Graz University of Technology

Institute for Computer Graphics and Vision

Dissertation

Efficient Metric Learning for Real-World Face Recognition

Martin Köstinger

Graz, Austria, December 2013

Thesis supervisors

Prof. Dr. Horst Bischof

Prof. Dr. Fernando De la Torre

The average Ph.D. thesis is nothing but the transference of bones from one graveyard to another.

Frank J. Dobie

Abstract

Real-world face recognition is a large-scale task. It is large-scale in a sense that image acquisition devices are omnipresent in our daily life. Thus, more and more images are taken every day that need to be processed, where often a human face is the object of interest. As data grows several challenges and opportunities are posed to computational face recognition algorithms, additional to the recognition challenge. A main criterion for the applicability of machine learning algorithms is the scalability in terms of learning, evaluation costs and also the needed effort to obtain labels and annotations. Scanning web-scale data sets of images containing millions of faces call for sophisticated search and retrieval strategies, where efficient algorithms are clearly beneficial.

Thus, in this thesis, we introduce novel Mahalanobis metric learning methods for real-world face recognition. The goal of Mahalanobis metric learning is to exploit prior information such as labels to learn a similarity measure that is better suited for a particular task like face recognition. In particular, we propose algorithms that offer scalability in terms of learning, evaluation and required annotation. Scalability in training is introduced by a formulation that allows for learning a Mahalanobis metric that does not rely on complex optimization problems requiring computationally expensive iterations. Our algorithm is flexible enough to learn form pairwise labels and is thus applicable for a wide range of large-scale problems. To speed up evaluation we first propose a metric-based hashing strategy and second bridge the gap between metric learning and prototype learning, enabling efficient retrieval. Further, by combining metric learning and multi-task learning we account the fact that faces share strong visual similarities, which can be exploited when learning discriminative models. Moreover, a face recognition pipeline heavily relies on face detection and landmark extraction as preprocessing steps, requiring large-scale data for training. Therefore, we introduce a publicly available database with suitable training data.

Experimental evaluations on recent face recognition benchmarks demonstrate the benefits of our methods. In particular, we are able to compare to the state-of-the-art in Mahalanobis metric learning, however at drastically reduced training and evaluation costs. Moreover, we benchmark our methods on standard machine learning, person re-identification and object matching datasets. On two benchmarks we even improve over the domain specific state-of-the-art.

Keywords: Face recognition • Metric learning • Multi-task learning • Hashing • Prototype methods • Face detection • Face verification • Facial landmark database • Machine learning

Kurzfassung

Gesichtserkennung unter unkontrollierten Bedingungen entwickelt sich zu einer omnipräsenten Aufgabe, da mehr und mehr Bilder durch Digitalkameras oder Smartphones aufgenommen werden. In vielen Fällen ist ein menschliches Gesicht der Gegenstand der Betrachtung. Um diese Unmengen von Bildern interpretieren und analysieren zu können müssen die Daten automatisch verarbeitet werden. Dies stellt gleichzeitig eine Herausforderung als auch eine Chance für computergestützte Gesichtserkennungsalgorithmen dar. Die Hauptherausforderung ist die Skalierbarkeit in Bezug auf die benötigte Trainings- und Testzeit, sowie die notwendige manuelle Annotation der Daten im Training. Große Datenmengen wie sie z.B. bei Sozialen Netzwerken auftreten benötigen intelligente Suchstrategien, die auf effiziente Algorithmen zurückgreifen.

Daher stellen wir in dieser Arbeit neue Mahalanobis Metrik-Lernverfahren vor, insbesondere für Gesichtserkennung unter unkontrollierten Bedingungen. Mahalanobis Metrik-Lernverfahren sind in der Literatur bekannt für ihre guten Ergebnisse, im Speziellen für Gesichtserkennungsprobleme. Die vorgeschlagenen Algorithmen behandeln die Thematik des effizienten Trainings, Testens, sowie der benötigten Annotationsmenge. Um das Training effizienter zu gestalten, wird eine Methode vorgestellt, die es erlaubt eine Mahalanobis Metrik zu lernen ohne dabei aufwändige iterative Optimierungsverfahren zu verwenden. Der Algorithmus ist des Weiteren in der Lage von paarweisen Annotationen zu lernen, die üblicherweise einfach erhoben werden können. Daher skaliert die Methode auch für große Datenmengen. Um die Evaluierungszeit drastisch zu verkürzen, wird erstens eine metrik-basierte Hashing-Methode vorgestellt, und zweitens vorgeschlagen Metrik-Lernverfahren mit Prototyp-Lernverfahren zu verbinden. Dies bringt deutliche Geschwindigkeitsvorteile. Des Weiteren wird eine Methode vorgestellt, die Metrik-Lernverfahren mit Multi-Task Lernverfahren kombiniert, um auszunutzen, dass Gesichter sehr starke Ähnlichkeiten aufweisen, auch wenn diskriminati-

ve Modelle gelernt werden. Darüber hinaus, stellen Gesichtsdetektion und die Extraktion von Gesichtsmerkmalen wichtige Vorverarbeitungsschritte in einem Gesichtserkennungsprozess dar, die auch große Datenmengen zum Trainieren benötigen. Für diesen Zweck stellen wir eine umfangreiche Datenbank zur Verfügung.

Die Evaluierungen und experimentellen Ergebnisse auf öffentlich zugänglichen Evaluierungsdatenbanken zeigen die Vorteile der vorgestellten Methoden. Im Besonderen sind die Methoden kompetitiv zum Stand der Technik im Bereich von Mahalanobis Metrik-Lernverfahren. Darüber hinaus werden die Trainings- und Evaluierungskosten drastisch reduziert. Um die Generalisierungsfähigkeit der Methoden zu unterstreichen, werden sie zusätzlich auch getestet für Standard Machine Learning Probleme, Personen Wiedererkennung und Objekt-Matching. Auf zwei Evaluierungsdatenbanken wird sogar der domänenspezifische Stand der Technik überboten.

Schlagwörter: Gesichtserkennung • Metrik-Lernverfahren • Multi-Task Lernverfahren • Hashing • Prototyp Methoden • Gesichtsdetektion • Gesichtsverifikation • Maschinelles Lernen

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

Place	Date	Signature

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Ort	Datum	Unterschrift

Acknowledgments

Finishing a PhD degree is like climbing on a high, unexplored mountain. It can be an exciting job even though its tough. With the right roped party, advisors and support it becomes manageable. First of all I want to thank the leader of the expedition Horst Bischof for being my supervisor and for enabling me to pitch up my tent in the research base camp, at the Institute of Computer Graphics and Vision (ICG). He engaged my interest in the amazing field of computer vision with his interesting lectures. I also want to thank Fernando De la Torre for his effort in being my second supervisor.

As time in the base camp at PhD mountain passed by many of my former fellow PhD students became my friends. I am indebted to all of them and all other staff and laboratory colleagues at ICG. Especially, I want to thank my tent mates Paul and Samuel. I will always remember the interesting, useful and in equal shares funny discussions in the lunch and coffee breaks. There was always someone having a joke or beer ready when things did not work out so well. Thank you for making my stay so pleasant. Further, I want to thank all members of the rope party, the Learning, Recognition and Surveillance group at ICG with its powerful lead climber Peter M. Roth, aka *the Boss*. Big thanks go to the other valuable supporters in the base camp: To Mike's reading group for allowing me to see the bigger picture, to the coffee group for keeping me awake with Italian mokka and also to the famous climbing group. Thanks to Mani and Sabine for the numerous after work rock pitches that kept my backbone upright. Needless to say that I want to express also my appreciation to my friends at home in Gastein, the place of my inspiration. I would like to thank all of them for the innumerable exciting moments, while snowboarding, mountaineering or climbing.

Finally, I want to thank the people who have been there all the way through, especially my girlfriend Ruth who tolerated the PhD lifestyle including all the deadline stress. Further, I want to thank my parents Manfred and Gunda, who enabled me to pursue my studies in the way I did. I will be forever grateful for their love and continuing support. Thanks to my siblings Karin, Manfred, Christoph, Wilhelm and the rest of the family for all their patience and support.

This thesis was created at the Institute for Computer Graphics and Vision at Graz University of Technology between 2009 and 2013. In that period this work was supported by the Austrian Science Foundation (FWF) project Advanced Learning for Tracking and Detection in Medical Workflow Analysis (I535-N23), by the Austrian Research Promotion Agency (FFG) projects Multimedia Documentation Lab (818800) and Human Factors Technologies and Services (2371236) and by the Public Employment Service Austria (AMS).

Contents

List of Figures

List of Tables

1 Introduction

Face recognition is an intrinsic part of the human visual perception and definitely one of our core abilities. Imagine looking at a portrait photo of yourself without noticing that it is you on the picture. Even worse, in your daily life you meet familiar people and see faces day in, day out absent of the capability to recognize these without other available cues as voice, hairstyle, gait, clothes or context information. In the human brain dedicated areas exist that offer us our remarkable face recognition capabilities. If there is a lesion in these specialized areas the described symptoms can occur, commonly referred as prosopagnosia or face blindness. Face blindness can occur despite the absence of low-level visual inabilities or cognitive alterations as loss of memory or mental confusions [92].

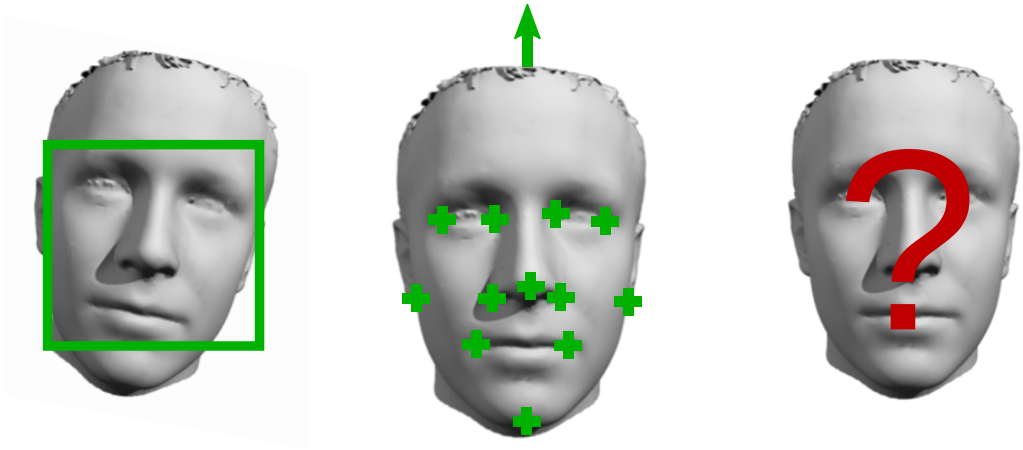
The significance of face recognition for humans is reflected by the variety of applications of computational face recognition. In the field of computer vision face recognition is omnipresent since decades and hence can be considered as one of the core tasks. For instance, it builds the basis for many law enforcement or commercial applications. In biometrics face recognition is of particular interest as it can be performed non-intrusive, without the cooperation or even the knowledge of the respective subject. Therefore, it has a wide range of applications such as access controls or video surveillance related tasks. Another field of application is the rapid evolving of consumer digital photography that leads to loose unlinked personal photo collections. Here face recognition software helps to automatically organize the collections and find your loved ones. Another important application domain is health care as the aid for visually impaired people, i.e., humans that suffer from face blindness.

For humans the recognition of a familiar face is straight forward, it has been even ob-

served that humans are able to reliably match familiar faces solely based on appearance in a variety of challenging situations [126]. For instance humans are able to recognize faces from low resolution images where only the overall face structure is apparent and fine face details are not necessarily preserved. Further, it has been observed that the human visual system is poor at matching photographs of unfamiliar persons under degradations. However, for familiar people the human visual system is able to tolerate severe degradations as variations in pose, lighting or occlusions.

Computational face recognition tries to mimic the remarkable capabilities of the human brain. The progress in face recognition research was monitored by several large scale evaluations as Face Recognition Technology (FERET) [112] or Face Recognition Vendor Test (FRVT) [114]. In the beginning the clear focus was on controlled studio like conditions. In these early face recognition systems performed already relatively well. The goal of each evaluation was to drop the error rates to one tenth of the preceding one. Ultimately, in 2006 it was observed for the first time that computational face recognition is able to beat the human capabilities, at least in some constrained situations. In particular, it has been shown that for frontal still-face images some algorithms are able to surpass the human recognition performance under illumination changes [114]. However, this result is not general and is only valid for controlled studio-alike high-resolution images. In contrast, in unconstrained real-world situations there is still a large performance gap [82]. Here imaging conditions as diversity in viewpoint, lighting, clutter or occlusion severely lower the recognition performance. Therefore, the study of face recognition under real-world conditions is the way to go. This means recognition from medium to low-resolution 2D images of potentially uncooperative subjects, in uncontrolled environments without available context information. For this purpose several large-scale benchmark databases have been proposed exhibiting large variability [5, 37, 48, 58, 82]. Despite the criticism that the datasets are still somewhat constrained, mainly by the way the images have been collected, humans still have by far the best real-world face recognition rates also on these datasets.

Real-world face recognition is commonly structured as a multi-stage process. Typically, it involves Face Detection, Alignment and Recognition (DAR). See Figure 1.1 for illustration. Depending on the particular application these steps are followed or accompanied by face tracking, pose estimation, emotion recognition or gaze estimation. In particular, face detection means to estimate the coarse location of one or multiple faces present in the image. Face alignment or normalization is the process to detect facial landmarks such as the corners of the eyes, mouth and nose. These can be used



(a) Detection (b) Alignment (c) Recognition

Figure 1.1: Real-world face recognition is typically structured as a Detection, Alignment and Recognition (DAR) process. (a) Face detection means to estimate the coarse location of faces in images. (b) Face alignment is to align the faces to a canonical pose or to extract local characteristics. (c) The recognition step performs either face verification or face identification.

to align the faces or to extract local characteristics, component wise. Furthermore, the extracted landmarks can be used to project the faces to a canonical pose. Accurate detection and alignment is very valuable for the final recognition performance. Obviously, face recognition is practically impossible without prior face detection. Also face alignment is of critical importance for face recognition as observed by many authors [6, 59, 133, 152, 164], especially in the context of real-world face recognition. Finally, the recognition step performs either face verification or face identification. Face verification or authentication means deciding if two images show the same person or not. This involves assessing the similarity between the two faces explicitly. Face identification deals with assigning a name or label to a particular query face, e.g., this face shows Jim. Intuitively, this requires a set of labeled gallery faces or a watch list. Typically, in real-world applications these galleries or watch lists are large-scale. Thus, additionally to the face verification and identification challenge computational face recognition has to deal with two main issues: Scalability and the ability to benefit from large-scale data in learning face recognition models.

Real-world face recognition is a large-scale task It is large-scale in a sense that image acquisition devices are omnipresent in our daily life. More and more photos are taken

every day that need to be processed. In many cases a human face is the object of interest. For personal photo collections images containing faces are taken and immediately uploaded to social networks. Surveillance cameras take images constantly at nearly every public hot spot. Millions of people cross borders each year and are pictured thereby in many countries. As data grows several challenges and opportunities are posed to computational face recognition algorithms.

Real-world face recognition benefits from sophisticated machine learning For face recognition we can exploit the available large-scale data for training and apply sophisticated machine learning algorithms that exploit the special characteristics of faces. This usually leads to better results and lower error rates. However, this is challenged by the computational burden and the needed effort to obtain labels and annotations. In some cases labels can be inferred automatically, e.g., from social network context [129], tracking [93] or with other context knowledge. Other algorithms are able to learn from pairwise labels which reduces the required level of supervision. Nevertheless, the learning time remains an issue at least for on-line applications or applications with limited time budget. Ultimately, for some applications the learning time may be not too critical. Nevertheless, one important aspect that is often neglected is the computational burden at test time. Scanning web-scale data sets of images containing millions of faces raises the demands on efficient search and retrieval times. Also for face detection and alignment large-scale training data is very valuable [77]. However, accurately annotated and labeled real-world data is often not publicly available.

1.1 Contribution

Addressing the above discussed problems we propose in this thesis learning strategies that enable single-image real-world face recognition on large-scale. In particular, we focus on scalable algorithms that allow for efficient training and evaluation. Further, we are also concerned with benefiting from the available large-scale data. The content of this thesis is mainly based on the work presented in [74, 75, 76, 78, 79]. Overall, this work is the result of a strong long-lasting collaboration with my colleagues Paul Wohlhart, Peter Roth, Horst Bischof and others. The main contributions of this thesis are the following:

- For accurate face detection and facial landmark extraction we propose a novel large-scale, real-world database termed Annotated Facial Landmarks in the Wild

(AFLW). Especially for face detection and landmark localization recent works rely heavily on machine learning algorithms using massive amounts of data, e.g. Huang et al. [56] require 75,000 faces to train their face detector. Thus, ultimately a key step for face recognition is also the availability of training data in large-scale. Unfortunately in recent developments little attention has been paid to the public availability of suitable training data. Once having introduced AFLW the intention is to show that existing detectors are not limited by their models but by the available training data. In particular, we are able to achieve a drastically increased face detection performance, using a standard algorithm with standard features. We are even able to outperform sophisticated state-of-the-art methods on the Face Detection Dataset and Benchmark (FDDB), that also use large training sets. This work was published in [74]. We present it in Chapter 3.

- To efficiently train face verification and identification models we propose a novel Mahalanobis metric learning method. Mahalanobis metric learning recently demonstrated competitive results for a variety of face recognition tasks. In particular we raise issues on the scalability and the required degree of supervision of existing Mahalanobis metric learning methods. Typically, learning metrics requires often to solve complex and thus computationally very expensive optimization problems. Further, if one considers the constantly growing amount of data it is often infeasible to specify fully supervised labels for all data points. Instead, it is easier to specify labels in form of equivalence constraints Thus, we introduce a simple though effective strategy to learn a distance metric from equivalence constraints, based on a statistical inference perspective. In contrast to existing methods the method does not rely on complex optimization problems requiring computationally expensive iterations. Hence, in training it is orders of magnitudes faster than comparable Mahalanobis metric learning methods. This work was published in [75]. We present it in Chapter 4.

- To speed-up the evaluation for Mahalanobis metric learning for face recognition, we address the problem of efficient k-NN classification. In particular we introduce two methods. First, we propose a metric-based hashing strategy, allowing for both, efficient learning and evaluation. In fact, if the intrinsic structure of the data is exploited by the metric in a meaningful way, using hashing we can compact the feature representation still obtaining competitive results. This work was published in [78]. We present it in Chapter 4. Second, we propose to represent

the dataset by a fixed number of discriminative prototypes. In particular, we introduce a new method that jointly chooses the positioning of prototypes and also optimizes the Mahalanobis distance metric with respect to these. We show that choosing the positioning of the prototypes and learning the metric carefully leads to a drastically reduced effort while maintaining the discriminative essence of the original dataset. This work was published in [79]. We present it in Chapter 6.

- To learn better face recognition models, we address the problem, neglected by most face recognition approaches, that faces share strong visual similarities. This can can be exploited when learning discriminative models. Hence, we propose to model face recognition as multi-task learning problem. This enables us to exploit both, shared common information and also individual characteristics of faces. In particular, we extend our Mahalanobis metric learning method to multi-task learning. The resulting algorithm supports label-incompatible learning which allows us to use the rather large pool of anonymously labeled face pairs to learn a more robust distance measure. Second, we show how to learn and combine person specific metrics for face identification improving the classification power. This work was published in [76]. We present it in Chapter 5.

1.2 Outline

This thesis is organized as follows: First, in Chapter 2 we review the related literature in real-world face recognition. An interesting aspect of our review is that we analyze the related works different from previous surveys. In particular we discuss the methods according to the feature representation, the applied machine learning algorithms and the face-specific recognition strategies. Second, in Chapter 3 the primary focus is on our AFLW database tailored to face detection and landmark localization that mitigate the issue of no publicly available large-scale real-world database for face detection and landmark localization. Once having introduced the face database the intention is to show that only having the large-scale data drastically increases face detection performance. Third, in Chapter 4 we propose a novel Mahalanobis metric learning algorithm that circumvents common problems in Mahalanobis metric learning. First, learning metrics often requires to solve complex and thus computationally very expensive optimization problems. Second, as the evaluation time linearly scales with the size of the data k-NN search becomes cumbersome for large-scale problems or real-time applications with limited time budget. Next, in Chapter 5 we address the problem that most face

recognition approaches neglect that faces share strong visual similarities, which can be exploited when learning discriminative models. Succeeding, in Chapter 6 we introduce a new method that jointly chooses the positioning of prototypes and also optimizes the Mahalanobis distance metric with respect to these. Finally, in Chapter 7 we summarize and conclude this thesis. Further, we provide an outlook to potential future works.

2 Real-World Face Recognition Review

In this chapter we briefly review related works in classical and real-world face recognition. While face recognition under controlled conditions is commonly considered as solved [49, 111] real-world face recognition remains still an unsolved challenge. The classical grouping of face recognition follows psychological (human) face perception. In particular, Zhao et al. [164] categorizes into holistic, facial feature based and hybrid approaches. Holistic methods perceive the face as whole, without considering its parts differently. Seminal works in this category are eigenfaces [135], based on principal component analysis [109], or fisherfaces [4] based on fishers linear discriminant [40]. Facial feature based methods identify face parts such as eyes, nose, mouth to extract geometry and/or local appearance information. This is motivated by findings in psychological neuroscience where it has been observed that particular parts of the face are more important for recognition than others [126, 164]. Early facial feature based works focus only on the spatial configuration using a number of geometric measurements [70, 72]. Later approaches, e.g. elastic bunch graph matching (EBGM) [148] capture also local appearance. Hybrid methods combine the holistic and feature based paradigms. The modular eigenfaces approach of [110] extracts on holistic level eigenfaces plus the local pendants called eigenfeatures. Others [18, 84] consider shape parameters, local appearance and shape-normalized global appearance for classification. Hereby, active shape models (ASM) [17] or active appearance models (AAM) [19, 28, 31] are used to find the shape parameters. Zhao et al. [164] reason that hybrid approaches promise superior performance compared to either holistic or feature based methods.

These seminal approaches point up important directions in face recognition research as part-based representations or face normalization. Further, in various evaluations [112] these performed reasonable for frontal faces under controlled conditions. However, in unconstrained real-world situations these are likely to fail as observed in real-world benchmarks such as Labeled Faces in the Wild (LFW) [58]. Here challenges as a limited resolution, harsh lighting, facial aging or uncooperative subjects severely lower the recognition performance. Recent methods focus on appropriate representations and sophisticated machine learning algorithms that are able to deal with these real-world challenges. Further, face-specific recognition strategies are applied that use an auxiliary set of faces for improved matching. Thus, in contrast to the psychological face perception grouping, we analyze related literature differently. In particular, the representation, the applied machine learning algorithms and the special recognition strategies. Further a brief overview of recent face recognition benchmarks is given that monitor the performance in real-world face recognition research.

(a) Controlled (b) Real-world

Figure 2.1: **Face recognition under controlled conditions (a) versus real-world face recognition (b)**. In realworld face recognition factors as expression, viewpoint, lighting, clutter or occlusion pose a challenge. The images are taken of FERET [112] and PubFig [82].

2.1 Representation

This section investigates which representations are favorable for real-world face recognition. First, the group of hand-crafted features are reviewed. Second, also the learned representations are analyzed which are especially tailored to face recognition.

2.1.1 Hand-Crafted Features

Several generic feature descriptors known from the computer vision literature are used for face recognition. These include pure intensity and gradient values, Haar-like features [137], Gabor wavelets [26], Local Binary Patterns (LBP) [106] and its extensions, Locally Adaptive Regression Kernels (LARK) [123], Scale Invariant Feature Transform (SIFT) [89] and 2D Discrete Cosine Transform (DCT) [45] coefficients. These hand-crafted descriptors are generic and have not been tailored especially to face recognition. Thus, it is intended to review the reported performance of the different descriptors on a standard real-world face recognition benchmark. In particular the face verification performance is compared on Labeled Faces in the Wild (LFW) [58] as it offers an accurately defined evaluation protocol. Further, LFW offers many recently published results. Other benchmark datasets are either not challenging enough or are not evaluated dense enough. To draw fair comparisons the raw descriptor performance is reported before discriminative learning. To that end this includes results obtained by unsupervised distance measures as the Euclidean, Hellinger or Chi-Square distance. Still one inherent drawback is that the published results use a different preprocessing in terms of face detection, alignment and feature normalization. This results in critical performance differences. Thus, we report the performance of all feature types and compare the relative performance within the respective publications.

Table 2.1 provides an overview of the reported results on LFW for a variety of features. It includes Gabors, LARK, the family of LBP and gradient orientation histograms as SIFT, HOG and HOG-like features. Feature types that have been rarely used on LFW are omitted such as DCT coefficients or Haar-like features. It is not possible to compare them objectively to others as in the respective publications no comparisons are made to other feature types.

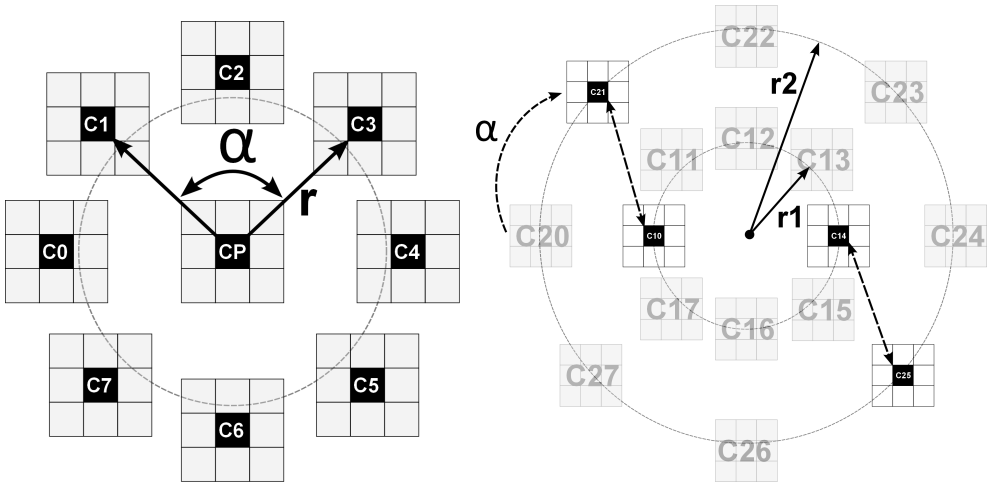
Wolf et al. [150, 151] and Taigman et al. [133] report the performance of a variety of feature types as LBP, SIFT and Gabor and compare to their proposed three-patch LBP (TPLBP) and four-patch LBP (FPLBP) extensions. Prior to matching, the faces have been normalized by a facial landmark based alignment using an affine transformation. The TPLBP and FPLBP extensions are illustrated in Figure 2.2. Interestingly, the different descriptors show only modest performance differences. Between LBP and SIFT there is only a slight difference. In [150, 151] the proposed TPLBP reaches the best performance followed closely by SIFT. TPLBP and SIFT perform best with the Euclidean distance. In their follow up paper [133] LBP performs slightly better than SIFT, both using the Hellinger distance. The TPLBP and FPLBP extensions perform worse.

	[150, 151]	[133]	[124]	[50]	[74]	[100]	[11]
Intensity	-	-	-	-	61.80%	65.67%	-
LBP [106]	68.24%	**70.85%**	68.53%	68.13%	**70.13%**	**70.27%**	**72.35%**
TPLBP [149]	**69.26%**	68.93%	68.78%	66.90%	-	-	-
FPLBP [149]	68.65%	68.35%	-	67.37%	-	-	-
HOG [24]	-	-	-	-	65.68%	-	71.25%
HOG-like [39]	-	-	-	-	68.43%	-	-
SIFT [89]	69.12%	70.82%	**71.05%**	**68.50%**	-	-	-
Gabor [26]	68.49%	-	-	-	-	69.42%	68.53%
LARK [123]	-	-	70.98%	-	-	-	-

Table 2.1: Face verification accuracy on the Labeled Faces in the Wild (LFW) [58] benchmark. For simplicity of presentation we report only the best score of the particular feature type over different standard distance measures as the Euclidean, Hellinger or Chi-Square distance. Please note that not all distance measures are reported in every publication. Numbers in bold indicate the best performing feature type of the respective publication. See text for additional details.

Seo and Milanfar [124] propose to use the LARK representation for face recognition. LARK encodes the gradient self-similarity in a local neighborhood. Therefore, the geodesic distance is measured between a center and its surrounding pixels. An illustration of the LARK descriptor is given in Figure 2.3. The evaluation shows that the performance of LARK is similar to SIFT. The authors claim that LARK improve even over SIFT if the face images are mirrored and also considered for matching. In particular the similarity score is the max over the similarity obtained with non-mirrored and mirrored images. For the normal (non-mirrored) images SIFT performs slightly better than LARK, while LBP perform modestly worse. The best results for all descriptors are obtained using the Hellinger distance. Also Guillaumin et al. [50] notices that SIFT performs slightly better than LBP. The relative results for the Euclidean, Hellinger and Chi-Square distance are the same. Also the absolute numbers are very similar. The best results are obtained with the Chi-Square distance both for SIFT and LBP.

Nguyen and Bai [100] and Köstinger et al. [74] report considerably worse performance for intensity values compared to other feature types as LBP, HOG or Gabor wavelets. In both works LBP perform best. The other feature types such as Gabors, HOG or HOG-like features perform between LBP and the intensity values. Unfortu-

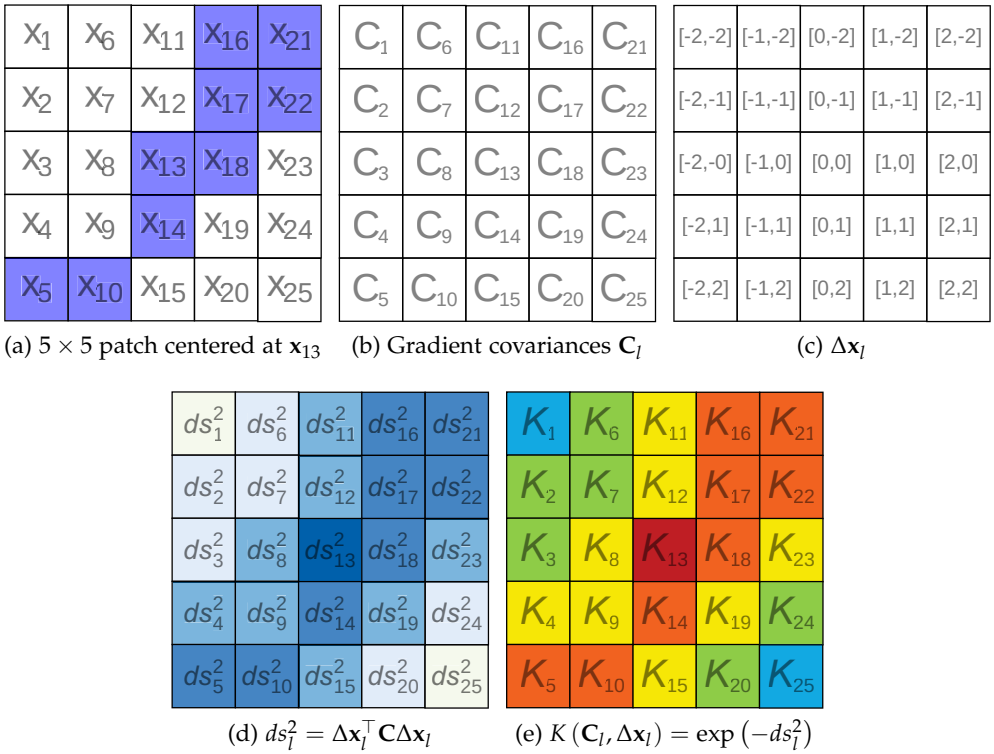


(a) Three-Patch LBP (TPLBP) (b) Four-Patch LBP (FPLBP)

Figure 2.2: Three-Patch LBP (TPLBP) and Four-Patch LBP (FPLBP). For each pixel or location a binary code is computed. (a) The computation of a bit value for TPLBP involves three patches. A bit value is assigned according to which of the two outer patches (C_1,C_3) is more similar to the central patch C_p. The patches are compared holistic. (b) The computation of a bit value for FPLBP involves four patches. In particular two pairs of patches are compared between an inner and outer ring with different radii (r_1,r_2). The bit value is assigned according to which of the two pairs is more similar to each other. Figure adapted from [151].

nately no comparison to SIFT is made. In [74] the LBP are matched with the Chi-square distance and [100] obtains the best results for the Gabors using Hellinger distance. Gabor and HOG features obtain the best results using the Euclidean distance. Also, Cao et al. [11] reports the performance for HOG features. Compared to LBP and Gabors they perform better than Gabor wavelets and modestly worse than LBP. In this work all descriptors are solely compared using the Euclidean distance.

If one recapitulates the different reported results it is obvious that there is only a modest difference in face verification accuracy for the different descriptor types. Also the different generic distance measures as Euclidean, Hellinger or Chi-Square have only a minor influence on the performance. Face representations based on LBP and SIFT seem to perform best. In 3 of 4 cases better results are reported for SIFT compared to LBP although there is only a slight difference. Thus, it is not clear which of the two should be favored. Further, the results reveal that LBP perform better than Gabors. This finding suggests that also SIFT should be favored over Gabors, although there is only one direct comparison. Further the results show that LBP and suggest that SIFT are superior to HOG and HOG-like features. The advantage of SIFT over HOG may be



(a) 5×5 patch centered at $\mathbf{x}_{13}$ (b) Gradient covariances $\mathbf{C}_l$ (c) $\Delta\mathbf{x}_l$

(d) $ds_l^2 = \Delta\mathbf{x}_l^\top \mathbf{C} \Delta\mathbf{x}_l$ (e) $K(\mathbf{C}_l, \Delta\mathbf{x}_l) = \exp(-ds_l^2)$

Figure 2.3: Locally Adaptive Regression Kernels (LARK) [123, 124]. A LARK descriptor encodes the gradient self-similarity between a center pixel and its neighborhood. For each surrounding pixel l the geodesic distance $ds_l^2 = \Delta\mathbf{x}_l^\top \mathbf{C}_l \Delta\mathbf{x}_l$ is computed. (b)-(c) Whereby $\Delta\mathbf{x}_l$ is the according spatial offset and $\mathbf{C}_l$ is the local gradient covariance matrix computed from a local analysis window centered at l. (e) The distance is transformed into a similarity via an exponential function. Finally, the LARK descriptor is a concatenation of the similarity values between the center pixel and its neighborhood. Figure adapted from [124].

accounted to the fact that SIFTs are extracted locally at facial landmarks, which allows to compensate holistic misalignments. Finally, intensity values perform worse than all other feature types.

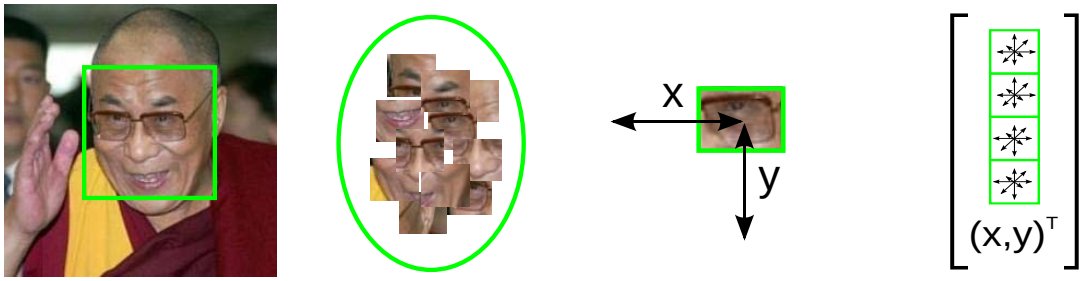
2.1.2 Feature Learning

The goal of feature learning is to obtain a representation that is better suited for face recognition compared to generic descriptors. Among others the methods involve codebook based approaches, methods that perform a large-scale feature search or that learn human-interpretable attributes. Some of these methods are unsupervised and need only the face images for training. Others are supervised and require additionally labels in

form of equivalence constraints or even in form of class labels.

The unsupervised codebook method of Sanderson and Lovell [121] describes faces by multi-region probabilistic histograms of visual words. A face is therefore divided into several regions which are further subdivided into small blocks. In these blocks 2D-DCT coefficients are extracted as low-level features. These are used to train a Gaussian mixture model by expectation maximization as visual codebook. For a face region the final coding is obtained by forming a histogram of the encoded block features by soft assignment. The face description is a concatenation of the region histograms.

Also Li et al. [85] learn an unsupervised visual codebook by Gaussian Mixture Model (GMM) for their elastic matching method. In particular, the elasticity is introduced by augmenting the appearance descriptors with the relative location inside the face. Then, the GMM with a fixed number of components is trained to maximize the likelihood of the location and appearance information. Li et al. [85] call this GMM Universal Background Model (UBM-GMM). For matching they propose two different strategies namely Probabilistic Elastic Matching (PEM) and Adaptive Probabilistic Elastic Matching (APEM). In the PEM matching for each GMM component the maximum likely feature in the face is assigned. To match a face pair the difference vector is computed. Finally, the difference vector is classified by SVM with RBF kernel. In contrast, the APEM matching requires to train a further GMM model at test time, termed Adaptive-GMM (A-GMM). In particular, the A-GMM is trained to maximize the likelihood of the features of the face pair under consideration. Further, the A-GMM components have to be likely under the UBM-GMM. See Figure 2.4 for illustration.



(a) Face Detection (b) Patches (c) Augmenting Location (d) Description

Figure 2.4: Visual codebook based on Gaussian Mixture Models (GMM) for elastic matching [85]. Elasticity is introduced by augmenting the appearance descriptors with the relative location inside the face.

Cao et al. [11] address the problem that hand crafted-encodings based on generic descriptors as LBP or HOG have no uniform distributed code histogram. Therefore, the

resulting encodings are not compact and less discriminative. Thus, the authors propose to learn a more uniform codebook with a random projection tree [41]. The uniformity criterion is implemented in the inner split nodes of the tree. In particular, they encode normalized DoG filter responses extracted in ring based patterns around each pixel. The encodings form a code histogram which is later used to measure the region based face similarity. The dimensionality of the code histogram is further reduced by PCA. Prior the code histogram is normalized to unit length. The authors show that in this case the PCA compression is even able to improve the face recognition performance. For matching the face regions are aligned separately based on two landmarks. Therefore nine facial landmarks are extracted. Further a pose-adaptive matching strategy is applied which involves to automatically determine the coarse face pose, in facing left, frontal or right. The pose classifier is trained on an auxiliary set of faces. For each of the nine possible combinations of face poses a specific classifier is trained, e.g. one that compares a left facings face to frontal. Similarly Hussain et al. [61] propose to learn an unsupervised codebook by k-means quantizing their high-dimensional Local Quantized Patterns (LQP) descriptors. LQP is a LBP extension that uses a larger neighborhood and more quantization steps, thus the dimensionality is drastically higher compared to standard LBPs. Hence, in practice the features are simply too high-dimensional. The encodings form a code histogram which is latter used to measure the region based face similarity. The dimensionality of the code histogram is further reduced by PCA. After dimensionality reduction the features are sphered and matched with the cosine distance.

Nowak and Jurie [105] learn a discriminative visual dictionary for faces. The key idea is that the discriminative information helps to optimize the encoding in a way that different images of the same person have a very alike code histogram. Therefore, the authors propose to train an extremely randomized clustering forest [97] based on face pairs labeled same or different. Therefore a local face representation is considered. Intuitively, face patches labeled same should impact in the same leaf nodes while dissimilar patches should impact in different leaf nodes. The authors propose appearance based split nodes based on SIFT features and additionally geometry based split conditions that monitor position and scale of the corresponding patches. Local patch correspondences between a pair of faces are established by normalized cross correlation (NCC). Finally, to decide if a face pair matches the global cluster membership histogram is used as distance measure. The distance is small if many face patch pairs impact in same leaves. A weighting factor balances the contribution of each leaf, trained by a linear SVM.

Nair and Hinton [99] propose an unsupervised, generative feature extraction model

based on Restricted Boltzmann machines (RBM) [53, 128]. RBMs are a special form of neural networks that aim to reconstruct the input data based on one layer of hidden units. The architecture of RBMs is restricted to enable fast training. In particular, no connections between hidden units and also between the inputs are allowed, so the architecture forms a bipartite graph. The higher order correlations between the inputs and the hidden units are captured by symmetrically weighted connections. In training the weights for the connections are learned iteratively. Therefore first the inputs trigger the activation for the hidden units over the weighted connections. Then, the activated hidden units trigger a reconstruction of the inputs. The correlation between the pairwise activations drive the weight updates. In testing the input image simply triggers the activations which are taken as feature responses. The feature vectors are compared by cosine distance.

Cox and Pinto [20] perform a brute-force feature search to determine better features for face recognition. Therefore, a vast number of feature proposals is randomly generated and screened on a validation set. The screened feature types incorporate single and multi-stage stacked architectures. The single stage features referred as V1-like should resemble the first order description of the human primary visual cortex V1. In particular these are normalized Gabor wavelets over various orientations and spatial frequencies. The multi-stage features build on a feed-forward architecture each including a cascade of linear and nonlinear operations. Intuitively, the input for the next stage is the output of the previous stage. Each stage performs the succeeding steps filtering, activation, spatial pooling and normalization. In these steps all parameters are randomly drawn from a range of meaningful parameters. First, the linear filtering step applies a bank of filters and generates a stack of feature maps. The filter shapes and numbers of filters is chosen randomly. Second, in the activation stage the output values are clipped to be within a defined interval. Third, in the pooling step the activations within a randomly chosen neighborhood are spatially combined. Finally, the output of the pooling step is normalized. The final output size of the different feature proposals ranges from 256 to 73,984 dimensions depending on the generated parameters. For recognition different element-wise comparison functions as the absolute difference, squared difference, square root of absolute difference and element-wise product are combined and weighted by SVM. Further, it is suggested to produce a blended classifier and combine a number of different feature proposals.

Kumar et al. [82] describe faces by human-interpretable visual attributes and so called "similes" that describe face similarity in relation to a set of reference people. The

main idea is that the responses of the attribute and "simile" classifiers agree better across pose, illumination and expression compared to generic descriptors. Hereby, the attribute classifiers recognize the presence of 65 face traits such as hair color, gender, race, and age. The classifiers are trained on a gallery set requiring manual labellings for the binary attributes (present/absent) and the corresponding relevant face regions. Thus, a tremendous amount of hand annotated labels is needed for training. As low-level features histograms of normalized pixel values, image gradient orientations and edge magnitudes are extracted. For a specific binary attribute the classification whether the attribute is present or absent is done by a linear SVM. The simile classifiers relate a face or face regions to a set of reference people, e.g. this nose is similar to that of Brad Pitt. Therefore the classifiers are trained to distinguish a person from the general population. As low-level features the same description as for the attributes is chosen. Also the classifier is a linear SVM. One advantage of the similes compared to the attributes is that these require only the names labeled not each individual attribute. Finally, the high-level face representation obtained with the attributes and similes is used for face recognition. In particular for face verification a SVM with RBF kernel is trained on feature differences.

Berg and Belhumeur [6] pick up the idea of "similes" and learn a large set of identity classifiers to describe faces. In particular, the identity classifiers are trained to distinguish two people which the authors refer to as as Tom-vs-Pete classifiers. To train the classifiers this requires a reference set of faces disjoint by identity from the test set. Depending on the size of the reference set plenty one-vs-one classifier combinations are possible. Nevertheless, evaluating all classifiers is computationally very expensive and furthermore many of those classifiers are not complementary. Thus, the authors propose to select a reasonable sized set of Tom-vs-Pete classifiers by AdaBoost. For a single binary Tom-vs-Pete classifier SIFTs are used as low-level features followed by a linear SVM as classifier. The SIFTs are extracted at reference points using multiple scales. Further, the one-vs-one SVM classification is trained on pairwise feature differences. In particular on absolute differences and element-wise products between the faces. The output of the Tom-vs-Pete classifiers is concatenated as face description. The final classification whether two faces match or not is done by SVM with RBF kernel. An important step in the Tom-vs-Pete pipeline is accurate face alignment. In particular the authors propose a method referred as identity preserving alignment. It is argued that a direct normalization of the detected facial landmarks to a neutral configuration is too strict and hence discards discriminative features. Therefore, the alignment procedure takes advantage of

the reference set by finding most similar facial landmark configurations. Thus, each face in the reference set (20,639 images) is annotated with 95 facial landmarks, 55 inner and 40 outer points. The inner points are used to find the similar landmark configurations. The outer points show the face contour but are not as accurately defined. Thus, these are augmented of the reference dataset. Once, the similar configurations are found the mean configuration is used for aligning the face to a neutral pose.

Analyzing the published feature learning results in the face recognition literature it is obvious that the authors largely agree that feature learning is beneficial for real-world face recognition. Thus, there is a consensus that a proper description plays a key role in obtaining good performance. Nevertheless, it has been observed that a single learned representation is not enough to obtain state-of-the-art performance. Therefore, in many cases the learned representations are further augmented with other complementary also generic feature types to obtain state-of-the-art performance.

2.2 Machine Learning

This section investigates recent advances in real-world face recognition with focus on successfully applied machine learning algorithms. Ideally the algorithms are able to exploit the obtained face representation for robust face verification or identification. Hereby, main objectives are to determine which face regions are more important for recognition than others, the selection of complementary classifiers, learning distance measures that are especially tailored to compare faces and combining multiple different representations and classifiers by blending strategies.

Jones and Viola [67] learn a local face similarity measure by boosting. The boosting process determines which face parts are meaningful for similarity computation. The authors argue that certain regions such as eye brows, nose or lips contain more discriminative information than others. In particular AdaBoost is applied to select local features which compare regions between face pairs. Therefore, the features measure absolute differences between Haar-alike filter responses. The filters are extracted over a range of locations, scales and orientations. Ideally for a matching face pair the filter responses agree, while they disagree for a non-matching face pair. An associated threshold monitors which intrapersonal variations are acceptable and which are unacceptable. The filter response plus threshold composes a weak-classifier for the boosting process. Once a weak-classifier is chosen a weight is assigned for a positive and negative response. The finally obtained classifier sums up the respective weights based on the individual

region based filter response.

Berg and Belhumeur [6] select a meaningful and reasonbable sized combination of their Tom-vs-Pete classifiers by AdaBoost. The Tom-vs-Pete classifiers are used to describe a face relative to a reference set of faces. In particular, each Tom-vs-Pete classifier describes if a specific face region is more similar to one of two faces of different people of the reference set. Therefore, plenty one-vs-one combinations are possible. The authors argue that many of those are not complementary and evaluating all would simply take to long. Therefore, first for each pair of people in the reference set a short list of favorable Tom-vs-Pete is generated by AdaBoost. Then, the short lists are combined by a heuristic to an overall list of Tom-vs-Pete classifiers. To obtain the short list the pre-trained. Tom-vs-Pete classifiers are evaluated if these are able to separate the two classes. As positive class the one is chosen that obtains better classifier scores in median. The threshold is fixed at at equal error rate. Next, AdaBoost is run to select the Tom-vs-Pete classifiers iteratively for each pair of people. Once this procedure has been repeated for all or a number of pairs of persons the overall list is generated by taking the best ranked elements of the individual short lists. Finally, the list of Tom-vs-Pete classifiers is condensed by taking only unique elements and also pruned if desired.

Another popular machine learning technique widely used for real-world face recognition is Mahalanobis metric learning, which aims at improving k-NN classification by exploiting the local structure of the data. Compared to other classification models Mahalanobis Metric Learning provides with k-NN search not only reasonable results but is also inherently multi-class and directly interpretable, based on the assigned neighbors. Compared to linear SVMs on difference vectors the learned metric is able to exploit a more general metric structure and account for different scalings and correlations of the feature space.

Taigman et al. [133] applies Information-Theoretic Metric Learning (ITML) [27] for optimizing distances between face pairs for face verification. ITML enforces that similar pairs are below a certain distance while dissimilar pairs exceed a certain distance. To avoid over-fitting ITML exploits the relationship between multivariate Gaussian distributions and the set of Mahalanobis distances. The idea is to search for a solution that trades off the satisfaction of the distance constraints while being close to a distance metric prior, e.g., the Euclidean distance.

Guillaumin et al. [50] proposes Logistic Discriminant-based Metric Learning (LDML), which offers a probabilistic view on learning a Mahalanobis distance metric. The a posteriori class probabilities are treated as similarity measures, whether a pair of faces

depicts the same person or not. Thus, the a posteriori probability is modeled as shifted sigmoid function over the distance function. The shift accounts that at this specific distance threshold the probability for a pair of faces is equal for being similar or dissimilar.

Nguyen and Bai [100] propose Cosine Similarity Metric Learning (CSML) for face verification. This is especially interesting as in the real-world face recognition literature many authors propose to match their normalized face representations by cosine similarity. Intuitively, the idea is to optimize the metric such that the cosine similarity between same pairs should be high while the similarity between dissimilar pairs should be as low as possible. Further, a regularization term monitors the deviation of a prior metric. As prior the authors propose to use the whitening matrix of the truncated PCA space. In particular the whitening matrix is a diagonal form composed of the eigenvalues of the covariance matrix. Further, the cosine similarity has the appealing property that it is bounded between -1 and 1.

Köstinger et al. [75] introduces Keep It Simple and Straightforward Metric (KISSME) learning with applications to face recognition. In particular the authors address the problem that traditional metric learning approaches require complex iterative, computationally expensive optimization schemes, making them often infeasible for large-scale problems which are also common in face recognition. Instead, KISSME overcomes these limitations by introducing an efficient statistical motivated formulation that allows to learn just from equivalence constraints. Analog to the KISS principle (keep it simple and straightforward!) the method is conceptually simple and efficient per design. The main idea is to assume a Gaussian structure of the difference space as distance functions basically operate on differences between pairs of samples. For observed commonalities of face pairs showing the same or different persons the method considers two independent generation processes. The dissimilarity is defined by the plausibility of belonging either to one or the other. By log-likelihood ratio test the method interpolates between the hypotheses if a pair of faces is considered more similar or dissimilar. The resulting distance metric reflects these properties.

Similarly Ying and Li [157] focus on speed issues in training a Mahalanobis metric. The authors propose an efficient eigenvalue optimization framework for learning a metric with applications to face recognition. The method termed Distance Metric Learning with Eigenvalue Optimization (DML-eig) focuses on maximizing the minimum distance between the dissimilar pairs and keeping the similar pairs below a fixed distance threshold. The solution requires in each iteration to compute only the eigenvector associated with the largest eigenvalue. This can be done very efficiently, e.g., by power iteration.

Wang and Guibas [139] extend traditional Earth Mover's Distance (EMD) for metric learning with applications to face recognition. EMD in general is a distance metric for comparing two histograms or probability distributions. It provides a distance value as well as a flow-network indicating how the probability mass is optimally transported between the bins. In particular, the authors extend EMD to overcome the limitations of the traditional EMD that the ground distance between the bins is pre-defined. The authors argue that the learned ground distance better reflects the cross-bin relationships and yields superior results.

Another successful machine learning technique that is often applied for face recognition is feature blending. It yields reasonable performance boosts by combining complementary representations into a single classifier. This can range from different element-wise comparison functions on feature level to different face crops or face descriptions. One of the simplest methods is to learn weights on the distances obtained by the different representations, simply by a linear SVM, e.g., [20, 100, 124, 133, 156, 157]. A more sophisticated approach to learn a feature combination and weighting is by multiple kernel learning (MKL). Pinto et al. [115] propose to jointly learn weights for the convex Kernel combination and the associated SVM weights for the kernel outputs. For face recognition the kernels are features differences using different element-wise comparison functions and feature types. In particular, rather simple features as normalized pixel values, color histograms and V1-like features are extracted. In total 36 different kernels based on pixel values are extracted and 48 different kernels based on the V1-like features. The authors show that the combination yields reasonable results but if too many kernels are combined the results get saturated.

Recapitulating the different published results and methods it turns out that machine learning is an important ingredient for real-world face recognition. In particular, distance metric learning and feature blending is widely applied to reach state-of-the-art performance. For feature blending some concerns exist if more sophisticated MKL strategies are necessary as some authors observed no performance difference compared to simple linear SVM blending [85]. Other authors raise concerns that too elaborate machine learning exploits low-level regularities of the respective benchmarks. Nevertheless, many methods incorporate regularization strategies to reduce over-fitting.

2.3 Domain Specific Recognition Strategies

Inspired by the human brain that devotes specialized mechanisms for face recognition and perception [126] algorithms have been developed that focus on face specific domain knowledge to boost face recognition performance. In particular, that involves methods that take advantage of an auxiliary set of faces at test time. This set with sequestered subjects can be used for sophisticated face alignment, pose-specific matching or transferring the appearance of a particular face to another setting. Hereby, the classification problem mainly concerns pairs of faces as no stand-alone models can be learned, without requiring the auxiliary set.

Cao et al. [11] propose a pose-adaptive matching scheme to handle large pose variations. In particular, the faces are classified in one of three poses, facing left, frontal or right. This is done by measuring the face similarity between the probe faces and an auxiliary set with known pose labels. In particular, the pose label of the most alike face is assigned. For each pose combination a linear SVM trained on this specific subset of the data as pose-specific classifier.

The method of Berg and Belhumeur [6] aligns faces such that features indicating identity are preserved. Their identity-preserving alignment turns faces to frontal pose with neutral expression. In particular, the authors propose to detect fifty-five inner face parts and retrieve of an auxiliary set a number of faces with a similar part configuration. These faces with known part positions feature 40 additional outer part positions. First, the mean of the retrieved configurations is computed and used to from a generic template. Second, the template is used to align the face parts by a piecewise affine warp to the canonical part positions. Using Delaunay triangulation each triangle is mapped to its canonical pendant, using an affine transformation established by the three vertices. The authors argue that direct warping of the detected landmarks looses features that indicate identity and thus the face images would be anonymized.

Sanderson and Lovell [121] propose a normalization strategy for distances between face pairs by an auxiliary set of faces. Hereby, the distance between faces is normalized in relation to the average distance to the auxiliary set. The main idea of the normalization is that it cancels the effects of changing image conditions such as a different face pose or lighting artifacts. Intuitively, for the subjects in the auxiliary set there is no overlap in identity to the test set.

Similarly, Wolf et al. [149] exploit an auxiliary set of faces to improve matching. Therefore, an image dependent similarity measure termed One-Shot Similarity (OSS)

score is proposed. For that reason a classifier is trained to discriminate a single positive example from the an auxiliary negative set. Later, the classifier is used to determine whether other images are more likely to the probe image or to the auxiliary set. Thus, to compare two faces the procedure has to be carried out twice as it is asymmetric. The two prediction scores are averaged. For speed reasons the authors propose to use LDA as classifier as the data term for the auxiliary set can be precalculated. A simple modification of the OSS score is proposed in [150], termed Two-Shot Similarity (TSS) score. Basically the only difference to OSS is that a pair of images is used as positive examples to be discriminated from the auxiliary set. Taigman et al. [133] proposes a further extension to the OSS score. In particular, the auxiliary set of faces is split into different subsets by factors as identity, pose or lightning. These sets are used to train multiple OSS scores which are blended together by a linear SVM.

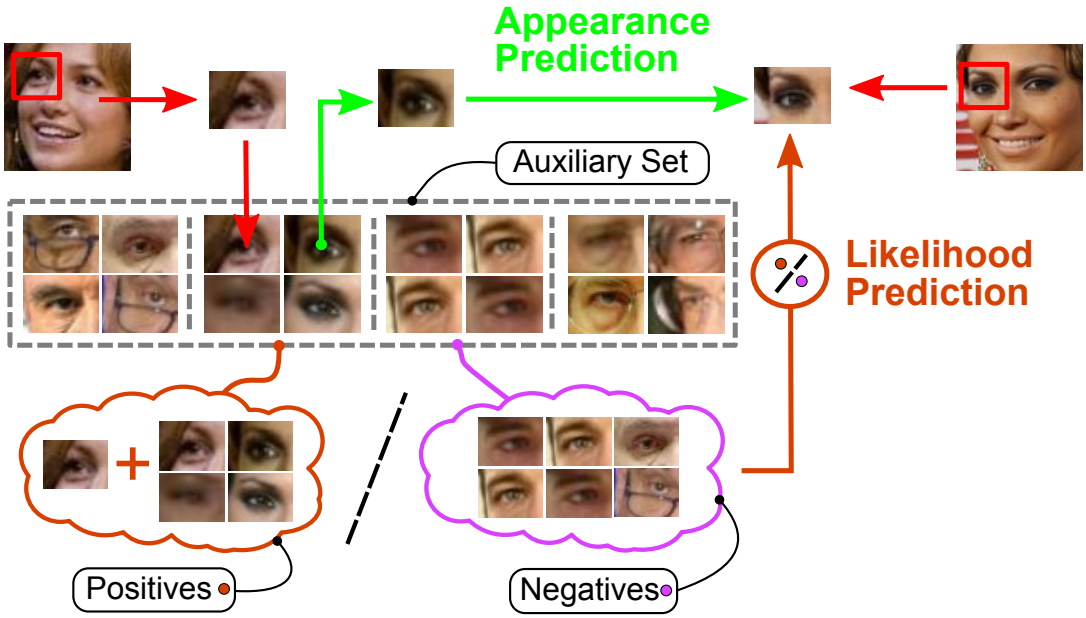


Figure 2.5: Associate predict. First, associate a given face to the auxiliary identity set. Second, predict the appearance of one face under the settings of the other face. Figure adapted from [156].

Yin et al. [156] propose for similarity estimation to transfer a pair of faces under dissimilar (e.g. pose or lightning) settings to similar settings. The main idea is to predict the appearance of one face under the settings of the other face. In particular, the authors propose two different prediction variants, namely appearance-prediction and likelihood-prediction. Therefore a probe face is first associated to one or different visually similar subjects of an auxiliary set. The auxiliary set contains for each subject

various images under different settings. In the appearance-prediction model first for each face the approximate face pose and lightning parameters are estimated. Second, the probe face is compared to the auxiliary set and a subject with similar appearance under the estimated parameters is selected. Then, an image of this subject is selected featuring the most similar parameters of the other image of the face pair. These two images are considered for matching. The likelihood prediction model selects for a given face the most similar identities. A classifier is trained to discriminate these from the rest of the auxiliary set. Finally, the learned classifier is used for similarity estimation between the face pair. See Figure 2.5 for illustration.

All algorithms described in this section exploit an auxiliary set of faces for improved matching. The authors argue that this additional information helps to hallucinate face appearance under different settings or to perform accurate face alignment that preserves features that indicate identity. Nevertheless, the auxiliary set of faces needs to be sequestered by identity from the test set, thus increasing the label effort. Further, the algorithms require more effort at test time and trade-off improved accuracy for increased computational effort.

2.4 Real-World Face Recognition Benchmarks

In the following, we give an brief overview over publicly available face recognition datasets and benchmarks. This list presents the databases which were recently of high scientific interest for real-world face recognition. On these datasets the study of face recognition is divided into two objectives: face identification (naming a face) and face verification (deciding if two face images are of the same individual). The nature of the face identification task requires a number of annotated faces per individual, not always complying with these real-world databases In contrast, face verification needs less annotations and can be evaluated more seriously also on a large scale.

2.4.1 Labeled Faces in the Wild

The Labeled Faces in the Wild (LFW) dataset [58] contains 13,233 unconstrained face images of 5,749 individuals and can be considered as the current state-of-the-art face recognition benchmark as it offerers many recently published results. For only a subset of 1,680 people there exist two or more images, for the remaining 4,069 just one. Thus, the focus lies on the face verification task. The database is considered as very challenging as it exhibits huge variations in pose, lighting, facial expression, age, gender, ethnicity

and general imaging and environmental conditions. Some illustrative examples are given in Figure 2.6. An important aspect of LFW is that per design the subjects are mutually exclusive in any split of the database. Thus, for the face verification task testing is done on individuals that have not been seen in training.

(a) Face pairs showing the **same** person (b) Face pairs showing **different** persons

Figure 2.6: Labeled Faces in the Wild (LFW) contains 13,233 faces of 5,749 individuals. For a subset of 1,680 people there exist two or more images.

The images have been gathered by harvesting faces from the web with the Viola and Jones [137] face detector. Further the images have been rescaled and cropped to a size of 250 $\times$ 250 pixels. False detections and images of unidentifiable persons have been eliminated manually. Further LFW offers two views on the data. The first view is designed for algorithmic development and tuning whereas the second view is used for performance reporting. In particular, in the second view the data is organized in stratified 10 fold cross-validation. Each fold consists of 300 same and 300 not same pairs. As performance metrics the creators of LFW suggest to use mean accuracy and the standard error of the mean. In the restricted protocol it is only allowed to consider the equivalence constraints given by the same / not same pairs. No inference on the identity of the subject, e.g., to sample more training data, is allowed. In contrast the unrestricted setting allows also to use the class labels of the samples. Intuitively the unsupervised protocol provides no labels at all.

2.4.2 Public Figures Face Database

The Public Figures Face Database (PubFig) dataset [82] is very similar to LFW. It is also an extremely challenging large-scale, real-world database, consisting of 58,797 images of 200 individuals. The images were gathered from Google images and FlickR. The face verification benchmark is a stratified 10 cross-validation folds with 1,000 intra and 1,000 extra-personal pairs each. Per fold the pairs are sampled of 14 individuals. Similar to the LFW benchmark individuals that appear in testing have not been seen before

in training. One drawback is that the authors do not provide the images but only the download links. Thus, up to date many of the links are expired and the images are unavailable.

Nevertheless, an interesting aspect of the database is that "high-level" features [82] are provided that describe the presence or absence of visual face traits. The appearance is automatically encoded in either nameable attributes such as gender, race, age, hair etc. or "similes" that relate the similarity of face regions to specific reference people. This indirect description yields nice properties such as a certain robustness to image variations compared to low-level features.

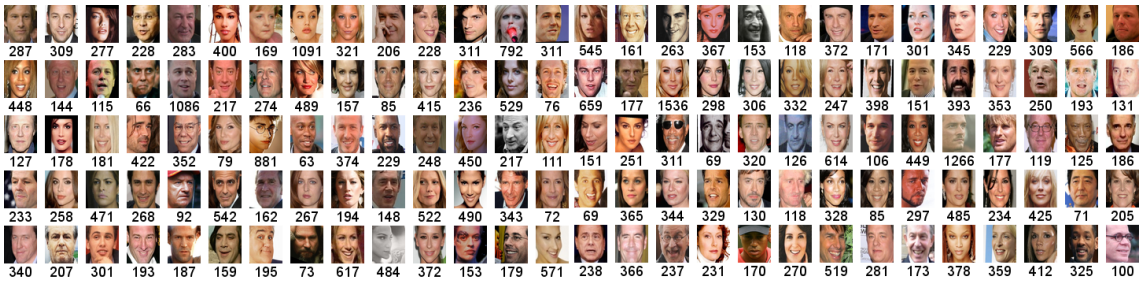


Figure 2.7: Public Figures Face Database (PubFig) samples. The evaluation set contains 42,461 images of 140 individuals. The numbers in parentheses denote the total amount of images per individual. The range is from 63 (Dave Chappelle) to 1536 (Lindsay Lohan).

2.4.3 Face Recognition Grand Challenge

The Face Recognition Grand Challenge (FRGC) [113] was an attempt of NIST for a standardized face recognition performance evaluation. The official evaluation is closed but the FRGC data is still available for research purposes. The FRGC data set contains 50,000 recordings divided into different subject sessions. Each session contains four controlled, two uncontrolled and one 3D still image of a person. The controlled images show full frontal faces either with smiling or neutral expression under two different lighting settings. The uncontrolled images are captured under real-world conditions in hallways, atria, or outside. The 3D images are taken indoors under controlled illumination conditions. One subset of FRGC is intended for validation purposes.

The 6 in the evaluation protocol defined experiments also focus on recognition between the different modalities. In particular between a controlled gallery and uncontrolled probe images and comparing two-dimensional images to a three dimensional

gallery. As performance metrics for the different experiments Receiver Operating Characteristics (ROCs) have to be reported. The FRGC distribution contains additionally to the raw recordings a performance evaluation framework and also a set of baseline algorithms.

2.5 Summary and Discussion

In this chapter an overview of recent works in the field of real-world face recognition was presented. Several methods were categorized and analyzed according to their applied feature extraction, machine learning and domain specific recognition mechanisms. Thereby the focus was on methods performing face recognition from two-dimensional still images. The advantages and disadvantages of the respective methods were discussed. Analyzing the current state-of-the-art revealed important directions for face recognition research. In particular, there is a consensus that a proper description plays a key role in obtaining good performance. Among the generic feature descriptors SIFT and LBP perform best for face recognition. Moreover, it has been observed that feature learning is beneficial for real-world face recognition. Nevertheless, a single generic or learned representation is not enough to obtain state-of-the-art performance. Hence, it has been proposed to blend multiple complementary representations and combine them into a single classifier. For feature blending concerns exist if more sophisticated MKL strategies improve over simple linear SVM blending. Another popular machine learning technique to boost face recognition accuracy is distance metric learning. In many cases considerable improvements have been observed. Some authors developed domain specific recognition strategies for faces. These algorithms exploit an auxiliary set of faces to improve matching. The respective authors argue that this additional information helps to hallucinate face appearance under different settings or to perform accurate face alignment that preserves features that indicate identity. Nevertheless the auxiliary set of faces needs to be sequestered by identity from the test set and thus increases label effort. Further, the algorithms require more effort at test time and trade-off improved accuracy for increased computational effort. Finally, in recent years real-world benchmarks showed that considerable progress has been made. Nevertheless current state-of-the-art algorithms are still far from the capabilities of the human visual system. Hence, robust face recognition is still challenging and also the obtained results are only valid for frontal faces.

3 Face Detection and Landmark Extraction

Face detection and landmark extraction are crucial preprocessing steps that heavily influence the performance of a face recognition pipeline. Facial landmarks are standard reference points, e.g. as the inner and outer corner of the eye fissure where the eyelids meet. The task of automatically localizing facial landmarks is beneficial for various reasons. For instance, an efficient estimate of the head pose can be obtained [98]. Moreover, facial landmarks can be used to align faces to each other, which is valuable in a detection, alignment and recognition pipeline; better aligned faces give better recognition results. Further, properties that have a local nature such as face attributes (e.g. *bushy eyebrows*, *skin color*, *mustache*) [82] or local descriptors [32] can be extracted. Nevertheless, facial landmark localization in unconstrained real-world scenarios is still a challenging task.

Both face detection and landmark extraction are rather data intensive and require elaborate annotations to train classifiers that perform well on real-world data. For instance Huang et al. [56] require 75,000 faces to train their detector. Including 30,000 frontal faces, 25,000 half profile faces and 20,000 full profile faces. Zhu and Ramanan [166] require for their face and landmark detector 68 annotated landmarks for frontal faces and 35 for profile faces. Thus, ultimately a key step for face recognition is also the availability of training data in large-scale.

Unfortunately in recent developments little attention has been paid about the public availability of suitable training data. Thus, in this chapter we introduce a large-scale face database tailored to real-world face detection and landmark localization that mitigates these issues. Further, we show the impact of better training data on the face detection

performance. In particular, we are even able to outperform sophisticated state-of-the-art methods on the Face Detection Dataset and Benchmark (FDDB), using a standard algorithm and standard features.

3.1 Annotated Facial Landmarks in the Wild: A Large-scale, Real-world Database for Facial Landmark Localization

Face detection and alignment is a crucial step for real-world face recognition. Especially, landmark localization for geometric face normalization or local feature extraction has shown to be very effective, clearly improving recognition results. However, no adequate datasets exist that provide a sufficient number of annotated facial landmarks. The datasets are either limited to frontal views, provide only a small number of annotated images or have been acquired under controlled conditions. Hence, we introduced a dataset overcoming these limitations: *Annotated Facial Landmarks in the Wild* (*AFLW*) [74]. *AFLW* provides a large-scale collection of images gathered from Flickr, exhibiting a large variety in face appearance (e.g., pose, expression, ethnicity, age, gender) as well as general imaging and environmental conditions. In total 24,385 faces in 21,342 real-world images are annotated with up to 21 landmarks per image. Due to the comprehensive set of annotations *AFLW* is well suited to train and test algorithms for multi-view face detection, facial landmark localization and face pose estimation.

3.1.1 Motivation

The accuracy of face recognition systems is drastically reduced in unconstrained real-world situations where imaging conditions as diversity in viewpoint, lighting, clutter or occlusion severely have to be handled. Many authors [6, 11, 105, 117, 127] observed that especially in real-world situations an accurate face detection and landmark localization step is very valuable. It is assumed that better aligned faces give better recognition results. One reason is that the description has not to cope with geometric invariance, thus enabling a more powerful description.

This, is also confirmed by experiments on the face verification benchmark of the *Labeled Faces in the Wild* (LFW) [58] dataset. The corresponding results for different feature types are illustrated in Table 3.1, where it can be seen that even a holistic face alignment step improves the recognition results. Nevertheless, many face alignment methods require rather elaborate annotations. Only some of the available face datasets provide

	Raw	HOG [24]	Felz. [39]	LBP [1]
not aligned	60,85%	63,22%	65,53%	66,13%
aligned	61,80%	65,68%	68,43%	70,13%
+	**0,95%**	**2,47%**	**2,90%**	**4,00%**

Table 3.1: **Importance of face alignment**: Face recognition accuracy on Labeled Faces in the Wild [58] for different feature types – a face alignment step clearly improves the recognition results, where the facial landmarks are automatically extracted by a Pictorial Structures [32] model.

these. However, in most cases these databases lack in several ways: First, they provide only a little number of annotated images or only sparse facial landmarks. Second, the databases focus on frontal views of faces. Finally, the images are often captured under controlled conditions (uniform background, controlled lightning etc.) and therefore do not capture real-world problems.

Hence, the main motivation for the Annotated Facial Landmarks in the Wild (*AFLW*) database [74] is the need for a multi-view, real-world face dataset for face detection and landmark localization. The images *AFLW* of were collected on *Flickr*[1] exhibiting a large variety in pose, expression, ethnicity and general imaging and environmental conditions. Further, the dataset offers uncontrolled backgrounds and many other parameters. A wide range of images related to face relevant tags were gathered and manually scanned for faces. Therefore, the collection is not restricted to frontal faces, as illustrated in Figure 3.5.

The remainder of this section is structured as follows. First, an overview of related datasets is provided and main shared features as well as the main differences are discussed in Section 3.1.2. Succeeding, in Section 3.1.3 the *AFLW* dataset is introduced and finally in Section 3.1.4 the intended usage scenarios are specified.

3.1.2 Related Datasets

The huge interest in automatic face analysis can also be seen from the numerous face datasets available publicly. However, only a subset of these datasets provides additional annotations such as facial landmarks, as summarized in Table 3.2. This number is even further reduced if multi-view faces or real-world imaging conditions are required. For instance the popular benchmark dataset LFW [58] provides a huge set of real-world im-

[1] http://www.flickr.com/

Dataset	# imgs.	# points	# ids	img. size	img. color	Ref.
Caltech 10,000 Web Faces	10,524	4	-	-	color	[2]
CMU / VASC Frontal	734	6	-	-	grayscale	[120]
CMU / VASC Profile	590	6 to 9	-	-	grayscale	[122]
IMM	240	58	40	648×480	mixed	[103]
MUG	401	80	26	896×896	color	[102]
AR Purdue	508	22	116	768×576	color	[91]
BioID	1,521	20	23	384×286	grayscale	[66]
XM2VTS	2,360	68	295	720×576	color	[94]
BUHMAP-DB	2,880	52	4	640×480	color	[3]
MUCT	3,755	76	276	480×640	color	[96]
PUT	9,971	30	100	2048×1536	color	[71]
LFW	13,233	10	5,749	250×250	color	[25, 58]
AFLW	**24,385**	**21**	-	-	**color**	

Table 3.2: **Face databases with annotated facial landmarks**.

ages that is gathered from news articles. Nevertheless, the faces are restricted to frontal poses. Other large-scale datasets such as Caltech 10,000 Web Faces [2], CAS-PEAL Face Database [43] or the CMU / VASC [122] datasets provide only a limited number of annotated landmarks. Databases with more annotated landmarks such as IMM [103] (58 landmarks, 240 images), MUG Facial Expression Database [102] (80 landmarks for a subset of 401 images) or AR Purdue [91] (22 point markup for 513 images, 130 for 897 images) provide only some hundreds of images.

In the following, datasets are discussed in more detail that are closely related to the proposed *AFLW*:

The BioID Face Database [66] consists of 1521 gray level images with a resolution of 384 × 286 pixels. The images show frontal views of 23 subjects with slightly varying poses, expressions and some ad hoc modifications, e.g., with and without glasses. The pictures were taken in an office environment with realistic background, although it stays constant for each subject. The initial eye position based markup scheme was extended by a 20 point markup scheme denoted in Figure 3.1 (a).

The XM2VTS data set [94] is intended to study the problem of multi-modal personal verification based on non-intrusive and user-friendly techniques such as speech recognition and face identification. The frontal image set, a subset of the audio-visual corpus, contains 2,360 color images at a resolution of 720×576 pixels. The images show frontal views of 295 individuals taken in 4 recording sessions. The markup scheme consists of 68 landmarks (Figure 3.1 (b)). The images were acquired with uniform background under constant imaging conditions. Subjects are not occluded and are mainly of Caucasian ethnicity.

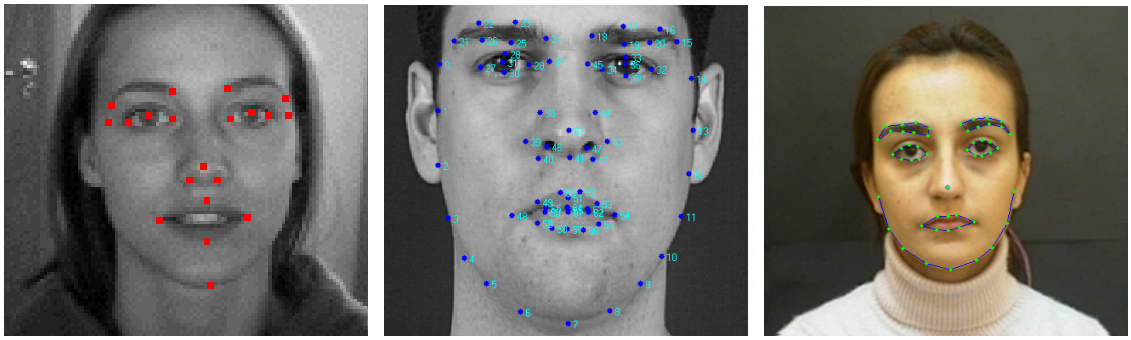
Boğaziçi University Head Motion Analysis Project Database (BUHMAP-DB) [3]. The dataset is intended to study Turkish Sign Language (TSL) and associated head/body motion and facial expressions. It involves 11 different subjects (6 female, 5 male) performing 5 repetitions of 8 different signs. In total the dataset consists of 440 videos with a resolution of 640×480. For a subset of 48 videos the dataset contains annotations of a 52 point markup (Figure 3.1 (c)). Roughly 2,880 frames are annotated. The videos are taken under controlled conditions in a darkened room with constant, uniform background. Further, no subjects are occluded, have beards, mustaches or eyeglasses. The number of subjects is limited and also the ethnicity is restricted.

Milborrow / University of Cape Town (MUCT) Face Database [96]. The MUCT dataset provides 3,755 frontal faces with neutral expression or a smile at a resolution of 640×480 pixels. The markup consists of 76 landmarks (Figure 3.1 (d)). One design goal was to provide more diversity of lighting, age and ethnicity compared to other datasets. In the image acquisition process controlled variation of lightning was introduced, up to three lightning sets per person. Further, the dataset contains a roughly equal number of males and females, with variation in age and ethnicity. Despite the introduced variation the dataset provides uniform background and no occlusions. The ethnic variation is predominately Caucasian and African.

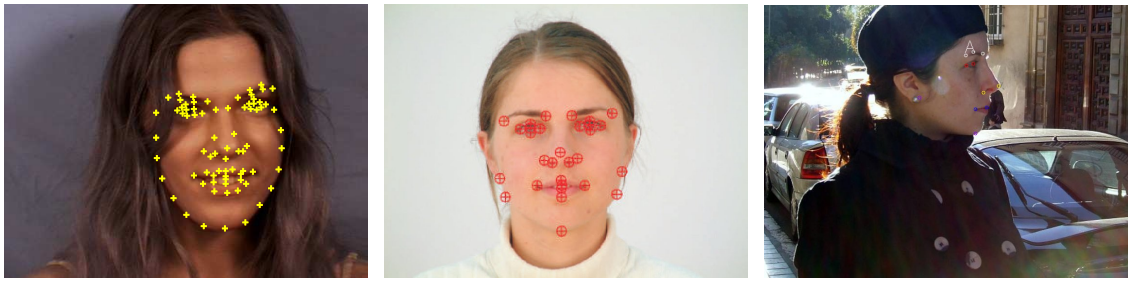
Poznań University of Technology (PUT) Face Database [71]. The dataset contains 9,971 images of 100 subjects acquired at a resolution of 2048×1536 pixels. The intended use of the dataset is the performance evaluation of face detection, facial landmark extraction and face recognition algorithms for the development of face verification methods. The authors argue that face pose is the main factor altering the face appearance in a verification system. Thus, the images were taken under controlled imaging conditions with uniform background showing various unconstrained face poses. The comprehensive set of annotations includes rectangles covering the face and also face parts. Further a set of 30 landmark points for all images (Figure 3.1 (e)). For a subset of 2,193 near-frontal faces

194 control points are annotated. Despite the large-scale nature of the dataset and the comprehensive set of provided annotations, as a drawback, the images were acquired under controlled conditions with uniform background.

If the characteristics and properties of the described datasets are recapitulated it is obvious that each collection serves several interesting properties. Nevertheless, there is no large-scale, multi-view collection of face images in the wild, annotated with facial landmarks.



(a) BioID [66] (b) XM2VTS [94] (c) BUHMAP-DB [3]



(d) MUCT [96] (e) PUT [71] (f) AFLW

Figure 3.1: **Comparison of different databases and their landmark positions**. *AFLW* provides less landmarks per image than other databases, however, it is the only database taken under real-world conditions.

3.1.3 Dataset Description

The motivation for the *AFLW* dataset[2] is the need for a large-scale, multi-view, real-world face database with annotated facial features. The images are gathered on *Flickr* using a wide range of face relevant tags (e.g., *face*, *mugshot*, *profile face*) to collect the images. Further, the initial set of downloaded images was manually scanned for faces.

[2] `http://lrs.icg.tugraz.at/research/aflw/`

Thus, the collection, which is illustrated in Figure 3.5, captures typical real-world scenarios. The key data and most important properties of the dataset are:

- The dataset contains 24,385 faces in 21,342 real-world images, with realistic background. Of these faces 59% are tagged as female, 41% are tagged as male; some images contain multiple faces. No rescaling or cropping has been performed. Most of the images are color although some of them gray-scale.
- In total *AFLW* contains 386,689 manually annotated facial landmarks of a 21 point markup. The facial landmarks are annotated upon visibility. So no annotation is present if a facial landmark, e.g., left ear lobe, is not visible.
- A wide range of natural face poses is captured The dataset is not limited to frontal or near frontal faces. To the best of our knowledge the ratio of non-frontal faces is higher than in any other dataset.
- Additional to the annotated landmarks the dataset provides face rectangles and ellipses. Further, the face ellipses support the FDDB [64] evaluation protocol.
- A rich set of tools to work with the annotations is provided, e.g., an SQL database backend that enables to import other face collections and annotation types. For popular datasets such as BioID [66], CMU / VASC profile [122] the importers are already included.

To recapitulate, *AFLW* contains more diversity and variation than any other face dataset with annotated facial landmarks. Further, due the comprehensive annotation it is well suited to train and test algorithms for

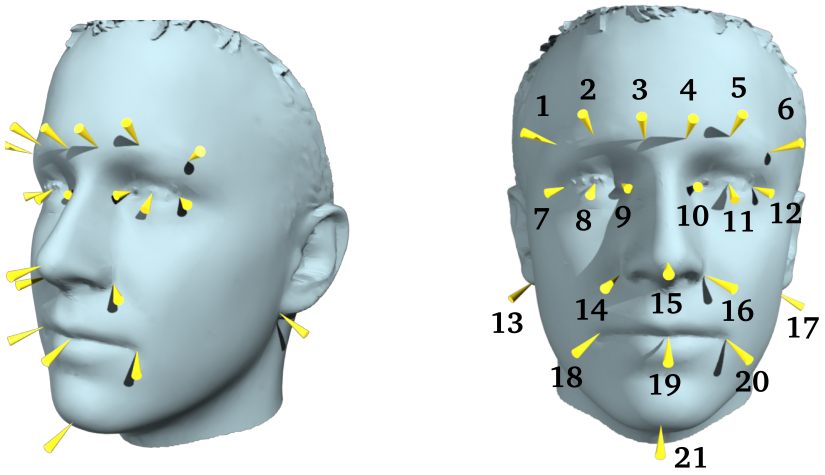
- facial landmark localization
- multi-view face detection
- coarse head pose estimation.

3.1.4 Intended Uses

The intended uses of *AFLW* are threefold. First, multi-view face detection under real-world conditions. Second, facial feature localization to support face recognition, face alignment or to train local detectors or descriptors. Third, face pose estimation to support, e.g., face tracking. An important difference to many other datasets is that *AFLW* is not only suited for testing and evaluation, but also for training.

3.1.4.1 Facial Landmark Localization

Facial landmarks are standard reference points, such as the inner and outer corner of the eye fissure where the eyelids meet. In many cases the landmarks used in computational face analysis are very similar to the anatomical soft-tissue landmarks used by physicians. The task of automatically localizing these landmarks is beneficial for various reasons. For instance, an efficient estimate of the head pose can be obtained [98] with only some landmarks. Moreover, facial landmarks can be used to align faces to each other, which is valuable in a detection, alignment and recognition pipeline; better aligned faces give better recognition results. Further, properties that have a local nature such as face attributes (e.g. *bushy eyebrows*, *skin color*, *mustache*) [82] or local descriptors [32] can be extracted. Nevertheless, facial landmark localization in unconstrained real-world scenarios is still a challenging task.



(a) Side View (b) Frontal Face

Figure 3.2: **The AFLW markup scheme**. It defines 21 facial landmarks that are located between eyebrows and chin. (b) shows the landmarks in a frontal view whereas (a) in a side view.

The landmark positions of *AFLW* are defined on a rigid 3D face model denoted in Figure 3.2. We use a markup of 21 reference landmarks mainly located in the area between eyebrows and chin. Starting at the forehead three landmarks are located at each eyebrow, on the leftmost, rightmost and medial point. Each eye area is covered by further three landmarks. The inner and outer corner of the eye fissure where the eyelids meet (endocanthion, exocanthion) and the pupil. On the external nose the left and right point of attachment of the nose cavity with the face (nasal alar crest) and the tip of the

nose (pronasale) are specified. On the external ear the lowest point of attachment to the head (otobasion inferius) is marked. On the mouth and lips the landmarks are placed on the left and right intersection point of the lips (cheilion) and the mouth center as medial point. Finally, on the chin the lowest point on the lower border (gnathion) is selected. In the annotation process landmarks are marked upon visibility. So if a landmark is not visible it is simply not annotated. In total 386,689 landmarks have been annotated so far. For individual landmarks the number of annotations ranges from 9,892 (left ear) to 24,378 (nose center). Table 3.3 contains detailed statistics.

3.1.4.2 Face Pose Estimation

Head pose estimation in images captured under uncontrolled conditions in natural environments is still a challenging task. Thus, *AFLW* comes with approximate pose information for each face, derived from the annotated facial landmarks. To this end, a mean 3D model [130] of the frontal part of a head (shown in Figure 3.2) is fitted to the annotated points in the image. The pose parameters are adjusted, to minimize the distance between the projections of the corresponding points on the 3D model and the actual landmark locations in the image in a least squares manner by POSIT [29]. The resulting pose is stored in terms of roll, pitch and yaw angles as depicted in Figure 3.3.

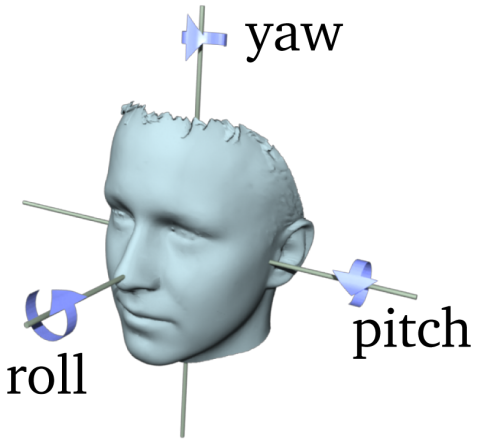


Figure 3.3: **Head pose**. It is described in form of the three rotation angles yaw, pitch and roll.

Further, the extracted pose estimate can be used to retrieve images from a limited range of poses only. This can be used to train sets of individual, pose dependent face detectors. Another possible application is the analysis of person independent relations between a given image representation and controlled variations in the pose.

ID	Landmark Description	Count
1	Left Brow Left Corner	15,346
2	Left Brow Center	19,310
3	Left Brow Right Corner	20,350
4	Right Brow Left Corner	20,583
5	Right Brow Center	19,493
6	Right Brow Right Corner	15,606
7	Left Eye Left Corner	18,152
8	Left Eye Center	19,984
9	Left Eye Right Corner	17,018
10	Right Eye Left Corner	16,689
11	Right Eye Center	20,460
12	Right Eye Right Corner	18,343
13	Left Ear	9,892
14	Nose Left	16,923
15	Nose Center	24,378
16	Nose Right	17,559
17	Right Ear	10,601
18	Mouth Left Corner	19,169
19	Mouth Center	23,802
20	Mouth Right Corner	19,882
21	Chin Center	23,149
		386,689

Table 3.3: **Overview of landmark annotations in AFLW**. The number of individual annotations ranges from 9,892 (left ear) to 24,378 (nose center).

3.1.4.3 Multi-View Face Detection

While frontal face detection is commonly considered as solved multi-view face detection in uncontrolled environments remains still an unsolved challenge. A main reason is that multi-view face detection requires a large number of annotated faces in training.

In particular, pose-specific detectors need to capture the full range of face poses. For instance the tree-structured detector of Huang et al. [56] partitions the face pose with a coarse-to-fine strategy. The detector is composed of 204 pose split nodes in 16 layers and trained with roughly 75,000 faces. However, the training dataset is not publicly available. Thus, *AFLW* features a large number of annotated faces covering the full range of face poses and additionally provides an estimate of the face pose.

To test multi-view face detection algorithms *AFLW* provides face ellipses consistent with the Face Detection Dataset and Benchmark (FDDB) [64] evaluation protocol. All faces are annotated by an ellipse outlining the 3D ellipsoid capturing the front of the head. This gives a closer boundary of the region of interest compared to face rectangles and in-plane rotation information. The ellipse annotations are automatically generated from the facial landmark annotations by fitting the mean 3D face model. The result of this process is demonstrated in Figure 3.4.

Figure 3.4: **Face ellipses** automatically created from the annotated facial landmarks, following the specification in the FDDB [64] evaluation framework.

3.2 The Impact of better Training Data for Face Detection

In this section we investigate the impact of suitable large-scale training data on the real-world face detection performance. Especially in unconstrained situations where variations in face pose or bad imaging conditions have to be handled face detection remains challenging. These problems are covered by recent benchmarks such as *Face Detection Dataset and Benchmark* (FDDB) [64], which reveals that established methods, e.g., Viola and Jones [137] suffer a distinct drop in performance compared to previous

Figure 3.5: **Impressions of the Annotated Facial Landmarks in the Wild (AFLW) database**. *AFLW* provides variation in pose, ethnicity, realistic background and natural uncontrolled imaging conditions.

evaluations. More effective approaches exist, but are closed source and not publicly available. Also the reimplementation is practically impossible as these algorithms heavily rely on statistical machine learning using massive amounts of data, typically unavailable to the public [56, 57, 132]. In that context the question arises if the performance gain is attributed to the improved algorithms or proprietary datasets. In particular, we investigate if a better face detection performance can be obtained by simply increasing the amount of data, with a standard algorithm and standard features. Moreover, we want to know how close we can get to state-of-the-art methods.

3.2.1 Experiments and Implementation

To show the impact of suitable large-scale training data on the real-world face detection performance, we propose to use an off-the-shelf implementation of the Viola and Jones [137] detector using multi-scale block LBPs [87]. Preliminary experiments showed that these provide a similar performance compared to Haar features. However these are more efficient in training.

In particular, we want to show that by using only better training data allows us to reach or even improve over methods that have a higher model complexity and higher runtime requirements. Further these methods require in most cases also large amounts of data in training. We gather the face crops of the *AFLW* dataset. As *AFLW* includes the coarse face pose we are able to retrieve up to 28k frontal faces by limiting the yaw angle between $\pm\frac{\pi}{6}$ and mirroring them. For each face we crop a square region between forehead and chin. The non-face patches are obtained by randomly sampling at multiple scales of the PASCAL VOC 2007 dataset [33], excluding the persons subset. Testing several patch size revealed that a standard patch size of 24×24 delivers the best results.

As first benchmark we evaluate our face detector on FDDB. We intend to compare it to state-of-the-art approaches and also investigate the influence of the amount of training data on the face detection performance. FDDB is designed for face detection in real-world scenarios. It features 2,845 images with a total of 5,171 faces captured under a wide range of imaging and environmental conditions including occlusions, non-frontal face poses, and low resolution and out-of-focus faces. Groundtruth is specified in form of face regions as ellipses. The authors argue that the ellipses capture better the shape of the human head compared to rectangles, also for profile views. As evaluation metrics two scores are proposed. For the discrete score it is proposed to use the PASCAL VOC overlap criterion between detected faces and groundtruth. In particular, for a true positive the overlap of intersected areas to joined areas has to be more than 50%. Then,

(a) (b)

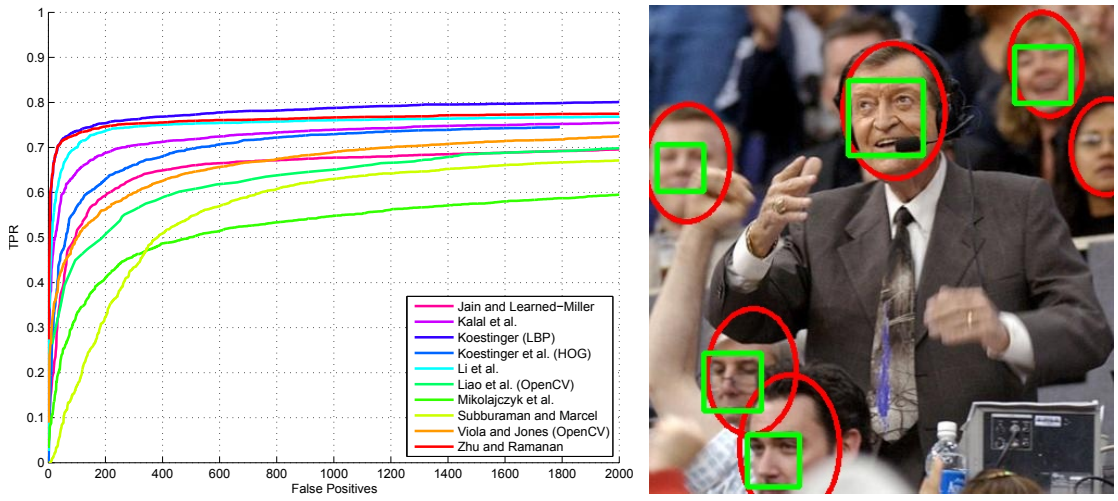
(c) (d)

Figure 3.6: **Face Detection Dataset and Benchmark (FDDB):** The benchmark comprises faces captured under a wide range of imaging and environmental conditions including occlusions, non-frontal face poses, and low resolution and out-of-focus faces. The ground truth annotation is denoted by the red ellipses.

for a detection the detector score is reported. The overlap criterion is a rather strict measure. The continuous score reports directly the overlap as score. Further, the cumulative performance is reported as the average ROC curve over a 10-fold cross-validation. Samples of the FDDB benchmark with according groundtruth ellipses are shown in Figure 3.6. The quality of the annotation can be considered as good, however in some images annotations are missing.

For our detector a perfect match to a ground truth ellipse is not possible as it outputs upright square candidate rectangles. For that reason we focus on the discrete score and scale the candidate rectangles by a fixed factor to match the elongated shape of the face ellipses.

In Figure 3.7 we report the performance of our final detector on the challenging



(a) (b)

Figure 3.7: **Face detection results on the FDDB benchmark**. In (a) we report ROC curves for [65, 68, 86, 95, 131, 137, 166] and our method. In (b) we provide an illustrative detection example. The red ellipses denote the FDDB ground truth, whereas the green rectangles are the output of our detector.

	25	50	100	200	500
Proposed method	**68.36%**	**71.84%**	**73.70%**	**75.36%**	**77.22%**
Zhu and Ramanan [166]	68.27%	71.61%	72.89%	74.65%	75.90%
Li et al. [86]	58.29%	65.33%	69.97%	73.74%	75.27%
Kalal et al. [69]	50.18%	59.74%	64.63%	68.81%	71.82%
Viola and Jones [137]	36.20%	43.40%	49.89%	56.20%	64.44%
Köstinger et al. [77]	35.47%	47.22%	56.80%	62.95%	69.70%
Liao et al. [87]	29.76%	37.16%	45.29%	50.73%	60.66%
Jain and Learned-Miller [65]	23.57%	38.84%	51.46%	59.31%	65.98%

Table 3.4: **Face detection results on the FDDB benchmark**. For the respective methods we compare the true positive rate versus the total false positives.

FDDB benchmark and compare it to state-of-the-art approaches. Despite the simplicity of our detector it is able to improve considerably over several other approaches. It improves clearly over the standard boosted classifier cascade of Viola and Jones [137], implemented in OpenCV, both for Haar and LBP features. Further the method outperforms the recent work of Jain and Learned-Miller [65], which adapts at test time a

pre-trained boosted classifier-cascade. The main idea of this work is to reclassifying hard examples near the decision boundary by considering other in the image present detections. Also the work of Kalal et al. [68] is outperformed which shows that in iterative classifier training bootstrapping has a significant impact on the final face detection performance. In particular they focus on sampling strategies for mining meaningful negative samples on large-scale image collections. We also improve over the work of Li et al. [86] which uses a boosted classifier cascade and SURF features for face detection. Finally we are comparable to the deformable part-based multi-view model of Zhu and Ramanan [166] which uses as shared pool of parts to detect up to 68 facial landmarks. The authors argue that the flexible local parts enable to effectively capture the elastic deformation of faces. Table 3.4 shows a detailed numerical comparison of the discussed methods with focus on the performance with a low number of false positives.

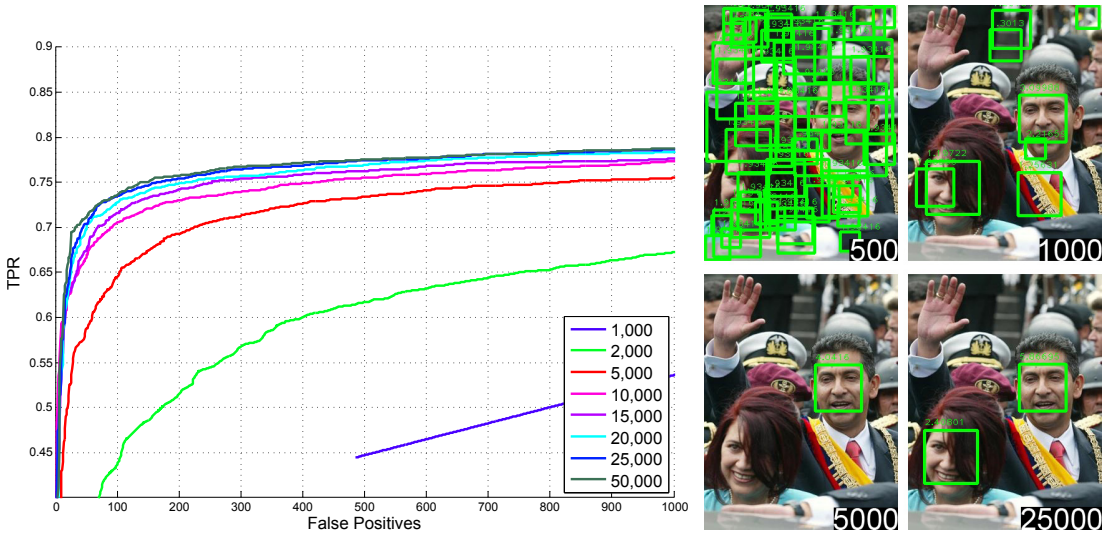
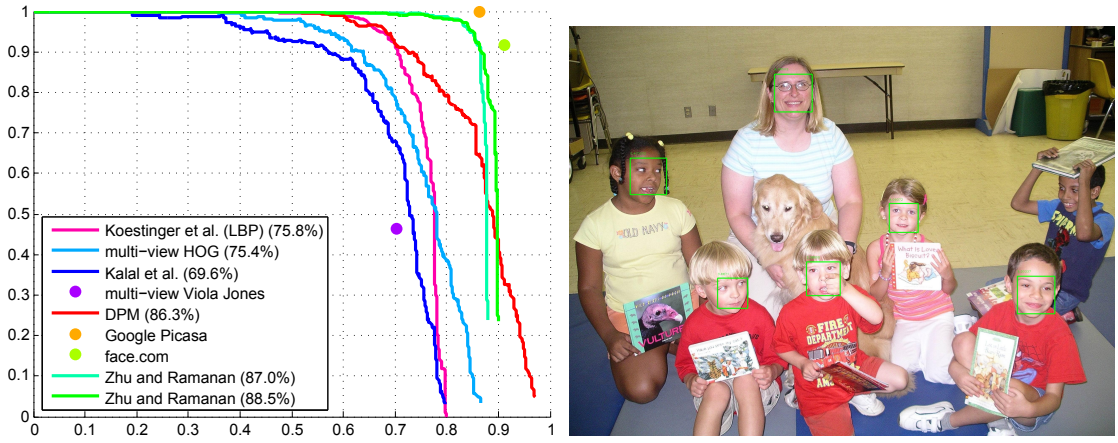


Figure 3.8: **Face Detection Dataset and Benchmark (FDDB):** (a) Influence of varying the amount of training data for our LBP based face detector. The legend displays the number of samples used for training. (b) Different detectors illustrated trained with 500 to 25,000 samples.

In Figure 3.8 (a) we compare the face detection performance on FDDB in relation to the size of the training database. Therefore we trained several detectors using the same parameters and varied the amount of training data. We started with only 50 samples and increased the amount of samples until we exhausted the training database. Starting from 1,000 samples the face detector starts to obtain a reasonable performance. At 2,000 samples a comparable performance to the OpenCV / Viola and Jones face detector is

reached. Starting from 15,000 samples the performance begins to saturate. Only minor improvements are obtained afterwards. Figure 3.8 (b) shows an illustrative example how the detector evolves by increasing the amount of training data. The in (b) inpainted numbers show the size of the training set. Initially, many false positives are randomly distributed over the images. As data grows the detector improves and a preference for faces is clearly visible. In the two bottom images it can be seen that in an intermediate stage the false positives are eliminated. However also one true positive is discarded. The true positive is recovered as data grows further.



(a) (b)

Figure 3.9: **Face detection results on the Annotated Faces in the Wild (AFW) dataset** [166]. In (a) we report Precision / Recall curves for [38, 68, 137, 166], two commercial systems and our method. In (b) we provide an exemplary detection example missing one profile face detection.

Further, we evaluate our proposed method on the Annotated Faces in the Wild (AFW) dataset [166]. AFW offers 205 high-resolution images collected on FlickR containing in total 468 faces. Similar to FDDB the images show faces with large variations in appearance, viewpoint and also difficult backgrounds. The dataset contains also a large portion of non-frontal faces. The ground truth is given in form of rectangles that outline the facial landmarks as used in [166]. As evaluation metric it is proposed to plot Precision-Recall curves based on the PASCAL VOC overlap criterion requiring 50% overlap.

In Figure 3.9 (a) we compare the face detection performance of our method with related works judged by Precision-Recall curves. The Precision-Recall curves for the related works are provided by the authors of [166]. The number in parentheses denote the average precision as defined in the VOC protocol. Therefore, a version of the

Precision-Recall curve of the respective method is calculated with the precision monotonically decreasing. Therefore, at a certain recall value the precision is set to the maximum precision of the same or any higher recall values. Then, the average precision is the area under the curve by numerical integration over all unique recall values.

Interestingly, the proposed frontal face detector is able to outperform some of the related works, also on this dataset with strong focus on multi-view face detection. This is despite the fact that our detector, in contrast to the other detectors, is, a rigid single-view detector. The curves reported for the approaches of Viola and Jones [138] (OpenCV), Kalal et al. [68] and the multi-view HOG detector integrate at least two-views. Of course models trained for multi-view face detection as the deformable part model (DPM) of [38] or the approach of Zhu and Ramanan offer a better performance on this dataset. Nevertheless, our detector is over many levels of recall comparable to these approaches.

If we recapitulate the main results on FDDB and AFW it is obvious that suitable large-scale training data has a significant impact on the face detection performance. Our rigid single-view face detector is comparable on FDDB to state-of-the-art approaches using more complex models or more sophisticated training algorithms. Thus, we hypothesize that at least for face detection existing detectors are not limited by the model complexity but by the available training data. Further, at least for frontal face detection using flexible models with higher runtime requirements compared to rigid detectors is questionable.

3.3 Conclusion

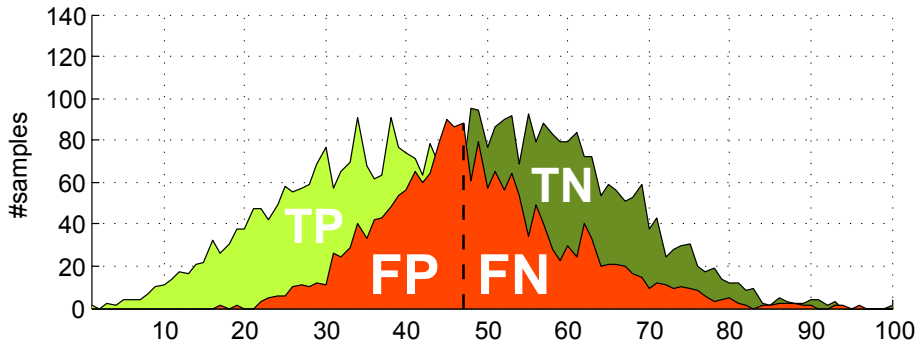
In this chapter we focused on large-scale training data for face detection and landmark extraction. Face detection and landmark extraction are crucial preprocessing steps that heavily influence the final face recognition performance. Obviously without detected faces recognition becomes impractical or even impossible. Detected landmarks enable to align faces to a canonical pose or to extract local features. Both face detection and landmark extraction are rather data intensive and require elaborate annotations to train classifiers that perform well on real-world data. Therefore, we introduced the *Annotated Facial Landmarks in the Wild (*AFLW*)* database. *AFLW* provides a large-scale, real-world collection of face images, gathered from FlickR. Compared to other datasets *AFLW* is the only one that is publicly available and well suited to train and test algorithms for multi-view face detection, facial landmark localization and face pose estimation. Once having introduced *AFLW* we investigated the impact of suitable large-scale training data

on the real-world face detection performance. Especially in unconstrained situations where variations in face pose or bad imaging conditions have to be handled face detection remains challenging. If we recapitulate the main results gained on two publicly available benchmarks it becomes obvious that suitable large-scale training data has a significant impact on the face detection performance. For frontal face detection our rigid single-view face detector is comparable to state-of-the-art approaches that use more complex models or more sophisticated training algorithms. Thus, we hypothesize that at least for face detection existing detectors are not limited by the model complexity but by the available training data. Further, at least for frontal face detection using flexible models with higher runtime requirements compared to rigid detectors is questionable.

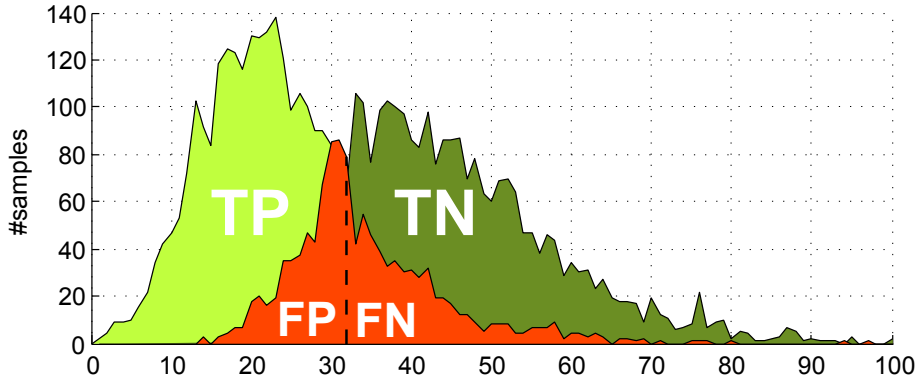
4

Efficient Large Scale Metric Learning and Retrieval for Face Recognition

The review in Chapter 2 showed that especially Mahalanobis metric learning methods recently demonstrated competitive results for real-world face recognition. However, as data grows several new challenges are posed to existing algorithms in terms of scalability and the and the required degree of supervision. These algorithms have two main drawbacks. First, learning metrics requires often to solve complex and thus computationally very expensive optimization problems. Second, as the evaluation time linearly scales with the size of the data k-NN search becomes cumbersome for large-scale problems or real-time applications with limited time budget. Further, considering the constantly growing amount of data it is often infeasible to specify fully supervised labels for all data points. Instead, it is easier to specify labels in form of pairwise equivalence constraints. Therefore, in this chapter we introduce a simple though effective strategy to learn a distance metric from equivalence constraints, based on a statistical inference perspective. In contrast to existing methods the method does not rely on complex optimization problems requiring computationally expensive iterations. Further, we propose a metric-based hashing strategy to speed k-NN search for large-scale problems.



(a) Distribution of distances before . . .



(b) . . . and after applying the proposed method.

Figure 4.1: **Face verification on LFW [58]:** The challenging task shows the benefit of metric learning. The proposed method significantly increases the TPR at EER from 67.4% (a) to 80.5% (b). Training takes only **0.05** seconds and is thus orders of magnitudes faster than related methods.

4.1 Introduction

Learning distance or similarity metrics is an emerging field in machine learning, with various applications in computer vision, not limited to face recognition. It can significantly improve results for tracking [158], image retrieval [55], clustering [155], alignment [101] or person re-identification [30]. The goal of metric learning algorithms is to take advantage of prior information in form of labels over simpler though more general similarity measures. For instance, Figure 4.1 illustrates the benefit of metric learning for face verification. The True Positive Rate (TPR) at Equal Error Rate (EER) is significantly increased.

A particular class of distance functions that exhibits good generalization performance for many machine learning problems is Mahalanobis metric learning. The goal is to find a global, linear transformation of the feature space such that relevant dimensions

are emphasized while irrelevant ones are discarded. As there exists a bijection between the set of Mahalanobis metrics and the set of multivariate Gaussians one can think of it in terms of the corresponding covariance matrix.

Machine learning algorithms that learn a Mahalanobis metric have recently attracted a lot of interest in computer vision. These include Large Margin Nearest Neighbor Learning (LMNN) [143, 145], Information Theoretic Metric Learning (ITML) [27] and Logistic Discriminant Metric Learning (LDML) [50], which can be considered as state-of-the-art. LMNN [143, 145] aims at improving k-nn classification. It establishes for each training sample a local perimeter. The perimeter surrounds the k-NNs with similar label (target neighbors), plus a safety margin. To reduce the amount of instances with dissimilar labels that invade the perimeter (impostors) the metric is iteratively adapted. This is done by strengthening the correlation to target neighbors while weakening it to impostors. Conceptually sound, LMNN is sometimes prone to over-fitting due to the lack of regularization. Davis et al. [27] avoid over-fitting by explicitly integrating a regularization step. Their formulation trades off between satisfying the given constraints on the distance function while minimizing the differential entropy to the initial prior distance metric distribution. Guillaumin et al. [50] introduce a probabilistic view on learning a Mahalanobis metric where the a posteriori class probabilities are treated as (dis)similarity measures. Thus, they propose to iteratively adapt the Mahalanobis metric to maximize the log-likelihood. The a posteriori probability is modeled by a sigmoid function that reflects that samples share labels if their distance is below a certain threshold. In principle, these methods generalize well to unseen data. They focus on robust loss functions and regularize solutions to avoid over-fitting.

Considering the ever growing amount of data, learning a Mahalanobis metric on a large scale dataset raises further issues on scalability and the required degree of supervision. Often it is infeasible to specify fully supervised labels for all data points. Instead, it is easier to specify labels pairwise in form of equivalence constraints. In particular if a pair of samples shares the same class label or not. In some cases it is even possible to obtain this form of weak supervision automatically, e.g., by tracking an object. Hence, to capitalize on large scale applications as real-world face recognition one faces the additional challenges of scalability and the ability to deal with equivalence constraints.

One further important aspect that is often neglected is the computational burden at test time as k-NN search in high-dimensional spaces is cumbersome. For real-time applications with limited time budget this is even more critical; especially on larger datasets with tens of thousands of samples that have to be explored.

To speed up nearest neighbor search a successful approach is to focus on sparsity in the variables and perform an efficient low dimensional embedding. For instance, one can accelerate nearest neighbor search by performing a binary Hamming embedding. This can be done by applying hashing functions directly [63] or on kernelized data [81]. In particular, hyperplanes or hyperspheres are used to partition the space. Data independent variants as [15, 44] ignore the structure of the data. Data dependent methods [51, 146] consider the structure of the data, however, these mostly build on an isotropic cluster assumption and thus do not exploit the general structure of the data.

To meet these requirements, it is proposed to learn an effective metric just based on equivalence constraints. Equivalence constraints are considered as natural inputs to distance metric learning algorithms as similarity functions basically establish a relation between pairs of points. The method is motivated by a statistical inference perspective based on a likelihood-ratio test. Results show that the resulting metric is not prone to over-fitting and very efficient to obtain. Compared to other approaches it does not rely on a tedious iterative optimization procedure. Therefore, the method is scalable to large datasets, as it just involves computation of two covariance matrices. As analog to the KISS principle (*keep it simple and straightforward!*) the method is easy and efficient per design therefore it is termed *KISS metric*. To speed up evaluation it is adapted for two different hashing strategies. The proposed method enables to drastically reduce the computational effort during training and evaluation while maintaining accuracy. The method is evaluated on various different benchmarks with focus on face recognition where it matches or even outperforms state-of-the-art Mahalanobis metric learning approaches, while being orders of magnitudes faster in training. In particular, results are provided for the unconstrained face recognition benchmarks LFW [58] and PubFig [82]. Further, we study the task of person re-identification across spatially disjoint cameras (VIPeR [47]) and the comparison of before never seen object instances on ToyCars [105]. On VIPeR and the ToyCars dataset the method even improves over the domain specific state-of-the-art. Further, for LFW it obtains the best reported results for standard SIFT features.

The rest of this chapter is organized as follows. Next in Section 4.2 we discuss the related work on Mahalanobis metric learning that motivates our approach. Succeeding, in Section 4.3 we introduce KISS metric learning. To speed up evaluation we introduce introduce our metric-based hashing strategy in Section 4.4. Extensive experiments and evaluations on performance and scalability are conducted in Section 4.5. Finally, Section 4.6 summarizes and provides concluding remarks.

4.2 Mahalanobis Metric Learning

Learning a distance or similarity metric based on the class of Mahalanobis distance functions has gained considerable interest in computer vision. The classical Mahalanobis distance [90] measures the squared distance between two data points $\mathbf{x}_i$, $\mathbf{x}_j \in \mathbb{R}^d$ in a multivariate Gaussian distribution:

$$d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T \Sigma^{-1} (\mathbf{x}_i - \mathbf{x}_j) \tag{4.1}$$

where

$$\mathbf{\Sigma} = \sum_{\forall (i,j)} (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_j - \boldsymbol{\mu})^T \tag{4.2}$$

is a $d \times d$ covariance matrix formed of all sample tuples (i, j) and $\boldsymbol{\mu}$ is the mean vector of the data. If the covariance matrix $\mathbf{\Sigma}$ matches the identity the Mahalanobis distance reduces to the isotropic Euclidean distance. In this case all feature dimensions are weighted equally. If the covariance matrix is a diagonal matrix the distance accounts for different scaled dimensions of the feature space and thus is a weighted Euclidean distance. A general form of the covariance matrix measures different scalings and correlations of the feature space. One can think of it by means of decomposing the covariance matrix Σ in terms of its eigenvectors and eigenvalues. The projection on the eigenvectors rotates the feature space and scales with the associated eigenvalues.

In contrast to the classical Mahalanobis distance, the goal of Mahalanobis metric learning is to exploit prior information such as labels to learn a similarity measure that is better suited for a particular task such as k-NN classification, clustering or image retrieval. In particular, the Mahalanobis distance is parameterized by a $d \times d$ matrix $\mathbf{M}$ that measures the squared distance between two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ as:

$$d^2_{\mathbf{M}}\left(\mathbf{x}_i, \mathbf{x}_j\right) = \left(\mathbf{x}_i - \mathbf{x}_j\right)^\top \mathbf{M} \left(\mathbf{x}_i - \mathbf{x}_j\right), \tag{4.3}$$

where $\mathbf{M} \succeq 0$ is positive semidefinite and $\mathbf{x}_i$,$\mathbf{x}_j \in \mathbb{R}^d$ is a pair of samples (i, j). The positive semi definiteness is required that $\mathbf{M}$ induces a valid pseudo metric. Otherwise, negative eigenvalues could result in negative distances. Ideally, the matrix $\mathbf{M}$ allows for modeling task-specific important scalings and correlations of the feature space.

For the following discussion let y_{ij} be a similarity label with the property $y_{ij} = 1$ for similar pairs, i.e., if the samples share the same class label ($y_i = y_j$) and $y_{ij} = 0$ otherwise.

To motivate the proposed method, in the following an overview of the state-of-the-art in learning a Mahalanobis metric is given. In particular, we examine LMNN [143, 145], ITML [27] and LDML [50] as these have recently shown good results and therefore attracted a lot of interest in the computer vision community.

4.2.1 Large Margin Nearest Neighbor Metric

The approach of Weinberger et al. [143, 145] aims at improving k-NN classification by exploiting the local structure of the data. For each sample a local perimeter surrounding the k nearest neighbors sharing the same label (target neighbors) is established. Samples having a different label that invade this perimeter (impostors) are penalized. This is realized via the following objective function:

$$\epsilon\left(\mathbf{M}\right) = \sum_{j \rightsquigarrow i} \left[d^2_{\mathbf{M}}\left(\mathbf{x}_i, \mathbf{x}_j\right) + \mu \sum_{l} \left(1 - y_{il}\right) \xi_{ijl}\left(\mathbf{M}\right) \right]. \tag{4.4}$$

The first term minimizes the distance between target neighbors $\mathbf{x}_i$,$\mathbf{x}_j$, indicated by $j \rightsquigarrow i$. The second term denotes the amount by which impostors invade the perimeter of i and j. An impostor l is a differently labeled input ($y_{il} = 0$) that has a positive slack variable:

$$\xi_{ijl}\left(\mathbf{M}\right) = 1 + d^2_{\mathbf{M}}\left(\mathbf{x}_i, \mathbf{x}_j\right) - d^2_{\mathbf{M}}\left(\mathbf{x}_i, \mathbf{x}_l\right). \tag{4.5}$$

To estimate $\mathbf{M}$, gradient descent is performed along the gradient defined by the triplets (i, j, l) having positive slack:

$$\frac{\partial \epsilon\left(\mathbf{M}^t\right)}{\partial \mathbf{M}^t} = \sum_{j \rightsquigarrow i} \mathbf{C}_{ij} + \mu \sum_{(i,j,l)} \left(\mathbf{C}_{ij} - \mathbf{C}_{il}\right), \tag{4.6}$$

where $\mathbf{C}_{ij} = \left(\mathbf{x}_i - \mathbf{x}_j\right)\left(\mathbf{x}_i - \mathbf{x}_j\right)^\top$ denotes the outer product of pairwise differences. Conceptually, for active triplets this formulation strengthens the correlation to target neighbors while weakening it to impostors. Further, LMNN it is not directly applicable to learn from equivalence constraints as it requires class labels to sample labeled triplets.

4.2.2 Information Theoretic Metric Learning

Davis et al. [27] exploit the relationship between multivariate Gaussian distributions and the set of Mahalanobis distances. The idea is to search for a solution that trades off the satisfaction of constraints while being close to a distance metric prior $\mathbf{M}_0$, e.g., the

identity matrix for the Euclidean distance. The closeness of the solution to the prior is measured by the Kullback-Leibler divergence of the corresponding distributions. The prior can be considered as a regularization term to avoid over-fitting. The constraints enforce that similar pairs are below a certain distance $d^2_{\mathbf{M}}\left(\mathbf{x}_i, \mathbf{x}_j\right) \leq u$ while dissimilar pairs exceed a certain distance $d^2_{\mathbf{M}}\left(\mathbf{x}_i, \mathbf{x}_j\right) \geq l$. The optimization builds on Bregman projections [9], which project the current solution onto a single constraint via the update rule:

$$\mathbf{M}_{t+1} = \mathbf{M}_t + \beta \mathbf{M}_t \mathbf{C}_{ij} \mathbf{M}_t \ . \tag{4.7}$$

The parameter β involves the label of the pair of samples and the step size. It is positive for similar pairs and negative for dissimilar pairs. Thus, for similar pairs the optimization is performed in direction of $\mathbf{C}_{ij}$ while for dissimilar pairs in the negative direction.

4.2.3 Linear Discriminant Metric Learning

Guillaumin et al. [50] offer a probabilistic view on learning a Mahalanobis distance metric. The a posteriori class probabilities are treated as (dis)similarity measures, whether a pair of images depicts the same object. For a given pair (i, j) the a posteriori probability is modeled as

$$p_{ij} = p\left(y_{ij} = 1 | \mathbf{x}_i, \mathbf{x}_j; \mathbf{M}, b\right) = \sigma\left(b - d^2_{\mathbf{M}}\left(\mathbf{x}_i, \mathbf{x}_j\right)\right) \ , \tag{4.8}$$

where $\sigma(z) = \left(1 + \exp(-z)\right)^{-1}$ is a sigmoid function and b is a bias term. Thus, to estimate $\mathbf{M}$, the Mahalanobis metric is iteratively adapted to maximize the log-likelihood:

$$L(\mathbf{M}) = \sum_{ij} y_{ij} \ln\left(p_{ij}\right) + \left(1 - y_{ij}\right) \ln\left(1 - p_{ij}\right) . \tag{4.9}$$

The maximization by gradient ascent is obtained in direction of $\mathbf{C}_{ij}$ for similar pairs and in the negative direction for dissimilar pairs:

$$\frac{\partial L(\mathbf{M})}{\partial \mathbf{M}} = \sum_{ij} \left(y_{ij} - p_{ij}\right) \mathbf{C}_{ij} \ . \tag{4.10}$$

The contribution of each pair on the gradient is controlled over the probability.

If we recapitulate the properties and characteristics of the described metric learning approaches two common features are observed. First, all methods rely on an iterative optimization scheme which can be computationally expensive for large scale datasets

common in face recognition. Second, if the update rules of the different methods are compared, given in Eqs. (4.6), (4.7) and (4.10), it can be seen that the optimization schemes operate in the space of pairwise differences. In particular, the optimization is performed in direction of $\mathbf{C}_{ij}$ for similar pairs and in the negative direction of $\mathbf{C}_{ij}$ for dissimilar pairs. In the following, a non-iterative formulation is introduced, which builds on a statistical inference perspective of the space of pairwise differences. This allows for facing the challenges of scalability and the ability to learn from equivalence constraints. The parameter-free approach is very efficient in training, enabling to exploit the constantly growing amount of data also for learning.

4.3 KISS Metric Learning

For the following discussion let $\mathbf{x}_i, \mathbf{x}_j \in \mathbb{R}^d$ be a pair of samples and $y_i, y_j \in \{1, 2, \ldots, c\}$ the according labels. Further let $\mathcal{S} = \{(i,j) \,|y_i = y_j\}$ be a a set of similar pairs and $\mathcal{D} = \{(i,j) \,|y_i \neq y_j\}$ a set of dissimilar pairs. The goal is to decide whether a pair (i,j) is similar or not. The proposed method considers two independent generation processes for observed commonalities of similar and dissimilar pairs. The similarity is defined by the plausibility of belonging either to one or the other. From a statistical inference point of view the optimal statistical decision whether a pair (i,j) is dissimilar or not can be obtained by a log-likelihood ratio test. Thus, we test the hypothesis H_0 that a pair is dissimilar versus the alternative H_1:

$$\delta\left(\mathbf{x}_i, \mathbf{x}_j\right) = \log\left(\frac{p\left(\mathbf{x}_i, \mathbf{x}_j | H_0\right)}{p\left(\mathbf{x}_i, \mathbf{x}_j | H_1\right)}\right) . \tag{4.11}$$

A high value of $\delta(\mathbf{x}_i, \mathbf{x}_j)$ means that hypothesis H_0 is validated and the pair of samples is considered as dissimilar. In contrast, a low value means that H_0 is rejected and the pair of samples is considered as similar.

Next, the metric learning problem is casted into the space of pairwise differences ($\mathbf{x}_{ij} = \mathbf{x}_i - \mathbf{x}_j$). As the pairwise differences $\mathbf{x}_{ij}$ are symmetric this space has zero mean and is invariant to the actual locality of the samples in the feature space. This allows to re-write Eq. (4.11) to

$$\delta\left(\mathbf{x}_{ij}\right) = \log\left(\frac{p\left(\mathbf{x}_{ij} | H_0\right)}{p\left(\mathbf{x}_{ij} | H_1\right)}\right) = \log\left(\frac{f\left(\mathbf{x}_{ij} | \theta_0\right)}{f\left(\mathbf{x}_{ij} | \theta_1\right)}\right) , \tag{4.12}$$

where $f\left(\mathbf{x}_{ij} | \theta_1\right)$ is a pdf with parameters θ_1 for hypothesis H_1 that a pair is similar

$(i,j) \in \mathcal{S}$. Vice-versa $f\left(\mathbf{x}_{ij}|\theta_0\right)$ is a pdf with parameters θ_0 for hypothesis H_0 for a pair being dissimilar $(i,j) \in \mathcal{D}$. Assuming a Gaussian structure of the difference space Eq. (4.12) can be re-written to

$$\delta\left(\mathbf{x}_{ij}\right) = \log\left(\frac{\frac{1}{\sqrt{2\pi|\Sigma_{\mathcal{S}}|}}\exp\left(-\frac{1}{2}\,\mathbf{x}_{ij}^{\top}\,\Sigma_{\mathcal{S}}^{-1}\,\mathbf{x}_{ij}\right)}{\frac{1}{\sqrt{2\pi|\Sigma_{\mathcal{D}}|}}\exp\left(-\frac{1}{2}\,\mathbf{x}_{ij}^{\top}\,\Sigma_{\mathcal{D}}^{-1}\,\mathbf{x}_{ij}\right)}\right), \tag{4.13}$$

where $\Sigma_{\mathcal{S}}$ and $\Sigma_{\mathcal{D}}$ are the covariance matrices of $\mathcal{S}$ and $\mathcal{D}$, respectively. Let

$$\mathbf{C}_{ij} = (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^{\top} \tag{4.14}$$

be the outer product of the pairwise differences of $\mathbf{x}_i$ and $\mathbf{x}_j$, then the covariance matrices can be written as

$$\Sigma_{\mathcal{S}} = \frac{1}{|\mathcal{S}|}\sum_{(i,j)\,\in\,\mathcal{S}}\mathbf{C}_{ij}\,, \tag{4.15}$$

$$\Sigma_{\mathcal{D}} = \frac{1}{|\mathcal{D}|}\sum_{(i,j)\,\in\,\mathcal{D}}\mathbf{C}_{ij}\,. \tag{4.16}$$

The maximum likelihood estimate of the Gaussian is equivalent to minimize the distances from the mean in a least squares manner. This allows for finding respective relevant directions for the set of similar pairs $\mathcal{S}$ and the set of dissimilar pairs $\mathcal{D}$. By taking the log, the likelihood-ratio test can be written as

$$\delta(\mathbf{x}_{ij}) = \mathbf{x}_{ij}^{\top}\,\Sigma_{\mathcal{S}}^{-1}\,\mathbf{x}_{ij} + \log\left(|\Sigma_{\mathcal{S}}|\right) - \mathbf{x}_{ij}^{\top}\,\Sigma_{\mathcal{D}}^{-1}\,\mathbf{x}_{ij} - \log\left(|\Sigma_{\mathcal{D}}|\right)\,. \tag{4.17}$$

Further, the constant terms can be discarded, simplifying Eq. (4.12) to

$$\delta\left(\mathbf{x}_{ij}\right) = \mathbf{x}_{ij}^{\top}\,\Sigma_{\mathcal{S}}^{-1}\,\mathbf{x}_{ij} - \mathbf{x}_{ij}^{\top}\,\Sigma_{\mathcal{D}}^{-1}\,\mathbf{x}_{ij} = \mathbf{x}_{ij}^{\top}\left(\Sigma_{\mathcal{S}}^{-1} - \Sigma_{\mathcal{D}}^{-1}\right)\mathbf{x}_{ij}. \tag{4.18}$$

Finally, the learned Mahalanobis distance metric parameterized by the $d \times d$ matrix $\mathbf{M}$

$$d_{\mathbf{M}}^{2}\left(\mathbf{x}_i, \mathbf{x}_j\right) = \left(\mathbf{x}_i - \mathbf{x}_j\right)^{\top}\mathbf{M}\left(\mathbf{x}_i - \mathbf{x}_j\right) \tag{4.19}$$

is obtained by

$$\mathbf{M} = \left[\Sigma_{\mathcal{S}}^{-1} - \Sigma_{\mathcal{D}}^{-1}\right]_* . \tag{4.20}$$

To guarantee that $\mathbf{M}$ is p.s.d. and induces a valid pseudo-metric we define $\left[\hat{\mathbf{M}}\right]_* = \mathbf{M}$ as a projection operator similar to [52] that allows for finding the nearest positive semi definite matrix. In particular, the operator re-projects $\hat{\mathbf{M}} = \left(\Sigma_{\mathcal{S}}^{-1} - \Sigma_{\mathcal{D}}^{-1}\right)$ onto the cone of positive semidefinite matrices by

$$\left[\hat{\mathbf{M}}\right]_* = \mathbf{M} \tag{4.21}$$

$$\hat{\mathbf{M}} = \mathbf{X}^\top \mathbf{\Lambda} \mathbf{X} \tag{4.22}$$

$$\mathbf{M} = \mathbf{X}^\top \mathbf{\Lambda}' \mathbf{X} \tag{4.23}$$

$$\mathbf{\Lambda}' = \operatorname{diag}(\max(0, \lambda_1), \ldots, \max(0, \lambda_n)) \tag{4.24}$$

clipping the negative spectrum of $\hat{\mathbf{M}}$ by eigen-decomposition. Alternatively, it is also possible to shift the spectrum of $\hat{\mathbf{M}}$. Further, if no vectorial representation of the data is desired this step can be omitted. The resulting metric (Eq. (4.20)) reflects the properties of the log-likelihood ratio test. Thus, ideally, dissimilar pairs score high values and similar pairs score low values.

Further, KISSME is in in training orders of magnitudes faster than comparable Mahalanobis metric learning methods as it does not require iterative optimization. Also the matrix inversion is not a too costly step for two reasons: First, the inverse has to be calculated only twice; Second, since we have symmetric matrices more efficient solvers can be used. However, at evaluation the computational burden remains as k-NN search in high-dimensional spaces is cumbersome. Thus, succeeding we propose our metric based hashing scheme which mitigates some of these issues.

4.4 KISS HASH

In the following, we introduce our metric-based hashing scheme taking advantage of both, efficient learning and evaluation. The main idea is to efficiently learn a Mahalanobis metric by KISSME, which captures the intrinsic structure of the feature space, and to approximate it using hashing techniques, enabling a significant speed-up at test time.

The main goal of hashing is to reduce the classification effort by using more compact representation. In particular, by mapping the features from a d dimensional original space to a lower m dimensional space, where $m \ll d$. A widely used approach is to apply a Hamming embedding, where the data is represented in form of binary strings. This allows to compare the data via XOR operations, which can be efficiently computed by special purpose instructions on modern computer hardware. Given a sample $\mathbf{x}$, its binary hash-code $\mathbf{h}$ ($m \times 1$) can be obtained via

$$\mathbf{h}(\mathbf{x}) = \text{sign}(\mathbf{P}\mathbf{x} + \mathbf{t}) \ , \tag{4.25}$$

where $\mathbf{P}$ is a hashing matrix ($m \times d$) and $\mathbf{t}$ ($m \times 1$) is a threshold vector.

As minimization of the distances in Hamming space is related to the minimization of the distances in original space, in the following two embedding strategies are derived, exploiting the information captured by a Mahalanobis distance. The only requirement for this relation is that the hashing function sustains the locality sensitive hashing (LSH) requirement [15, 44] that the probability of a collision in the hash table is related to the similarity in the original space. In the following, the two different metric-based hashing strategies are described: (a) Via random hyperplane hashing (Section 4.4.1) and (b) via eigen-hashing (Section 4.4.2). In addition, in Section 4.4.3 a simple re-ranking scheme for hashing is introduced.

4.4.1 Hashing by random hyperplanes

As the metric matrix $\mathbf{M}$ obtained in Section 4.3 Eq. (4.20) is positive semi-definite (p.s.d.) it can be decomposed as $\mathbf{M} = \mathbf{L}^\top \mathbf{L}$ by Cholesky factorization. The matrix $\mathbf{L}$ can be seen as linear transformation that scales and rotates the feature space according to $\mathbf{M}$. After applying the linear transformation one can perform standard locality sensitive hashing techniques as random hyperplane hashing similar to [15, 63].

Thus, to obtain the hash value for a single bit h_i the feature vector $\mathbf{x}$ is first transformed by $\mathbf{L}$ and then projected onto a random vector $\mathbf{r}_i$ that is drawn from a Gaussian distribution with zero mean and unit variance:

$$h_i(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{r}_i^\top \mathbf{L}\mathbf{x} \geq t_i \\ -1 & \text{otherwise .} \end{cases} \tag{4.26}$$

For the threshold t_i we propose to set it unsupervised case either to zero or to balance

the split. In the supervised case the threshold can be selected by line search to minimize, e.g., the false positive and false negative rate. Further, let

$$\mathbf{R}_m = [\mathbf{r}_1 \dots \mathbf{r}_m] \tag{4.27}$$

be a matrix composed of m random vectors r_i, where m is the desired dimensionality of the binary Hamming string. Thus, a m-dimensional hash code $\mathbf{h}(\mathbf{x})$ over all feature dimensions can be estimated as follows:

$$\mathbf{h}(\mathbf{x}) = \text{sign}\left(\mathbf{R}_m^\top \mathbf{L}\mathbf{x} + \mathbf{t}\right) . \tag{4.28}$$

4.4.2 Hashing by eigen-decomposition

The second proposed hashing method is by eigen-decomposition. Since $\mathbf{M}$ is p.s.d. it can be decomposed as $\mathbf{M} = \mathbf{V}\mathbf{D}\mathbf{V}^\top$. This allows for hashing with the eigenvectors $\mathbf{v}_i$ as follows:

$$h_i(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{v}_i^\top \mathbf{x}_i \geq t_i \\ -1 & \text{otherwise .} \end{cases} \tag{4.29}$$

Again, let

$$\mathbf{V}_m = [\mathbf{v}_1 \dots \mathbf{v}_m] \tag{4.30}$$

be the matrix containing the eigenvectors associated with the largest eigenvalues, the m-dimensional hash code for the the feature vector $\mathbf{x}$ can be estimated by

$$\mathbf{h}(\mathbf{x}) = \text{sign}\left(\mathbf{V}_m^\top \mathbf{x} + \mathbf{t}\right) . \tag{4.31}$$

Similar to the random hyperplane hashing we propose to set the thresholds to balance the split in the unsupervised case or to select them by line search to minimize e.g. the false positive and false negative rate in the supervised case.

4.4.3 Retrieval of hashed Examples

The Hamming embedding enables a very efficient search based on the compact binary representation. Further, on modern CPUs special purpose instructions exist that are

even able to calculate the Hamming distance in a few clock-cycles. For instance Intel introduced together with SSE 4.2 the application-targeted accelerator instruction POPCNT. It allows to efficiently count the number of bits set after performing an logical XOR. Also approximate search strategies exist that are tailored to the search in Hamming space (e.g., [15] or [104]).

For the proposed method the focus is on short binary codes that can be efficiently matched followed by a re-ranking step. In particular, a short list of samples is generated by searching in Hamming space, which is then used for exact k-NN search with the learned metric. To ensure efficiency, compact codes are used in the first step and only a rather small subset of samples is re-ranked. In particular, the aim is to re-rank $\mathcal{O}(N^{\frac{1}{1+\epsilon}})$ samples, where N is the number of training samples in the respective dataset. For instance, if $\epsilon = 1$ only $\mathcal{O}(\sqrt{N})$ samples have to be checked. Thus, for higher values of ϵ less samples have to be re-ranked.

4.5 Experiments

In this section we aim at experimentally validating the proposed efficient metric learning and hashing method. On the one hand, KISSME enables very efficient training and on the other hand by hashing a significant speed-up at test time can be obtained due to the approximate nearest neighbor search. Both methods are quite general and can be applied to a variety of applications. Thus, additionally to face recognition the methods are also evaluated on standard machine learning and object matching benchmarks. The tasks include handwritten digit recognition, person re-identification and matching previously unseen object instances.

4.5.1 Efficient Large Scale Metric Learning

To show the broad applicability of KISSME, the goals of the experiments are twofold. The first objective is to show that KISSME is able to generalize to unseen data as well as or even better than state-of-the-art metric learning approaches, especially for face recognition. The second objective is to prove that the training is orders of magnitudes faster. This is clearly beneficial for large scale or online applications.

For the comparison to other metric learning approaches the numbers are generated with original code and same input data. The code is kindly provided by the respective authors. Further, KISSME is compared to related domain specific approaches. These algorithms use of course a different extraction, pre-, postprocessing and machine learning

pipeline. For these the numbers are taken from the corresponding publications. For all plots the numbers in parentheses denote the Equal Error Rate (EER) of the respective method.

4.5.1.1 Face Verification

In the following, the performance of KISSME is demonstrated on two challenging face recognition datasets, namely on Labeled Faces in the Wild (LFW) [58] and Public Figures Face Database (PubFig) [82]. Hereby, the study of face recognition is divided into two different objectives: Face identification (naming a face) and face verification (deciding if two face images show the same individual). The nature of the face identification task requires a number of annotated faces per individual. Not all real-world databases comply with this requirement. In contrast, face verification needs cheaper annotations and therefore can be evaluated more seriously also on a large scale. In this section we focus on the face verification task.

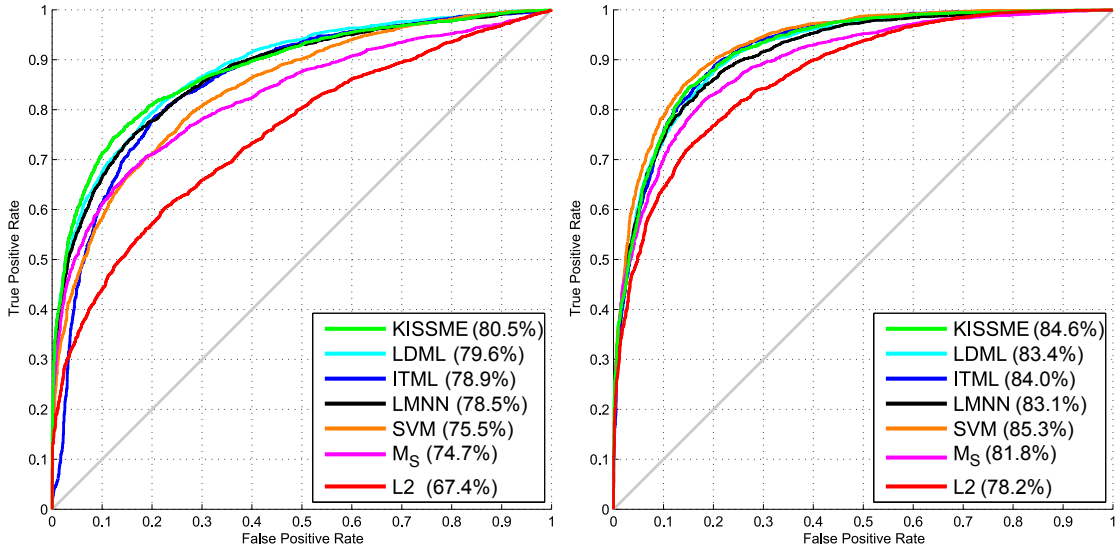
Labeled Faces in the Wild The dataset is organized in 10 folds for cross-validation. Each fold consists of 300 similar and 300 dissimilar pairs. The result scores are averaged over the 10 folds. In the restricted protocol it is only allowed to consider the equivalence constraints given by the similar / dissimilar pairs. No inference on the identity of the subject, e.g., to sample more training data, is allowed.

For the experiments the face representation proposed by Guillaumin et al. [50] is used. Basically, it extracts SIFT descriptors [89] at 9 automatically detected facial landmarks (corners of the mouth, eyes and nose), over three scales. The resulting descriptor is a 3,456 dimensional vector. To make the calculation more tractable for the distance metric learning algorithms we perform a dimensionality reduction by PCA to a 100 dimensional subspace.

To evaluate the different metric learning methods and to enable a fair comparison the methods are trained with exactly the same features and PCA dimensions. Preliminary experiments showed that the influence of the PCA dimensionality is not too critical. Using different dimensionalities for all tested methods reveals that there is no significant change in the final face verification accuracy.

For the SVM baseline we represent a face pair by different element-wise comparisons of the two feature vectors. In particular, by obtaining the absolute value of the feature difference and also the element-wise product. This allows to reformulate the verification task as a standard two class classification problem. Further, the linear SVM baseline is

trained directly on the full features without dimensionality reduction as this delivers the best results.



(a) ROC/SIFT (b) ROC/Attributes

Figure 4.2: **Face verification results on the LFW dataset**: ROC curves for different feature types and learners: In (a) we report the performance for the SIFT features of [50] and in (b) for the "high-level" description of visual face traits [82]. For the SIFT features our method outperforms several metric learning approaches slightly. For the attributes it matches with the SVM based approach proposed by [82].

In Figure 4.2 (a) Receiver Operator Characteristic (ROC) curves are provided for LDML [50], ITML [27], LMNN [145], SVM [35], KISSME, the Mahalanobis distance of the similar pairs $\mathbf{M}_S$ and the Euclidean distance as baseline. Please note that for LMNN more supervision in form of the actual class labels has to be provided (not just equivalence constraints) as the algorithm needs to sample labeled triplets. Thus, LMNN is slightly favored over the other methods.

The Mahalanobis distance of the similar pairs $\mathbf{M}_S$ performs already quite well in comparison to the Euclidean distance. It increases the performance by about 7%. Interestingly, LMNN is not really able to capitalize on the additional information over the other learned metrics as ITML or LDML. Nevertheless, KISSME outperforms the other metrics slightly. It reaches with an Equal Error Rate of 80.5% the best reported results up to now for this feature type. Of course recent state-of-the-art on LFW provides better results but also requires considerably more domain knowledge, i.e., pose specific classifiers or an auxialiary identity set.

When analyzing the training times given in Table 4.1 the main advantage of KISSME is

Method	LFW [58]	PubFig [82]	VIPeR [47]	ToyCars [105]
KISSME	**0.05s**	**0.07s**	**0.01s**	**0.04s**
SVM [35]	12.78s	0.84s	0.10s	0.60s
ITML [27]	24.81s	20.82s	8.60s	14.05s
LDML [50]	307.23s	2868.91s	0.72s	1.21s
LMNN [145]	1198.69s	783.66s	27.56s	0.79s

Table 4.1: **Average training times**. LDML, LMNN make use of multi-threading. Evaluated on a 3.06 GHz Xeon with 24 cores and 96 GB ram.

obvious. In fact, compared to LMNN, ITML and LDML the method is computationally much more efficient, however, still yielding competitive results.

Public Figures Face Database The face verification benchmark of PubFig consists of 10 cross-validation folds with 1,000 intra and 1,000 extra-personal pairs each. Per fold the pairs are sampled of 14 individuals. Similar to the LFW benchmark individuals that appear in testing have not been seen before in training.

An interesting aspect of the database is that "high-level" features are provided that describe the presence or absence of visual face traits. The appearance is automatically encoded in either nameable attributes such as gender, race, age, hair etc. or "similes" that relate the similarity of face regions to specific reference people. This indirect description yields nice properties such as a certain robustness to image variations compared to low-level features. Further, it offers us a complementary feature type to evaluate the performance of the distance metric learning algorithms.

In Figure 4.3 ROC curves are provided for LDML [50], ITML [27], LMNN [145], SVM [14], KISSME and two baselines. It can be seen that KISSME outperforms ITML, LMNN and matches the state-of-the-art performance of the SVM based method proposed by Kumar et al. [82]. LDML delivers similar results to our algorithm while being orders of magnitudes slower in training (see Table 4.1). This makes the algorithm impracticable for online or large-scale use. Interestingly, the performance of ITML drops even below the Euclidean distance. In Figure 4.2 (a) the performance of the attribute features are also reported on LFW.

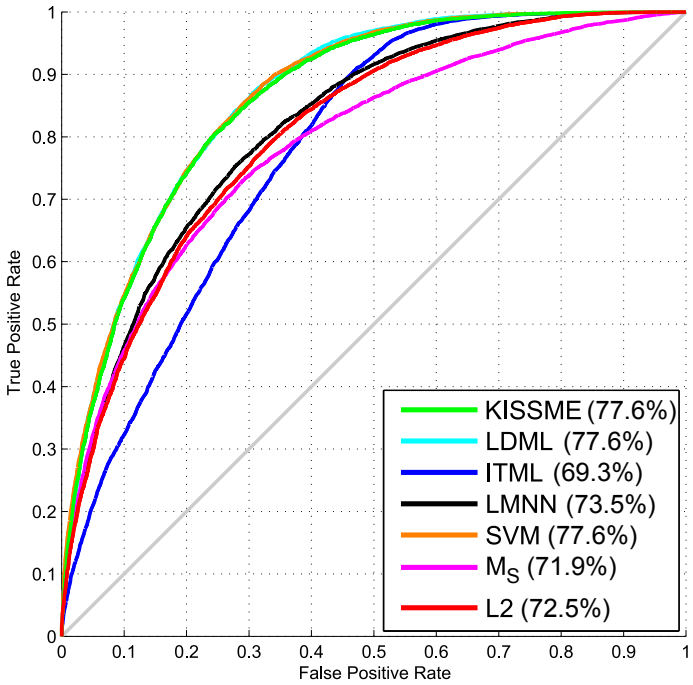
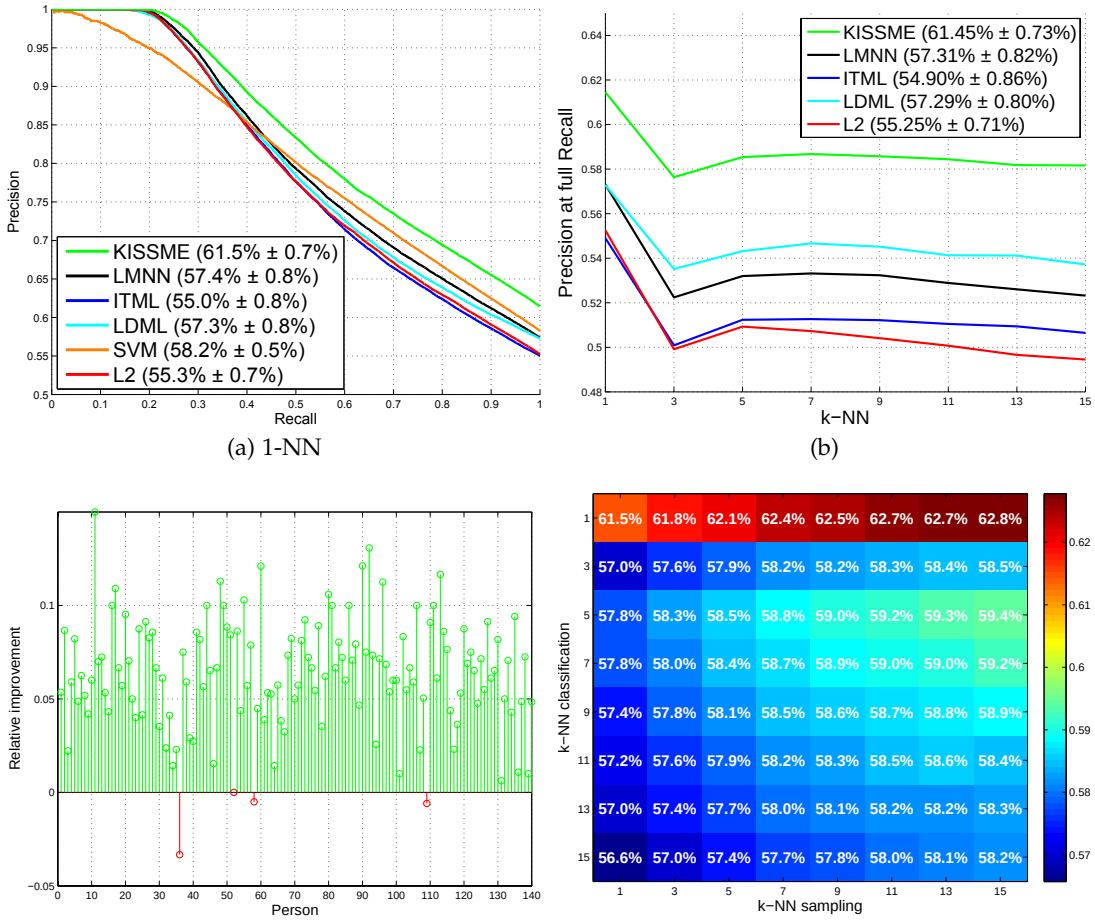


Figure 4.3: **Face verification results on the PubFig dataset**. For all methods we use the "high-level" description of visual face traits [82]. Our method (KISSME) matches the performance of LDML and the state-of-the-art [82] while being orders of magnitudes faster in training.

4.5.1.2 Face Identification

In the following, KISSME is compared to the related metric learning approaches for face identification on the challenging Public Figures Face Database (PubFig) [82]. PubFig offers compared to LFW more images per person. In total the database contains in the evaluation set 140 people with in average roughly 300 images per person. Therefore, it is also possible to evaluate face identification. For the intended face identification benchmark we organize the dataset according to the existing verification protocol in 10 folds for cross-validation. Therefore, the images of each person are split into 10 disjoint sets.

In Figure 4.4 (a)-(b) we benchmark KISSME to recent Mahalanobis metric learning methods and a standard linear SVM [35]. Please note that the face identification performance is reported in a refusal to predict style. In that sense, recall means the percentage of samples which have a higher classifier score than the current threshold. Precision means the ratio of correctly labeled samples. For instance a recall of 10% means that the classifier is requested to label the 10% most confident samples, the remaining 90% are



(a) 1-NN (b)

(c) KISSME vs Euclidean (d) KISSME influence of k

Figure 4.4: **Comparison of 1-NN classification accuracy (%) on Public Figures Face Database (PubFig)**. (a) recall / precision by ranking and thresholding classifier scores. (b) Precision at full recall over various k-NN values. (c) Relative improvement per subject between KISSME and the Euclidean distance. (d) Influence of the parameter k in classification and sampling of the pairs for KISSME.

not taken into consideration. Then the precision referes to the ratio of samples, in this specific 10% subset, which are correctly labeled.

To train KISSME the pairwise labels are generated from the class labels. In particular, for each sample the k-NN sharing the class label are picked to form matching pairs with the sample. To form non-matching pairs the samples are chosen that have a different class label and invade the k-NN perimeter, similar to [145]. Further, for the metric learning methods k-NN is applied as classifier. Testing several values of k revealed that 1-NN classification delivers the best results for all methods on this dataset. Nevertheless, the relative results are comparable for all values of k, as can be seen in Figure 4.4 (b).

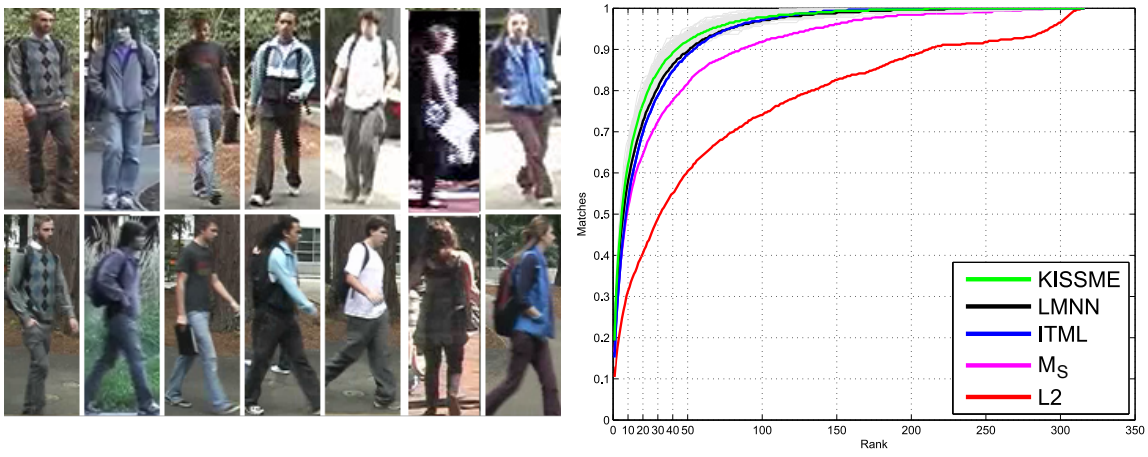
Additionally, we evaluate for KISSME the influence of the size of the k-NN perimeter in sampling of the similar / dissimilar pairs. In illustration in Figure 4.4 (d) it can be observed that more sampled pairs deliver a better performance. Note that for 1-NN classification using a k-NN perimeter of 15 even increases the accuracy to 62.8%, compared to 61.5% using a k-NN perimeter of 1 sample. In the following discussion we focus on the 1-NN experiment, using 1-NN sampling for KISSME, illustrated in Figure 4.4 (a).

In particular in Figure 4.4 (a) it can be seen that KISSME generalizes better than LMNN [145], ITML [27] or LDML [50], which require more computational effort in training. Until a recall of 20% all the metric learning methods deliver a similar performance close to 100%. In comparison the performance of the linear SVM decreases right from the beginning. Finally, at full recall the performance difference comparing KISSME to ITML is 6.5%, to LDML 4.2% and to LMNN 4.1%. The linear SVM is with 58.2% at full recall comparable to the other metric learning methods. Nevertheless, the performance difference to KISSME is 4.3%. Thus, KISSME is able to reduce the required training effort and to deliver a better performance compared to the other metric learning methods and the baselines.

4.5.1.3 Additional Benchmarks

Succeeding, KISSME is also studied for the problem of person re-identification across spatially disjoint cameras and used to compare before unseen object instances on the INRIA ToyCars dataset. The main intuition is to experimentally validate that KISSME is general and thus not limited to face recognition.

Person Re-Identification The VIPeR dataset [47] consists of 632 intra-personal image pairs of two different camera views, captured outdoors. The low-resolution images (48×128 px) exhibit significant variations in pose, viewpoint and also considerable changes in illumination, like highlights or shadows. Most of the example pairs contain a perspective change of about 90 degrees, making person re-identification very challenging. Some examples are given in Figure 4.5 (a). To compare KISSME to other approaches, the evaluation protocol defined in [36, 46] is used. Therefore, the set of 632 image pairs is randomly split into two sets of 316 image pairs each, one for training and one for testing, and compute the average over several runs. There is no predefined set or procedure how to sample dissimilar pairs. Hence, dissimilar pairs are generated by randomly combining images of different persons.



(a) Example Pairs (b) CMC

Figure 4.5: **Person re-identification results on the VIPeR dataset**. Example image pairs are shown in (a). In (b) average Cumulative Matching Characteristic (CMC) curves over 100 runs are plotted. The proposed method (KISSME) slightly outperforms the other metrics. In light-gray all 100 runs of KISSME are indicated.

To represent the images a rather simple descriptor is compiled. First, the images are divided into overlapping blocks of size 8×16. Second, color and texture cues are extracted. For the color cues the HSV and Lab quantized block mean is extracted per channel with a stride of 4×4. Texture information is captured by LBPs [106]. The LBP blocks are extracted with a stride of 8×8. Finally, for the distance metric learning approaches the concatenated 20,480 dimensional descriptors are projected in a 34 dimensional PCA subspace, although the influence of the PCA dimensionality is not too critical. Using different dimensionalities for all tested methods reveals that there is no significant change in performance.

To indicate the performance of the various algorithms Cumulative Matching Characteristic (CMC) curves [141] are reported. These represent the expectation of the true match being found within the first n ranks. To obtain a reasonable statistical significance the results are averaged over 100 runs.

In Figure 4.5 (b) the CMC curves are reported for the various metric learning algorithms. Moreover, in Table 4.2 (b) the performance of KISSME is compared in the range of the first 50 ranks to state-of-the-art person re-identification methods [30, 36, 54, 165]. As can be seen, competitive results are obtained across all ranks. KISSME outperforms the other methods [36, 46, 118] even though in contrast to them it does not require a foreground-background segmentation. Further, KISSME is computationally more effi-

RANK	**1**	**10**	**25**	**50**
KISSME	**19.6%**	**62.2%**	**80.7%**	**91.8%**
LMNN [145]	19.0%	58.1%	76.9%	89.6%
ITML [27]	15.2%	53.3%	74.7%	88.8%
LDML [50]	10.4%	31.3%	44.6%	60.4%
$\mathbf{M}_S$	16.8%	50.9%	68.7%	82.0%
L2	10.6%	31.8%	44.9%	60,8%

(a)

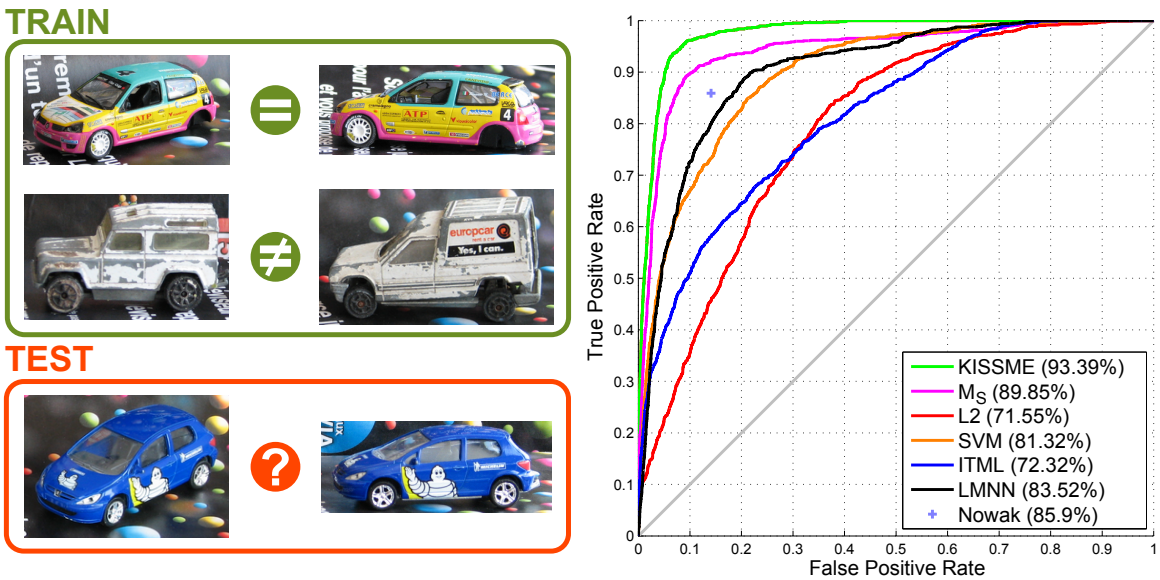
RANK	**1**	**10**	**25**	**50**
KISSME	**20%**	**62%**	**81%**	**92%**
SDALF [36]	**20%**	50%	70%	85%
DDC [54]	19%	52%	69%	80%
PRDC [165]	16%	54%	76%	87%
KISSME*	**22%**	**68%**	**85%**	**93%**
LMNN-R* [30]	20%	**68%**	84%	**93%**

(b)

Table 4.2: **Person re-identification matching rates on the VIPeR dataset**. Table (a) shows the metric learning approaches (average of 100 runs) whereas (b) gives an overview of the state-of-the-art. To be comparable to LMNN-R we also report the best run (*).

cient as can be seen in Table 4.1.

ToyCars The LEAR ToyCars [105] dataset consists of 256 image crops of 14 different toy cars and trucks. The dataset exhibits changes in pose, lighting and cluttered background. The intention of the database is to compare before unseen object instances of the known class *cars* (see Figure 4.6 (a) for illustration). Thus, in testing the task is to classify if a pair of images shows the same object or not. The training set contains 7 object instances with associated 1,185 similar and 7,330 dissimilar image pairs. The remaining 7 object instances are in the test set. The images differ in horizontal resolution. Thus, these are zero-padded to obtain a canonical image size.



(a) (b)

Figure 4.6: **ROC curves on LEAR ToyCars dataset**. (a) The task is to decide if two before unseen object instances of the known class cars are similar or not. (b) KISSME is able to drop error rates significantly compared to the previous work of Nowak and Jurie [105].

A similar image representation as in the person re-identification experiment is extracted. Therefore, the images are divided into 30×30 non-overlapping blocks. Color cues are captured by HSV and Lab means while texture is described by LBPs [106]. The global image descriptor is a concatenation of the local ones. Using PCA the descriptor is projected onto a 50 dimensional subspace.

The experiments on this dataset compare KISSME to the approach of Nowak and Jurie [105], which builds on an ensemble of extremely randomized trees. The ensemble quantizes corresponding patch pair differences by enforcing that corresponding patches of matching pairs yield similar responses. Corresponding patches are located in a local neighborhood by Normalized Cross Correlation (NCC). In testing the similarity between an image pair is the weighted sum of corresponding patches that end up in the same leaf node. The weights are learned by a linear SVM.

In Figure 4.6 ROC curves are plotted which compare KISSME to the work of Nowak and Jurie [105] and the related metric learning approaches. Further, a standard baseline is provided with a off-the-shelf linear SVM [35]. Using SVM yields an EER of 81%, already a reasonable performance. Interestingly, some of the metric learning approaches are not able to improve over the Euclidean distance. Only LMNN performs similar to the SVM. The Mahalanobis distance learned of the positive pairs already outperforms

the approach of Nowak and Jurie [105] and reaches an EER of 89.8%. KISSME boosts the performance further up to 93.5% at a computation time of 0.04 seconds. Considering the computation time of [105] with 17 hours (P4-3.4GHz) KISSME once more shows its benefits in terms of efficiency and effectiveness.

4.5.2 Efficient Retrieval for Large Scale Metric Learning

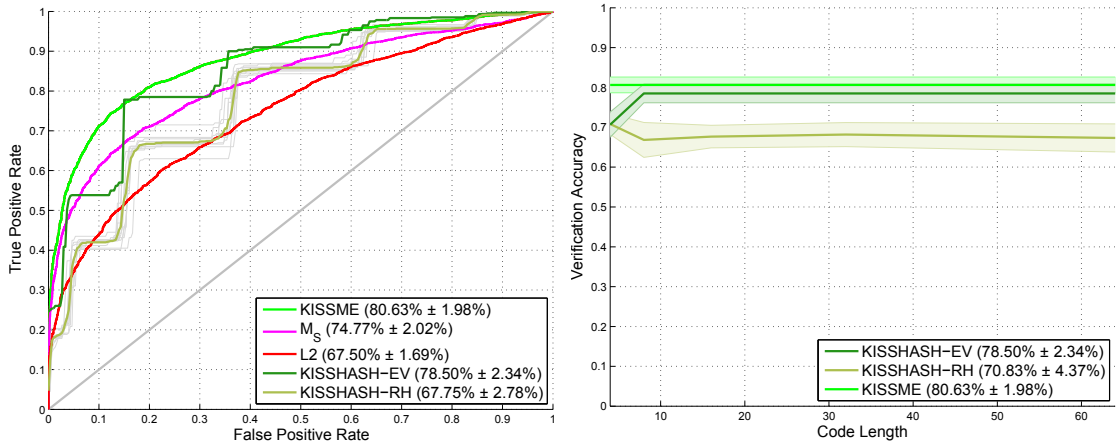
In this section the aim is to experimentally validate the proposed metric-based hashing strategy. In particular the focus is to show that by approximate nearest neighbor search KISSHASH leads to a drastically reduced evaluation effort at virtually no loss in accuracy for k-NN classification. In fact, if the intrinsic structure of the data is exploited by KISSME using hashing the feature representation can be compacted still obtaining competitive results. First, KISSHASH is evaluated for face verification and identification. Please note that for face verification the re-ranking step is not applicable as only two descriptors are compared. Thus, performance drops are observed in this case. Nevertheless, the experiments are provided for a direct comparison to KISSME. Second, KISSHASH is additionally evaluated on standard machine learning benchmarks to enable a comparison to the related work of locality sensitive hashing methods.

4.5.2.1 Face Verification

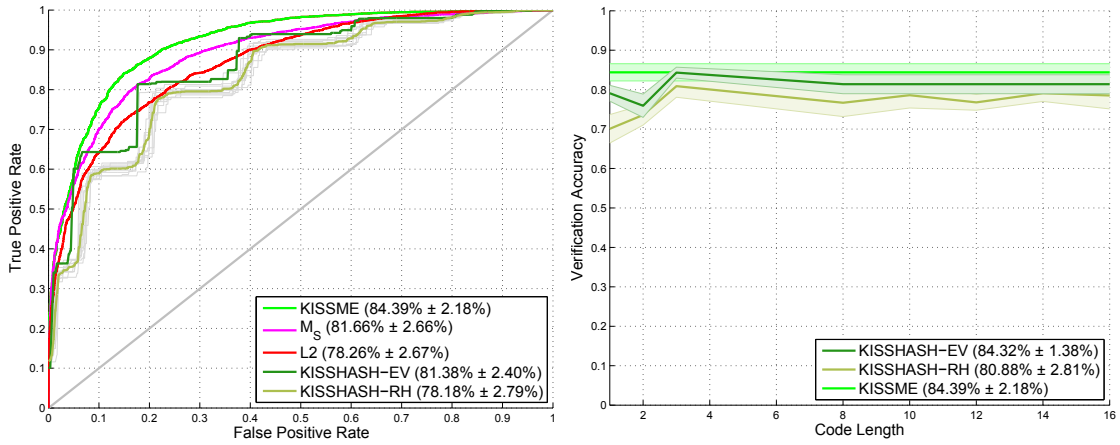
In the following, the performance of KISSHASH is demonstrated for face verification comparing to KISSME on LFW [58] and PubFig [82].

Labeled Faces in the Wild To compare KISSHASH to KISSME once more the restricted protocol is used. The restricted protocol only allows to consider the equivalence constraints given by the similar / dissimilar pairs. No inference on the identity of the subjects, e.g., to sample more training data, is allowed. The result scores are averaged over a stratified 10 fold cross-validation. Each fold consists of 300 similar and 300 dissimilar face pairs.

Figure 4.7 shows the results comparing KISSHASH to KISSME for the SIFT based face representation proposed by Guillaumin et al. [50] and the visual attribute features of Kumar et al. [82]. In Figure 4.7 (a) KISSHASH using 64 bit codes is compared to KISSME for SIFT features. The numbers in parentheses denote the mean accuracy at EER and the standard deviation of the 10 data-folds. The ROC curves clearly show that the two proposed hashing variants are only an approximation of the original distance



(a) ROC/SIFT (b) Codelength/SIFT



(c) ROC/Attributes (d) Codelength/Attributes

Figure 4.7: **Face verification results comparing KISSHASH to KISSME on the LFW dataset**: In (a)-(b) we report the performance for SIFT features [50] and in (c)-(d) for the attribute features of [82]. (a),(c) show ROC curves whereas (b),(d) plot the verification accuracy at EER versus code length. Error bars indicate the standard deviation of the LFW folds.

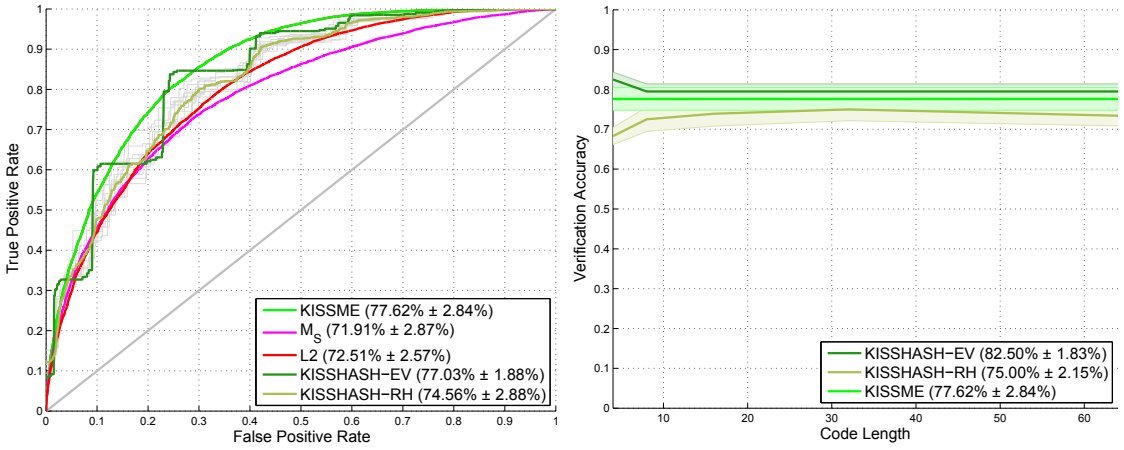
metric. Nevertheless, hashing by random hyperplanes (KISSHASH-RH) reaches with 67.75% EER a performance compareable to the Euclidean distance using drastically less effort for matching. Note that the results of KISSHASH-RH have been averaged over 10 runs as the hashing involves randomization, illustrated in light gray. The standard deviation over the 10 runs is 1.23%. The hashing by eigen-decomposition of the metric matrix (KISSHASH-EV) improves considerably over the ranomized hyperplane hashing and reaches an EER of 78.50%. Further KISSHASH-EV also improves 3.73% over the Mahalanobis distance of the similar pairs $\mathbf{M}_S$ and is comparable to LMNN and ITML.

See Figure 4.2 (a) for further details. Moreover, Figure 4.7 (b) shows the effects of varying the code length on the verification accuracy at EER. For each method the numbers in parentheses show the best obtained accuracy at EER and the standard deviation over the 10 cross-validation folds of the LFW restricted protocol. It can be seen that also very short codes allow for obtaining reasonable results. In general longer codes deliver better results although some minor deviations are possible, as for instance for very short codes for KISSHASH-RH.

Figure 4.7 (c) shows the ROC curves for the attribute based face representation. For KISSHASH-RH and KISSHASH-EV a compact code length of 16 bits is used. Compared to the code length applied for the SIFT features the codes are shorter since the attribute features are much lower dimensional. Nevertheless, KISSHASH-RH reaches the performance of the Euclidean distance whereas KISSHASH-EV is able to match the performance of the Mahalanobis distance of the similar pairs $\mathbf{M}_S$. However, at much lower computational costs in evaluation. Figure 4.7 (d) also compares the accuracy at EER versus the code length. For each method the numbers in parentheses show the best obtained accuracy at EER and the standard deviation over the 10 cross-validation folds of the LFW restricted protocol. It can be seen that also very short codes allow for obtaining reasonable results.

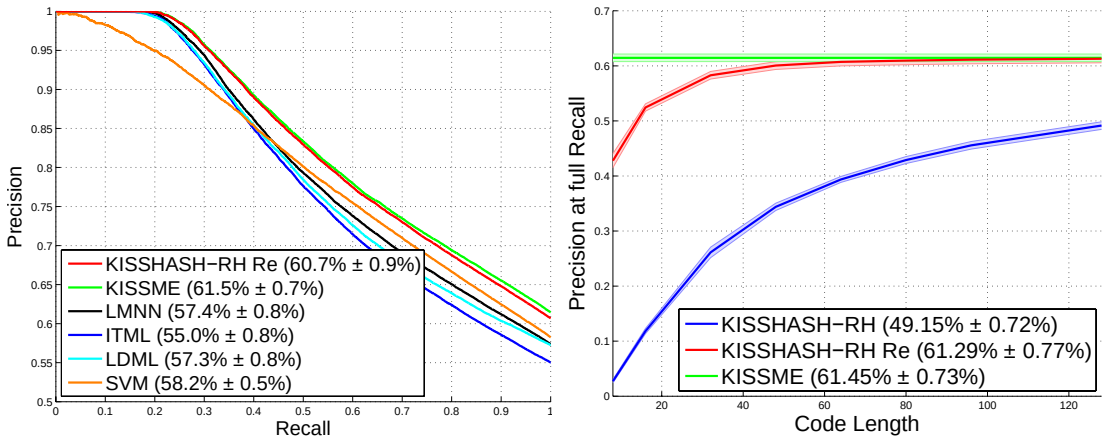
Public Figures Face Database To compare KISSHASH to the KISSME baseline the standard benchmark consisting of a stratified 10 fold cross-validation is used. Each fold contains 1,000 intra and 1,000 extra-personal pairs. Per fold the pairs are sampled of 14 individuals. Similar to the LFW benchmark individuals that appear in testing have not been seen before in training. As underlying feature representation once more the description of visual face traits is used [82].

In Figure 4.8 (a) ROC curves are provided for KISSHASH-RH, KISSHASH-EV and the baseline methods. The hashing methods use a code length of 16 bits. The results of KISSHASH-RH are averaged over 10 runs, which are illustrated in light gray. KISSHASH-RH improves with 74.56% slightly over the Euclidean distance and Mahalanobis distance of the similar pairs $\mathbf{M}_S$. The performance difference compared to the Euclidean distance is 2.05%, compared to $\mathbf{M}_S$ 2.65%. The standard devidation of the different runs is 1.36%. Interestingly, KISSHASH-EV shows once more better results for the face verification task. KISSHASH-EV reaches an accuracy of 77.03% at EER. KISSME reaches with 77.62% the best perfromance, although the performance gap to KISSHASH-EV is moderate. In Figure 4.8 (b) the face verification accuracy at EER is



(a) ROC/Attributes (b) Codelength/Accuracy

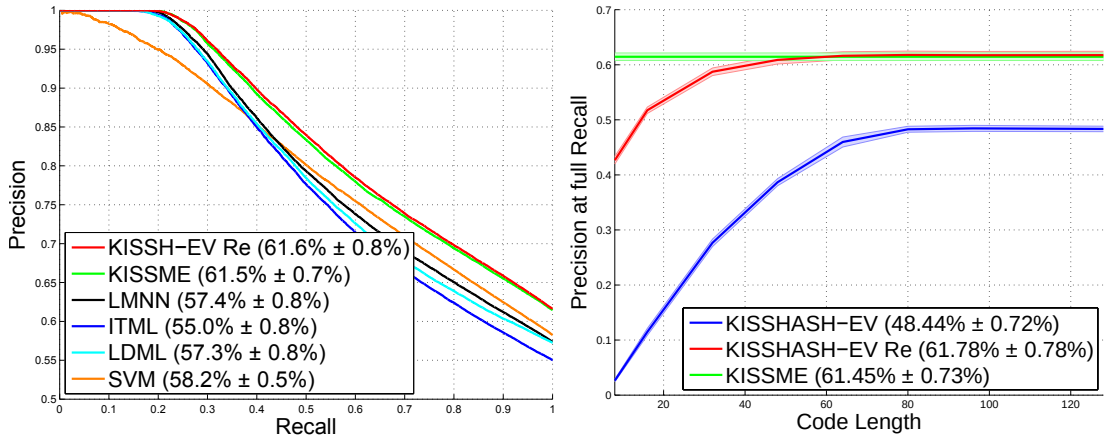
Figure 4.8: **Face verification results comparing the proposed hashing methods to KISSME on the PubFig dataset**. For all methods the visual attribute features of [82] are used. (a) For the hashing methods a code length of 16 bits is applied. In (b) the influence of the code length on the accuracy is studied. Error bars indicate the standard deviation of the dataset folds.



(a) (b)

Figure 4.9: **KISSHASH-RH with short list re-ranking.** Comparison of 1-NN classification accuracy (%) on Public Figures Face Database (PubFig). (a) recall / precision by ranking and thresholding classifier scores. Code length of 64 bits, $\epsilon = 1$. (b) Precision at full recall vs code length. Error bars indicate the standard deviation of the dataset folds.

compared to the code length.



(a) (b)

Figure 4.10: **KISSHASH-EV with short list re-ranking.** Comparison of 1-NN classification accuracy (%) on Public Figures Face Database (PubFig). (a) recall / precision by ranking and thresholding classifier scores. Code length of 64 bits, $\epsilon = 1$. (b) Precision at full recall vs code length. Error bars indicate the standard deviation of the dataset folds.

4.5.2.2 Face Identification

In the following, results are demonstrated comparing the metric learning with the hashing methods for face identification on the Public Figures Face Database (PubFig) [82]. For the face identification benchmark the data is organized similar to the existing verification protocol in 10 folds for cross-validation. Therefore, the images of each person are split into 10 disjoint sets. The face identification performance is reported in a refusal to predict style. In that sense, recall means the percentage of samples which have a higher classifier score than the current threshold. Precision means the ratio of correctly labeled samples. To train KISSME and the hashes, the needed pairwise labels are generated from the class labels. In particular, for each sample the k-NNs sharing the class label are picked to form matching pairs with the sample. Similar to form non-matching pairs the samples are chosen that have a different class label and invade the k-NN perimeter. Further, for all metric learning methods k-NN is applied as classifier. Testing several values of k revealed that 1-NN classification usually delivers the best results for all methods.

In Figure 4.9 (a) the random hyperplane hashing (KISSHASH-RH-Re) with short list re-ranking is benchmarked to recent Mahalanobis metric learning methods using 1-NN classification. The number in parentheses denote the precision at full recall and the standard deviation of the database folds. A code length of 64 bits is used and ϵ is fixed to one. The influence of ϵ on the performance is not too critical. A value of

one provides a good trade off between accuracy and runtime performance. Thus, $\sqrt{N}$ samples are re-ranked with the original distance after approximate nearest neighbor search in Hamming space. In this case only 195 of 37,647 training samples have to be re-ranked. Thus, the method is still efficient. Further, the proposed method generalizes better than LMNN [145], ITML [27] or LDML [50], which require more computational effort in evaluation. At full recall the performance difference to LMNN is 3.3%, to LDML 3.4% and to ITML even 6.5%. Further, the method also improves clearly over the linear SVM over all levels of revall. Comparing KISSHASH-RH to KISSME only a rather little performance difference of 0.8% remains.

In Figure 4.9 (b) the influence of the code length on the precision at full recall is studied. For the respective method the numbers in parentheses denote the best obtained precision at full recall and the standard deviation of the database folds. In particular, the performance of the hashing with and without short list re-ranking is compared using code lengths between 8 and 128 bits with ϵ fixed to one. Intuitively, for both methods the performance gets better if the code length is increased. Using 16 bit codes the performance gap between KISSHASH-RH-Re and KISSHASH-RH is about 40%. Also at a code length of 128 bit a performance gap of more than 10% remains. This clearly indicates the importance of the short list re-ranking step. The best performance of the plain hashing (49.15%) is obtained using a code length of 128. In contrast the best performance of hashing with short list re-ranking (61.29%) is obtained at a bit length of 80. Even using shorter codes KISSHASH-RH-Re reaches compareable performance to KISSME. Thus, using the re-ranking step enables to use shorter codes in Hamming space. This alleviates some of the increased effort induced by the additional matching. Finally this leads to a better overall performance.

In Figure 4.10 (a) hashing by eigen-decomposition of the metric matrix with short list re-ranking (KISSHASH-EV-Re) is benchmarked to recent Mahalanobis metric learning methods using 1-NN classification. Also for the face identification task KISSHASH-EV reaches slightly better results compared to KISSHASH-RH. In Figure 4.10 (a) it can be seen that KISSHASH-EV even slightly improves (0.1%) over KISSME, although this is clearly no systematic improvement as the hashing is an approximation. In Figure 4.10 (b) the influence of the code length on the precision at full recall is illustrated. The best performance of KISSHASH-EV-Re is reached with 61.78% at a code length of 80 bits. In contrast, KISSHASH-EV reaches at a code lenght of 80 bits a performance of 48.44%.

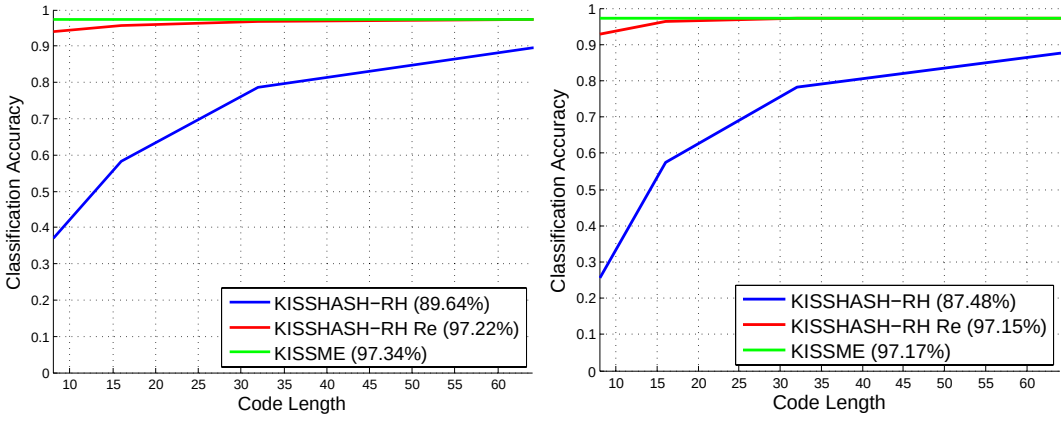
4.5.2.3 Machine Learning Databases

In the following, our metric based hashing method is benchmarked on MNIST [62], USPS [60], LETTER [62] and CHARS74k [10]. The focus is to compare KISSHASH-RH and KISSHASH-EV to the related state-of-the-art hashing approaches. First, we give a brief overview of the databases. Second, we compare the performance related to the evaluation complexity between our method and other hashing approaches.

The MNIST database [62] of hand written digits contains in total 70,000 images in one train-test split. 60,000 samples are used for training and 10,000 for testing. The images have a resolution of 28×28 pixels and are in grayscale. In contrast, the LETTER [62] database contains a large number of synthesized images showing one of the 26 capital letters of the English alphabet. The images are represented as 16-dimensional feature vectors which describe statistical moments and edge counts. Chars74K [10] contains a large mixed set of natural and synthesized characters. The images comprise one of the 26 capital or lowercase letters and digits, respectively. Thus, the dataset contains 62 classes. 7,705 characters are cropped of natural images, 3,410 are hand drawn and 62,992 are synthesized. Similar to [163] a color space conversion to grayscale is applied and each image is resized to 8×8 pixels. Further, the database is split into one train/test set where 7400 samples are organized for testing and the rest for training. For MNIST a dimensionality reduction of the raw features by PCA to a 164 dimensional subspace is performed. For all other databases the raw data without calculating any complex features is used, in order to get a fair comparison.

In Figure 4.11 the random hyperplane hashing method is compared to its baseline on MNIST, LETTER, USPS, and CHARS74k. Therefore, the 1-NN classification error in relation to the code length is plotted, where the maximum code length is restricted to 64 bits. In particular, the following results are reported: (a) Standard KISSME without hashing, (b) nearest neighbor search in Hamming space, and (c) nearest neighbor search in Hamming space with short list re-ranking. For for the re-ranking step ϵ is fixed to 1, retrieving $\mathcal{O}(\sqrt{N})$ samples, which is around 1% of samples in these cases.

The following discussion focuses on the respective results on MNIST of the random hyperplane based hashing method, visualized in Figure 4.11 (a). Nevertheless, the relative results are comparable on the different datasets. The direct nearest neighbor search in the Hamming space performs initially significantly worse than short list re-ranking. By increasing the number of codes the performance gap gets smaller. However, ultimately for MNIST a performance gap of 7.58% remains with a code length of 64 bits. This once more confirms the importance of the re-ranking step. If the short list is kept



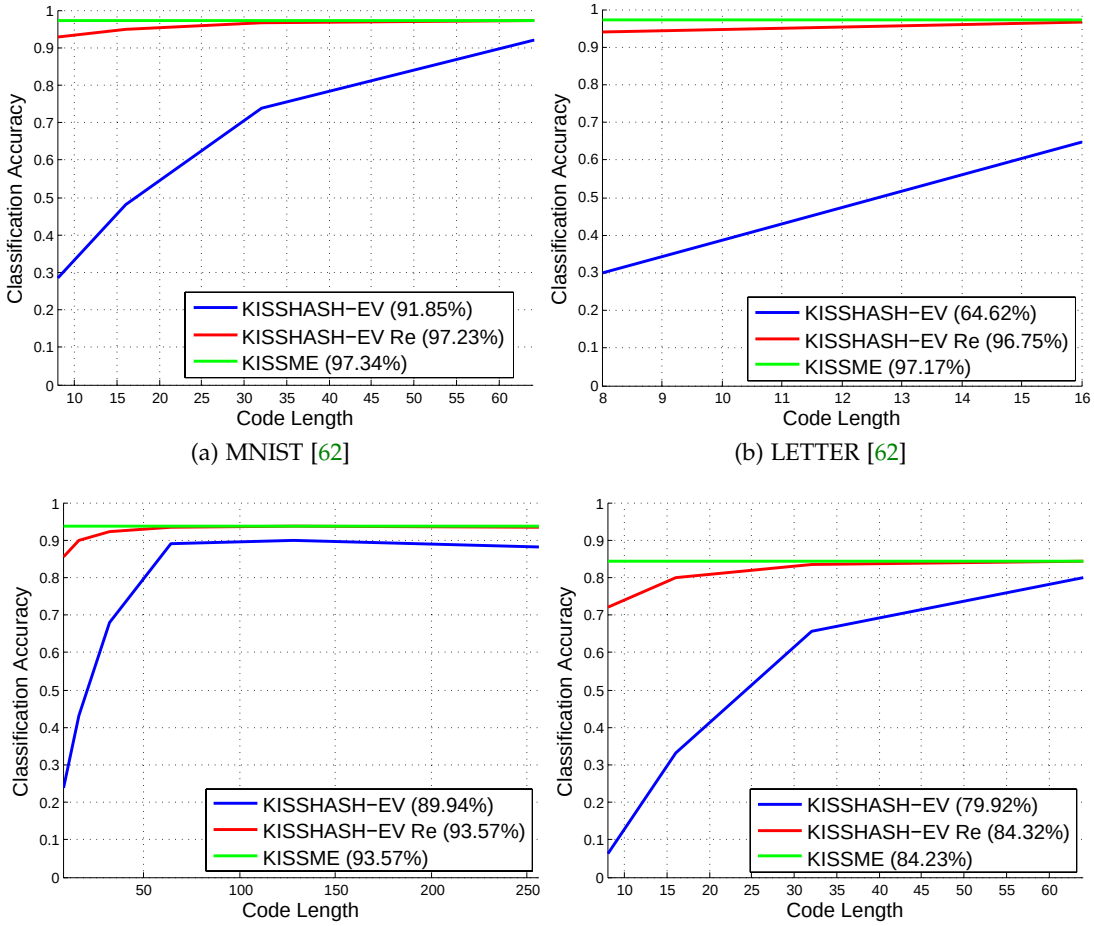
(a) MNIST [62] (b) LETTER [62]

(c) USPS [60] (d) CHARS74k [10]

Figure 4.11: **Comparison of 1-NN classification accuracy (%) on (a) MNIST, (b) LETTER, (c) USPS and (d) CHARS74k for KISSHASH-RH**. Numbers in parentheses denote the classification accuracy using a code length of 64 bits.

reasonable sized the computational effort is manageable. Comparing KISSHASH-RH with re-ranking to KISSME reveals that even with short codes comparable performance can be obtained. Starting from 16 bits nearly the same performance (-1.55%) is reached at a much lower computational cost.

In Figure 4.12 the eigenvector hashing method is compared to the KISSME baseline on MNIST, LETTER, USPS, and CHARS74k. The results on the different databases are similar to the random hyperplane hashing. The direct nearest neighbor search in the Hamming space performs initially significantly worse than the short list re-ranking method. By increasing the number of codes the performance gap gets smaller. However, ultimately for MNIST a performance gap of 5.38% remains with a code length of 64 bits. In contrast the performance difference between KISSHASH-EV with re-ranking



(a) MNIST [62] (b) LETTER [62]

(c) USPS [60] (d) CHARS74k [10]

Figure 4.12: **Comparison of 1-NN classification accuracy (%) on (a) MNIST, (b) LETTER, (c) USPS and (d) CHARS74k for KISSHASH-EV**. Numbers in parentheses denote the classification accuracy with 64 bits.

and KISSME is only 0,11%.

Next, in Table 4.3 the methods are benchmarked to various state-of-the-art approaches. In particular, a closer look on different well-established Mahalanobis metric learning methods and hashing schemes is provided. Comparing KISSME to other metric learning methods, i.e., ITML, LDMLand LMNN reveals that it is competitive in most cases, though requiring drastically less training time. Further, the random hyperplane hashing method as well as the eigenvector hashing have a comparable performance to KISSME, though drastically reducing the evaluation time. Next, the classification error between the proposed methods and others is compared and related to their evaluation complexity. For the kernelized hashing approach of [81] the evaluation scales linearly with the

Methods	MNIST	USPS	LETTER	Chars74K
Nearest Neighbors				
Nearest Neighbor (1-NN, 3-NN)	2.92 - 3.09	4.88 - 5.08	4.30 - 4.35	17.97 - 19.99
LMNN $_{\text{3-NN}}$ [143, 145]	1.70	4.78	3.54	22.89
ITML $_{\text{1-NN}}$ [27]	2.17	5.23	4.75	17.00
ITML $_{\text{3-NN}}$ [27]	2.02	5.03	4.68	18.54
LDML $_{\text{1-NN}}$ [50]	4.04	9.12	11.25	18.62
LDML $_{\text{3-NN}}$ [50]	3.59	8.27	10.35	20.32
KISSME $_{\text{1-NN}}$ [75]	2.66	6.43	2.83	15.77
KISSME $_{\text{3-NN}}$ [75]	2.36	6.38	2.73	18.64
Locality-sensitive hashing				
KISS-HASH-RH $_{\text{1-NN}}$ (64 bit, $\epsilon = 1$)	2.78	6.28	2.85	16.05
KISS-HASH-EV $_{\text{1-NN}}$ (64 bit, $\epsilon = 1$)	2.77	6.53	3.25	15.68
KLSH [80, 81] (10,000 kernel samples)	6.15	7.46	7.38	88.76
Image Search f. Learn. Metrics [63] ($\epsilon = 0.6$)	5.51	-	8.55	-
Spectral Hashing [146]	4.25	-	7.42	26.03
Multidimensional Spectral Hashing [147]	5.27	-	33.67	-
Spherical Hashing [51] (256 bit)	3.19	-	31.4	18.59

Table 4.3: **Comparison of classification error rates (%) on MNIST, LETTER and Chars74k**. In particular we provide a closer look on different well-established Mahalanobis metric learning methods and further provide additional results for different locality-sensitive hashing methods.

number of kernel samples S times the kernel complexity K_c: $\mathcal{O}(SK_c)$. In most cases the kernel complexity is similar to a distance evaluation, thus comparable to the evaluation of a distance metric. KLSH requires many kernel samples to obtain similar results. In particular, RBF and learned kernels (ITML) have been tested. The locality-sensitive hashing approach of [63] scales with $\mathcal{O}(MD)$, where M is the length of the short list of samples generated by approximate search in Hamming space [15]. Even at a lower value of ϵ a performance gap remains. A lower value of ϵ means to retrieve more samples. Spherical hashing [51] scales with $\mathcal{O}(AD)$ where A is the number of anchor points (matches the code length) where the hyper spheres reside. However, the method does not match the performance of the proposed methods using a comparable number of anchor points.

Recapitulating the different results and relating them to the evaluation complexity of related works reveals that competitive are obtained results requiring less effort. Moreover, it is clearly beneficial to integrate a metric learning in hashing and to be able to model different scalings and correlations of the feature space.

4.6 Conclusion

This chapter investigated two main drawbacks of existing Mahalanobis metric learning methods: High computational effort during (a) training and (b) evaluation. To overcome these problems an efficient metric learning method (KISSME) and metric-based hashing scheme (KISSHASH) has been introduced. KISSME allows for learning a distance metric from simple equivalence constraints. In training based on a statistical inference perspective a very efficient solution is obtained which is also effective in in terms of generalization performance. Analog to the KISS principle it is conceptually simple and valuable in practice. Further, with KISSHASH a hashing scheme based on KISSME has been introduced. KISSHASH allows for exploiting the learned metric structure by hashing. This leads to a drastically reduced evaluation effort while maintaining the discriminative essence of the data. Competitive classification results are obtained, however, on a significantly reduced computational effort. To show the merits of the methods several experiments on various challenging large-scale benchmarks have been conducted, including the real-world face benchmarks LFW and PubFig. On all benchmarks KISSME is able to match or slightly outperform state-of-the-art metric learning approaches, while being orders of magnitudes faster in training. On two additional datasets (VIPeR, ToyCars) the method even outperforms approaches especially tailored to these tasks. For KISSHASH the experiments showed that in most cases the performance of the KISSME baseline is reached while being more efficient at test time. In addition, comparable or slightly better results than state-of-the-art hashing approaches are obtained. On PubFig metric learning approaches using by far more data are even outperformed.

5 Synergy-based Learning of Facial Identity

In this chapter we address the problem that most face recognition approaches neglect that faces share strong visual similarities, which can be exploited when learning discriminative models. Hence, we propose to model face recognition as multi-task learning problem. This enables us to exploit both, shared common information and also individual characteristics of faces. In particular, we build on our KISS metric learning method, which has in the previous chapter shown good performance for many computer vision problems. Our main contribution is twofold. First, we extend KISSME to multi-task learning. The resulting algorithm supports label-incompatible learning which allows us to tap the rather large pool of anonymously labeled face pairs also for face identification. Second, we show how to learn and combine person specific metrics for face identification improving the classification power.

5.1 Introduction

Between learning in the human visual system and machine learning are essential differences. Typically, when machine learning techniques learn a specific visual model they focus on individual characteristics and neglect general concepts or visual commonalities of similar objects. In contrast, the human visual system learns in a more synergistic way that benefits from commonalities and takes into account prior knowledge. Hence, for computational face recognition systems it would be beneficial also to exploit such information.

One popular concept that addresses this demand is transfer learning, which aims at improving the performance of a target learning task by also exploiting collected knowledge of different sources [107]. Two related aspects are domain adaptation and multi-task learning. Domain adaptation tries to bridge the gap between a source domain with sufficient labeled data to a specific target domain with little or no labels [107]. In contrast, multi-task learning (MTL) [12] approaches a cluster of similar tasks in parallel. Each task describes a target learning problem and contributes labeled data. The knowledge transfer between the tasks is then established through a shared intermediate representation. The basic assumption is that it is easier to learn several hard tasks simultaneously than to learn those isolated. In this way underrepresented tasks that have only a limited number of labeled samples can be handled. Prominent approaches rely on neural nets [13, 16] (sharing layers) or support vector machines [34] (sharing weight vectors).

In this chapter, we adapt multi-task learning for real-world, large-scale face recognition. In order to cope with the real-world challenges we want to incorporate as much relevant information as possible. In particular, given by similar/dissimilar labeled face pairs, where we have no access to the actual class labels. These labeled pairs are mainly used for face verification (deciding if two faces match) and are rather easy to obtain also on a large scale. For face identification it is not immediately obvious how to make use of this anonymous information. But these additional face pairs allow us to learn a more robust measure of face similarity. Multi-task learning then spreads this knowledge between the tasks. Hereby, to enable meaningful transfer of knowledge, multi-task learning faces the problem of different label sets. On the one hand side for face identification the label set consists of class labels while on the other hand side we have only equivalence labels. Thus, one important aspect of multi-task learning is label-incompatible learning, the support of different label sets for different learning tasks. Particularly, the successful multi-task adaptation of support vector machines [34] lacks this feature.

Mahalanobis metric learning methods usually operate on the space of pairwise differences, thus enabling label-incompatible learning. The method of Parameswaran and Weinberger [108] extends Mahalanobis metric learning to the multi-task paradigm. Nevertheless, due to the particular optimization it requires class labels and can thus not benefit from data just labeled with equivalence constraints. Further, it requires computationally expensive iterations making it impractical for large-scale applications. Hence, to capitalize on multi-task learning for face recognition, one faces the additional challenges of scalability and the ability to deal just with equivalence labels.

To meet these requirements, we extend our KISS learning algorithm presentend in Chapter 4 to the multi-task paradigm. The resulting algorithm enables label-incompatible learning as it only relies on pairwise equivalence labels. These are considered as natural inputs to distance metric learning algorithms as similarity functions basically establish a relation between pairs of points. In particular, we want to learn specific Mahalanobis distance metrics for each person. This is inspired by the recent finding of Weinberger and Saul [143] that especially for large-scale applications better results can be obtained by learning multiple distance metrics. Also many other learning algorithms cast a complex multi-class problem in series of simpler, often two class, problems, followed by a voting rule to form the final decision [119]. Thus, inspired by the successful strategy applied for multi-class support vector machines we intend to learn individual distance metrics. To demonstrate the merits of our method we compare it to recent multi-task and metric learning approaches on the challenging PubFig [82] face recognition benchmark.

5.2 Multi-Task Metric Learning for Face Recognition

In the following, we introduce our new multi-task metric learning approach for face recognition. In particular, we extend the metric learning algorithm (KISSME) presented in Chapter 4 to the multi-task domain. Compared to other metric learning algorithms it is well suited to learn multiple distance metrics due to its efficiency in training. We introduce also a voting scheme that allows for classification using multiple distance metrics. The overall goal is to combine several person specific metrics to a multi-class decision which should lead to lower error rates.

5.2.0.4 Multi-Task Metric Learning

The general idea of multi-task learning is to consider T different, but related learning tasks in parallel. In our case a task is to learn a face verification model for a specific person, and the relation is intuitively given via the shared visual properties of faces. There are different concepts to realize such a setting. In particular, we adopt the formulation of Parameswaran and Weinberger [108]. We model the individual metric for each task $t \in \{1, 2, \ldots, T\}$ as combination of a shared metric $\mathbf{M}_0$ and a task-specific metric $\mathbf{M}_t$:

$$d_t^2(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^\top (\mathbf{M}_0 + \mathbf{M}_t)(\mathbf{x}_i - \mathbf{x}_j) \; . \tag{5.1}$$

Each task defines a subset of task specific samples given by the index set $\mathcal{I}_t$. Hence, to adopt the formulation Eq. (5.1) for KISS metric, we have to define a task-specific subset of similar and dissimilar sample pairs: $\mathcal{S}_t = \{(i,j) \in \mathcal{I}_t | y_i = y_j\}$ and $\mathcal{D}_t = \{(i,j) \in \mathcal{I}_t | y_i \neq y_j\}$. In cases where $\mathcal{S}_t$ and $\mathcal{D}_t$ is not given, these sets can be sampled randomly of the actual class labels. Hence, according to Eq. (4.20) we can estimate task specific metrics by

$$\mathbf{M}_t = \left(\frac{1}{|\mathcal{S}_t|} \sum_{(i,j) \in \mathcal{S}_t} \mathbf{C}_{ij} \right)^{-1} - \left(\frac{1}{|\mathcal{D}_t|} \sum_{(i,j) \in \mathcal{D}_t} \mathbf{C}_{ij} \right)^{-1} . \tag{5.2}$$

Similarly, by estimating the weighted sum over the individual task specific characteristic we get the shared or common metric

$$\mathbf{M}_0 = \left(\frac{1}{T} \sum_{t=1}^{T} \frac{1}{|\mathcal{S}_t|} \sum_{(i,j) \in \mathcal{S}_t} \mathbf{C}_{ij} \right)^{-1} - \left(\frac{1}{T} \sum_{t=1}^{T} \frac{1}{|\mathcal{D}_t|} \sum_{(i,j) \in \mathcal{D}_t} \mathbf{C}_{ij} \right)^{-1} . \tag{5.3}$$

Then, the final individual Mahalanobis distance metric is given by

$$\hat{\mathbf{M}}_t = \mathbf{M}_0 + \mu \, \mathbf{M}_t \, . \tag{5.4}$$

Intuitively, $\mathbf{M}_0$ picks up general trends across all tasks and thus models commonalities. In contrast, $\mathbf{M}_t$ models task-specific characteristics. As only free parameter we retain a balancing factor μ between the task specific metric $\mathbf{M}_t$ and the shared metric $\mathbf{M}_0$. Intuitively, the more samples a task contributes the more focus lies on its specific metric.

5.2.0.5 Multi-Task Voting

To fully exploit the power of our multi-task metric learning method for face recognition, we combine multiple, person specific, metrics into a multi-class decision. However, the outputs of the different metrics are not necessarily compatible and cannot be compared directly. A prominent strategy to reconcile classifier outputs is to calibrate them by fitting a sigmoid curve to a held-out set [116]. Nevertheless, since such an approach requires a large amount of labeled data, it is inapplicable for our purpose. Another successful strategy is to assign the class that wins most pairwise comparisons [42], also referred as *max-wins* rule.

To adapt this strategy for multi-task metric learning, we assume that the positive

samples for task t coincidence with the class label $\mathbf{x}_i : y_i = t$. Then the combination rule

$$\begin{aligned} &\arg\max_t(\mathbf{x}_i) = \\ &\arg\max_t \sum_{u \neq t} \left[\mathrm{I}\left(\min_{j \in \mathcal{I}_t \wedge y_j = t} d_t^2(\mathbf{x}_i, \mathbf{x}_j) \leq \min_{k \in \mathcal{I}_u \wedge y_k = u} d_t^2(\mathbf{x}_i, \mathbf{x}_k) \right) \right. \\ &\qquad \left. + \mathrm{I}\left(\min_{j \in \mathcal{I}_t \wedge y_j = t} d_u^2(\mathbf{x}_i, \mathbf{x}_j) \leq \min_{k \in \mathcal{I}_u \wedge y_k = u} d_u^2(\mathbf{x}_i, \mathbf{x}_k) \right) \right] \end{aligned} \tag{5.5}$$

checks if the minimum distance of a given test sample $\mathbf{x}_i$ to class t is smaller than to class u. The indicator function

$$\mathrm{I}(x) = \begin{cases} 1 \text{ if } x \text{ is true} \\ \\ 0 \text{ otherwise} \end{cases} \tag{5.6}$$

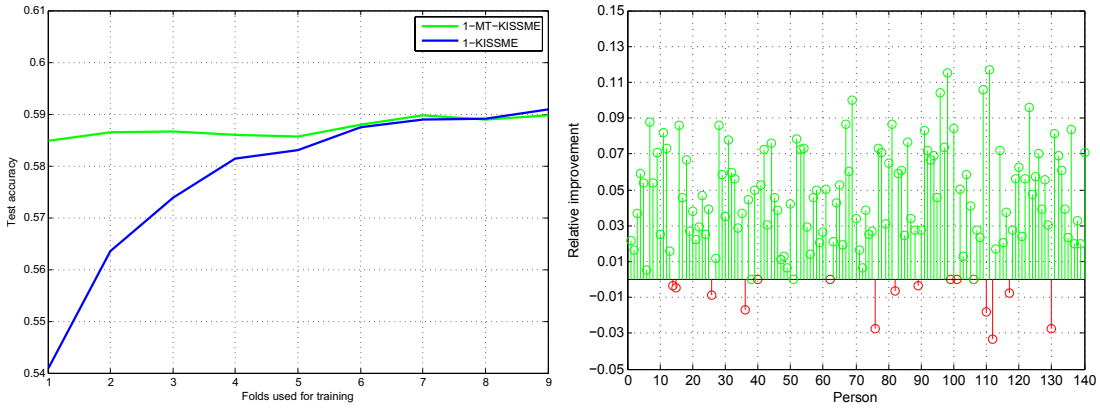
scores for class t if this is true. This comparison is done with the individual distance metric of task t. Further, we also compare the distances under the complementary distance metric of task u. The basic idea is that if class t scores even under that metric it is an indicator for class t. Intuitively, the final decision is for the class that wins most pairwise comparisons.

5.3 Experiments and Evaluations

In the following, we demonstrate the performance of our method on the Public Figures Face Database (PubFig) [82]. For the intended face identification benchmark we organize the data similar to the existing verification protocol in 10 folds for cross-validation. Therefore, we split the images of each individual into 10 disjoint sets. The goals of our experiments are twofold. First, in Section 5.3.1 we show that multi-task learning allows us to successfully exploit additional data with anonymous pairwise labels for face identification. Next, in Section 5.3.2 we show that multi-task learning of person specific metrics boosts the performance for face identification. In particular, we show that the power lies in the combination of multi-task learning and the person specific metrics, as it is not sufficient to learn them off-the-shelf. Further, we compare our results to standard metric learning and related multi-task learning approaches.

5.3.1 Inducing Knowledge from Anonymous Face Pairs to Face Identification

First, we show that multi-task learning allows us to transfer general knowledge about face similarity from anonymous face pairs to face identification. In order to enable a meaningful transfer of knowledge hereby multi-task learning faces the problem of different label sets. We test a multi-task learning scenario with two learning tasks, one with pairwise equivalence labels for the face pairs and one with class labels for face identification. The goal is to show that the additional anonymous face pairs help to improve the face identification performance. We sample the pairs randomly of the predefined development split of the dataset, containing 60 people. For the identification task we use the evaluation set, containing 140 people. Thus, we ensure that the subjects for the tasks are mutually exclusive. For a given test sample we perform k-NN classification using a single metric to the 140 classes. Using different values for k revealed that there is no significant performance change, although simple nearest neighbor assignment leads to the best performance. Thus, we stick to a simple nearest neighbor assignment.



(a) (b)

Figure 5.1: **Benefiting from additional pairwise labels for face identification on the PubFig dataset**: (a) k-NN classification accuracy of KISSME multi-task vs. standard single-task learning in relation to the amount of training data; (b) relative performance change per person from single-task to multi-task learning after using one fold for training. Green indicates positive induction while red indicates a negative induction.

In Figure 5.1 (a) we plot the face identification performance in relation to amount of data used to train the metric. Testing is done on a held-out set via 10 fold cross-validation. In each step we increase the number of folds used to train the identification task by one. As expected, the distance metric trained via multi-task learning (1-MT-KISSME) yields reasonable results right from the beginning. Obviously, it is able to reuse knowledge of

the anonymous face pairs. In contrast, the distance metric trained without the additional pairwise labels (1-KISSME) needs by far more data to reach the same performance. In Figure 5.1 (b), we compare the relative performance change per person from standard single-task learning to multi-task learning, after one training fold. In most cases an improvement can be obtained.

5.3.2 Person specific Metric Learning

Second, we demonstrate the performance of our MTL method to learn person specific distance metrics. To show the merit of our method we compare it to recent MTL methods [34, 108] and also benchmark to multi-class support vector machines [14, 21]. We report the face identification performance in a refusal to predict style. Therefore, we rank and threshold the classifier scores. In that sense, recall means the percentage of samples which have a higher score than the current threshold and thus are labeled. Precision means the ratio of correctly labeled samples.

In Figure 5.2 (a) we compare, as a sanity check, the performance of estimating person specific metrics via multi-task vs. single-task learning. The MTL method outperforms the single-task learning over most levels of recall. At full recall the performance difference is about 4.5%. The main advantage of our MTL method is revealed if we compare the recognition accuracy per person. With multi-task learning we reach a person accuracy of 63.10% while single-task reaches only 54.08%. Thus, it is favorable to learn person specific metrics multi-task. In Figure 5.2 (b) we compare the relative performance change per person. Only for a small number of classes the performance drops slightly while for the vast number the performance increases.

Next, in Figure 5.2 (c) we benchmark to recent MTL methods, MT-LMNN [108] and MT-SVM [34]. Both methods are not really able to capitalize on the synergies of the face identification task. Both methods are outperformed by MT-KISSME over all levels of recall. At full recall the respective performance gain compared to MT-LMNN is 12.4%, compared to MT-SVM 8%. In Figure 5.2 (d) we plot the relative performance change on person level compared to MT-SVM. Hence, our method is able also to compete with two recent MTL approaches. Compared to the MT-SVM one advantage may be that MT-KISSME operates in the space of pairwise differences, which eases meaningful transfer of knowledge between the learning tasks. Further, compared to both competing MTL methods MT-KISSME is able to gain information from pairwise labels.

Finally, in Figure 5.2 (e) we benchmark our method to multi-class support vector machines. Particularly, the method of Crammer and Singer [21] has shown recent success

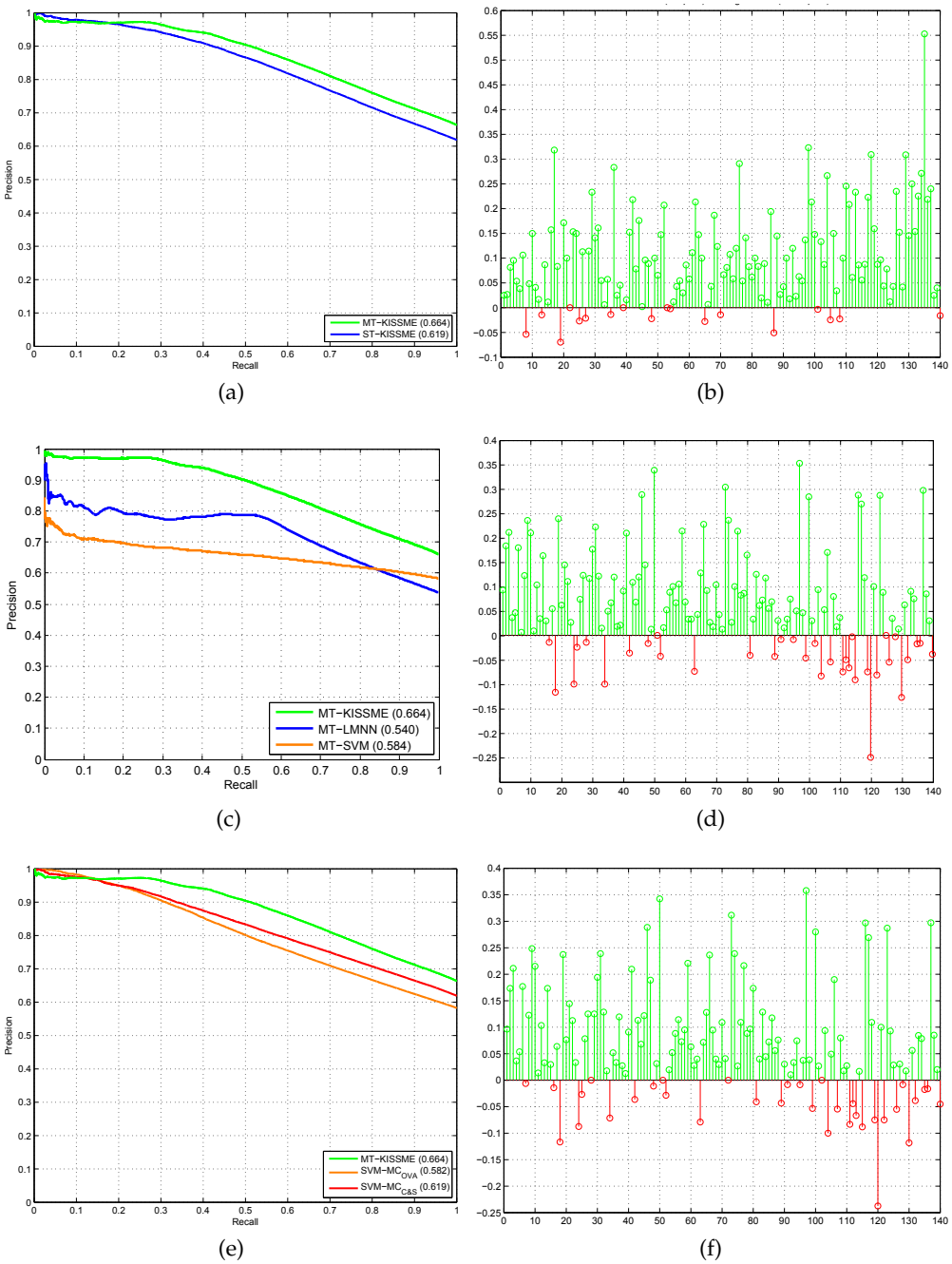


Figure 5.2: **PubFig face identification benchmark**. Comparison of the proposed method (MT-KISSME) to (a) single-task learning, (c) to other MTL methods, and (e) to SVMs. Numbers in parentheses denote the precision of the respective method at full recall. Right row, (b),(d),(f), compares the accuracy per person of the best performing competing method of the left plot to MT-KISSME.

also compared to metric learning methods [145]. The standard multi-class one-vs-all SVM reaches with 58.4% at full recall about the same performance as the MT-SVM. The method of Crammer and Singer [21] beats this by 3.7%. This may be accounted to the fact that it attempts to solve a single multi-class optimization problem that is better suited for unbalanced datasets. Nevertheless, MT-KISSME outperforms the one-vs-all method by 8.5% and the method of Crammer and Singer by 4.5%.

5.4 Conclusion

In this chapter we presented a synergistic approach to exploit shared common as well as person specific information for face recognition. By extending Keep It Simple and Straightforward Metric learning (KISSME) we developed a multi-task learning method that is able to learn from just equivalence constraints, thus, enabling label-incompatible learning. Overall, we get a conceptually simple but very effective model, which is scalable to large datasets. Further, we showed that learning person specific metrics boosts the performance for face identification. In particular, we revealed that the power lies in the combination of multi-task learning and person specific metrics, as it is not sufficient to learn the metrics decoupled. To show the merits of our method we conducted two experiments on the challenging large-scale PubFig face benchmark. We are able to match or slightly outperform recent multi-task learning methods and also multi-class support vector machines.

6 Discriminative Metric and Prototype Learning for Face Recognition

In this Chapter, we revisit the topic of evaluation complexity in Mahalanobis metric learning. The complexity scales linearly with the size of the dataset. This is especially cumbersome on large scale or for real-time applications with limited time budget. To alleviate this problem in this chapter we propose to represent the dataset by a fixed number of discriminative prototypes. In particular, we introduce a new method that jointly chooses the positioning of prototypes and also optimizes the Mahalanobis distance metric with respect to these. We show that choosing the positioning of the prototypes and learning the metric in parallel leads to a drastically reduced evaluation effort while maintaining the discriminative essence of the original dataset. Moreover, for most problems our method performing k-nearest prototype (k-NP) classification on the condensed dataset leads to even better generalization compared to k-NN classification using all data.

6.1 Introduction

The large-scale nature of computer vision applications poses several challenges and opportunities to the class of Mahalanobis metric learning algorithms. For instance one can take the chance and learn a sophisticated distance metric that captures the structure of the dataset, or learn multiple local metrics that better adapt to the intrinsic characteris-

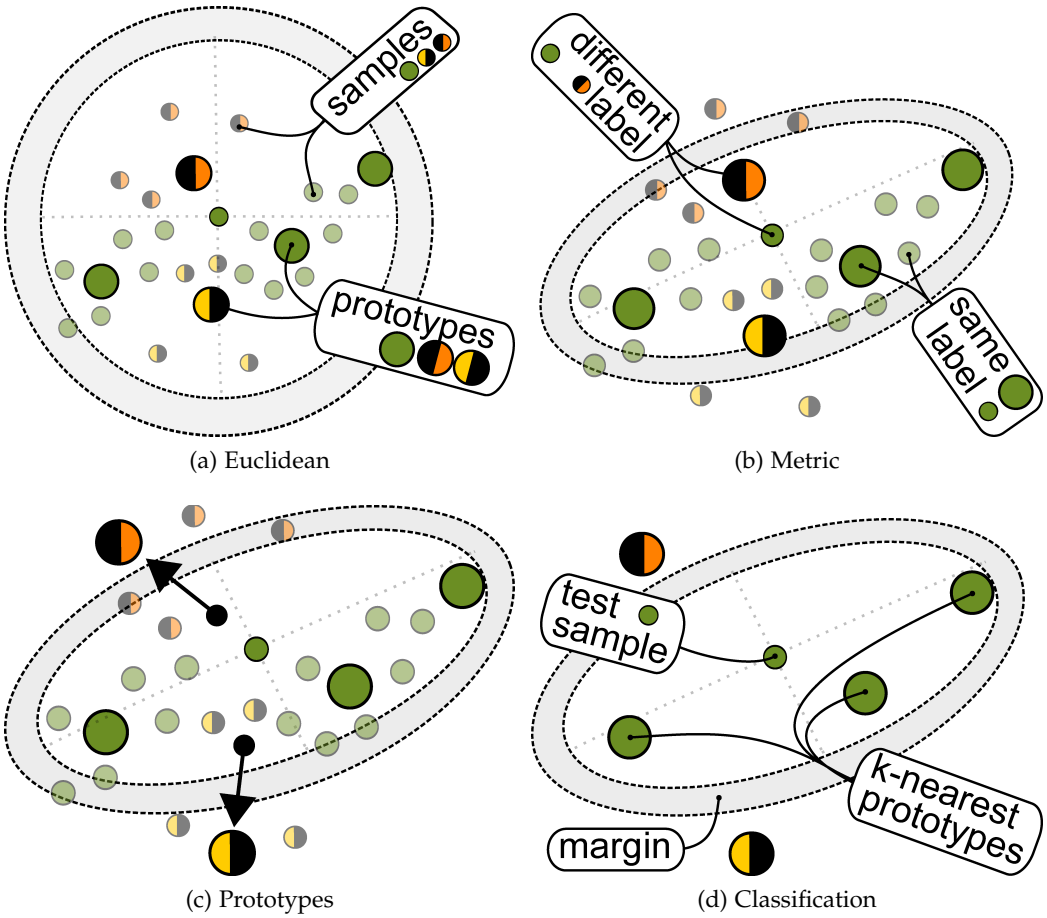
tics of the feature space. On larger datasets this usually leads to lower error rates [143]. In contrast, this is challenged by the computational burden in training and the needed label effort. To reduce the required level of supervision, algorithms such as [27, 75] have been introduced that are able to learn from pairwise labels. Others tackle the problem of time complexity in learning by special optimization techniques [27, 143]. Ultimately, for many applications the time complexity in learning is not too critical. Nevertheless, one important aspect that is often neglected is the computational burden at test time.

One inherent drawback of Mahalanobis metric learning based methods is that the k-NN search in high-dimensional spaces is time-consuming, even on moderate sized datasets. For real-time applications with limited time budget this is even more critical. To alleviate this problem, different solutions have been proposed that focus on low dimensional embeddings. The resulting space should enable efficient retrieval and reflect the characteristics of the learned metric. For instance, one can accelerate nearest neighbor search by performing a low dimensional Hamming embedding. This can be done by applying locality sensitive hash functions directly [63] or on kernelized data [81]. Another strategy is to learn a low-rank Mahalanobis distance metric [143] that performs dimensionality reduction. Nevertheless, a too coarse approximation diminishes at least some of the benefits of learning a metric. Further, special data structures as metric ball trees have been introduced to speed up nearest neighbor search. Unfortunately, there is no significant time gain for high dimensional spaces.

Another technique is to reduce the number of training samples and introduce sparsity in the samples. Ideally, one maintains only a relatively small set of representative prototypes which capture the discriminative essence of the dataset. This condensation can be either seen as drawback, as it's likely to loose classification power, or taken as opportunity. In fact, the theoretical findings of Crammer et al. [22] provide even evidence that prototype-based methods can be more accurate than nearest neighbor classification. One reason might be that the condensation reduces overfitting. Choosing the positioning of the prototypes wisely can lead to a drastically reduced effort while maintaining the discriminative power of the original dataset.

Addressing challenges and opportunities of larger data sets and applications with limited time budget, we propose to bridge the gap between Mahalanobis metric learning and discriminative prototype learning as illustrated in Figure 6.1. In particular, we are interested in joint optimization of the distance metric with respect to the discriminative prototypes and also of the positioning of the prototypes. This combination enables us to drastically reduce the computational effort while maintaining accuracy. Furthermore,

we provide evidence that in most cases the proposed Discriminative Metric and Prototype Learning (DMPL) method generalizes even better to unseen data compared to recent Mahalanobis metric k-NN classifiers.



(a) Euclidean (b) Metric

(c) Prototypes (d) Classification

Figure 6.1: **Condensating a dataset by discriminative prototypes:** Learning the distance metric (b) and the positioning of the prototypes (c) in parallel allows to drastically reduce the evaluation effort while maintaining full discriminative power. With our method k-nearest prototype classification results improve even over k-NN classification for most problems (d).

The rest of this chapter is structured as follows: In Section 6.2 we give a brief overview of related work in the field of Mahalanobis metric and prototype learning. Succeeding, in Section 6.3 we describe our Discriminative Metric and Prototype Learning (DMPL) method as an alternating optimization problem. Detailed experiments on standard machine learning datasets and on the challenging PubFig [82] face recognition benchmark are provided in Section 6.4.

6.2 Related Work

Compared to other classification models Mahalanobis Metric Learning provides with k-NN search not only reasonable results but is also inherently multi-class and directly interpretable, based on the assigned neighbors. Several different methods (e.g., [145], [27], or [50]) have been proposed that show good results for many real world problems. An overview is given in Chapter 4.

A particular successful instance of this class of algorithms is the approach of Weinberger et al. [143, 145], which aims at improving k-NN classification by exploiting the local structure of the data. It mimics the non-continuous and non-differentiable classification error of the k-NN scheme by a convex loss function. The main idea bases on two simple intuitions. First, the k-NNs of each sample that share the class label (target neighbors) should move close to each other. Second, no differently labeled sample should invade this local k-NN perimeter plus a safety margin. This safety margin allows for focusing on samples near the local k-NN decision boundary and ensures that the model is robust to small amounts of noise.

Prototype methods such as learning vector quantization (LVQ) share some of the favorable characteristics of Mahalanobis metric learning. They deliver intuitive, interpretable classifiers based on the representation of classes by prototypes. The seminal work of Kohonen [73] updates prototypes iteratively based on a clever heuristic. A data point attracts the closest prototype in its direction if it matches the class label. Vice-versa it is repelled if it shows a different class label. Various extensions have been proposed that modify the original update heuristic. For instance, updating both the closest matching and non-matching prototype or restricting the updates close to the decision boundary. Otherwise LVQ can show divergent behavior.

Seo and Obermayer [125] explicitly avoid the divergent behavior by an underlying optimization problem. The main idea is to treat the prototypes as unit size, isotropic Gaussians and maximize the likelihood ratio of the probability of correct assignment versus the total probability in the Gaussian mixture model. The resulting robust learning scheme updates only prototypes close to the decision boundary by incorrectly classified samples. Also, the work of Crammer et al. [22] derives a loss-based algorithm for prototype positioning based on the maximal margin principle. LVQ arises as special case of this algorithm. Remarkably, the authors provide evidence that prototype methods follow max-margin principles. However, for this class of algorithms classification is solely based on a predefined metric.

Therefore, to alleviate this issue variants have been proposed that learn some parameters of the distance function. For instance, Parametric Nearest Neighbor (P-NN) and its ensemble extension (EP-NN) [163] learn weights on the Euclidean distance function. Bonilla and Robles-Kelly [7] propose a probabilistic discriminative generalization of vector quantization. They jointly learn discriminative weights on soft-assignments to prototypes and further the prototype positions. Nevertheless, as these approaches learn only restricted parameters of the distance function these may miss different scalings or correlations of the features.

In contrast to these previous works we want to exploit a more general metric structure. In particular, we are interested in improving runtime and classification power by combining the favorable characteristics of Mahalanobis metric learning and prototype methods. Our method integrates a large margin formulation with focus on samples close to the decision boundary. Further, it naturally integrates with k-NN, which may be in some situations the favorable choice over nearest neighbor assignment.

6.3 Discriminative Mahalanobis Metric and Prototype Learning

In the following, we derive a new formulation that jointly chooses the positioning of prototypes and also optimizes the distance metric with respect to these. This allows us to exploit the global structure of the data (via metric) and to drastically reduce the computational effort during evaluation (via prototypes). Finally, this reduces evaluation time and improves k-NP classification.

6.3.1 Problem Formulation

For the following discussion let us introduce a training set $\mathcal{X} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, with N samples $\mathbf{x}_i \in \mathbb{R}^D$ and corresponding labels $y_i \in \{1, 2, \ldots, C\}$. Let $\mathcal{Z} = \{(\mathbf{z}_1, y_1), \ldots, (\mathbf{z}_K, y_K)\}$ correspond to a set of K prototypes. Then, the squared Mahalanobis distance between a data sample $\mathbf{x}_i$ and a prototype $\mathbf{z}_k$ is defined as

$$d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{z}_k) = (\mathbf{x}_i - \mathbf{z}_k)^\top \mathbf{M} (\mathbf{x}_i - \mathbf{z}_k), \tag{6.1}$$

where $\mathbf{M} \succeq 0$ is a symmetric positive semidefinite matrix. Our goal is to estimate the metric matrix $\mathbf{M}$ and the prototypes $\{\mathbf{z}_k\}_1^K$ in parallel. The idea to fuse metric and prototype learning is general and can be adapted to various Mahalanobis metric learning methods. In particular, we adopt ideas of LMNN [145] and locally establish a perimeter

surrounding each data sample. Prototypes with different class label should not invade this perimeter plus a safety margin. This behavior can be realized by minimizing the following energy:

$$\epsilon^i(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^K) = (1-\mu) \sum_{j \rightsquigarrow i} d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{z}_j) \\ +\mu \sum_{j \rightsquigarrow i} \sum_l (1 - y_{il}) \xi_{ijl}(\mathbf{M}), \tag{6.2}$$

where $j \rightsquigarrow i$ indicates that $\mathbf{z}_j$ is a target prototype of sample $\mathbf{x}_i$ and $\mu \in [0,1]$ is a weighting factor. The first term attracts target prototypes $\mathbf{z}_j$ while the second term emits a repelling force on differently labeled prototypes $\mathbf{z}_l$ that invade the perimeter. We refer to these invaders as impostor prototypes. Note that the pairwise label y_{il} is zero if $y_i \neq y_l$ and one otherwise.

If a prototype invades the local perimeter plus margin is monitored by

$$\xi_{ijl}(\mathbf{M}) = \left[1 + d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{z}_j) - d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{z}_l)\right]_+, \tag{6.3}$$

where $[a]_+ = \max(a, 0)$ is the hinge loss. It activates only if the prototype is closer to the sample $\mathbf{x}_i$ than the target prototype $\mathbf{z}_j$ plus margin. Finally, the overall energy function is a sum of the local contributions:

$$\epsilon(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^K) = \sum_{i=1}^N \epsilon^i(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^K). \tag{6.4}$$

In order to minimize the energy function we use an alternating optimization based on gradient descent w.r.t. the prototype positions $\{\mathbf{z}_k\}_{k=1}^K$ and the distance metric $\mathbf{M}$. At each iteration we take a sufficiently small gradient step and monitor boundary conditions as $\mathbf{M} \succeq 0$. In the following, we derive alternating update rules in terms of prototypes and the metric matrix.

6.3.2 Learning Prototypes

First, we derive the update rules w.r.t. to the prototype locations. As the particular role of an individual prototype as target or impostor is ambiguous on global scope we express the gradient

$$\frac{\partial \epsilon(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^K)}{\partial \mathbf{z}_k} = \sum_{i=1}^{N} \frac{\partial \epsilon^i(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^K)}{\partial \mathbf{z}_k} \tag{6.5}$$

as sum over the (unambiguous) gradient contribution of each data sample $\mathbf{x}_i$ on the respective prototype $\mathbf{z}_k$. A prototype can be a target neighbor ($k = j$), an impostor ($k = l$), or simply irrelevant as too far away. Therefore, we specify the gradient contribution of a sample on a prototype as follows:

$$\frac{\partial \epsilon^i(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^K)}{\partial \mathbf{z}_k} = \begin{cases} \frac{\partial \epsilon^i(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^K)}{\partial \mathbf{z}_j} & \text{if } k{=}j \\ \frac{\partial \epsilon^i(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^K)}{\partial \mathbf{z}_l} & \text{if } k{=}l \\ \mathbf{0} & \text{otherwise.} \end{cases} \tag{6.6}$$

Taking into account that

$$\frac{\partial d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{z}_k)}{\partial \mathbf{z}_k} = -2(\mathbf{x}_i - \mathbf{z}_k)^\top \mathbf{M} \tag{6.7}$$

we can re-write the gradients defined in Eq. (6.6). Substituting Eq. (6.7) into Eq. (6.6) for a target prototype ($k = j$, attraction force) we get

$$\begin{aligned} \frac{\partial \epsilon^i(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^K)}{\partial \mathbf{z}_j} = (1-\mu) \sum_{j \rightsquigarrow i} -2(\mathbf{x}_i - \mathbf{z}_j)^\top \mathbf{M} \\ + \mu \sum_{l \text{ s.t. } (i,j,l) \in \mathcal{I}} -2(\mathbf{x}_i - \mathbf{z}_j)^\top \mathbf{M}, \end{aligned} \tag{6.8}$$

where $\mathcal{I} = \{(i, j, l) | \xi_{ijl} > 0\}$ is the set of active sample-impostor triplets. Similarly, we get

$$\frac{\partial \epsilon^i(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^K)}{\partial \mathbf{z}_l} = -\mu \sum_{l \text{ s.t. } (i,j,l) \in \mathcal{I}} -2(\mathbf{x}_i - \mathbf{z}_l)^\top \mathbf{M} \tag{6.9}$$

for an impostor prototype ($k = l$, repelling force). Finally, we can specify the iterative update rule at iteration t for the prototypes as

$$\mathbf{z}_k^{(t+1)} = \mathbf{z}_k^{(t)} - \eta \frac{\partial \epsilon \left(\mathbf{M}^{(t)}, \left\{ \mathbf{z}_k^{(t)} \right\}_{k=1}^{K} \right)}{\partial \mathbf{z}_k^{(t)}}, \tag{6.10}$$

where η denotes the learning rate. Reasonable choices for the initial prototypes are all variants of clustering algorithms such as k-means or using training samples as initialization. We emphasize that compared to the update rules of LVQ or P-NN [163] our formulation is more general and natively integrates in k-NP classification. Further, it accounts for different scalings and correlations of the feature space.

6.3.3 Distance Metric Learning

Next, we derive the update rule w.r.t. the distance metric in terms of the local contribution of each sample to its neighboring prototypes. Hence, the derivative can be expressed as

$$\frac{\partial \epsilon(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^{K})}{\partial \mathbf{M}} = \sum_{i=1}^{N} \frac{\partial \epsilon^i(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^{K})}{\partial \mathbf{M}}. \tag{6.11}$$

To estimate $\mathbf{M}$, gradient descent is performed along the gradient defined by the set of active sample-impostor triplets $\mathcal{I}$. We can write the gradient as

$$\begin{aligned} \frac{\partial \epsilon^i \left(\mathbf{M}, \{\mathbf{z}_k\}_{k=1}^{K} \right)}{\partial \mathbf{M}} &= (1-\mu) \sum_{j \rightsquigarrow i} \mathbf{C}_{ij} \\ &\quad + \mu \sum_{(j,l) \text{ s.t. } (i,j,l) \in \mathcal{I}} \left(\mathbf{C}_{ij} - \mathbf{C}_{il} \right) , \end{aligned} \tag{6.12}$$

where $\mathbf{C}_{ik}$ denotes the outer product of pairwise differences. This is the gradient of the distance function $d^2_{\mathbf{M}}$:

$$\mathbf{C}_{ik} = (\mathbf{x}_i - \mathbf{z}_k)(\mathbf{x}_i - \mathbf{z}_k)^\top = \frac{\partial d^2_{\mathbf{M}}(\mathbf{x}_i, \mathbf{z}_k)}{\partial \mathbf{M}}. \tag{6.13}$$

Eq. (6.12) conceptually tries to strengthen the correlation between the sample and

target prototypes while weakening it between the sample and impostor prototypes. Finally, we can specify the iterative update rule at iteration t as

$$\mathbf{M}^{(t+1)} = \mathbf{M}^{(t)} - \eta \frac{\partial \epsilon \left(\mathbf{M}^{(t)}, \left\{ \mathbf{z}_k^{(t)} \right\}_{k=1}^{K} \right)}{\partial \mathbf{M}^{(t)}}. \tag{6.14}$$

Initially, we start with the Euclidean distance ($\mathbf{M} = \mathbf{I}$). Note that after each iteration we check if $\mathbf{M}$ induces a valid pseudo-metric. To satisfy metric conditions we use a projection operator similar to [52] by back-projecting the current solution on the cone of positive semidefinite (p.s.d.) matrices.

6.4 Experiments

To show the broad applicability of our method we conduct experiments on various standard benchmarks with rather diverse characteristics. Further, we study the problem of large-scale face recognition in unconstrained environments on the Public Figures Face Database [82]. The goals of our experiments are twofold. First, we want to show that with a drastically reduced prototype set we get comparable or even better results than related work. Second, we want to prove that we are more efficient in evaluation. This is clearly beneficial for large scale or real-time applications.

6.4.1 Machine Learning Databases

In the following, we benchmark our proposed method on MNIST [62], USPS [60], LETTER [62] and CHARS74k [10]. First, we give a brief overview of the databases. Second, we compare our method Discriminative Metric and Prototype Learning (DMPL) to several baselines such as learning only the prototypes or the distance metric. Finally, we compare the performance related to the evaluation complexity between our method and state-of-the-art approaches.

The MNIST database [62] of hand written digits contains in total 70,000 images in one train-test split. 60,000 samples are used for training and 10,000 for testing. The images have a resolution of 28×28 pixels and are in grayscale. Similarly USPS [60] contain grayscale images of hand written digits with a resolution of 16×16 pixels. 7,291 images are organized for training and 2007 images for testing.

In contrast, the LETTER [62] database contains a large number of synthesized images showing one of the 26 capital letters of the English alphabet. The images are represented

as 16 dimensional feature vector which describes statistical moments and edge counts.

Chars74K [10] contains a large mixed set of natural and synthesized characters. The images comprise either one of the 26 capital or lowercase letters and digits. Thus, the dataset features 62 classes. 7,705 characters are cropped of natural images, 3,410 are hand drawn and 62,992 are synthesized. Similar to [163] we apply a color space conversion to grayscale and resizes each image to 8×8 pixels. Further, the database is split into one train-test set where 7400 samples are organized for testing and the rest for training.

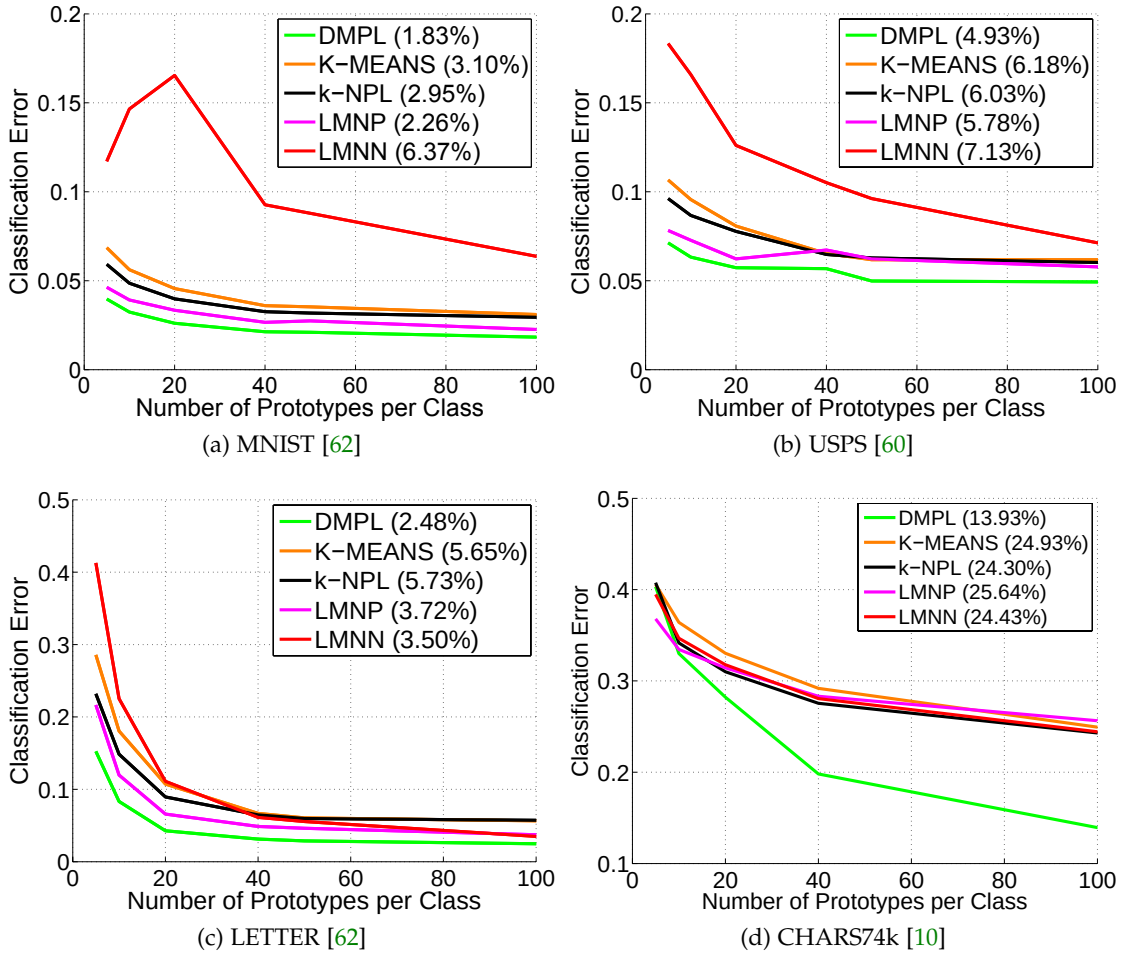
For MNIST we perform a dimensionality reduction of the raw features by PCA to a 164 dimensional subspace, to make the learning more tractable. For all other databases we use the raw data without calculating any complex features, in order to get a fair comparison.

In Figure 6.2 we compare our method (DMPL) to baseline approaches on the respective benchmarks. Therefore, we plot the classification error in relation to the number of prototypes. In particular, we report the following results: The direct assignment of the k-means cluster label, thus ignoring discriminative information in learning at all. Second, we compare to training standard LMNN on the prototypes. Here, the main goal is to stress the difference between optimizing for k-NP classification or k-NN classification. Third, we compare to only tuning the positioning of the prototypes, referred as k-Nearest Prototype Learning (kNPL). Finally, we optimize only the distance metric assuming fixed prototypes, referred as Large Margin Nearest Prototype (LMNP) learning.

For the following discussion we focus on the respective results on MNIST visualized in Figure 6.2 (a), although the relative results are comparable on the different datasets. As expected LMNN and k-means perform initially worse than the prototype based methods. In case of LMNN the performance gap is rather big. By increasing the number of prototypes the gap gets smaller as the k-means centroids behave more similar to the actual data samples. However, ultimately for MNIST a performance gap of about 4.5% remains. Comparing LMNN to LMNP reveals that it is beneficial to optimize the distance metric in respect to the prototypes. The drop in terms of classification error is about 4% with 100 prototypes. Interestingly, k-means is more competitive compared to LMNN right from the beginning. Nevertheless, it is outperformed by both, kNPL and also LMNP. Comparing the baselines to our discriminative metric and prototype learning (DMPL) method reveals that the power lies in the combination of distance metric learning and prototype methods. DMPL outperforms LMNN by roughly 4.5% and k-means by 1.3%. As MNIST is a rather competitive dataset this is a reasonable performance gain.

Methods	MNIST	USPS	LETTER	Chars74K
Protoype Methods				
DMPL $_{1\text{-NP}}$ (40 prototypes)	2.13	5.68	3.13	19.81
DMPL $_{1\text{-NP}}$ (100 prototypes)	1.83	4.93	2.48	13.99
DMPL $_{3\text{-NP}}$ (200 prototypes)	1.66	4.83	2.50	14.05
Parametric NN (40 prototypes) [163]	3.13	7.87	6.95	29.46
Ensemble of P-NN (800 prototypes) [163]	1.65	4.88	2.90	19.53
Nearest Neighbors				
Nearest Neighbor (1-NN, 3-NN)	2.92 - 3.09	4.88 - 5.08	4.30 - 4.35	17.97 - 19.99
LMNN $_{1\text{-NN}}$ [143, 145]	2.09	4.73	2.93	17.07
LMNN $_{3\text{-NN}}$ [143, 145]	1.70	4.78	3.54	19.08
ITML $_{1\text{-NN}}$ [27]	2.17	5.23	4.75	17.00
ITML $_{3\text{-NN}}$ [27]	2.02	5.03	4.68	18.54
LDML $_{1\text{-NN}}$ [50]	4.04	9.12	11.25	18.62
LDML $_{3\text{-NN}}$ [50]	3.59	8.27	10.35	20.32
KISSME $_{1\text{-NN}}$ [75]	2.66	6.43	2.83	15.77
KISSME $_{3\text{-NN}}$ [75]	2.36	6.38	2.73	18.64
Support Vector Machines				
Linear [35]	8.18	8.32	23.63	35.08
Linear + EFM (Intersection kernel) [136]	9.11	8.12	8.22	29.08
Kernel [14, 21, 134]	1.36 - 1.44	4.24 - 4.58	2.12 - 2.42	16.86
SVM-KNN [161]	1.66	4.29	-	-
LA-RANK [8]	1.41	4.25	2.80	-
Locally linear classifiers				
Lin. SVM + LCC (4,096 anchor p.) [140, 159, 160]	1.64 - 2.28	4.38	4.12	20.88
Lin. SVM + DCN (L1 = 64,L2 = 512) [88]	1.51	-	-	-
Local Linear SVM (100 anchor p.) [83]	1.85	5.78	5.32	25.11
LIB-LLSVM + OCC [162]	1.61	3.94	6.85	18.72
ALH [154]	2.15	4.19	2.95	16.26
Locality-sensitive hashing				
KLSH (10,000 kernel samples) [80, 81]	6.15	5.68	7.38	88.76
Fast Image Search for Learned Metrics (ϵ = 0.6) [63]	5.51	5.53	8.55	-
WTA [153]	4.59	9.92	8.03	15.64
Spectral Hashing [146]	4.25	8.72	7.42	26.03
Multidimensional Spectral Hashing [147]	5.27	13.35	33.67	-
Spherical Hashing [51]	2.22	5.13	19.00	16.65
LSH [44]	2.92	5.63	5.03	20.01
Others				
BPM+MRG [142]	-	6.10	10.50	-

Table 6.1: **Comparison of classification error rates on MNIST, USPS, LETTER and Chars74k**. Our method (denoted DMPL) outperforms several state-of-the-art approaches while being more efficient. With 200 prototypes we improve even over LMNN which requires the full dataset for classification. The top performing method of each category is highlighted. K-NP refers to the number of prototypes used for classification.



(a) MNIST [62] (b) USPS [60]

(c) LETTER [62] (d) CHARS74k [10]

Figure 6.2: **Benchmark of our proposed method (DMPL) to baseline approaches on MNIST, USPS, LETTER and Chars74k**: Compared to the main paper the figure shows also the results for USPS and LETTER. The baselines are K-Nearest Prototype Learning (kNPL, Eq. 10), Large Margin Nearest Prototype (LMNP, Eq. 14) learning, k-means and plain LMNN [144]. We compare the 1-NP classification error in relation to the number of prototypes per class. The numbers in parenthesis denote the classification error with 100 prototypes per class.

	DMPL vs ...	**LMNN** [145]	**SVM** [35]	**LLC** [140, 159, 160]	**LL-SVM** [83]	**P-NN** [163]	**EP-NN** [163]	**FSLM** [63]
100	Rel. Compl.	60	$\frac{1}{100}$	40	1	$\frac{1}{2.5}$	8	1.94
	Rel. Error	+ 0.13%	- 6.35%	- 0.45% to + 0.19%	- 0.02%	- 1.30%	+ 0.18%	- 3.68%
200	Rel. Compl.	30	$\frac{1}{200}$	20	$\frac{1}{2}$	$\frac{1}{5}$	4	0.97
	Rel. Error	- 0.05%	- 6.52%	- 0.62% to + 0.02%	- 0.19%	- 1.47%	+ 0.01%	- 3.86%

Table 6.2: **Relative comparison of the evaluation complexity and the difference of classification errors** using MNIST [62]. We compare DMPL 1-NP with 100 and DMPL 3-NP with 200 prototypes vs related state-of-the-art. For instance, comparing DMPL 3-NP to LMNN with 200 prototypes DMPL is 30 times faster and has a 0.05 percentage points lower classification error.

Next, in Table 6.1 we benchmark our method to various state-of-the-art approaches. These include recent local linear methods, support vector machines, nearest neighbor and prototype based methods. Further, Table 6.2 gives a relative comparison of the evaluation complexity of selected methods and their classification error on MNIST. The performance comparison between local linear methods as LL-SVM [83] and prototype methods is especially interesting as a nearest prototype classifier is essentially a local linear classifier. The decision boundaries are perpendicular to the connection lines between the prototypes.

The first important finding is that DMPL outperforms methods that either use a predefined or learned metric even though being more efficient. Compared to vanilla k-NN search with plain Euclidean distance the main advantage is the ability to model different scalings and correlations of the feature space. Further, using less prototypes DMPL improves over a recent prototype based method [163] that learns only a relevance weighting of the features. One advantage is that DMPL is able to account for different correlations of the feature space. Compared to LMNN the flexibility remains to discriminatively adapt the positioning of the prototypes.

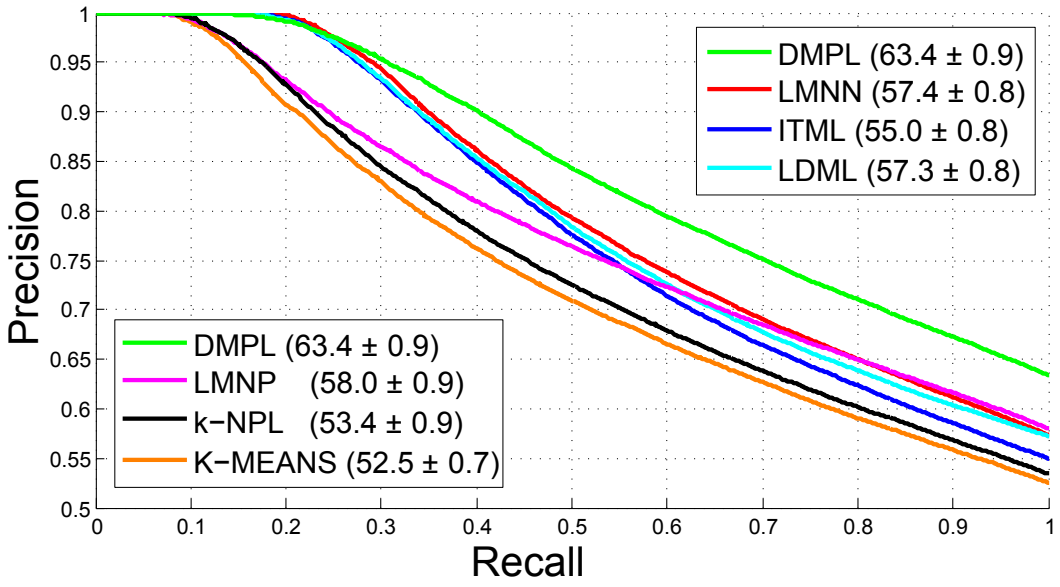
Second, like DMPL locality sensitive hashing based approaches focus on efficient retrieval. Nevertheless compared to our method they trade off classification power for efficiency. The results show that kernelized hashing needs a large number of kernel samples to obtain comparable results. Standard LSH approaches need to consider a large number of samples for exact search, diminishing at least some of the speed advantages.

Finally, compared to kernel SVMs our method is outperformed only slightly while being able to perform classification with a fixed time budget. For kernel SVMs it is known that the number of support vectors scale linearly with the size of the dataset. Local linear methods such as LL-SVM [83] bypass this issue. Interestingly, they share our computational costs. On MNIST LL-SVM matches our performance, however on USPS and LETTER we are able to improve over LL-SVM. Only local linear methods using a much larger number of anchor points are able to improve over our method.

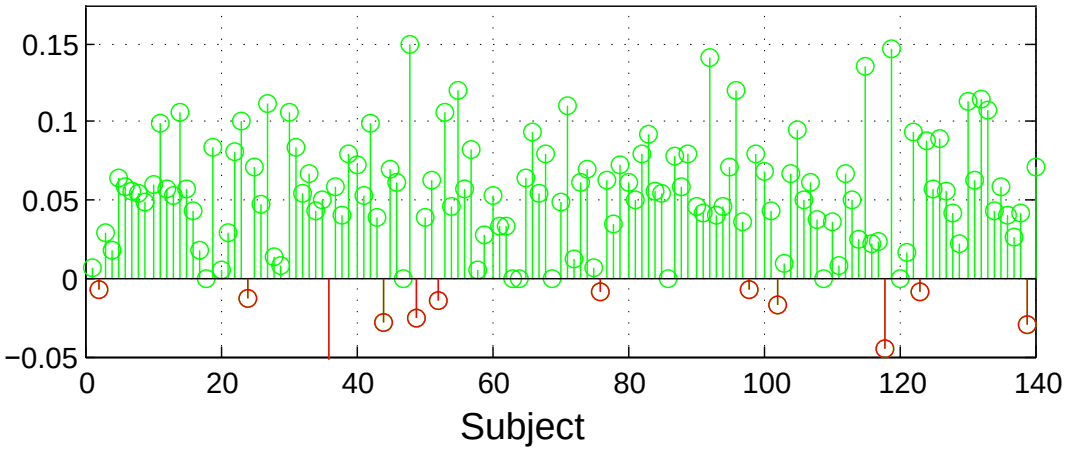
Recapitulating the different results and relating them to the evaluation complexity of related works it reveals that we get competitive results and are more efficient.

6.4.2 Public Figures Face Database

In the following, we demonstrate our method for face identification on the Public Figures Face Database (PubFig) [82]. To represent the faces we use the description of visual face traits [82]. They describe the presence or absence of 73 visual attributes, such as



(a)



(b)

Figure 6.3: **Face identification benchmark on the PubFig database** [82]: The data is organized in 10 non-overlapping folds for cross-validation. (a) Precision / Recall curves by ranking and thresholding classifier scores. Numbers in parenthesis denote the precision and std. dev. at full recall. (b) Difference of precision per person between DMPL and LMNN.

gender, race, hair color etc. Further, we apply a homogeneous χ^2 feature mapping [136]. For the face identification benchmark we organize the data similar to the existing verification protocol in 10 folds for cross-validation. Therefore, we split the images of each individual into 10 disjoint sets.

In Figure 6.3 (a) we benchmark our method using 100 prototypes per class to recent Mahalanobis metric learning methods. We report the face identification performance in

a refusal to predict style. In that sense, recall means the percentage of samples which have a higher classifier score than the current threshold. Precision means the ratio of correctly labeled samples.

In particular, we show that DMPL generalizes better than LMNN [145], ITML [27] or LDML [50] which require on the full training set for classification. At full recall the performance difference to LMNN is 6.00%. Further, comparing the results to the baseline approaches kNPL and LMNP reveals once more that the power lies in the combination of metric learning and prototype learning. In Figure 6.3 (b) we compare the relative change in classification accuracy per person between our method and LMNN. Only for a small number of classes the performance drops slightly while for the vast number the performance increases. Thus, there is no bias in terms of overrepresented classes. Intuitively, one interpretation is that the fixed number of prototypes helps to compensate for overfitting.

6.5 Conclusion

In this chapter we presented a novel method to condense a dataset to a fixed number of discriminative prototypes with application to face recognition. In particular, we jointly choose the positioning of prototypes and also optimize the Mahalanobis distance metric with respect to these. This leads to a drastically reduced effort at test time while maintaining the discriminative essence of the original dataset. Our method performing k-nearest prototype classification on the condensed dataset leads to even better generalization compared to k-NN classification. To show the merit of our method we conducted several experiments on various challenging large-scale benchmarks. On all benchmarks we are able to compare to or slightly outperform state-of-the-art approaches, while being more efficient at test time. On the Public Figures Face Database we even outperform metric learning approaches using by far more data.

7 Conclusion

In this thesis we addressed limitations of traditional learning strategies for real-world face recognition. In real-world situations imaging conditions as diversity in viewpoint, lighting, clutter or occlusion severely lower the recognition performance. A promising class of machine learning algorithms is Mahalanobis metric learning, which has recently demonstrated competitive results for a variety of face recognition tasks. Real-world face recognition more and more becomes a large-scale task, in a sense that image acquisition devices are omnipresent in our daily life. Thus, more and more photos are taken every day that need to be processed, where often a human face is the object of interest. As data grows several challenges and opportunities are posed to computational face recognition algorithms despite the recognition challenge. First, a main criterion for the applicability of machine learning algorithms is the scalability in terms of learning, evaluation costs and also the needed effort to obtain labels and annotations. Scanning web-scale data sets of images containing millions of faces call for efficient search and retrieval strategies, where efficient algorithms are clearly beneficial. Second, to benefit from the available data in learning face recognition models there is the need for sophisticated machine learning methods that exploit the special characteristics of faces.

7.1 Discussion

Addressing the limitations of traditional Mahalanobis metric learning approaches in this thesis we developed tools that enable single-image real-world face recognition on large-scale. In particular, we focused on scalable algorithms that allow for efficient training, evaluation and require less labeling effort.

To efficiently train face verification and identification models we introduced in Chap-

ter 4 our Keep It Simple and Straightforward Metric (KISSME) learning method. Typically, learning Mahalanobis metrics requires often to solve complex and thus computationally very expensive optimization problems. Further, for many scenarios it is infeasible or simply too laborious to specify fully supervised labels. Instead, it is easier to specify labels in form of equivalence constraints. Our proposed algorithm, KISSME, is a simple though effective strategy to learn a distance metric from equivalence constraints, based on a statistical inference perspective. In contrast to existing methods KISSME does not rely on complex optimization problems requiring computationally expensive iterations. We showed in extensive experiments and evaluations that KISSME is able to compete with the state-of-the-art in Mahalanobis metric learning for various tasks. However, as main benefit, it is orders of magnitudes faster in training than comparable methods. Of course for face recognition the recent state-of-the-art on LFW [58] provides better results but also requires considerably more domain knowledge, e.g., pose specific classifiers or an auxiliary identity set of faces.

To speed-up the evaluation for Mahalanobis metric learning for face recognition, we addressed the problem of efficient k-NN classification. In particular, we introduced two methods. First, in Chapter 4 we proposed a metric-based hashing strategy, allowing for both, efficient learning and evaluation. In fact, we showed that if the intrinsic structure of the data is exploited by the metric in a meaningful way, using hashing we can compact the feature representation still obtaining competitive results. Second, in Chapter 6 we proposed to represent the dataset by a fixed number of discriminative prototypes. In particular, we introduced a new method termed Discriminative Metric and Prototype Learning (DMPL) that jointly chooses the positioning of prototypes and also optimizes the Mahalanobis distance metric with respect to these. We showed that choosing the positioning of the prototypes and learning the metric leads to a drastically reduced effort while maintaining the discriminative essence of the original data. To show the merit of DMPL we conducted several experiments on various challenging benchmarks. We were able to compete with state-of-the-art local linear and prototype methods while being more efficient in evaluation. For face identification we even outperform metric learning approaches requiring the full training set for classification.

In Chapter 5, we addressed the problem, neglected by most face recognition approaches, that faces share strong visual similarities. This can be exploited when learning discriminative models. Hence, we proposed to model face recognition as multi-task learning problem, which enables us to exploit both, shared common information and also individual characteristics of faces. In particular, we extended KISSME to multi-task

learning. The resulting algorithm supports label-incompatible learning which allows us to use the rather large pool of anonymously labeled face pairs to learn a more robust distance measure. Second, we showed how to learn and combine person specific metrics for face identification improving the classification power.

In a face recognition pipeline face detection and landmark extraction are crucial preprocessing steps that heavily influence the final face recognition performance. Especially for face detection and landmark localization recent works rely heavily on machine learning algorithms using massive amounts of data. Thus, ultimately a key step for face detection is also the availability of training data in large-scale. Therefore, we proposed in Chapter 3 our large-scale, real-world database termed Annotated Facial Landmarks in the Wild (AFLW). Once having introduced AFLW we showed that existing face detectors are not always limited by their models but by the available training data. In particular, we were able to achieve a drastically increased face detection performance, using a standard algorithm [138] with standard features [87]. Moreover, we outperformed sophisticated state-of-the-art methods on the Face Detection Dataset and Benchmark (FDDB) [64].

Further, in Chapter 2 we briefly reviewed related works in classical and real-world face recognition. In particular, we discussed recent methods that focus on appropriate representations and sophisticated machine learning algorithms that are able to deal real-world challenges. Further, face-specific recognition strategies have been discussed that use an auxiliary set of faces for improved matching or sophisticated alignment strategies.

7.2 Future Work

If we recapitulate our contributions and put them into context it is obvious that some issues remain open that offer interesting perspectives for future research. Revisiting the real-world face recognition review in Chapter 2 it becomes obvious that also our methods would benefit of an auxiliary set of faces similar to [156]. In particular, the auxiliary set of faces could be used to predict the appearance of one face under different pose or illumination settings. This could allow for better handling of face pose or illumination changes. The metric for associating one face to a similar face of the gallery set could be learned. Another interesting aspect would be to use different learned-metrics for different face regions, similar to [23]. In our current work we artificially limited ourselves to a single feature type for representing faces. Clearly, it would be beneficial to ex-

tract and blend together multiple complementary feature types, e.g., [115]. Concerning our work on Mahalanobis metric learning it would be beneficial to derive a version of KISSME allowing for kernelization of the metric learning problem. This could be favorable in situations where dimensionality reduction would result in a loss of information. Further, extending KISSME for multiple-instance and / or semi-supervised learning should be fairly straight forward and would offer many new fields of application. For Discriminative Metric and Prototype Learning we have to fix a priori the number of prototypes. This number might not be optimal for all classes in a setting where the number of samples is not balanced. Therefore, future work should investigate the possibility of a dynamic number of prototypes. Concerning the promising face detection results presented in Chapter 3 future work should deal with training a multi-view detector tree similar to [56]. Our Annotated Facial Landmarks in the Wild database contains enough non-frontal face samples for this task.

Addressing all contributions and limitations of this thesis real-world face recognition remains still challenging. However, this thesis provides promising results in terms of scalability of Mahalanobis metric learning algorithms allowing for efficient training and evaluation of face recognition models.

A List of Publications

2013

Joint Learning of Discriminative Prototypes and Large Margin Nearest Neighbor Classifiers

Martin Köstinger, Paul Wohlhart, Peter M. Roth, and Horst Bischof
In: *Proc. IEEE International Conference on Computer Vision* (ICCV),
December 2013, Sydney, Australia.
(Accepted for poster presentation, 27.87 % acceptance rate)

Efficient Retrieval for Large Scale Metric Learning

Martin Köstinger, Peter M. Roth, and Horst Bischof
In: *Proc. German Conference on Pattern Recognition* (GCPR/DAGM),
September 2013, Saarbrücken, Germany.
(Accepted for oral presentation)

Optimizing 1-Nearest Prototype Classifiers

Paul Wohlhart, Martin Köstinger, Michael Donoser, Peter M. Roth, and Horst Bischof
In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (CVPR),
June 2013, Portland, Oregon, U.S.A.
(Accepted for poster presentation, 25.2 % acceptance rate)

2012

Large Scale Metric Learning from Equivalence Constraints

Martin Köstinger, Martin Hirzer, Paul Wohlhart, Peter M. Roth, and Horst Bischof
In: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* (CVPR),
June 2012, Providence, Rhode Island, U.S.A.
(Accepted for poster presentation, 24.05 % acceptance rate)

Synergy-based Learning of Facial Identity

Martin Köstinger, Peter M. Roth, and Horst Bischof
In: *Proc. DAGM-OEAGM Symposium* (DAGM-OEAGM),
August 2012, Graz, Austria.
(Winner of the Best Paper Award)

Robust Face Detection by Simple Means

Martin Köstinger, Paul Wohlhart, Peter M. Roth, and Horst Bischof
In: *Computer Vision in Applications Workshop* (DAGM-OEAGM-WS),
August 2012, Graz, Austria.
(Accepted for oral presentation)

Relaxed Pairwise Learned Metric for Person Re-Identification

Martin Hirzer, Peter M. Roth, Martin Köstinger, and Horst Bischof
In: *Proc. European Conf. on Computer Vision* (ECCV),
October 2012, Firenze, Italy.
(Accepted for poster presentation, 24.5 % acceptance rate)

Dense Appearance Modeling and Efficient Learning of Camera Transitions for Person Re-Identification

Martin Hirzer, Csaba Beleznai, Martin Köstinger, Peter M. Roth, and Horst Bischof
In: *Proc. IEEE Int'l Conf. on Image Processing* (ICIP),
October 2012, Orlando, Florida, U.S.A.
(Accepted for oral presentation)

Discriminative Hough Forests for Object Detection

Paul Wohlhart, Samuel Schulter, Martin Köstinger, Peter M. Roth, and Horst Bischof
In: *Proc. British Machine Vision Conf.* (BMVC),
September 2012, Guildford, United Kingdom.
(Accepted for poster presentation, 24 % acceptance rate)

2011

Annotated Facial Landmarks in the Wild: A Large-scale, Real-world Database for Facial Landmark Localization

Martin Köstinger, Paul Wohlhart, Peter M. Roth, and Horst Bischof
In: *IEEE International Workshop on Benchmarking Facial Image Analysis Technologies (ICCV),* (BEFIT),
November 2011, Barcelona, Spain.
(Accepted for oral presentation)

Learning to Recognize Faces from Videos and Weakly Related Information Cues

Martin Köstinger, Paul Wohlhart, Peter M. Roth, and Horst Bischof
In: *Proc. IEEE Int'l Conf. on Advanced Video and Signal-Based Surveillance* (AVSS),
August–September 2011, Klagenfurt, Austria.
(Accepted for oral presentation)

Multiple Instance Boosting for Face Recognition in Videos

Paul Wohlhart, Martin Köstinger, Peter M. Roth, and Horst Bischof
In: *DAGM Symposium* (DAGM),
August–September 2011, Frankfurt, Germany.
(Accepted for oral presentation)

Learning Face Recognition in Videos from Associated Information Sources

Paul Wohlhart, Martin Köstinger, Peter M. Roth, and Horst Bischof
In: *In Proc. Workshop of the Austrian Association for Pattern Recognition* (AAPR),

May 2011, Graz, Austria.
(Accepted for oral presentation)

2010

Automatic Detection and Reading of Dangerous Goods Plates

Peter M. Roth, Martin Köstinger, Paul Wohlhart, Horst Bischof, and Josef Birchbauer
In: *Proc. IEEE Int'l Conf. on Advanced Video and Signal-Based Surveillance* (AVSS),
August–September 2010, Boston, Massachusetts, USA.
(Accepted for poster presentation)

Video Detection of Dangerous Goods Vehicles in Road Tunnels

Josef Birchbauer, Martin Köstinger, Paul Wohlhart, Peter M. Roth, Horst Bischof, and Claudia Windisch
In: *Proc. Tunnel Safety and Ventilation*,
May 2010, Graz, Austria.
(Accepted for oral presentation)

Bibliography

[1] Timo Ahonen, Abdenour Hadid, and Matti Pietikänien. Face Recognition with Local Binary Patterns. In *Proceedings European Conference on Computer Vision*, 2004. (cited on page 31)

[2] Anelia Angelova, Yaser Abu-Mostafam, and Pietro Perona. Pruning training sets for learning of object categories. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2005. (cited on page 32)

[3] Oya Aran, Ismail Ari, Amac A. Guvensan, Hakan Haberdar, Zeyneb Kurt, H. Irem Turkmen, Ash Uyar, and Lale Akarun. A database of non-manual signs in turkish sign language. In *Proceedings Signal Processing and Communications Applications*, 2007. (cited on pages 32, 33, and 34)

[4] Peter N. Belhumeur, João P. Hespanha, and David J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997. (cited on page 9)

[5] Tamara L. Berg, Alex C. Berg, Jaety Edwards, Michael Maire, Ryan White, Yee-Whye Teh, Erik G. Learned-Miller, and David A. Forsyth. Names and faces in the news. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2004. (cited on page 2)

[6] Thomas Berg and Peter Belhumeur. Tom-vs-Pete classifiers and identity-preserving alignment for face verification. In *Proceedings British Machine Vision Conference*, 2012. (cited on pages 3, 18, 20, 23, and 30)

[7] Edwin Bonilla and Antonio Robles-Kelly. Discriminative probabilistic prototype learning. In *Proceedings International Conference on Machine Learning*, 2012. (cited on page 97)

[8] Antoine Bordes, Léon Bottou, Patrick Gallinari, and Jason Weston. Solving multiclass support vector machines with larank. In *Proceedings International Conference on Machine Learning*, 2007. (cited on page 103)

[9] Lev M. Bregman. The Relaxation Method of Finding the Common Point of Convex Sets and Its Application to the Solution of Problems in Convex Programming. *USSR Computational Mathematics and Mathematical Physics*, 7:200–217, 1967. (cited on page 55)

[10] Teófilo E. de Campo, Bodla Rakesh Babu, and Manik Varma. Character Recognition in Natural Images. In *Proceedings International Conference on Computer Vision Theory and Applications*, 2009. (cited on pages 77, 78, 79, 101, 102, and 104)

[11] Zhimin Cao, Qi Yin, Xiaoou Tang, and Jian Sun. Face recognition with learning-based descriptor. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2010. (cited on pages 12, 13, 15, 23, and 30)

[12] Richard Caruana. Multitask learning: A knowledge-based source of inductive bias. In *Proceedings International Conference on Machine Learning*, 1993. (cited on page 84)

[13] Richard Caruana. Multitask learning. *Machine Learning*, 28:41–75, 1997. (cited on page 84)

[14] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. (cited on pages 64, 89, and 103)

[15] Moses S. Charikar. Similarity estimation techniques from rounding algorithms. In *ACM Symposium on Theory of Computing*, 2002. (cited on pages 52, 59, 61, and 80)

[16] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: deep neural networks with multitask learning. In *Proceedings International Conference on Machine Learning*, 2008. (cited on page 84)

[17] Tim F. Cootes and Christopher J. Taylor. Active shape models. In *Proceedings British Machine Vision Conference*, pages 266–275, 1992. (cited on page 9)

[18] Timothy F. Cootes, Gareth J. Edwards, and Christopher J. Taylor. Active appearance models. In *Proceedings European Conference on Computer Vision*, 1998. (cited on page 9)

[19] Timothy F. Cootes, Gareth J. Edwards, and Christopher J. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6): 681–685, 2001. (cited on page 9)

[20] David D. Cox and Nicolas Pinto. Beyond simple features: A large-scale feature search approach to unconstrained face recognition. In *Proceedings IEEE International Conference on Automatic Face Gesture Recognition*, 2011. (cited on pages 17 and 22)

[21] Koby Crammer, Yoram Singer, Nello Cristianini, John Shawe-taylor, and Bob Williamson. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2:265–292, 2001. (cited on pages 89, 91, and 103)

[22] Koby Crammer, Ran Gilad-bachrach, Amir Navot, and Naftali Tishby. Margin analysis of the LVQ algorithm. In *Advances in Neural Information Processing Systems*, 2002. (cited on pages 94 and 96)

[23] Zhen Cui, Wen Li, Dong Xu, Shiguang Shan, and Xilin Chen. Fusing robust face region descriptors via multiple metric learning for face recognition in the wild. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2013. (cited on page 111)

[24] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2005. (cited on pages 12 and 31)

[25] Matthias Dantone, Jürgen Gall, Gabriele Fanelli, and Luc van Gool. Real-time facial feature detection using conditional regression forests. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2012. (cited on page 32)

[26] John G. Daugman. Complete discrete 2-d gabor transforms by neural networks for image analysis and compression. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 36(7):1169 –1179, 1988. (cited on pages 11 and 12)

[27] Jason V. Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S. Dhillon. Information-theoretic metric learning. In *Proceedings International Conference on Machine Learning*, 2007. (cited on pages 20, 51, 54, 63, 64, 67, 69, 76, 80, 94, 96, 103, and 107)

[28] Fernando De la Torre and Minh Hoai Nguyen. Parameterized kernel principal component analysis: Theory and applications to supervised and unsupervised image alignment. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2008. (cited on page 9)

[29] Daniel F. Dementhon and Larry S. Davis. Model-based object pose in 25 lines of code. *International Journal of Computer Vision*, 15:123–141, 1995. (cited on page 37)

[30] Mert Dikmen, Emre Akbas, Thomas S. Huang, and Narendra Ahuja. Pedestrian recognition with a learned metric. In *Proceedings Asian Conference on Computer Vision*, 2010. (cited on pages 50, 68, and 69)

[31] Gareth J. Edwards, Timothy F. Cootes, and Christopher J. Taylor. Face recognition using active appearance models. In *Proceedings European Conference on Computer Vision*, 1998. (cited on page 9)

[32] Mark Everingham, Josef Sivic, and Andrew Zisserman. "Hello! My name is... Buffy" – automatic naming of characters in TV video. In *Proceedings British Machine Vision Conference*, 2006. (cited on pages 29, 31, and 36)

[33] Mark Everingham, Luc van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results, 2007. (cited on page 41)

[34] Theodoros Evgeniou and Massimiliano Pontil. Regularized multi–task learning. In *Proceedings International Conference on Knowledge discovery and data mining*, 2004. (cited on pages 84 and 89)

[35] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874, 2008. (cited on pages 63, 64, 65, 70, 103, and 104)

[36] Michela Farenzena, Loris Bazzani, Alessandro Perina, Vittorio Murino, and Marco Cristani. Person re-identification by symmetry-driven accumulation of local fea-

tures. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2010. (cited on pages 67, 68, and 69)

[37] Li Fei-Fei, Rod Fergus, and Pietro Perona. Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. In *Proceedings CVPR Workshop on Generative-Model Based Vision*, 2004. (cited on page 2)

[38] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010. (cited on pages 45 and 46)

[39] Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627 –1645, 2010. (cited on pages 12 and 31)

[40] Ronald Aylmer Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936. (cited on page 9)

[41] Yoav Freund, Sanjoy Dasgupta, Mayank Kabra, and Nakul Verma. Learning the structure of manifolds using random projections. In *Advances in Neural Information Processing Systems*, 2007. (cited on page 16)

[42] Jerome H. Friedman. Another approach to polychotomous classification. Technical report, Department of Statistics, Stanford University, 1996. (cited on page 86)

[43] Wen Gao, Bo Cao, Shiguang Shan, Xilin Chen, Delong Zhou, Xiaohua Zhang, and Debin Zhao. The CAS-PEAL large-scale chinese face database and baseline evaluations. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, 38(1):149–161, 2008. (cited on page 32)

[44] Aristides Gionis, Piotr Indyk, and Rajeev Motwani. Similarity search in high dimensions via hashing. In *Proceedings Very Large Data Bases*, 1999. (cited on pages 52, 59, and 103)

[45] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison-Wesley Longman Publishing Co., Inc., 2nd edition, 2001. (cited on page 11)

[46] Douglas Gray and Hai Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *Proceedings European Conference on Computer Vision*, 2008. (cited on pages 67 and 68)

[47] Douglas Gray, Shane Brennan, and Hai Tao. Evaluating appearance models for recongnition, reacquisition and tracking. In *Proceedings IEEE Workshop on Performance Evaluation of Tracking and Surveillance*, 2007. (cited on pages 52, 64, and 67)

[48] Gregory Griffin, Alex D. Holub, and Pietro Perona. Caltech-256 object category dataset. Technical Report 7694, California Institute of Technology, 2007. (cited on page 2)

[49] Patrick J. Grother, George W. Quinn, and P. Jonathon Phillips. Report on the evaluation of 2d still-image face recognition algorithms. Technical Report 7709, National Institute of Standards and Technology, 2011. (cited on page 9)

[50] Matthieu Guillaumin, Jakob Verbeek, and Cordelia Schmid. Is that you? Metric learning approaches for face identification. In *Proceedings IEEE International Conference on Computer Vision*, 2009. (cited on pages 12, 20, 51, 54, 55, 62, 63, 64, 67, 69, 71, 72, 76, 80, 96, 103, and 107)

[51] Jae-Pil Heo, Youngwoon Lee, Junfeng He, Shih-Fu Chang, and Sung-Eui Yoon. Spherical hashing. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2012. (cited on pages 52, 80, and 103)

[52] Nicholas J. Higham. Computing a nearest symmetric positive semidefinite matrix. *Linear Algebra and its Applications*, 103:103–118, 1988. (cited on pages 58 and 101)

[53] Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002. (cited on page 17)

[54] Martin Hirzer, Csaba Beleznai, Peter Roth, and Horst Bischof. Person re-identification by descriptive and discriminative classification. In *Proceedings Scandinavian Conference on Image Analysis*, 2011. (cited on pages 68 and 69)

[55] Steven C.H. Hoi, Wei Liu, Michael R. Lyu, and Wei-Ying Ma. Learning distance metrics with contextual constraints for image retrieval. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2006. (cited on page 50)

[56] Chang Huang, Haizhou Ai, Yuan Li, and Shihong Lao. Vector boosting for rotation invariant multi-view face detection. In *Proceedings IEEE International Conference on Computer Vision*, 2005. (cited on pages 5, 29, 39, 41, and 112)

[57] Chang Huang, Haizhou Ai, Yuan Li, and Shihong Lao. High-performance rotation invariant multiview face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4):671–686, 2007. (cited on page 41)

[58] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik G. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, UMASS, Amherst, 2007. (cited on pages 2, 10, 11, 12, 25, 30, 31, 32, 50, 52, 62, 64, 71, and 110)

[59] Gary B. Huang, Marwan A. Mattar, Honglak Lee, and Erik G. Learned-Miller. Learning to align from scratch. In *Advances in Neural Information Processing Systems*, 2012. (cited on page 3)

[60] Jonathan J. Hull. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994. (cited on pages 77, 78, 79, 101, and 104)

[61] Sibt Ul Hussain, Thibault Napoléon, and Frédéric Jurie. Face recognition using local quantized patterns. In *Proceedings British Machine Vision Conference*, 2012. (cited on page 16)

[62] Andrew J. and Arthur Asuncion. UCI machine learning repository, 2010. University of California, Irvine, School of Information and Computer Sciences. Available online at http://archive.ics.uci.edu/ml. (cited on pages 77, 78, 79, 101, and 104)

[63] Prateek Jain, Brian Kulis, and Kristen Grauman. Fast image search for learned metrics. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2008. (cited on pages 52, 59, 80, 94, 103, and 104)

[64] Vidit Jain and Erik G. Learned-Miller. FDDB: A benchmark for face detection in unconstrained settings. Technical Report UM-CS-2010-009, UMASS, Amherst, 2010. (cited on pages 35, 39, and 111)

[65] Vidit Jain and Erik G. Learned-Miller. Online domain adaptation of a pre-trained cascade of classifiers. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 577–584, 2011. (cited on page 43)

[66] Oliver Jesorsky, Klaus J. Kirchberg, and Robert W. Frischholz. Robust face detection using the Hausdorff distance. In *Proceedings Audio and Video-based Biometric Person Authentication*, 2001. (cited on pages 32, 34, and 35)

[67] Michael J. Jones and Paul Viola. Face recognition using boosted local features. Technical report, Mitsubishi Electric Research Laboratories, 2003. (cited on page 19)

[68] Zdenek Kalal, Jiri Matas, and Krystian Mikolajczyk. Weighted sampling for large-scale boosting. In *Proceedings British Machine Vision Conference*, 2008. (cited on pages 43, 44, 45, and 46)

[69] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. Face-TLD: Tracking-Learning-Detection Applied to Faces. In *Proceedings International Conference on Image Processing*, 2010. (cited on pages 43 and 44)

[70] Takeo Kanade. Computer recognition of human faces. *Interdisciplinary Systems Research*, 47, 1977. (cited on page 9)

[71] Schmidt A. Kasiński A., Florek A. The PUT face database. *Image Processing & Communications*, 13(3–4):59–64, 2008. (cited on pages 32, 33, and 34)

[72] Michael David Kelly. *Visual identification of people by computer*. PhD thesis, Stanford University, Stanford, CA, USA, 1970. (cited on page 9)

[73] Teuvo Kohonen. *Self-organization and associative memory*. Springer-Verlag New York, Inc., 1989. (cited on page 96)

[74] Martin Köstinger, Paul Wohlhart, Peter M. Roth, and Horst Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *Proceedings ICCV Workshop on Benchmarking Facial Image Analysis Technologies*, 2011. (cited on pages 4, 5, 12, 13, 30, and 31)

[75] Martin Köstinger, Martin Hirzer, Paul Wohlhart, Peter M. Roth, and Horst Bischof. Large scale metric learning from equivalence constraints. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2012. (cited on pages 4, 5, 21, 80, 94, and 103)

[76] Martin Köstinger, Peter M. Roth, and Horst Bischof. Synergy-based learning of facial identity. In *Proceedings DAGM Symposium*, 2012. (cited on pages 4 and 6)

[77] Martin Köstinger, Paul Wohlhart, Peter M. Roth, and Horst Bischof. Robust face detection by simple means. In *Proceedings DAGM Workshop on Computer Vision in Applications*, 2012. (cited on pages 4 and 43)

[78] Martin Köstinger, Peter M. Roth, and Horst Bischof. Efficient retrieval for large scale metric learning. In *Proceedings German Conference on Pattern Recognition*, 2013. (cited on pages 4 and 5)

[79] Martin Köstinger, Paul Wohlhart, Peter M. Roth, and Horst Bischof. Joint learning of discriminative prototypes and large margin nearest neighbor classifiers. In *Proceedings IEEE International Conference on Computer Vision*, 2013. (cited on pages 4 and 6)

[80] Brian Kulis and Kristen Grauman. Kernelized locality-sensitive hashing for scalable image search. In *Proceedings IEEE International Conference on Computer Vision*, 2009. (cited on pages 80 and 103)

[81] Brian Kulis and Kristen Grauman. Kernelized locality-sensitive hashing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(6):1092–1104, 2012. (cited on pages 52, 79, 80, 94, and 103)

[82] Neeraj Kumar, Alexander C. Berg, Peter N. Belhumeur, and Shree K. Nayar. Attribute and Simile Classifiers for Face Verification. In *Proceedings IEEE International Conference on Computer Vision*, 2009. (cited on pages 2, 10, 17, 26, 27, 29, 36, 52, 62, 63, 64, 65, 71, 72, 73, 74, 75, 85, 87, 95, 101, 105, and 106)

[83] Lubor Ladicky and Philip H. S. Torr. Locally linear support vector machines. In *Proceedings International Conference on Machine Learning*, 2011. (cited on pages 103, 104, and 105)

[84] Andreas Lanitis, Christopher J. Taylor, and Timothy F. Cootes. Automatic face identification system using flexible appearance models. *Image and Vision Computing*, 13(5):393 – 401, 1995. (cited on page 9)

[85] Haoxiang Li, Gang Hua, Zhe Lin, Jonathan Brandt, and Jianchao Yang. Probabilistic elastic matching for pose variant face verification. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2012. (cited on pages 15 and 22)

[86] Jianguo Li, Tao Wang, and Yimin Zhang. Face detection using surf cascade. In

Proceedings ICCV Workshop on Benchmarking Facial Image Analysis Technologies, 2011. (cited on pages 43 and 44)

[87] Shengcai Liao, Xiangxin Zhu, Zhen Lei, Lun Zhang, and Stan Z. Li. Learning multi-scale block local binary patterns for face recognition. In *International Conference on Biometrics*, 2007. (cited on pages 41, 43, and 111)

[88] Yuanqing Lin, Tong Zhang, Shenghuo Zhu, and Kai Yu. Deep coding network. In *Advances in Neural Information Processing Systems*, 2010. (cited on page 103)

[89] David Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. (cited on pages 11, 12, and 62)

[90] Prasanta C. Mahalanobis. On the generalised distance in statistics. In *Proceedings National Institute of Science, India*, volume 2, pages 49–55, 1936. (cited on page 53)

[91] Aleix Martínez and R. Benavente. The AR face database. Technical Report 24, Computer Vision Center, University of Barcelona, 1998. (cited on page 32)

[92] Eugene Mayer and Rossion Bruno. Prosopagnosia. In *The Behavioral and Cognitive Neurology of Stroke*. Cambridge University Press, 2007. (cited on page 1)

[93] Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. Metric Learning for Large Scale Image Classification: Generalizing to New Classes at Near-Zero Cost. In *Proceedings European Conference on Computer Vision*, 2012. (cited on page 4)

[94] Kieron Messer, Jiri Matas, Josef Kittler, and Kenneth Jonsson. XM2VTSDB: The extended M2VTS database. In *Proceedings Audio and Video-based Biometric Person Authentication*, 1999. (cited on pages 32, 33, and 34)

[95] Krystian Mikolajczyk, Cordelia Schmid, and Andrew Zisserman. Human detection based on a probabilistic assembly of robust part detectors. In *Proceedings European Conference on Computer Vision*, 2004. (cited on page 43)

[96] Stephen Milborrow, John Morkel, and Fred Nicolls. The MUCT Landmarked Face Database. In *Proceedings Pattern Recognition Association of South Africa*, 2010. (cited on pages 32, 33, and 34)

[97] Frank Moosmann, Eric Nowak, and Frederic Jurie. Randomized clustering forests for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(9):1632–1646, 2008. (cited on page 16)

[98] Erik Murphy-Chutorian and Mohan Manubhai Trivedi. Head pose estimation in computer vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(4):607–626, 2009. (cited on pages 29 and 36)

[99] Vinod Nair and Geoffrey E. Hinton. Rectified Linear Units Improve Restricted Boltzmann Machines. In *Proceedings International Conference on Machine Learning*, 2012. (cited on page 16)

[100] Hieu V. Nguyen and Li Bai. Cosine similarity metric learning for face verification. In *Proceedings Asian Conference on Computer Vision*, 2011. (cited on pages 12, 13, 21, and 22)

[101] Minh Hoai Nguyen and Fernando De la Torre. Metric learning for image alignment. *International Journal of Computer Vision*, 88(1):69–84, 2010. (cited on page 50)

[102] Christos Papachristou Niki Aifanti and Anastasios Delopoulos. The MUG facial expression database. In *Proceedings Workshop on Image Analysis for Multimedia Interactive Services*, 2005. (cited on page 32)

[103] Michael M. Nordstrøm, Mads Larsen, Janusz Sierakowski, and Mikkel B. Stegmann. The IMM face database - an annotated dataset of 240 face images. Technical report, Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2004. (cited on page 32)

[104] Mohammad Norouzi, Ali Punjani, and David J. Fleet. Fast search in hamming space with multi-index hashing. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2012. (cited on page 61)

[105] Eric Nowak and Frédéric Jurie. Learning visual similarity measures for comparing never seen objects. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2007. (cited on pages 16, 30, 52, 64, 69, 70, and 71)

[106] Timo Ojala, Matti Pietikänien, and Topi Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002. (cited on pages 11, 12, 68, and 70)

[107] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22:1345–1359, 2010. (cited on page 84)

[108] Shibin Parameswaran and Kilian Q. Weinberger. Large margin multi-task metric learning. In *Advances in Neural Information Processing Systems*, 2010. (cited on pages 84, 85, and 89)

[109] Karl Pearson. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Sciences*, 6(2): 559–572, 1901. (cited on page 9)

[110] Alex P. Pentland, Baback Moghaddam, and Thad Starner. View-based and modular eigenspaces for face recognition. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 1994. (cited on page 9)

[111] P. Jonathon Phillips. Still face challenge problem multiple biometric grand challenge preliminary results of version 2. Website, 2009. Available online at `http://biometrics.nist.gov/cs_links/face/mbgc/2009/FACE_V2_FINAL.pdf`; visited on February 15th 2013. (cited on page 9)

[112] P. Jonathon Phillips, Hyeonjoon Moon, Syed .A. Rizvi, and Patrick J. Rauss. The feret evaluation methodology for face-recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10):1090–1104, 2000. (cited on pages 2 and 10)

[113] P. Jonathon Phillips, Patrick J. Flynn, Todd Scruggs, Kevin W. Bowyer, Jin Chang, Kevin Hoffman, Joe Marques, Jaesik Min, and William Worek. Overview of the face recognition grand challenge. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2005. (cited on page 27)

[114] P. Jonathon Phillips, W. Todd Scruggs, Alice J. O'Toole, Patrick J. Flynn, Kevin W. Bowyer, Cathy L. Schott, and Matthew Sharpe. Frvt 2006 and ice 2006 large-scale experimental results. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(5):831 –846, 2010. (cited on page 2)

[115] Nicolas Pinto, James J. DiCarlo, and David D. Cox. How far can you get with a modern face recognition test set using only simple features? In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2009. (cited on pages 22 and 112)

[116] John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large-Margin Classifiers*. MIT Press, 1999. (cited on page 86)

[117] Simon Prince, Peng Li, Yun Fu, Umar Mohammed, and James Elder. Probabilistic models for inference about identity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(1):144–157, 2011. (cited on page 30)

[118] Bryan Prosser, Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. Person re-identification by support vector ranking. In *Proceedings British Machine Vision Conference*, pages 21.1–21.11, 2010. (cited on page 68)

[119] Ryan Rifkin and Aldebaro Klautau. In defense of one-vs-all classification. *Journal of Machine Learning Research*, 5:101–141, 2004. (cited on page 85)

[120] Henry Rowley, Shumeet Baluja, and Takeo Kanade. Rotation invariant neural network-based face detection. Technical Report CMU-CS-97-201, Computer Science Department, Carnegie Mellon University (CMU), 1997. (cited on page 32)

[121] Conrad Sanderson and Brian C. Lovell. Multi-Region Probabilistic Histograms for Robust and Scalable Identity Inference. In *International Conference on Biometrics*, 2009. (cited on pages 15 and 23)

[122] Henry Schneiderman and Takeo Kanade. A statistical model for 3D object detection applied to faces and cars. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2000. (cited on pages 32 and 35)

[123] Hae Jong Seo and Peyman Milanfar. Training-free, generic object detection using locally adaptive regression kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1688–1704, 2008. (cited on pages 11, 12, and 14)

[124] Hae Jong Seo and Peyman Milanfar. Face verification using the lark representation. *IEEE Transactions on Information Forensics and Security*, 6(4):1275 –1286, 2011. (cited on pages 11, 12, 14, and 22)

[125] Sambu Seo and Klaus Obermayer. Soft learning vector quantization. *Neural Computation*, 15:1589–1604, 2002. (cited on page 96)

[126] Pawan Sinha, Benjamin Balas, Yuri Ostrovsky, and Richard Russell. Face recognition by humans: Nineteen results all computer vision researchers should know about. *Proceedings of the IEEE*, 94(11):1948–1962, 2006. (cited on pages 2, 9, and 23)

[127] Josef Sivic, Mark Everingham, and Andrew Zisserman. "Who are you?" Learning person specific classifiers from video. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2009. (cited on page 30)

[128] Paul Smolensky. *Information processing in dynamical systems: Foundations of harmony theory*, volume 1, pages 194–281. MIT Press, 1986. (cited on page 17)

[129] Zak Stone, Todd Zickler, and Trevor Darrell. Toward large-scale face recognition using social network context. *Proceedings of the IEEE*, 98(8):1408–1415, 2010. (cited on page 4)

[130] Markus Storer, Martin Urschler, and Horst Bischof. R3D-MAM: 3D morphable appearance model for efficient fine head pose estimation from still images. In *Proceedings ICCV Workshop on Subspace Methods*, 2009. (cited on page 37)

[131] Venkatesh Bala Subburaman and Sebastien Marcel. Fast Bounding Box Estimation based Face Detection. In *Proceedings ECCV Workshop on Face Detection*, 2010. (cited on page 43)

[132] Yaniv Taigman and Lior Wolf. Leveraging billions of faces to overcome performance barriers in unconstrained face recognition. *arXiv.org*, abs/1108.1122, 2011. (cited on page 41)

[133] Yaniv Taigman, Lior Wolf, and Tal Hassner. Multiple One-Shots for Utilizing Class Label Information. In *Proceedings British Machine Vision Conference*, 2009. (cited on pages 3, 11, 12, 20, 22, and 24)

[134] Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, and Yasemin Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6:1453–1484, 2005. (cited on page 103)

[135] Matthew A. Turk and Alex P. Pentland. Face recognition using eigenfaces. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 1991. (cited on page 9)

[136] Andrea Vedaldi and Andrew Zisserman. Efficient additive kernels via explicit feature maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(3): 480–492, 2011. (cited on pages 103 and 106)

[137] Paul Viola and Michael J. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2001. (cited on pages 11, 26, 39, 41, 43, and 45)

[138] Paul Viola and Michael J. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, volume I, pages 511–518, 2001. (cited on pages 46 and 111)

[139] Fan Wang and Leonidas J. Guibas. Supervised earth mover's distance learning and its computer vision applications. In *Proceedings European Conference on Computer Vision*, 2012. (cited on pages 21 and 22)

[140] Jinjun Wang, Jianchao Yang, Kai Yu, Fengjun Lv, Thomas S. Huang, and Yihong Gong. Locality-constrained linear coding for image classification. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2010. (cited on pages 103 and 104)

[141] X. Wang, G. Doretto, T. B. Sebastian, J. Rittscher, and P. H. Tu. Shape and appearance context modeling. In *Proceedings IEEE International Conference on Computer Vision*, 2007. (cited on page 68)

[142] Zhuang Wang, Koby Crammer, and Slobodan Vucetic. Multi-Class Pegasos on a Budget. In *Proceedings International Conference on Machine Learning*, 2010. (cited on page 103)

[143] Kilian Q. Weinberger and Lawrence K. Saul. Fast solvers and efficient implementations for distance metric learning. In *Proceedings International Conference on Machine Learning*, 2008. (cited on pages 51, 54, 80, 85, 94, 96, and 103)

[144] Kilian Q. Weinberger and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10: 207–244, 2009. (cited on page 104)

[145] Kilian Q. Weinberger, John Blitzer, and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances in Neural Information Processing Systems*, 2005. (cited on pages 51, 54, 63, 64, 66, 67, 69, 76, 80, 91, 96, 97, 103, 104, and 107)

[146] Yair Weiss, Antonio Torralba, and Robert Fergus. Spectral hashing. In *Advances in Neural Information Processing Systems*, 2008. (cited on pages 52, 80, and 103)

[147] Yair Weiss, Rob Fergus, and Antonio Torralba. Multidimensional spectral hashing. In *Proceedings European Conference on Computer Vision*, 2012. (cited on pages 80 and 103)

[148] Laurenz Wiskott, Jean-Marc Fellous, Norbert Krüger, and Christoph von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):775–779, 1997. (cited on page 9)

[149] Lior Wolf, Tal Hassner, and Yaniv Taigman. Descriptor based methods in the wild. In *Proceedings ECCV Workshop on Faces in Real-Life Images*, 2008. (cited on pages 12 and 23)

[150] Lior Wolf, Tal Hassner, and Yaniv Taigman. Similarity scores based on background samples. In *Proceedings Asian Conference on Computer Vision*, 2009. (cited on pages 11, 12, and 24)

[151] Lior Wolf, Tal Hassner, and Yaniv Taigman. Effective unconstrained face recognition by combining multiple descriptors and learned background statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(10):1978–1990, 2011. (cited on pages 11, 12, and 13)

[152] Xuehan Xiong and Fernando De la Torre. Supervised descent method and its applications to face alignment. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2013. (cited on page 3)

[153] Jay Yagnik, Dennis Strelow, David A. Ross, and Ruei-sung Lin. The power of comparative reasoning. In *Proceedings IEEE International Conference on Computer Vision*, 2011. (cited on page 103)

[154] Tao Yang and Vojislav Kecman. Adaptive local hyperplane classification. *Neurocomputing*, 71(13–15):3001–3004, 2008. (cited on page 103)

[155] Jieping Ye, Zheng Zhao, and Huan Liu. Adaptive distance metric learning for clustering. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2007. (cited on page 50)

[156] Qi Yin, Xiaoou Tang, and Jian Sun. An associate-predict model for face recognition. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2011. (cited on pages 22, 24, and 111)

[157] Yiming Ying and Peng Li. Distance metric learning with eigenvalue optimization. *Journal of Machine Learning Research*, 2012. (cited on pages 21 and 22)

[158] Jie Yu, Jaume Amores, Nicu Sebe, and Qi Tian. Toward robust distance metric analysis for similarity estimation. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2006. (cited on page 50)

[159] Kai Yu and Tong Zhang. Improved local coordinate coding using local tangents. In *Proceedings International Conference on Machine Learning*, 2010. (cited on pages 103 and 104)

[160] Kai Yu, Tong Zhang, and Yihong Gong. Nonlinear learning using local coordinate coding. In *Advances in Neural Information Processing Systems*, 2009. (cited on pages 103 and 104)

[161] Hao Zhang, Alexander C. Berg, Michael Maire, and Jitendra Malik. Svm-knn: Discriminative nearest neighbor classification for visual category recognition. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2006. (cited on page 103)

[162] Ziming Zhang, Lubor Ladicky, Philip H. S. Torr, and Amir Saffari. Learning anchor planes for classification. In *Advances in Neural Information Processing Systems*, 2011. (cited on page 103)

[163] Ziming Zhang, Paul Sturgess, Sunando Sengupta, Nigel Crook, and Philip H. S. Torr. Efficient discriminative learning of parametric nearest neighbor classifiers. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2012. (cited on pages 77, 97, 100, 102, 103, 104, and 105)

[164] Wen-Yi Zhao, Rama Chellappa, P. Jonathon Phillips, and Azriel Rosenfeld. Face recognition: A literature survey. *ACM Computing Surveys*, 35(4):399–458, 2003. (cited on pages 3 and 9)

[165] Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. Person re-identification by probabilistic relative distance comparison. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2011. (cited on pages 68 and 69)

[166] Xiangxin Zhu and Deva Ramanan. Face detection, pose estimation, and landmark localization in the wild. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, 2012. (cited on pages 29, 43, 44, 45, and 46)