduce another kernel for face membership authentication with similarities to the Projection kernel, a Grassmann kernel. Using the proposed kernel, dimensionality reduction is of no concern and, thus, so is data loss. Moreover, data covariance matrices are directly exploited. Experimental results on face membership verification task show the effectiveness of the proposed kernel over the Grassmann kernels and the Grassmann Distance Mutual Subspace Method (GD-MSM).

1 Introduction

Over the last decades, authentication using biometric-based techniques has gained attention from academics due to their promising applications in security and surveillance purposes [7]. Among them, face recognition, the task of identifying a certain query from a face database, has emerged as the most popular subject of research. This may be approached as a binary classification or a multiclassification problem. Many techniques have already been developed, such as those using support vector machines (SVM) [2, 4, 9, 10], and the famous Eigenfaces [17] and Fisherface [1].

Though early face recognition algorithms consider a single image as one data point, recent trends allow us to represent data points as matrices instead of vectors [3, 5, 8, 12, 13, 14, 15, 18, 19]. The algorithm learns given multiple images for a certain subject and a data is composed of image sequences. Latest techniques involve subspace-based learning methods which are developed under the assumption that data can be modeled as a low-dimensional subspace of the image space instead of vectors [20]. Data such as sequences of images from a video feed can be considered, and face recognition can be performed if given multiple pictures or a sequence of images for each subject. [3, 5, 8, 12, 13, 15, 18, 19]. Each image sequence corresponds to some linear subspace and similarity between sequences are obtained by exploiting the angles between subspaces. However, dimension reduction performed in subspace-based methods, which is usually done by retaining the valuable features for discrimination, remains a challenging task. Even with the use of the usual techniques such as Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA), there is always a possibility of information loss. This motivated us to construct a kernel capable of retaining data information while being computationally inexpensive. We can preserve valuable data features while managing to avoid the curse of high-dimensionality and avoiding any form of data loss.

In this paper we propose a new kernel, the *mean polynomial kernel*. This kernel can be directly used to data involving digital image sequences which can be modeled as a set of vectors. We evaluate the performance of the proposed kernel in face membership verification modeled as a binary classification problem. The goal of this operation is to determine whether a subject image is a 'member' or not. This can also be extended to determining whether the given query is the authorized user or owner, which are common situations in accessing secured buildings or offices, logging on to computers, and other access control systems. We use this in conjunction with SVM and also examine the performance of other subspace-based discriminant analysis for comparison.

2 Related literature

As linear subspaces can be represented as points in the Grassmann manifold, recent subspace-based techniques have been formulated in this setting [5, 6, 16]. The usual approach in defining similarity between subspaces involves exploiting the principal angles between them. We give here a brief overview of recent discriminant analysis methods in Grassmann manifolds, and their analogy with the proposed method.

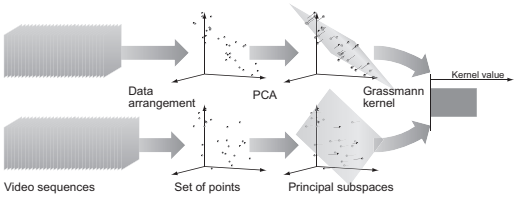
Grassmann kernels, in general, as defined in [5] are positive definite kernel functions in a Grassmann manifold, a set of linear subspaces with a fixed number of dimensions m. Video image sequences are represented by points in the Grassmann manifold, where a single point corresponds to a linear subspace. The Grassmann kernels are thus used to compute similarities among principal subspaces of image sequences.

Definition 1. *Let $\boldsymbol{U}_x$ and $\boldsymbol{U}_y$ be orthonormal matrices whose columns are bases of linear subspaces. The Projection kernel is defined as*

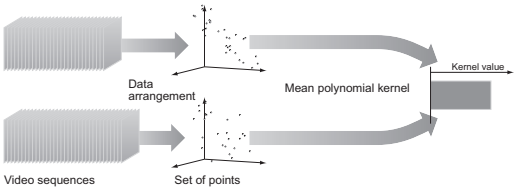
$$k_{PROJ}(\boldsymbol{U}_x, \boldsymbol{U}_y) = \left\| \boldsymbol{U}_x^\top \boldsymbol{U}_y \right\|_F^2,$$

where $\|\cdot\|_F$ denotes the Frobenius norm.

The Projection kernel was derived by defining a metric, the Projection metric, on the Grassmann manifold



(a) Flow for Grassmann kernels



(b) Flow for mean polynomial kernel

Figure 1. Flow of methodology for computing the Grassmann kernels and the mean polynomial kernel. Each image sequence needs to be converted into a subspace when Grassmann kernels are used. An image frame in each video is regarded as a data point. First, principal subspaces are computed, then values of a Grassmann kernel between principal subspaces are calculated. However, some information are lost during the principal subspace conversion. The proposed mean polynomial kernel, on the other hand, directly computes kernel values from the set of data points and avoids information loss.

[5]. Another kernel, the Binet-Cauchy kernel [5, 18], was also introduced in a similar manner. For two subspaces with orthonormal matrices $\boldsymbol{U}_x$ and $\boldsymbol{U}_y$, the Binet-Cauchy kernel is given by

$$k_{BC}(\boldsymbol{U}_x, \boldsymbol{U}_y) = (\det \boldsymbol{U}_x^\top \boldsymbol{U}_y)^2.$$

These kernels, termed Grassmann kernels, were employed in the Grassmann kernel support vector machine (GK-SVM) proposed in [16]. Using GK-SVM, the kernel matrices are first computed and serve as the input for SVM. In an analogous manner, this is how we utilize the proposed kernel with SVM as classifier. A general illustration of the flow of computation of the Grassmann kernels and the proposed kernel is given in Figure 1.

The Grassmann distance mutual subspace method (GD-MSM) [16] is a comparable method to GK-SVM where, instead of Grassmann kernels and SVM, Grassmann distances are combined with the Mutual Subspace Method. The GD-MSM uses some metric $D(\boldsymbol{U}_x, \boldsymbol{U}_y)$ defined on the Grassmann manifold. In the face sequence recognition problem setting, the training stage of the method concatenates the video image sequences in the training set for each subject, and computes a single principal subspace for each subject. Thus, the number of principal subspaces computed in the training stage is equal to the number of subjects. We refer to the set of principal subspaces as the *subject-wise dictionary*. In the test stage, the principal subspace of an unknown video sequence of interest is computed, and the subject that has the minimal Grassmann distance to the unknown principal subspace is determined. To utilize their method for our desired application, we may authorize the unknown person if the subject from which it has a minimum Grassmann distance has a positive membership, otherwise, we give no authorization to the query.

Another approach employing Grassmann distances for the face membership verification problem is by computing a single principal subspace from all the video sequences in each class. Thus, two principal subspaces are obtained in the training stage: one for the positive class and another one for the negative class. We refer to the two principal subspaces as the *class-wise dictionary*. In the test stage, the Grassmann distance of the principal subspace of an unknown video sequence to the subspace of each class is computed. The membership of the unknown sequence is authorized if and only if its distance from the subspace of the positive class is smaller.

3 Mean polynomial kernel

In this section we introduce a new kernel, the *mean polynomial kernel*, for the face recognition task where an example is considered as a set of vectors instead of a single vector.

Let us consider two image sequences $\boldsymbol{\mathcal{X}} = \{\boldsymbol{x}_i\}_{i=1}^{\ell}$ and $\boldsymbol{\mathcal{Y}} = \{\boldsymbol{y}_j\}_{j=1}^{\ell'}$. The two sequences $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$ contain ℓ and ℓ' images, respectively, and each image is represented by a d-dimensional vector containing intensity values of $d = d_1 \times d_2$ pixels in a $d_1 \times d_2$ image. To define a new kernel for image sequences, we introduce a notation of a set of image sequences as $\boldsymbol{\mathcal{S}} = \big\{\{\boldsymbol{z}_i\}_{i=1}^{n} \,|\, n \in \mathbb{N} \text{ and } \forall i \in \mathbb{N}_n,\, \boldsymbol{z}_i \in \mathbb{R}^d\big\}$, where $\mathbb{N}$ is the set of natural numbers, and $\mathbb{N}_n = \{i \in \mathbb{N} \,|\, i \le n\}$. The set $\boldsymbol{\mathcal{S}}$ is the input domain for the new kernel defined as follows.

Definition 2. *Let $k_q : \boldsymbol{\mathcal{S}} \times \boldsymbol{\mathcal{S}} \to \mathbb{R}$ such that*

$$k_q(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}}) = \frac{1}{\ell\ell'} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell'} \langle \boldsymbol{x}_i, \boldsymbol{y}_j \rangle^q,$$

where $\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}} \in \boldsymbol{\mathcal{S}}$ and $q \in \mathbb{N}$. We shall refer to k_q as the qth order mean polynomial kernel.

From the definition, we can say that the uncentered covariance matrix is directly used as a feature vector when $q = 2$. This can be shown by verifying that the Euclidean inner product among vectorized uncentered covariance matrices is equal to the second order mean polynomial. Let us denote the uncentered covariance matrices of $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$ by $\boldsymbol{\Sigma}_x = \frac{1}{\ell} \sum_{i=1}^{\ell} \boldsymbol{x}_i \boldsymbol{x}_i^\top$ and $\boldsymbol{\Sigma}_y = \frac{1}{\ell'} \sum_{j=1}^{\ell'} \boldsymbol{y}_j \boldsymbol{y}_j^\top$, respectively. By defining a feature mapping $\boldsymbol{\phi}(\boldsymbol{\mathcal{X}}) = \mathrm{vec}(\boldsymbol{\Sigma}_x)$, we get

$$\begin{aligned} \langle \boldsymbol{\phi}(\boldsymbol{\mathcal{X}}), \boldsymbol{\phi}(\boldsymbol{\mathcal{Y}}) \rangle &= \langle \mathrm{vec}(\boldsymbol{\Sigma}_x), \mathrm{vec}(\boldsymbol{\Sigma}_y) \rangle = \mathrm{tr}\big(\boldsymbol{\Sigma}_x \boldsymbol{\Sigma}_y\big) \\ &= \frac{1}{\ell\ell'} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell'} \mathrm{tr}\big(\boldsymbol{x}_i \boldsymbol{x}_i^\top \boldsymbol{y}_j \boldsymbol{y}_j{}^\top\big) = \frac{1}{\ell\ell'} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell'} \langle \boldsymbol{x}_i, \boldsymbol{y}_j \rangle^2, \end{aligned} \tag{1}$$

which is k_q when $q = 2$. Thus, all information contained in the uncentered covariance matrices are preserved and utilized.

4 Mean polynomial kernel and Projection kernel relationship

We now present a connection between the proposed mean polynomial kernel above and the Projection kernel defined earlier. Typically, in the case of sequence recognition, Grassmann kernels are considered as kernel functions for principal subspaces of data points in sequences. To compute the value of the Projection kernel for two sequences $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$, we first perform eigendecomposition of two symmetric matrices $\boldsymbol{\Sigma}_x$ and $\boldsymbol{\Sigma}_y$. It can be shown that the m major eigenvectors are the bases of the m-dimensional principal subspaces. We obtain the value of the Projection kernel between the two principal subspaces by storing the m major eigenvectors in the columns of the $d \times m$ matrices $\boldsymbol{U}_x$ and $\boldsymbol{U}_y$.

The following equality constructs the connection between the mean polynomial kernel and the Projection kernel:

$$k_{PROJ}(\boldsymbol{U}_x, \boldsymbol{U}_y) = \left\langle \mathrm{vec}(\boldsymbol{\Sigma}'_x), \mathrm{vec}(\boldsymbol{\Sigma}'_y) \right\rangle, \qquad (2)$$

where we define $\boldsymbol{\Sigma}'_x = \boldsymbol{U}_x \boldsymbol{U}_x^\top$ and $\boldsymbol{\Sigma}'_y = \boldsymbol{U}_y \boldsymbol{U}_y^\top$. Indeed, if $\boldsymbol{\Sigma}'_x$ and $\boldsymbol{\Sigma}'_y$ are the uncentered covariance matrices of which the m major eigenvalues are replaced with one, and the rest of the eigenvalues are deleted, then we obtain

$$k_{PROJ}\big(\boldsymbol{U}_x, \boldsymbol{U}_y\big) = \big\|\boldsymbol{U}_x^\top \boldsymbol{U}_y\big\|_F^2 = \mathrm{tr}\big(\boldsymbol{U}_x^\top \boldsymbol{U}_y \boldsymbol{U}_y^\top \boldsymbol{U}_x\big)$$
$$= \mathrm{tr}\big(\boldsymbol{U}_x \boldsymbol{U}_x^\top \boldsymbol{U}_y \boldsymbol{U}_y^\top\big) = \mathrm{tr}\big(\boldsymbol{\Sigma}'_x \boldsymbol{\Sigma}'_y\big) = \left\langle \mathrm{vec}\big(\boldsymbol{\Sigma}'_x\big), \mathrm{vec}\big(\boldsymbol{\Sigma}'_y\big) \right\rangle.$$

Comparison between equations (1) and (2) implies that the Projection kernel potentially loses the information on the importance of each dimension of the principal subspaces and all the information on their orthogonal complements, whereas the second order mean polynomial kernel keeps all information in the uncentered covariance matrices. Hamm and Lee [6] extended the Projection kernel so that the information of the scale of each dimension in linear subspaces is preserved. However, their kernel still disregards the information in the orthogonal complement.

5 Face membership verification performance

5.1 Dataset and experimental settings

The MOBIO database [11] was used for the experiments. The database contains video data taken from video sessions divided into two: six sessions for Phase I and six sessions for Phase II. We only utilized data from 25 subjects and the six sessions from Phase I. Each session contains 21 image sequences of varying length. For the experiments, we set the sequence length to 25 images, where each image is a cropped facial image of the subject, obtained using a face detection program, transformed to gray scale and resized to 25×25 pixels. Among the 25 subjects, 10 were randomly selected and labeled as 'member' ($+1$), and the remaining 15 as 'nonmember' (-1).

Two methods were employed: one using kernels with SVM and the other one using GD-MSM. For the first method, three types of kernel functions were utilized: the Projection and Binet-Cauchy kernels, which are both Grassmann kernels, and the proposed kernel. For the GD-MSM, eight metrics were used for comparison: average distance, Binet-Cauchy metric, Geodesic distance, maximum correlation, minimum correlation, Frobenius norm based Procrustes distance, 2-norm based Procrustes distance, and Projection metric, as defined in [16]. For the SVM setting, 6-fold cross-validation was employed to evaluate the performance of the kernels such that one session per subject is used as test data while the remaining five sessions are used for training. For evaluating the performance of each method, the area under the ROC curve (AUC), accuracy, and F-measure values were obtained. Higher values of these performance measures indicate better quality of classifier.

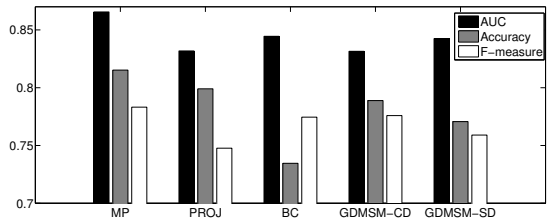


Figure 2. Average performance of all methods.

Similar to the polynomial kernel degree, the value of q of the mean polynomial kernel also controls the flexibility of the classifier. In the experiments, the values of q were only varied from one to five since, as in polynomial kernels, higher values may tend to overfit data. This value, together with the regularization parameter C for SVM, was set using a grid search, where $q \in \{1, \ldots, 5\}$ and $C \in \{10^0, 10^1, 10^2, 10^3, 10^4, 10^5\}$, by 3-fold cross-validation on the training data for each cross-validation set. The pair (q, C) was chosen such that the highest accuracy value is obtained. Similarly for the Grassmann kernels, the value of C and the dimension of the subspace were simultaneously determined using a grid search on all possible pairs (m, C), where m varies from one to ten. As for the mutual subspace method, the dimension of the subspace was varied from one to ten, and was selected as one yielding the highest accuracy using the training data.

5.2 Results

The average AUC, accuracy, and F-measure values for all six cross-validation sets are shown in Figure 2. The labels GDMSM-CD and GDMSM-SD correspond to the class and subject-wise dictionaries, respectively, while the first three methods are SVM used with the mean polynomial (MP), the Projection kernel (PROJ) and the Binet-Cauchy kernel (BC), respectively. The mean polynomial obtains the highest AUC, accuracy and F-measure values (0.866, 81.5% and 0.783, respectively) among all methods. The second best AUC value was obtained using BC kernel in GK-SVM at 0.845. The Projection kernel attained an accuracy rate of 79.9% following the proposed kernel. And an F-measure of 0.776 from employing the class-wise dictionary for GD-MSM was the second best among all methods. The results presented for the GD-MSM are the best among all eight metrics used, which is incidentally the maximum correlation for both methods using

class-wise and subject-wise dictionaries. Hence, it follows that the mean polynomial kernel performs better than the GD-MSM regardless of the metric used. In addition to this, it was also found that for most cross-validation sets, the BC kernel performs well when $m = 3$ and $C = 10^3$, the PROJ kernel when $m = 7$ and $C = 10^2$, and the proposed kernel when $q = 3$ and $C = 10^5$.

Another advantage of the proposed kernel is its low computational cost. The CPU time for the computation of the mean polynomial kernel matrix for any value of q is around 383 seconds. As for the Grassmann kernels, when $m = 5$, CPU time is around 1.21×10^4 seconds, and when $m = 10$, the Projection kernel matrix takes around 1.24×10^4 seconds, and 1.25×10^4 seconds for the Binet-Cauchy kernel matrix. Even for low values of m, processing time is at least 1.19×10^4 seconds. Moreover, should the dimension d of the image increase, their cost will also increase more drastically compared to the mean polynomial kernel.

6 Discussion

In this paper we proposed a new kernel for face membership authentication when data at hand is modeled as a subspace of the image space, such as data in the form of sequences of video frames or sequences of images. Data images were treated as matrices of pixel intensities and these values were used as data features.

We conclude this paper by discussing an extension of the mean polynomial kernel. An interesting extension can be obtained by replacing the sample mean of $\langle \boldsymbol{x}_i, \boldsymbol{y}_j \rangle^q$ with the expected value with respect to a probabilistic distribution: $k'_q(\boldsymbol{X}, \boldsymbol{Y}) = \mathbb{E}\big(\langle \boldsymbol{x}, \boldsymbol{y} \rangle^q \big)$. From this, the original mean polynomial kernel (Definition 2) can be derived as a special case when $p_x(\boldsymbol{x}) = \frac{1}{\ell} \sum_{i=1}^{\ell} \delta\big(\boldsymbol{x} - \boldsymbol{x}_i\big)$ and $p_y(\boldsymbol{y}) = \frac{1}{\ell'} \sum_{i=1}^{\ell'} \delta\big(\boldsymbol{y} - \boldsymbol{y}_i\big)$, where $\delta(\cdot)$ is the Dirac delta function.

Another choice of a probabilistic distribution can be Gaussian mixture. Suppose we are given the Gaussian mixture $p_z(\boldsymbol{z}) = \sum_{i=1}^{\ell} \pi_{z,i} \mathcal{N}\big(\boldsymbol{z}; \boldsymbol{\mu}_{z,i}, \boldsymbol{\Sigma}_{z,i}\big)$, where ℓ is the number of Gaussian components for the probabilistic distribution p_z, $\pi_{z,i}$ is the mixing coefficient satisfying $\sum_{i=1}^{n} \pi_{z,i} = 1$, and $\boldsymbol{\mu}_{z,i}$ and $\boldsymbol{\Sigma}_{z,i}$ are the mean vector and covariance matrix of the ith Gaussian component, respectively. The second order mean polynomial kernel can be readily computed as

$$k'_2(p_x, p_y) = \sum_{i=1}^{\ell} \sum_{j=1}^{\ell'} \pi_{x,i} \pi_{y,j} \Big(\big(\boldsymbol{\mu}_{x,i}^\top \boldsymbol{\mu}_{y,j} \big)^2 + \operatorname{tr}\big(\boldsymbol{\Sigma}_{x,i} \boldsymbol{\Sigma}_{y,j} \big) + \boldsymbol{\mu}_{x,i}^\top \boldsymbol{\Sigma}_{y,j} \boldsymbol{\mu}_{x,i} + \boldsymbol{\mu}_{y,j}^\top \boldsymbol{\Sigma}_{x,i} \boldsymbol{\mu}_{y,j} \Big).$$

This example includes the original definition of the mean polynomial kernel in Definition 2, which can be shown by letting

$$\pi_{x,i} = 1/\ell, \qquad \boldsymbol{\mu}_{x,i} = \boldsymbol{x}_i, \qquad \boldsymbol{\Sigma}_{x,i} = \sigma_{x,i}^2 \boldsymbol{I},$$
$$\pi_{y,j} = 1/\ell', \qquad \boldsymbol{\mu}_{y,j} = \boldsymbol{y}_j, \qquad \boldsymbol{\Sigma}_{y,j} = \sigma_{y,j}^2 \boldsymbol{I},$$

for all $i \in \mathbb{N}_\ell$ and $j \in \mathbb{N}_{\ell'}$, and taking the limit as $\sigma_{x,i}^2, \sigma_{y,j}^2 \to 0$. When one wishes to weight each frame in image sequences, the weights can be set to $\pi_{x,i}$ or $\pi_{y,j}$. Positive $\sigma_{x,i}^2$ or positive $\sigma_{y,j}^2$ can be used to represent uncertainties in observations. Such extensions bring interesting future work.

References

[1] V. Belhumeur, et al., *Eigenfaces vs. fisherfaces: Recognition using class specific linear projection*, IEEE Trans PAMI **19(7)** (July 1997), 711–720.

[2] O. Deniz, et al., *Face recognition using independent component analysis and support vector machines*, Pattern Recogn. Lett. **24** (2003), 2153–2157.

[3] K. Fukui and O. Yamaguchi, *Face recognition using multi-viewpoint pattern for robot vision*, Int. Symp. Robotics Research, 2003, pp. 192–201.

[4] G. Guo, et al., *Support vector machines for face recognition*, Image Vision Comput. (2001), 631–638.

[5] J. Hamm and D. Lee, *Grassmann discriminant analysis: a unifying view on subspace-based learning*, ICML, 2008, pp. 376–383.

[6] J. Hamm and D. Lee, *Extended Grassmann kernels for subspace-based learning*, NIPS, 2009.

[7] R. Jafri and H.R. Arabnia, *A survey of face recognition techniques*, JIPS (2009), 41–68.

[8] T.-K. Kim, et al., *Learning over sets using boosted manifold principal angles (BoMPA)*, British Machine Vision Conf., 2005, pp. 779–788.

[9] S.Z. Li, et al., *Kernel machine based learning for multi-view face detection and pose estimation*, ICCV, 2001, pp. 674–679.

[10] Y. Li, et al., *Support vector machine based multi-view face detection and recognition*, Image Vision Comput. (2004), 413–427.

[11] C. McCool, et al., *Bi-modal person recognition on a mobile phone: Using mobile phone data*, IEEE ICME Workshop on Hot Topics in Mobile Multimedia, 2012.

[12] M. Nishiyama, et al., *Face recognition with the multiple constrained mutual subspace method*, 5th Int. Conf. on Audio- and Video-based Biometric Person Authentication (AVBPA), 2005, pp. 71–80.

[13] H. Sakano and N. Mukawa, *Kernel mutual subspace method for robust facial image recognition*, Int. Conf. on Knowledge-Based Intell. Eng. Sys. And App. Tech, 2000, pp. 245–248.

[14] S. Satoh, *Comparative evaluation of face sequence matching for content-based video access*, Int. Conf. Automatic Face and Gesture Recognition, 2000, pp. 163–168.

[15] G. Shakhnarovich, et al., *Face recognition from long-term observations*, European Conf. Computer Vision (ECCV), 2002, pp. 851–868.

[16] R. Shigenaka, et al., *Face sequence recognition using Grassmann distances and Grassmann kernels*, IJCNN, 2012, pp. 1–7.

[17] M. Turk and A. Pentland, *Eigenfaces for recognition*, Journal of Cognitive Neuroscience **3(1)** (1991), 71–86.

[18] L. Wolf and A. Shashua, *Learning over sets using kernel principal angles*, J. Mach. Learn. Res. **4** (2003), 913–931.

[19] O. Yamaguchi, et al., *Face recognition using temporal image sequence*, Int. Conf. Automatic Face and Gesture Recognition, 1998, pp. 318–323.

[20] Q. Yang and X. Tang, *Recent advances in subspace analysis for face recognition*, SINOBIOMETRICS, 2004, pp. 275–287.