AARHUS
UNIVERSITET

Scientific Writing

Henrik Karstoft
@
Aarhus University, Department of Engineering

Content

Here we:

- We will briefly describe what is a (scientific) report
- We will give some guidelines on how to write such a report
- We will follow the ways proposed/used in some of the most popular scientific conferences/journals of Computer Vision

Keep in mind that:

- The way that a report is written varies in different disciplines, depending on the type of information it should convey to the reader
- Very similar content can be written in different ways, depending on the templates/norms followed by different conferences and journals. This means that what is important is the information we want to convey and not the exact specific way that the paper is structured!

What is a paper/report?

A commonly used structure:

- Title
- Abstract
- Introduction
- Theoretical Analysis – Method description
- Experiments and results
- Discussion (can be part of experimental section)
- Conclusions and Summary
- Appendix (many times it is omitted)
- References

What is a paper/report?

Examples:

Pattern Recognition 74 (2018) 359–372

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Probabilistic saliency estimation

Caglar Aytekin*, Alexandros Iosifidis, Moncef Gabbouj

Department of Signal Processing, Tampere University of Technology, Tampere, Finland

ARTICLE INFO

Article history:
Received 30 January 2017
Revised 20 August 2017
Accepted 12 September 2017
Available online 20 September 2017

Keywords:
Saliency
Salient object detection
Spectral graph cut
Diffusion maps
Probabilistic model
One-class classification

ABSTRACT

In this paper, we model the salient object detection problem under a probabilistic framework encoding the boundary connectivity saliency cue and smoothness constraints into an optimization problem. We show that this problem has a closed form global optimum solution, which estimates the salient object. We further show that along with the probabilistic framework, the proposed method also enjoys a wide range of interpretations, i.e. graph cut, diffusion maps and one-class classification. With an analysis according to these interpretations, we also find that our proposed method provides approximations to the global optimum to another criterion that integrates local/global contrast and large area saliency cues. The proposed unsupervised approach achieves mostly leading performance compared to the state-of-the-art unsupervised algorithms over a large set of salient object detection datasets including around 17k images for several evaluation metrics. Furthermore, the computational complexity of the proposed method is favorable/comparable to many state-of-the-art unsupervised techniques.

1. Introduction

Salient object detection is a computer vision topic with growing interest over the last decade. The goal is to highlight the visually interesting regions in a given scene. The problem of saliency detection has been motivated by related work in neuroscience; humans select important visual information based on attention mechanisms in the brain [1]. Given this motivation, earlier works on saliency detection concentrate more on predicting sparse human eye-gaze points that are detected by eye-trackers [2]. Accordingly, most of the research on this track is based on biologically inspired algorithms, which try to imitate the dynamics of the human attention mechanism [2–9]. During the last few years another related track emerged where the goal is to segment *salient objects* [10–20], instead of predicting some sparse eye-fixations. Both research tracks produce saliency maps that are useful for tasks such as video surveillance [21], compression [22], image manipulation [23], automatic image cropping [24], foreground detection [25], and coding [26]. However, the output of salient object detector techniques is more useful, when compared to eye fixation predictions, for higher level computer vision and pattern recognition tasks such as tracking [27], object region proposals [29] and object recognition [28].

In this paper, we focus on the salient object detection task. Since ultimately we consider salient object detection as a preprocessing block for higher-level tasks as mentioned above, fast and generic methods would be preferred. Therefore, in this work we focus on unsupervised salient object detection. While supervised approaches, such as those in [42,33], have the potential of finding more accurate results, their performance depends on the training process followed and the data that has been exploited for training. Recent works have also indicated that unsupervised saliency detection approaches can compete (or even outperform) supervised methods [40].

Unsupervised salient object detection methods can be categorized based on the saliency cues they use. Commonly exploited cues include local and global contrast, boundary connectivity, shape and location cues. The local contrast cue is based on the assumption that the salient object is in contrast with its immediate surroundings [11,12,15,20]. A spectral foreground detection method was proposed in [20], which optimizes a criterion involving the minimization of cut-value, which is equivalent to the maximization of the local contrast. A region contrast based method was proposed in [11], which computes a salient region as a weighted sum of local contrasts with its surroundings. The global contrast cue is similarly defined as by assuming that the salient object is in high contrast with its surrounding. A histogram-based method enhancing regions with global contrast to the rest of the image was proposed in [11]. The boundary connectivity prior is one of the most widely used cues and is based on the assumption that most of

IEEE TRANSACTIONS ON CYBERNETICS, VOL. 47, NO. 12, DECEMBER 2017

Class-Specific Kernel Discriminant Analysis Revisited: Further Analysis and Extensions

Alexandros Iosifidis, *Senior Member, IEEE*, and Moncef Gabbouj, *Fellow, IEEE*

***Abstract*—In this paper, we revisit class-specific kernel discriminant analysis (KDA) formulation, which has been applied in various problems, such as human face verification and human action recognition. We show that the original optimization problem solved for the determination of class-specific discriminant projections is equivalent to a low-rank kernel regression (LRKR) problem using training data-independent target vectors. In addition, we show that the regularized version of class-specific KDA is equivalent to a regularized LRKR problem, exploiting the same targets. This analysis allows us to devise a novel fast solution. Furthermore, we devise novel incremental, approximate and deep (hierarchical) variants. The proposed methods are tested in human facial image and action video verification problems, where their effectiveness and efficiency is shown.**

***Index Terms*—Approximation, class-specific kernel discriminant analysis (CSKDA), incremental learning, low-rank kernel regression (LRKR), regularization.**

I. Introduction

SUBSPACE learning techniques have an important role in statistical machine learning, due to the well-known *curse of dimensionality* [1], which states that when the data dimensionality increases, the volume of the data representation space increases and the available (training) data become sparse. In such high-dimensional spaces, learning statistical models requires the utilization of an enormous amount of training data. The most well-known and commonly employed subspace learning technique is principal component analysis (PCA) [2], which defines a subspace for data projection that preserves most of the available information (in the sense of minimal l_2 norm-based reconstruction error). While PCA is able to determine the directions of the highest variance of the original high-dimensional feature space, it is an unsupervised technique and it is not able to increase class discrimination.

When the objective is to determine a subspace that enhances class discrimination, supervised techniques exploiting the labeling information available for the training data are employed. Linear discriminant analysis (LDA) [3]–[8] is one of the most well-studied techniques and has been found very effective in many applications [9]–[12]. By assuming unimodal classes following normal distributions, the optimal discriminant subspace is obtained by maximizing the so-called Fisher ratio which is defined on the between-class and within-class scatter matrices. Several variants that overcome the unimodality assumption of LDA have also been proposed [13]–[16]. In addition, regression models exploiting the within-class and between-class scatter defined in LDA have been recently proposed [17]. Another important limitation of LDA is the fact that the maximal subspace dimensionality is restricted by the number of classes C forming the problem at hand. This is due to the fact that the rank of the between-class scatter matrix is at most equal to $C-1$. A direct consequence of this is that in binary (two-class) problems, LDA and its variants are able to determine a subspace formed by only one dimension, which might not be the optimal choice for class discrimination.

In order to overcome the latter limitation of LDA, class-specific approaches have been proposed [18]–[22]. In class-specific subspace learning techniques, the objective is to determine the optimal feature space highlighting the discrimination between a given class of interest (noted as positive class hereafter) from the remaining world (i.e., the data not belonging to the positive class, forming the hereafter called negative class). An advantage of this approach is that, by defining appropriate out-of-class and intraclass scatter matrices, the maximal dimensionality of the learned subspace is not restricted by the number of classes, but can be up to the number of samples forming the class under consideration and, thus, feature spaces of higher dimensionality (when compared to LDA) can be determined. This in turn leads to better class discrimination and better performance [19], [21], [22].

For the cases where nonlinear discrimination criteria lead to better performance, both multiclass and class-specific subspace learning techniques can be extended to their nonlinear counterparts by exploiting the well-known *kernel trick* [23]. Kernel extensions exploit the so-called *kernel function* $\kappa(\cdot,\cdot)$ defined on pairs of D-dimensional data $\{\mathbf{x}_i, \mathbf{x}_j\}$ and expressing dot products of the data representations in the so-called *feature space* $\mathcal{F}$, i.e., $\kappa(\mathbf{x}_i, \mathbf{x}_j) \triangleq \boldsymbol{\phi}(\mathbf{x}_i)^T \boldsymbol{\phi}(\mathbf{x}_j)$, where $\boldsymbol{\phi}(\cdot)$ is a nonlinear function such that $\mathbf{x}_i \in \mathbb{R}^D \rightarrow \boldsymbol{\phi}(\mathbf{x}_i) \in \mathcal{F}$. The application of a linear subspace learning technique in $\mathcal{F}$ corresponds to

Manuscript received May 19, 2016; revised July 28, 2016; accepted September 18, 2016. Date of publication October 13, 2016; date of current version November 15, 2017. This work was supported by the Academy of Finland Post-Doctoral Research Fellowship under Grant 295854. This paper was recommended by Associate Editor X. Li.

The authors are with the Department of Signal Processing, Tampere University of Technology, FIN-33720 Tampere, Finland (e-mail:

What is a paper/report?

Examples:

Title, authors and affiliations

Pattern Recognition 74 (2018) 359–372

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Probabilistic saliency estimation

Caglar Aytekin*, Alexandros Iosifidis, Moncef Gabbouj

Department of Signal Processing, Tampere University of Technology, Tampere, Finland

ARTICLE INFO

Article history:
Received 30 January 2017
Revised 20 August 2017
Accepted 12 September 2017
Available online 20 September 2017

Keywords:
Saliency
Salient object detection
Spectral graph cut
Diffusion maps
Probabilistic model
One-class classification

ABSTRACT

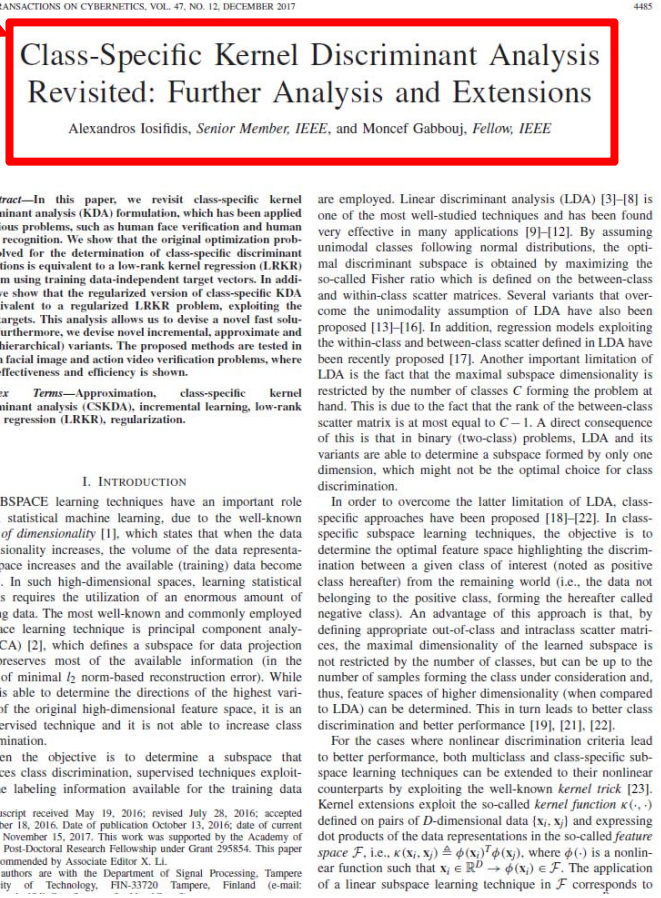
In this paper, we model the salient object detection problem under a probabilistic framework encoding the boundary connectivity saliency cue and smoothness constraints into an optimization problem. We show that this problem has a closed form global optimum solution, which estimates the salient object. We further show that along with the probabilistic framework, the proposed method also enjoys a wide range of interpretations, i.e. graph cut, diffusion maps and one-class classification. With an analysis according to these interpretations, we also find that our proposed method provides approximations to the global optimum to another criterion that integrates local/global contrast and large area saliency cues. The proposed unsupervised approach achieves mostly leading performance compared to the state-of-the-art unsupervised algorithms over a large set of salient object detection datasets including around 17k images for several evaluation metrics. Furthermore, the computational complexity of the proposed method is favorable/comparable to many state-of-the-art unsupervised techniques.

1. Introduction

Salient object detection is a computer vision topic with growing interest over the last decade. The goal is to highlight the visually interesting regions in a given scene. The problem of saliency detection has been motivated by related work in neuroscience; humans select important visual information based on attention mechanisms in the brain [1]. Given this motivation, earlier works on saliency detection concentrate more on predicting sparse human eye-gaze points that are detected by eye-trackers [2]. Accordingly, most of the research on this track is based on biologically inspired algorithms, which try to imitate the dynamics of the human attention mechanism [2–9]. During the last few years another related track emerged where the goal is to segment *salient objects* [10–20], instead of predicting some sparse eye-fixations. Both research tracks produce saliency maps that are useful for tasks such as video surveillance [21], compression [22], image manipulation [23], automatic image cropping [24], foreground detection [25], and coding [26]. However, the output of salient object detector techniques is more useful, when compared to eye fixation predictions, for higher level computer vision and pattern recognition tasks such as tracking [27], object region proposals [29] and object recognition [28].

In this paper, we focus on the salient object detection task. Since ultimately we consider salient object detection as a preprocessing block for higher-level tasks as mentioned above, fast and generic methods would be preferred. Therefore, in this work we focus on unsupervised salient object detection. While supervised approaches, such as those in [42,33], have the potential of finding more accurate results, their performance depends on the training process followed and the data that has been exploited for training. Recent works have also indicated that unsupervised saliency detection approaches can compete (or even outperform) supervised methods [40].

Unsupervised salient object detection methods can be categorized based on the saliency cues they use. Commonly exploited cues include local and global contrast, boundary connectivity, shape and location cues. The local contrast cue is based on the assumption that the salient object is in contrast with its immediate surroundings [11,12,15,20]. A spectral foreground detection method was proposed in [20], which optimizes a criterion involving the minimization of cut-value, which is equivalent to the maximization of the local contrast. A region contrast based method was proposed in [11], which computes a salient region as a weighted sum of local contrasts with its surroundings. The global contrast cue is similarly defined as by assuming that the salient object is in high contrast with its surrounding. A histogram-based method enhancing regions with global contrast to the rest of the image was proposed in [11]. The boundary connectivity prior is one of the most widely used cues and is based on the assumption that most of



Class-Specific Kernel Discriminant Analysis Revisited: Further Analysis and Extensions

Alexandros Iosifidis, *Senior Member, IEEE*, and Moncef Gabbouj, *Fellow, IEEE*

***Abstract*—In this paper, we revisit class-specific kernel discriminant analysis (KDA) formulation, which has been applied in various problems, such as human face verification and human action recognition. We show that the original optimization problem solved for the determination of class-specific discriminant projections is equivalent to a low-rank kernel regression (LRKR) problem using training data-independent target vectors. In addition, we show that the regularized version of class-specific KDA is equivalent to a regularized LRKR problem, exploiting the same targets. This analysis allows us to devise a novel fast solution. Furthermore, we devise novel incremental, approximate and deep (hierarchical) variants. The proposed methods are tested in human facial image and action video verification problems, where their effectiveness and efficiency is shown.**

***Index Terms*—Approximation, class-specific kernel discriminant analysis (CSKDA), incremental learning, low-rank kernel regression (LRKR), regularization.**

I. Introduction

SUBSPACE learning techniques have an important role in statistical machine learning, due to the well-known *curse of dimensionality* [1], which states that when the data dimensionality increases, the volume of the data representation space increases and the available (training) data become sparse. In such high-dimensional spaces, learning statistical models requires the utilization of an enormous amount of training data. The most well-known and commonly employed subspace learning technique is principal component analysis (PCA) [2], which defines a subspace for data projection that preserves most of the available information (in the sense of minimal l_2 norm-based reconstruction error). While PCA is able to determine the directions of the highest variance of the original high-dimensional feature space, it is an unsupervised technique and it is not able to increase class discrimination.

When the objective is to determine a subspace that enhances class discrimination, supervised techniques exploiting the labeling information available for the training data are employed. Linear discriminant analysis (LDA) [3]–[8] is one of the most well-studied techniques and has been found very effective in many applications [9]–[12]. By assuming unimodal classes following normal distributions, the optimal discriminant subspace is obtained by maximizing the so-called Fisher ratio which is defined on the between-class and within-class scatter matrices. Several variants that overcome the unimodality assumption of LDA have also been proposed [13]–[16]. In addition, regression models exploiting the within-class and between-class scatter defined in LDA have been recently proposed [17]. Another important limitation of LDA is the fact that the maximal subspace dimensionality is restricted by the number of classes C forming the problem at hand. This is due to the fact that the rank of the between-class scatter matrix is at most equal to $C-1$. A direct consequence of this is that in binary (two-class) problems, LDA and its variants are able to determine a subspace formed by only one dimension, which might not be the optimal choice for class discrimination.

In order to overcome the latter limitation of LDA, class-specific approaches have been proposed [18]–[22]. In class-specific subspace learning techniques, the objective is to determine the optimal feature space highlighting the discrimination between a given class of interest (noted as positive class hereafter) from the remaining world (i.e., the data not belonging to the positive class, forming the hereafter called negative class). An advantage of this approach is that, by defining appropriate out-of-class and intraclass scatter matrices, the maximal dimensionality of the learned subspace is not restricted by the number of classes, but can be up to the number of samples forming the class under consideration and, thus, feature spaces of higher dimensionality (when compared to LDA) can be determined. This in turn leads to better class discrimination and better performance [19], [21], [22].

For the cases where nonlinear discrimination criteria lead to better performance, both multiclass and class-specific subspace learning techniques can be extended to their nonlinear counterparts by exploiting the well-known *kernel trick* [23]. Kernel extensions exploit the so-called *kernel function* $\kappa(\cdot,\cdot)$ defined on pairs of D-dimensional data $\{\mathbf{x}_i, \mathbf{x}_j\}$ and expressing dot products of the data representations in the so-called *feature space* $\mathcal{F}$, i.e., $\kappa(\mathbf{x}_i, \mathbf{x}_j) \triangleq \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$, where $\phi(\cdot)$ is a nonlinear function such that $\mathbf{x}_i \in \mathbb{R}^D \rightarrow \phi(\mathbf{x}_i) \in \mathcal{F}$. The application of a linear subspace learning technique in $\mathcal{F}$ corresponds to

Manuscript received May 19, 2016; revised July 28, 2016; accepted September 18, 2016. Date of publication October 13, 2016; date of current version November 15, 2017. This work was supported by the Academy of Finland Post-Doctoral Research Fellowship under Grant 295854. This paper was recommended by Associate Editor X. Li.

The authors are with the Department of Signal Processing, Tampere University of Technology, FIN-33720 Tampere, Finland (e-mail:

Parts of a paper

Abstract:

- Gives a summary of the whole report
- Outlines -purpose, research method, findings, main conclusions and recommendations
- Mainly past tense
- Usually written last!

Keywords:

- Buzzwords summarizing the fields of work
- Usually used for searching/grouping papers
- Usually 3-8 keywords

IEEE TRANSACTIONS ON CYBERNETICS, VOL. 47, NO. 12, DECEMBER 2017

Class-Specific Kernel Discriminant Analysis Revisited: Further Analysis and Extensions

Alexandros Iosifidis, *Senior Member, IEEE*, and Moncef Gabbouj, *Fellow, IEEE*

***Abstract*—In this paper, we revisit class-specific kernel discriminant analysis (KDA) formulation, which has been applied in various problems, such as human face verification and human action recognition. We show that the original optimization problem solved for the determination of class-specific discriminant projections is equivalent to a low-rank kernel regression (LRKR) problem using training data-independent target vectors. In addition, we show that the regularized version of class-specific KDA is equivalent to a regularized LRKR problem, exploiting the same targets. This analysis allows us to devise a novel fast solution. Furthermore, we devise novel incremental, approximate and deep (hierarchical) variants. The proposed methods are tested in human facial image and action video verification problems, where their effectiveness and efficiency is shown.**

***Index Terms*—Approximation, class-specific kernel discriminant analysis (CSKDA), incremental learning, low-rank kernel regression (LRKR), regularization.**

I. Introduction

SUBSPACE learning techniques have an important role in statistical machine learning, due to the well-known *curse of dimensionality* [1], which states that when the data dimensionality increases, the volume of the data representation space increases and the available (training) data become sparse. In such high-dimensional spaces, learning statistical models requires the utilization of an enormous amount of training data. The most well-known and commonly employed subspace learning technique is principal component analysis (PCA) [2], which defines a subspace for data projection that preserves most of the available information (in the sense of minimal l_2 norm-based reconstruction error). While PCA is able to determine the directions of the highest variance of the original high-dimensional feature space, it is an unsupervised technique and it is not able to increase class discrimination.

When the objective is to determine a subspace that enhances class discrimination, supervised techniques exploiting the labeling information available for the training data

Manuscript received May 19, 2016; revised July 28, 2016; accepted September 18, 2016. Date of publication October 13, 2016; date of current version November 15, 2017. This work was supported by the Academy of Finland Post-Doctoral Research Fellowship under Grant 295854. This paper was recommended by Associate Editor X. Li.

The authors are with the Department of Signal Processing, Tampere University of Technology, FIN-33720 Tampere, Finland (e-mail:

re employed. Linear discriminant analysis (LDA) [3]–[8] is ne of the most well-studied techniques and has been found ery effective in many applications [9]–[12]. By assuming nimodal classes following normal distributions, the optiial discriminant subspace is obtained by maximizing the o-called Fisher ratio which is defined on the between-class nd within-class scatter matrices. Several variants that overome the unimodality assumption of LDA have also been roposed [13]–[16]. In addition, regression models exploiting he within-class and between-class scatter defined in LDA have een recently proposed [17]. Another important limitation of DA is the fact that the maximal subspace dimensionality is estricted by the number of classes C forming the problem at and. This is due to the fact that the rank of the between-class catter matrix is at most equal to $C-1$. A direct consequence of this is that in binary (two-class) problems, LDA and its variants are able to determine a subspace formed by only one dimension, which might not be the optimal choice for class discrimination.

In order to overcome the latter limitation of LDA, class-specific approaches have been proposed [18]–[22]. In class-specific subspace learning techniques, the objective is to determine the optimal feature space highlighting the discrimination between a given class of interest (noted as positive class hereafter) from the remaining world (i.e., the data not belonging to the positive class, forming the hereafter called negative class). An advantage of this approach is that, by defining appropriate out-of-class and intraclass scatter matrices, the maximal dimensionality of the learned subspace is not restricted by the number of classes, but can be up to the number of samples forming the class under consideration and, thus, feature spaces of higher dimensionality (when compared to LDA) can be determined. This in turn leads to better class discrimination and better performance [19], [21], [22].

For the cases where nonlinear discrimination criteria lead to better performance, both multiclass and class-specific subspace learning techniques can be extended to their nonlinear counterparts by exploiting the well-known *kernel trick* [23]. Kernel extensions exploit the so-called *kernel function* $\kappa(\cdot,\cdot)$ defined on pairs of D-dimensional data $\{\mathbf{x}_i, \mathbf{x}_j\}$ and expressing dot products of the data representations in the so-called *feature space* $\mathcal{F}$, i.e., $\kappa(\mathbf{x}_i, \mathbf{x}_j) \triangleq \boldsymbol{\phi}(\mathbf{x}_i)^T \boldsymbol{\phi}(\mathbf{x}_j)$, where $\boldsymbol{\phi}(\cdot)$ is a nonlinear function such that $\mathbf{x}_i \in \mathbb{R}^D \rightarrow \boldsymbol{\phi}(\mathbf{x}_i) \in \mathcal{F}$. The application of a linear subspace learning technique in $\mathcal{F}$ corresponds to

Parts of a paper

Introduction:

- Outlines context, background and purpose of the work
- Defines terms and sets limits of the research
- The reader/audience can easily identify what, how, why
- Provides literature review in text form, along with limitations of the existing works that the current work addresses
- Mainly uses past tense and can be written later although presented first

IEEE TRANSACTIONS ON CYBERNETICS, VOL. 47, NO. 12, DECEMBER 2017

Class-Specific Kernel Discriminant Analysis Revisited: Further Analysis and Extensions

Alexandros Iosifidis, *Senior Member, IEEE*, and Moncef Gabbouj, *Fellow, IEEE*

***Abstract*—In this paper, we revisit class-specific kernel discriminant analysis (KDA) formulation, which has been applied in various problems, such as human face verification and human action recognition. We show that the original optimization problem solved for the determination of class-specific discriminant projections is equivalent to a low-rank kernel regression (LRKR) problem using training data-independent target vectors. In addition, we show that the regularized version of class-specific KDA is equivalent to a regularized LRKR problem, exploiting the same targets. This analysis allows us to devise a novel fast solution. Furthermore, we devise novel incremental, approximate and deep (hierarchical) variants. The proposed methods are tested in human facial image and action video verification problems, where their effectiveness and efficiency is shown.**

***Index Terms*—Approximation, class-specific kernel discriminant analysis (CSKDA), incremental learning, low-rank kernel regression (LRKR), regularization.**

I. Introduction

SUBSPACE learning techniques have an important role in statistical machine learning, due to the well-known *curse of dimensionality* [1], which states that when the data dimensionality increases, the volume of the data representation space increases and the available (training) data become sparse. In such high-dimensional spaces, learning statistical models requires the utilization of an enormous amount of training data. The most well-known and commonly employed subspace learning technique is principal component analysis (PCA) [2], which defines a subspace for data projection that preserves most of the available information (in the sense of minimal l_2 norm-based reconstruction error). While PCA is able to determine the directions of the highest variance of the original high-dimensional feature space, it is an unsupervised technique and it is not able to increase class discrimination.

When the objective is to determine a subspace that enhances class discrimination, supervised techniques exploiting the labeling information available for the training data are employed. Linear discriminant analysis (LDA) [3]–[8] is one of the most well-studied techniques and has been found very effective in many applications [9]–[12]. By assuming unimodal classes following normal distributions, the optimal discriminant subspace is obtained by maximizing the so-called Fisher ratio which is defined on the between-class and within-class scatter matrices. Several variants that overcome the unimodality assumption of LDA have also been proposed [13]–[16]. In addition, regression models exploiting the within-class and between-class scatter defined in LDA have been recently proposed [17]. Another important limitation of LDA is the fact that the maximal subspace dimensionality is restricted by the number of classes C forming the problem at hand. This is due to the fact that the rank of the between-class scatter matrix is at most equal to $C-1$. A direct consequence of this is that in binary (two-class) problems, LDA and its variants are able to determine a subspace formed by only one dimension, which might not be the optimal choice for class discrimination.

In order to overcome the latter limitation of LDA, class-specific approaches have been proposed [18]–[22]. In class-specific subspace learning techniques, the objective is to determine the optimal feature space highlighting the discrimination between a given class of interest (noted as positive class hereafter) from the remaining world (i.e., the data not belonging to the positive class, forming the hereafter called negative class). An advantage of this approach is that, by defining appropriate out-of-class and intraclass scatter matrices, the maximal dimensionality of the learned subspace is not restricted by the number of classes, but can be up to the number of samples forming the class under consideration and, thus, feature spaces of higher dimensionality (when compared to LDA) can be determined. This in turn leads to better class discrimination and better performance [19], [21], [22].

For the cases where nonlinear discrimination criteria lead to better performance, both multiclass and class-specific subspace learning techniques can be extended to their nonlinear counterparts by exploiting the well-known *kernel trick* [23]. Kernel extensions exploit the so-called *kernel function* $\kappa(\cdot,\cdot)$ defined on pairs of D-dimensional data $\{\mathbf{x}_i, \mathbf{x}_j\}$ and expressing dot products of the data representations in the so-called *feature space* $\mathcal{F}$, i.e., $\kappa(\mathbf{x}_i, \mathbf{x}_j) \triangleq \boldsymbol{\phi}(\mathbf{x}_i)^T \boldsymbol{\phi}(\mathbf{x}_j)$, where $\boldsymbol{\phi}(\cdot)$ is a nonlinear function such that $\mathbf{x}_i \in \mathbb{R}^D \rightarrow \boldsymbol{\phi}(\mathbf{x}_i) \in \mathcal{F}$. The application of a linear subspace learning technique in $\mathcal{F}$ corresponds to

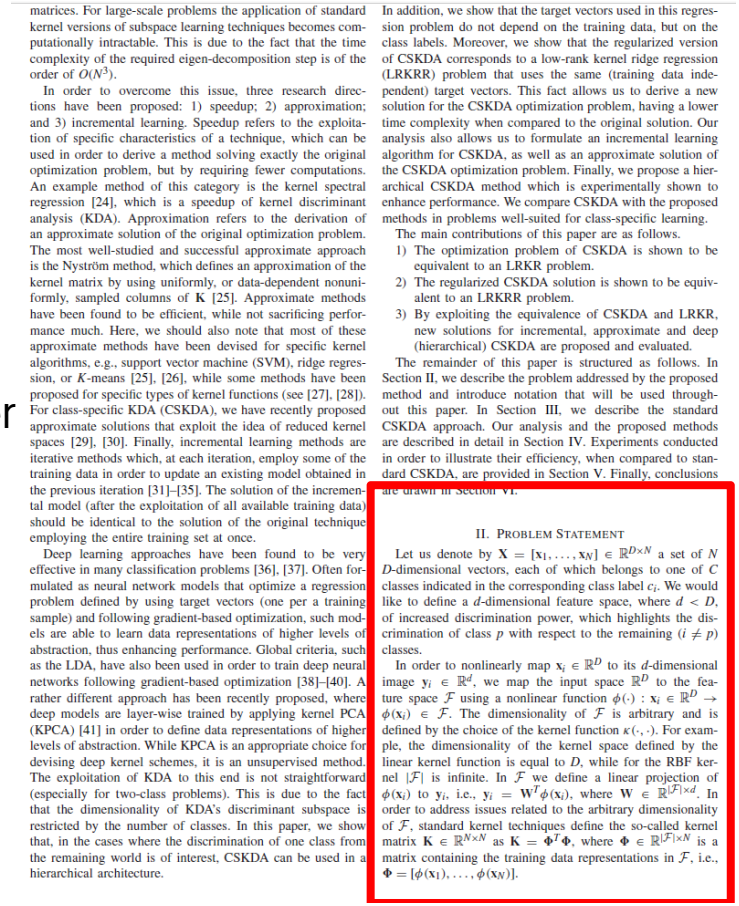
Manuscript received May 19, 2016; revised July 28, 2016; accepted September 18, 2016. Date of publication October 13, 2016; date of current version November 15, 2017. This work was supported by the Academy of Finland Post-Doctoral Research Fellowship under Grant 295854. This paper was recommended by Associate Editor X. Li.

The authors are with the Department of Signal Processing, Tampere University of Technology, FIN-33720 Tampere, Finland (e-mail:

Parts of a paper

Problem statement:

- In case of methodological/theoretical work, it can be used to describe the problem that is targeted in the work
- It provides notations and definitions that will be used later in the paper
- It can be omitted (but the definitions and notations will be in any case provided in later sections)



matrices. For large-scale problems the application of standard kernel versions of subspace learning techniques becomes computationally intractable. This is due to the fact that the time complexity of the required eigen-decomposition step is of the order of $O(N^3)$.

In order to overcome this issue, three research directions have been proposed: 1) speedup; 2) approximation; and 3) incremental learning. Speedup refers to the exploitation of specific characteristics of a technique, which can be used in order to derive a method solving exactly the original optimization problem, but by requiring fewer computations. An example method of this category is the kernel spectral regression [24], which is a speedup of kernel discriminant analysis (KDA). Approximation refers to the derivation of an approximate solution of the original optimization problem. The most well-studied and successful approximate approach is the Nyström method, which defines an approximation of the kernel matrix by using uniformly, or data-dependent nonuniformly, sampled columns of $\mathbf{K}$ [25]. Approximate methods have been found to be efficient, while not sacrificing performance much. Here, we should also note that most of these approximate methods have been devised for specific kernel algorithms, e.g., support vector machine (SVM), ridge regression, or K-means [25], [26], while some methods have been proposed for specific types of kernel functions (see [27], [28]). For class-specific KDA (CSKDA), we have recently proposed approximate solutions that exploit the idea of reduced kernel spaces [29], [30]. Finally, incremental learning methods are iterative methods which, at each iteration, employ some of the training data in order to update an existing model obtained in the previous iteration [31]–[35]. The solution of the incremental model (after the exploitation of all available training data) should be identical to the solution of the original technique employing the entire training set at once.

Deep learning approaches have been found to be very effective in many classification problems [36], [37]. Often formulated as neural network models that optimize a regression problem defined by using target vectors (one per a training sample) and following gradient-based optimization, such models are able to learn data representations of higher levels of abstraction, thus enhancing performance. Global criteria, such as the LDA, have also been used in order to train deep neural networks following gradient-based optimization [38]–[40]. A rather different approach has been recently proposed, where deep models are layer-wise trained by applying kernel PCA (KPCA) [41] in order to define data representations of higher levels of abstraction. While KPCA is an appropriate choice for devising deep kernel schemes, it is an unsupervised method. The exploitation of KDA to this end is not straightforward (especially for two-class problems). This is due to the fact that the dimensionality of KDA's discriminant subspace is restricted by the number of classes. In this paper, we show that, in the cases where the discrimination of one class from the remaining world is of interest, CSKDA can be used in a hierarchical architecture.

In addition, we show that the target vectors used in this regression problem do not depend on the training data, but on the class labels. Moreover, we show that the regularized version of CSKDA corresponds to a low-rank kernel ridge regression (LRKRR) problem that uses the same (training data independent) target vectors. This fact allows us to derive a new solution for the CSKDA optimization problem, having a lower time complexity when compared to the original solution. Our analysis also allows us to formulate an incremental learning algorithm for CSKDA, as well as an approximate solution of the CSKDA optimization problem. Finally, we propose a hierarchical CSKDA method which is experimentally shown to enhance performance. We compare CSKDA with the proposed methods in problems well-suited for class-specific learning.

The main contributions of this paper are as follows.

1) The optimization problem of CSKDA is shown to be equivalent to an LRKR problem.
2) The regularized CSKDA solution is shown to be equivalent to an LRKRR problem.
3) By exploiting the equivalence of CSKDA and LRKR, new solutions for incremental, approximate and deep (hierarchical) CSKDA are proposed and evaluated.

The remainder of this paper is structured as follows. In Section II, we describe the problem addressed by the proposed method and introduce notation that will be used throughout this paper. In Section III, we describe the standard CSKDA approach. Our analysis and the proposed methods are described in detail in Section IV. Experiments conducted in order to illustrate their efficiency, when compared to standard CSKDA, are provided in Section V. Finally, conclusions are drawn in Section VI.

II. Problem Statement

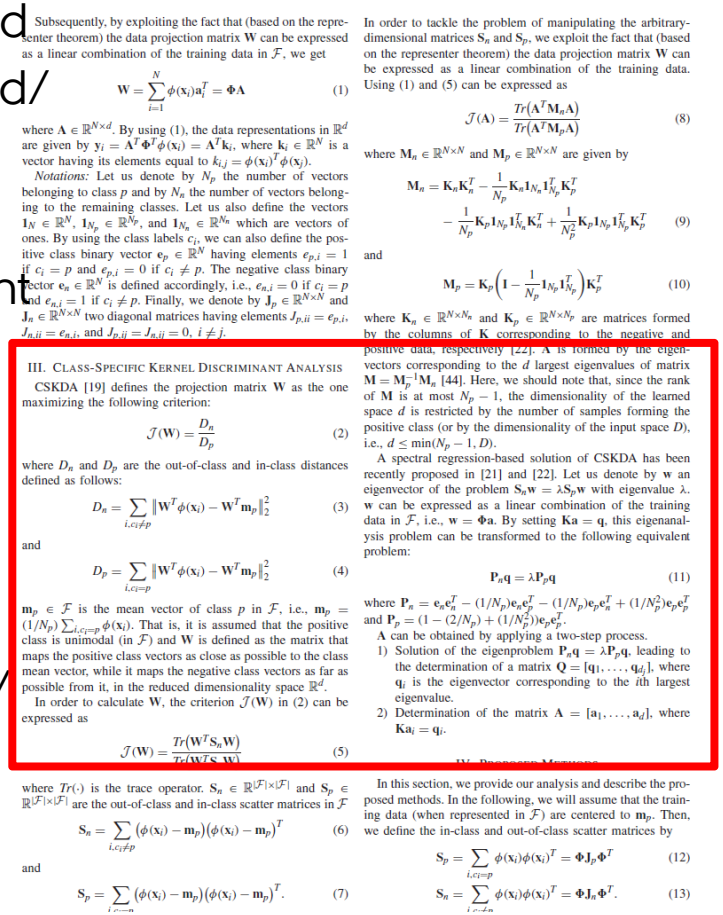
Let us denote by $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N] \in \mathbb{R}^{D \times N}$ a set of N D-dimensional vectors, each of which belongs to one of C classes indicated in the corresponding class label c_i. We would like to define a d-dimensional feature space, where $d < D$, of increased discrimination power, which highlights the discrimination of class p with respect to the remaining $(i \neq p)$ classes.

In order to nonlinearly map $\mathbf{x}_i \in \mathbb{R}^D$ to its d-dimensional image $\mathbf{y}_i \in \mathbb{R}^d$, we map the input space $\mathbb{R}^D$ to the feature space $\mathcal{F}$ using a nonlinear function $\phi(\cdot) : \mathbf{x}_i \in \mathbb{R}^D \rightarrow \phi(\mathbf{x}_i) \in \mathcal{F}$. The dimensionality of $\mathcal{F}$ is arbitrary and is defined by the choice of the kernel function $\kappa(\cdot, \cdot)$. For example, the dimensionality of the kernel space defined by the linear kernel function is equal to D, while for the RBF kernel $|\mathcal{F}|$ is infinite. In $\mathcal{F}$ we define a linear projection of $\phi(\mathbf{x}_i)$ to $\mathbf{y}_i$, i.e., $\mathbf{y}_i = \mathbf{W}^T \phi(\mathbf{x}_i)$, where $\mathbf{W} \in \mathbb{R}^{|\mathcal{F}| \times d}$. In order to address issues related to the arbitrary dimensionality of $\mathcal{F}$, standard kernel techniques define the so-called kernel matrix $\mathbf{K} \in \mathbb{R}^{N \times N}$ as $\mathbf{K} = \mathbf{\Phi}^T \mathbf{\Phi}$, where $\mathbf{\Phi} \in \mathbb{R}^{|\mathcal{F}| \times N}$ is a matrix containing the training data representations in $\mathcal{F}$, i.e., $\mathbf{\Phi} = [\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_N)]$.

Parts of a paper

Related work:

- It provides a detailed description of the related techniques/methods that have been proposed/ used until now
- It is used to give to the reader a relatively good knowledge on the problem and highlight which are the specific roles that necessitate the existence of the current work
- It can be mathematical (describing the math behind the existing solutions) when the work is technical/theoretical
- It can be text describing existing approaches/ solutions when it is a methodological work. In such cases it can be omitted, if this discussion is included in the introduction.



Subsequently, by exploiting the fact that (based on the representer theorem) the data projection matrix $\mathbf{W}$ can be expressed as a linear combination of the training data in $\mathcal{F}$, we get

$$\mathbf{W} = \sum_{i=1}^{N} \phi(\mathbf{x}_i)\mathbf{a}_i^T = \mathbf{\Phi}\mathbf{A} \quad (1)$$

where $\mathbf{A} \in \mathbb{R}^{N\times d}$. By using (1), the data representations in $\mathbb{R}^d$ are given by $\mathbf{y}_i = \mathbf{A}^T\mathbf{\Phi}^T\phi(\mathbf{x}_i) = \mathbf{A}^T\mathbf{k}_i$, where $\mathbf{k}_i \in \mathbb{R}^N$ is a vector having its elements equal to $k_{i,j} = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$.

Notations: Let us denote by N_p the number of vectors belonging to class p and by N_n the number of vectors belonging to the remaining classes. Let us also define the vectors $\mathbf{1}_N \in \mathbb{R}^N$, $\mathbf{1}_{N_p} \in \mathbb{R}^{N_p}$, and $\mathbf{1}_{N_n} \in \mathbb{R}^{N_n}$ which are vectors of ones. By using the class labels c_i, we can also define the positive class binary vector $\mathbf{e}_p \in \mathbb{R}^N$ having elements $e_{p,i} = 1$ if $c_i = p$ and $e_{p,i} = 0$ if $c_i \neq p$. The negative class binary vector $\mathbf{e}_n \in \mathbb{R}^N$ is defined accordingly, i.e., $e_{n,i} = 0$ if $c_i = p$ and $e_{n,i} = 1$ if $c_i \neq p$. Finally, we denote by $\mathbf{J}_p \in \mathbb{R}^{N\times N}$ and $\mathbf{J}_n \in \mathbb{R}^{N\times N}$ two diagonal matrices having elements $J_{p,ii} = e_{p,i}$, $J_{n,ii} = e_{n,i}$, and $J_{p,ij} = J_{n,ij} = 0$, $i \neq j$.

III. Class-Specific Kernel Discriminant Analysis

CSKDA [19] defines the projection matrix $\mathbf{W}$ as the one maximizing the following criterion:

$$\mathcal{J}(\mathbf{W}) = \frac{D_n}{D_p} \quad (2)$$

where D_n and D_p are the out-of-class and in-class distances defined as follows:

$$D_n = \sum_{i,c_i\neq p} \left\|\mathbf{W}^T\phi(\mathbf{x}_i) - \mathbf{W}^T\mathbf{m}_p\right\|_2^2 \quad (3)$$

and

$$D_p = \sum_{i,c_i=p} \left\|\mathbf{W}^T\phi(\mathbf{x}_i) - \mathbf{W}^T\mathbf{m}_p\right\|_2^2 \quad (4)$$

$\mathbf{m}_p \in \mathcal{F}$ is the mean vector of class p in $\mathcal{F}$, i.e., $\mathbf{m}_p = (1/N_p)\sum_{i,c_i=p}\phi(\mathbf{x}_i)$. That is, it is assumed that the positive class is unimodal (in $\mathcal{F}$) and $\mathbf{W}$ is defined as the matrix that maps the positive class vectors as close as possible to the class mean vector, while it maps the negative class vectors as far as possible from it, in the reduced dimensionality space $\mathbb{R}^d$.

In order to calculate $\mathbf{W}$, the criterion $\mathcal{J}(\mathbf{W})$ in (2) can be expressed as

$$\mathcal{J}(\mathbf{W}) = \frac{Tr\left(\mathbf{W}^T\mathbf{S}_n\mathbf{W}\right)}{Tr\left(\mathbf{W}^T\mathbf{S}_p\mathbf{W}\right)} \quad (5)$$

where $Tr(\cdot)$ is the trace operator. $\mathbf{S}_n \in \mathbb{R}^{|\mathcal{F}|\times|\mathcal{F}|}$ and $\mathbf{S}_p \in \mathbb{R}^{|\mathcal{F}|\times|\mathcal{F}|}$ are the out-of-class and in-class scatter matrices in $\mathcal{F}$

$$\mathbf{S}_n = \sum_{i,c_i\neq p}\left(\phi(\mathbf{x}_i) - \mathbf{m}_p\right)\left(\phi(\mathbf{x}_i) - \mathbf{m}_p\right)^T \quad (6)$$

and

$$\mathbf{S}_p = \sum_{i,c_i=p}\left(\phi(\mathbf{x}_i) - \mathbf{m}_p\right)\left(\phi(\mathbf{x}_i) - \mathbf{m}_p\right)^T. \quad (7)$$

In order to tackle the problem of manipulating the arbitrary-dimensional matrices $\mathbf{S}_n$ and $\mathbf{S}_p$, we exploit the fact that (based on the representer theorem) the data projection matrix $\mathbf{W}$ can be expressed as a linear combination of the training data. Using (1) and (5) can be expressed as

$$\mathcal{J}(\mathbf{A}) = \frac{Tr\left(\mathbf{A}^T\mathbf{M}_n\mathbf{A}\right)}{Tr\left(\mathbf{A}^T\mathbf{M}_p\mathbf{A}\right)} \quad (8)$$

where $\mathbf{M}_n \in \mathbb{R}^{N\times N}$ and $\mathbf{M}_p \in \mathbb{R}^{N\times N}$ are given by

$$\mathbf{M}_n = \mathbf{K}_n\mathbf{K}_n^T - \frac{1}{N_p}\mathbf{K}_n\mathbf{1}_{N_n}\mathbf{1}_{N_p}^T\mathbf{K}_p^T - \frac{1}{N_p}\mathbf{K}_p\mathbf{1}_{N_p}\mathbf{1}_{N_n}^T\mathbf{K}_n^T + \frac{1}{N_p^2}\mathbf{K}_p\mathbf{1}_{N_p}\mathbf{1}_{N_p}^T\mathbf{K}_p^T \quad (9)$$

and

$$\mathbf{M}_p = \mathbf{K}_p\left(\mathbf{I} - \frac{1}{N_p}\mathbf{1}_{N_p}\mathbf{1}_{N_p}^T\right)\mathbf{K}_p^T \quad (10)$$

where $\mathbf{K}_n \in \mathbb{R}^{N\times N_n}$ and $\mathbf{K}_p \in \mathbb{R}^{N\times N_p}$ are matrices formed by the columns of $\mathbf{K}$ corresponding to the negative and positive data, respectively [22]. $\mathbf{A}$ is formed by the eigenvectors corresponding to the d largest eigenvalues of matrix $\mathbf{M} = \mathbf{M}_p^{-1}\mathbf{M}_n$ [44]. Here, we should note that, since the rank of $\mathbf{M}$ is at most $N_p - 1$, the dimensionality of the learned space d is restricted by the number of samples forming the positive class (or by the dimensionality of the input space D), i.e., $d \leq \min(N_p - 1, D)$.

A spectral regression-based solution of CSKDA has been recently proposed in [21] and [22]. Let us denote by $\mathbf{w}$ an eigenvector of the problem $\mathbf{S}_n\mathbf{w} = \lambda\mathbf{S}_p\mathbf{w}$ with eigenvalue λ. $\mathbf{w}$ can be expressed as a linear combination of the training data in $\mathcal{F}$, i.e., $\mathbf{w} = \mathbf{\Phi}\mathbf{a}$. By setting $\mathbf{K}\mathbf{a} = \mathbf{q}$, this eigenanalysis problem can be transformed to the following equivalent problem:

$$\mathbf{P}_n\mathbf{q} = \lambda\mathbf{P}_p\mathbf{q} \quad (11)$$

where $\mathbf{P}_n = \mathbf{e}_n\mathbf{e}_n^T - (1/N_p)\mathbf{e}_n\mathbf{e}_p^T - (1/N_p)\mathbf{e}_p\mathbf{e}_n^T + (1/N_p^2)\mathbf{e}_p\mathbf{e}_p^T$ and $\mathbf{P}_p = (1 - (2/N_p) + (1/N_p^2))\mathbf{e}_p\mathbf{e}_p^T$.

$\mathbf{A}$ can be obtained by applying a two-step process.

1) Solution of the eigenproblem $\mathbf{P}_n\mathbf{q} = \lambda\mathbf{P}_p\mathbf{q}$, leading to the determination of a matrix $\mathbf{Q} = [\mathbf{q}_1, \ldots, \mathbf{q}_d]$, where $\mathbf{q}_i$ is the eigenvector corresponding to the ith largest eigenvalue.
2) Determination of the matrix $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_d]$, where $\mathbf{K}\mathbf{a}_i = \mathbf{q}_i$.

In this section, we provide our analysis and describe the proposed methods. In the following, we will assume that the training data (when represented in $\mathcal{F}$) are centered to $\mathbf{m}_p$. Then, we define the in-class and out-of-class scatter matrices by

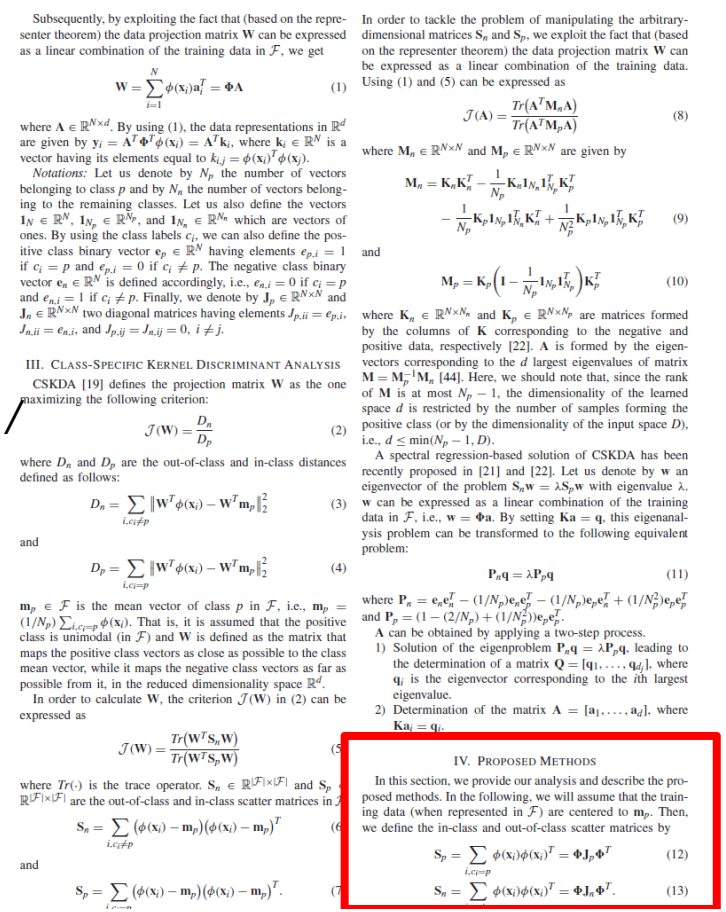
$$\mathbf{S}_p = \sum_{i,c_i=p}\phi(\mathbf{x}_i)\phi(\mathbf{x}_i)^T = \mathbf{\Phi}\mathbf{J}_p\mathbf{\Phi}^T \quad (12)$$

$$\mathbf{S}_n = \sum_{i,c_i\neq p}\phi(\mathbf{x}_i)\phi(\mathbf{x}_i)^T = \mathbf{\Phi}\mathbf{J}_n\mathbf{\Phi}^T. \quad (13)$$

Parts of a paper

Methodology:

- Describes how the work was done. It can contain:
  - methodology description (e.g. description of different blocks of a pipeline)
  - mathematical description of the new model, along with derivations/proofs, etc.
  - Analysis of the complexity/cost of the model/method.



Subsequently, by exploiting the fact that (based on the representer theorem) the data projection matrix $\mathbf{W}$ can be expressed as a linear combination of the training data in $\mathcal{F}$, we get

$$\mathbf{W} = \sum_{i=1}^{N} \phi(\mathbf{x}_i)\mathbf{a}_i^T = \mathbf{\Phi}\mathbf{A} \quad (1)$$

where $\mathbf{A} \in \mathbb{R}^{N\times d}$. By using (1), the data representations in $\mathbb{R}^d$ are given by $\mathbf{y}_i = \mathbf{A}^T\mathbf{\Phi}^T\phi(\mathbf{x}_i) = \mathbf{A}^T\mathbf{k}_i$, where $\mathbf{k}_i \in \mathbb{R}^N$ is a vector having its elements equal to $k_{i,j} = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$.

Notations: Let us denote by N_p the number of vectors belonging to class p and by N_n the number of vectors belonging to the remaining classes. Let us also define the vectors $\mathbf{1}_N \in \mathbb{R}^N$, $\mathbf{1}_{N_p} \in \mathbb{R}^{N_p}$, and $\mathbf{1}_{N_n} \in \mathbb{R}^{N_n}$ which are vectors of ones. By using the class labels c_i, we can also define the positive class binary vector $\mathbf{e}_p \in \mathbb{R}^N$ having elements $e_{p,i} = 1$ if $c_i = p$ and $e_{p,i} = 0$ if $c_i \neq p$. The negative class binary vector $\mathbf{e}_n \in \mathbb{R}^N$ is defined accordingly, i.e., $e_{n,i} = 0$ if $c_i = p$ and $e_{n,i} = 1$ if $c_i \neq p$. Finally, we denote by $\mathbf{J}_p \in \mathbb{R}^{N\times N}$ and $\mathbf{J}_n \in \mathbb{R}^{N\times N}$ two diagonal matrices having elements $J_{p,ii} = e_{p,i}$, $J_{n,ii} = e_{n,i}$, and $J_{p,ij} = J_{n,ij} = 0$, $i \neq j$.

III. Class-Specific Kernel Discriminant Analysis

CSKDA [19] defines the projection matrix $\mathbf{W}$ as the one maximizing the following criterion:

$$\mathcal{J}(\mathbf{W}) = \frac{D_n}{D_p} \quad (2)$$

where D_n and D_p are the out-of-class and in-class distances defined as follows:

$$D_n = \sum_{i,c_i\neq p} \left\|\mathbf{W}^T\phi(\mathbf{x}_i) - \mathbf{W}^T\mathbf{m}_p\right\|_2^2 \quad (3)$$

and

$$D_p = \sum_{i,c_i=p} \left\|\mathbf{W}^T\phi(\mathbf{x}_i) - \mathbf{W}^T\mathbf{m}_p\right\|_2^2 \quad (4)$$

$\mathbf{m}_p \in \mathcal{F}$ is the mean vector of class p in $\mathcal{F}$, i.e., $\mathbf{m}_p = (1/N_p)\sum_{i,c_i=p}\phi(\mathbf{x}_i)$. That is, it is assumed that the positive class is unimodal (in $\mathcal{F}$) and $\mathbf{W}$ is defined as the matrix that maps the positive class vectors as close as possible to the class mean vector, while it maps the negative class vectors as far as possible from it, in the reduced dimensionality space $\mathbb{R}^d$.

In order to calculate $\mathbf{W}$, the criterion $\mathcal{J}(\mathbf{W})$ in (2) can be expressed as

$$\mathcal{J}(\mathbf{W}) = \frac{Tr\left(\mathbf{W}^T\mathbf{S}_n\mathbf{W}\right)}{Tr\left(\mathbf{W}^T\mathbf{S}_p\mathbf{W}\right)} \quad (5$$

where $Tr(\cdot)$ is the trace operator. $\mathbf{S}_n \in \mathbb{R}^{|\mathcal{F}|\times|\mathcal{F}|}$ and $\mathbf{S}_p$ $\mathbb{R}^{|\mathcal{F}|\times|\mathcal{F}|}$ are the out-of-class and in-class scatter matrices in $\mathcal{J}$

$$\mathbf{S}_n = \sum_{i,c_i\neq p}\left(\phi(\mathbf{x}_i) - \mathbf{m}_p\right)\left(\phi(\mathbf{x}_i) - \mathbf{m}_p\right)^T \quad (6$$

and

$$\mathbf{S}_p = \sum_{i,c_i=p}\left(\phi(\mathbf{x}_i) - \mathbf{m}_p\right)\left(\phi(\mathbf{x}_i) - \mathbf{m}_p\right)^T. \quad (7$$

In order to tackle the problem of manipulating the arbitrary-dimensional matrices $\mathbf{S}_n$ and $\mathbf{S}_p$, we exploit the fact that (based on the representer theorem) the data projection matrix $\mathbf{W}$ can be expressed as a linear combination of the training data. Using (1) and (5) can be expressed as

$$\mathcal{J}(\mathbf{A}) = \frac{Tr\left(\mathbf{A}^T\mathbf{M}_n\mathbf{A}\right)}{Tr\left(\mathbf{A}^T\mathbf{M}_p\mathbf{A}\right)} \quad (8)$$

where $\mathbf{M}_n \in \mathbb{R}^{N\times N}$ and $\mathbf{M}_p \in \mathbb{R}^{N\times N}$ are given by

$$\mathbf{M}_n = \mathbf{K}_n\mathbf{K}_n^T - \frac{1}{N_p}\mathbf{K}_n\mathbf{1}_{N_n}\mathbf{1}_{N_p}^T\mathbf{K}_p^T - \frac{1}{N_p}\mathbf{K}_p\mathbf{1}_{N_p}\mathbf{1}_{N_n}^T\mathbf{K}_n^T + \frac{1}{N_p^2}\mathbf{K}_p\mathbf{1}_{N_p}\mathbf{1}_{N_p}^T\mathbf{K}_p^T \quad (9)$$

and

$$\mathbf{M}_p = \mathbf{K}_p\left(\mathbf{I} - \frac{1}{N_p}\mathbf{1}_{N_p}\mathbf{1}_{N_p}^T\right)\mathbf{K}_p^T \quad (10)$$

where $\mathbf{K}_n \in \mathbb{R}^{N\times N_n}$ and $\mathbf{K}_p \in \mathbb{R}^{N\times N_p}$ are matrices formed by the columns of $\mathbf{K}$ corresponding to the negative and positive data, respectively [22]. $\mathbf{A}$ is formed by the eigenvectors corresponding to the d largest eigenvalues of matrix $\mathbf{M} = \mathbf{M}_p^{-1}\mathbf{M}_n$ [44]. Here, we should note that, since the rank of $\mathbf{M}$ is at most $N_p - 1$, the dimensionality of the learned space d is restricted by the number of samples forming the positive class (or by the dimensionality of the input space D), i.e., $d \leq \min(N_p - 1, D)$.

A spectral regression-based solution of CSKDA has been recently proposed in [21] and [22]. Let us denote by $\mathbf{w}$ an eigenvector of the problem $\mathbf{S}_n\mathbf{w} = \lambda\mathbf{S}_p\mathbf{w}$ with eigenvalue λ. $\mathbf{w}$ can be expressed as a linear combination of the training data in $\mathcal{F}$, i.e., $\mathbf{w} = \mathbf{\Phi}\mathbf{a}$. By setting $\mathbf{K}\mathbf{a} = \mathbf{q}$, this eigenanalysis problem can be transformed to the following equivalent problem:

$$\mathbf{P}_n\mathbf{q} = \lambda\mathbf{P}_p\mathbf{q} \quad (11)$$

where $\mathbf{P}_n = \mathbf{e}_n\mathbf{e}_n^T - (1/N_p)\mathbf{e}_n\mathbf{e}_p^T - (1/N_p)\mathbf{e}_p\mathbf{e}_n^T + (1/N_p^2)\mathbf{e}_p\mathbf{e}_p^T$ and $\mathbf{P}_p = (1 - (2/N_p) + (1/N_p^2))\mathbf{e}_p\mathbf{e}_p^T$.

$\mathbf{A}$ can be obtained by applying a two-step process.

1) Solution of the eigenproblem $\mathbf{P}_n\mathbf{q} = \lambda\mathbf{P}_p\mathbf{q}$, leading to the determination of a matrix $\mathbf{Q} = [\mathbf{q}_1, \ldots, \mathbf{q}_d]$, where $\mathbf{q}_i$ is the eigenvector corresponding to the ith largest eigenvalue.
2) Determination of the matrix $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_d]$, where $\mathbf{K}\mathbf{a}_i = \mathbf{q}_i$.

IV. Proposed Methods

In this section, we provide our analysis and describe the proposed methods. In the following, we will assume that the training data (when represented in $\mathcal{F}$) are centered to $\mathbf{m}_p$. Then, we define the in-class and out-of-class scatter matrices by

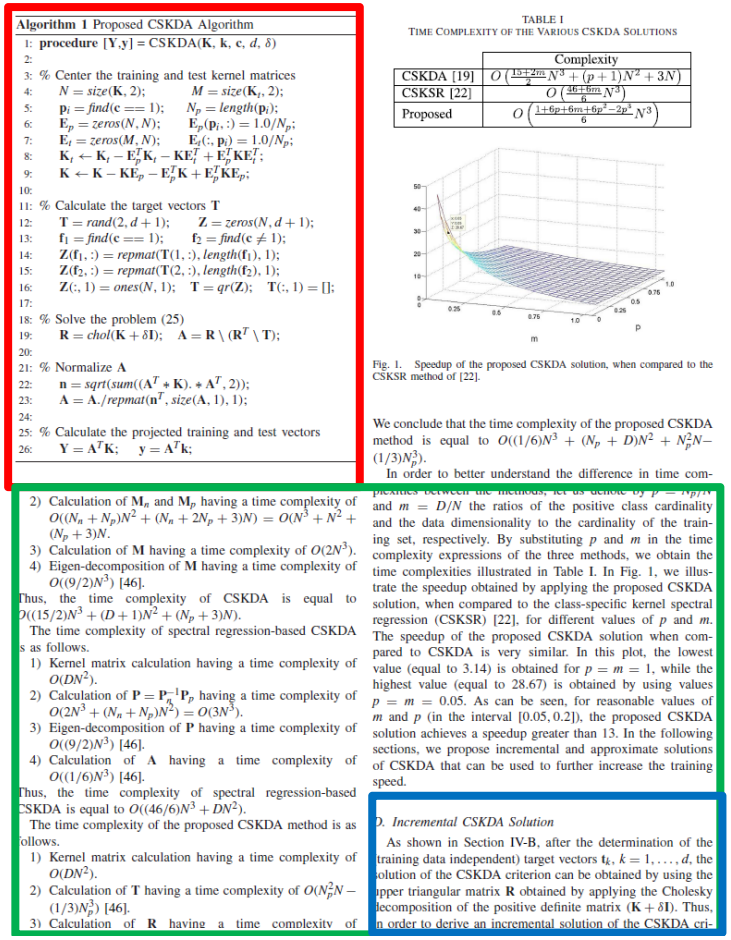
$$\mathbf{S}_p = \sum_{i,c_i=p}\phi(\mathbf{x}_i)\phi(\mathbf{x}_i)^T = \mathbf{\Phi}\mathbf{J}_p\mathbf{\Phi}^T \quad (12)$$

$$\mathbf{S}_n = \sum\phi(\mathbf{x}_i)\phi(\mathbf{x}_i)^T = \mathbf{\Phi}\mathbf{J}_n\mathbf{\Phi}^T. \quad (13)$$

Parts of a paper

Methodology:

- Describes how the work was done. It can contain:
  - methodology description (e.g. description of different blocks of a pipeline)
  - mathematical description of the new model along with derivations/proofs, etc.
  - algorithmic description/pseudocode, etc.
  - discussion on the properties of the model/ method, and possibly extensions/simplifications



Algorithm 1 Proposed CSKDA Algorithm

procedure [Y,y] = CSKDA($\mathbf{K}$, $\mathbf{k}$, $\mathbf{c}$, d, δ)

% Center the training and test kernel matrices
$N = size(\mathbf{K}, 2)$; $M = size(\mathbf{K}_t, 2)$;
$\mathbf{p}_l = find(\mathbf{c} == 1)$; $N_p = length(\mathbf{p}_l)$;
$\mathbf{E}_p = zeros(N, N)$; $\mathbf{E}_p(\mathbf{p}_l, :) = 1.0/N_p$;
$\mathbf{E}_t = zeros(M, N)$; $\mathbf{E}_t(:, \mathbf{p}_l) = 1.0/N_p$;
$\mathbf{K}_t \leftarrow \mathbf{K}_t - \mathbf{E}_t^T\mathbf{K}_t - \mathbf{K}\mathbf{E}_t^T + \mathbf{E}_t^T\mathbf{K}\mathbf{E}_t^T$;
$\mathbf{K} \leftarrow \mathbf{K} - \mathbf{K}\mathbf{E}_p - \mathbf{E}_p^T\mathbf{K} + \mathbf{E}_p^T\mathbf{K}\mathbf{E}_p$;

% Calculate the target vectors $\mathbf{T}$
$\mathbf{T} = rand(2, d+1)$; $\mathbf{Z} = zeros(N, d+1)$;
$\mathbf{f}_1 = find(\mathbf{c} == 1)$; $\mathbf{f}_2 = find(\mathbf{c} \neq 1)$;
$\mathbf{Z}(\mathbf{f}_1, :) = repmat(\mathbf{T}(1, :), length(\mathbf{f}_1), 1)$;
$\mathbf{Z}(\mathbf{f}_2, :) = repmat(\mathbf{T}(2, :), length(\mathbf{f}_2), 1)$;
$\mathbf{Z}(:, 1) = ones(N, 1)$; $\mathbf{T} = qr(\mathbf{Z})$; $\mathbf{T}(:, 1) = []$;

% Solve the problem (25)
$\mathbf{R} = chol(\mathbf{K} + \delta\mathbf{I})$; $\mathbf{A} = \mathbf{R} \setminus (\mathbf{R}^T \setminus \mathbf{T})$;

% Normalize $\mathbf{A}$
$\mathbf{n} = sqrt(sum((\mathbf{A}^T * \mathbf{K}). * \mathbf{A}^T, 2))$;
$\mathbf{A} = \mathbf{A}./repmat(\mathbf{n}^T, size(\mathbf{A}, 1), 1)$;

% Calculate the projected training and test vectors
$\mathbf{Y} = \mathbf{A}^T\mathbf{K}$; $\mathbf{y} = \mathbf{A}^T\mathbf{k}$;

TABLE I
TIME COMPLEXITY OF THE VARIOUS CSKDA SOLUTIONS

	Complexity
CSKDA [19]	$O\left(\frac{15+2m}{2}N^3 + (p+1)N^2 + 3N\right)$
CSKSR [22]	$O\left(\frac{36+6m}{6}N^3\right)$
Proposed	$O\left(\frac{1+6p+6m+6p^2-2p^3}{6}N^3\right)$

Fig. 1. Speedup of the proposed CSKDA solution, when compared to the CSKSR method of [22].

2) Calculation of $\mathbf{M}_n$ and $\mathbf{M}_p$ having a time complexity of $O((N_n + N_p)N^2 + (N_n + 2N_p + 3)N) = O(N^3 + N^2 + (N_p + 3)N$.
3) Calculation of $\mathbf{M}$ having a time complexity of $O(2N^3)$.
4) Eigen-decomposition of $\mathbf{M}$ having a time complexity of $O((9/2)N^3)$ [46].

Thus, the time complexity of CSKDA is equal to $O((15/2)N^3 + (D+1)N^2 + (N_p+3)N)$.

The time complexity of spectral regression-based CSKDA s as follows.

1) Kernel matrix calculation having a time complexity of $O(DN^2)$.
2) Calculation of $\mathbf{P} = \mathbf{P}_{\mathbf{l}}^{-1}\mathbf{P}_p$ having a time complexity of $O(2N^3 + (N_n + N_p)N^2) = O(3N^3)$.
3) Eigen-decomposition of $\mathbf{P}$ having a time complexity of $O((9/2)N^3)$ [46].
4) Calculation of $\mathbf{A}$ having a time complexity of $O((1/6)N^3)$ [46].

Thus, the time complexity of spectral regression-based CSKDA is equal to $O((46/6)N^3 + DN^2)$.

The time complexity of the proposed CSKDA method is as ollows.

1) Kernel matrix calculation having a time complexity of $O(DN^2)$.
2) Calculation of $\mathbf{T}$ having a time complexity of $O(N_p^2N - (1/3)N_p^3)$ [46].
3) Calculation of $\mathbf{R}$ having a time complexity of

We conclude that the time complexity of the proposed CSKDA method is equal to $O((1/6)N^3 + (N_p + D)N^2 + N_p^2N - (1/3)N_p^3)$.

In order to better understand the difference in time complexities between the methods, let us denote by $p = N_p/N$ and $m = D/N$ the ratios of the positive class cardinality and the data dimensionality to the cardinality of the training set, respectively. By substituting p and m in the time complexity expressions of the three methods, we obtain the time complexities illustrated in Table I. In Fig. 1, we illustrate the speedup obtained by applying the proposed CSKDA solution, when compared to the class-specific kernel spectral regression (CSKSR) [22], for different values of p and m. The speedup of the proposed CSKDA solution when compared to CSKDA is very similar. In this plot, the lowest value (equal to 3.14) is obtained for $p = m = 1$, while the highest value (equal to 28.67) is obtained by using values $p = m = 0.05$. As can be seen, for reasonable values of m and p (in the interval $[0.05, 0.2]$), the proposed CSKDA solution achieves a speedup greater than 13. In the following sections, we propose incremental and approximate solutions of CSKDA that can be used to further increase the training speed.

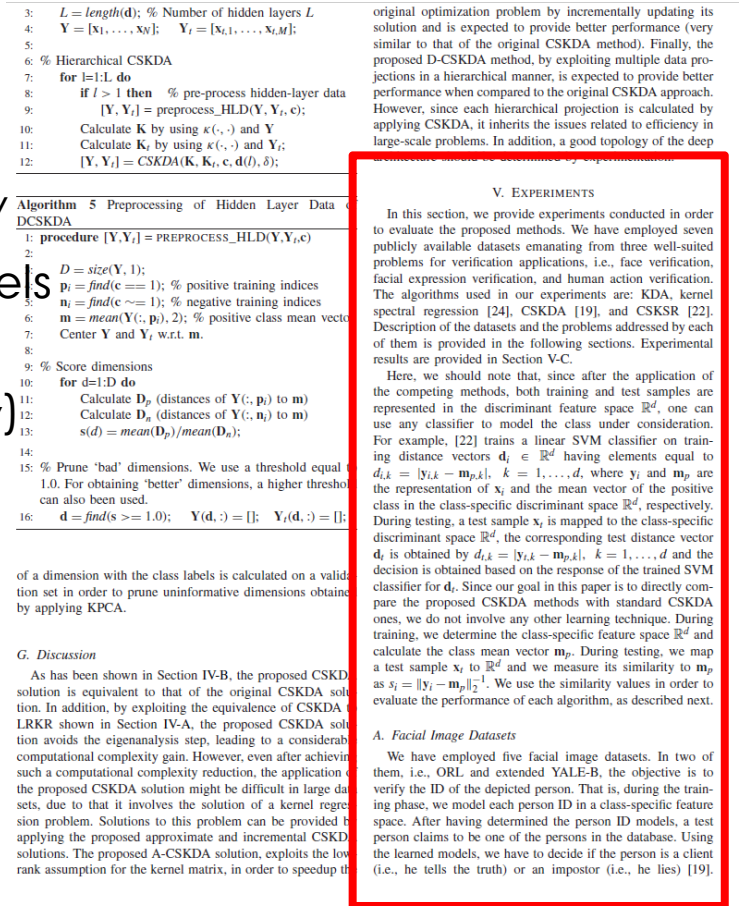
D. Incremental CSKDA Solution

As shown in Section IV-B, after the determination of the training data independent) target vectors $\mathbf{t}_k$, $k = 1, \ldots, d$, the solution of the CSKDA criterion can be obtained by using the upper triangular matrix $\mathbf{R}$ obtained by applying the Cholesky decomposition of the positive definite matrix $(\mathbf{K} + \delta\mathbf{I})$. Thus, n order to derive an incremental solution of the CSKDA cri-

Parts of a paper

Experiments:

- It provides the description of experiments conducted in order to test the performance of the current model/method
- It provides comparisons of the current model/method with existing (related) methods/models
- Comparisons need to be both in terms of performance and efficiency (when necessary)



$L = length(\mathbf{d})$; % Number of hidden layers L
$\mathbf{Y} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]$; $\mathbf{Y}_l = [\mathbf{x}_{t,1}, \ldots, \mathbf{x}_{t,M}]$;

% Hierarchical CSKDA
for l=1:L **do**
if $l > 1$ **then** % pre-process hidden-layer data
$[\mathbf{Y}, \mathbf{Y}_t]$ = preprocess_HLD($\mathbf{Y}, \mathbf{Y}_t, \mathbf{c}$);
Calculate $\mathbf{K}$ by using $\kappa(\cdot,\cdot)$ and $\mathbf{Y}$
Calculate $\mathbf{K}_t$ by using $\kappa(\cdot,\cdot)$ and $\mathbf{Y}_t$;
$[\mathbf{Y}, \mathbf{Y}_t] = CSKDA(\mathbf{K}, \mathbf{K}_t, \mathbf{c}, \mathbf{d}(l), \delta)$;

Algorithm 5 Preprocessing of Hidden Layer Data
DCSKDA
procedure $[\mathbf{Y},\mathbf{Y}_t]$ = PREPROCESS_HLD($\mathbf{Y},\mathbf{Y}_t,\mathbf{c}$)

$D = size(\mathbf{Y}, 1)$;
$\mathbf{p}_i = find(\mathbf{c} == 1)$; % positive training indices
$\mathbf{n}_i = find(\mathbf{c} \sim= 1)$; % negative training indices
$\mathbf{m} = mean(\mathbf{Y}(:, \mathbf{p}_i), 2)$; % positive class mean vecto
Center $\mathbf{Y}$ and $\mathbf{Y}_t$ w.r.t. $\mathbf{m}$.

% Score dimensions
for d=1:D **do**
Calculate $\mathbf{D}_p$ (distances of $\mathbf{Y}(:, \mathbf{p}_i)$ to $\mathbf{m}$)
Calculate $\mathbf{D}_n$ (distances of $\mathbf{Y}(:, \mathbf{n}_i)$ to $\mathbf{m}$)
$s(d) = mean(\mathbf{D}_p)/mean(\mathbf{D}_n)$;

% Prune 'bad' dimensions. We use a threshold equal
1.0. For obtaining 'better' dimensions, a higher thresho
can also been used.
$\mathbf{d} = find(\mathbf{s} >= 1.0)$; $\mathbf{Y}(\mathbf{d}, :) = []$; $\mathbf{Y}_t(\mathbf{d}, :) = []$;

of a dimension with the class labels is calculated on a valid
tion set in order to prune uninformative dimensions obtain
by applying KPCA.

G. Discussion

As has been shown in Section IV-B, the proposed CSKD
solution is equivalent to that of the original CSKDA solu
tion. In addition, by exploiting the equivalence of CSKDA
LRKR shown in Section IV-A, the proposed CSKDA solu
tion avoids the eigenanalysis step, leading to a considerab
computational complexity gain. However, even after achievi
such a computational complexity reduction, the application
the proposed CSKDA solution might be difficult in large da
sets, due to that it involves the solution of a kernel regre
sion problem. Solutions to this problem can be provided b
applying the proposed approximate and incremental CSKD
solutions. The proposed A-CSKDA solution, exploits the lo
rank assumption for the kernel matrix, in order to speedup t

original optimization problem by incrementally updating its solution and is expected to provide better performance (very similar to that of the original CSKDA method). Finally, the proposed D-CSKDA method, by exploiting multiple data projections in a hierarchical manner, is expected to provide better performance when compared to the original CSKDA approach. However, since each hierarchical projection is calculated by applying CSKDA, it inherits the issues related to efficiency in large-scale problems. In addition, a good topology of the deep

V. Experiments

In this section, we provide experiments conducted in order to evaluate the proposed methods. We have employed seven publicly available datasets emanating from three well-suited problems for verification applications, i.e., face verification, facial expression verification, and human action verification. The algorithms used in our experiments are: KDA, kernel spectral regression [24], CSKDA [19], and CSKSR [22]. Description of the datasets and the problems addressed by each of them is provided in the following sections. Experimental results are provided in Section V-C.

Here, we should note that, since after the application of the competing methods, both training and test samples are represented in the discriminant feature space $\mathbb{R}^d$, one can use any classifier to model the class under consideration. For example, [22] trains a linear SVM classifier on training distance vectors $\mathbf{d}_i \in \mathbb{R}^d$ having elements equal to $d_{i,k} = |\mathbf{y}_{i,k} - \mathbf{m}_{p,k}|$, $k = 1, \ldots, d$, where $\mathbf{y}_i$ and $\mathbf{m}_p$ are the representation of $\mathbf{x}_i$ and the mean vector of the positive class in the class-specific discriminant space $\mathbb{R}^d$, respectively. During testing, a test sample $\mathbf{x}_t$ is mapped to the class-specific discriminant space $\mathbb{R}^d$, the corresponding test distance vector $\mathbf{d}_t$ is obtained by $d_{t,k} = |\mathbf{y}_{t,k} - \mathbf{m}_{p,k}|$, $k = 1, \ldots, d$ and the decision is obtained based on the response of the trained SVM classifier for $\mathbf{d}_t$. Since our goal in this paper is to directly compare the proposed CSKDA methods with standard CSKDA ones, we do not involve any other learning technique. During training, we determine the class-specific feature space $\mathbb{R}^d$ and calculate the class mean vector $\mathbf{m}_p$. During testing, we map a test sample $\mathbf{x}_t$ to $\mathbb{R}^d$ and we measure its similarity to $\mathbf{m}_p$ as $s_i = \|\mathbf{y}_i - \mathbf{m}_p\|_2^{-1}$. We use the similarity values in order to evaluate the performance of each algorithm, as described next.

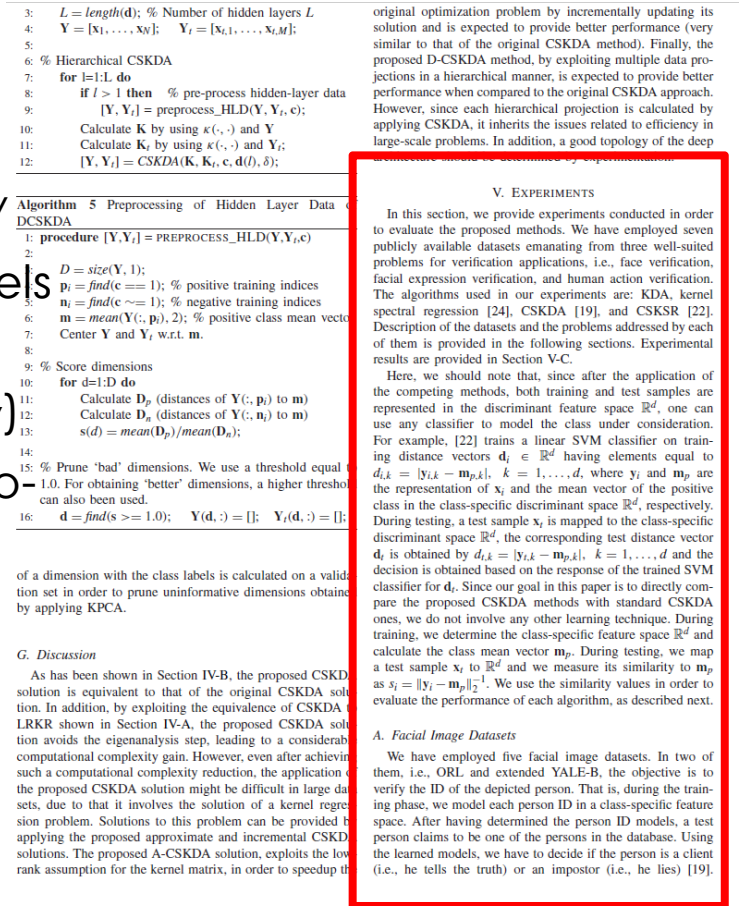
A. Facial Image Datasets

We have employed five facial image datasets. In two of them, i.e., ORL and extended YALE-B, the objective is to verify the ID of the depicted person. That is, during the training phase, we model each person ID in a class-specific feature space. After having determined the person ID models, a test person claims to be one of the persons in the database. Using the learned models, we have to decide if the person is a client (i.e., he tells the truth) or an impostor (i.e., he lies) [19].

Parts of a paper

Experiments:

- It provides the description of experiments conducted in order to test the performance of the current model/method
- It provides comparisons of the current model/method with existing (related) methods/models
- Comparisons need to be both in terms of performance and efficiency (when necessary)
- It should clearly define the experimental protocol, the hyper-parameter selection process, and the competing methods/models



$L = length(\mathbf{d})$; % Number of hidden layers L
$\mathbf{Y} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]$; $\mathbf{Y}_l = [\mathbf{x}_{l,1}, \ldots, \mathbf{x}_{l,M}]$;

% Hierarchical CSKDA
for l=1:L **do**
if $l > 1$ **then** % pre-process hidden-layer data
$[\mathbf{Y}, \mathbf{Y}_l]$ = preprocess_HLD($\mathbf{Y}, \mathbf{Y}_l, \mathbf{c}$);
Calculate $\mathbf{K}$ by using $\kappa(\cdot,\cdot)$ and $\mathbf{Y}$
Calculate $\mathbf{K}_l$ by using $\kappa(\cdot,\cdot)$ and $\mathbf{Y}_l$;
$[\mathbf{Y}, \mathbf{Y}_l] = CSKDA(\mathbf{K}, \mathbf{K}_l, \mathbf{c}, \mathbf{d}(l), \delta)$;

Algorithm 5 Preprocessing of Hidden Layer Data
DCSKDA
procedure $[\mathbf{Y},\mathbf{Y}_l]$ = PREPROCESS_HLD($\mathbf{Y},\mathbf{Y}_l,\mathbf{c}$)

$D = size(\mathbf{Y}, 1)$;
$\mathbf{p}_i = find(\mathbf{c} == 1)$; % positive training indices
$\mathbf{n}_i = find(\mathbf{c} \sim= 1)$; % negative training indices
$\mathbf{m} = mean(\mathbf{Y}(:, \mathbf{p}_i), 2)$; % positive class mean vecto
Center $\mathbf{Y}$ and $\mathbf{Y}_l$ w.r.t. $\mathbf{m}$.

% Score dimensions
for d=1:D **do**
Calculate $\mathbf{D}_p$ (distances of $\mathbf{Y}(:, \mathbf{p}_i)$ to $\mathbf{m}$)
Calculate $\mathbf{D}_n$ (distances of $\mathbf{Y}(:, \mathbf{n}_i)$ to $\mathbf{m}$)
$s(d) = mean(\mathbf{D}_p)/mean(\mathbf{D}_n)$;

% Prune 'bad' dimensions. We use a threshold equal
1.0. For obtaining 'better' dimensions, a higher thresho
can also been used.
$\mathbf{d} = find(\mathbf{s} >= 1.0)$; $\mathbf{Y}(\mathbf{d}, :) = []$; $\mathbf{Y}_l(\mathbf{d}, :) = []$;

of a dimension with the class labels is calculated on a valid
tion set in order to prune uninformative dimensions obtain
by applying KPCA.

G. Discussion

As has been shown in Section IV-B, the proposed CSKD
solution is equivalent to that of the original CSKDA solu
tion. In addition, by exploiting the equivalence of CSKDA
LRKR shown in Section IV-A, the proposed CSKDA solu
tion avoids the eigenanalysis step, leading to a considerab
computational complexity gain. However, even after achievi
such a computational complexity reduction, the application
the proposed CSKDA solution might be difficult in large da
sets, due to that it involves the solution of a kernel regre
sion problem. Solutions to this problem can be provided b
applying the proposed approximate and incremental CSKD
solutions. The proposed A-CSKDA solution, exploits the lo
rank assumption for the kernel matrix, in order to speedup th

original optimization problem by incrementally updating its solution and is expected to provide better performance (very similar to that of the original CSKDA method). Finally, the proposed D-CSKDA method, by exploiting multiple data projections in a hierarchical manner, is expected to provide better performance when compared to the original CSKDA approach. However, since each hierarchical projection is calculated by applying CSKDA, it inherits the issues related to efficiency in large-scale problems. In addition, a good topology of the deep

V. Experiments

In this section, we provide experiments conducted in order to evaluate the proposed methods. We have employed seven publicly available datasets emanating from three well-suited problems for verification applications, i.e., face verification, facial expression verification, and human action verification. The algorithms used in our experiments are: KDA, kernel spectral regression [24], CSKDA [19], and CSKSR [22]. Description of the datasets and the problems addressed by each of them is provided in the following sections. Experimental results are provided in Section V-C.

Here, we should note that, since after the application of the competing methods, both training and test samples are represented in the discriminant feature space $\mathbb{R}^d$, one can use any classifier to model the class under consideration. For example, [22] trains a linear SVM classifier on training distance vectors $\mathbf{d}_i \in \mathbb{R}^d$ having elements equal to $d_{i,k} = |\mathbf{y}_{i,k} - \mathbf{m}_{p,k}|$, $k = 1, \ldots, d$, where $\mathbf{y}_i$ and $\mathbf{m}_p$ are the representation of $\mathbf{x}_i$ and the mean vector of the positive class in the class-specific discriminant space $\mathbb{R}^d$, respectively. During testing, a test sample $\mathbf{x}_t$ is mapped to the class-specific discriminant space $\mathbb{R}^d$, the corresponding test distance vector $\mathbf{d}_t$ is obtained by $d_{t,k} = |\mathbf{y}_{t,k} - \mathbf{m}_{p,k}|$, $k = 1, \ldots, d$ and the decision is obtained based on the response of the trained SVM classifier for $\mathbf{d}_t$. Since our goal in this paper is to directly compare the proposed CSKDA methods with standard CSKDA ones, we do not involve any other learning technique. During training, we determine the class-specific feature space $\mathbb{R}^d$ and calculate the class mean vector $\mathbf{m}_p$. During testing, we map a test sample $\mathbf{x}_t$ to $\mathbb{R}^d$ and we measure its similarity to $\mathbf{m}_p$ as $s_i = \|\mathbf{y}_i - \mathbf{m}_p\|_2^{-1}$. We use the similarity values in order to evaluate the performance of each algorithm, as described next.

A. Facial Image Datasets

We have employed five facial image datasets. In two of them, i.e., ORL and extended YALE-B, the objective is to verify the ID of the depicted person. That is, during the training phase, we model each person ID in a class-specific feature space. After having determined the person ID models, a test person claims to be one of the persons in the database. Using the learned models, we have to decide if the person is a client (i.e., he tells the truth) or an impostor (i.e., he lies) [19].

Parts of a paper

Experiments:

- It provides the description of experiments conducted in order to test the performance of the current model/method
- It provides comparisons of the current model/method with existing (related) methods/models
- Comparisons need to be both in terms of performance and efficiency (when necessary)
- It should clearly define the experimental protocol, the hyper-parameter selection process, and the competing methods/models
- Data description (and availability to the reader)
- Any pre-processing steps applied need to be adequately described

Fig. 2. Facial images depicting a person from the ORL dataset.

Fig. 3. Facial images depicting a person from the extended YALE-B dataset.

Fig. 4. Facial images depicting a person from the BU dataset.

Fig. 5. Facial images from the COHN-KANADE dataset.

Fig. 6. Facial images depicting a person from the JAFFE dataset.

Fig. 7. Video frames of the Hollywood2 dataset depicting instances of all the 12 actions.

s remaining three datasets, i.e., BU, COHN-CANADE, and Jaffe, the objective is to verify the facial expression of the depicted person, where the possible choices are the seven basic facial epxression, i.e., neutral, anger, disgust, fear, happy, sad, and surprise. Since in the face verification problem the performance of a method is usually measured using the equal error rate (ERR) metric [19], we also use it in our experiments. Since there is not a widely adopted experimental protocol for these datasets, in our experiments we randomly keep half of the images depicting each class for training and we use the remaining images for evaluation. We perform ive experiments and report the mean performance of ea algorithm.

1) ORL Dataset [50]: It consists of 400 facial imag depicting 40 persons (ten images each). The images were ca tured at different times and with different conditions, in ter of lighting, facial expressions (smiling/not smiling), and fac details (open/closed eyes and with/without glasses). Fac images were taken in frontal position with a tolerance for fa rotation and tilting up to 20°. Example images of the data are illustrated in Fig. 2.

2) Extended YALE-B Dataset [51]: It consists of fac images depicting 38 persons in nine poses, under 64 illumin tion conditions. In our experiments, we have used the fron cropped images provided by the database. Example images f the dataset are illustrated in Fig. 3.

3) BU Dataset [52]: It consists of facial images depicting over 100 persons (60% female and 40% male) with a variety of ethnic/racial background, including white, black, East-Asian, Middle-East Asian, Hispanic Latino, etc. All expressions, except the neutral one, are expressed at four intensity levels. In our experiments, we have employed the images depicting the most expressive intensity of each facial expression. Example images from the dataset are illustrated in Fig. 4.

4) COHN-KANADE Dataset [53]: It consists of facial images depicting 210 persons of age between 18 and 50 (69% female, 31% male, 81% Euro-American, 13% Afro-American, and 6% other groups). We have randomly selected 35 images for each facial expression, i.e., anger, disgust, fear, happiness,

5) JAFFE Dataset [54]: It consists of 210 facial imag depicting ten Japanese female persons. Each person is depic in three images for each expression. Example images of t dataset are illustrated in Fig. 6.

B. Action Video Datasets

We have employed two publicly available action datase i.e., Hollywood2 and Olympic Sports, which have be designed for the problem of action recognition in the wi Since in such a problem multiple actions might be depict in a video, all methods targeting the problem are, in fa trained for a class-specific scenario distinguishing each act class from the remaining ones. Evaluation is performed using the mean average precision (mAP) metric, which well-suited for (class-specific-based) video retrieval proble In our experiments, we have employed the standard tra ing/test partitioning of the videos, which is provided by databases.

depicting 12 actions. The videos have been collected from 69 different Hollywood movies. Example video frames of the dataset are illustrated in Fig. 7. The standard training-test split provided by the database consists of 823 training videos and 884 test videos. Training and test videos come from different movies.

2) Olympic Sports Dataset [56]: It consists of 783 videos depicting athletes practicing 16 sports. Example video frames of the dataset are illustrated in Fig. 8. The standard training-test split provided by the database consists of 649 training videos 134 test videos.

C. Results

Parts of a paper

Experiments:

- RESULTS!
- Comparisons (quantitative and qualitative)
- Observations and possible reasons

Fig. 8. Video frames of the Olympic sports dataset depicting instances of all the 16 actions.

function $\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) = \exp(-(\|\mathbf{x}_i - \mathbf{x}_j\|_2^2/2\sigma^2))$, where the value of σ is set equal to the mean Euclidean distance between the (vectorized) training facial images $\mathbf{x}_i$. For I-CSKDA, we have used an initial training set consisting of $0.25N$ and updated the model using chunks of size $0.1N$. For the proposed D-CSKDA method we have tested three architectures. The first consists of three layers with dimensions $\{D, N_p - 1, d\}$ and the remaining ones consist of four layers having dimensions $\{D, N_p - 1, 0.6N_p, d\}$ and $\{D, N_p - 1, 0.5N_p, d\}$, respectively. For the calculation of the kernel matrix in the first layer we employed the RBF kernel, while for the remaining layers of the deep architecture we employed the linear kernel, i.e., $\mathbf{K} = \mathbf{Y}^T\mathbf{Y}$. The exploitation of RBF kernel, gave similar results to the linear one.

The performance of the competing methods on the facial image datasets, along with the time required to learn the models for all the classes, are illustrated in Tables II–VI. From these tables, it can be observed that the performance of the proposed CSKDA method is (in most cases) identical to the performance provided by standard CSKDA, while the proposed method operates faster. Specifically, on the ORL dataset, the proposed CSKDA solution achieves performance equal to 97.71% mAP, which is the same with that of the original CSKDA and CSKDA solutions, while it achieves a slightly better performance when considering ERR metric (0.2% compared to the 0.21% for the original solutions). On the Yale dataset, the performance of the proposed CSKDA solution is equal to 97.32% mAP and 1.42% ERR, which is slightly better when compared to the performance of the original solutions. On the BU dataset, the proposed CSKDA solution achieves the same performance with CSKDA (equal to 58.46% mAP and 21.67% ERR), while CSKSR achieves performance equal to 58.17% mAP and 21.38% ERR. On the COHN-KANADE dataset the proposed CSKDA solution achieves the same performance with CSKSR (equal to 70.12% mAP and 17.93% ERR), outperforming CSKDA achieving performance equal to 69.22% mAP and 21.11% ERR. Finally, on the JAFFE dataset the proposed CSKDA solution achieves

TABLE II
PERFORMANCE AND TRAINING TIMES ON THE ORL DATASET

	EER ↓	mAP ↑	Time (sec)	Dimensions
KDA	0.21%	97.64%	1.180	$D \to 1$
KSR	0.22%	**97.71%**	0.696	$D \to 1$
CSKDA	0.21%	**97.71%**	1.588	$D \to 1$
CSKSR	0.21%	**97.71%**	0.992	$D \to 1$
Prop. CSKDA	**0.2%**	**97.71%**	0.692	$D \to 1$
Prop. I-CSKDA	**0.2%**	**97.71%**	0.624	$D \to 1$
Prop. A-CSKDA	2.2%	86.45%	0.610	$D \to 1$
Prop. D-CSKDA	**0.21%**	**97.85%**	1.380	$D \to N_p\text{-}1 \to 2$

TABLE III
PERFORMANCE AND TRAINING TIMES ON THE EXTENDED YALE-B DATASET

	EER ↓	mAP ↑	Time (sec)	Dimensions
KDA	12.65%	96.75%	56.85	$D \to 1$
KSR	6.09%	97.31%	11.91	$D \to 1$
CSKDA	1.93%	97.31%	57.83	$D \to 2$
CSKSR	1.83%	97.31%	37.83	$D \to 2$
Prop. CSKDA	**1.42%**	**97.32%**	12.05	$D \to 3$
Prop. I-CSKDA	**1.42%**	**97.32%**	11.26	$D \to 3$
Prop. A-CSKDA	15.28%	83.44%	10.95	$D \to 3$
Prop. D-CSKDA	**1.07%**	**97.54%**	65.24	$D \to N_p\text{-}1 \to 6$

TABLE IV
PERFORMANCE AND TRAINING TIMES ON THE BU DATASET

	EER ↓	mAP ↑	Time (sec)	Dimensions
KDA	23.6%	54.14%	0.518	$D \to 1$
KSR	22.05%	58.17%	0.274	$D \to 1$
CSKDA	21.67%	58.17%	0.835	$D \to 3$
CSKSR	**21.38%**	58.17%	0.421	$D \to 2$
Prop. CSKDA	21.67%	**58.46%**	0.273	$D \to 3$
Prop. I-CSKDA	21.67%	**58.46%**	0.261	$D \to 3$
Prop. A-CSKDA	39.67%	32.35%	0.213	$D \to 4$
Prop. D-CSKDA	**21.55%**	**59.23%**	0.632	$D \to N_p\text{-}1 \to 3$

the original solutions. The incremental solution of the proposed CSKDA provides the same performance, but further increases operation speed. On the other hand, the proposed A-CSKDA solving an approximation of the original criterion, provides inferior performance in all cases. However, we should note here that A-CSKDA might be a better choice for large-scale learning, since its overall computational cost is lower. Finally, the exploitation of multiple (hierarchical) layers, each trained by applying CSKDA, enhances performance in all cases.

In the second set of experiments, we applied the competing methods for the human action verification. As a baseline approach we use the methods proposed in [57]: we employ the bag of words (BoWs)-based video representation using histogram of oriented gradient, histogram of optical flow, motion boundary histograms along the direction of x and y, and (normalized) trajectory descriptors evaluated on the trajectories of densely sampled interest points [57]. After calculating the five BoW-based video representations of each video $\mathbf{x}_i^k$, $k = 1, \ldots, 5$, we employ the RBF-χ^2 kernel, where different descriptors are combined in a multichannel approach [58] $\mathbf{K}(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\sum_k (1/4A^k) D(\mathbf{x}_i^k, \mathbf{x}_j^k))$. $D(\mathbf{x}_i^k, \mathbf{x}_j^k)$ is the χ^2 distance between the BoW-based video

Parts of a paper

Experiments:

- RESULTS!
- Comparisons (quantitative and qualitative)
- Observations and possible reasons
- Illustrations of performance (depending on the metrics used, e.g. Precision-Recall, ROC)
- Qualitative illustrations of results

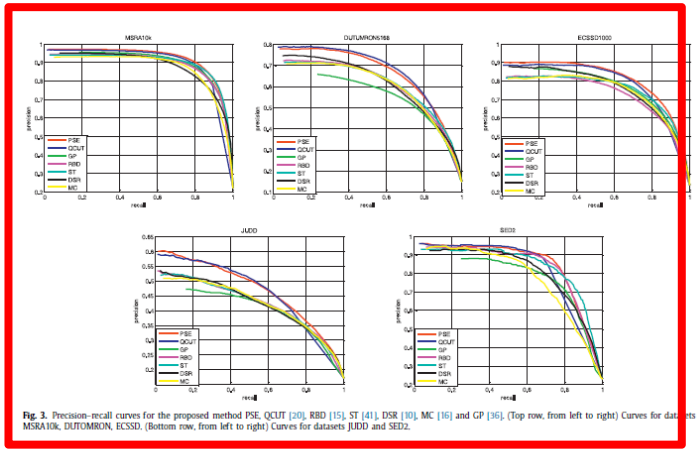


Fig. 3. Precision–recall curves for the proposed method PSE, QCUT [20], RBD [15], ST [41], DSR [10], MC [16] and GP [36]. (Top row, from left to right) Curves for datasets MSRA10k, DUTOMRON, ECSSD. (Bottom row, from left to right) Curves for datasets JUDD and SED2.

Table 1
Maximum F_β measures of the proposed PSE and the unsupervised state-of-the art methods. Top, second and third performing methods are indicated by red, blue and green colors respectively.

	PSE	QCUT	GP	RBD	ST	DSR	MC
MSRA10k	0.8797	0.8720	0.8610	0.8559	0.8669	0.8346	0.8476
DutOmron	0.6705	0.6821	0.5838	0.6305	0.6296	0.6265	0.6274
ECSSD	0.7825	0.7778	0.7457	0.7181	0.7504	0.7371	0.7419
JUDD	0.5034	0.5093	0.4519	0.4590	0.4571	0.4558	0.4623
SED2	0.8444	0.8233	0.7704	0.8302	0.8149	0.7900	0.7713

A good method should perform well with respect to most if not all evaluation metrics. Fig. 4 provides ROC curves for the proposed method against the competing ones. The proposed method PSE is the top performing method in terms of ROC curves, except for SED2 dataset where PSE gives very high recall performance in low false positive rates. Note that one of the leading methods, in precision–recall curves, QCUT does not perform well with respect to ROC curves. The area under the curve (AUC) measures are provided in Table 2.

From Table 2, we observe that PSE is leading in all datasets except for SED2 where it trails the leading method ST by a small margin. GP, DSR and ST also perform well in AUC measures (mostly 2nd and 3rd places); however, they are not as successful in precision–recall curves and F-measure.

Finally, we compare the performance of the compared methods in terms of mean squared errors in Table 3. PSE is the top perform-

Table 2
Performance of the compared methods with respect to AUC measures. Top, second and third performing methods are highlighted in red, blue and green colors, respectively.

	PSE	QCUT	GP	RBD	ST	DSR	MC
MSRA10k	0.9581	0.9240	0.9579	0.9489	0.9557	0.9528	0.9451
DutOmron	0.9010	0.8664	0.8672	0.8895	0.8913	0.8945	0.8827
ECSSD	0.9205	0.8836	0.9109	0.8896	0.9092	0.9089	0.9051
JUDD	0.8344	0.7740	0.7889	0.8177	0.7964	0.8186	0.8160
SED2	0.8925	0.8269	0.8717	0.8695	0.8956	0.8922	0.8549

Table 3
Performance of the compared methods in terms of MSE errors. Top, second and third performing methods are highlighted in red, blue and green colors, respectively.

	PSE	QCUT	GP	RBD	ST	DSR	MC
MSRA10k	0.0588	0.1064	0.0606	0.0647	0.0568	0.0779	0.0690
DutOmron	0.0843	0.0869	0.1197	0.0833	0.0911	0.0848	0.0905
ECSSD	0.0948	0.1488	0.1056	0.1156	0.0984	0.1188	0.1067
JUDD	0.1264	0.1356	0.1579	0.1484	0.1370	0.1347	0.1296
SED2	0.0923	0.1456	0.1146	0.1047	0.0929	0.1144	0.1147

Parts of a paper

Experiments:

- RESULTS!
- Comparisons (quantitative and qualitative)
- Observations and possible reasons
- Illustrations of performance (depending on the metrics used, e.g. Precision-Recall, ROC)
- Qualitative illustrations of results

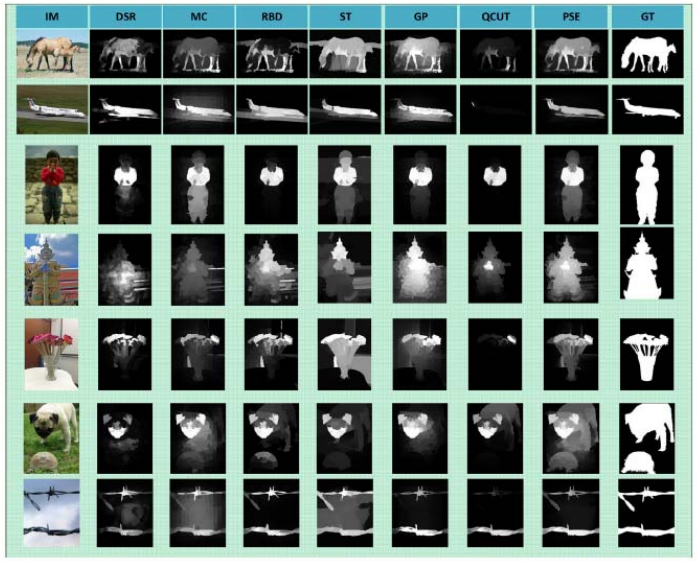


Fig. 2. Qualitative comparison of unsupervised methods. (From left to right: Image (IM), and the saliency maps obtained by DSR [10], MC [16], RBD [15], ST [41], GP [36], QCUT [20] and our PSE, and the ground truth (GT).

3.4. Comparison with the unsupervised state-of-the-art

We compare the proposed PSE method with the top-5 performing unsupervised techniques, according to a recent study [40], namely QCUT [20], RBD [15], ST [41], DSR [10] and MC [16]. DRFI [42] was also ranked as one of the top methods in the study; however, it is a supervised method and, hence, it is not included in our comparisons. The saliency maps for the above algorithms were downloaded from the website [30] related to the study in [40]. We further include comparisons with a recently proposed method GP [36]. The saliency maps for GP were extracted using the code provided by the authors.

First, we provide a visual comparison of the competing algorithms in Fig 2. The images contain relatively simple images (rows 1,2), an object with a few visually distinct parts (row 3,5), an object that is highly textured (row 4) and objects touching the image boundary (row 6,7). In all images, we observe superior performance of PSE when compared to the other methods.

In Fig. 3, we provide precision–recall curves for the unsupervised state-of-the-art methods in the datasets mentioned above. One may quickly observe that although some methods such as ST perform well in relatively simpler datasets, such as SED2 and MSRA10k, they fail to provide a good performance in more complex datasets, such as JUDD and DutOmron. One of the top performing methods QCut tends to produce saliency maps that have relatively low precision on higher recalls when compared to other methods. This is evident from the quick *fall* of the curve in this region. Furthermore, due to its hard boundary background assumption in SED2 dataset, this method fails to preserve its leading performance. It is worth noting that the proposed PSE method is consistently one of the top two performing methods in precision–recall curves. PSE does not suffer from the sharp drop in precision at high recall as QCUT. Furthermore, it handles objects touching the image boundary as observed from the plots of SED2 dataset.

In Table 1, we provide maximum F_β measures for $\beta^2 = 0.3$. In three of the datasets (i.e. MSRA10k, ECSSD and SED2) PSE gives the top performance and on DutOmron and JUDD datasets, it achieves second best performance with a relatively small gap with the leading method.

Since their evaluation is based on different criteria, it is highly likely that one may observe a different leaderboard in ROC curves and AUC measures than precision–recall curves and *F*-measures.

Parts of a paper

Conclusions:

- Summary of the work
- Key observations
- Limitations and possible extensions

TABLE V
PERFORMANCE AND TRAINING TIMES ON THE COHN-KANADE DATASET

	EER ↓	mAP ↑	Time (sec)	Dimensions
KDA	22.13%	69.85%	0.014	$D \rightarrow 1$
KSR	18.49%	70.04%	0.011	$D \rightarrow 1$
CSKDA	21.11%	69.22%	0.017	$D \rightarrow 2$
CSKSR	**17.93%**	**70.12%**	0.013	$D \rightarrow 2$
Prop. CSKDA	**17.93%**	**70.12%**	0.010	$D \rightarrow 2$
Prop. I-CSKDA	**17.93%**	**70.12%**	0.010	$D \rightarrow 2$
Prop. A-CSKDA	33.19%	44.12%	0.010	$D \rightarrow 2$
Prop. D-CSKDA	**17.65%**	69.87%	0.017	$D \rightarrow N_p\text{-}1 \rightarrow 2$

TABLE VI
PERFORMANCE AND TRAINING TIMES ON THE JAFFE DATASET

	EER ↓	mAP ↑	Time (sec)	Dimensions
KDA	27.46%	58.84%	0.069	$D \rightarrow 1$
KSR	26.43%	**59.26%**	0.048	$D \rightarrow 1$
CSKDA	26.43%	**59.26%**	0.085	$D \rightarrow 1$
CSKSR	26.43%	**59.26%**	0.063	$D \rightarrow 1$
Prop. CSKDA	**26.11%**	**59.26%**	0.087	$D \rightarrow 4$
Prop. I-CSKDA	**26.11%**	**59.26%**	0.041	$D \rightarrow 4$
Prop. A-CSKDA	36.71%	35.67%	0.041	$D \rightarrow 5$
Prop. D-CSKDA	**25.79%**	**60.95%**	0.112	$D \rightarrow N_p\text{-}1 \rightarrow 0.5\ N_p \rightarrow 5$

TABLE VII
PERFORMANCE (MAP) ON THE OLYMPIC SPORTS AND HOLLYWOOD2 DATASETS

	Olympic Sports	Hollywood2
KDA	83.64% ($D \rightarrow 1$)	61.64% ($D \rightarrow 1$)
KSR	83.54% ($D \rightarrow 1$)	61.62% ($D \rightarrow 1$)
CSKDA	83.78% ($D \rightarrow 10$)	61.41% ($D \rightarrow 1$)
CSKSR	83.98% ($D \rightarrow 10$)	61.86% ($D \rightarrow 2$)
Prop. CSKDA	83.82% ($D \rightarrow 5$)	61.64% ($D \rightarrow 1$)
Prop. I-CSKDA	83.82% ($D \rightarrow 5$)	61.64% ($D \rightarrow 1$)
Prop. A-CSKDA	59.28% ($D \rightarrow 5$)	31.93% ($D \rightarrow 1$)
Prop. D-CSKDA	**85.6%** ($D \rightarrow N_p\text{-}1 \rightarrow 0.5\ N_p \rightarrow 4$)	**62.05%** ($D \rightarrow N_p\text{-}1 \rightarrow 0.6\ N_p \rightarrow 4$)

the methods, since it has been proven to be the state-of-the-art choice for BoW-based representations [58].

Similarly to the facial image verification experiments, for the proposed D-CSKDA method we have tested three architectures. The first consists of three layers with dimensions $\{D, N_p - 1, d\}$ and the remaining two consist of four layers having dimensions $\{D, N_p - 1, 0.6N_p, d\}$ and $\{D, N_p - 1, 0.5N_p, d\}$, respectively. For the calculation of the kernel matrix in the first layer we employed the RBF χ^2 kernel, while for the remaining layers of the deep architecture we employed the linear kernel. The performance of the competing algorithms is illustrated in Table VII. Overall, the proposed CSKDA solution achieves a performance equal to 83.82% and 61.64% mAP on the Olympic Sports and Hollywood2 datasets, respectively. This performance is slightly worse from that of CSKSR, which achieved performance equal to 83.98% and 61.86% mAP, and slightly better from the performance achieved by CSKDA, which achieved performance equal to 83.78% and 61.41% mAP, on Olympic Sports and Hollywood2 datasets, respectively. Similarly to the face image datasets, the proposed I-CSKDA solution provided the same performance with that of the proposed CSKDA solution and A-CSKDA provided inferior performance. Finally, the proposed D-CSKDA by exploiting multiple (hierarchical) projection layers, each trained by applying CSKDA, enhances performance

CSKSR) solutions, while its training process is faster. The proposed incremental CSKDA solution provides the same performance, while further increasing operation speed. In addition, the proposed D-CSKDA method, by exploiting deep architectures trained in a layer-wise manner by applying CSKDA, can be used in order to enhance performance. Finally, the approximate CSKDA solution, while it is not expected to provide better performance than the original one, can be suitable for applying CSKDA in large-scale datasets, due to its lower computational cost.

VI. Conclusion

In this paper, we revisited the CSKDA formulation and we showed that it is equivalent to an LRKR problem using data independent target vectors. In addition, we showed that by using the same (training data independent) target vectors in an LRKRR problem, a regularized CSKDA solution is obtained. Analysis of the nature of the data independent targets exploited in both CSKDA and regularized CSKDA, allowed us to devise a fast CSKDA solution, as well as incremental and approximate CSKDA variants. Finally, we proposed a D-CSKDA method, which is formed by multiple layers trained in a hierarchical manner by applying CSKDA. Experimental results on publicly available datasets well-suited for class-specific learning verify our analysis.

Equivalence of the Criteria in (5) and (15)

Here, we show that the maximization of $\mathcal{J}$ in (5) is equivalent to the minimization of $\tilde{\mathcal{J}}$ in (15). By using (12)–(14), $\mathcal{J}$ can be expressed as

$$\mathcal{J}(\mathbf{W}) = \frac{Tr(\mathbf{W}^T\mathbf{S}_n\mathbf{W})}{Tr(\mathbf{W}^T\mathbf{S}_p\mathbf{W})} = \frac{Tr(\mathbf{W}^T(\mathbf{S}-\mathbf{S}_p)\mathbf{W})}{Tr(\mathbf{W}^T\mathbf{S}_p\mathbf{W})} = \frac{Tr(\mathbf{W}^T\mathbf{S}\mathbf{W})}{Tr(\mathbf{W}^T\mathbf{S}_p\mathbf{W})} - 1. \quad (29)$$

From (29), it can be easily seen that maximization of $\mathcal{J}$ with respect to $\mathbf{W}$ is equivalent to the minimization of $\tilde{\mathcal{J}}$ with respect to $\mathbf{W}$. Using (1) and (12)–(14), the right part of (15) is obtained.

Appendix B
Equivalence of Regularized CSKDA to LRKRR

In Section IV-A, we have shown the equivalence between CSKDA and LRKR. Here, we show that the (l_2-norm) regularized LRKR using the target vectors satisfying (21) is equivalent to a regularized version of CSKDA. The LRKRR optimization problem is defined as

$$\hat{\mathcal{J}}_{\text{LRKRR}} = \left\|\mathbf{W}^T\mathbf{\Phi} - \mathbf{T}\right\|_F^2 + C\|\mathbf{W}\|_F^2, \quad \text{s.t.: } \text{rank}(W) \leq d \quad (30)$$

Parts of a paper

Appendices:

- In case an in-depth analysis, mathematical proof needs to be included for completeness of the work it is placed here
- It is used in order to make the 'flow' of the paper easier

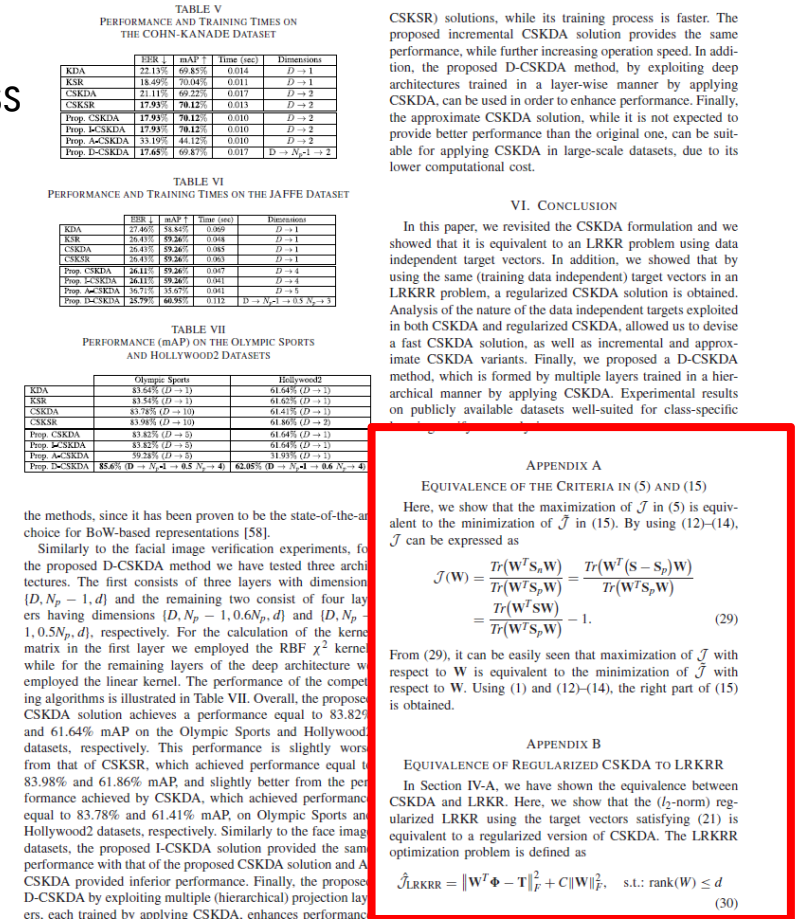


TABLE V
PERFORMANCE AND TRAINING TIMES ON THE COHN-KANADE DATASET

	EER ↓	mAP ↑	Time (sec)	Dimensions
KDA	22.13%	69.85%	0.014	$D \rightarrow 1$
KSR	18.49%	70.04%	0.011	$D \rightarrow 1$
CSKDA	21.11%	69.22%	0.017	$D \rightarrow 2$
CSKSR	**17.93%**	**70.12%**	0.013	$D \rightarrow 2$
Prop. CSKDA	**17.93%**	**70.12%**	0.010	$D \rightarrow 2$
Prop. I-CSKDA	**17.93%**	**70.12%**	0.010	$D \rightarrow 2$
Prop. A-CSKDA	33.19%	44.12%	0.010	$D \rightarrow 2$
Prop. D-CSKDA	**17.65%**	69.87%	0.017	$D \rightarrow N_p\text{-}1 \rightarrow 2$

TABLE VI
PERFORMANCE AND TRAINING TIMES ON THE JAFFE DATASET

	EER ↓	mAP ↑	Time (sec)	Dimensions
KDA	27.46%	58.84%	0.069	$D \rightarrow 1$
KSR	26.43%	**59.26%**	0.048	$D \rightarrow 1$
CSKDA	26.43%	**59.26%**	0.085	$D \rightarrow 1$
CSKSR	26.43%	**59.26%**	0.063	$D \rightarrow 1$
Prop. CSKDA	**26.11%**	**59.26%**	0.087	$D \rightarrow 4$
Prop. I-CSKDA	**26.11%**	**59.26%**	0.041	$D \rightarrow 4$
Prop. A-CSKDA	36.71%	35.67%	0.041	$D \rightarrow 5$
Prop. D-CSKDA	**25.79%**	**60.95%**	0.112	$D \rightarrow N_p\text{-}1 \rightarrow 0.5\ N_p \rightarrow 5$

TABLE VII
PERFORMANCE (MAP) ON THE OLYMPIC SPORTS AND HOLLYWOOD2 DATASETS

	Olympic Sports	Hollywood2
KDA	83.64% ($D \rightarrow 1$)	61.64% ($D \rightarrow 1$)
KSR	83.54% ($D \rightarrow 1$)	61.62% ($D \rightarrow 1$)
CSKDA	83.78% ($D \rightarrow 10$)	61.41% ($D \rightarrow 1$)
CSKSR	83.98% ($D \rightarrow 10$)	61.86% ($D \rightarrow 2$)
Prop. CSKDA	83.82% ($D \rightarrow 5$)	61.64% ($D \rightarrow 1$)
Prop. I-CSKDA	83.82% ($D \rightarrow 5$)	61.64% ($D \rightarrow 1$)
Prop. A-CSKDA	59.28% ($D \rightarrow 5$)	31.93% ($D \rightarrow 1$)
Prop. D-CSKDA	**85.6%** ($D \rightarrow N_p\text{-}1 \rightarrow 0.5\ N_p \rightarrow 4$)	**62.05%** ($D \rightarrow N_p\text{-}1 \rightarrow 0.6\ N_p \rightarrow 4$)

the methods, since it has been proven to be the state-of-the-ar
choice for BoW-based representations [58].

Similarly to the facial image verification experiments, fo
the proposed D-CSKDA method we have tested three archi
tectures. The first consists of three layers with dimension
$\{D, N_p - 1, d\}$ and the remaining two consist of four lay
ers having dimensions $\{D, N_p - 1, 0.6N_p, d\}$ and $\{D, N_p -$
$1, 0.5N_p, d\}$, respectively. For the calculation of the kerne
matrix in the first layer we employed the RBF χ^2 kerne
while for the remaining layers of the deep architecture w
employed the linear kernel. The performance of the compet
ing algorithms is illustrated in Table VII. Overall, the propose
CSKDA solution achieves a performance equal to 83.82%
and 61.64% mAP on the Olympic Sports and Hollywood
datasets, respectively. This performance is slightly wors
from that of CSKSR, which achieved performance equal t
83.98% and 61.86% mAP, and slightly better from the per
formance achieved by CSKDA, which achieved performanc
equal to 83.78% and 61.41% mAP, on Olympic Sports an
Hollywood2 datasets, respectively. Similarly to the face imag
datasets, the proposed I-CSKDA solution provided the sam
performance with that of the proposed CSKDA solution and A
CSKDA provided inferior performance. Finally, the propose
D-CSKDA by exploiting multiple (hierarchical) projection lay
ers, each trained by applying CSKDA, enhances performanc

CSKSR) solutions, while its training process is faster. The proposed incremental CSKDA solution provides the same performance, while further increasing operation speed. In addition, the proposed D-CSKDA method, by exploiting deep architectures trained in a layer-wise manner by applying CSKDA, can be used in order to enhance performance. Finally, the approximate CSKDA solution, while it is not expected to provide better performance than the original one, can be suitable for applying CSKDA in large-scale datasets, due to its lower computational cost.

VI. Conclusion

In this paper, we revisited the CSKDA formulation and we showed that it is equivalent to an LRKR problem using data independent target vectors. In addition, we showed that by using the same (training data independent) target vectors in an LRKRR problem, a regularized CSKDA solution is obtained. Analysis of the nature of the data independent targets exploited in both CSKDA and regularized CSKDA, allowed us to devise a fast CSKDA solution, as well as incremental and approximate CSKDA variants. Finally, we proposed a D-CSKDA method, which is formed by multiple layers trained in a hierarchical manner by applying CSKDA. Experimental results on publicly available datasets well-suited for class-specific

Appendix A
Equivalence of the Criteria in (5) and (15)

Here, we show that the maximization of $\mathcal{J}$ in (5) is equivalent to the minimization of $\tilde{\mathcal{J}}$ in (15). By using (12)–(14), $\mathcal{J}$ can be expressed as

$$\mathcal{J}(\mathbf{W}) = \frac{Tr(\mathbf{W}^T\mathbf{S}_n\mathbf{W})}{Tr(\mathbf{W}^T\mathbf{S}_p\mathbf{W})} = \frac{Tr(\mathbf{W}^T(\mathbf{S}-\mathbf{S}_p)\mathbf{W})}{Tr(\mathbf{W}^T\mathbf{S}_p\mathbf{W})} = \frac{Tr(\mathbf{W}^T\mathbf{S}\mathbf{W})}{Tr(\mathbf{W}^T\mathbf{S}_p\mathbf{W})} - 1. \quad (29)$$

From (29), it can be easily seen that maximization of $\mathcal{J}$ with respect to $\mathbf{W}$ is equivalent to the minimization of $\tilde{\mathcal{J}}$ with respect to $\mathbf{W}$. Using (1) and (12)–(14), the right part of (15) is obtained.

Appendix B
Equivalence of Regularized CSKDA to LRKRR

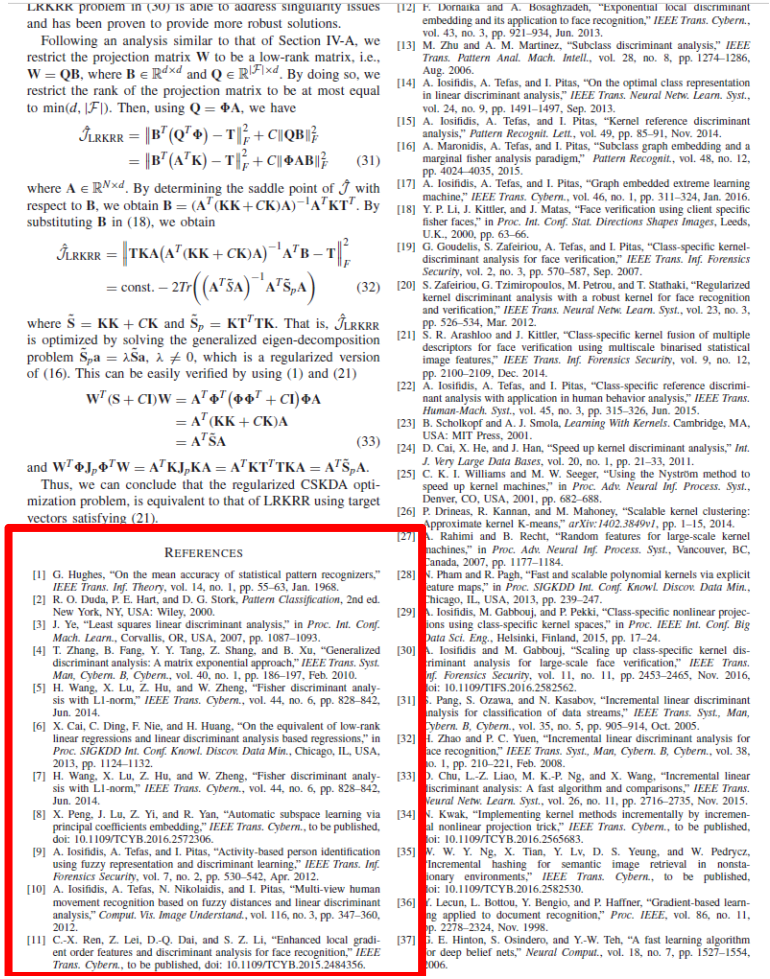
In Section IV-A, we have shown the equivalence between CSKDA and LRKR. Here, we show that the (l_2-norm) regularized LRKR using the target vectors satisfying (21) is equivalent to a regularized version of CSKDA. The LRKRR optimization problem is defined as

$$\hat{\mathcal{J}}_{\text{LRKRR}} = \left\|\mathbf{W}^T\mathbf{\Phi} - \mathbf{T}\right\|_F^2 + C\|\mathbf{W}\|_F^2, \quad \text{s.t.: } \text{rank}(W) \leq d \quad (30)$$

Parts of a paper

References:

- List of prior work



LRKRR problem in (30) is able to address singularity issues and has been proven to provide more robust solutions.

Following an analysis similar to that of Section IV-A, we restrict the projection matrix $\mathbf{W}$ to be a low-rank matrix, i.e., $\mathbf{W} = \mathbf{QB}$, where $\mathbf{B} \in \mathbb{R}^{d\times d}$ and $\mathbf{Q} \in \mathbb{R}^{|\mathcal{F}|\times d}$. By doing so, we restrict the rank of the projection matrix to be at most equal to $\min(d, |\mathcal{F}|)$. Then, using $\mathbf{Q} = \mathbf{\Phi A}$, we have

$$\begin{aligned}\hat{\mathcal{J}}_{\text{LRKRR}} &= \left\|\mathbf{B}^T(\mathbf{Q}^T\mathbf{\Phi}) - \mathbf{T}\right\|_F^2 + C\|\mathbf{QB}\|_F^2 \\ &= \left\|\mathbf{B}^T(\mathbf{A}^T\mathbf{K}) - \mathbf{T}\right\|_F^2 + C\|\mathbf{\Phi AB}\|_F^2 \end{aligned} \quad (31)$$

where $\mathbf{A} \in \mathbb{R}^{N\times d}$. By determining the saddle point of $\hat{\mathcal{J}}$ with respect to $\mathbf{B}$, we obtain $\mathbf{B} = (\mathbf{A}^T(\mathbf{KK}+C\mathbf{K})\mathbf{A})^{-1}\mathbf{A}^T\mathbf{KT}^T$. By substituting $\mathbf{B}$ in (18), we obtain

$$\begin{aligned}\hat{\mathcal{J}}_{\text{LRKRR}} &= \left\|\mathbf{TKA}\left(\mathbf{A}^T(\mathbf{KK}+C\mathbf{K})\mathbf{A}\right)^{-1}\mathbf{A}^T\mathbf{B} - \mathbf{T}\right\|_F^2 \\ &= \text{const.} - 2Tr\left(\left(\mathbf{A}^T\tilde{\mathbf{S}}\mathbf{A}\right)^{-1}\mathbf{A}^T\tilde{\mathbf{S}}_p\mathbf{A}\right) \end{aligned} \quad (32)$$

where $\tilde{\mathbf{S}} = \mathbf{KK} + C\mathbf{K}$ and $\tilde{\mathbf{S}}_p = \mathbf{KT}^T\mathbf{TK}$. That is, $\hat{\mathcal{J}}_{\text{LRKRR}}$ is optimized by solving the generalized eigen-decomposition problem $\tilde{\mathbf{S}}_p\mathbf{a} = \lambda\tilde{\mathbf{S}}\mathbf{a}$, $\lambda \neq 0$, which is a regularized version of (16). This can be easily verified by using (1) and (21)

$$\begin{aligned}\mathbf{W}^T(\mathbf{S}+C\mathbf{I})\mathbf{W} &= \mathbf{A}^T\mathbf{\Phi}^T\left(\mathbf{\Phi\Phi}^T + C\mathbf{I}\right)\mathbf{\Phi A} \\ &= \mathbf{A}^T(\mathbf{KK}+C\mathbf{K})\mathbf{A} \\ &= \mathbf{A}^T\tilde{\mathbf{S}}\mathbf{A} \end{aligned} \quad (33)$$

and $\mathbf{W}^T\mathbf{\Phi}\mathbf{J}_p\mathbf{\Phi}^T\mathbf{W} = \mathbf{A}^T\mathbf{KJ}_p\mathbf{KA} = \mathbf{A}^T\mathbf{KT}^T\mathbf{TKA} = \mathbf{A}^T\tilde{\mathbf{S}}_p\mathbf{A}$.

Thus, we can conclude that the regularized CSKDA optimization problem, is equivalent to that of LRKRR using target vectors satisfying (21).

REFERENCES

[1] G. Hughes, "On the mean accuracy of statistical pattern recognizers," *IEEE Trans. Inf. Theory*, vol. 14, no. 1, pp. 55–63, Jan. 1968.
[2] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York, NY, USA: Wiley, 2000.
[3] J. Ye, "Least squares linear discriminant analysis," in *Proc. Int. Conf. Mach. Learn.*, Corvallis, OR, USA, 2007, pp. 1087–1093.
[4] T. Zhang, B. Fang, Y. Y. Tang, Z. Shang, and B. Xu, "Generalized discriminant analysis: A matrix exponential approach," *IEEE Trans. Syst. Man, Cybern. B, Cybern.*, vol. 40, no. 1, pp. 186–197, Feb. 2010.
[5] H. Wang, X. Lu, Z. Hu, and W. Zheng, "Fisher discriminant analysis with L1-norm," *IEEE Trans. Cybern.*, vol. 44, no. 6, pp. 828–842, Jun. 2014.
[6] X. Cai, C. Ding, F. Nie, and H. Huang, "On the equivalent of low-rank linear regressions and linear discriminant analysis based regressions," in *Proc. SIGKDD Int. Conf. Knowl. Discov. Data Min.*, Chicago, IL, USA, 2013, pp. 1124–1132.
[7] H. Wang, X. Lu, Z. Hu, and W. Zheng, "Fisher discriminant analysis with L1-norm," *IEEE Trans. Cybern.*, vol. 44, no. 6, pp. 828–842, Jun. 2014.
[8] X. Peng, J. Lu, Z. Yi, and R. Yan, "Automatic subspace learning via principal coefficients embedding," *IEEE Trans. Cybern.*, to be published, doi: 10.1109/TCYB.2016.2572306.
[9] A. Iosifidis, A. Tefas, and I. Pitas, "Activity-based person identification using fuzzy representation and discriminant learning," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 2, pp. 530–542, Apr. 2012.
[10] A. Iosifidis, A. Tefas, N. Nikolaidis, and I. Pitas, "Multi-view human movement recognition based on fuzzy distances and linear discriminant analysis," *Comput. Vis. Image Understand.*, vol. 116, no. 3, pp. 347–360, 2012.
[11] C.-X. Ren, Z. Lei, D.-Q. Dai, and S. Z. Li, "Enhanced local gradient order features and discriminant analysis for face recognition," *IEEE Trans. Cybern.*, to be published, doi: 10.1109/TCYB.2015.2484356.
[12] F. Dornaika and A. Bosaghzadeh, "Exponential local discriminant embedding and its application to face recognition," *IEEE Trans. Cybern.*, vol. 43, no. 3, pp. 921–934, Jun. 2013.
[13] M. Zhu and A. M. Martinez, "Subclass discriminant analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 8, pp. 1274–1286, Aug. 2006.
[14] A. Iosifidis, A. Tefas, and I. Pitas, "On the optimal class representation in linear discriminant analysis," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 9, pp. 1491–1497, Sep. 2013.
[15] A. Iosifidis, A. Tefas, and I. Pitas, "Kernel reference discriminant analysis," *Pattern Recognit. Lett.*, vol. 49, pp. 85–91, Nov. 2014.
[16] A. Maronidis, A. Tefas, and I. Pitas, "Subclass graph embedding and a marginal fisher analysis paradigm," *Pattern Recognit.*, vol. 48, no. 12, pp. 4024–4035, 2015.
[17] A. Iosifidis, A. Tefas, and I. Pitas, "Graph embedded extreme learning machine," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 311–324, Jan. 2016.
[18] Y. P. Li, J. Kittler, and J. Matas, "Face verification using client specific fisher faces," in *Proc. Int. Conf. Stat. Directions Shapes Images*, Leeds, U.K., 2000, pp. 63–66.
[19] G. Goudelis, S. Zafeiriou, A. Tefas, and I. Pitas, "Class-specific kernel-discriminant analysis for face verification," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 3, pp. 570–587, Sep. 2007.
[20] S. Zafeiriou, G. Tzimiropoulos, M. Petrou, and T. Stathaki, "Regularized kernel discriminant analysis with a robust kernel for face recognition and verification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 3, pp. 526–534, Mar. 2012.
[21] S. R. Arashloo and J. Kittler, "Class-specific kernel fusion of multiple descriptors for face verification using multiscale binarised statistical image features," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 12, pp. 2100–2109, Dec. 2014.
[22] A. Iosifidis, A. Tefas, and I. Pitas, "Class-specific reference discriminant analysis with application in human behavior analysis," *IEEE Trans. Human-Mach. Syst.*, vol. 45, no. 3, pp. 315–326, Jun. 2015.
[23] B. Scholkopf and A. J. Smola, *Learning With Kernels*. Cambridge, MA, USA: MIT Press, 2001.
[24] D. Cai, X. He, and J. Han, "Speed up kernel discriminant analysis," *Int. J. Very Large Data Bases*, vol. 20, no. 1, pp. 21–33, 2011.
[25] C. K. I. Williams and M. W. Seeger, "Using the Nyström method to speed up kernel machines," in *Proc. Adv. Neural Inf. Process. Syst.*, Denver, CO, USA, 2001, pp. 682–688.
[26] P. Drineas, R. Kannan, and M. Mahoney, "Scalable kernel clustering: Approximate kernel K-means," *arXiv:1402.3849v1*, pp. 1–15, 2014.
[27] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, 2007, pp. 1177–1184.
[28] N. Pham and R. Pagh, "Fast and scalable polynomial kernels via explicit feature maps," in *Proc. SIGKDD Int. Conf. Knowl. Discov. Data Min.*, Chicago, IL, USA, 2013, pp. 239–247.
[29] A. Iosifidis, M. Gabbouj, and P. Pekki, "Class-specific nonlinear projections using class-specific kernel spaces," in *Proc. IEEE Int. Conf. Big Data Sci. Eng.*, Helsinki, Finland, 2015, pp. 17–24.
[30] A. Iosifidis and M. Gabbouj, "Scaling up class-specific kernel discriminant analysis for large-scale face verification," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 11, pp. 2453–2465, Nov. 2016, doi: 10.1109/TIFS.2016.2582562.
[31] S. Pang, S. Ozawa, and N. Kasabov, "Incremental linear discriminant analysis for classification of data streams," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 5, pp. 905–914, Oct. 2005.
[32] H. Zhao and P. C. Yuen, "Incremental linear discriminant analysis for face recognition," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 1, pp. 210–221, Feb. 2008.
[33] D. Chu, L.-Z. Liao, M. K.-P. Ng, and X. Wang, "Incremental linear discriminant analysis: A fast algorithm and comparisons," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 11, pp. 2716–2735, Nov. 2015.
[34] N. Kwak, "Implementing kernel methods incrementally by incremental nonlinear projection trick," *IEEE Trans. Cybern.*, to be published, doi: 10.1109/TCYB.2016.2565683.
[35] W. W. Y. Ng, X. Tian, Y. Lv, D. S. Yeung, and W. Pedrycz, "Incremental hashing for semantic image retrieval in nonstationary environments," *IEEE Trans. Cybern.*, to be published, doi: 10.1109/TCYB.2016.2582530.
[36] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[37] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.