Boosting Classification Based Similarity Learning by using Standard Distances

Emilia LÓPEZ-IÑESTA, Miguel AREVALILLO-HERRÁEZ and Francisco GRIMALDO

Departament d'Informàtica, Universitat de València
Av. de la Universitat s/n. 46100-Burjassot (Spain)
eloi@alumni.uv.es, miguel.arevalillo@uv.es
francisco.grimaldo@uv.es

Abstract. Metric learning has been shown to outperform standard classification based similarity learning in a number of different contexts. In this paper, we show that the performance of classification similarity learning strongly depends on the sample format used to learn the model. We also propose an enriched classification based set-up that uses a set of standard distances to supplement the information provided by the feature vectors of the training samples. The method is compared to state-of-the-art metric learning methods, using a linear SVM for classification. Results obtained show comparable performances, slightly in favour of the method proposed.

Keywords. Metric learning, Classification, Support Vector Machine, Distances

1. Introduction

Comparisons are an essential task in many Pattern Recognition and Machine Learning methods. Given a collection of objects $X = \{x_1, x_2 \cdots x_n\}$, with associated representations in a multidimensional vector space, $X = \{\mathbf{x}_1, \mathbf{x}_2 \cdots \mathbf{x}_n\} \subseteq \mathbb{R}^d$, the simplest way to compare objects is by using standard similarity/distance functions on the feature-based representation e.g. Euclidean, Mahalanobis or cosine, to name a few. However, similarity measures are context dependent and they do not necessarily yield the best results.

An alternative and more sophisticated approach consists of learning the similarity function from training data, using a set of known comparison results. These training results may be pairwise (x_i and x_j are similar/dissimilar) or relative (x_i is closer to x_j than to x_k). Metric learning and classification based learning are two common approaches to tackle this problem.

In metric learning, the goal is to define a distance $d_M(x_i, x_j) = (x_i - x_j)^T M(x_i - x_j)$ $\forall x_i, x_j \in X$, where M is a positive semi-definite (PSD) matrix learned from the training data by minimizing (or maximizing) some criteria related to the performance of the function d_M. A major advantage of these methods is that the distance d_M is a pseudometric. Hence d_M can be seamlessly integrated into existing classification approaches that assume pseudometric spaces e.g. nearest neighbor.

In classification based similarity learning, the training data is used to learn a classifier. Once it has been trained, this classifier is used to yield scores that are related to the similarity between the objects. Although the resulting values do not satisfy the properties of a pseudometric, these methods are a competitive alternative approach when these properties are not needed e.g. ranking purposes.

Latest works on metric learning have reported consistently better results than classification similarity learning e.g. [9], when tested on a variety of contexts. In this paper, we present a technique that raises classification based results to comparable performances. The technique is based on a careful selection of the input format. Apart from pre-processing the feature vectors that correspond to the image pairs as in other recent works ([9],[12]), the values produced by a set of standard distances are added (in both training and prediction). Results obtained are slightly above state-of-the-art metric learning methods.

The rest of the paper is organized as follows: Section 2 describes the state of the art related to Metric Learning and Section 3 introduces the Classification-Based learning framework. The experimental setting and the analysis of obtained results are presented in Sections 4 and 5, respectively; finally, Section 6 states the conclusions and discusses future work.

2. State of the art

A number of methods in classification, computer vision and pattern recognition rely on the application of a similarity function on data samples. The relatively strong performance dependence between the methods and the similarity function has motivated an extensive research on approaches that attempt to learn the function from example data, in order to produce a customized function that is more adequate for the problem at hand.

In the context of Supervised Metric Learning problems, the example data consists of pairs of labeled instances $(\mathbf{x}_i, y_i)$ where $\mathbf{x}_i \in X$ and y_i is the label or class. These label instances are generally used to define pairs or triplet constraints in (dis)similarity terms as follows:

$$S = \{(x_i, x_j) : x_i \, and \, x_j \, should \, be \, similar\}$$
$$D = \{(x_i, x_j) : x_i \, and \, x_j \, should \, be \, dissimilar\}$$
$$R = \{(x_i, x_j, x_k) : x_i \, should \, be \, more \, similar \, to \, x_j \, than \, to \, x_k\}$$

Metric Learning uses these constraints to compute a distance $d_M(x_i, x_j) = (x_i - x_j)^T M (x_i - x_j)$ where $M \in \mathbb{S}_+^d$ and $\mathbb{S}_+^d$ is the cone of symmetric SPD d x d real-valued matrices. $M \in \mathbb{S}_+^d$ ensures that d_M satisfies the properties of a pseudo-distance and parametrizes a Mahalanobis distance family. As any PSD matrix can be decomposed as $M = W^T W$, the above distance is equivalent to computing a linear projection of the data into a new space where constraints are satisfied better, and then using the Euclidean distance to compare the samples. In the absence of this projection ($M = W = I$), d_M is the Euclidean distance.

In general, metric learning can be formulated as an optimization problem [1] that has the following general form:

$$\min_M L(M) = \ell(M, S, D, R) + \lambda R(M)$$

where $L(M)$ is a loss function. The first term (loss term) applies a penalty when constraints are not fulfilled; $R(M)$ is a regularizer term on the parameters M of the learned metric; and λ is a trade-off between the regularizer and the loss.

The different methods in the literature are characterised by using different loss functions, regularizers on M and constraints. We concentrate on some well-performing learning algorithms that motivate the approach presented in this paper. For the interested reader we refer to the complete surveys by Bellet [1] and Kulis [10].

A first seminal work in metric learning was presented in 2002 by Xing et al. [18]. In this work, they presented the formulation of Metric Learning as a convex optimization problem given a supervised data framework where the relative similarity of pairs of images defined the optimization problem constraints (also known as must-link/cannot link constraints). The goal was to maximize the sum of distances between all pairs of dissimilar instances and minimize the distances for all similar pairs by using Semidefinite Programming with a projected gradient descent algorithm. The metric learn was used to improve the performance of the k-Nearest Neighbors algorithm (k-NN).

Another popular algorithm used for k-nn classification is the Large Margin Nearest Neighbour (LMNN) approach [17]. In this case, the authors inspired their work on neighborhood component analysis [4], and introduced the concept of *target neighbors* for an instance x_i as the *k* nearest neighbors with the same label y_i that belong to a local neighborhood defined by a sphere of some radius. They also established a safety perimeter to push away instances with different labels (*impostors*). LMNN's formulation tries to increase similarity to target neighbours, while reducing it to impostors lying within the k-nearest neighbour region. To estimate the solution matrix M, they use gradient descent on the objective function. Despite that LMNN performs very well in practice, it is sometimes prone to over-fitting.

In Information Theoretic Metric Learning (ITML) [3], an information-theoretic measure is used and the LogDet divergence is introduced as a regularizer term in the optimization problem to avoid over-fitting. The authors translate the problem of learning an optimal distance metric to that of learning the optimal Gaussian with respect to an entropic objective. ITML considers simple distance constraints enforcing that similar instances have a distance lower than a given upper bound $d_M(x_i, x_j) \leq u$; and dissimilarity instances be further than a specific lower bound $d_M(x_i, x_j) \geq \nu$. The optimization method computes Bregman projections and no Semidefinite Programming is required.

Logistic Discriminant Metric Learning (LDML) [5] presents an approach for the particular context of Face Identification. The authors model the probability p_{ij} of whether two images (x_i, x_j) depict the same person as $p_{ij} = p(y_i = y_j | x_i, x_j; M, b) = \sigma(b - d_M(x_i, x_j))$ where $\sigma(z) = (1 + exp(-z))^{-1}$ is the sigmoid function that maps the distance to class probability and b is a bias term that acts as the optimal distance threshold value and is learned together with metric parameters. As $d_M(x_i, x_j)$ is linear with respect to the elements of M, it is possible to rewrite $p_{ij} = \sigma(bW^T X_{ij})$ where W is the vector containing the elements of M and X_{ij} the entries of $(x_i - x_j)^T (x_i - x_j)$ so the model p_{ij} appears as a standard linear logistic discriminant model $\sum_{i,j} t_{ij} ln(p_{ij}) + (1 - t_{ij}) ln(1 - p_{ij})$ where $t_{ij} = 1$ denotes the equality of labels y_i and y_j. The matrix M

is estimated using the maximum likelihood by projected gradient ascent in an iterative manner.

A more recent approach, namely 'Keep It Simple and Straightforward MEtric' (KISSME) has more recently been proposed in [9]. In this case, the distance metric is learned from equivalence constraints (S and D) applied to Face verification and person re-identification. This method tackles the problem from a statistical inference perspective and, in contrast to others, does not pose a complex optimization problem. Hence, it does not require computationally expensive iterations and it is orders of magnitudes faster than other comparable techniques.

In a different category of methods, we have Classification Similarity Learning. In this case, the learning of the metric matrix M is replaced by a classifier. Although the properties of a pseudo-metric do not hold in this case, the approach is still valid when the pairwise similarity measure learned does not need to be integrated in other methods whose theoretical formulation is based on the use of a pseudo-metric e.g. algorithms like k-NN. A typical scope of application is the construction of similarity-based rankings, which are necessary in a wide range of applications e.g. multimedia retrieval.

Some metric learning methods, e.g. KISSME, have reported a higher performance than classification based approaches. However, the results of the latter are highly dependent on the classification set-up, the input format used in the classification and the preprocessing of the feature vectors. In this paper we deal with these issues in a classification domain composed of images extracted from well-known repositories about face verification and object-recognition.

3. Classification-based learning

Given a number of n labeled image pairs $p_k = (x_{k_1}, x_{k_2})\ k = 1 \dots n, x_{k_1}, x_{k_2} \in X$ and their corresponding labels $l_k \in \{similar, dissimilar\}\ k = 1 \dots n$, the simplest way to train and predict with a classifier is by using the feature based representation of the images. The training information in this case is given as information-label pairs $\{\mathbf{p_k}, l_k\}$, where $\mathbf{p_k} = \mathbf{x_{k_1}}||\mathbf{x_{k_2}}$ and $||$ denotes the concatenation operator.

A significant improvement in the results was achieved by the alternative format presented in [12] and employed in [9]. In this case, the term $\mathbf{p_k}$ in the information-label pairs $\{\mathbf{p_k}, l_k\}$ is defined as $\mathbf{p_k} = \mathbf{abs}(\mathbf{x_{k_1}} - \mathbf{x_{k_2}})||\mathbf{x_{k_1}} * \mathbf{x_{k_2}}$ with abs, $-$ and $*$ denoting element-wise absolute value, subtraction and multiplication operations on the feature vectors, respectively. This format is used in a face verification context, and it is based on the argument that the differences between the features will be small if elements are similar and that the sign of the multiplication is important to separate samples around 0. A similar format was also used in an image retrieval context [15], showing a higher performance than using the original feature vectors in the low sample case.

Recently, a convenient combination of standard distances has been successfully applied in a classification set-up, to boost performance when several modalities are available [13]. In particular, a pool composed of four Minkowski distances (Lp norms for $p \in \{0.5, 1, 1.5, 2\}$) was used to replace the original feature vector in p_k by a new set of features composed of distance values defined on the multiple descriptor spaces. In total, $4m$ distances were used, with m the number of descriptors. Despite of the loss of some of the information contained in the original features, the intrinsic dimensionality reduc-

tion associated with the method has a compensatory effect and leads to higher precision rates.

In this paper, we also use a set of standard distances as an input to the classifier, but these are used to supplement (rather than replace) the information provided by the original feature vectors. This leads to an improved Enriched Classification Similarity Learning (ECSL) method. In this case the format used as input are information-label pairs $\{\mathbf{p_k}, l_k\}$, defined according to Eq.1

$$\mathbf{p_k} = \mathbf{abs}(\mathbf{x_{k_1}} - \mathbf{x_{k_2}})||\mathbf{x_{k_1}} * \mathbf{x_{k_2}}||d_1(\mathbf{x_{k_1}}, \mathbf{x_{k_2}})|| \ldots ||d_s(\mathbf{x_{k_1}}, \mathbf{x_{k_2}}) \quad (1)$$

where $d_1 \ldots d_s$ represent a set of standard distance functions defined on the original feature space. The rationale behind this enlarged data input format is that the distinct nature of each distance may contribute to the learning by catching a different relation between the features in a pair. For example, while the cosine distance considers the features as vectors and focuses on the angle between them, the Euclidean distance considers them as points and measures the straight line distance between the points. This may help the classifier learn a more informed similarity score function, thus leading to better classification results.

4. Experimental setting

An extensive experimentation has been carried out, to compare the performance of our method to that of other state-of-the-art metric learning approaches. In particular, we have evaluated our method against the SVM method with the input format used in [11] and [12], Information-Theoretic Metric Learning (ITML) [3], Large-Margin Nearest Neighbors (LMNN)[17], Linear Discriminant Metric Learning (LDML) [5] and KISS Metric Learning (KISSME) [9]. The performance of all methods has been tested in four different well-known and challenging datasets for face verification and object recognition that have in common that contain images with important variations in illumination, poses or scale. A summary with the main characteristics is in Table 1.

- The ***Labeled Faces in the Wild (LFW)*** [6][7] consists of 13233 face images of 5749 people taken from the Yahoo! News Web. We use the image restricted test protocol on LFW where the only available information is whether each pair of training images are from the same subject or not. Out of all the possible feature vectors available for the LFW data set, we use the Scale-Invariant Feature Transform (SIFT) based feature vectors [5] of LFW and the high-level face representation obtained in [11]. These are referred to as LFW-SIFT and LFW-Attr, respectively.
- The ***PubFig database*** [11] is a large, real-world face dataset consisting of 58797 images of 200 people collected from Google and Flickr. Its image attributes provide high-level semantic features indicating the presence or absence of visual face traits (such as hair, glass, age, race, smiling and so on) and allow a semantic description that is more robust against large image variations and that can lead to good verification performance.

- The ***ToyCars*** [16] data set consists of 256 image crops of 14 different toy cars and trucks and is considered for compare our approach in a different setting of face verification using the image representation described in [9]. The intention of the database is to compare before unseen object instances of the known class cars. Thus, in testing, the task is to classify if a pair of images shows the same object or not.

To reduce the dimensionality a Principal Component Analysis was conducted in each dataset in a pre-processing stage. The LFW-SFIT dataset is projected onto a 100 dimensional subspace, LFW-Attr and Pubfig are reduced to a 65 dimensional subspace and, for the ToyCars database we use 50 dimensional subspace. For further information on these data sets, the user is referred to [5][9].

For the sake of comparison, we inherit the experimental framework presented in [9], that allows for the evaluation of the methods according to their performance at ranking a number of pairs according to their estimated similarity. To this end, each repository is divided into f folds of disjoint objects and a cross validation approach is used. For each experiment, one fold is chosen for test and the remainder ones are used for training. Training and test sets are generated at random, according to the class information available. Results on each fold are appropriately combined to produce a ROC curve for each method.

To allow for a fair comparison, all methods are fed with the same training data, and the resulting model is used to rank a new common set of pairs by similarity. The number of folds used and the number of pairs in the training and test sets are the same as in [9] in all cases and are summarized in Table 1.

The supplementary distances used in our method are the Mahalanobis distances, the Cosine distance and four Minkowski distances (L_p norms) with values $p = 0.5, 1, 1.5, 2$ as the standard distances added in Eq.1. We have used a linear SVM as the classifier, setting the cost parameter to the default value of 1, as in [9].

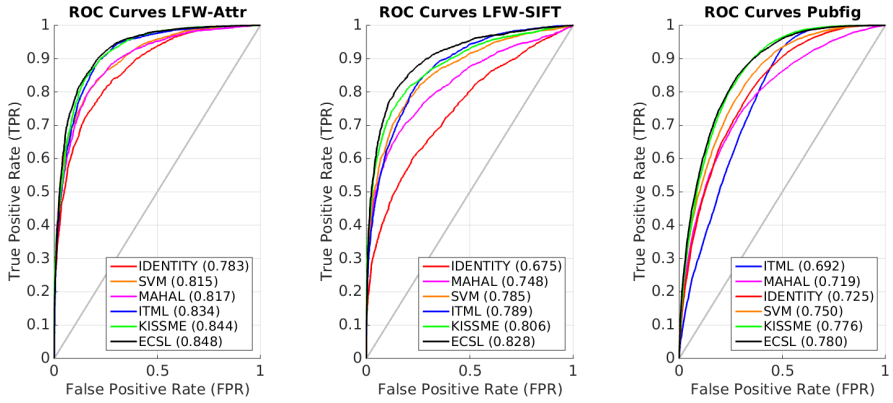
Table 1. A summary of the four data sets used in the experiments.

Data set	Size	Objects	Folds	Num. Training	Num. Testing
ToyCars	256	14	2	8515	7381
LFW-SIFT	13233	5479	10	5400	600
LFW-Attr	13233	5479	10	5351	596
PubFig	58797	200	10	18000	2000

5. Results

Figure 1 shows comparative ROC curves in the databases LFW-Attr, LFW-SIFT and PubFig. For the sake of clarity, plots presented in this section only include two metric learning methods, namely KISSME and ITML. The first has been chosen because it yields consistently the best results across all metric learning methods in all repositories. The second because it is a method frequently used in the literature for comparison purposes in metric learning contexts [2][8]. We have also included the results of the Mahalanobis (MAHAL) and Euclidean distances (IDENTITY) as baselines. The Equal Error Rate is

provided in brackets as part of the legend, and also shown in Table 2 for all methods, including those not plotted in Figure 1.



(a) (b) (c)

Figure 1. Comparative performance between for ECSL, KISSME, ITML, MAHAL, SVM, IDENTITY in a) LFW-Attr; b) LFW-SIFT; and c) PubFig.

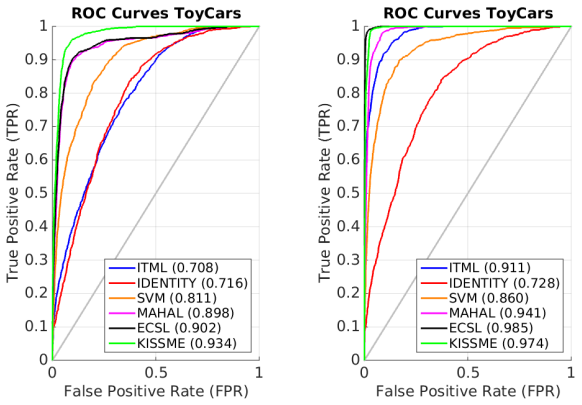
We can observe that the classification performance of the proposed method equals the best of the metric learning techniques in the PubFig and LFW (Attributes) repositories. In LFW-SIFT, the method outperforms all of the metric learning algorithms included in the comparison. In addition, a substantial performance increase can be observed with respect to the use of a SVM without the added distances in LFW-SIFT, LFW-Attr and PubFig. The standard SVM method consistently obtains worst results than any of the two metric learning approaches shown in the plots. The only exception is in PubFig, where the standard SVM method appears below the KISS method but performs better than ITML. In Table 2, it can also be observed that LDML and LMNN methods are ranked different depending on the database but they are always outperformed by ECSL and KISSME.

Table 2. Equal Error Rate for ECSL and the rest of the methods in all databases. Best results in each dataset are marked in bold.

Method	LFW-Attr	LFW-SIFT	PubFig	ToyCars	ToyCars*
ECSL	**0.848**	**0.828**	**0.780**	0.902	**0.985**
KISSME	0.844	0.806	0.776	**0.934**	0.974
SVM	0.815	0.785	0.750	0.811	0.860
MAHAL	0.817	0.748	0.719	0.898	0.941
ITML	0.834	0.789	0.692	0.708	0.911
LDML	0.834	0.796	0.776	0.716	0.720
LMNN	0.831	0.785	0.735	0.805	0.919
IDENTITY	0.783	0.675	0.725	0.716	0.728

An exception to ECSL showing the best performance has been observed in the ToyCars repository (see Fig. 2(a)). In this case, KISSME outperforms ECSL, that ranks sec-

ond and above the rest of the learning methods. The low performance of ITML is also noticeable in this case, scoring below the Euclidean distance. Again the improvement of ECSL with respect to the standard SVM is remarkable in this database. To further study the performance of the methods in this database, we have run a second experiment (see ToyCars* in Table 2). Instead of selecting pairs from disjoint sets of objects for training and test, pairs have been randomly chosen (taking care that no pair is simultaneously used for training and test). These two experiments represent different scenarios. In the first case, the similarity function is learned from a set of objects, and applied on a different set of never seen objects. This set-up is useful when we have a collection of classified objects that can be used to generate the similar and dissimilar pairs. In the second case, the function is learned from a selection of pairs extracted from the same repository, as typically happens in retrieval problems where it is the user who judges the similarity. The results for this second experiment are shown in Fig. 2(b). In this case, ECSL scores the best, again slightly better than the best of the metric learning methods.



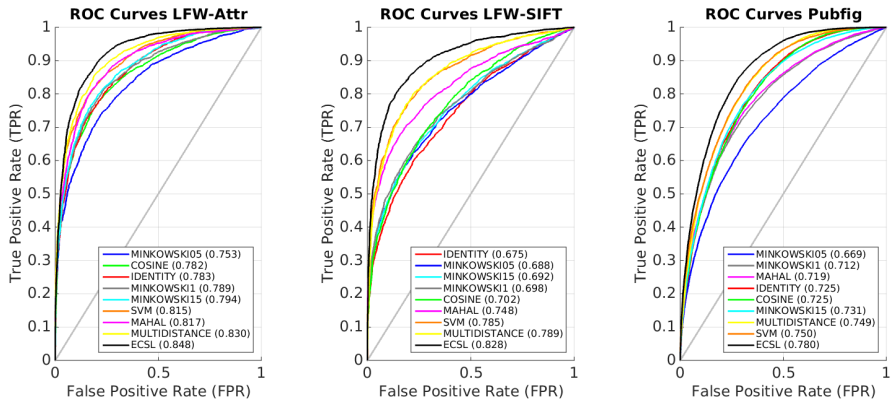
(a) (b)

Figure 2. Comparative performance between the method proposed (ECSL) and the comparative methods in the ToyCars database a) using pairs from disjoint sets in training and test; b) using disjoint sets of random pairs for training and test (ToyCars*)

To test whether the improvement achieved is due to a particularly good behavior of one of the standard distances and whether the concatenation of a set of well-known distance values as part of the training data format makes any benefit, in Fig. 3 we compare the performance of the proposed method to that obtained by using each of the added distances in isolation. The result of using a SVM as a distance combination method [14] has also been included as a reference (SVM in the figure). In all cases, we can observe that the performance of the SVM algorithm stays close to that of the best of the single distances but, when all the standard distances are used within the classification set-up of ECSL, the performance is significantly boosted.

6. Conclusion

Recent methods on the metric learning literature have outperformed classification similarity learning. The algorithm presented in this paper acts on the data input to the clas-



(a) (b) (c)

Figure 3. Comparative performance between the method proposed (ECSL) and standard distances in a) LFW-Attr; b) LFW-SIFT; and c) PubFig.

sifier to turn classification similarity learning into a competitive method, with a performance slightly above recent metric learning methods. In particular, for each pair of objects p_k, we combine the information contained in their feature vectors as shown in Eq.1, considering the result of a set of distance functions on the original feature space.

Although the SVM has been used to develop a proof of concept, the method is by no means restricted to the use of this particular classifier. On the contrary, the framework presented is open to the use of alternative classification methods and/or meta-estimators. It is also possible to extend the similarity paradigm to support degrees of similarity by using regression. When using a SVM, the extension of ECSL to the non-linear case is particularly straight forward. In addition, an adequate choice of the kernel according to the specific characteristics of the problem may help achieving further improvements to the method.

Another important aspect not considered in this research is the robustness of the methods to variations of the training size. Responses to small sample size situation are specially relevant when the number of examples is scarce (e.g. Content Based Image Retrieval). In this context, previous work e.g.[15] has already outlined the potential of integrating standard distance values into a classification approach for similarity learning.

Acknowledgements

We would like to thank Martin Köstinger and Paul Wohlhart for their kind support. This work has been partly supported by the Spanish Ministry of Science and Innovation through projects TIN2011-29221-C03-02 and TIN2014-59641-C2-01.

References

[1] A. Bellet, A. Habrard, and M. Sebban. *Metric Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2015.

[2] Q. Cao, Y. Ying, and P. Li. Similarity metric learning for face recognition. In *The 16th IEEE International Conference on Computer Vision (ICCV)*, pages 2408–2415, December 2013.

[3] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In Z. Ghahramani, editor, *Proceedings of the 24th International Conference Machine Learning (ICML)*, volume 227, pages 209–216. ACM, 2007.

[4] J. Goldberger, S. T. Roweis, G. E. Hinton, and R. Salakhutdinov. Neighbourhood components analysis. In *Advances in Neural Information Processing Systems 17 (NIPS)*. MIT Press, 2004.

[5] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *The 12th IEEE International Conference on Computer Vision (ICCV)*, October 2009.

[6] G. B. Huang and E. Learned-Miller. Labeled faces in the wild: Updates and new reporting procedures. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[7] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[8] P. Jain, B. Kulis, I. S. Dhillon, and K. Grauman. Online metric learning and fast similarity search. In *Advances in Neural Information Processing Systems 21 (NIPS)*, pages 761–768. MIT Press, 2008.

[9] M. Köstinger, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof. Large scale metric learning from equivalence constraints. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2288–2295, June 2012.

[10] B. Kulis. Metric learning: A survey. *Foundations and Trends in Machine Learning*, 5(4):287–364, 2013.

[11] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *The 12th IEEE International Conference on Computer Vision (ICCV)*, October 2009.

[12] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Describable visual attributes for face verification and image search. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, volume 33, pages 1962–1977, October 2011.

[13] E. López-Iñesta, M. Arevalillo-Herráez, and F. Grimaldo. Classification-based multimodality fusion approach for similarity ranking. In *17th International Conference on Information Fusion (FUSION)*, pages 1–6, July 2014.

[14] E. López-Iñesta, F. Grimaldo, and M. Arevalillo-Herráez. Comparing feature-based and distance-based representations for classification similarity learning. In *Artificial Intelligence Research and Development - Recent Advances and Applications, (CCIA)*, pages 23–32, October 2014.

[15] E. López-Iñesta, F. Grimaldo, and M. Arevalillo-Herráez. Classification similarity learning using feature-based and distance-based representations: A comparative study. *Applied Artificial Intelligence*, 29(5):445–458, 2015.

[16] E. Nowak and F. Jurie. Learning visual similarity measures for comparing never seen objects. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, June 2007.

[17] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research*, 10:207–244, 2009.

[18] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. J. Russell. Distance metric learning with application to clustering with side-information. In *Advances in Neural Information Processing Systems 15 (NIPS)*, pages 505–512. MIT Press, 2002.