Face Recognition with DAISY Descriptors

Carmelo Velardo
Eurecom
2229 Route des Cretes
06560, Sophia Antipolis, France
velardo@eurecom.fr

Jean–Luc Dugelay
Eurecom
2229 Route des Cretes
06560, Sophia Antipolis, France
dugelay@eurecom.fr

ABSTRACT

In this paper we propose a new face recognition approach based on DAISY, a dense computed SIFT–like descriptor. Our algorithm is designed to be fast for dense computation, and useful for re–identification as it is able to distinguish pairs of images as belonging to the same subject or not. The descriptors are computed densely and matched with a new strategy that represents an efficient trade off between accuracy and computational load; afterwards a Support Vector Machine is used to classify the output of the matching to recognize if the pair of images belongs to the same person. An analysis of performance will be conducted on two different databases in order to compare our results with the already existing ones. We show that better performance than SIFT techniques can be achieved using our algorithm.

Categories and Subject Descriptors

I.5 [**Computing Methodologies**]: Pattern Recognition

General Terms

Security

Keywords

Face recognition, DAISY, SIFT, person re–identification

1. INTRODUCTION

As the use of video surveillance systems becomes massive, accurate methods for person re–identification are needed. The intention of such a technique is to recognize all instances of the same person at any given location and at any given time instant. In a typical surveillance scenario a subject crosses the field of view of one camera and the operator wants to track him/her. Common approaches rely on samples of the appearance of the tracked person, those features are taken at distance as the typical scenario is thought for cameras placed in big public areas (squares, airport halls, ...) far away from the target, and hence often unprecise.

MM&Sec'10, September 9–10, 2010, Roma, Italy.

However a new scenario is nowadays appearing. Since video surveillance cameras show already cutting edge performances of more than one mega pixel, a sample of subject face could show good characteristics, enough to be considered from a re–identification system. That is to say computing a similarity score between two representations of the same (or different) face. Such similarity score can be exploited to match all the previous or future appearances of the same subject.

Many interesting algorithms exist that address the problem of face recognition. Among all, recently, several face recognition techniques were presented which make intensive use of Scale Invariant Feature Transform (SIFT) [7] descriptors. This technique was originally conceived for object recognition as its major characteristic is the extraction of view–invariant representation of 2D patterns.

Bicego et al. pioneered the SIFT use in this field; in [1] they apply SIFT keypoint extraction and comparison using different methods. In their first approach they extract the pair of descriptors that provides the minimum distance between the query and the template image. This value of distance is then used to compute the similarity between faces. In a second approach, they use a regular grid to extract and to compare the descriptors, the distances are then averaged in order to compute a similarity score for the given images.

A similar technique is shown in [8] where faces are segmented in 5 different areas (the two eyes, the nose, and the two sides of the mouth). The features belonging to each area on the query are compared with the ones present in the corresponding area on the template image. Additionally a full match strategy is performed that does not consider the 5 regions constraint. Both the scores for local and global analysis are then considered for the identification.

Other approaches [4, 11] exploit the SIFT features to build a connected graph that links all the features extracted from a face. A graph–matching algorithm is then introduced to compute the matching score. This approach arises the problem of multiple feature matches: due to the complexity of faces, many local features may present similar descriptors. Systems that do not take into account spatial constraints may then present multiple matches for a single point.

In [3] the authors propose two modifications of SIFT. The former involves the keypoints selection scheme: the distance metric is modified in order to obtain more robust locations for the computation of the descriptor; the latter introduces the concept of partial descriptor in order to deal with the boundaries of the image.

A different methodology exploits SIFT directly as input

features of the recognition system. In [5] a bag of words approach is exploited that creates clusters of SIFT words. A face is then represented as a collection of those words, and the recognition is performed using a Support Vector Machine (SVM) classifier. A Multi Layer Perceptron is trained in [6] which uses the bootstrap technique. Each of the SIFT features extracted from a face are given as input to the multilayer perceptron that provides as output a probability histogram representing the score for each subject previously enrolled.

The aforementioned systems rely on a number of keypoints that may appear insufficient as a good feature present in one pose can disappear or be considered unstable in another one. This exposes those techniques to the risk of decreasing performance in the case of pose variations or in case other factors cause the occlusion of some of the points needed for the recognition.

We believe that a dense computation of descriptors could address this issue since if some points will lack, others will take their places. For this reason our new face recognition system is conceived to exploit DAISY [14], a recent descriptor particularly efficient for dense computation. Nonetheless a high number of points can represent a challenge as we need to compare all of them. A new matching procedure is then presented that allows an efficient feature–to–feature matching computation.

Given a pair of face images our system classifies them as a Client match if the two images belong to the same person, otherwise they are marked as an Impostor pair.

In [13] the mechanism of face recognition is described as twofold: holistic or feature based. Nevertheless the study shows that even when feature based, the recognition mechanism has to be considered holistically, that is to say all the features have to be considered together in the computation. The algorithm we present follows this statement as all the features are locally extracted and then evaluated as a whole via a Support Vector Machine classifier. Our idea is based on the use of DAISY descriptor and a local analysis of the points based on a recursive grid approach. The former technique manages the dense computation of the descriptors (potentially for each pixel of the image), the latter allows to prune the search for the best match, without incurring in the curse of dimensionality.

As from [12] the face of an individual can be affected from a series of slight variations, those variations may affect the performance of face recognitions systems, this can be particularly present in the video surveillance scenarios. We developed our system in order to be robust to those variations, the descriptor should provide the robustness to light variation, and the search algorithm robustness to changes in the pose and expression of the subject.

The remaining part of the paper will be firstly devoted to DAISY, the descriptor used for features extraction, then in Section 3 the face matching approach will be explained. Sections devoted to experimental results and the conclusions will then follow.

2. FAST DESCRIPTOR FOR DENSE COMPUTATION

Scale invariant features (SIFT) were introduced in [7] as keypoints descriptors for detecting and extracting distinctive local features from images, and lately many applications exploited them for object recognition and tracking.

The main drawback of such technique lays on the amount of resources required and for this reason, other descriptors as GLOH from Mikolajczyk and Schmid [10] and DAISY of Tola et. al [14], were designed to have better performance.

Although DAISY has been used in case of strong appearance changes (e.g. wide baseline stereo matching problem), such descriptor has never been applied in other problems dealing with high non linear transformation of the target like in face recognition. Particularly, the design of DAISY has shown promising performance independently in [14] and [15]. The authors of both works point out the outperforming results of several possible combinations of the same descriptor against SIFT.

Because of this reason, and since the design of our face identification algorithm requires a dense representation of face feature descriptors, we chose to use DAISY. Besides giving results that are comparable to the one of SIFT and GLOH [14, 15], DAISY is perfectly appropriate for dense computation. As from the work of Tola et al. [14] the new descriptor performs 66 times faster than SIFT in the dense scenario, that makes it suitable for our work.

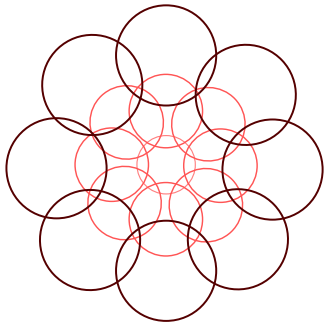


Figure 1: DAISY descriptor shape. The size of the circle represents the size of the Gaussian kernel considered when computing the histogram.

DAISY shape is depicted in Fig.1. It is based on convolution of gradients in specific directions. An image is converted into a series of orientation maps (one for each quantized direction). Given an image I and a direction o such orientation maps are defined as $M_o = (\frac{\partial I}{\partial o})^+$ and represent the positive values of the image gradient norm for each pixel position.

Subsequently the orientation maps are multiplied by Gaussian kernels of increasing standard deviation values $G_o^\Sigma = \mathcal{N}_\Sigma * M_o$. Each of these represents a different level of image content and is kept for further computations.

For all the pixels, the neighborhood of radius δ is divided in a series of intersecting circles displaced as in Fig.1. The radii of the circles are proportional to their distance from the center of the descriptor (i.e. the pixel for which we are computing the descriptor). Each circle represents the location where a histogram is computed from all the values of the G_o^Σ belonging to that particular orientation. Histograms are computed similarly to what happens with SIFT and GLOH.

Once those computations are over, the full descriptor is built as concatenation of all the other small histograms. Then, for each pixel position, we obtain a sequence of numbers representing the normalized histograms coming from the orientation maps on the neighborhood of the given pixel.

As distance metric for computing the dissimilarity of two features, we followed the suggestion of DAISY authors of using a straightforward Euclidean distance. The choice is also due to the preservation of speed that such metric allows.

3. FACE MATCHING

This section introduces our matching strategy that exploits the dense computation of DAISY descriptors and a recursive approach for matching the two images. Our intention is to produce a similarity mask for the data and use such mask to classify the pairs of images exploiting an SVM classifier.

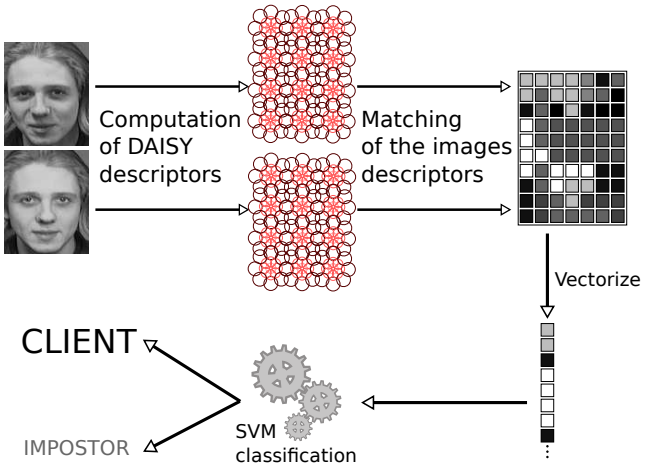


Figure 2: Scheme that provide an overview of the proposed approach.

A high level description of our approach is proposed in Fig.2. The algorithm takes as input two images to be evaluated. Those images can come from two different cameras of a surveillance scenario, or being a query image to be checked and a template image coming from a database of enrolled persons.

The system firstly initializes DAISY computation in order to extract the descriptors, then the recursive grid search extracts the local distances between query and template for each grid location. Afterwards the series of local distances is vectorized and passed through an SVM classifier that distinguishes between a client or an impostor match in case the two images represent the same subject or two different ones.

This set up is useful for both the re–identification scenario and the face recognition one. In the former the images come from two different cameras, while in the latter the first is the enrollment picture, the other is the query image.

3.1 Recursive grid

In [1] a grid of intersecting squared areas is used to compute SIFT features. Those sub images are of 1/4 and 1/2 of image width and height respectively. The keypoints extracted are matched using corresponding areas in query and template images. Our matching strategy also considers a grid in the query and template image, but both the granularity and the computation of the matching differ a lot. In [1] the keypoints selection scheme does not allow control on where the descriptor is computed (i.e. the authors have to accept as good the selection performed by SIFT). In our case, instead, the dense computation of DAISY potentially allows the matching at each pixel location, hence we are not bound to any keypoint selection scheme. Moreover, its computational speed ensures the division of the image in smaller areas and then allows to consider a higher number of sub images. Nevertheless having a large number of descriptors can also represent a drawback. The time required for the computation increases proportionally to the number of keypoints to be matched (in the worst case quadratically to the number of features).

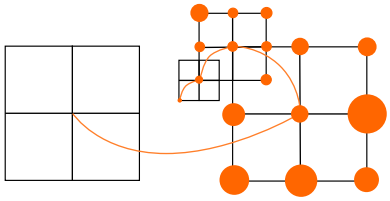


Figure 3: A graphical example of the concept of recursive search in a grid. The size of the circle represents the magnitude of the distance value. We can notice the strong similarity with three step search performed in block matching for motion estimation.

For this reason we introduce here a recursive grid search that is able to extract the local distances for each point of the grid. The recursive approach is used to prune the search tree considering only the portion of the area that seems to have small distance w.r.t. the original keypoint. A graphical example of the algorithm is shown in Fig.3.

A fixed grid is superimposed to the query image, for each grid point in the query we compute the corresponding descriptor as explained in the previous section. We compute then its distance with the corresponding point on the template grid in a recursive way.

Algorithm 1 Procedure that computes the array of local scores

Require: Query and Template images.
  for all i in grid(Query) **do**
    Take j, corresponding point into the grid(Template)
    Compute distance between the two descriptors
    ld $\leftarrow$ Euclidean(DAISY(i), DAISY(j))
    Extract min value from the neighborhood
    nd $\leftarrow$ NeighDistance(i,j,δ)
    Min between local and neighbors distances
    outResult(i) $\leftarrow$ min(ld, gd)
  end for
  Return the array of local minima
  return outResult

The distance is computed taking into account also the neighborhood (at distance δ) of the current point. Afterward, the point showing the minimum distance is considered for further analysis, the algorithm takes it as center, halves the distance, and the computation continue until the smallest radius is reached.

A facial expression can be seen as a displacement of the feature points over the face surface; the design of the search algorithm guarantees robustness to this kind of variations. In fact each feature descriptor can be searched in a neigh-

Algorithm 2 Function that recursively computes the local matching score

Require: Points i, j and radius δ
  function NeighDistance(i, j, δ)
  if δ is 1 **then**
    Return the biggest number available in the system
    return BIG
  end if
  For all the j neighbors at distance δ
  for all k in Neighborhood(j, δ) **do**
    Compute the distance between k and i
    d(k) $\leftarrow$ Euclidean(DAISY(k), DAISY(i))
  end for
  Select the point with minimum distance from i
  kmin $\leftarrow$ argmin$_k$(d)
  Return the real minima
  return min(d(kmin), NeighDistance(kmin, i, $\delta/2$))

borhood that varies according to the magnitude of the local search distance δ. Moreover, even if the size of the descriptor is smaller than the local search distance the multi–step algorithm allows a small overlap ensuring stability against features displacement.

Using the grid search allows us to exploit the dense computation of the descriptors without incurring into the curse of dimensionality. That is to say we compute the descriptors at each pixel position, but we perform the comparison for the portions of the Template image that show similarities with the Query ones.

An additional consideration has to be done on the computational load of our algorithm. A system exploiting a full dense computation would need to consider a number of matches far away bigger than our approach. For the numerical comparison we will consider a coarse grid dimensions of 11×9 points and a search depth of 3 steps. In case of full dense matching approach the number of matches to be evaluated is equal to $9 \times 8 \times 8 \times 99 = 57024$. Using the recursive grid search, instead, allows to focus the search on the portion of the image patch that looks more similar to the original patch. From a numerical point of view: $(9 + 8 + 8) \times 99 = 2475$, that means exploiting our matching design allows 23 times faster performances w.r.t. the complete dense approach.

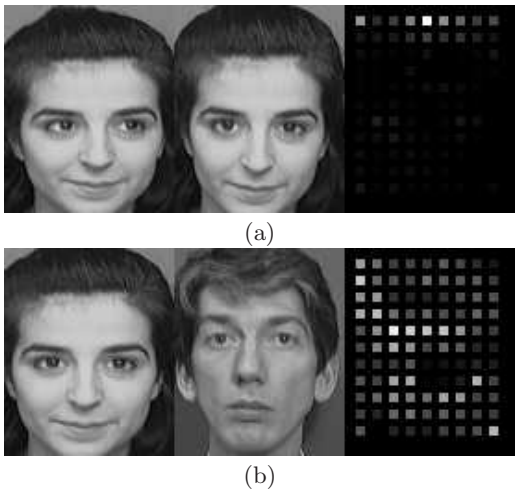
3.2 Data representation and classification

Data are processed in pairs using our recursive grid search algorithm. It computes the local minimum for a given grid location using a recursive approach and exploiting the dense computation of the DAISY descriptor. The computation is dense as the descriptor can be potentially needed at each pixel location.

An example of the recursive grid search output is shown in Fig.4, two pairs are shown belonging to a client and an impostor match. As from the scheme in Fig.2, at the end of the features extraction block, data are provided to a Support Vector Machine that analyzes and classifies the image pair as belonging to the Client or Impostor class.

4. EXPERIMENTAL RESULTS

This section is dedicated to the experimental results that validate our face identification system. A comparison with other similar techniques will be presented showing the increase of performance of our system w.r.t. the others. In order to compare our results with the techniques that shares similarities with our study we consider the well–known Olivetti Research Lab (ORL) and the FERET databases.



(a)

(b)

Figure 4: A graphical example of the recursive grid search output. In the client (a) and impostor (b) pairs, each small square represents the local distance value associated to the grid point.

After the local minima are computed for the given pair of images, all the values are vectorized and normalized in the range $[-1, +1]$, such normalization is needed for the classification of the data. In our case the grid size is fixed to 11×9 points, this means 99 values feature vectors that are elaborated from our classifier. For the SVM classification the tool provided by [2] was used. The classification was performed using a k–fold methodology (k equal to 5) in order to assure that all the available data could be used for training and testing, and to avoid overtraining. The outcome of the cross validation is obtained averaging the results of each fold.

For the sake of clarity the presentation of the results will be divided per database.

4.1 Olivetti Research Lab database

The ORL database is a general collection of faces composed by 40 subjects and 10 pictures per subject. No constraints where imposed at collection time, so that it shows different facial expressions and pose. The images size is 112 pixel for the height and 92 for the width.

Regarding the ORL database the methodology used is the following: for each person all the possible pairs of images are considered so that a total of 1800 pairs of Clients are generated. The same number of Impostors is generated using random couples of subjects. A total of 3600 different output vectors are then analyzed using the SVM classifier.

Regarding the results over the ORL dataset, Table 1 shows the recognition rates. We can clearly see that using our system increase the recognition performance w.r.t. the others. The similar results of the techniques based on SIFT can be explained as a limitation of the technique itself and especially on the number of features they use. The number of descriptors used for the recognition process by the three methods is comparable, that validates our idea about the

Table 1: Performance over ORL database. The first two results are taken from [9] while the third one from [3].

Methods	Recognition rate
SIFT GRID	95.2
Fisher Ratio SIFT	95.5
PDSIFT	95.5
Proposed approach	**98.2**

need of a descriptor suitable for dense computation in face recognition.

4.2 FERET database

The FERET database is well known as it became the standard de facto for face recognition techniques. Although it is composed of several galleries, in this work we consider the first two (*fafb* and *fafc*) as they are useful for comparison purposes. The first gallery is a collection of 1195 images with expression variations and the second contains 194 images with illumination variation, both the probes are compared with an original gallery of 1196 pictures from 1196 different persons.

The same strategy was adopted for the two galleries of the FERET database. In the first case a number of 1195 clients and impostors was created and for the second gallery 194 pairs where matched. Each image was downscaled to 92×112 from its original size for the sake of comparison. FERET images are normally bigger and the rescaling may affect our recognition system, in any case still the results are better than the previous approaches.

Table 2: Performance over FERET database. The results where taken from [8].

Methods	FERET fafb	FERET fafc
SIFT GRID	94	35
Local/Global SIFT	97	47
Local Binary Pattern	97	79
Proposed approach	**97**	**70**

Table 2 shows the performance of our algorithm using the FERET database. Here our algorithm is comparable with the one of [8] for the analysis on the first probe gallery (fafb), however is in the second one that our algorithm shows better performance than the others based on SIFT and it approaches the good results of the Local Binary Pattern technique.

Our system performs a search in the full image spaces; this explain the clear gap between our result and the ones of SIFT techniques. Indeed the recursive grid search algorithm allows the comparison of all the patches that compose the two images (since we compute the descriptor densely), at the contrary the tecniques exploiting SIFT analyze only some of the descriptors present in the image (i.e. the ones of the most stable keypoints). Additionally the illumination variations (present in the *fafc* gallery) affect the appearance of the face, and by reflex also the keypoint selection scheme of SIFT.

4.3 Intra–database cross validation

For further validation of our algorithm an intra–database cross validation was performed. In other words our system was trained using alternatively one of the databases and tested on the second one. Such methodology, not always exploited, may represent a solution to assure that the results are not biased by the use of a single database. It guarantees indeed that the training and testing steps are done on completely different data populations. Additionally it approximates better real case scenarios where the enrollment is not done in the same conditions (usually different people and different cameras, i.e. the typical video surveillance scenario). The results of such analysis are reported in Table 3. The stability of our system is then validated as the results are coherent between the two tests.

Table 3: Intra–database cross validation. For the FERET database the *fafb* gallery was used.

Methods		Rate
Train	Test	
ORL	FERET	96.1
FERET	ORL	96.5

This result shows how both the models of the Client and the Impostor are learned correctly from the SVM. It shows that a system could be set up where the enrollment is given a priori and only the test has to be performed, increasing then the speed of the system itself.

5. CONCLUSIONS

A new face recognition approach was presented that classifies a pair of images as client or impostor pair. The system exploits DAISY a novel keypoint descriptor originally conceived for stereo matching via dense computation. DAISY is used in this paper for the dense extraction of the features exploited for face matching as its performance overwhelms for speed the well–known SIFT.

The experiments were conducted through the ORL and FERET database; they show better results compared to similar approaches using SIFT, validating the exploitation of the new descriptor and the design of the matching. Future works may concern the use of the matching techniques introduced by [14] together with DAISY in order to make it robust in occlusion handling that can occur in strong pose variation.

Acknowledgments

The authors would like to thank Angela D'Angelo and Judith Redi for their precious comments. Additionally we thank the anonymous reviewers for their very insightful comments. Finally we acknowledge the projects that partly supported this work: the French national projects ANR VideoID and OSEO Biorafale.

6. REFERENCES

[1] M. Bicego, A. Lagorio, E. Grosso, and M. Tistarelli. On the use of SIFT features for face authentication. In *Computer Vision and Pattern Recognition Workshop, 2006 Conference on*, pages 35–35, 2006.

[2] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at *http://www.csie.ntu.edu.tw/ cjlin/libsvm*.

[3] C. Geng and X. Jiang. Face recognition using sift features. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 3313 –3316, 7-10 2009.

[4] D. Kisku, A. Rattani, E. Grosso, and M. Tistarelli. Face identification by SIFT-based complete graph topology. In *2007 IEEE Workshop on Automatic Identification Advanced Technologies*, pages 63–68, 2007.

[5] D. Liu, D. mei Sun, and Z. ding Qiu. Bag-of-words vector quantization based face identification. In *Electronic Commerce and Security, 2009. ISECS '09. Second International Symposium on*, volume 2, pages 29 –33, 22-24 2009.

[6] T. Liu, S.-H. Kim, H.-S. Lee, and H.-H. Kim. Face recognition based on a new design of classifier with sift keypoints. In *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, volume 4, pages 366 –370, 20-22 2009.

[7] D. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[8] J. Luo, Y. Ma, E. Takikawa, S. Lao, M. Kawade, and B. Lu. Person-specific SIFT features for face recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2007. ICASSP 2007*, volume 2, 2007.

[9] A. Majumdar and R. Ward. Discriminative SIFT Features for Face Recognition. In *Proc. of Canadian Conference on Electrical and Computer Engineering, 2009*, pages 27–30, 2009.

[10] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[11] D. Ozkan and P. Duygulu. A graph based approach for naming faces in news photos. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, 2006.

[12] F. Perronnin, J.-L. Dugelay, and K. Rose. Deformable face mapping for person identification. In *International Conference on Image Processing*, volume 1, pages 14–17, 2003.

[13] P. Sinha, B. Balas, Y. Ostrovsky, and R. Russell. Face recognition by humans. *Face Processing: Advanced Modeling and Methods. Academic Press, San Diego*, 2006.

[14] E. Tola, V. Lepetit, and P. Fua. Daisy: an Efficient Dense Descriptor Applied to Wide Baseline Stereo. 2010.

[15] S. Winder and M. Brown. Learning local image descriptors. In *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR'07*, pages 1–8, 2007.