

# *Face recognition from caption-based supervision*

Matthieu Guillaumin — Thomas Mensink — Jakob Verbeek — Cordelia Schmid

**N° 0392**

20/09/2010

Vision, Perception and Multimedia Understanding

 **R**  
*apport  
technique*



## Face recognition from caption-based supervision

Matthieu Guillaumin , Thomas Mensink , Jakob Verbeek , Cordelia Schmid

Theme : Vision, Perception and Multimedia Understanding  
Équipe-Projet lear

Rapport technique n° 0392 — 20/09/2010 — 34 pages

**Abstract:** In this report, we present methods for face recognition using a collection of images with captions. We consider two tasks: retrieving all faces of a particular person in a data set, and establishing the correct association between the names in the captions and the faces in the images. This is challenging because of the very large appearance variation in the images, as well as the potential mismatch between images and their captions.

We survey graph-based, generative and discriminative approaches for both tasks. We extend them by considering different metric learning techniques to obtain appropriate face representations that reduce intra person variability and increase inter person separation. For the retrieval task, we also study the benefit of query expansion.

To evaluate performance, we use a new fully labeled data set of 31147 faces which extends the recent *Labeled Faces in the Wild* data set. We present extensive experimental results which show that metric learning significantly improves the performance of all approaches on both tasks.

**Key-words:** Face recognition, Metric Learning, Weakly supervised learning, Face retrieval, Constrained clustering

This research is partially funded by the Cognitive-Level Annotation using Latent Statistical Structure (CLASS) project of the European Union Information Society Technologies unit E5 (Cognition). We would also like to thank Tamara Berg, Mark Everingham, and Gary Huang for their help by providing data and code. We also thank Benoît Mordelet, Nicolas Breitenr and Lucie Daubigny for their participation in the annotation effort.

## Reconnaissance de visage partir d'images lgendes

**Résumé :** Dans ce rapport, nous présentons des méthodes pour la reconnaissance de visages qui utilisent des images accompagnées de légendes. Nous considérons deux applications: la recherche dans une base d'image de tous les visages d'une personne donnée, et l'établissement des correspondances correctes entre les noms qui apparaissent dans les légendes et les visages dans les images. Ces tâches sont difficiles à cause des très grandes variations d'apparence dans les images, et de la possible inadéquation entre les images et leurs légendes.

Nous comparons des approches génératives, discriminatives et à base de graphes pour les deux objectifs. Nous étendons ces approches en considérant l'utilisation de techniques d'apprentissage de distance pour obtenir des représentations vectorielles de visages qui réduisent la variabilité intra-personne tout en augmentant la séparation entre personnes. Pour la tâche de recherche, nous examinons également les gains apportés par les méthodes d'expansion de requête.

Pour évaluer la performance de nos systèmes, nous introduisons une nouvelle base de données de 31147 visages que nous avons manuellement annotés, et qui étend la base *Labeled Faces in the Wild*. Nous présentons des résultats expérimentaux exhaustifs qui montrent que l'utilisation de l'apprentissage de distance améliore significativement les performances de toutes les approches considérées.

**Mots-clés :** Reconnaissance de visage, Apprentissage de distance, Apprentissage faiblement supervisé, Recherche de visage, Agglomération de données contrainte

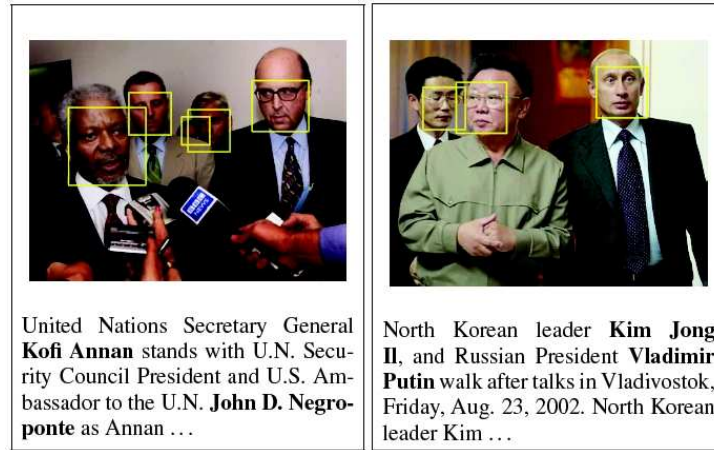


Figure 1: Two example images with captions. Detected named entities are in bold font, and detected faces are marked by yellow rectangles.

## 1 Introduction

Over the last decade we have witnessed an explosive growth of image and video data available both on-line and off-line, through digitalization efforts by broadcasting services, news oriented media publishing online, or user-provided content concentrated on websites such as YouTube and Flickr. This has led to the need for methods to index, search, and manipulate such data in a semantically meaningful manner. These methods effectively try to bridge the gap between low-level features and semantics (Smeulders et al., 2000). The volume of data in such archives is generally large, and the semantic concepts of interest differ greatly between different archives. Much research has addressed this problem using supervised techniques that require explicit manual annotations to establish correspondences between low-level features and semantics.

Learning semantic relations from weaker forms of supervision is currently an active and broad line of research (Barnard et al., 2003, Bekkerman and Jeon, 2007, Fergus et al., 2005, Li et al., 2007). The crux of those systems is to exploit the relations between different media, such as the relation between images and text, and between video and subtitles combined with scripts (Barnard et al., 2003, Everingham et al., 2006, Guillaumin et al., 2009a, Satoh et al., 1999, Sivic et al., 2009, Verbeek and Triggs, 2007). The correlations that can be automatically detected are typically less accurate – e.g. images and text associated using a web search engine like Google (Berg and Forsyth, 2006, Fergus et al., 2005) – than supervised information provided by explicit manual efforts. However, the important difference is that the former can be obtained at a lower cost, and therefore from much larger amounts of data, which may in practice outweigh the higher quality of supervised information.

In this paper, we focus on face recognition using weak supervision in the form of captions, see Figure 1 for illustrations. This paper presents a integrated overview of our results presented earlier elsewhere (Guillaumin et al., 2008, 2009b, Mensink and Verbeek, 2008). In addition, we extend the earlier work by integrating and improving the facial similarity learning approach of Guillaumin et al. (2009b) with the caption-based face recognition methods presented in Guillaumin et al. (2008), Mensink and



Figure 2: The extended YaleB data set includes illumination and pose variations for each subject, but not other variations such as ones due to expression.

Verbeek (2008). We propose a standardized evaluation protocol on a data set that we make publicly available, and also recently used in Guillaumin et al. (2010).

We will address two specific problems, the first is to retrieve all the faces belonging to a specific person from a given data set, and the second is to name all persons in a given image. The data set we use consists of images and captions from news streams, which are important as they are major sources in the information need of people, and news articles are published at a high frequency. Identification of faces in news photographs is a challenging task, significantly more so than recognition in the usual controlled setting of face recognition: we have to deal with imperfect face detection and alignment procedures, and also with great changes in pose, expression, and lighting conditions, and poor image resolution and quality. To stress the difficulty of face recognition in this setting, we show in Figure 2 images from the YaleB data set (Georghiadis et al., 2005), which are obtained in a controlled way, compared to images from the *Labeled Faces in the Wild* data set (Huang et al., 2007b) shown in Figure 3.

In this paper we consider the use of learned similarity measures to compare faces for these two tasks. We use the techniques we developed in Guillaumin et al. (2009b) for face identification. Face identification is a binary classification problem over pairs of face images: we have to determine whether or not the same person is depicted in the images. More generally, visual identification refers to deciding whether or not two images depict the same object from a certain class. The confidence scores, or a posteriori class probabilities, for the visual identification problem can be thought of as an object-category-specific dissimilarity measure between instances of the category. Ideally it is 1 for images of different instances, and 0 for images of the same object. Importantly, scores for visual identification can also be applied for other problems such as visualisation (Nowak and Jurie, 2007), recognition from a single example (Fei-Fei et al., 2006), associating names and faces in images (as done in this paper) or video (Everingham et al., 2006), or people oriented topic models (Jain et al., 2007). The face similarity measures can be learned from two types of supervision. Either a set of faces labeled by identity can be used, or a collection of face pairs that are labeled as containing twice the same person, or two different people. The similarity measures are learned on faces of a set of people that is disjoint from the set of people that are used in the people search and face naming tasks. In this manner we assure that the learned similarity measures generalize to other people, and are therefore more useful in practice.



Figure 3: Several examples of face pairs of the same person from the *Labeled Faces in the Wild* data set. There are wide variations in illumination, scale, expression, pose, hair styles, hats, make-up, etc.

In the following, we first review related work in Section 2. We present the data set that we used for our tasks in Section 3, as well as the name and face detection procedures, and our facial feature extraction procedure. We then continue in Section 4 with a discussion of several basic similarity measures between the face representations, and also detail methods to learn a similarity measure between faces from labeled data. Methods that are geared toward retrieving all the faces of a specific person are presented in Section 5. In Section 6 we describe methods that aim at establishing all name-face associations. An extensive collection of experimental results that compare the different recognition methods and face representations is then considered in Section 7. In Section 8, we end the paper by presenting our conclusions and we identify lines of further research.

## 2 Related work

Learning semantic relations from weaker forms of supervision is currently an active and broad line of research. Work along these lines includes learning correspondence between keywords and image regions (Lazebnik et al., 2003, Verbeek and Triggs, 2007), and learning image retrieval and auto-annotation with keywords (Barnard et al., 2003, Grangier et al., 2006). In these approaches, images are labeled with multiple keywords per image, requiring resolution of correspondences between image regions and semantic categories. Supervision from even weaker forms of annotation are also explored, e.g. based on images and accompanying text (Bressan et al., 2008, Jain et al., 2007), and video with scripts and subtitles (Everingham et al., 2006, Laptev et al., 2008).

The earliest work on automatically associating names and faces in news photographs is probably the PICTION system (Srihari, 1991). This system is a natural language processing system that analyzes the caption to help the visual interpretation of the picture.

The main feature of the system is that identification is performed only using face locations and spatial constraints obtained from the caption. No face similarity, description or characterization is used, although weak discriminative clues (like male vs. female) were included. Similar ideas have been successfully used in, for instance, the Name-it system (Sato et al., 1999), although their work concerned face-name association in news videos. The name extraction is done by localising names in the transcripts and video captions, and, optionally, sound track. Instead of simple still images, they extract face sequences using face tracking, so that the best frontal face of each sequence can be used for naming. These frontal faces are described using Eigenfaces method (Turk and Pentland, 1991). The face-name association can then be obtained with additional contextual cues, *e.g.* candidate names should appear just before the person appears on the video, because speeches are most often introduced by an anchor person.

Related work considering associating names to faces in a image includes the generative mixture model (Berg et al., 2004) of the facial features in a database, where a mixture component is associated with each name. The main idea of this approach is to perform a constrained clustering, where constraints are provided by the names in a document, and the assumption that each person appears at most once in each image, which rules out assignments of several faces in an image to the same name. While in practice some violations of this assumption occur, *e.g.* people that stand in front of a poster or mirror that features the same person, there are sufficiently rare to be ignored. Additionally, the names in the document provide a constraint on which names may be used to explain the facial features in the document. A Gaussian distribution in a facial feature space is associated with each name. The clustering of facial features is performed by fitting a mixture of Gaussians (MoG) to the facial features with the expectation-maximization (EM) algorithm (Dempster et al., 1977), and is analogous to the constrained k-means clustering approach of Wagstaff and Rogers (2001).

Rather than learning a mixture model over faces constrained by the names in the caption, the reverse was considered in Pham et al. (2008). They clustered face descriptors and names in a pre-processing step, after which each name and each face are both represented by an index in a corresponding discrete set of cluster indices. The problem of matching names and faces is then reduced to a discrete matching problem, which is solved using probabilistic models. The model defines correspondences between name clusters and face clusters using multinomial distributions, which are estimated using an EM algorithm.

Previous work that considers retrieving faces of specific people from caption-based supervision includes Ozkan and Duygulu (2006, 2009), and ours (Guillaumin et al., 2008, Mensink and Verbeek, 2008). These methods perform a text-based query over the captions, returning the documents that have the queried name in the caption. The faces found in the corresponding images are then further visually analyzed. The assumption underlying these methods is that the returned documents contain a large group of highly similar faces of the queried person, and additional faces of many other people appearing each just a few times. The goal is thus to find a single coherent compact cluster in a space that also contains many outliers. A graph-based method was proposed in Ozkan and Duygulu (2006): nodes represent faces, and edges encode similarity between faces. The faces in the subset of nodes with maximum density are returned as the faces representing the queried person. In Guillaumin et al. (2008), Mensink and Verbeek (2008) we extended the graph-based approach, and compared it to generative MoG approach similar to that used for face naming, and a discriminative approach that learns a classifier to recognize the person of interest.

We found performance of these methods to deteriorate strongly as the frequency of the queried person among the faces returned after the text search drops below about 40%, contradicting their underlying assumption. In this case, the faces of the queried person are obscured by many faces of other people, some of which also appear quite often due to strong co-occurrence patterns between people. To alleviate this problem, we proposed in Mensink and Verbeek (2008) a method that explicitly tries to find faces of co-occurring people and use them as ‘negative’ examples. The names of co-occurring people are found by scanning the captions that contain the person of interest, and counting which other names appear most frequently. Thus, the name co-occurrences are used to enlarge the set of faces that is visually analyzed: the initial set only contains those from images where the queried name appears, and the new set also includes those from images with co-occurring people. This is related to query expansion methods for document and image retrieval (Buckley et al., 1995, Chum et al., 2007), where query expansion is used to re-query the database to obtain more similar documents or images.

In this paper we deploy our logistic discriminant metric learning approach (LDML, Guillaumin et al. (2009b)) for these two tasks. Metric learning has received a lot of attention, for recent work in this area see *e.g.* Bar-Hillel et al. (2005), Davis et al. (2007), Globerson and Roweis (2006), Ramanan and Baker (2009), Weinberger et al. (2006), Xing et al. (2004). Most methods learn a Mahalanobis metric based on an objective function defined by means of a labelled training set, or from sets of positive (same class) and negative (different class) pairs. The difference among these methods mainly lies in their objective functions, which are designed for their specific tasks, *e.g.* clustering (Xing et al., 2004), or kNN classification (Weinberger et al., 2006). Some methods explicitly need all pairwise distances between points (Globerson and Roweis, 2006), which makes them difficult to apply in large scale applications (say more than 10000 data points). Among the existing methods, large margin nearest neighbour (LMNN) metrics (Weinberger et al., 2006) and information theoretic metric learning (ITML) (Davis et al., 2007), together with LDML, are state-of-the-art.

Metric learning is one of the numerous types of methods that can provide robust similarity measures for the problem of face and, more generally, visual identification. Recently there has been considerable interest for such identification methods (Chopra et al., 2005, Ferencz et al., 2008, Holub et al., 2008, Jain et al., 2006, Nowak and Jurie, 2007, Pinto et al., 2009, Wolf et al., 2008). It is noticeable that some of these approaches would not fit the Metric Learning framework because they do not work with a vectorial representation of faces. Instead, the similarity measure between faces is evaluated by matching low-level features between images, and this matching has to be performed for any pair of images for which we need the similarity score. Therefore, such approaches are potentially more computationally expensive.

### 3 Data sets, tasks and features

In this section, we describe the data sets we have used in our work. These data sets, *Labeled Faces in the Wild* (Huang et al., 2007b) and *Labeled Yahoo! News* (Guillaumin et al., 2010), are the result of annotation efforts on subsets of the *Yahoo! News* data set, with different tasks at mind. The former aims at developing identification methods, while the latter adds information about the structure of the data which can be used for retrieval, clustering or other tasks.

The *Yahoo! News* database was introduced by Berg et al. (2004), it was collected in 2002–2003 and consists of images and accompanying captions. There are wide

variations in appearances with respect to pose, expression, and illumination, as shown in two examples in Figure 1. Ultimately, the goal was to automatically build a large data set of annotated faces, so as to be able to train complex face recognition systems on it.

### 3.1 *Labeled Faces in the Wild*

From the *Yahoo! News* data set, the *Labeled Faces in the Wild* (Huang et al., 2007b) data set was manually built, using the captions as an aid for the human annotator. It contains 13233 face images labelled by the identity of the person. In total 5749 people appear in the images, 1680 of them appear in two or more images. The faces show a big variety in pose, expression, lighting, etc., see Figure 3 for some examples. An aligned version of all faces is available, referred to as “funneled”, which we use throughout our experiments. This data set can be viewed as a partial ground-truth for the *Yahoo! News* data set. *Labeled Faces in the Wild* has become the *de facto* standard data set for face identification, with new methods being regularly added to the comparison. The data set comes with a division in 10 parts that can be used for cross validation experiments. The folds contain between 527 and 609 different people each, and between 1016 and 1783 faces. From all possible pairs, a small set of 300 positive and 300 negative image pairs are provided for each fold. Using only these pairs for training is referred to as the “image-restricted” paradigm; in this case the identity of the people in the pairs can not be used. The “unrestricted” paradigm is used to refer to training methods that can use all available data, including the identity of the people in the images.

### 3.2 *Labeled Yahoo! News*

With growing efforts towards systems that can efficiently query data sets for images of a given person, or use the constraints given by documents to help face clustering (Guillaumin et al., 2008, Mensink and Verbeek, 2008, Ozkan and Duygulu, 2006), it has become important for the community to be able to compare those systems with a standardised data set. We therefore introduced the *Labeled Yahoo! News* data (Guillaumin et al., 2010) set and make it available online for download<sup>1</sup>. On the original *Yahoo! News* data obtained from Berg, we have applied the OpenCV implementation of the Viola-Jones face detector (Viola and Jones, 2004) and removed documents without detections. We then applied a named entity detector (Deschacht and Moens, 2006) to find names appearing in the captions, and also used the names from the *Labeled Faces in the Wild* data set as a dictionary for a caption filter to compensate some missed detections.

Our manual annotation effort on the 28204 documents that contain at least one name and one face provided each document will the following information:

1. The correct association of faces and names.
2. For faces that are not matched to a name, the annotations indicate which of the three following possibilities is the case: (i) The image is an incorrect face detection. (ii) The image depicts a person whose name is not in the caption. (iii) The image depicts a person whose name was missed by the named entity detector.
3. For names that do not correspond to a detected face, the annotation indicates whether the face is absent from the image or missed by the detector.

<sup>1</sup>Our data set is available at: <http://lear.inrialpes.fr/data/>

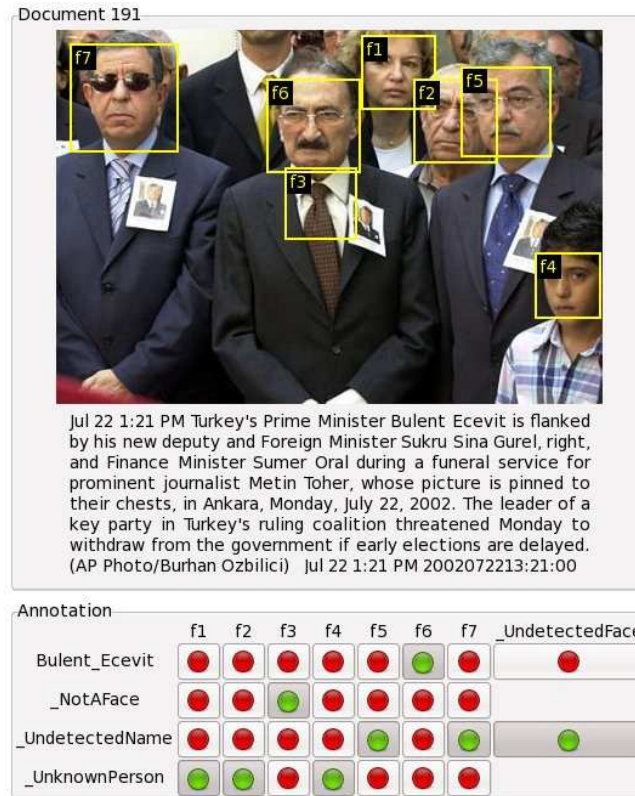


Figure 4: Example of a document in the *Labeled Yahoo! News* data set that contains faces of unknown persons, an incorrect face detection, some missed names and missed faces. Below, we show the corresponding structured manual annotation.

Finally, we also indicate if the document contains an undetected face with an undetected name. Although this information is not used in our system, it would allow for a very efficient update of the ground-truth annotations if we were to change the face detector or named entity detector. An example of annotation is shown in Figure 4.

In order to be able to use learning algorithms while evaluating on a distinct subset, we divide the data set into two completely independent sets. The *test* subset first includes the images of the 23 persons that have been used in Guillaumin et al. (2008), Mensink and Verbeek (2008), Ozkan and Duygulu (2006, 2009) for evaluating face retrieval from text-based queries. This set is extended with documents containing “friends” of these 23 persons, where friends are defined as people that co-occur in at least one document. The set of other documents, the *training* set, is pruned so that friends of friends of queried people are removed. Thus, the two sets are now independent in terms of identity of people appearing in them. 8133 documents are lost in the process.

The *test* set has 9362 documents, 14827 faces and 1071 different people: because of the specific choice of queries (namely: *Abdullah Gul*, *Roh Moo-huyn*, *Jiang Zemin*, *David Beckham*, *Silvio Berlusconi*, *Gray Davis*, *Luiz Inacio Lula da Silva*, *John Paul II*, *Kofi Annan*, *Jacques Chirac*, *Vladimir Putin*, *Junichiro Koizumi*, *Hans Blix*, *Jean Chre-*



Figure 5: Illustration of our SIFT-based face descriptor. SIFT features (128D) are extracted at 9 locations and 3 scales. Each row represents a scale at which the patches are extracted: the top row is scale 1, the middle row is scale 2 and the bottom row is scale 3. The first column shows the locations of the facial features, and the remaining nine columns show the corresponding patches on which 128D SIFT descriptors are computed. The descriptor is the concatenation of these  $3 \times 9$  SIFT features.

tien, Hugo Chavez, John Ashcroft, Ariel Sharon, Gerhard Schroeder, Donald Rumsfeld, Tony Blair, Colin Powell, Saddam Hussein, George W. Bush), it has a strong bias towards news of political events. The *training* set has 10709 documents, 16320 faces and 4799 different people: on the opposite, it contains mostly news relating to sport events. Notably, the average number of face images for each person is significantly different between the two sets.

### 3.3 Face description

Face images are extracted using the bounding box of the Viola-Jones detector and aligned using the funneling method (Huang et al., 2007a) of the *Labeled Faces in the Wild* data set. This alignment procedure finds an affine transformation of the face images so as to minimize the entropy of the image stack. On these aligned faces, we apply a facial feature detector (Everingham et al., 2006). The facial feature detector locates nine points on the face using an appearance-based model regularized with a tree-like constellation model. For each of the nine points on the face, we calculate 128 dimensional SIFT descriptors at three different scales, yielding a  $9 \times 3 \times 128 = 3456$  dimensional feature vector for each face as in Guillaumin et al. (2009b). An illustration is given in Figure 5. The patches at the nine locations and three scales overlap enough to cover the full face. Therefore, we do not consider adding other facial feature locations by interpolation as in Guillaumin et al. (2008), where 13 points were considered on a unique low scale.

There is a large variety of face descriptors proposed in the literature. This includes approaches that extract features based on Gabor filters or local binary patterns. Our work in Guillaumin et al. (2009b) showed that our descriptor performs similarly to recent optimized variants of LBP for face recognition (Wolf et al., 2008) when using standard distances. Our features are available with the data set.

## 4 Metrics for face identification

Given a vectorial representation  $\mathbf{x}_i \in \mathbb{R}^D$  of a face image (indexed by  $i$ ), we now seek to design good metrics for identification.

For both the face retrieval tasks and the face naming tasks, we indeed need to assess the similarity between two faces with respect to the identity of the depicted person. Intuitively, this means that a good metric for identification should produce small distances – or higher similarity – between face images of the same individual, while yielding higher values – or lower similarity – for different people. The metric should suppress differences due to pose, expression, lighting conditions, clothes, hair style, sun glasses while retaining the information relevant to identity. These metrics can be designed in an ad-hoc fashion, set heuristically, or learned from manually annotated data.

We restrict ourselves here to Mahalanobis metrics, which generalize the Euclidean distance. The Mahalanobis distance between  $\mathbf{x}_i$  and  $\mathbf{x}_j$  is defined as

$$d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^{\top} \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j), \quad (1)$$

where  $\mathbf{M} \in \mathbb{R}^{D \times D}$  is a symmetric positive semi-definite matrix that parametrizes the distance. Since  $\mathbf{M}$  is positive semi-definite, we can decompose it as  $\mathbf{M} = \mathbf{L}^{\top} \mathbf{L}$ . Learning the Mahalanobis distance can be equivalently performed by optimising  $\mathbf{L}$ , or  $\mathbf{M}$  directly.  $\mathbf{L}$  acts as a linear projection of the original space, and the Euclidean distance after projection equals the Mahalanobis distance defined on the original space by  $\mathbf{M}$ .

First, as a baseline, we can fix  $\mathbf{M}$  to be the identity matrix. This results simply in the Euclidean distance (L2) between the vectorial representations of the faces.

We also consider setting  $\mathbf{L}$  using principal components analysis (PCA), which has also previously been used for face recognition (Turk and Pentland, 1991). The basic idea is to find a linear projection  $\mathbf{L}$  that retains the highest possible amount of data variance. This unsupervised method improves the performance of face recognition by making the face representation more robust to noise. These projected representations can also be more compact, allowing the use of metric learning methods that scale with the square of the data dimensionality.

Metric learning techniques are methods to learn  $\mathbf{M}$  or  $\mathbf{L}$  in a supervised fashion. To achieve this, class labels of images are assumed to be known. For image  $i$ , we denote  $y_i$  its class label. Images  $i$  and  $j$  form a positive pair if  $y_i = y_j$ , and a negative pair otherwise.

In the following paragraphs, we describe three metric learning algorithms: large margin nearest neighbors (LMNN, Weinberger et al. (2006)), information theoretic metric learning (ITML, Davis et al. (2007)), and logistic discriminant based metric learning (LDML, Guillaumin et al. (2009b)). We also present an extension of LDML for supervised dimensionality reduction (Guillaumin et al., 2010).

#### 4.1 Large margin nearest neighbour metrics

Recently, Weinberger et al. (2006) introduced a metric learning method, that learns a Mahalanobis distance metric designed to improve results of  $k$  nearest neighbour (kNN) classification. A good metric for kNN classification should make for each data point the  $k$  nearest neighbours of its own class closer than points from other classes. To formalize, we define target neighbours of  $\mathbf{x}_i$  as the  $k$  closest points  $\mathbf{x}_j$  with  $y_i = y_j$ , let  $\eta_{ij} = 1$  if  $\mathbf{x}_j$  is a target neighbour of  $\mathbf{x}_i$ , and  $\eta_{ij} = 0$  otherwise. Furthermore, let

$\rho_{ij} = 1$  if  $y_i \neq y_j$ , and  $\rho_{ij} = 0$  otherwise. The objective function is

$$\begin{aligned} \varepsilon(\mathbf{M}) &= \sum_{i,j} \eta_{ij} d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) \\ &+ \sum_{i,j,l} \eta_{ij} \rho_{il} [1 + d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) - d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_l)]_+, \end{aligned} \quad (2)$$

where  $[z]_+ = \max(z, 0)$ . The first term of this objective minimises the distances between target neighbours, whereas the second term is a hinge-loss that encourages target neighbours to be at least one distance unit closer than points from other classes. The objective is convex in  $\mathbf{M}$  and can be minimised using sub-gradient methods under the constraint that  $\mathbf{M}$  is positive semi-definite, and using an active-set strategy for the constraints. We refer to metrics learned in this manner as Large Margin Nearest Neighbour (LMNN) metrics.

Rather than requiring pairs of images labelled positive or negative, this method requires labelled triples  $(i, j, l)$  of target neighbours  $(i, j)$  and points which should not be neighbours  $(i, l)$ . In practice we apply this method<sup>2</sup> using labelled training data  $(\mathbf{x}_i, y_i)$ , and implicitly use all pairs although many never appear as active constraints.

The cost function is designed to yield a good metric for kNN classification, and does not try to make all positive pairs have smaller distances than negative pairs. Therefore, directly applying a threshold on this metric for visual identification might not give optimal results but they are nevertheless very good. In practice, the value of  $k$  did not strongly influence the results. We therefore kept the default value proposed by the authors of the original work ( $k = 3$ ).

## 4.2 Information theoretic metric learning

Davis et al. (2007) have taken an information theoretic approach to optimize  $\mathbf{M}$  under a wide range of possible constraints and prior knowledge on the Mahalanobis distance. This is done by regularizing the matrix  $\mathbf{M}$  such that it is as close as possible to a known prior  $\mathbf{M}_0$ . This closeness is interpreted as a Kullback-Leibler divergence between the two multivariate Gaussian distributions corresponding to  $\mathbf{M}$  and  $\mathbf{M}_0$ :  $p(\mathbf{x}; \mathbf{M})$  and  $p(\mathbf{x}; \mathbf{M}_0)$ . The constraints that can be used to drive the optimization include those of the form:  $d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) \leq u$  for positive pairs and  $d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) \geq l$  for negative pairs, where  $u$  and  $l$  are constant values. Scenarios with unsatisfiable constraints are handled by introducing slack variables  $\boldsymbol{\xi} = \{\xi_{ij}\}$  and using a Lagrange multiplier  $\gamma$  that controls the trade-off between satisfying the constraints and using  $\mathbf{M}_0$  as metric. The final objective function equals

$$\begin{aligned} \min_{\mathbf{M} \geq 0, \boldsymbol{\xi}} & \text{KL}(p(\mathbf{x}; \mathbf{M}_0) || p(\mathbf{x}; \mathbf{M})) + \gamma \cdot f(\boldsymbol{\xi}, \boldsymbol{\xi}^0) \\ \text{s.t.} & \quad d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) \leq \xi_{ij} \quad \text{for positive pairs} \\ & \quad \text{or} \quad d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) \geq \xi_{ij} \quad \text{for negative pairs,} \end{aligned} \quad (3)$$

where  $f$  is a loss function between  $\boldsymbol{\xi}$  and target  $\boldsymbol{\xi}^0$  that contains  $\xi_{ij}^0 = u$  for positive pairs and  $\xi_{ij}^0 = l$  for negative pairs.

The parameters  $\mathbf{M}_0$  and  $\gamma$  have to be provided, although it is also possible to resort to cross-validation techniques. Usually,  $\mathbf{M}_0$  can be set to the identity matrix.

<sup>2</sup>We used code available at <http://www.weinbergerweb.net/>.

The proposed algorithm scales with  $O(CD^2)$  where  $C$  is the number of constraints on the Mahalanobis distance. Since we want to separate positive and negative pairs, we define  $N^2$  constraints of the form  $d_M(\mathbf{x}_i, \mathbf{x}_j) \leq b$  for positive pairs and  $d_M(\mathbf{x}_i, \mathbf{x}_j) \geq b$  for negative pairs, and we set  $b = 1$  as the decision threshold<sup>3</sup>. The complexity is therefore  $O(N^2D^2)$ .

### 4.3 Logistic discriminant-based metric learning

In Guillaumin et al. (2009b) we proposed a method, similar in spirit to Davis et al. (2007), that learns a metric from labelled pairs. The model is based on the intuition that we would like the distance between images in positive pairs, *i.e.* images  $i$  and  $j$  such that  $y_i = y_j$  (we note  $t_{ij} = 1$ ), to be smaller than the distances corresponding to negative pairs ( $t_{ij} = 0$ ). Using the Mahalanobis distance between two images, the probability  $p_{ij}$  that they contain the same object is defined in our model as

$$p_{ij} = p(t_{ij} | \mathbf{x}_i, \mathbf{x}_j; \mathbf{M}, b) = \sigma(b - d_M(\mathbf{x}_i, \mathbf{x}_j)), \quad (4)$$

where  $\sigma(z) = (1 + \exp(-z))^{-1}$  is the sigmoid function and  $b$  a bias term. Interestingly for the visual identification task, the bias directly works as a threshold value and is learned together with the distance metric parameters.

The direct maximum likelihood estimation of  $\mathbf{M}$  and  $b$  is a standard logistic discriminant model (Guillaumin et al., 2009b), which allows convex constraints to be applied using *e.g.* the projected gradient method (Bertsekas, 1976) or interior point methods to enforce positive semi-definiteness. This is done by performing an eigenvalue decomposition of  $\mathbf{M}$  at each iteration step, which is costly. Maximum likelihood estimation of  $\mathbf{L}$  instead of  $\mathbf{M}$  has the advantage of using simple gradient descent. Additionally,  $\mathbf{L} \in \mathbb{R}^{d \times D}$  need not be a square matrix, and in the case of  $d < D$  a supervised dimensionality reduction is performed. Therefore, in the following, we optimize  $\mathbf{L}$ , as in Guillaumin et al. (2010).<sup>4</sup>

The log-likelihood of the observed pairs  $(i, j)$ , with probability  $p_{ij}$  and binary labels  $t_{ij}$ , is

$$\mathcal{L} = \sum_{i,j} t_{ij} \log p_{ij} + (1 - t_{ij}) \log(1 - p_{ij}) \quad (5)$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{L}} = \mathbf{L} \sum_{i,j} (t_{ij} - p_{ij})(\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^\top. \quad (6)$$

When all the pairwise distances of a data set are considered, we can rewrite the gradient as

$$\frac{\partial \mathcal{L}}{\partial \mathbf{L}} = 2\mathbf{L}\mathbf{X}\mathbf{H}\mathbf{X}^\top \quad (7)$$

where  $\mathbf{X} = [\mathbf{x}_i] \in \mathbb{R}^{D \times N}$  and  $\mathbf{H} = [h_{ij}] \in \mathbb{R}^{N \times N}$  with  $h_{ii} = \sum_{j \neq i} (t_{ij} - p_{ij})$  and  $h_{ij} = p_{ij} - t_{ij}$  for  $j \neq i$ .

In Figure 6, we show the data distribution of two individuals after projecting their face descriptors on a 2D plane, comparing supervised dimensionality reduction learned on the training set of the *Labeled Yahoo! News* data set and unsupervised PCA. As we can see, supervised dimensionality reduction is a powerful tool to grasp in low-dimensional spaces the important discriminative features useful for the identification task.

<sup>3</sup>We used code available at <http://www.cs.utexas.edu/users/pjain/itml/>.

<sup>4</sup>Our code is available at <http://lear.inrialpes.fr/software/>

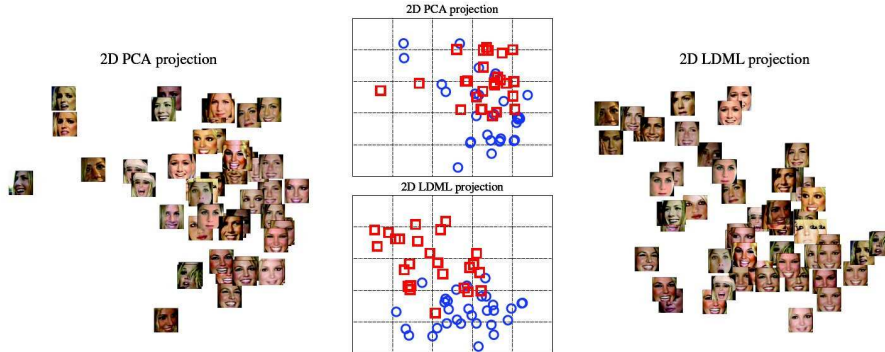


Figure 6: Comparison of PCA and LDML for 2D projections. The data of only two co-occurring persons are shown: *Britney Spears* and *Jennifer Aniston*. The identity labels given in the central part of the figure show that LDML projections better separate the two persons although the embedding seems less visually coherent than PCA.

## 5 Retrieving images of specific people

The first problem we consider is retrieving images of people within large databases of captioned news images. Typically, when searching for images of a certain person, a system (i) queries the database for captions containing the name, (ii) finds the set of faces in those images given a face detector, and (iii) ranks the faces based on (visual) similarity, so that the images of the queried person appear first in the list. An example of a system which uses the first two stages is Google Portrait (Marcel et al., 2007).

As observed in Guillaumin et al. (2008), Mensink and Verbeek (2008), Ozkan and Duygulu (2006), Sivic et al. (2009), approaches which also use the third stage generally outperform methods based only on text. The assumption underlying stage (iii) is that the faces in the result set of the text-based search consist of a large group of highly similar faces of the queried person, plus faces of many other people appearing each just a few times. The goal is thus to find a single coherent compact cluster in a space that also contains many outliers.

In the rest of this section we present methods from Guillaumin et al. (2008), Mensink and Verbeek (2008) to perform the ranking based on visual similarities. We present three methods: a graph-based method (Section 5.1), a method based on a Gaussian mixture model (Section 5.2), and a discriminant method (Section 5.3). In Section 5.4 we describe the idea of query expansion, adding faces of frequent co-occurring persons to obtain a notion of whom we are not looking for. In our experiments, we will compare these methods using similarities originating from both unsupervised and learned metrics.

### 5.1 Graph-based approach

In the graph-based approach of Guillaumin et al. (2008), Ozkan and Duygulu (2006), faces are represented as nodes and edges encode the similarity between two faces. The assumption that faces of the queried person occur relatively frequent and are highly similar, yields a search for the densest sub graph.

We define a graph  $G = (V, E)$  where the vertices in  $V$  represent faces and edges in  $E$  are weighted according to similarity  $w_{ij}$  between faces  $i$  and  $j$ . To filter our initial text-based results, we search for the densest subgraph  $S \subseteq V$ , of  $G$ , where the density  $f(S)$  of  $S$  is given by

$$f(S) = \frac{\sum_{i,j \in S} w_{ij}}{|S|}. \quad (8)$$

In Ozkan and Duygulu (2006), a greedy 2-approximate algorithm is used to find the densest component. It starts with the entire graph as subset ( $S = V$ ), and iteratively removes nodes until  $|S| = 1$ . At each iteration, the node with the minimum sum of edge weights within  $S$  is removed, and  $f(S_i)$  is computed. The subset  $S_i$  with the highest encountered density, which is at least half of the maximal density (Charikar, 2000), is returned as the densest component.

In Guillaumin et al. (2008), we have introduced a modification, to incorporate the constraint that a face is only depicted once in an image. We consider only subsets  $S$  with at most one face from each image, and initialise  $S$  with the faces that have the highest sum of edge weights in each image. The greedy algorithm is used to select a subset of these faces. However, selecting another face from an image might now yield a higher density for  $S$  than the initial choice. Consequently, we add a local search, which proceeds by iterating over the images and selecting the single face, if any, which yields the highest density. The process terminates when all nodes have been considered without obtaining further increases.

We define the weights  $w_{ij}$  following Guillaumin et al. (2008) and use the distances between the face representations to build an  $\epsilon$ -neighbour graph or a  $k$ -nearest neighbours graph. In  $\epsilon$ -graphs, weights are set to  $w_{ij} = 1$  if the distance between  $i$  and  $j$  is below a certain threshold  $\epsilon$ , and 0 otherwise. In  $k$ -nearest neighbours graphs,  $w_{ij} = 1$  if  $i$  is among the  $k$  closest points to  $j$  or vice-versa.

## 5.2 Gaussian mixture model approach

In the Gaussian mixture model approach, the search problem is viewed as a two-class clustering problem, where the Gaussian mixture is limited to just two components, *c.f.* Guillaumin et al. (2008): one foreground model representing the queried person, and one generic face model.

For each image in the result set of the text-based query, we introduce an (unknown) assignment variable  $\gamma$  to represent which, if any, face in the image belongs to the queried person. An image with  $F$  face detections has  $(F + 1)$  possible assignments: selecting one of the  $F$  faces, or none ( $\gamma = 0$ ).

Marginalizing over the assignment variable  $\gamma$ , a mixture model is obtained over the features of the detected faces  $\mathcal{F} = \{\mathbf{x}_1, \dots, \mathbf{x}_F\}$

$$p(\mathcal{F}) = \sum_{\gamma=0}^F p(\gamma)p(\mathcal{F}|\gamma), \quad (9)$$

$$p(\mathcal{F}|\gamma) = \prod_{i=1}^F p(\mathbf{x}_i|\gamma), \quad (10)$$

$$p(\mathbf{x}_i|\gamma) = \begin{cases} p_{\text{BG}}(f_i) = \mathcal{N}(\mathbf{x}_i; \mu_{\text{BG}}, \Sigma_{\text{BG}}) & \text{if } \gamma \neq i \\ p_{\text{FG}}(f_i) = \mathcal{N}(\mathbf{x}_i; \mu_{\text{FG}}, \Sigma_{\text{FG}}) & \text{if } \gamma = i \end{cases} \quad (11)$$

We use a prior over  $\gamma$  which is uniform over all non-zero assignments, i.e.  $p(\gamma = 0) = \pi$  and  $p(\gamma = i) = (1 - \pi)/F$  for  $i \in \{1, \dots, F\}$ . To reduce the number of parameters, we use diagonal covariance matrices for the Gaussians. The parameters of the generic background face model are fixed to the mean and variance of the faces in the result set of the text-based query. We estimate the other parameters  $\{\pi, \mu_{FG}, \Sigma_{FG}\}$ , using the EM algorithm. The EM algorithm is initialised in the E-step by using uniform responsibilities over the assignments, thus emphasizing on faces in documents with only a few other faces. After parameter optimization, we use the assignment maximizing  $p(\gamma|\mathcal{F})$  to determine which, if any, face represents the queried person.

### 5.3 Discriminant method

The motivation for using a discriminant approach is to improve over generative approaches like the Gaussian mixture, while avoiding to explicitly compute the pairwise similarities as in Guillaumin et al. (2008), Ozkan and Duygulu (2006), which is relatively costly when the query set contains many faces. We chose to use sparse multinomial logistic regression (SMLR, Krishnapuram et al. (2005)) since we are using high-dimensional face features.

Still denoting features with  $\mathbf{x}$ , and class labels with  $y \in \{FG, BG\}$ , the conditional probability of  $y$  given  $\mathbf{x}$  is defined as a sigmoid over linear score functions

$$p(y = FG|\mathbf{x}) = \sigma(w_{FG}^T \mathbf{x}), \quad (12)$$

where  $\sigma(\cdot)$  is defined as in Section 4.3. The likelihood is combined with a Laplace prior which promotes the sparsity of the parameters:  $p(w) \propto \exp(-\lambda \|w\|_1)$ , where  $\|\cdot\|_1$  denotes the  $L_1$  norm, and  $\lambda$  is set by cross-validation.

To learn the weight vectors we use the noisy set of positive examples ( $y = FG$ ) from the result set of the text-based query and a random sample of faces from the databases as negative examples ( $y = BG$ ). To take into account that each image in the query may contain at most one face of the queried person, we alter the learning procedure as follows. We learn the classifier iteratively, starting with all faces in the result set of the text-based query as positive examples, and at each iteration transferring the faces that are least likely to be the queried person from the positive to the negative set. At each iteration we transfer a fixed number of faces, which could involve several faces from a document as long as there remains at least one face from each document in the positive set. The last condition is necessary to avoid that a trivial classifier will be learned that classifies all faces as negative.

Once the classifier weights have been learned, we score the  $(F + 1)$  assignments with the log-probability of the corresponding classifier responses, e.g. for  $\gamma = 1$  the score would be  $\ln p(y_1 = FG|\mathbf{x}_1) + \sum_{i=2}^F \ln p(y_i = BG|\mathbf{x}_i)$ .

### 5.4 Query expansion

Using ideas from query expansion, the search results can be considerably improved, as we showed in Mensink and Verbeek (2008). The query expansion framework brings us somehow closer to the complete name-face association problem discussed in Section 6. The underlying observation is that errors in finding the correct faces come from the confusion with co-occurring people.

For example, suppose that in captions for the query *Tony Blair* the names *George Bush* and *Gordon Brown* occur often. By querying the system for *George Bush* and

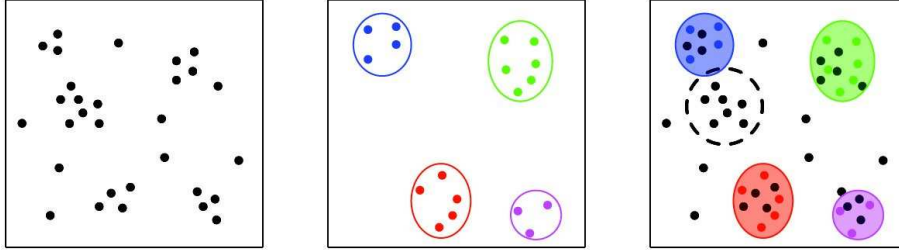


Figure 7: Schematic illustration of how friends help to find people. The distribution of face features obtained by querying captions for a name (left), the query expansion with color coded faces of four people that co-occur with the queried person (middle), and how models of these people help to identify which faces in the query set are not the queried person (right).

*Gordon Brown* we can then rule out faces in the result set from the text-based query for *Tony Blair* that are very similar to the faces returned for *George Bush* or *Gordon Brown*. See Figure 7 for a schematic illustration of the idea.

We therefore extend the result set of the text-based query by querying the database for names that appear frequently together with the queried person; we refer to these people as “friends” of the queried person. For each friend we use only images in which the queried person does not appear in the caption. We use at most 15 friends for a query, and for each friend there should be at least 5 images. There is no obvious answer how exactly to exploit this idea in the graph-based approach, so below we describe its use in the Gaussian mixture and discriminative approaches only.

#### 5.4.1 Query expansion for Gaussian mixture filtering

The first way to use the query expansion in the Gaussian mixture model is to fit the background Gaussian to the query expansion instead of the query set. So the background Gaussian will be biased towards the “friends” of the queried person, and the foreground Gaussian is less likely to lock into one of the friends.

The second way to use query expansion, is to create a mixture background model, this forms a more detailed query-specific background model. For each friend  $n$  among the  $N$  friends, we apply the method without query expansion while *excluding* images that contain the queried person in the caption. These “friend” foreground Gaussians are added to the background mixture, and we include an additional background Gaussian

$$p_{\text{BG}}(f) = \frac{1}{N+1} \sum_{n=0}^N \mathcal{N}(\mathbf{x}; \mu_n, \Sigma_n), \quad (13)$$

where  $n = 0$  refers to the generic background model. We proceed as before, with a fixed  $p_{\text{BG}}$  and using the EM algorithm to find  $p_{\text{FG}}$  and the most likely assignment  $\gamma$  in each image.

#### 5.4.2 Query expansion for linear discriminant filtering

The linear discriminant method presented in Section 5.3 uses a random sample from the database as negative examples to discriminate from the (noisy) positive examples in

the query set. The way we use query expansion here is to replace this random sample with faces found when querying for friends. When there are not enough faces in the expansion set (we require at least as many faces as the dimensionality to avoid trivial separation of the classes), we use additional randomly selected faces.

## 6 Associating names and faces

In this section we consider associating names to all the faces in a database of captioned news images. For each face we want to know to which name in the caption it corresponds, or possibly that it corresponds to none of them: a *null* assignment. In this setting, we can use the following constraints: (i) a face can be assigned to at most one name, (ii) this name must appear in the caption, and (iii) a name can be assigned to at most one face in a given image.

This task can be thought of as querying simultaneously for each name using a single-person retrieval method which would comply with (ii) and (iii). But doing so in a straightforward manner, the results could violate constraint (i). This approach would also be computationally expensive if the data set contains thousands of different people, since each face is processed for each query corresponding to the names in the caption. Another benefit of resolving all name-face associations together is that it will better handle the many people that appear just a few times in the database, say less than 5. For such rare people, the methods in Section 5 are likely to fail as there are too few examples to form a clear cluster in the feature space.

Moreover, the discriminative approach for retrieval is impractical to adapt here. A straightforward model would replace Equation 12 with a multi-class soft-max. This would imply learning  $D$  weights for each of the classes, *i.e.* people. For rare people, this approach is likely to fail.

Below, we describe the graph-based approach presented in Guillaumin et al. (2008) in Section 6.1, and the constrained mixture modeling approach of Berg et al. (2004) in Section 6.2. Both methods try to find a set  $S_n$  of faces to associate to each name  $n$ , the task is therefore seen as a constrained clustering problem.

### 6.1 Graph-based approach

In the graph-based approach to single-person face retrieval, the densest subgraph  $S$  was searched in the similarity graph  $G$  obtained from faces returned by the text-based query. We extend this as follows: the similarity graph  $G$  is now computed considering all faces in the dataset. In this graph, we search simultaneously for all subgraphs  $S_n$  corresponding to names, indexed by  $n$ .

As already noted, the number of example faces for different people varies greatly, from just one or two to hundreds. As a result, optimising the sum of the densities of subgraphs  $S_n$  leads to very poor results, as shown in Guillaumin et al. (2008). Using the sum of the densities tends to assign an equal number of faces to each name, as far as allowed by the constraints, and therefore does not work well for very frequent and rare people. Instead we maximise the sum of edge weights within each subgraph

$$F(\{S_n\}) = \sum_n \sum_{i,j \in S_n} w_{ij}. \quad (14)$$

Note that when  $w_{ii} = 0$  this criterion does not differentiate between empty clusters and clusters with a single face. To avoid clusters with a single associated face, for which

there are no other faces to corroborate the correctness of the assignment, we set  $w_{ii}$  to small negative values.

Then, the subgraphs  $S_n$  can be obtained concurrently by directly maximizing Eq. (14), while preserving the image constraints. Finding the optimal global assignment is computationally intractable, and we thus resort to approximate methods. The subgraphs are initialized with all faces that could be assigned, thus temporarily relaxing constraint (i) and (iii), but keeping (ii). Then we iterate over images and optimise Eq. (14) per image. As a consequence, (i) and (iii) are progressively enforced. After a full iteration over images, constraints (i), (ii) and (iii) are correctly enforced. The iteration continues until a fixed-point is reached, which takes in practice 4 to 10 iterations.

The number of admissible assignments for a document with  $F$  faces and  $N$  names is  $\sum_{p=0}^{\min(F,N)} p! \binom{F}{p} \binom{N}{p}$ , and thus quickly becomes impractically large. For instance, our fully-labeled data set contains a document with  $F = 12$  faces and  $N = 7$  names, yielding more than 11 million admissible assignments. Notably, the five largest documents account for more than 98% of the number of admissible assignments to be evaluated over the full dataset.

Given the fact that assignments share many common sub-assignments, a large efficiency gain can be expected by not re-evaluating the shared sub-assignments. We therefore introduced in Guillaumin et al. (2008) a reduction of the optimisation problem to a well-studied minimum cost matching in a weighted bipartite graph (Cormen et al., 2001). This modelling takes advantage of this underlying structure and can be implemented efficiently. Its use is limited to objectives that can be written as a sum of “costs”  $c(f, n)$  for assigning face  $f$  to name  $n$ . The corresponding graphical representation is shown in Figure 8.

The names and faces problem differs from usual bipartite graph matching problem because we have to take into account *null* assignments, and this *null* value can be taken by any number of faces in a document. This is handled by having as many *null* nodes as there are faces and names. A face  $f$  can be paired with any name or its own copy of *null*, which is written  $\bar{f}$ , and reciprocally, a name  $n$  can be paired with any face or its own copy of *null*, written  $\bar{n}$ . A pairing between  $f$  and  $n$  will require the pairing of  $\bar{n}$  and  $\bar{f}$  because of document constraints. The weights of the pairings are simply the costs of assigning a face  $f_i$  to the subgraph  $S_n$ , i.e.  $-\sum_{f_j \in S_n} w_{ij}$ , or to *null*.

A bipartite graph matching problem is efficiently solved using the Kuhn-Munkres algorithm (also known as the Hungarian algorithm) which directly works on a cost matrix. The cost matrix modeling our document-level optimization is a squared matrix with  $n + f$  rows and columns where the absence of edge is modeled with infinite cost. The rows represent faces and *null* copies of names, while columns represent names and *null* copies of faces. See Figure 9 for a example cost matrix modeling our matching problem. It is then straightforward to obtain the minimum cost and the corresponding assignment, as highlighted in the example matrix.

In Figure 10 we show how the processing time grows as a function of the number of admissible assignments in a document for the Kuhn-Munkres algorithm compared to a “brute-force” loop over all admissible assignments. For reference, we also include the min-cost max-flow algorithm of Guillaumin et al. (2008), but it is slower than Kuhn-Munkres because the solver is more general than bipartite graph matching.

## 6.2 Gaussian mixture model approach

In order to compare to previous work on naming faces in news images (Berg et al., 2004), we have implemented a constrained mixture model approach similar to the gen-

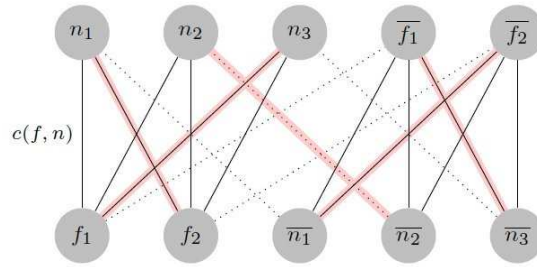


Figure 8: Example of the weighted bipartite graph corresponding to a document with two faces and three names. For clarity, costs are not indicated, and edges between vertices and their *null* copies are dotted. An example of a matching solution is given with the highlighted lines, it is interpreted as assigning face  $f_1$  to name  $n_3$ ,  $f_2$  to  $n_1$ , and not assigning name  $n_2$ .

$$\begin{bmatrix} c(f_1, n_1) & c(f_1, n_2) & c(f_1, n_3) & c(f_1, \bar{f}_1) & \infty \\ c(f_2, n_1) & c(f_2, n_2) & c(f_2, n_3) & \infty & c(f_2, \bar{f}_2) \\ c(\bar{n}_1, n_1) & \infty & \infty & c(\bar{n}_1, \bar{f}_1) & c(\bar{n}_1, \bar{f}_2) \\ \infty & c(\bar{n}_2, n_2) & \infty & c(\bar{n}_2, \bar{f}_1) & c(\bar{n}_2, \bar{f}_2) \\ \infty & \infty & c(\bar{n}_3, n_3) & c(\bar{n}_3, \bar{f}_1) & c(\bar{n}_3, \bar{f}_2) \end{bmatrix}$$

Figure 9: Example of the  $5 \times 5$  cost matrix representing the bipartite graph matching formulation of document-level optimization for the Kuhn-Munkres algorithm, for a document with two faces and three names. The costs  $c(f_i, n_j)$  are set to the negative sum of similarities from  $f_i$  to vertices in the subgraph  $S_{n_j}$ ,  $c(f_i, \bar{f}_i)$  are set to a constant threshold value  $\theta$ , and  $c(\bar{n}_j, \cdot)$  are set to zero. For  $c(\bar{n}_j, n_j)$ , this is because we do not model any preference for using or not certain subgraphs. Infinite costs account for absence of vertex. The same solution as in Figure 8 is highlighted.

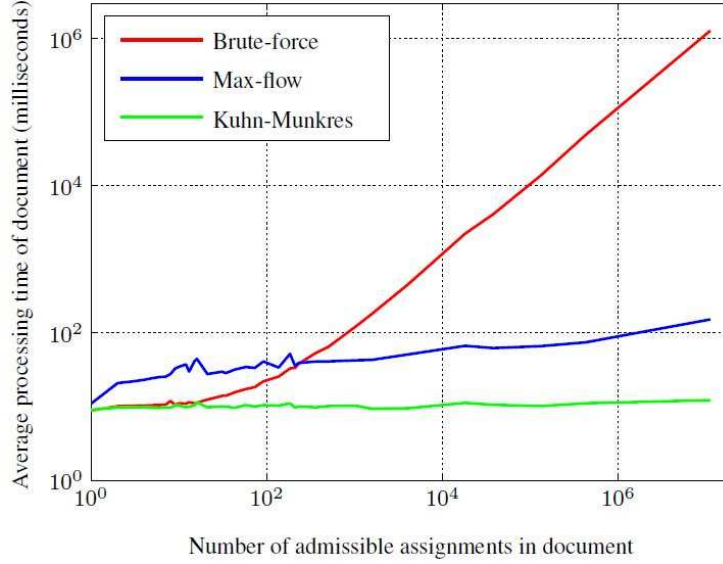


Figure 10: Average processing time of the three algorithms with respect to the number of admissible assignments in documents. The average is computed over 5 runs of random costs, and over all documents that have the same number of admissible assignments. The Kuhn-Munkres algorithm combines low overhead and slow growth with document complexity. Note that there is a log scale on both axes.

erative model presented in Section 5.2. We associate a Gaussian density in the feature space with each name, and an additional Gaussian is associated with *null*. The parameters of the latter will be fixed to the mean and variance of the ensemble of all faces in the data set, while the former will be estimated from the data. The model for an image with faces  $\mathcal{F} = \{\mathbf{x}_1, \dots, \mathbf{x}_F\}$  is the following

$$p(\mathcal{F}) = \sum_{\gamma} p(\gamma)p(\mathcal{F}|\gamma) \quad (15)$$

$$p(\mathcal{F}|\gamma) = \prod_{i=1}^F p(\mathbf{x}_i|\gamma) \quad (16)$$

$$p(\mathbf{x}_i|\gamma) = \mathcal{N}(\mathbf{x}_i; \mu_n, \Sigma_n) \quad (17)$$

where  $n$  is the name (or *null*) as given by the assignment  $(\mathbf{x}_i, n) \in \gamma$ . Given the assignment we have assumed the features  $\mathbf{x}_i$  of each face  $f_i$  to be independently generated from the associated Gaussian. The prior on  $\gamma$  influences the preference of *null* assignments. Using parameter  $\theta \in \mathbb{R}$ , we define

$$p(\gamma) = \frac{\exp(-n_{\gamma}\theta)}{\sum_{\gamma'} \exp(-n_{\gamma'}\theta)} \propto \exp(-n_{\gamma}\theta) \quad (18)$$

where  $n_{\gamma}$  is the number of *null* assignments in  $\gamma$ . For  $\theta = 0$ , the prior is uniform over the admissible assignments.

We use Expectation-Maximisation to learn the maximum likelihood parameters  $\mu_n$ ,  $\Sigma_n$  and  $\gamma$  from the data. This requires computing the posterior probability  $p(\gamma|\mathcal{F})$  for

each possible assignment  $\gamma$  for each image in the E-step, which is intractable. Instead, we constrain the E-step to selecting the assignment with maximum posterior probability. This procedure does not necessarily lead to a local optimum of the parameters, but is guaranteed to maximize a lower bound on the data likelihood (Neal and Hinton, 1998). Moreover, compared to an expected assignment, the a posteriori maximum likelihood assignment defines a proper naming of the faces in the documents.

This model is straightforwardly framed into the bipartite graph matching formulation. The costs  $c(f, n)$  are set to  $-\ln \mathcal{N}(\mathbf{x}; \mu_n, \Sigma_n)$ , where  $\mathbf{x}$  represents face  $f$  in the feature space, and the cost of not associating a face to a name is  $c(f, \bar{f}) = -\ln \mathcal{N}(\mathbf{x}; \mu_{null}, \Sigma_{null}) + \theta$ . Null assignments are favored as  $\theta$  decreases.

The generative model in Berg et al. (2004) incorporates more information from the caption. We leave this out here, so we can compare directly with the graph-based method. Caption features can be incorporated by introducing additional terms that favor names of people who are likely to appear in the image based on textual analysis, see *e.g.* Jain et al. (2007).

## 7 Experimental results

We present our experimental results in three parts. In the first, we use the *Labeled Faces in the Wild* data set to study the influence of parameters of the face descriptor and learned similarity measures. Then, using our *Labeled Yahoo! News* data set, we evaluate our different methods for retrieval of faces, and associating names and faces. In these experiments, we also consider the impact of using learned metrics for these tasks.

### 7.1 Metrics for face similarity

In this section we analyse the performance of our face descriptor with respect to its main parameters. This is done on *Labeled Faces in the Wild*, to avoid overfitting on our data set and tasks. Evaluation on the *Labeled Faces in the Wild* data set is done in the following way. For each of the ten folds defined in the data set, the distance between the 600 pairs is computed after optimizing it on the nine other folds, when applicable. This corresponds to the “unrestricted” setting, where the faces and their identities are used to form all the possible negative and positive pairs. The Equal Error Rate of the ROC curve over the ten folds is then used as accuracy measure, see Huang et al. (2007b).

The following parameters are studied:

1. *The scales of the descriptor.* We compare the performance of each individual scale (see Figure 5) independently, and their combination.
2. *The dimensionality of the descriptor.* Except for the Euclidean distance, using more than 500 dimensions is impractical, since metric learning involves algorithms that scale as  $O(D^2)$  where  $D$  is the data dimensionality. Moreover, we can expect to overfit when trying to optimize over a large number of parameters. Therefore, we compared in Figure 11 the performance of metric learning algorithms by first reducing the data dimensionality using PCA, to 35, 55, 100, 200 and 500 dimensions. LDML is also able to learn metrics with this reduced dimensionality directly.

3. *Metrics for the descriptor.* We compare the following measures: Euclidean distance (L2), Euclidean distance after PCA (PCA-L2), LDML metric after PCA (PCA-LDML), LMNN metric after PCA (PCA-LMNN), ITML metric after PCA (PCA-ITML), and finally Euclidean distance after low-rank LDML projection (LDML-L2).

In Figure 11, we present the performance on *Labeled Faces in the Wild* of the different metrics for each individual scales of the descriptor, as a function of the data dimensionality. As a first observation, we note that all the learned metrics perform much better than the unsupervised metrics like L2 and PCA-L2. The difference of performance between learned metrics is smaller than the gap between learned metrics and unsupervised ones.

When comparing performance obtained with the different scales, we see that scales 2 and 3 perform similarly, and better than scale 1. The combination of the scales brings an improvement over the individual scales.

From Figure 11, we also observe that metric learning methods benefit from pre-processing with larger PCA dimensionalities up to 200 dimensions. For low dimensionalities, the methods are limited by the weak discriminative power of PCA. We can observe a hierarchy of methods: PCA-LDML performs better than PCA-LMNN, which itself performs better than PCA-ITML. But the difference is rarely more than 2% between PCA-ITML and PCA-LDML below 200 dimensions. Performances seem to decrease when the data dimensionality is above 200, which might be due to overfitting. For ITML, the drop can be explained by unoptimized code which required early stopping in the optimisation. Keeping 100 to 200 PCA dimensions appears as a good trade-off between dimensionality reduction and discriminative power. When using LDML for supervised dimensionality reduction, the performance is maintained at a very good level when the dimension is reduced, and typically LDML-L2 is the best performing method in low dimensions.

The performance of LDML-L2 for dimensionalities ranging from 1 to 500 can be seen in Figure 12, with an illustration already shown in Figure 6. We show the influence of target space dimensionality on performance for the best scale (the third), the two best scales (second and third) and all three scales together. We can clearly observe that combining scales benefits the performance, at the expense of a higher dimensional input space. Notably, adding scale 1 does not seem to have any significant effect on performance.

In the rest of the experiments, we will use the descriptor composed of scale 2 and 3 only, because it is 2304D compared to 3456D for the full descriptor, without any loss of performance. In the following, we compare the performance of the raw descriptor to 100D PCA and LDML projections for the two tasks considered in the paper.

## 7.2 Experiments on face retrieval

In this section we describe the experiments for face retrieval of a specific person. We use the training set of *Labeled Yahoo! News* to obtain PCA and LDML projections for the data, apply them to the test set and query for the 23 person mentioned in Section 3.2.

In our experiments we compare the original features (L2-2304D), PCA with 100D and LDML with 100D. We evaluate the methods using the mean Average Precision (mAP), over the 23 queries.

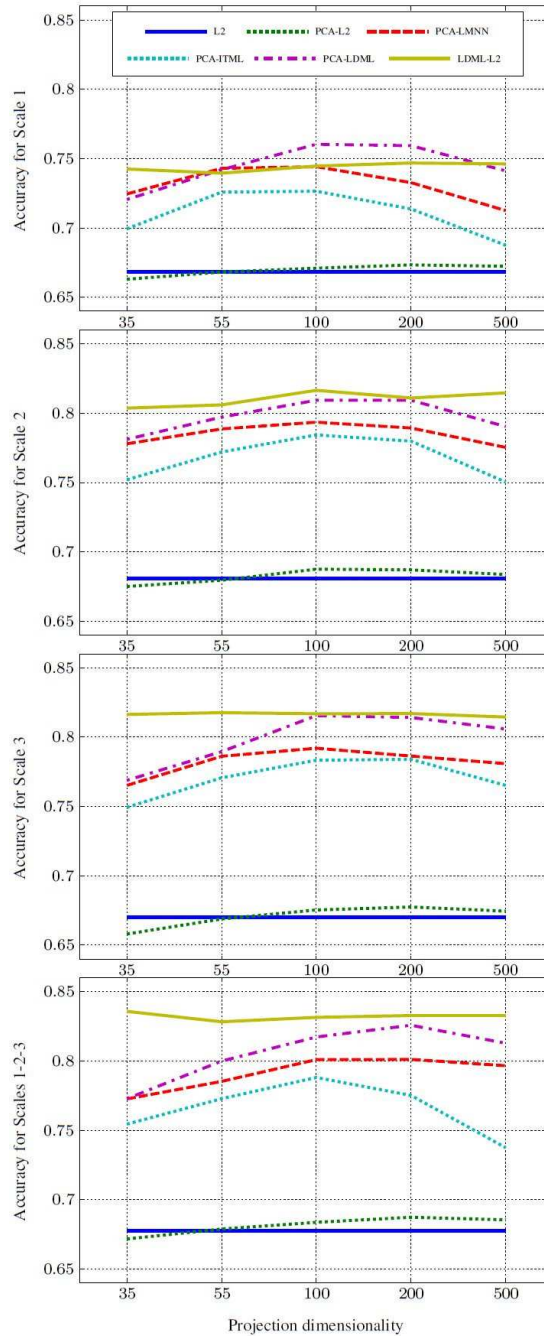


Figure 11: Comparison of methods for the three scales of the face descriptor and the concatenated descriptor of all three scales. We show the accuracy of the projection methods with respect to the dimensionality, except for L2 where it is irrelevant. Scales 2 and 3 appear more discriminative than scale 1 using learned metrics, and the concatenation brings an improvement. Except for scale 1, LDML-L2 performs best on a wide range of dimensionalities.

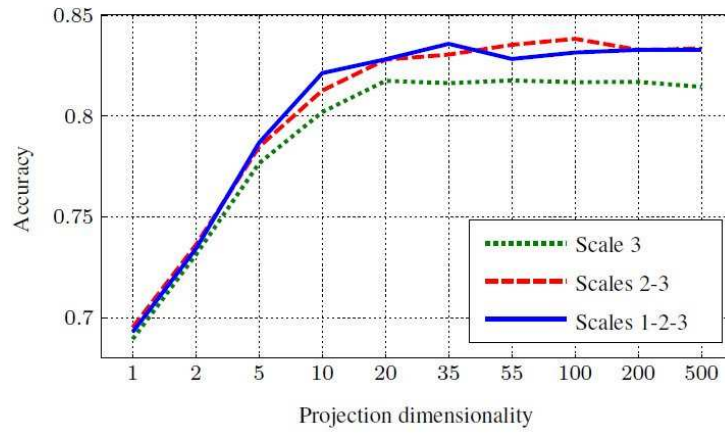


Figure 12: Accuracy of LDML projections over a wide range of space dimensionalities, for scale 3, the combination of scale 2 and 3, and the three scales.

	L2-2304D	PCA-100D	LDML-100D
<b>SMLR model</b>			
Random set	89.1	86.1	88.3
Expansion set	88.8	86.6	88.6
<b>Generative model</b>			
Query set	69.4	85.0	91.3
Expansion set	70.7	85.6	91.5
Friends as Mixture	79.6	91.9	95.3
<b>Graph-based</b>			
eps	74.5	73.6	87.0
kNN	74.9	77.1	85.5

Table 1: In this table we give an overview of the mAP scores over 23 queries for the different methods and features.

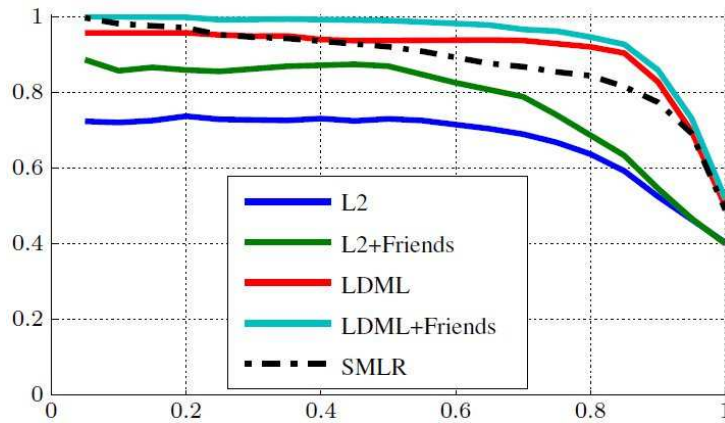


Figure 13: Precision (y-axis) versus Recall (x-axis) of the Generative Methods using Friends or not, and using LDML or L2. For comparison we also show the SMLR method.

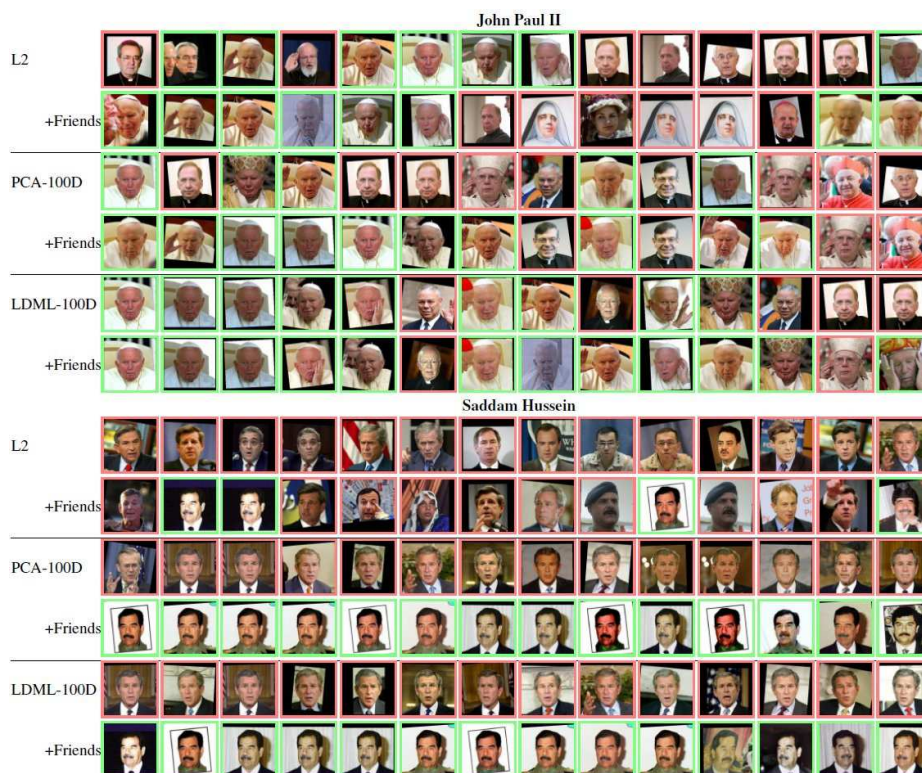


Figure 14: First fourteen retrieved faces for the queries *John-Paul II* (top) and *Saddam Hussein* (bottom) using the generative approach. We highlight in green the correctly retrieved faces and in red the incorrect ones. This shows the merit of metric learning for most queries and illustrate the necessity of modelling friends for difficult queries.

In Table 1 we show the results of the described methods, using the 3 different similarity measures. We observe that the SMLR model obtains the best performance on the original face descriptor, and its performance is only slightly modified when using dimensionality reduction techniques. This can be explained by the fact that the SMLR model itself is finding which dimensions to use, and both PCA and LDML have less dimensions to select from.

We further observe that the generative method benefits from both dimension reduction techniques, the performance of the standard method increases by approximately 15% using PCA, and around 22% using LDML. Although PCA is an unsupervised dimensionality reduction scheme, the increase in performance can be explained by the reduced number of parameters that has to be fit and decorrelating the variables. The best scoring method is the generative method using a background consisting of a mixture of friends, with LDML features. This constitutes an interesting combination of the discriminatively learned LDML features with a generative model.

Finally, in Table 1, we see that the graph-based method also greatly takes advantage of LDML features, whereas PCA dimensionality reduction performs similarly to L2.

In Figure 13, we show the precision for several levels of recall, again averaged over the 23 queries. The improvement by using LDML is made again clear, there is an improvement of more than 20% in precision for recall levels up to 90%.

In Figure 14, we show the retrieval results for the generative approach using PCA or LDML, with or without modelling friends. We observe that on a query like *John-Paul II*, LDML offers better results than PCA. Modelling friends helps PCA reach the performance of LDML. The friends extension is mainly advantageous for the most difficult queries. From the faces retrieved by the text-based query for *Saddam Hussein*, the majority is in fact from George Bush. Using LDML, it is not surprising that the model focuses even more strongly on images of Bush. Using friends, however, we specifically model George Bush to suppress its retrieval, and so we are able to find the faces of Saddam Hussein.

### 7.3 Experiments on names and faces association

For solving all names and faces associations in images, we also use the training and test sets. We learn the similarity measures using LDML and PCA on the training set. Then, we apply on the test set the methods described in Section 6 and measure their performance. We call the performance measure we use the “naming precision”. It measures the ratio between the number of correctly named faces over the total number of named faces. Recall that some faces might not be named by the methods (*null*-assignments).

Concerning the definition of weights for the graph, we found that using  $w_{ij} = \theta - d(\mathbf{x}_i, \mathbf{x}_j)$  yields more stable results than the binary weights obtained using  $\theta$  as a hard threshold for the distance value. This is simply because the thresholding process completely ignores the differences between values if they fall on the same side of the threshold. The value of  $\theta$  influences the preference of *null* assignments. If  $\theta$  is high, faces are more likely to have positive weights with many faces in a cluster, and therefore is more likely to be assigned to a name. At the opposite, with a small  $\theta$ , a given face is more likely to have negative similarities with most faces in admissible clusters, and therefore is less likely to be associated to any name. Similarly, we can vary the parameter  $\theta$  of the prior for the generative approach as given in Eq. (18). For both approaches, we plot the naming precision for a range of possible number of named

faces. This is done by exploring the parameter space in a dichotomic way to obtain fifty points in regular intervals.

In Figure 16, we show the performance of the graph-based approach (Graph) compared to the generative approach of mixture of Gaussians (Gen.) for 100 dimensional data, obtained either by PCA or by LDML. We also show the performance of L2, *i.e.* the Euclidean distance for the graph and the original descriptor for the generative approach.

We can first observe that PCA is comparable to the Euclidean distance for the graph-based approach. This is expected since PCA effectively tries to minimize the data reconstruction error. The generative approach benefits from the reduced number of parameters to set when using PCA projections, and therefore PCA is able to obtain better clustering results, up to 10 points when naming around 5000 faces. We also observe that LDML performs always better than its PCA counterpart for any given method. The increase in performance is most constant for the generative approach, for which the precision is approximately 10 points higher. For the graph-based approach, up to 16 points are gained around 8700 named faces but the difference is smaller at the extremes. This is because the precision is already high with L2 and PCA when naming few faces. When naming almost all faces, the parameter  $\theta$  of the graph-based method is too high so that most faces are considered similar. Therefore the optimisation process favors the largest clusters when assigning faces, which decreases the performance of all graph-based approaches.

For both projection methods and for the original descriptor, the graph-based approach performs better than the generative approach when fewer faces are named, whereas the generative approach outperforms the graph-based when more faces are named. The latter observation has the same explanation as above: the performance of graph-based methods decreases when it names too many faces. The former was expected: when too few faces are assigned to clusters, the estimation of the corresponding Gaussian parameters are less robust, thus leading to decreased performance.

Finally, in Table 2, we show the number of correct and incorrect associations obtained by the different methods, using the parameter that leads to the maximum number of correctly associated names and faces. In Figure 15, we show qualitative results for the comparison between LDML-100d and PCA-100d for our graph-based naming procedure. These difficult examples show how LDML helps detecting null-assignments and performs better than PCA for selecting the correct association between faces and names.

## 8 Conclusions

In this paper, we have successfully integrated our LDML metric learning technique (Guillaumin et al., 2009b) to improve performance of text-based image retrieval of people (Guillaumin et al., 2008, Mensink and Verbeek, 2008, Ozkan and Duygulu, 2006), and names and faces association in news photographs (Berg et al., 2004, Guillaumin et al., 2008).

Using the well studied *Labeled Faces in the Wild* data set (Huang et al., 2007b), we have conducted extensive experiments in order to compare metric learning techniques for face identification and study the influence of the parameters of our face descriptor. These experiments extend and improve over Guillaumin et al. (2009b).

In order to measure the performance of our retrieval and assignment techniques, we have fully annotated a data set of around 20000 documents with more than 30000

	<p>LDML</p> <ol style="list-style-type: none"> <li>1. George W. Bush</li> <li>2. <b>null</b></li> <li>3. <b>Tony Blair</b></li> </ol> <hr/> <p>PCA</p> <ol style="list-style-type: none"> <li>1. George W. Bush</li> <li>2. Junichiro Koizumi</li> <li>3. <b>Tony Blair</b></li> </ol>
	<p>LDML</p> <ol style="list-style-type: none"> <li>1. <b>null</b></li> <li>2. <b>Natalie Maines</b></li> <li>3. <b>Emily Robison</b></li> <li>4. <b>Martie Maguire</b></li> </ol> <hr/> <p>PCA</p> <ol style="list-style-type: none"> <li>1. <b>null</b></li> <li>2. <b>Natalie Maines</b></li> <li>3. Martie Maguire</li> <li>4. Emily Robison</li> </ol>
	<p>LDML</p> <ol style="list-style-type: none"> <li>1. <b>null</b></li> <li>2. <b>Tony Blair</b></li> <li>3. <b>Jiang Zemin</b></li> </ol> <hr/> <p>PCA</p> <ol style="list-style-type: none"> <li>1. David Kelly</li> <li>2. <b>Tony Blair</b></li> <li>3. <b>Jiang Zemin</b></li> </ol>
	<p>LDML</p> <ol style="list-style-type: none"> <li>1. Saddam Hussein</li> <li>2. <b>John Warner</b></li> <li>3. <b>Paul Bremer</b></li> </ol> <hr/> <p>PCA</p> <ol style="list-style-type: none"> <li>1. <b>Bill Frist</b></li> <li>2. Paul Bremer</li> <li>3. Saddam Hussein</li> </ol>

Figure 15: Four document examples with their naming results for LDML-100d and PCA-100d when the maximum number of correctly associated names and faces is reached. The correct associations are indicated in bold. On these examples, the names that can be used for association with the faces are all shown: they were used by LDML or PCA, or both. Typically, LDML is better at detecting null-assignments and is more precise when associating a face to a name.

	PCA-100d	LDML-100d
Graph-based		
Correct: name assigned	6585	7672
Correct: no name assigned	3485	4008
Incorrect: not assigned to name	1007	1215
Incorrect: wrong name assigned	3750	1932
Generative model		
Correct: name assigned	8327	8958
Correct: no name assigned	2600	2818
Incorrect: not assigned to name	765	504
Incorrect: wrong name assigned	3135	2547

Table 2: Summary of names and faces association performance obtained by the different methods when the maximum number of correctly associated names and faces is reached.

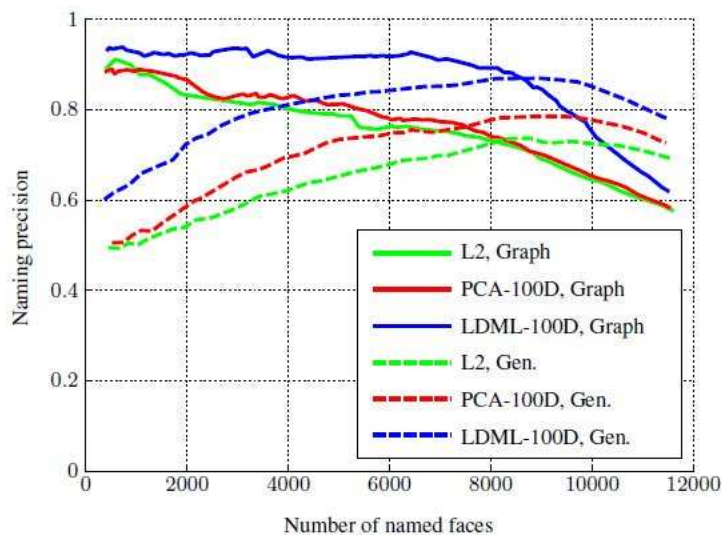


Figure 16: Precision of LDML and PCA-L2 with respect to the number of assigned faces, obtained by varying the threshold, for 100 dimensions.

faces (Guillaumin et al., 2010). This data set is publicly available for fair and standardised future comparison with other approaches.

Using this data set, we have shown that metric learning improves both graph-based and generative approaches for both tasks. For face retrieval of persons, we have improved the mean average precision of the graph-based approach from 77% using PCA projection to more than 87% using LDML. Using the metric learning projection, the performance reaches 95% when using a generative approach that also models people frequently co-occurring with the queried person, compared to 80% with the original descriptor.

For names and faces association, we have attained precision levels above 90% with the graph-based approach, and around 87% for the generative approach, which is in both cases 6 points above the best score obtained using PCA. Since these maxima are attained for different numbers of named faces, the generative approach is in fact able to correctly name a larger number of faces, up to almost 9000 faces.

In future work, we plan to use the caption-based supervision to alleviate the need for manual annotation for metric learning. This could be obtained by using the face naming process for automatically annotating the face images, or by casting the problem in a multiple instance learning framework.

## References

- Bar-Hillel, A., Hertz, T., Shental, N., Weinshall, D.: Learning a Mahalanobis metric from equivalence constraints. *JMLR* **6**, 937–965 (2005)
- Barnard, K., Duygulu, P., Forsyth, D., de Freitas, N., Blei, D., Jordan, M.: Matching words and pictures. *JMLR* **3**, 1107–1135 (2003)
- Bekkerman, R., Jeon, J.: Multi-modal clustering for multimedia collections. In: *CVPR* (2007)
- Berg, T., Berg, A., Edwards, J., Maire, M., White, R., Teh, Y., Learned-Miller, E., Forsyth, D.: Names and faces in the news. In: *CVPR* (2004)
- Berg, T., Forsyth, D.: Animals on the web. In: *CVPR* (2006)
- Bertsekas, D.: On the Goldstein-Levitin-Polyak gradient projection method. *IEEE Transactions on Automatic Control* **21**(2), 174–184 (1976)
- Bressan, M., Csurka, G., Hoppenot, Y., Renders, J.: Travel blog assistant system. In: *Proceedings of the International Conference on Computer Vision Theory and Applications* (2008)
- Buckley, C., Salton, G., Allan, J., Singhal, A.: Automatic query expansion using SMART: TREC 3. In: *Proceedings of the Text Retrieval Conference*, pp. 69–80 (1995)
- Charikar, M.: Greedy approximation algorithms for finding dense components in a graph. In: *Proceedings of International Workshop Approximation Algorithms for Combinatorial Optimization*, pp. 139–152 (2000)
- Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: *CVPR* (2005)
- Chum, O., Philbin, J., Sivic, J., Isard, M., Zisserman, A.: Total recall: Automatic query expansion with a generative feature model for object retrieval. In: *ICCV* (2007)
- Cormen, T., Leiserson, C., Rivest, R., Stein, C.: *Introduction to Algorithms*, Second Edition. The MIT Press and McGraw-Hill (2001)

- Davis, J., Kulis, B., Jain, P., Sra, S., Dhillon, I.: Information-theoretic metric learning. In: ICML (2007)
- Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* **39**(1), 1–38 (1977)
- Deschacht, K., Moens, M.: Efficient hierarchical entity classification using conditional random fields. In: *Proceedings of Workshop on Ontology Learning and Population* (2006)
- Everingham, M., Sivic, J., Zisserman, A.: ‘Hello! My name is... Buffy’ - automatic naming of characters in TV video. In: *BMVC* (2006)
- Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. *PAMI* **28**(4), 594–611 (2006)
- Ferencz, A., Learned-Miller, E., Malik, J.: Learning to locate informative features for visual identification. *IJCV* **77**, 3–24 (2008)
- Fergus, R., Fei-Fei, L., Perona, P., Zisserman, A.: Learning object categories from Google’s image search. In: *ICCV*, vol. 10, pp. 1816–1823 (2005)
- Georghiades, A., Belhumeur, P., Kriegman, D.: From few to many: Illumination cone models for face recognition under variable lighting and pose. *PAMI* **26**(6), 643–660 (2005)
- Globerson, A., Roweis, S.: Metric learning by collapsing classes. In: *NIPS* (2006)
- Grangier, D., Monay, F., Bengio, S.: A discriminative approach for the retrieval of images from text queries. In: *Proceedings of the European Conference on Machine Learning*, pp. 162–173 (2006)
- Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: Automatic face naming with caption-based supervision. In: *CVPR* (2008)
- Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In: *ICCV* (2009a)
- Guillaumin, M., Verbeek, J., Schmid, C.: Is that you? Metric learning approaches for face identification. In: *ICCV* (2009b)
- Guillaumin, M., Verbeek, J., Schmid, C.: Multiple instance metric learning from automatically labeled bags of faces. In: *ECCV* (2010)
- Holub, A., Moreels, P., Perona, P.: Unsupervised clustering for Google searches of celebrity images. In: *IEEE Conference on Face and Gesture Recognition* (2008)
- Huang, G., Jain, V., Learned-Miller, E.: Unsupervised joint alignment of complex images. In: *ICCV* (2007a)
- Huang, G., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled Faces in the Wild: a database for studying face recognition in unconstrained environments. Tech. Rep. 07-49, University of Massachusetts, Amherst (2007b)
- Jain, V., Ferencz, A., Learned-Miller, E.: Discriminative training of hyper-feature models for object identification. In: *BMVC* (2006)
- Jain, V., Learned-Miller, E., McCallum, A.: People-LDA: Anchoring topics to people using face recognition. In: *ICCV* (2007)

- Krishnapuram, B., Carin, L., Figueiredo, M., Hartemink, A.: Sparse multinomial logistic regression: Fast algorithms and generalization bounds. *PAMI* **27**(6), 957–968 (2005)
- Laptev, I., Marszałek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: *CVPR* (2008)
- Lazebnik, S., Schmid, C., Ponce, J.: Affine-invariant local descriptors and neighborhood statistics for texture recognition. In: *ICCV*, pp. 649–655 (2003)
- Li, L., Wang, G., Fei-Fei, L.: OPTIMOL: Automatic object picture collection via incremental model learning. In: *CVPR* (2007)
- Marcel, S., Abbet, P., Guillemot, M.: Google portrait. Tech. Rep. IDIAP-COM-07-07, IDIAP (2007)
- Mensink, T., Verbeek, J.: Improving people search using query expansions: How friends help to find people. In: *ECCV* (2008)
- Neal, R., Hinton, G.: A view of the EM algorithm that justifies incremental, sparse, and other variants. In: Jordan, M. (ed.) *Learning in Graphical Models*, pp. 355–368. Kluwer (1998)
- Nowak, E., Jurie, F.: Learning visual similarity measures for comparing never seen objects. In: *CVPR* (2007)
- Ozkan, D., Duygulu, P.: A graph based approach for naming faces in news photos. In: *CVPR*, pp. 1477–1482 (2006)
- Ozkan, D., Duygulu, P.: Interesting faces: A graph-based approach for finding people in news. *Pattern Recognition* (2009)
- Pham, P., Moens, M., Tuytelaars, T.: Linking names and faces: Seeing the problem in different ways. In: *Proceedings of ECCV Workshop on Faces in Real-Life Images* (2008)
- Pinto, N., DiCarlo, J., Cox, D.: How far can you get with a modern face recognition test set using only simple features? In: *CVPR* (2009)
- Ramanan, D., Baker, S.: Local distance functions: A taxonomy, new algorithms, and an evaluation. In: *ICCV* (2009)
- Satoh, S., Nakamura, Y., Kanade, T.: Name-it: Naming and detecting faces in news videos. *IEEE MultiMedia* **6**(1), 22–35 (1999)
- Sivic, J., Everingham, M., Zisserman, A.: “Who are you?”: Learning person specific classifiers from video. In: *CVPR* (2009)
- Smeulders, A., Worring, M., Santini, S., Gupta, A., Jain, R.: Content-based image retrieval at the end of the early years. *PAMI* **22**(12), 1349–1380 (2000)
- Srihari, R.: PICTION: A system that uses captions to label human faces in newspaper photographs. In: Press, A. (ed.) *Proceedings of the AAAI-91*, pp. 80–85 (1991)
- Turk, M., Pentland, A.: Eigenfaces for recognition. *Journal of Cognitive Neuroscience* **3**(1), 71–86 (1991)
- Verbeek, J., Triggs, B.: Region classification with Markov field aspect models. In: *CVPR* (2007)
- Viola, P., Jones, M.: Robust real-time object detection. *International Journal of Computer Vision* **57**(2), 137–154 (2004)

- Wagstaff, K., Rogers, S.: Constrained k-means clustering with background knowledge. In: ICML, pp. 577–584 (2001)
- Weinberger, K., Blitzer, J., Saul, L.: Distance metric learning for large margin nearest neighbor classification. In: NIPS (2006)
- Wolf, L., Hassner, T., Taigman, Y.: Descriptor based methods in the wild. In: Workshop on Faces Real-Life Images at ECCV (2008)
- Xing, E., Ng, A., Jordan, M., Russell, S.: Distance metric learning, with application to clustering with side-information. In: NIPS (2004)



---

Centre de recherche INRIA Grenoble – Rhône-Alpes  
655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex  
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq  
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique  
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex  
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex  
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex  
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex  
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

---

Éditeur  
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)  
<http://www.inria.fr>  
ISSN 0249-0803