Linear Dimensionality Reduction Applied to SIFT and SURF Feature Descriptors

Ricardo Eugenio González Valenzuela,[a] William Robson Schwartz,[b] Helio Pedrini[a]
[a]Institute of Computing, University of Campinas, Av. Albert Einstein 1251, Campinas, SP, Brazil, 13083-852
[b]Department of Computer Science, Universidade Federal de Minas Gerais, Av. Antônio Carlos 6627, Belo Horizonte, MG, Brazil, 31270-010

Abstract. Robust local descriptors usually consist of high dimensional feature vectors to describe distinctive characteristics of images. The high dimensionality of a feature vector incurs into considerable costs in terms of computational time and storage. It also results in the curse of dimensionality, which affects the performance of several tasks that use such feature vectors, such as matching, retrieval and classification of images. To address these problems, it is possible to employ some dimensionality reduction techniques, leading frequently to information lost and, consequently, accuracy reduction. This work aims at applying linear dimensionality reduction to the SIFT and SURF descriptors. The objective is to demonstrate that even risking to decrease the accuracy of the feature vectors, it results in a satisfactory trade-off between computational time and storage requirements. We perform linear dimensionality reduction through Random Projections (RP), Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Partial Least Squares (PLS) in order to create lower dimensional feature vectors. These new reduced descriptors lead us to less computational time and memory storage requirements, even improving accuracy in some cases. We evaluate such reduced feature vectors in a matching application, as well as their distinctiveness in image retrieval. Finally, we assess the computational time and storage requirements by comparing the original and the reduced feature vectors.

Keywords: linear dimensionality reduction, feature vector, image descriptors.

Address all correspondence to: Helio Pedrini, University of Campinas, Institute of Computing, Av. Albert Einstein 1251, Campinas, SP, Brazil, 13083-852; Tel: +55 19-3521-5919; Fax: +55 19-3521-5847; E-mail: helio@ic.unicamp.br

1 Introduction

Local feature descriptors are employed in the task of describing images[1,2] by representing numerous characteristics such as color, texture and shape. Many of these local descriptors represent simple concepts and might be inadequate to describe an image when employed individually. Furthermore, robust local features group several simple local features or may even build some large dimensional feature vectors based on the neighborhood on an interest point (SIFT[3] and SURF[4] to cite few examples). Consequently, as a large amount of characteristics seems to better describe an image, higher dimensional feature vectors are obtained as the result.

After a local feature is extracted, it can be used over different image processing tasks. However, for such tasks, large dimensional feature vectors will compromise large amounts of storage, high computational time, and could even lead into less accuracy due to noise and redundancy mixed within the data or the curse of dimensionality. Then, it becomes necessary to investigate approaches to solving or at least attenuating this problem.

Several researches have inquired into feature vector compressions.[5] Some others have proposed to reduce the dimensions of a feature vector by projecting it onto a pre-defined projection matrix.[6] Furthermore, there exist interesting approaches involving linear dimensionality reduction, such as PCA[7,8] over any descriptor.[9–11]

Some linear dimensionality reduction techniques consider aspects related to some characteristics of the data to construct the projection matrices. That is, vectors integrating the projection matrices are computed by weighting aspects such as covariance or correlation between the classes and variables, which is expected to conduct any data projected onto a projection matrix for a better reduced description. That is the main reason that motivates us to apply reduction techniques through the task of reducing feature vectors.

This paper aims at applying different linear dimensionality reduction techniques for decreasing SIFT and SURF feature vector dimensionality. The objective of this work is not to create a descriptor by taking into account new characteristics, but to demonstrate that, through the reduction of existing descriptors, it is also possible to enhance their distinctiveness or maintain a similar one, while obtaining improvements on computational time and memory storage.

We perform and evaluate the matching and image retrieval applications by conducting experiments in three categories: matching comparison to evaluate accuracy by using recall-vs-precision curves; image retrieval comparison using the retrieved precision measure to evaluate accuracy; and the description and matching application to measure the consumed computational time and used storage between the original feature vectors and reduced feature vectors.

The remaining of this work is organized as follows. Section 2 briefly describes SIFT and SURF, as well as some different techniques such as Random Projections (RP), Principal Component Analysis (PCA), Partial Least Square (PLS) and Linear Discriminant Analysis (LDA) to perform dimensionality reduction. The evaluation methodology used to validate our results is presented in Section 3. Section 4 shows some results to demonstrate the effectiveness of our methodology. Finally, Section 6 concludes our work.

2 Technical Background

In this section, we review SIFT and SURF feature descriptors. To reduce the dimensionality of a feature vector, we explore and describe some linear dimensionality reduction methods: Random Projections, Principal Component Analysis, Linear Discriminant Analysis and Partial Least Squares.

2.1 Feature Descriptors

In order to better describe image content, several feature descriptors have been developed,[2,12,13] among these are SIFT[3] and SURF[4] descriptors.

The Scale Invariant Feature Transformation (SIFT) algorithm is well-known for detecting interest points and extracting descriptors invariant to scale, rotation, translation, and partially changes in illumination.[14] In the description stage, local image gradients weighted by a Gaussian window are computed for a neighborhood of every interest point detected in a selected scale. A SIFT descriptor is composed of a 128-dimensional vector (8 orientation bins for each 4×4 location bins), which can be considered as a high dimensional descriptor.

Differently from SIFT, the Speeded Up Robust Feature (SURF) algorithm attains to compute a feature vector descriptor which is half of the SIFT feature vector size, while maintaining and even improving the performance over different image distortions. After the detection of interest points,[15] the feature vectors are created by building a grid consisting of 4×4 square sub-regions centered at the interest point. A Haar-like wavelet transform[16] is applied to describe each subregion and the responses (dx, $|dx|$, dy, $|dy|$) are stored in 2×2 subdivisions. As a result, SURF constructs a 64-dimensional vector, whose responses represent the underlying intensity pattern.

Bay et al.[4] established the hypothesis that SURF outperforms SIFT since it integrates the gradient information, whereas the SIFT feature vector represents independent gradient information.[17]

2.2 Linear Dimensionality Reduction Techniques

Linear dimensionality reduction techniques represent a satisfying solution to reduce the feature vector dimensionality, since they are able to compute a projection matrix which can be used to reduce dimensions of a feature vector. Moreover, the reduced feature vectors increase computational speed and decrease space storage needs, while maintaining a similar accuracy to the original feature vectors.

One of the most common techniques in the literature is the Principal Component Analysis (PCA).[7,8,18,19] PCA pursuits to reduce data dimensionality by calculating a covariance matrix and its corresponding eigenvalues and eigenvectors. Subsequently, a projection matrix is formed with the eigenvectors which are ordered according to their respective eigenvalues, from the higher to the lower eigenvalues.

Another alternative to perform dimensionality reduction is using Random Projections (RP). The RP technique[20] constructs an orthogonalized random projection matrix to project data onto it. The basis of this technique is supported by the Johnson-Lindenstrauss lemma,[21,22] which shows that the distances between points in some space is nearly preserved if they are projected onto a randomly selected space. It has also being applied to image and text data.[23]

The Linear Discriminant Analysis (LDA) technique[24–26] is used not only to reduce dimensionality of a given data, but also to discriminate different classes in the data. This is done by maximizing the ratio of the between-class variance and the within-class variance, guaranteeing separability, which is useful for applications where it is possible to linearly separate data into different classes.

As well as LDA, the Partial Least Squares (PLS) technique[27–29] allows performing dimension reduction and classification tasks. This method basically predicts a variable Y from another variable X by finding the projections that maximize the covariance between X and Y, then storing them into a projection matrix. The classical form of the PLS algorithm is based on the NIPALS (Nonlinear Interval Partial Least Squares) algorithm.[30]

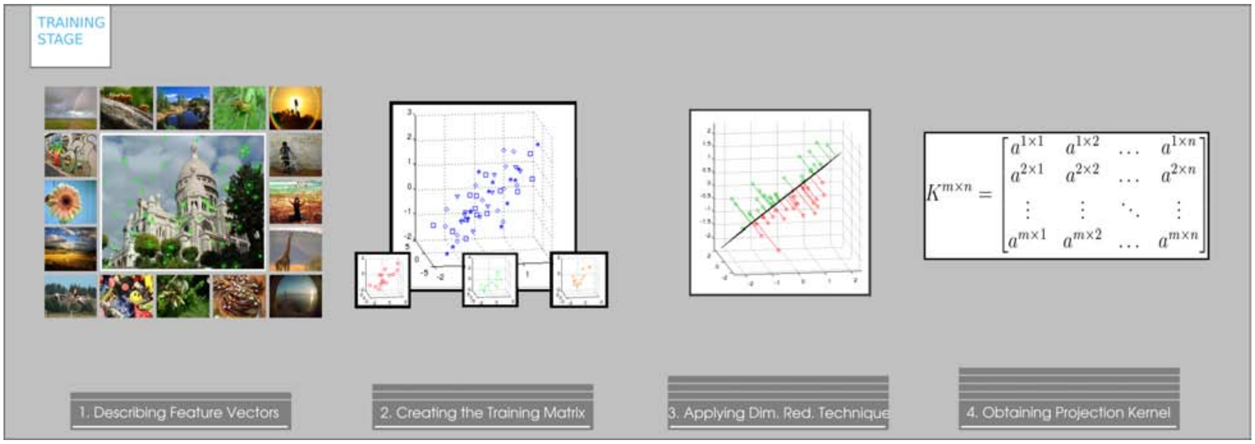
3 Evaluation Methodology

This work aims at reducing the dimensionality of SIFT and SURF feature vectors by applying the dimensionality reduction techniques mentioned in the previous section. To accomplish that, the main steps of the proposed methodology are illustrated in Figure 1. Each stage is described in the following sections.

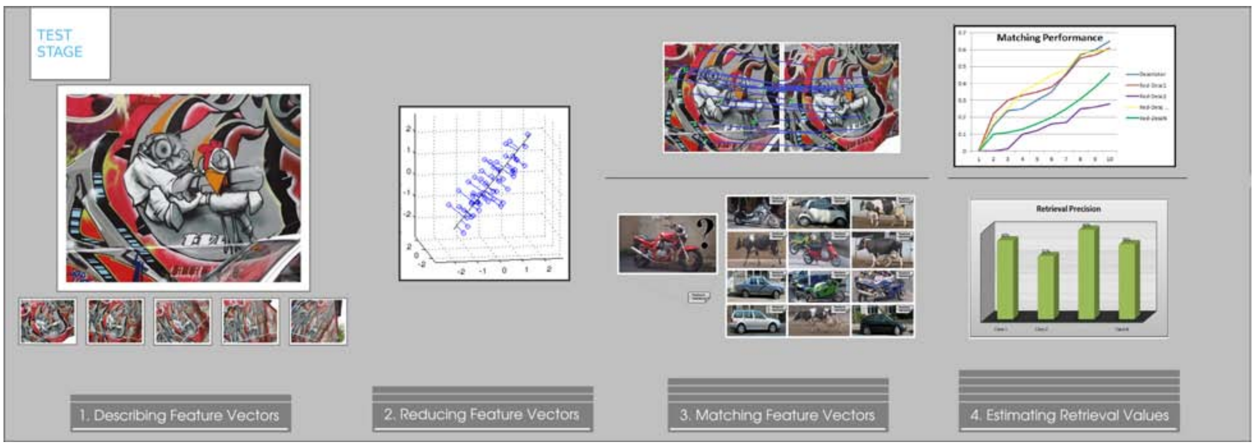
3.1 Learning the Projection Matrices

An off-line computation is performed in this stage. This is due to the expensive computational costs demanded by the linear reduction techniques to calculate a projection matrix. Furthermore, there is no need to re-compute the projection matrices as the considered applications area performed.

In order to learn a projection matrix, the dimensionality reduction technique receives a set of feature vectors grouped into a matrix, referred to as training matrix. This training matrix is denoted as $X^{n \times d}$, where n represents the number of feature vectors and d represents the dimension of each feature vector. After applying the dimensionality reduction, a projection matrix is obtained, denoted as $K^{d \times p}$ (with $p < d$). Then, it is possible to reduce a m-dimensional feature vector to at



(a) Training Stage



(b) Test Stage

Fig 1 Training and test stages. (a) during the training stage, feature vectors are described over images that will not be used in the test stage; the obtained feature vectors are joined into a single training matrix, then a dimensionality reduction technique is applied over this training matrix in order to obtain the projection matrix; (b) the test stage detects and describes interest points over the test images, then it projects them over a projection matrix to obtain the reduced feature vectors, subsequently the matching task is computed between one image and a set of selected images and, finally, a comparison between the matching task and a ground truth is evaluated and shown.

most n dimensions. To reduce feature vectors to lower dimensions, it is unnecessary to recompute the projection matrix, only performing a projection of its descriptors to the projection matrix as

$$T^{l\times p} = M^{l\times d} \times K^{d\times p} \tag{1}$$

resulting in the low dimension matrix T.

3.1.1 Projection Matrix in the Matching Task

A set of randomly selected images, collected from the Mirflickr-1M Dataset,[31] was used to extract $40,000$ feature vectors to compose the training matrix for PCA.

The training matrix for LDA and PLS techniques differs from the PCA matrix in the fact that every feature vector has one more dimension, assuming a value of 1 if it corresponds to an interest point that was detected within the default threshold or 0, otherwise. These arbitrarily selected values allow PLS and LDA techniques to distinguish the difference between both types of interest point descriptor by giving different weights to their related projections.

3.1.2 Projection Matrix in the Image Retrieval Task

The feature vectors of the first seventy-five images from each class were computed to compound the training matrix for PCA. For the LDA and PLS techniques, it is needed to add the identifier of the corresponding class to each feature vector in the training matrix.

3.2 Feature Vector Matching

In order to perform the matching process, two sets of feature vectors, A and B, are necessary with their respective interest point locations. The Euclidean distance, denoted as D_E, is computed from each feature vector in A to each feature vector in B. Then, for each pair of feature vectors in A and B, if their D_E is smaller than an estimated threshold, we consider to have a match between the respective interest points.

In this work, we employ two different approaches to evaluating a corresponding interest point: the nearest neighbor (NN) strategy and the the nearest neighbor distance ratio (NNDR) strategy. The NN strategy selects the corresponding interest points which present the smallest Euclidean distance under the set threshold. On the other hand, the NNDR strategy considers to have a match when the distance ratio between the two smallest Euclidean distances is under the set threshold. If the mentioned statement is true, then it selects the corresponding interest point with smaller Euclidean distance.

The feature vector matching stage is applied in both matching and image retrieval tasks.

4 Experiments and Results

Experiments conducted on two tasks, matching and image retrieval, were proposed to validate the following statement (the data set considered and the evaluation metrics are described in Sections 4.1 and 4.2). SIFT and SURF feature vectors, when reduced to lower dimensions, can maintain or even improve a similar accuracy as they would achieve in the original space (results shown in Sections 4.3 and 4.4). An additional experiment to measure the computational time and storage usage was performed to compare the consumption of these resources when using the original and the reduced feature vectors (results shown in Section 4.5).

4.1 Datasets and Ground Truth

Two datasets were selected to perform matching and image retrieval experiments.

Matching Task For matching experiments, we used the Inria Graffiti Dataset.[32] This dataset contains 8 groups of images (6 images per group). Each group of images is subject to different geometric and photometric transformations such as rotation, scaling, blurring, warping, illumination variance, and JPEG compression. The first three sets of transformed images have two inner subsets, one of them contains images with distinctive edge boundaries, the other one contains repeated textures of different forms.

Every group of images, in the Inria Graffiti Dataset, contains five 3×3 homography matrices. Each of these homography matrices represents a projective transformation from the first image to one of the other five images, which allows us to map any point from the first image to any other image belonging to its group. Therefore, to validate a match, it is necessary to have two interest points: p in the first image and q in any other image, denoted by i, belonging to the same group. The homography matrix related to the image 1 (base image) to the image i allows to map p into i, obtaining p'. Then, p and q are considered a correct match if p' and q are sufficiently close in space and scale. As mentioned in,[33] two points are close in space if the distance between them is less than σ pixels, where σ is the standard deviation to generate the used scale. Two points are close in scale if their scales are within $\sqrt{2}$ of each other.

Image Retrieval Task For image retrieval experiments, we employed the TU Darmstadt Dataset.[34] This dataset contains 300 images divided into three categories: cars, motorbikes and cows, where each one contains 100 images.

4.2 Evaluation Metrics

This subsection presents the metrics employed to measure results, matching and image retrieval experiments, to better understand the results obtained through the experiments.

Matching Task To evaluate the matching performance, we use *recall vs. 1-precision* curves, as recommended in.[35] Recall (Equation 2) represents the measure of the ratio between the number of correct matches retrieved over the total of matches that are expected to be retrieved. It is important to notice that a recall of 100% would be achieved if a set with all possible matches is returned, so if the recall measure is presented alone, it loses its relevance. Therefore, since the precision measure means the ratio between the number of correct matches retrieved over the total of matches retrieved, the 1-precision value is also considered to know when the recall represents some good result. These two measures are defined as

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives + False Negatives}} \tag{2}$$

$$\text{1-Precision} = \frac{\text{False Positives}}{\text{True Positives + False Positives}} \tag{3}$$

Image Retrieval Task As there is still a remaining of 25 images per class from the TU dataset and saved to be used in the testing stage, we mixed them into the set and compute for each image a rank list of the top corresponding 24 images. The ranking list can also be performed for different threshold values. Finally, from all these lists we found the best threshold to perform our experiments.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives + False Positives}} \tag{4}$$

4.3 Image Matching Task

In this experiment, SIFT and SURF feature vectors are compared to their respective reduced feature vectors by using the recall vs 1-precision curves. The original feature vectors were projected onto the projection matrices built with RP, PCA, LDA and PLS methods. It is important to notice that the sequence composed of dimension reduction with PCA and feature extraction with SIFT, tested in this work (referred to as PCA+SIFT), is not the same as the approach called PCA-SIFT proposed in.[33,36]

The recall vs 1-precision comparison curves are shown in Figures 2, 3, 4 and 5. Figures 2 and 4 show plots for every transformation, represented in the Inria Dataset, using the NN strategy, whereas Figures 3 and 5 show plots for every transformation using the NNDR strategy. In addition, each plot presents the lower dimensional feature vector that achieved an accuracy similar to its respective original feature vector. SIFT feature vectors were reduced to 12, 20, 32, 36, 46 and 64 dimensions, whereas SURF feature vectors were reduced to 12, 20, 32 and 36 dimensions.

Figures 2 and 3 show that PCA+SIFT achieved similar results to the original SIFT descriptor using only 32, 36 and 64 dimensions. This means that the PCA+SIFT descriptor can achieve a high accuracy even when the feature vector is reduced to 25 to 50% of its original size.

Figures 4 and 5 also show that PCA+SURF achieved similar responses to the original SURF descriptor using 20 and 32 dimensions. This means that the PCA+SURF descriptor can achieve a high accuracy even when the feature vector is reduced to 31.25 to 50% of its original size.

For both descriptors, the PCA method performed better. Figures 6 and 7 compare SIFT and SURF performance to their corresponding PCA+SIFT and PCA+SURF using different dimensions. It is shown that PCA+SIFT achieves a similar accuracy to SIFT at 32 dimensions, as well as PCA+SURF achieves a similar accuracy at 20 dimensions.

4.4 Image Retrieval Task

Image retrieval experiments were conduced to demonstrate that the reduced feature vectors can perform as well or even better than the original feature vectors.

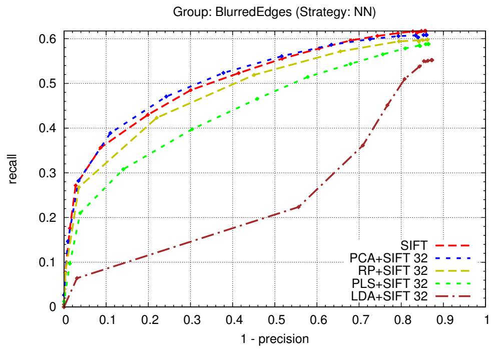
This experiment differs from the one presented in[10] because of the nature of the data set used. The data set used in[10] considered 30 images separated into 10 groups, where each group contained the same object viewed from a different angle. On the other hand, the TU Darmstadt data set contains 300 images separated into 3 groups, and normally each group presents different objects. Such difference leads to a lower retrieval precision.

The SIFT and SURF feature vectors were reduced to a set of 2, 4, 8, 12, 20, 32, 36, 46, 64 dimensions and 2, 4, 8, 12, 20, 32, 36 dimensions, respectively.

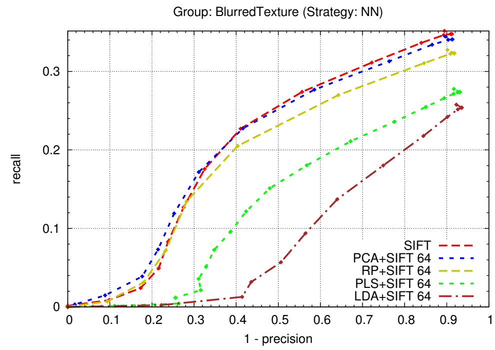
Each table presented in this subsection contains the following fields: descriptor, which refers to the original descriptor or a reduced descriptor; dimensions indicating the number of dimensions used by each descriptor; threshold, which indicates the value where the best retrieval was obtained; percentage, which indicates the position in the interval where the current threshold is suitable; the retrieval indicating the percentage of correct images retrieved over the total of images retrieved.

It is important to observe that the percentage reveals the tolerance of the current threshold. A high percentage means a high threshold tolerance, which leads to a higher imprecision.

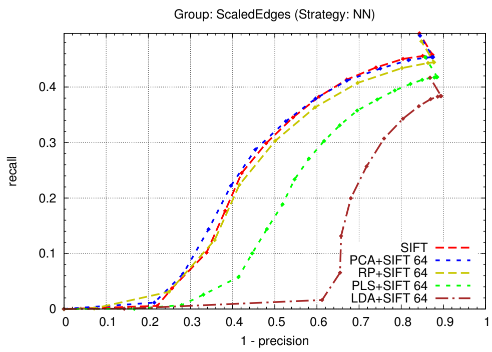
Figure 8 presents a comparison between the retrieval results computed for SIFT feature vectors and every reduced feature vector computed. It can be seen that the features reduced by using the PCA projection matrix outperformed all others. Table 1 focuses on the 32-dimensional feature



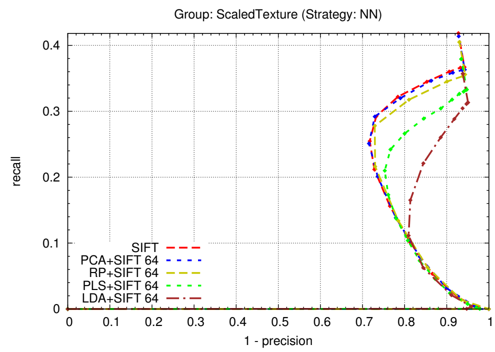
(a) Blurred Edges - NN



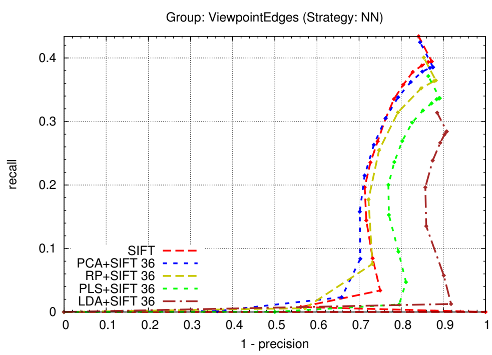
(b) Blurred Texture - NN



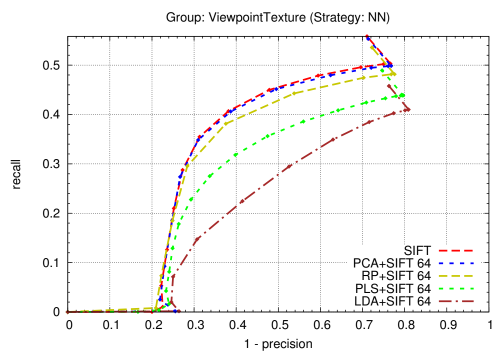
(c) Scaled Edges - NN



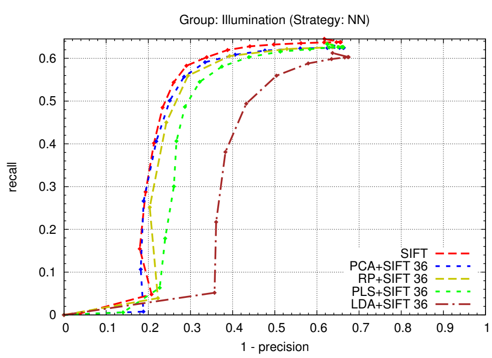
(d) Scaled Texture - NN



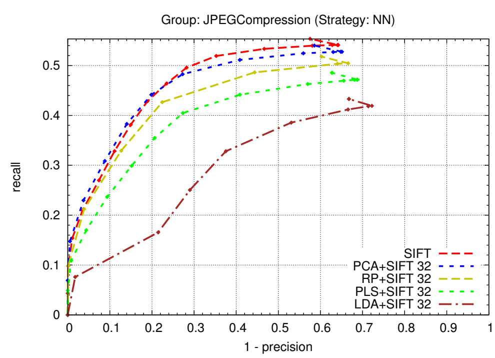
(e) Viewpoint Edges - NN



(f) Viewpoint Texture - NN

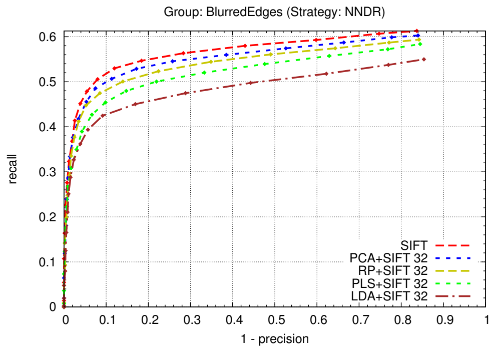


(g) Illumination - NN

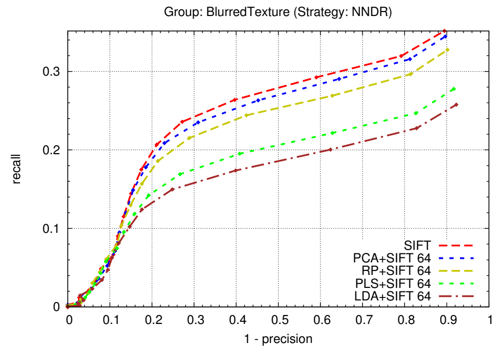


(h) JPEG Compression - NN

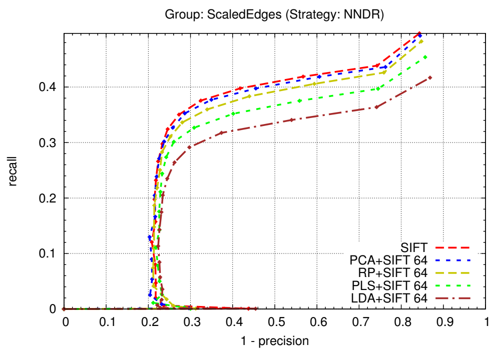
Fig 2 SIFT Feature Vectors and corresponding Reduced Feature Vector comparison using the NN strategy



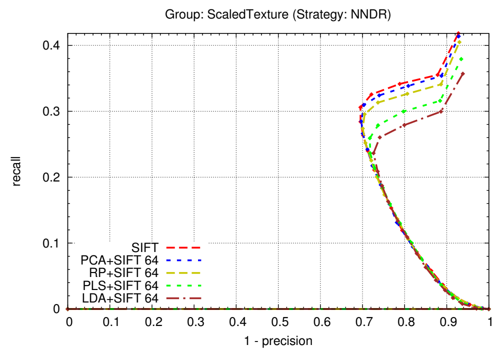
(a) Blurred Edges - NNDR



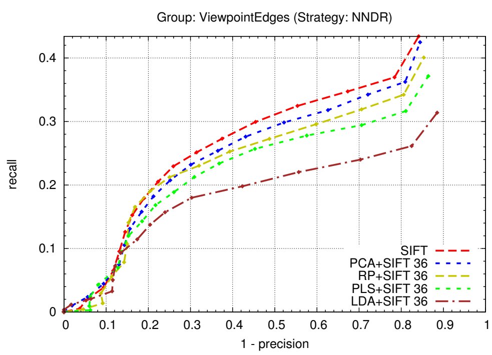
(b) Blurred Texture - NNDR



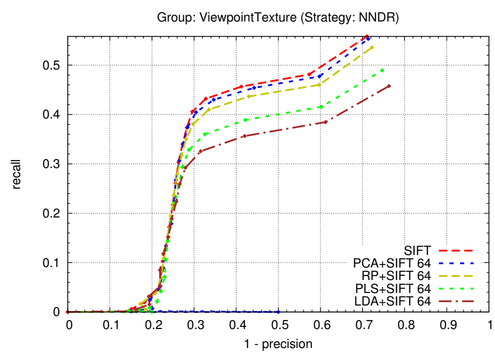
(c) Scaled Edges - NNDR



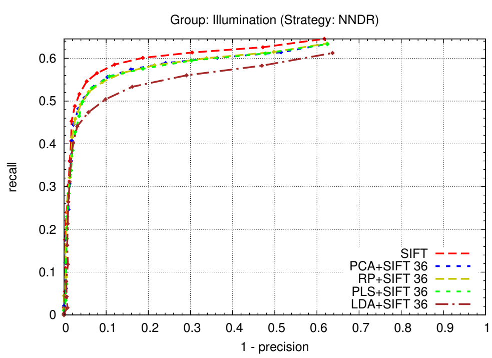
(d) Scaled Texture - NNDR



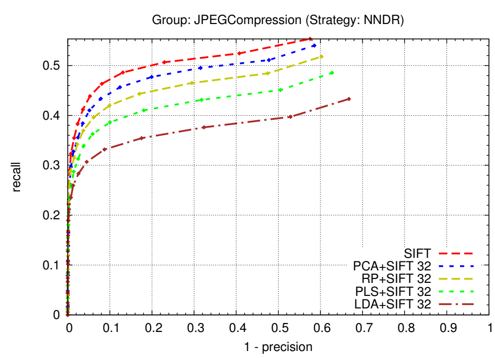
(e) Viewpoint Edges - NNDR



(f) Viewpoint Texture - NNDR

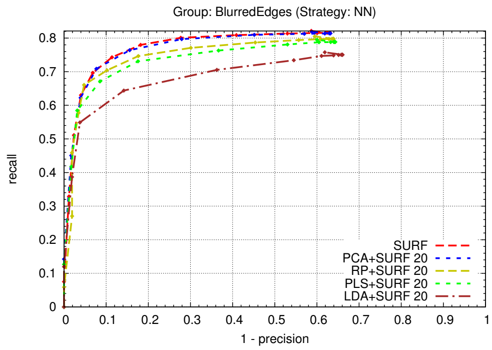


(g) Illumination - NNDR

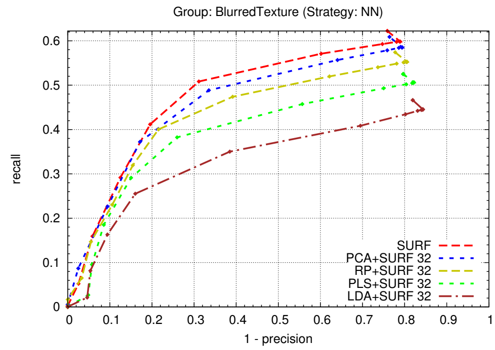


(h) JPEG Compression - NNDR

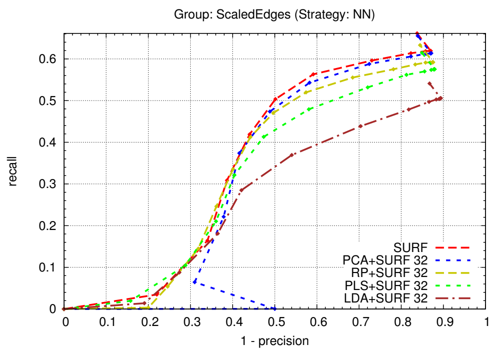
Fig 3 SIFT Feature Vectors and corresponding Reduced Feature Vector comparison using the NNDR strategy



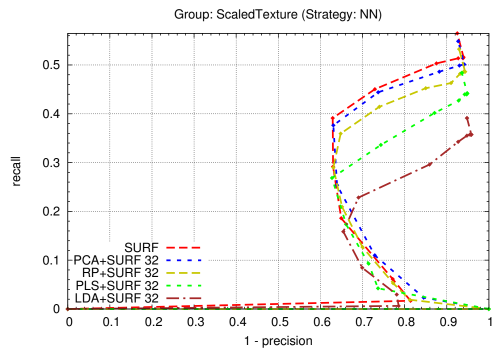
(a) Blurred Edges - NN



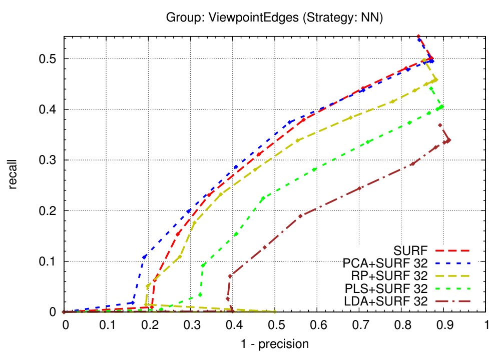
(b) Blurred Texture - NN



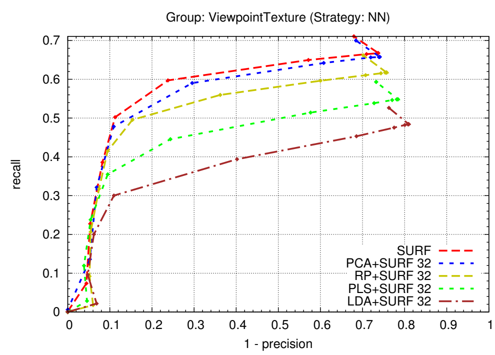
(c) Scaled Edges - NN



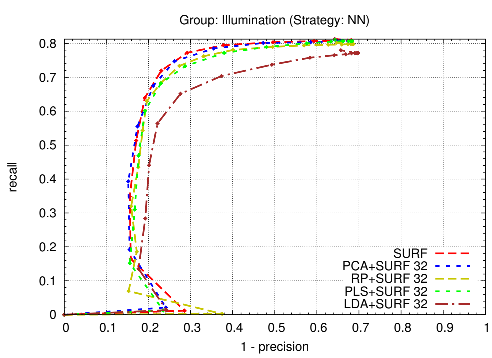
(d) Scaled Texture - NN



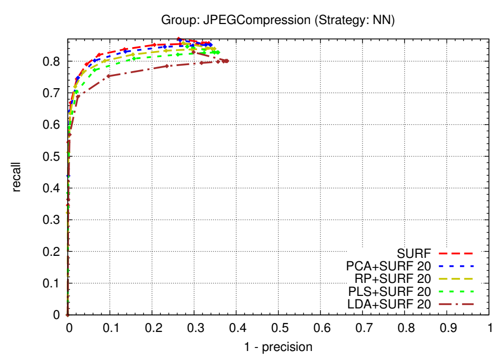
(e) Viewpoint Edges - NN



(f) Viewpoint Texture - NN

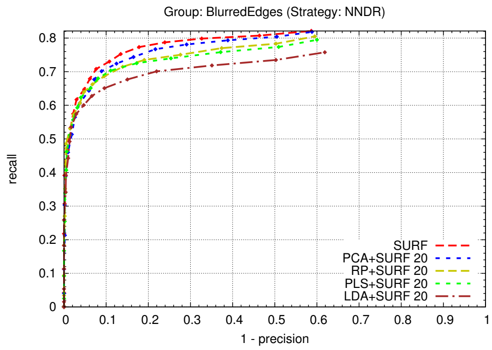


(g) Illumination - NN

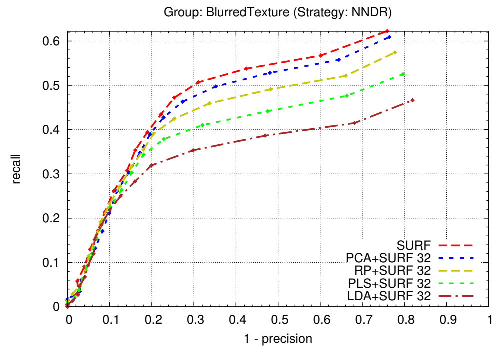


(h) JPEG Compression - NN

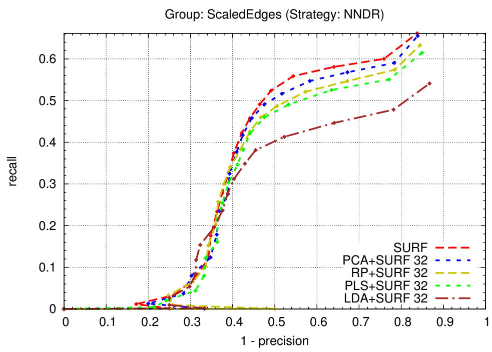
Fig 4 SURF Feature Vectors and corresponding Reduced Feature Vector comparison using the NN strategy



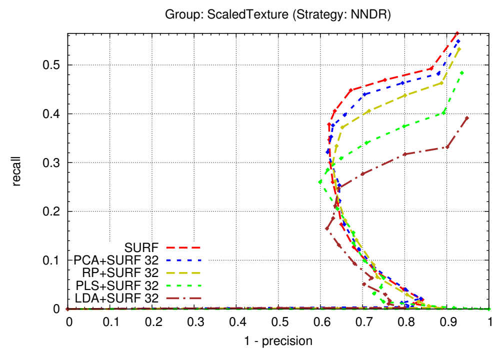
(a) Blurred Edges - NNDR



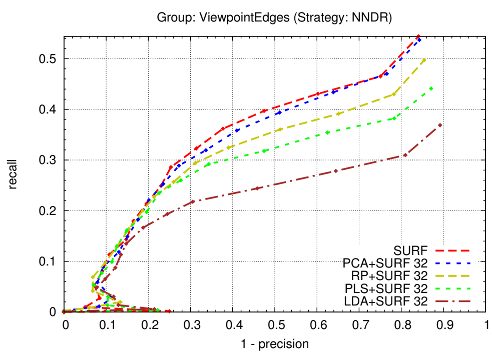
(b) Blurred Texture - NNDR



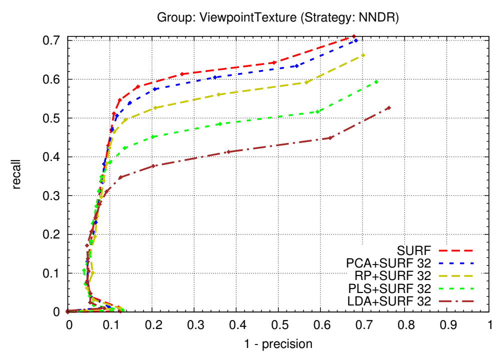
(c) Scaled Edges - NNDR



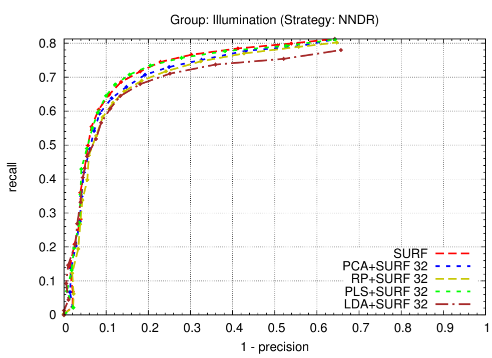
(d) Scaled Texture - NNDR



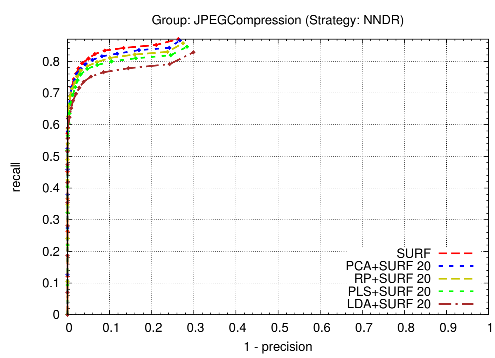
(e) Viewpoint Edges - NNDR



(f) Viewpoint Texture - NNDR

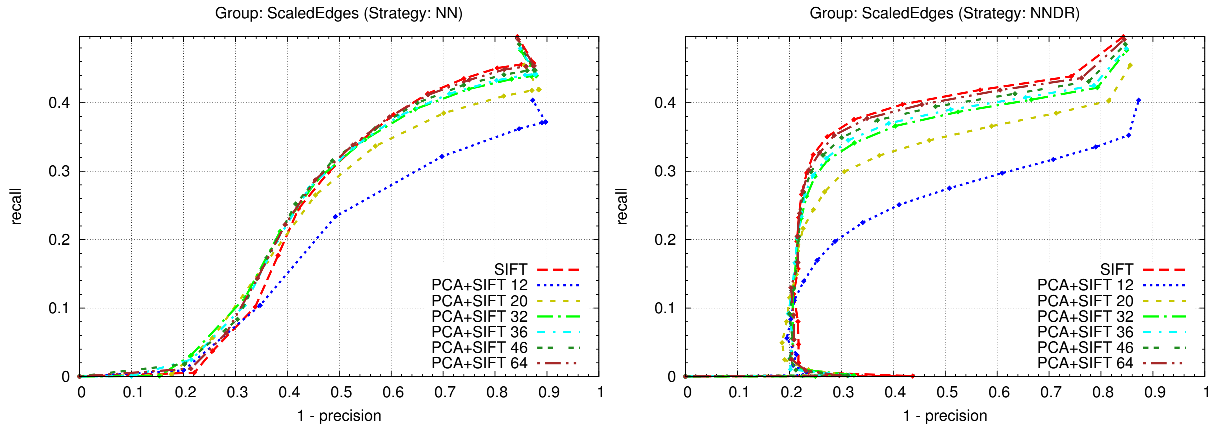


(g) Illumination - NNDR



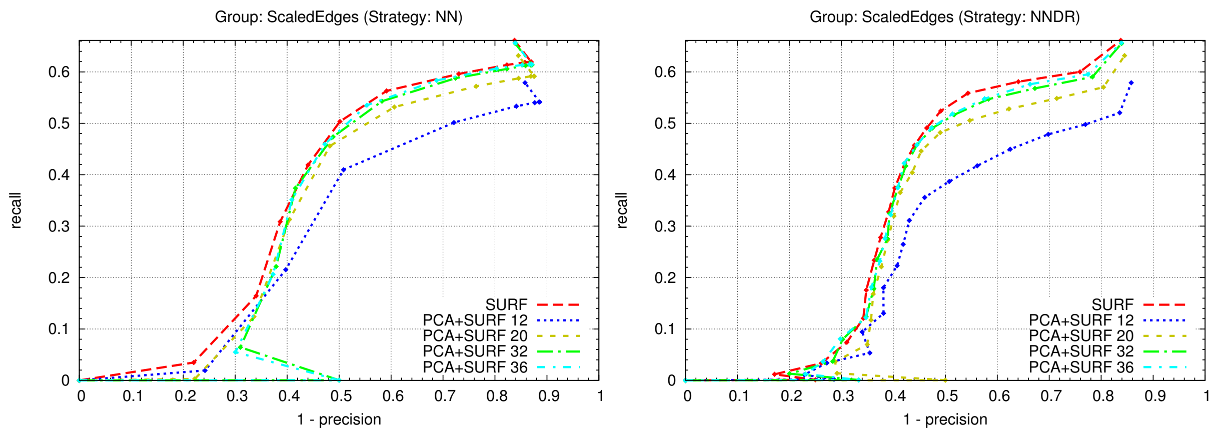
(h) JPEG Compression - NNDR

Fig 5 SURF Feature Vectors and corresponding Reduced Feature Vector comparison using the NNDR strategy



(a) Different dimensions - NN (b) Different dimensions - NNDR

Fig 6 Performance achieved by PCA+SIFT with both strategies at different dimensions



(a) Different dimensions - NN (b) Different dimensions - NNDR

Fig 7 Performance achieved by PCA+SURF with both strategies at different dimensions

Table 1 Reduced-SIFT - Retrieval Comparisons (Str: NN).

Descriptor	Dims.	Thr.	Perc.	Retrieval
SIFT	128	100.00	16.67%	57.81%
RP+SIFT	32	50.00	16.67%	57.60%
PCA+SIFT	32	75.00	15.00%	59.20%
LDA+SIFT	32	4.00	33.33%	49.12%
PLS+SIFT	32	0.01	33.33%	53.39%

vector results since it achieved a retrieval value close to the achieved by the original descriptor. Furthermore, as PCA technique demonstrated to perform better, Table 2 presents in detail the retrieval values achieved by the PCA+SIFT descriptor with different feature dimensions. It can be seen that, between 12 and 36 dimensions, the retrieval response is usually higher and more precise than considering the original descriptor.

Figure 9 shows a comparison between the retrieval results achieved with the NNDR strategy for SIFT feature vectors and every reduced feature vector. In this case, the features projected onto the LDA projection matrix and onto PLS projection matrix outperformed the other reduced features.

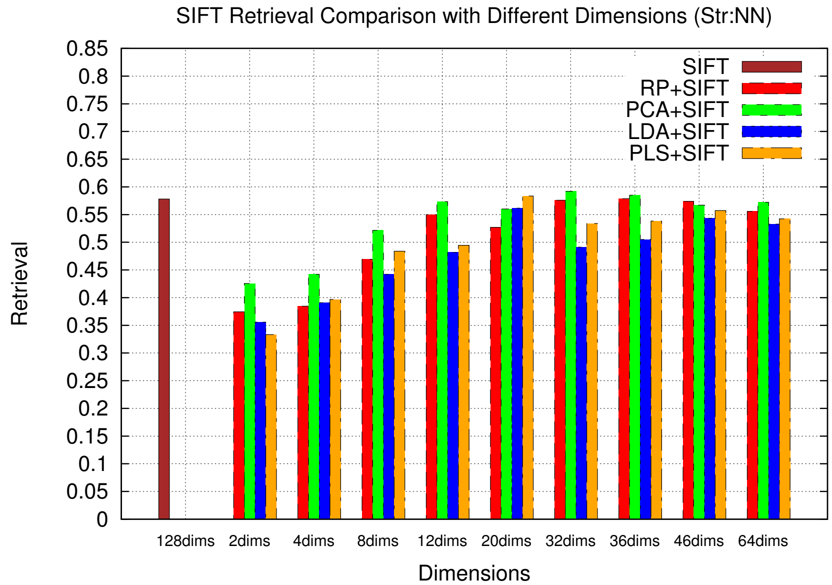


Fig 8 SIFT - Retrieval Comparisons (Str: NN).

Table 2 Retrieval Comparison between Dimensions (Str: NN).

Descriptor	Dims.	Thr.	Perc.	Retrieval
SIFT	128	100.0	16.67%	57.18%
PCA+SIFT	2	50.0	25.00%	42.56%
PCA+SIFT	4	75.0	25.00%	44.21%
PCA+SIFT	8	25.0	7.14%	52.16%
PCA+SIFT	12	50.0	13.33%	57.33%
PCA+SIFT	20	50.0	11.11%	56.00%
PCA+SIFT	32	75.0	15.00%	59.20%
PCA+SIFT	36	75.0	15.00%	58.51%
PCA+SIFT	46	75.0	14.29%	56.69%
PCA+SIFT	64	100.0	18.18%	57.23%

We believe this is due to the classification ability.

Table 3 Reduced-SIFT - Retrieval Comparisons (Str: NNDR).

Descriptor	Dims.	Thr.	Perc.	Retrieval
SIFT	128	0.525	52.50%	58.24%
RP+SIFT	36	0.525	52.50%	57.49%
PCA+SIFT	36	0.475	47.50%	57.76%
LDA+SIFT	36	0.600	60.00%	59.20%
PLS+SIFT	36	0.525	52.50%	58.72%

Table 3 focuses on the 36 dimensional feature vector achieving better retrieval than the original feature vector. Table 4 presents the retrieval values achieved by the LDA+SIFT descriptor for different dimensions. It can be seen that, between 32 and 64 dimensions, the retrieval response is higher than the original descriptor, however, the precision is lower.

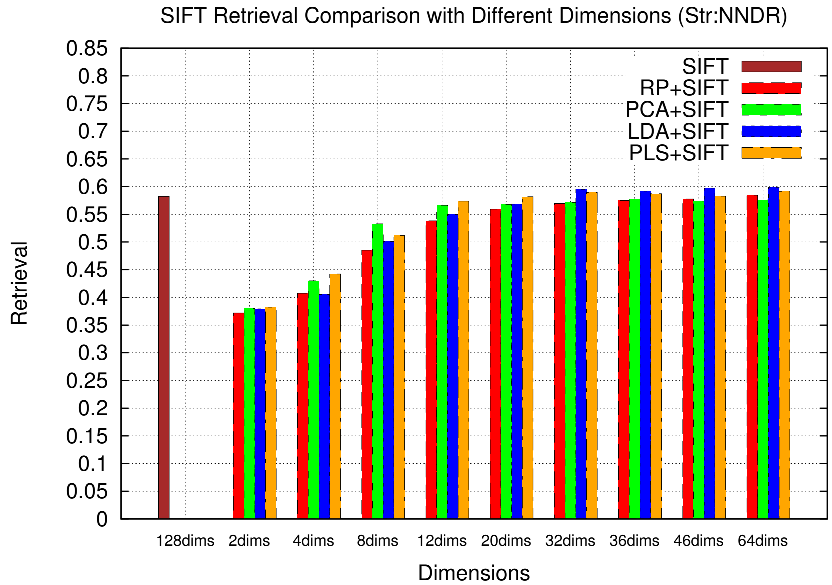


Fig 9 SIFT - Retrieval Comparisons (Str: NNDR).

Table 4 Retrieval Comparison between Dimensions (Str: NNDR).

Descriptor	Dims.	Thr.	Perc.	Retrieval
SIFT	128	0.525	52.50%	58.24%
LDA+SIFT	2	0.850	85.00%	37.92%
LDA+SIFT	4	0.700	70.00%	40.53%
LDA+SIFT	8	0.325	32.50%	50.08%
LDA+SIFT	12	0.425	42.50%	54.99%
LDA+SIFT	20	0.550	55.00%	56.85%
LDA+SIFT	32	0.600	60.00%	59.52%
LDA+SIFT	36	0.600	60.00%	59.20%
LDA+SIFT	46	0.600	60.00%	59.73%
LDA+SIFT	64	0.650	65.00%	59.89%

Figure 10 shows the NN strategy obtaining better results than the NNDR strategy. In this experiment, the PCA reduced feature vectors achieved a better performance, being reduced to 18.75% of the original size. Table 5 shows that the other reduced descriptors achieved close results to the original feature vector for 12 dimensions. Table 6 presents the retrieval values achieved by the PCA+SURF descriptor for different dimensions. It can be seen that, between 12 and 36 dimensions, the retrieval and precision responses are higher than considering the original descriptor.

Table 5 Reduced-SURF - Retrieval Comparisons (Str: NN).

Descriptor	Dims.	Thr.	Perc.	Retrieval
SURF	64	0.425	38.64%	54.13%
RP+SURF	12	0.075	14.29%	52.85%
PCA+SURF	12	0.200	22.22%	54.29%
LDA+SURF	12	1.000	11.76%	49.33%
PLS+SURF	12	0.005	11.76%	52.00%

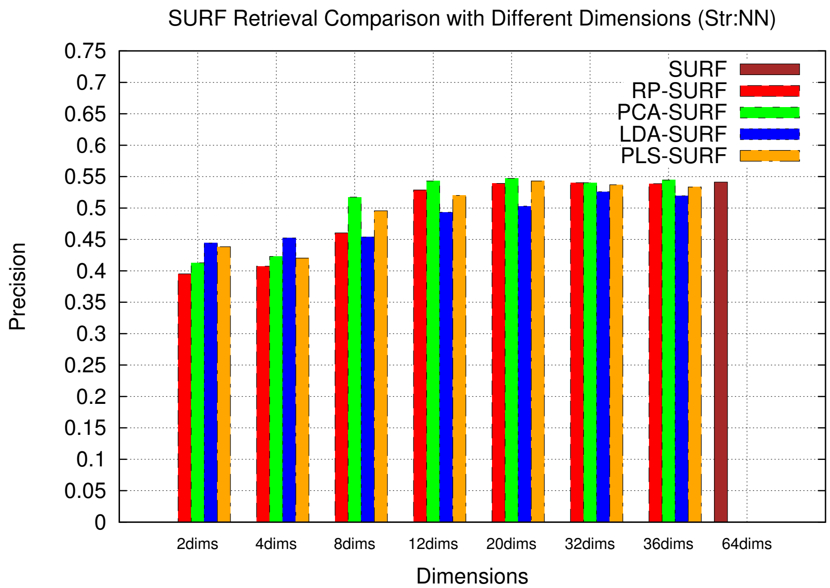


Fig 10 SURF - Retrieval Comparisons (Str: NN).

Table 6 Retrieval Comparison between Dimensions (Str: NN).

Descriptor	Dims.	Thr.	Perc.	Retrieval
SURF	64	0.425	38.64%	54.13%
PCA+SURF	2	0.100	18.18%	41.28%
PCA+SURF	4	0.050	71.43%	42.29%
PCA+SURF	8	0.150	18.75%	51.73%
PCA+SURF	12	0.200	22.22%	54.29%
PCA+SURF	20	0.300	31.58%	54.72%
PCA+SURF	32	0.350	33.33%	54.03%
PCA+SURF	36	0.400	38.10%	54.45%

Table 7 Reduced-SURF - Retrieval Comparisons (Str: NNDR).

Descriptor	Dims.	Thr.	Perc.	Retrieval
SURF	64	0.650	65.00%	50.56%
RP+SURF	32	0.575	57.50%	50.45%
PCA+SURF	32	0.625	62.50%	50.99%
LDA+SURF	32	0.575	57.50%	48.32%
PLS+SURF	32	0.625	62.50%	50.35%

Finally, Figure 11 shows a comparison between the retrieval results computed with the NNDR strategy for SURF feature vectors and each reduced feature vector. Once again, the PCA reduced features obtained superior results. Table 7 focuses on the 32 dimensional feature vectors reduced with the PCA projection matrix, achieving better results than the original feature vector. The PLS reduced feature vector retrieval is close to the original feature vector retrieval. Table 8 presents the retrieval values achieved by the PCA+SURF descriptor for different dimensions, where features reduced to 32 and 36 dimensions obtained higher accuracy while having a lower imprecision than the original feature.

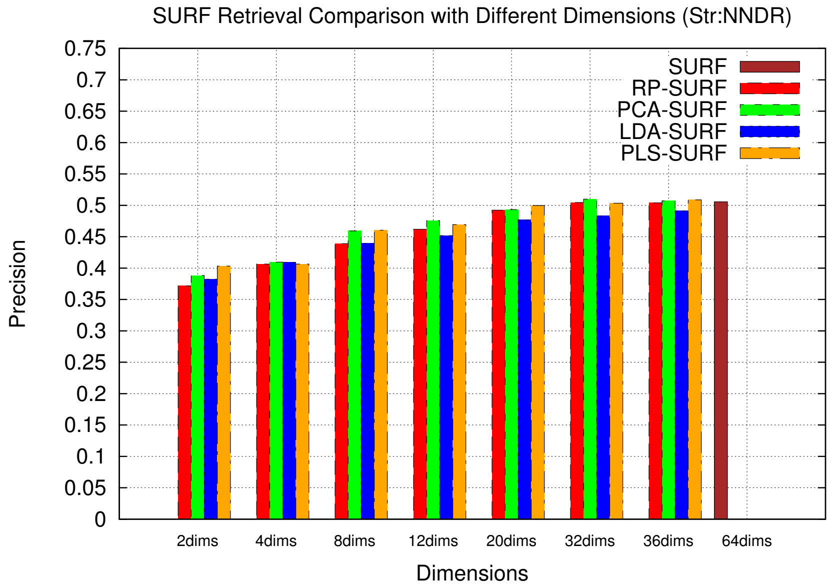


Fig 11 SURF - Retrieval Comparisons (Str: NNDR).

Table 8 Retrieval Comparison between Dimensions (Str: NNDR).

Descriptor	Dims.	Thr.	Perc.	Retrieval
SURF	64	0.650	65.00%	50.56%
PCA+SURF	2	0.225	22.50%	38.83%
PCA+SURF	4	0.875	87.50%	40.96%
PCA+SURF	8	0.700	70.00%	45.92%
PCA+SURF	12	0.500	50.00%	47.57%
PCA+SURF	20	0.575	57.50%	49.33%
PCA+SURF	32	0.625	62.50%	50.99%
PCA+SURF	36	0.625	62.50%	50.72%

4.5 Memory Storage and Computational Time

To perform this experiment, we randomly selected $10,000$ images from the Mirflickr-1M Dataset. Computational time and memory storage required to compute these processes can be seen in the following figures and tables presented in this subsection. We used an Intel Core i7-2670QM CPU computer with 2.20 GHz and 8 Gbytes of RAM.

Tables 9 and 10 show the space required to store $10,000$ image descriptors using an average of 804 and 325 keypoints per SIFT and SURF description per image, respectively. On the other hand, Figures 12 and 13 present a comparison to better observe the benefits, in terms of space required, of using reduced feature vectors. As it can be observed, the descriptor storage is proportional to their dimensions. The most distinctive PCA+SIFT with 36-dimensions and PCA+SURF with 32 dimensions use approximately one third of the memory required by their respective original SIFT and SURF descriptors.

Tables 11 and 12 show the computational time, in minutes, consumed to finally perform the matching process. First, the description stage is performed. Then, SIFT and SURF original feature vectors are ready to start the matching process, while the reduction stage is still needed in order to

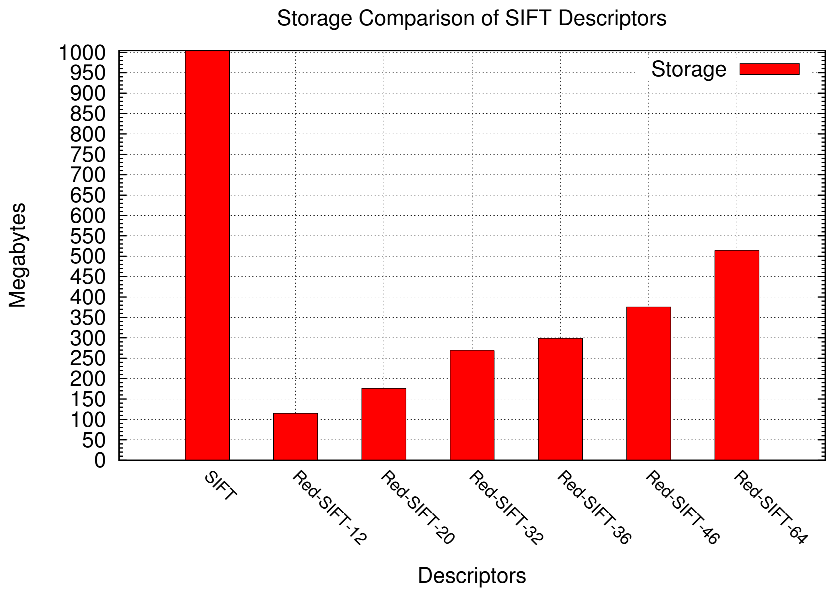


Fig 12 SIFT Descriptors - Storage Comparison.

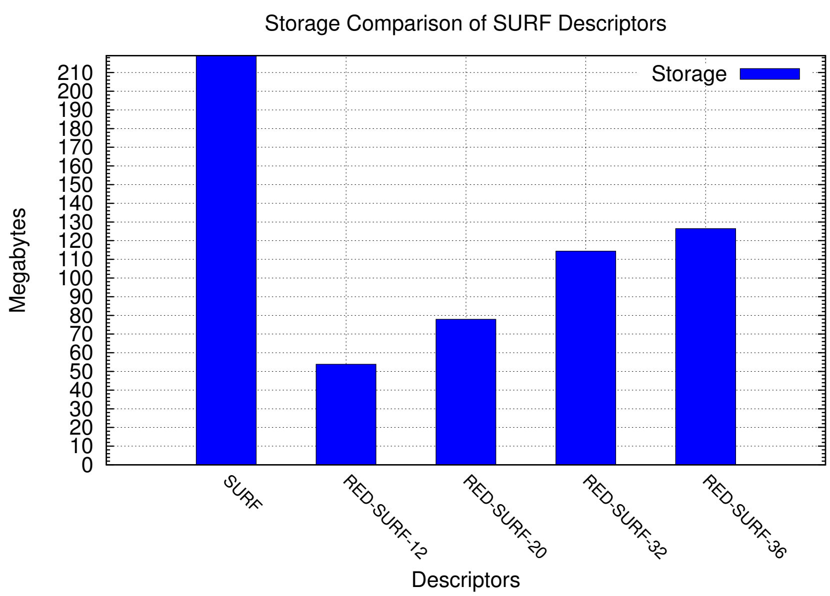


Fig 13 SURF Descriptors - Storage Comparison.

obtain the reduced feature vectors. Once the reduction stage is performed, every reduced descriptor is ready to be matched.

Description and reduction stages are computed once, which is not supposed to occur with the matching stage. In these experiments, the first two stages were computed over the selected $10,000$ images, but only 10 images (with the average number of keypoints) were selected to be matched against all the others. Note that performing an all-vs-all matching leads to a dominating time on matching, where description and reduction times would be hardly perceived, and the gap between matching times for feature vectors and reduced feature vectors would increase significantly.

Finally, it is important to note that memory storage and computational time requirements for

Table 9 SIFT descriptor storage.

Descriptor	Dims.	Storage
SIFT	128	1004.09 MB
Reduced-SIFT	12	115.01 MB
Reduced-SIFT	20	173.33 MB
Reduced-SIFT	32	268.30 MB
Reduced-SIFT	36	298.96 MB
Reduced-SIFT	46	375.60 MB
Reduced-SIFT	64	513.56 MB

Table 10 SURF descriptor storage.

Descriptor	Dims.	Storage
SURF	64	219.07 MB
Reduced-SURF	12	53.85 MB
Reduced-SURF	20	77.95 MB
Reduced-SURF	32	114.35 MB
Reduced-SURF	36	126.43 MB

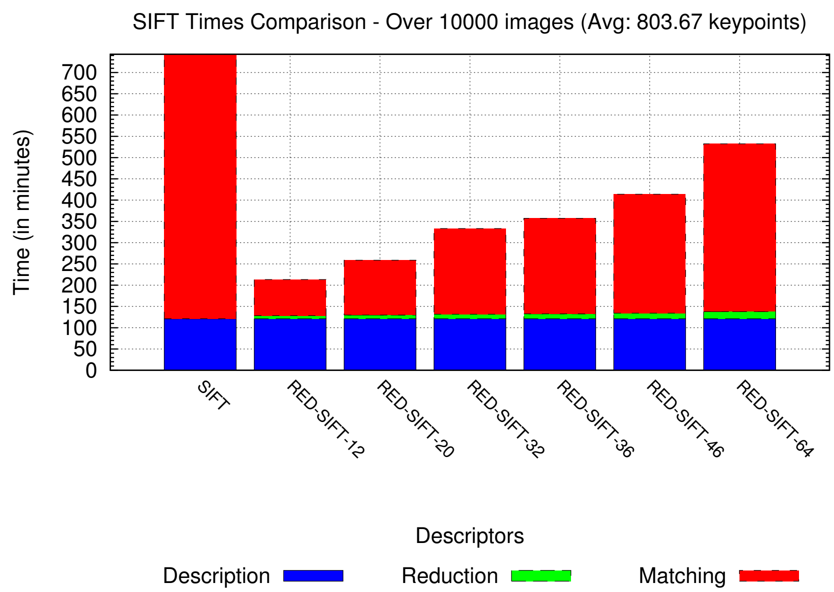


Fig 14 SIFT - Time for matching comparison.

the SIFT descriptor are not proportional to the SURF requirements, since both descriptors can detect different number of interest points to describe a same image.

5 Discussion

This work shows experimental evidences regarding the viability of reducing feature vectors, up to 90% of their original sizes, while maintaining or even improving the accuracy and precision achieved by their original feature vectors.

Feature vectors reduced by the projection matrix PCA performed better in the majority of the cases. This can be due to the fact that it takes into consideration the relation between features.

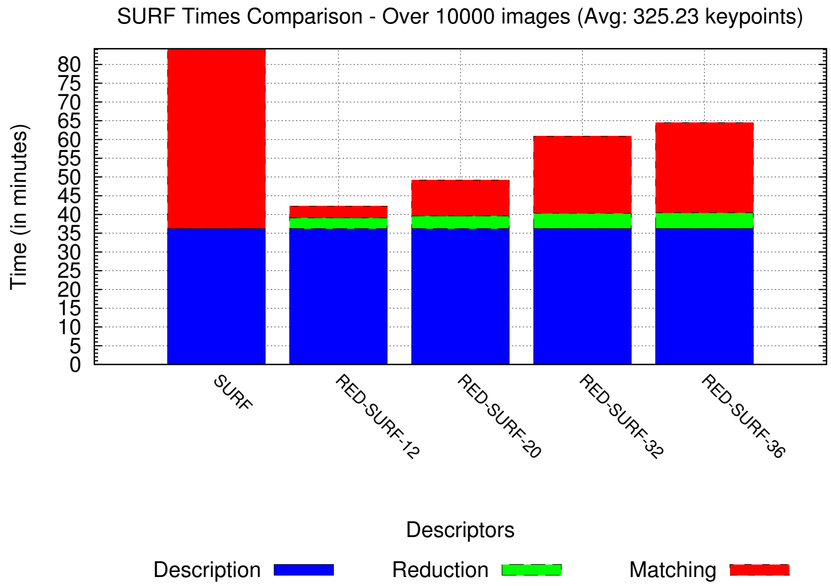


Fig 15 SURF - Time for matching comparison.

Table 11 SIFT - Time to perform Matching.

		Time in minutes		
Descriptor	**Dims.**	**Desc.**	**Reduc.**	**Matching**
SIFT	128	120.99	0.00	621.90
Reduced-SIFT	12	120.99	7.49	84.80
Reduced-SIFT	20	120.99	8.77	129.10
Reduced-SIFT	32	120.99	10.89	201.30
Reduced-SIFT	36	120.99	11.65	224.50
Reduced-SIFT	46	120.99	13.40	279.60
Reduced-SIFT	64	120.99	16.62	394.90

Table 12 SURF - Time to perform Matching.

		Time in minutes		
Descriptor	**Dims.**	**Desc.**	**Reduc.**	**Matching**
SURF	64	36.26	0.00	47.93
Reduced-SURF	12	36.26	2.84	3.11
Reduced-SURF	20	36.26	3.36	9.57
Reduced-SURF	32	36.26	4.08	20.54
Reduced-SURF	36	36.26	4.22	24.00

Therefore, the PCA technique is suitable for applications where there is no need for classification.

The Random Projection technique can construct a projection matrix faster than the other techniques since it does not take any data into consideration. However, this feature is not interesting for this work since it pre-computes the projection matrices due to its fixed training set.

We believe that LDA and PLS techniques can perform even better than PCA for applications where it is important to identify several classes. As it was demonstrated through the image retrieval experiments, with SIFT descriptors and the NNDR strategy, LDA and PLS yielded superior results. Additional tests with more classes are intended to be performed as future work to prove our premise.

The storage required for a reduced feature vector is far lower than the required for an original feature vector. This represents a significant advantage in using the reduced set of features.

The extra computational time spent in reducing the feature vectors is well paid off when the application demands exhaustive work with the described reduced feature vectors, such as in image retrieval and classification tasks.

6 Conclusions

This work shows experimental evidences regarding the viability of reducing feature vectors, up to 90% of their original sizes, while maintaining or even improving the accuracy and precision achieved by their original feature vectors.

Feature vectors reduced by the projection matrix PCA performed better in the majority of the cases. This can be due to the fact that it takes into consideration the relation between features. Therefore, the PCA technique is suitable for applications where there is no need for classification.

The Random Projection technique can construct a projection matrix faster than the other techniques since it does not take any data into consideration. However, this feature demonstrated to be not interesting to this work since it pre-computes the projection matrices due to its fixed training set.

We believe that LDA and PLS techniques can perform even better than PCA for applications where it is important to identify several classes. As it was demonstrated through the image retrieval experiments, with SIFT descriptors and the NNDR strategy, LDA and PLS yielded superior results. Additional tests with more classes are intended to be performed as future work to prove our premise.

The storage required for a reduced feature vector is far lower than the required for an original feature vector. This represents a significant advantage in using the reduced set of features.

The extra computational time spent in reducing the feature vectors is well paid off when the application demands exhaustive work with the described reduced feature vectors, such as in image retrieval and classification tasks.

7 Acknowledgements

The authors are thankful to FAPESP, CNPq, FAPEMIG and CAPES for their financial support.

References

1 T. Tuytelaars and K. Mikolajczyk, "Local Invariant Feature Detectors: A Survey," in *Foundation and Trends in Computer Graphics and Vision*, 177–280 (2008).

2 K. Mikolajczyk and C. Schmid, "A Performance Evaluation of Local Descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **27**, 1615–1630 (2005).

3 D. G. Lowe, "Object Recognition from Local Scale-Invariant Features," in *IEEE International Conference on Computer Vision*, 1150–1157, IEEE Computer Society (1999).

4 H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded Up Robust Features," in *European Conference in Computer Vision*, 404–417 (2006).

5 V. Chandrasekhar, M. Makar, G. Takacs, D. Chen, S. S. Tsai, N.-M. Cheung, R. Grzeszczuk, Y. Reznik, and B. Girod, "Survey of SIFT Compression Schemes," in *Second International Workshop on Mobile Multimedia Processing*, (Istanbul, Turkey) (2010).

6 G. Zhao, L. Chen, G. Chen, and J. Yuan, "KPB-SIFT: A Compact Local Feature Descriptor," in *ACM Multimedia*, 1175–1178 (2010).

7 K. Pearson, "On Lines and Planes of Closest Fit to Systems of Points in Space," *Philosophical Magazine* **2**(6), 559–572 (1901).

8 I. Jolliffe, *Principal Component Analysis*, Springer-Verlag, New York (2002).

9 N. Watcharapinchai, S. Aramvith, S. Siddhichai, and S. Marukatat, "Dimensionality Reduction of SIFT using PCA for Object Categorization," in *International Symposium on Intelligent Signal Processing and Communications Systems*, 1–4, (Bangkok, Thailand) (2009).

10 R. Valenzuela, W. Schwartz, and H. Pedrini, "Dimensionality Reduction Through PCA over SIFT and SURF Descriptors," in *11th IEEE Conference on Cybernetic Intelligent Systems*, 58–63, (Limerick, Ireland) (2012).

11 R. Valenzuela, H. Pedrini, and W. Schwartz, "Dimensionality Reduction Through LDA and Bag-of-Features Applied to Image Retrieval," in *IV ECCOMAS Thematic Conference on Computational Vision and Medical Image Processing*, 31–37, (Funchal, Portugal) (2013).

12 T. Randen and J. H. Husoy, "Filtering for Texture Classification: A Comparative Study," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **21**, 291–310 (1999).

13 J. Li and A. Nigel, "A Comprehensive review of Current Local Features for Computer Vision," *Neurocomputing* **71**(10-12), 1771–1787 (2008).

14 D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision* **60**, 91–110 (2004).

15 M. Brown and D. Lowe, "Invariant Features from Interest Point Groups," in *British Machine Vision Conference*, 656–665 (2002).

16 S. Mallat, *A Wavelet Tour of Signal Processing, Second Edition (Wavelet Analysis & Its Applications)*, Academic Press, 2 ed. (1999).

17 H. Bay, T. Tuytelaars, and L. Van Gool, "Speeded-Up Robust Features (SURF)," *Computer Vision and Image Understanding* **11**(3), 346–359 (2008).

18 R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd ed. (2001).

19 H. Hoffmann, "Kernel PCA for Novelty Detection," *Pattern Recognition* **40**, 863–874 (2007).

20 V. Sulic, J. Perš, M. Kristan, and S. Kovacic, "Efficient Dimensionality Reduction Using Random Projection," in *15th Computer Vision Winter Workshop*, 29–36, (Nové Hrady, Czech Republic) (2010).

21 S. Dasgupta and A. Gupta, "An Elementary Proof of the Johnson-Lindenstrauss Lemma," Tech. Rep. TR-99-006, Berkeley, CA (1999).

22 W. Johnson and J. Lindenstrauss, "Extensions of Lipschitz Mappings into a Hilbert Space," in *Conference in Modern Analysis and Probability*, *Contemporary Mathematics* **26**, 189–206, American Mathematical Society (1984).

23 E. Bingham and H. Mannila, "Random Projection in Dimensionality Reduction: Applications to Image and Text Data," in *Knowledge Discovery and Data Mining*, 245–250, ACM Press (2001).

24 J. Ye and S. Ji, *Discriminant Analysis for Dimensionality Reduction: An Overview of Recent Developments*, 1–19. John Wiley & Sons, Inc. (2009).

25 S. Balakrishnama and A. Ganapathiraju, "Linear Discriminant Analysis - A Brief Tutorial," in *International Symposium on Information Processing*, (Mississippi State, MS, USA) (1998).
26 C. H. Park and H. Park, "A Comparison of Generalized Linear Discriminant Analysis Algorithms," *Pattern Recognition* **41**(3), 1083–1097 (2008).
27 S. Wold, A. Ruhe, H. Wold, and W. J. Dunn, "The Collinearity Problem in Linear Regression. The Partial Least Squares (PLS) Approach to Generalized Inverses," *SIAM Journal on Scientific and Statistical Computing* **5**(3), 735–743 (1984).
28 R. Rosipal and N. Krämer, "Overview and Recent Advances in Partial Least Squares," in *Subspace, Latent Structure and Feature Selection Techniques*, 34–51, Springer Berlin / Heidelberg (2006).
29 S. Maitra and J. Yan, "Principle Component Analysis and Partial Least Squares: Two Dimension Reduction Techniques for Regression," in *Applying Multivariate Statistical Models*, 79–90 (2008).
30 R. Noonan and H. Wold, "NIPALS Path Modelling with Latent Variables," *Scandinavian Journal of Educational Research* **21**(1), 33–61 (1977).
31 MIR Flickr, "MIR Flickr-1M," (2010). `http://press.liacs.nl/mirflickr/`.
32 INRIA, "INRIA Graffiti Dataset," (2011). `http://www.inrialpes.fr/movi`.
33 Y. Ke and R. Sukthankar, "PCA-SIFT: A More Distinctive Representation for Local Image Descriptors," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 506–513 (2004).
34 B. Leibe and B. Schiele, "The TU Darmstadt Database (formerly the ETHZ Database)," (2010). `http://pascallin.ecs.soton.ac.uk/challenges/VOC/databases.html#TUD`.
35 S. Agarwal and D. Roth, "Learning a Sparse Representation for Object Detection," in *7th European Conference on Computer Vision*, 113–130, (Copenhagen, Denmark) (2002).
36 L. Juan and O. Gwun, "A Comparison of SIFT , PCA-SIFT and SURF," *International Journal of Image Processing* **3**(4), 143–152 (2009).

Author Biographies

Ricardo Eugenio González Valenzuela received his B.Sc. degree in Computer Science from the Catholic University of Santa Maria, Arequipa, Peru, and his M.Sc. degree from the Institute of Computing at the University of Campinas, Brazil. His research interests include image processing, computer vision and machine learning.

William Robson Schwartz received his Ph.D. degree in Computer Science from University of Maryland, College Park, USA. He received his B.Sc. and M.Sc. degrees in Computer Science from the Federal University of Parana, Curitiba, Brazil. He is currently a professor in the Department of Computer Science at the Federal University of Minas Gerais, Brazil. His research interests include computer vision, biometrics and image processing.

Helio Pedrini received his Ph.D. degree in Electrical and Computer Engineering from Rensselaer Polytechnic Institute, Troy, NY, USA. He received his M.Sc. degree in Electrical Engineering and his B.Sc. in Computer Science, both from the University of Campinas, Brazil. He is currently a professor in the Institute of Computing at the University of Campinas, Brazil. His research interests include image processing, computer vision, pattern recognition, computer graphics, computational geometry.