The Quantitative Characterization of the Distinctiveness and Robustness of Local Image Descriptors

Gustavo Carneiro *

Integrated Data Systems Department

Siemens Corporate Research

755 College Road East

Princeton, NJ, USA

Allan D. Jepson

Department of Computer Science,

University of Toronto

D.L. Pratt Building, Rm 283C

6 King's College Rd.

Toronto, ON, Canada

Abstract

We introduce a new method that characterizes quantitatively local image descriptors in terms of their distinctiveness and robustness to geometric transformations and brightness deformations. The quantitative characterization of these properties is important for recognition systems based on local descriptors because it allows for the implementation of a classifier that selects descriptors based on their distinctiveness and robustness properties. This classification results in: a) recognition time reduction due to a smaller number of de-

scriptors present in the test image and in the database of model descriptors; b) improvement of the recognition accuracy since only the most reliable descriptors for the recognition task are kept in the model and test images; and c) better scalability given the smaller number of descriptors per model. Moreover, the quantitative characterization of distinctiveness and robustness of local descriptors provides a more accurate formulation of the recognition process, which has the potential to improve the recognition accuracy. We show how to train a multi-layer perceptron that quickly classifies robust and distinctive local image descriptors. A regressor is also trained to provide quantitative models for each descriptor. Experimental results show that the use of these trained models not only improves the performance of our recognition system, but it also reduces significantly the computation time for the recognition process. [1]

Key words:
visual object recognition, local image descriptors, discriminant classifier, regression models

1 Introduction

In the last few years, there has been a growing interest in recognition systems using a collection of local image descriptors for the tasks of object recognition [22], image matching [31], object discovery and recognition [35], among others. The model representation used in these systems is based on a collection of image de-

[1] This work was performed while Gustavo Carneiro was at the University of Toronto.

* Corresponding author.
Email addresses: carneiro@cs.toronto.edu (Gustavo Carneiro), jepson@cs.utoronto.ca (Allan D. Jepson).
URLs: http:/www.cs.ubc.ca/~carneiro (Gustavo Carneiro), http:/www.cs.toronto.edu/~jepson (Allan D. Jepson).

scriptors with small spatial support extracted from salient image regions, such as corners [18], difference of Gaussians [22], etc. When compared to image representations based on a large spatial support (i.e., global image feature) [25], local representations achieve a better robustness to clutter, partial occlusion, and common image deformations.

Current state-of-the-art local image descriptors have been carefully designed to be robust to geometric transformations and photometric deformations and also to be distinctive [23]. However, individual local descriptors have, in general, different discriminating and robustness properties, even though they are extracted using the same algorithm. This happens because some local descriptors are detected from regions with different stability properties with respect to image deformations, and also because some descriptors lie in regions of the feature space more or less densely populated. Therefore, an explicit quantitative characterization of the distinctiveness and robustness of local descriptors is important in order to: 1) provide a classification scheme that selects descriptors with superior discriminating and robustness properties, and 2) allow for a more accurate formulation of the recognition process. The descriptor selection decreases the size of the model database by keeping only the most useful model descriptors for the recognition task, which results in a faster and more accurate recognition process and in a more scalable system (i.e., the system is able to deal with a higher number of visual classes). Finally, the more accurate formulation of the recognition process can improve the recognition accuracy.

In the literature the characterization of local image descriptors for classification and for estimating their relative importance during a recognition process has usually been treated separately by several authors.

The use of distinctiveness in order to estimate the relative importance of the model descriptors has been exploited by Amit and Geman [2]. In this work, the authors estimate the distribution of the descriptor similarities with respect to background descriptors, thus estimating the distinctiveness of the descriptor. This characterization is used for selecting local descriptors better suited for the recognition process, but note that the authors do not propose a classification scheme, nor do they use the local descriptor robustness. The use of robustness for estimating the relative importance of model local descriptors was the focus of various works [14,29,34], where the authors use an exponential distribution to approximate the robustness distribution. Additionally, other works try to estimate the detectability and discriminating power of a descriptor by calculating how often it appears in the learning stage [27,29].

Methods to classify local image descriptors without quantitatively characterizing their robustness and distinctiveness properties have been intensively studied lately [1,14,12,19,28,37,40]. Note that these approaches are useful for the selection process, but the absence of a quantitative characterization does not allow these methods for estimating the relative importance of local descriptors. Specifically, Ohba and Ikeuchi [28] select robust descriptors by verifying how their feature values vary with deformations, and unique descriptors are filtered by checking their distinctiveness when compared to other training image descriptors (i.e., two descriptors are discarded as ambiguous if they lie too close to each other in the feature space). Alternatively, Dorko and Schmid [12] proposed an approach that selects descriptors based exclusively on their discriminating power. Zhang also worked on a descriptor selection method using not only the discriminating, but also their robustness properties. In other related methods [1,14,37], a clustering algorithm selects the descriptors that appear more often during the training stage. However, none

of the methods above estimates quantitatively the robustness and distinctiveness distributions in order to properly classify each descriptor, as we propose here. In robotics, there has been some interest in the problem of selecting local descriptors for reducing the complexity of the simultaneous localization and mapping (SLAM) approaches. However, the proposed methods generally involve a way of selecting local descriptors without explicitly characterizing their distinctiveness and robustness properties, as we propose in this paper. For example, Sala et al. [30] propose a descriptor selection method for the problem of vision based navigation of a robot in a small environment. Their approach, based on graph theory, involves the partition of the environment into a minimal set of maximally sized regions, such that for all positions of a given region, the same set of k descriptors is visible.

In pattern recognition theory, there has been numerous methods proposed for the problem of feature selection and extraction[17]. Generally, the feature selection and extraction problems consist of building a lower dimensional feature space from the original one, where tasks such as classification or regression are performed more accurately and/or efficiently. The goal of our paper is that of descriptor selection (and characterization). Therefore, the feature space of each descriptor remains intact throughout the algorithm, but the set of descriptors representing an image will be reduced to include only the most robust and distinctive ones. Even though the problem being presented by this paper is on descriptor selection and characterization, traditional methods of feature selection (and extraction) could be adapted. The main idea to permit such adaptation is to build a feature space using the model descriptors. The issue involved in such approach is that the dimensionality of the feature space can grow indefinitely high (note that each new descriptor would define a new dimension in this feature space), and traditional techniques for feature selection and extraction (e.g., principal components analysis, manifold learning, linear

discriminant analysis) are unlikely to work in these very high dimensional spaces. A practical example on how to make this approach work is the bag of features [9], where a feature space is built based on the clusters formed by the distribution of local descriptors. This means that the new feature space has a number of dimensions equal to the number of clusters, and the feature values are determined by the number of votes cast to each cluster. This way, the feature dimensionality has a fixed value, and consequently the traditional techniques mentioned above can work for the feature selection/extraction problems. Nevertheless, the approaches in the literature following such idea focus more on the recognition task than on the feature selection process (e.g., how to build a classifier capable of working in such high dimensional space and how to cluster the features in order to help the classification task). A recent trend in the computer vision community is to build descriptor selection methods for specific recognition tasks, such as the face and facial features detector by Ding and Martinez [11]. This method works based on a sequence of several classifiers, each trained to detect a specific facial feature (note that each facial feature is manually determined). This approach differs from ours since there is no explicit characterization of the descriptors and the design of the method is quite specific for the problem at hand.

There has been studies similar to ours for specific goals in robotics, which makes a direct comparison hard to implement. For example, He et al. [19] characterize explicitly the distinctiveness and robustness of local descriptors in order to provide a classification scheme to filter out descriptors that will not be effective for a recognition process. In particular, the authors study the problem of vision based environment localization using single images (as opposed to works on SLAM [10,33] that generally use pairs of images). Their system uses a temporal sequence of training images to learn a manifold with the property that nearby images in the environment

are also close together in the manifold. Using this constraint, the authors propose an incremental learning framework that selects robust and distinctive descriptors for representing images. Notice that although the goal of He et al. [19] is similar to ours, they formulate the problem specifically to solve the environment localization task. The method we propose here is more generic because it is designed for the problem of visual object recognition.

1.1 Contributions

This paper introduces a novel way of characterizing quantitatively the distinctiveness and robustness of local image descriptors [8]. In a visual object recognition framework, this characterization is used for: 1) selecting the most appropriate descriptors based on their robustness and distinctiveness properties; and 2) formulating more accurately the recognition process. We further show that it is possible to train a multi-layer perceptron (MLP) classifier for fast descriptor selection. We also train an MLP regressor for quick quantification of the distinctiveness and robustness properties of the descriptors. The proposed quantitative characterization and training of the MLP classifier and regressor are quite generalizable in the sense that the same basic approach can be applied to several different types of local image descriptors. We show this by applying the whole process of local descriptor characterization and MLP training to the following two different types of local descriptors: local phase [5] and SIFT [22] descriptors. We also use the classification and regression procedures as a pre-processing step for our recognition system [7]. Empirical results using this system show that this pre-processing stage significantly decreases the time to process test images and also improves the recognition accuracy.

1.2 Paper Organization

This paper is organized as follows. Section 2 introduces the quantitative characterization of local image descriptors. The classification of descriptors based on robustness and distinctiveness is presented in Section 3. The discussion in Section 4 shows the main problems of the quantification and classification methods presented in Section 3, and solutions to these problems are presented in Sections 5 and 6. Experiments showing the advantages of using this quantification and classification approaches are demonstrated with a full-blown recognition system in Section 7, and Section 8 concludes the work.

2 Quantitative Characterization of Local Image Descriptors

This section introduces a method to quantitatively characterize the distinctiveness and robustness properties of local image descriptors. The main purpose of this quantitative characterization is to classify useful descriptors and also to weight the importance of each descriptor for the recognition process.

2.1 Local Image Descriptor

Local image descriptors are photometric features extracted from image regions with limited spatial support. There is not a precise definition in the literature about the actual size of this spatial support, but the assumption is that the size of a local image descriptor can be between one pixel and 32 pixels in a typical image of size around 500 x 500 pixels. These features are generally extracted from image regions presenting two basic properties known to be useful for recognition and

matching processes. The first property is robustness to image deformations, such as rotation, scale, translation, and brightness variations. The second property is a high degree of information content that helps discriminate these regions. The algorithms that automatically select such regions are generally known as interest point detectors [18,22]. From these regions, image features are extracted such that they possess similar properties (i.e., robustness and uniqueness). In this paper, we define a local image descriptor as the following feature vector:

$$\mathbf{f}_l = [\mathbf{x}_l, \mathbf{v}_l] \,, \tag{1}$$

where $\mathbf{x}_l \in \Re^2$ is the image position of the descriptor $\mathbf{f}_l$, and $\mathbf{v}_l \in \Re^V$ is the descriptor vector with V photometric values. Section 6 shows two examples of local feature photometric values. The database of model descriptors extracted from a model image I_m is then denoted as $\mathcal{O}_m = \{\mathbf{f}_l | \mathbf{x}_l \in \mathcal{I}_m\}$, where $\mathcal{I}_m$ is defined as the set of interest point locations $\mathbf{x}_l$ (1) of each local descriptor $\mathbf{f}_l$ extracted from image I_m. Finally, the similarity between two descriptors $\mathbf{f}_l$ and $\mathbf{f}_o$ is computed by the function $s_f(\mathbf{f}_l, \mathbf{f}_o) \in [0, 1]$ ($s_f(.) \approx 1$ means high similarity).

2.2 Quantitative Characterization of Distributions

As mentioned before, local image descriptors must be distinctive and stable to image deformations to be useful for several computer vision applications. Although local descriptors are designed to be distinctive and robust to image deformations, each individual descriptor has different degrees of these properties. In this section, we explain our method to estimate the following three statistics of each local descriptor: a) distribution of robustness to image deformations, b) distributions of distinctiveness, and c) probability of detection. Using these three statistics, we implement a classification process that keeps only the most appropriate descriptors

for visual recognition tasks.

Our method of estimating the distinctiveness and robustness distributions of local descriptors is inspired by Yuille's approach [39], which uses the probability distributions P_{on} and P_{off} corresponding to the true positive and false positive distributions, respectively, for the problem of road tracking. We describe the probability distribution for robustness $P_{\text{on}}(s_f(\mathbf{f}_l, \mathbf{f}_o); \mathbf{f}_l)$, i.e., the probability of observing descriptor similarity $s_f(\mathbf{f}_l, \mathbf{f}_o) \in [0, 1]$ given that the descriptor $\mathbf{f}_o$ is a true match for the descriptor $\mathbf{f}_l$. The robustness of a local descriptor $\mathbf{f}_l$ also depends on the probability that the interest point detector will fire at its relative position $\mathbf{x}_l$. We define this probability as $P_{\text{det}}(\mathbf{x}_l)$, which is the probability that an interest point is detected in the test image near the location corresponding to $\mathbf{x}_l$ of descriptor $\mathbf{f}_l$. The distinctiveness $P_{\text{off}}(s_f(\mathbf{f}_l, \mathbf{f}_o); \mathbf{f}_l)$ is the probability of observing $s_f(\mathbf{f}_l, \mathbf{f}_o)$ given that the descriptor $\mathbf{f}_o$ is a false match for the descriptor $\mathbf{f}_l$.

The main goal of this section is to present a simple way of characterizing the distributions P_{on}, P_{off}, and P_{det} involving a small number of parameters. It is important to have a representation with a small number of parameters since the visual models we consider in this work generally consist of thousands of descriptors, so the complexity of the representation can increase significantly with the number of parameters for P_{on}, P_{off}, and P_{det}. The basic idea of the whole process is depicted in Figure 1. Step 1 comprises the following tasks: 1) select a model image containing the visual object of interest; 2) apply several synthetic image deformations to this model image; and 3) build a database of local descriptors extracted from a database of images that does not contain the model image (this forms the database of random descriptors). Step 2 consists of: 1) matching each local descriptor from the model image to the correct position at each deformed image; 2) from this matching process, it is possible to build a histogram of similarity distribution for each model

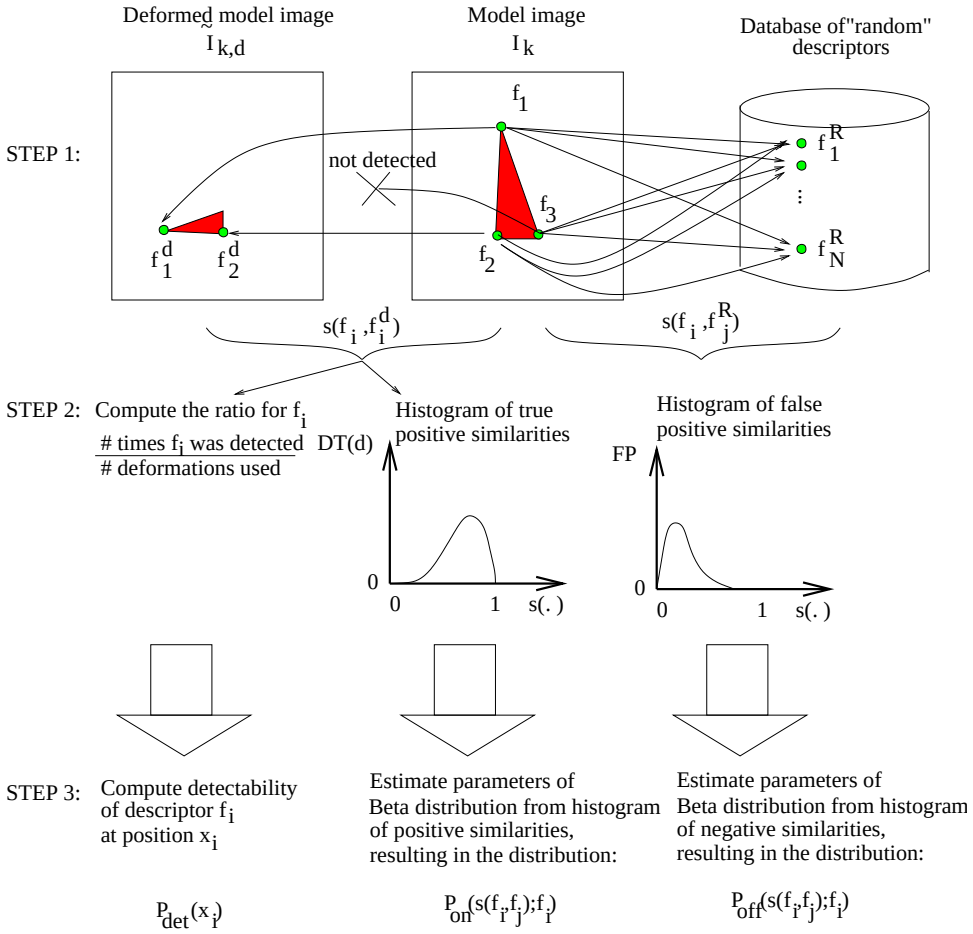


Fig. 1. General view of the method to estimate of the detectability, robustness, and distinctiveness of local image descriptors.

descriptor and also to determine its ratio of detection (the ratio of detection of each model descriptor is represented by the percentage that the descriptor is detected at the deformed model images); and 3) matching each local descriptor from the model image to each descriptor in the database of random descriptors and building a histogram of false positive matches. Finally, in step 3, it is possible to quantitatively characterize the detectability, robustness, and distinctiveness for each model descriptor. We first describe how to automatically learn these parametric models, and then we define which model we use and how to estimate its parameters.

To train the models, we make use of a training set consisting of a fixed set of

foreground and background images (see Appendix B), along with synthetic image deformations (see Appendix A). Note that the use of synthetic image deformations has become relatively popular lately in order to increase the robustness of classifiers [3,21]. However these works usually do not address the same issues of our paper. We propose a method that not only improves the robustness of local descriptors, but also that selects and quantitatively characterizes the descriptors for improving the accuracy of the probabilistic detection. The set of foreground images $\mathcal{T}$ has 30 images, and the set of background images $\mathcal{R}$ contains 240 images, where $\mathcal{T} \cap \mathcal{R} = \emptyset$. As shown in Appendix B, the sets of foreground and background images are taken from the same pool of images, which contain pictures of landscape, people, animals, and texture. There is no conceptual difference between the two sets of images. This implementation with foreground and background images taken from the same pool of images has the potential to improve the generalization capabilities of the learned models. Given an image $I_k \in \mathcal{T}$, the set of local descriptors extracted from this image is represented by $\mathcal{O}_k = \{\mathbf{f}_l\}_{l=1,\ldots,N}$, and the set of interest points detected in the image I_k is denoted as $\mathcal{I}_k = \{\mathbf{x}_l\}_{l=1,\ldots,N}$, where each $\mathbf{x}_l \in \mathcal{I}_k$ is the respective position of the descriptor $\mathbf{f}_l \in \mathcal{O}_k$. Typically, the number of local features per image varies between 1,000 to 10,000. Consequently, the total number of features in the foreground set is between 30,000 and 300,000, depending on the type of local feature used (for details on the specific number of descriptor per feature type, please see Section 6). Moreover, the set of descriptors extracted from the background images is represented by $\mathcal{O}(\mathcal{R})$, which has between $100,000$ and $1,000,000$ descriptors, depending on the type of local feature (Section 6). The $P_{\text{off}}(s_f(\mathbf{f}_l, .)), \mathbf{f}_l)$ of each descriptor $\mathbf{f}_l \in \mathcal{O}_k$ is computed from the histogram of false positive matches

$$\{s_f(\mathbf{f}_l, \mathbf{f}_o) | \mathbf{f}_o \in \mathcal{O}(\mathcal{R})\}. \tag{2}$$

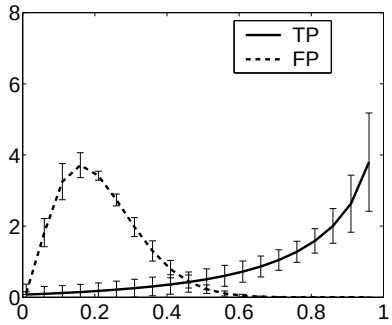


Fig. 2. Mean and standard deviation of the distribution of the phase similarity values between true positive (TP) and false positive (FP) matches for the phase feature [6].

On the other hand, $P_{\text{on}}(s_f(\mathbf{f}_l, .), \mathbf{f}_l)$ is computed from the histogram of descriptor similarities with respect to an image deformation $d \in \mathcal{DF}$, where $\mathcal{DF}$ is a set of synthetic image deformations (see Appendix A). Assuming that $\mathbf{x}_l$ is the position of the descriptor $\mathbf{f}_l \in \mathcal{O}_k$, and that the synthetic deformation $d \in \mathcal{DF}$ applied to I_k forms the image $\tilde{I}_{k,d}$, where points in I_k are mapped to points in $\tilde{I}_{k,d}$ as follows: $\tilde{\mathbf{x}}_{l,d} = \mathbf{M}(d)\mathbf{x}_l + \mathbf{b}(d)$, where $\mathbf{M}(d)$ and $\mathbf{b}(d)$ represent the spatial warp for the deformation d. Since we depend on the interest point detector to fire sufficiently close to that position, we search the corresponding descriptor on the deformed image as:

$$\tilde{\mathbf{f}}_{l,d} = \arg\max_{\mathbf{f}_o}\{s_f(\mathbf{f}_l, \mathbf{f}_o) | \mathbf{f}_l \in \mathcal{O}_k, \mathbf{f}_o \in \mathcal{O}(\tilde{I}_{k,d}), \|\mathbf{M}(d)\mathbf{x}_l + \mathbf{b}(d) - \mathbf{x}_o\| < \epsilon\}, \quad (3)$$

where ϵ is fixed at 2.0 pixels (as measured in the image $\tilde{I}_{k,d}$, which is down-sampled according to scale). It is important to mention that the local descriptors considered in this work are extracted with bandpass filters with peak frequency response at $\omega_d = 2\pi/(4.36\sigma_d)$, corresponding to a wavelength of $\lambda_d = 4.36\sigma_d$, where σ denotes the standard deviation of the filters. Also, test images are processed at $\lambda_d = 8$, which makes $\sigma_d \approx 2.0$ pixels (empirically, the use of $\lambda_d = 8$ achieves a good signal-to-noise-ratio). Thus, the uncertainty in terms of the local image descriptor position is around 2.0 pixels, hence $\epsilon = 2.0$.

Figure 2 shows the mean and standard deviation of the histogram of false positive

(2) and true positive matches (3) for the phase feature [6] using the sets $\mathcal{T}$ and $\mathcal{R}$ described above, where the descriptor similarity $s_f(.)$ is the phase correlation. Notice that the true positive (TP) and false positive (FP) histograms present a unimodal structure with a heavy tail, which resembles a beta distribution (see Fig. 4). Quite similar TP and FP distributions are also shown by Lowe [22]. Hence, we approximate the distributions P_{on} and P_{off} with the *beta* parametric distribution, which is defined as follows:

$$P_\beta(x;a,b) = \begin{cases} \frac{1}{\int_0^1 t^{a-1}(1-t)^{b-1}dt} x^{a-1}(1-x)^{b-1}, & \text{if } x \in (0,1) \text{ and } a,b>0 \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

This distribution is defined within the range $[0,1]$ (i.e., the same range of $s_f(.)$). Notice that we need to store only two parameters for the beta distribution, which can be considered as a low complexity representation. In Fig. 3, we see the approximation of the histograms above with the beta distribution using the local phase descriptors [5,6].

The method of moments (MM) provides a good one-step estimate of the beta parameters a and b providing results very similar to maximum likelihood estimation [38]. It is based on the first and second moments, namely μ_β and σ_β^2, of the histograms for P_{off} and P_{on}. The parameters (a,b) of the fitted beta distribution are then

$$b = \frac{\mu_\beta(1-2\mu_\beta+\mu_\beta^2)}{\sigma_\beta^2} + \mu_\beta \text{ and } a = \frac{\mu_\beta b}{1-\mu_\beta}\,. \quad (5)$$

Finally, in order to determine P_{det} of a model descriptor position $\mathbf{x}_l \in \mathcal{I}(I_k)$, we have to investigate how stable this position is with respect to the deformations $d \in \mathcal{DF}$ (see Appendix A). Specifically, let $\mathcal{C}(\mathbf{x}_l)$ be the set of deformations for which a corresponding interest point can be found in the original image I_k, so $\mathcal{C}(\mathbf{x}_l) =$

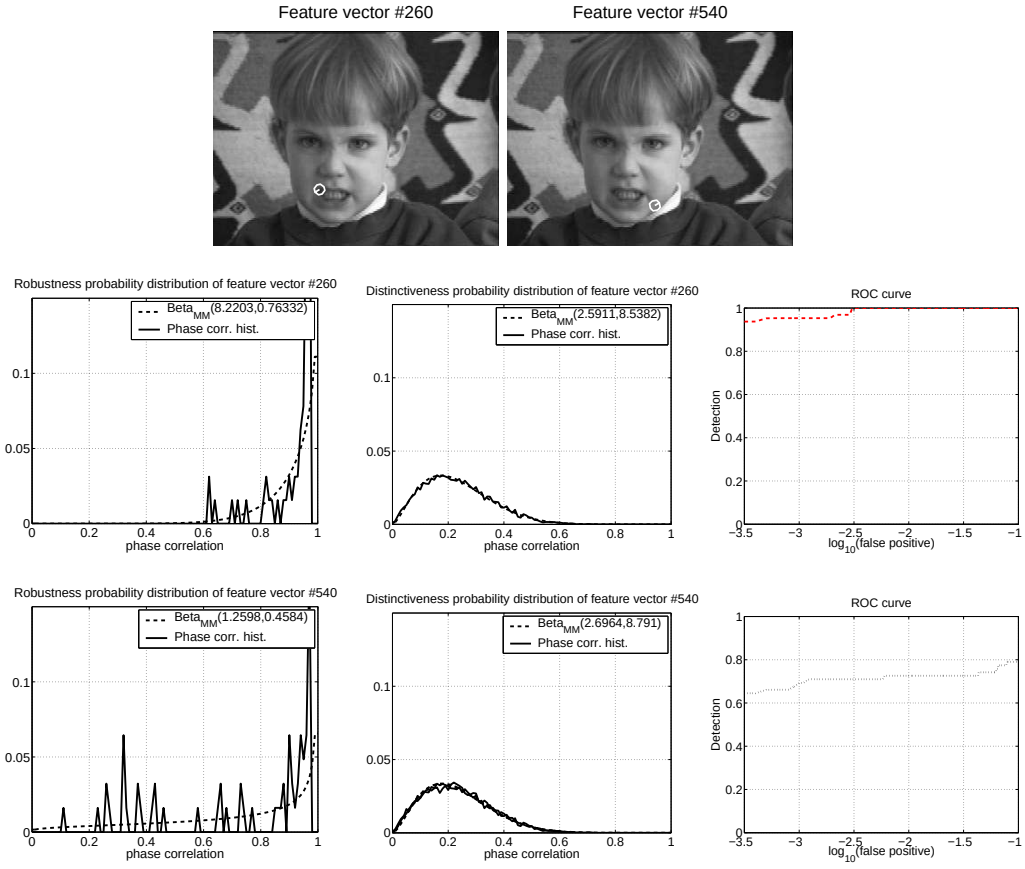
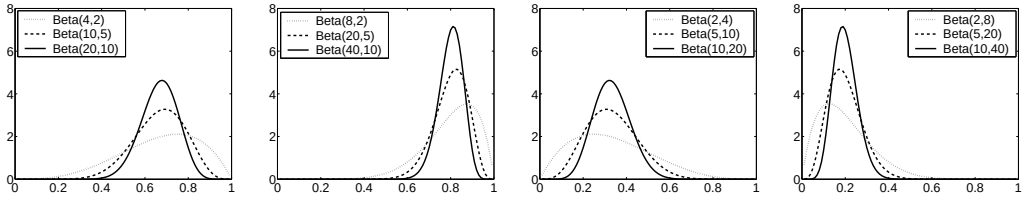


Fig. 3. Approximation of distinctiveness and robustness histograms using the beta distribution for the local phase descriptors [5,6] (the first row displays the local phase descriptors being studied, represented by the white circle on the image). Note that the descriptor in the first image is identified by number 260, and the second has number 540. The receiver operating characteristic (ROC) curves of robustness vs. distinctiveness for descriptors 260 (second row) and 540 (third row) are shown in the last column. The P_{det} of the descriptor 260 is 87%, and for descriptor 540 is 67%. The two numbers after the legend 'Beta$_{\text{MM}}$' are the estimated parameters a and b, respectively (see Eq. 5). Descriptor 540 is filtered out due to low robustness (see a and b parameters for robustness graph in first row) and low detectability, while descriptor 260 is kept for the model representation.

$\{d|\exists \mathbf{x}_j \in \mathcal{I}(\tilde{I}_{k,d})$ s.t. $\|\mathbf{x}_j - \mathbf{M}(d)\mathbf{x}_l - \mathbf{b}(d)\| < \epsilon\}$ with ϵ fixed at 2.0 pixels (as measured in the image $\tilde{I}_{k,d}$, which is down-sampled according to scale), and $\mathbf{M}(d)$ and $\mathbf{b}(d)$ represent the spatial warp for the deformation d. Hence the detectability



(a) (b) (c) (d)

Fig. 4. Examples of beta distributions with different values for the parameters a and b. In the legend, the first and second parameters of the function $Beta$ represent a and b, respectively.

probability is denoted by

$$P_{\text{det}}(\mathbf{x}_l) = \frac{|\mathcal{C}(\mathbf{x}_l)|}{|\mathcal{DF}|}. \tag{6}$$

3 Local Descriptor Classification

We use one key observation about the beta distribution in order to define our classification process, as depicted in Fig. 4. Notice that in general, as $a > b$, the mode of the distribution is close to one, and when $b > a$, the mode is closer to zero. Therefore, the ideal distribution for P_{on} should resemble the graphs (a) and (b) in Fig. 4, where $a > b$ because it is desirable that the similarity values for correct matches are as close as possible to one, which means that the descriptor values are relatively insensitive to image deformations. On the other hand, the ideal distribution P_{off} of a model descriptor should be similar to the graphs (c) and (d), where $b > a$ since we want that model descriptors and wrong matches have low similarity values.

Therefore, our classification procedure consists of checking the following properties: a) high robustness $a_{\text{on}}(\mathbf{f}) > \tau_{\text{on}} b_{\text{on}}(\mathbf{f})$ (i.e., the mode of the P_{on} distribution gets closer to one); b) high distinctiveness $b_{\text{off}}(\mathbf{f}) > \tau_{\text{off}} a_{\text{off}}(\mathbf{f})$ (i.e., the mode of the P_{off} distribution gets closer to zero); and c) high detectability $P_{\text{det}}(\mathbf{x}) > p\%$. As a result, we obtain a subset of descriptors $\mathcal{O}_k^* \subseteq \mathcal{O}_k$ that have the three properties

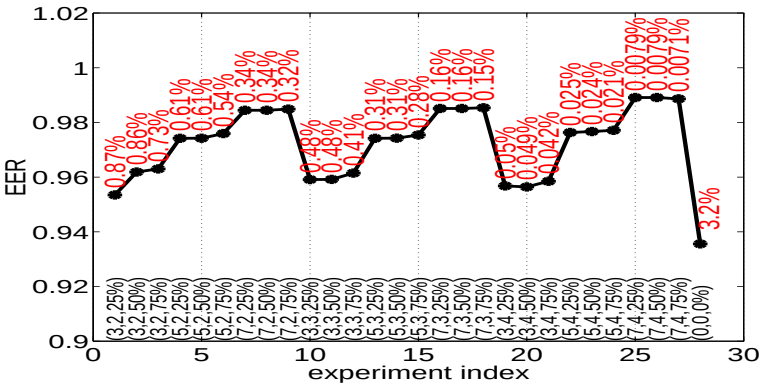


Fig. 5. Experiment showing the EER (vertical axis) and percentage of interest points with respect to the image size (this percentage is denoted by the number over each marker). The horizontal axis display the specific parameter values used in each of the 28 experiments as follows: $(\tau_{\text{on}}, \tau_{\text{off}}, p\%)$.

above. The values τ_{on}, τ_{off}, and p above are determined in order to have, on average, the percentage of interest points around 0.3% of total image size. This percentage is based on the study by Carneiro and Jepson [6] who noticed that the number of interest points is around 0.3% of total image size for the state-of-the-art methods developed by Lowe [22] and by Mikolajczyk and Schmid [24]. In Fig. 5, we show an experiment with varying values of the parameters above with respective equal error rate (EER) [2] and the percentage of interest points with respect to the image size. According to this graph, we set $\tau_{\text{on}} = 7$, $\tau_{\text{off}} = 2$, and $p\% = 75\%$ because these values produced a percentage of interest points around 0.3% compared to the image size and also because the EER is relatively high (compared to other parameter values).

Fig. 3 illustrates examples of selected and rejected local phase descriptors, where $\tau_{\text{on}} = 7$, $\tau_{\text{off}} = 2$, and $p\% = 75\%$. Also, Fig. 6 shows the significant improvement of the ROC curve and the reduction of the number of descriptors from 3.2% to

[2] The EER is the point at which the true positive rate equals one minus the false positive rate.

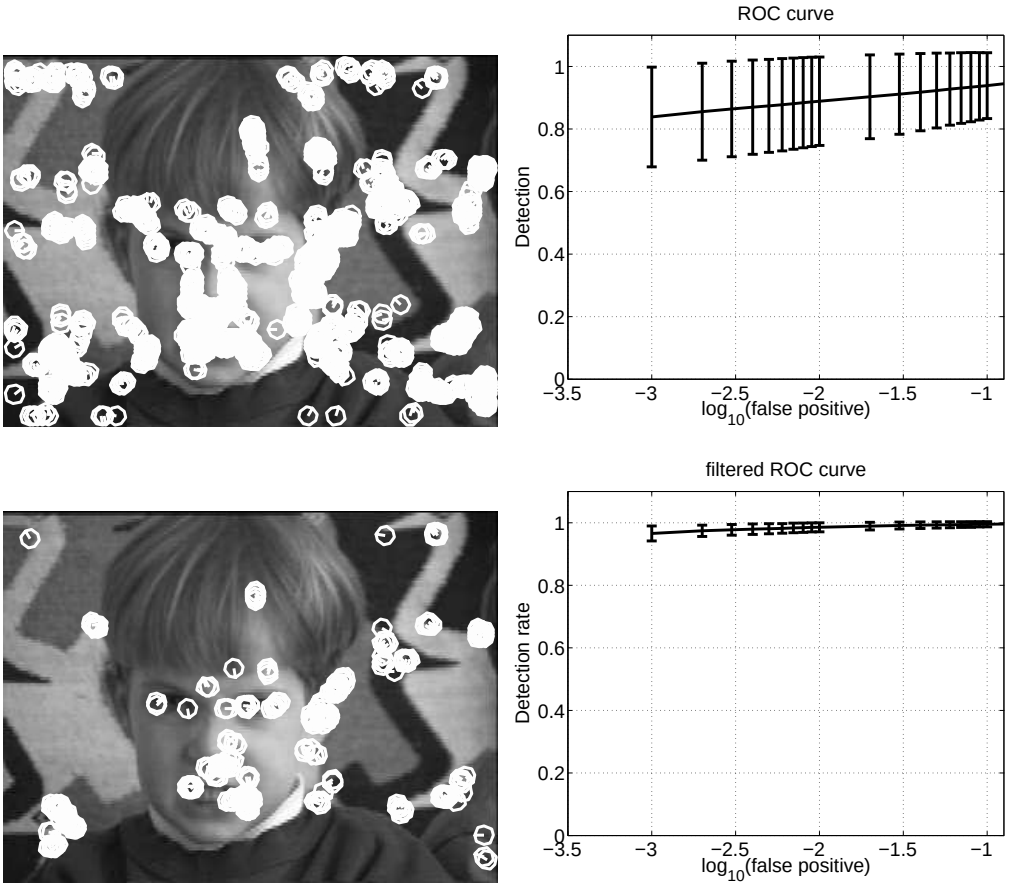


Fig. 6. ROC curve computed from the descriptors (white circles) in each figure above. The graph in the first row, second column shows the mean and standard deviation graph of the ROC curves computed from all the local descriptors at wavelength $\lambda = 8$ from the image shown on the top-left corner. The graph in the second row, second column shows the ROC curves with the points filtered by the procedure described in Sec. 3. Notice the significant improvement in terms of robustness vs. distinctiveness, and also the reduction of the number of descriptors detected.

0.3% of total image size when the classification procedure above for local phase descriptors is applied on all the descriptors of the image.

4 Discussion

There are two problems with the method described above for computing the descriptor robustness and distinctiveness, namely: 1) there is no guarantee that those distributions learned in artificially deformed images can be extended to real deformations; and 2) the time needed to learn those distributions is quite large.

The first problem is addressed in Sec. 5 through empirical experiments, where we show that the parameters learned in the artificially deformed models are indeed applicable to real image perturbations. Further quantitative analysis given controlled image deformations would also be worthwhile although this is beyond the scope of this work.

The second problem is solved in Sec. 6 by training two multi-layer perceptron models [26] using a supervised learning scheme. The first multi-layer perceptron classifies descriptors according to the properties above (i.e., robustness and distinctiveness), and the second estimates through non-linear regression the parameter values for each descriptor selected by the classifier. Both multi-layer perceptron models are trained using the filter responses of the local descriptor as the input. The distribution parameters provide the target output for the regression problem, and the classification results provide the target output for the classification task.

5 Comparison Between Real and Artificial Deformations

The main reason why artificial image deformations are used for learning the descriptor probability distributions is to allow for a complete control over the corresponding descriptor positions in the deformed images. Ideally, this learning pro-

cedure should be done on real image deformations that would produce a better estimation of those distributions. However, that would require a knowledge of the descriptor positions of the model in the images containing the deformed model. The question to be answered here is whether the densities learned over the sequence of artificially deformed images are applicable to actual deformations of the model image.

Our quantitative evaluation of local descriptor performance consists of the following steps:

- Take a sequence of N images $\{I_i\}_{i\in\{1,\ldots,N\}}$ containing the model to be studied under real image deformations. Effectively, a model is a region present in all those images (e.g., a person's face).
- Extract the local descriptors from the model image I_1 to form the set $\mathcal{O}_1$. Learn the probability distributions (i.e., P_{on}, P_{off}, and P_{det}) of each descriptor present in $\mathcal{O}_1$ using the scheme described in Section 2.
- Extract the local descriptors of each subsequent test image, which produces $\mathcal{O}_i$ for $i > 1$.
- Find the correspondences between the set of model descriptors $\mathcal{O}_1$ and each set of test descriptors $\mathcal{O}_i$ for $i > 1$, separately, as follows:

$$\mathcal{N}_{1i} = \{(\mathbf{f}_l, \tilde{\mathbf{f}}_l) | \tilde{\mathbf{f}}_l \in \mathcal{O}_i, \mathbf{f}_l \in \mathcal{K}(\tilde{\mathbf{f}}_l, \mathcal{O}_1, \kappa_{\mathcal{N}}), s_f(\mathbf{f}_l, \tilde{\mathbf{f}}_l) > \tau_s\}, \tag{7}$$

where $s_f(.) \in [0, 1]$ represents the descriptor similarity function such that values close to one mean high similarity, $\tau_s = 0.75$, and $\mathcal{K}(.)$ is the set of the top $\kappa_{\mathcal{N}}$ correspondences with respect to $s_f(.)$ between test image descriptor $\tilde{\mathbf{f}}_l \in \mathcal{O}_i$ and the database of model descriptors $\mathcal{O}_1$. For this experiment, the value of $\kappa_{\mathcal{N}}$ is not very relevant, but setting it at two produces a good trade off between speed and robustness; that is, smaller values produces faster results and larger values results

in more robust but slower estimation. Either way, the final results presented here are not significantly affected. With these correspondences, use RANSAC [36] to estimate the affine transformation to align the model descriptors in $\mathcal{O}_1$ to the test image descriptors in $\mathcal{O}_i$. Note that the affine transform is computed using robust parameter estimation. This affine transform provides a rough approximation of the deformation that took place between these two images.

- Use the estimated affine transform to compute the approximate positions of the descriptors from I_1 to I_i, for $i > 1$, so that it is possible to compute the ROC curves for: 1) all model descriptors $\mathcal{O}_1$, 2) the filtered model descriptors $\mathcal{O}_1^*$, and 3) the set of rejected descriptors formed by $\mathcal{O}_1 - \mathcal{O}_1^*$.

Using the ROC curves computed with the artificial image deformations, it is possible to verify how well they approximate the ROC produced by the real image deformations $d \in \{\mathcal{DF}\}$ (see Appendix A). We show one instance of the experiment described above in Figures 7 and 8 using the local phase descriptor [6]. Notice that the ROC curves produced by the artificially deformed images are generally better than the ones yielded by the real deformations. This could have been caused by numerous processes, which include: the computed affine transform used to determine the approximate positions of the descriptors from I_1 to I_i is not sufficiently precise; or the set of artificial deformations $d \in \{\mathcal{DF}\}$ are not a reliable approximation of the real deformations. However, we see that the curves for the filtered set of descriptors is always comparable or better than the sets of all and rejected descriptors. This indicates that the learning process can be considered reliable since it can be generalized for small real deformations.

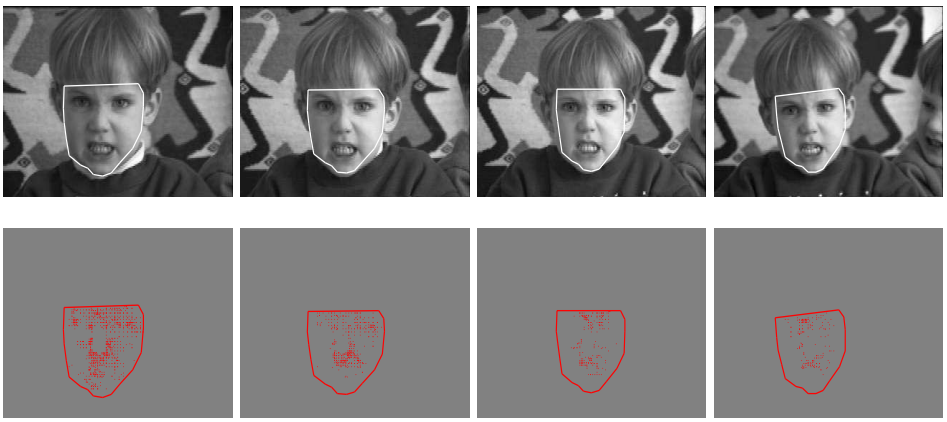


Fig. 7. Real image deformations approximated by an affine deformation. The first column of the first row shows the first image of the sequence containing the model 'kevin's face' (i.e., $\mathcal{O}_1$). The remaining images from the second to the fourth columns present the deformed model contour using the affine transform computed with the matches depicted on the second row as the red dots. Since the affine transform was computed using a robust parameter estimation, some matches can be left out of the contour if they were considered to be outliers. The whole sequence contains 30 images.

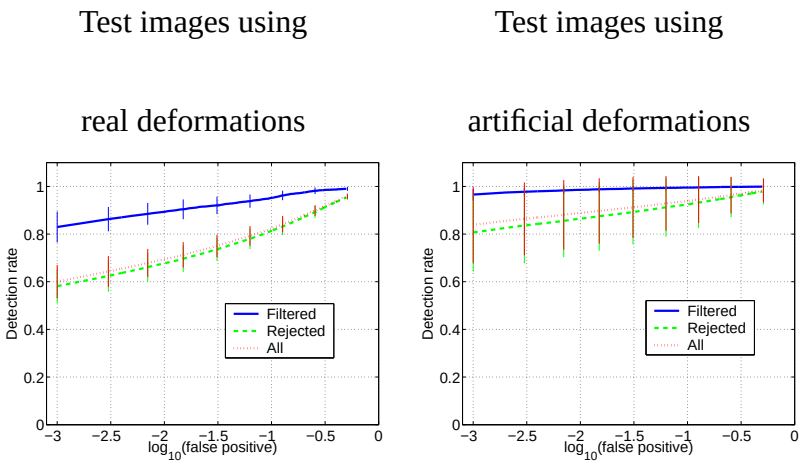


Fig. 8. Comparison between the ROC curves produced by real and artificial test image deformations for the case depicted in Fig. 7. The solid blue curve represents the detection performance for the filtered descriptors $\mathcal{O}_1^*$, while the dotted red curve is for the unfiltered descriptors $\mathcal{O}_1$, and the dashed green line is for the set of rejected descriptors $\mathcal{O}_1 - \mathcal{O}_1^*$.

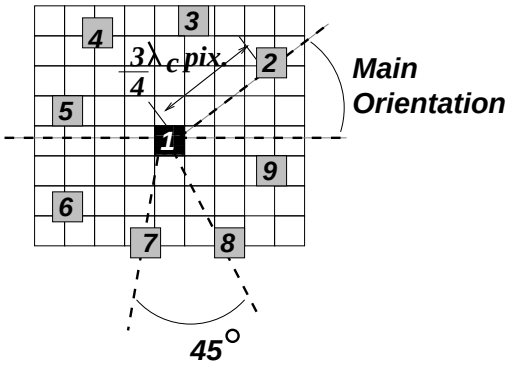


Fig. 9. Configuration of the phase-based local descriptor [5]. The center point represents the location selected by the interest point detector, and the nine points around it are the sampling points of the local descriptor.

6 Reducing the Time to Learn the Distributions

The learning procedure explained in Section 2 is computationally very intensive due to the requirement for explicitly deforming the image in order to estimate the performance statistics of each descriptor. On average, it can take between 20 and 30 hours to estimate the descriptor probabilities for a single model, which is clearly non-practical for the training and recognition tasks. Specifically, two tasks can be identified: a) a classification problem that categorizes a descriptor as part of the set of filtered descriptors $\mathcal{O}_k^*$; and b) a regression task to predict the parameters of P_{on}, P_{off}, and P_{det}. The important question is whether it is possible to do the classification/regression using the filter responses alone (i.e., without going through the whole learning procedure).

For the classification task we trained a multi-layer perceptron (we also refer to it as a neural network classifier) using Netlab [26], where the input layer received the following filter responses from the local phase descriptor $\mathbf{f}_l$ [6] extracted from a given location $\mathbf{x}_l$ [3]:

[3] The local phase descriptor is detected using the scale filtered Harris corner [6], and the

- the values at the sampling points (see Fig. 9) of the second derivative of a Gaussian (i.e., the G_2 filter) and its Hilbert transform (i.e., the H_2 filter) tuned to the orientations $0^o + \theta_l$, $45^o + \theta_l$, $90^o + \theta_l$, and $135^o + \theta_l$, where θ_l is the dominant orientation at descriptor position $\mathbf{x}_l$, and to the scales λ_c, $\lambda_c/\sqrt{2}$, and $\lambda_c\sqrt{2}$ [16](a total of 216 dimensions);
- I_x, I_y (i.e., horizontal and vertical image derivatives) within a 5x5 window around $\mathbf{x}_l$ (a total of 50 dimensions);
- eigenvalues μ_1, μ_2 used to compute the Harris cornerness function [18] and the following cornerness function value [5]:

$$t(\mathbf{x}_l) = \frac{\mu_2(\mathbf{x}_l)}{c + (1/2)(\mu_1(\mathbf{x}_l) + \mu_2(\mathbf{x}_l))}$$

  where c is a constant to avoid a division by zero (a total of 3 dimensions);
- deviation between the local wavelength of the descriptor and local frequency tuning of the G_2 and H_2 filters, denoted by $|\log(\lambda(\mathbf{x}_l, \lambda_c)) - log(\lambda_c)|$ at the scales λ_c, $\lambda_c/\sqrt{2}$, $\lambda_c\sqrt{2}$, where $\lambda(.)$ denotes the local frequency computed from position $\mathbf{x}_l$ [15], and λ_c represents the local frequency tuning of the filters (a total of 3 dimensions).

Thus, these filter responses form a 274 dimensional local descriptor $\mathbf{f}_l$. The neural network ideally produces logistic output of 0 if the descriptor should be filtered out, and 1 otherwise. Recall from Sec. 3 that a selected descriptor must present $a_{\text{on}}(\mathbf{f}) > \tau_{\text{on}} b_{\text{on}}(\mathbf{f})$, $b_{\text{off}}(\mathbf{f}) > \tau_{\text{off}} a_{\text{off}}(\mathbf{f})$, and $P_{\text{det}}(\mathbf{x}) > p\%$, where $\tau_{\text{on}} = 7$, $\tau_{\text{off}} = 2$, and $p\% = 75\%$. Therefore, the target function for each descriptor $\mathbf{f}_l$ in this supervised learning problem is 1 if $\mathbf{f}_l \in \mathcal{O}_i^*$, and 0 otherwise. The training algorithm is the standard error back propagation with weight decay, using scaled conjugate gradient for the optimization. Also, we used 300 units for the simple hidden layer.

similarity is computed by the phase correlation function [6]

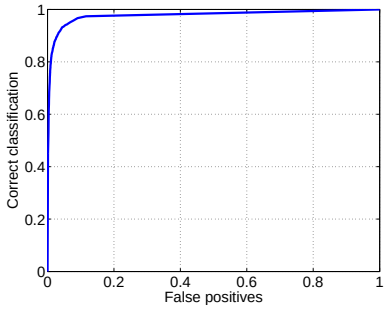


Fig. 10. ROC curve that shows the classifier performance on the test set.

The input for the regression problem is the same as the one for the classification problem, but the target values are the two parameters for the $P_{\text{on}}(s_f(\mathbf{f}_l, \mathbf{f}; \mathbf{f}_l)$ distribution, the two parameters for the $P_{\text{off}}(s_f(\mathbf{f}_l, \mathbf{f}; \mathbf{f}_l)$ distribution, and the $P_{\text{det}}(\mathbf{x}_l)$. As a result, we have five linear output units. Moreover, a descriptor $\mathbf{f}_l$ is part of the training set only if $\mathbf{f}_l \in \mathcal{O}_i^*$. We also used the Netlab package [26] for this problem.

In order to determine a sufficient number of training samples, we use the common rule of thumb that there has to be 5 to 10 times more training samples than model parameters [13]. Given that we have $274 \times 300 \times 1 = 67,400 = O(10^4)$ parameters, then we must have $O(10^5)$ training samples. Hence, we built a training set with 235,000 descriptors and a test set with 26,000 descriptors. Fig. 10 shows the ROC curve for the classification task computed using the test cases, and Fig. 11 shows the actual values of the P_{on} and P_{off} parameters, and P_{det} compared to the output of the regression network for the test cases.

In order to compare the performance provided by the classification procedure using the neural network above, we show the following experiment. We compare the descriptors in the set $\mathcal{O}_k^*$ produced by the standard learning procedure shown in Section 3 and the descriptors in $\tilde{\mathcal{O}}_k^*$ generated by the neural network classifier using a threshold 0.5 on the logistic classifier output. The threshold at 0.5 was estimated using a hold-out validation set such that the remaining percentage of descriptors

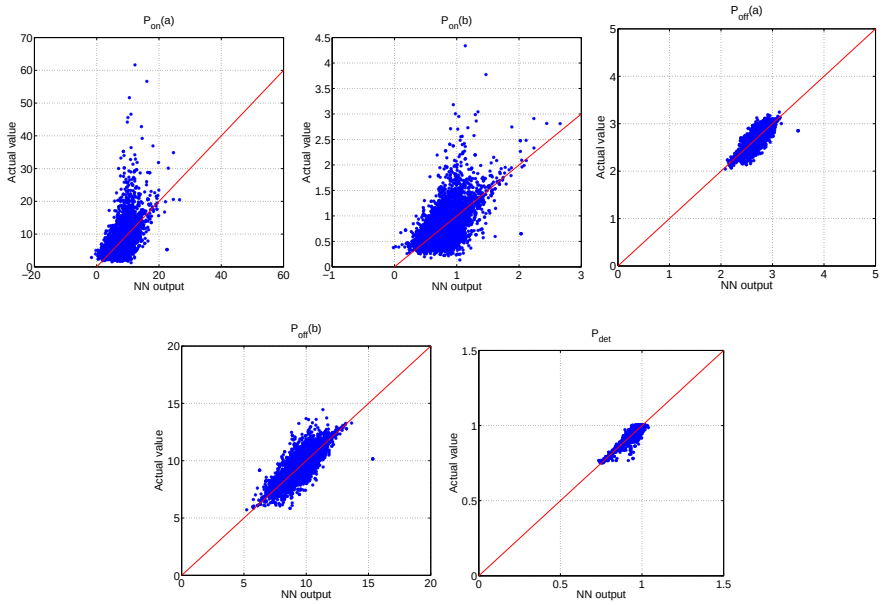


Fig. 11. Performance of the regression algorithm to predict the P_{on} and P_{off} parameters, and P_{det} value. The 45^o red line is used as a reference only.

was around 0.3% of the original image size (see Section 3). Fig. 12 presents this comparison, showing the mean and standard deviation produced by $\mathcal{O}_k^*$ on the center and $\tilde{\mathcal{O}}_k^*$ on the right for the respective test images in the leftmost column. Note that these two images were not used for training the neural network. The neural network classifier produces a result that is relatively similar to the original filtering method, and the relative number of descriptors is again reduced from 3.2% to 0.3% of the total number of image points. Notice that although there is a loss in terms of performance for the "Filtered" set when compared to the results produced by the original filtering method, it still produces results that are relatively better than both the "All" and "Rejected" sets. Moreover, the time for classifying the model local descriptors and to determine their P_{on}, P_{off}, and P_{det} parameter values is significantly reduced with the use of the neural networks described in this section. Specifically, the time needed to classify and to determine the P_{on}, P_{off}, and P_{det} using the direct simulation of deformations is between 20 and 30 hours, while the time spent in this same activity using the neural networks is around 5 seconds, as shown in Table 1.

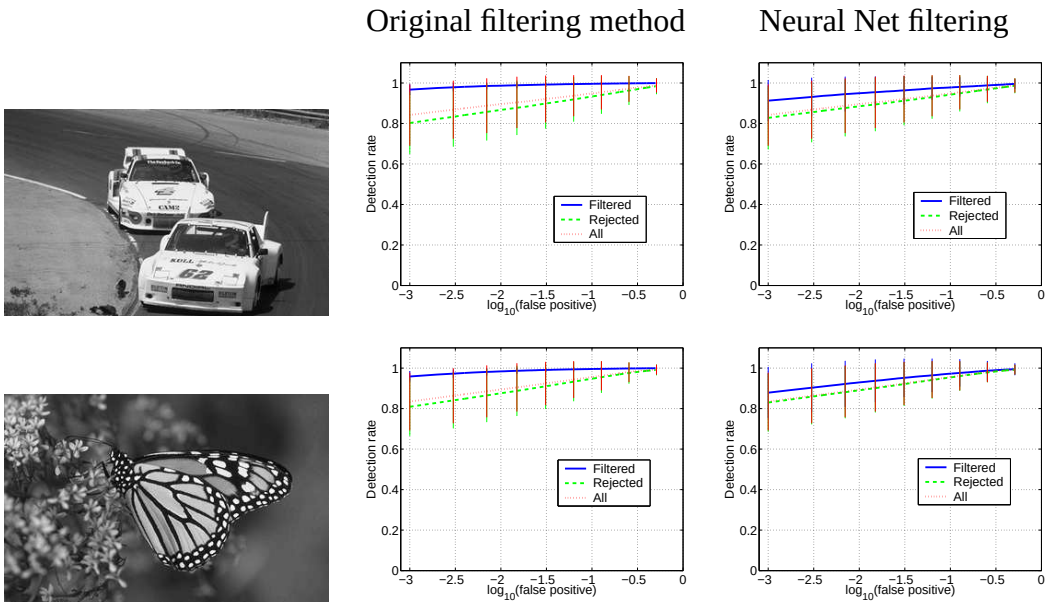


Fig. 12. Comparison between the ROC curves produced by the original classification procedure and the neural network for the images on the left, which were not used for training the neural net classifier.

Table 1

Average time taken for each procedure (i.e., direct method and neural networks) to learn the parameters of distributions P_{on}, P_{off}, and P_{det}.

	Direct Method	Neural Network
P_{on}, P_{off}, and P_{det} param. estimation	25 *hours*	5 *seconds*

Therefore, when adopting such strategy, one has to consider the trade off between time and performance.

6.1 Using the Multi-layer Perceptron with Other Local Descriptors

In order to show that the classifier and regressor can be used with different types of local descriptors, we also used the input of the SIFT descriptor [22] to train the same multi-layer perceptron. The main difference between the networks trained in Sec-

tion 6, and the networks below are the input data and the parameters to select robust and distinctive descriptors. For the SIFT descriptor, we use the 128-dimensional SIFT descriptor [22] basically consisting of the image gradient histograms computed at eight orientation planes around the neighborhood of the descriptor position $\mathbf{x}_l$ with the derivative filter tuned to scale λ [4].

The training set has 30,000 SIFT descriptors and the test set has 4,000 descriptors. The training procedure for SIFT descriptor differs from the one used for local phase descriptors only in the selection criteria for defining well behaved descriptors. More precisely, we use $\tau_{\text{on}} = 7$, $\tau_{\text{off}} = 0.5$, and $p\% = 50\%$. We observe that the percentage of descriptors that are kept in an image processed at scale $\lambda = 8$ is reduced from 0.3% to 0.12%. Fig. 13 shows the ROC curve produce by the classifier on a test set, and Fig. 14 shows the results for the regression problem with the actual values of the P_{on} and P_{off} parameters, and P_{det} compared to the output of the regression network for the test cases for the local phase descriptors. Notice that the results for SIFT in Figures 13-11 appear to be more accurate than the results for the local phase features in Figures 10-11. One possible reason for that is that SIFT can populate an effectively smaller dimensional feature space, and for this reason the parameters for the discriminative model studied in this section can be learned more easily. For instance, the work by Ke [20] showed that the SIFT descriptor can be reduced to around one sixth of its original dimensionality (i.e. 20 out of the original 128 dimensions) without affecting its performance in terms of discriminative properties, and actually improving the robustness properties of the descriptor. Though interesting, the study of the precise reason of this behavior is out of the scope of this work. Another interesting point raised by this experiment

[4] The SIFT descriptors are detected using the difference of Gaussians (DoG) interest point detector, and the similarity function is the Euclidean distance.

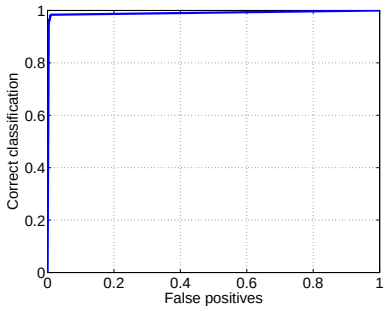


Fig. 13. ROC curve that shows the classifier performance on the test set using the SIFT descriptors.

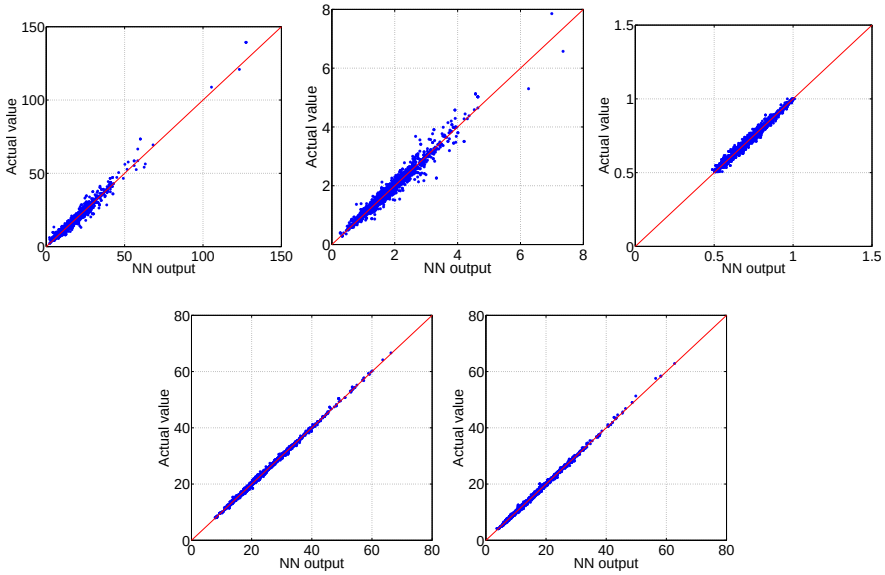


Fig. 14. Performance of the regression algorithm for the SIFT descriptors (see Fig. 11 for details).

is the fact that different types of local image descriptors generally present different trade-offs between robustness and distinctiveness. Therefore, a natural way of improving recognition results is then to combine different types of local descriptors. For instance, Carneiro and Lowe [4] combined local phase and SIFT descriptors, and developed powerful system capable of recognizing challenging visual object classes.

7 Experiments using a Recognition System

In this section we assess the performance of the recognition system described by Carneiro and Jepson [7] using the classification and regression networks proposed in Section 6 as a pre-processing step for the training and testing descriptors. Note that, originally, this system does not make use of a classifier or a regression net. We only ran the experiments using this recognition model with the local phase descriptors [5], where the training algorithm comprises the following steps:

- Extract the local descriptors from the model image I_M, which builds the set of model descriptors $\mathcal{O}_M$
- Select the well behaved descriptors using the classifier described in Sec. 6 (this forms the set $\mathcal{O}_M^* \subseteq \mathcal{O}_M$), estimate the parameters of the distinctiveness and robustness models using the regressor introduced in Sec. 6, and store the descriptors and respective model parameters in the model database. This results in the model database $M = \{[\mathbf{f}, a_{\text{on}}(\mathbf{f}), b_{\text{on}}(\mathbf{f}), a_{\text{off}}(\mathbf{f}), b_{\text{off}}(\mathbf{f}), P_{\text{det}}(\mathbf{x})] | \mathbf{f} \in \mathcal{O}_M^*\}$
- Learn the pairwise geometric relations of the selected descriptors [7], which forms the set $G_M = \{g(\mathbf{f}_l, \mathbf{f}_o) | \mathbf{f}_l, \mathbf{f}_o \in \mathcal{O}_M^*\}$, where $g(.)$ is a function that describes the geometric pairwise relations between $\mathbf{f}_l$ and $\mathbf{f}_o$.

The recognition algorithm consists of the following steps:

- Extract the local descriptors from the test image I, forming the set $\mathcal{O}$ (image processing step)
- Select the well behaved descriptors using the classifier described in Sec. 6, which builds the set $\mathcal{O}^* \subseteq \mathcal{O}$ (pre-processing step)

[5] Note that based on the results of Sec. 6.1, this classification and regression MLPs could also be used in the recognition model designed by Lowe [22].

- Form the correspondence set by finding the closest model descriptors to each test descriptor, building the set $\mathcal{N} = \{(\mathbf{f}_l, \tilde{\mathbf{f}}_l) | \mathbf{f}_l \in \mathcal{O}_M^*, \tilde{\mathbf{f}}_l \in \mathcal{O}^*, s_f(\mathbf{f}_l, \tilde{\mathbf{f}}_l) > \tau_s\}$, where $\tau_s = 0.75$ (database search step)
- Using pairwise geometric constraints, eliminate outliers from the correspondence set [7] (outlier rejection step)
- Build several independent hypotheses $\mathcal{E}_{h,\text{ for } h=1,\ldots,H}$, where H denotes the number of hypotheses and $\mathcal{E}_h = \{(\mathbf{f}_l, \tilde{\mathbf{f}}_l) | \forall \mathbf{f}_l \in \mathcal{O}_M^*, (\mathbf{f}_l, \tilde{\mathbf{f}}_l) \in \mathcal{N} \text{ or } \tilde{\mathbf{f}}_l = \emptyset\}$. Notice that each hypothesis $\mathcal{E}_h$ contains all the model descriptors from $\mathcal{O}_M^*$, which means that, for each model descriptor, either a match has been found (i.e., $(\mathbf{f}_l, \tilde{\mathbf{f}}_l) \in \mathcal{N}$) or no match is present in $\mathcal{N}$ (i.e., $\tilde{\mathbf{f}}_l = \emptyset$)
- Compute the probability of the model presence in each of the hypothesis as follows:

$$P(M|\mathcal{E}_i, T) = \frac{P(\mathcal{E}_h|T, M)P(T|M)P(M)}{P(\mathcal{E}_h|T, M)P(T|M)P(M) + P(\mathcal{E}_h|T, \neg M)P(T|\neg M)P(\neg M)}, \tag{8}$$

where $P(M)$ means our prior expectation that the model is present, and $P(\neg M) = 1 - P(M)$. Notice that $P(T|M)$ represents the global geometric configuration of local descriptors given M, which we treat to be similar to $P(T|\neg M)$ and cancel these terms from (8). The probabilistic formulation, based on [29], is as follows:

(1) $P(\mathcal{E}_h|T, M) \approx \prod_{(\mathbf{f}_l, \tilde{\mathbf{f}}_l) \in \mathcal{E}_h} P((\mathbf{f}_l, \tilde{\mathbf{f}}_l)|T, M)$, where we have the following two cases:

(a) $(\mathbf{f}_l, \emptyset) \in \mathcal{E}_h$:

$$P((\mathbf{f}_l, \emptyset) \in \mathcal{E}_h|T, M) \approx (1 - P_{\text{det}}(\mathbf{x}_l)) + P_{\text{det}}(\tilde{\mathbf{x}}_l)P_{\text{on}}(s < \tau_s; \mathbf{f}_l),$$

(b) $(\mathbf{f}_l, \tilde{\mathbf{f}}_l) \in \mathcal{E}_h$:

$$P((\mathbf{f}_l, \tilde{\mathbf{f}}_l) \in \mathcal{E}_h|T, M) \approx P_{\text{det}}(\mathbf{x}_l)P_{\text{on}}(s(\mathbf{f}_l, \tilde{\mathbf{f}}_l); \mathbf{f}_l)p(\mathbf{f}_l, \tilde{\mathbf{f}}_l),$$

where $p(\mathbf{f}_l, \tilde{\mathbf{f}}_o)$ denotes the probability that the geometric configuration of the model descriptor $\mathbf{f}_l$ matches the configuration of the test descriptor $\tilde{\mathbf{f}}_l$ [7].

(2) $P(\mathcal{E}_h|T, \neg M) = \prod_{(\mathbf{f}_l, \tilde{\mathbf{f}}_l) \in \mathcal{E}_h} P((\mathbf{f}_l, \tilde{\mathbf{f}}_l)|T, \neg M)$, where we have the following two cases:

(a) $(\mathbf{f}_l, \emptyset) \in \mathcal{E}_h$:

$$P((\mathbf{f}_l, \emptyset) \in \mathcal{E}_h|T, \neg M) \approx (1 - 0.003) + 0.003(1 - P_{\text{off}}(s(\mathbf{f}_l, \tilde{\mathbf{f}}_l) < \tau_s; \mathbf{f}_l)),$$

where the number 0.003 represents the average number of interest points per test image divided by the size of the image (see Sec. 3);

(b) $(\mathbf{f}_l, \tilde{\mathbf{f}}_l) \in \mathcal{E}_h$:

$$P((\mathbf{f}_k, \mathbf{f}_o) \in \mathcal{E}_h|T, \neg M) \approx (0.003) P_{\text{off}}(s(\mathbf{f}_l, \tilde{\mathbf{f}}_l); \mathbf{f}_l) \frac{1}{size(I)} \frac{1}{13} \frac{1}{2\pi}.$$

In the last term, we assume uniform distribution of position (one divided by the image size), main orientation (one divided by 2π), and scale (one divided by the total number of scales – see Sec.2.1) given a background feature.

- Select the hypothesis with maximum value for the Eq. 8, and if this value is above a threshold (here, this threshold is 0.5), accept it as a match.

The last three points represent the verification step. Two image sequences were used (see Fig. 15), where the Kevin sequence contains 120 frames, and the Dudek sequence contains 140 frames. Table 2 shows the recognition performance for the sequences of Fig. 15. Notice the significantly better performance in terms of true/false positives and false negatives matched in both sequences. Table 3 shows the average time spent (in seconds per test image) in the main activities of the recognition system run on a state-of-the-art PC computer for the sequences of Fig. 15. Notice the

(a) Kevin sequence (four of 120 frames)

(b) Dudek sequence (four of 140 frames)

Fig. 15. Sequences used to assess the performance of the recognition system. The contour represents the model (first column) and the matches (columns 2-4) in the respective sequences.

substantial reduction in computation time per test image achieved with the use of the classifier.

8 Summary and Conclusions

In this paper, we introduce a method to quantitatively characterize the distinctiveness and robustness of local image descriptors. This characterization is shown to provide a useful classification method that selects well behaved descriptors to be stored in the model database. Moreover, this characterization is used to formulate more accurately the recognition process. We further present a discriminative classifier that provides a fast and reliable descriptor selection, and a regressor that estimates the robustness and distinctiveness properties of the descriptor. Finally, we show that such classifier and regressor models not only reduce significantly the

Table 2

Performance of the recognition system in terms of true positive (TP), false positive (FP), and false negative (FN) produced in the sequences of Fig. 15 (with and without the neural net (NN) classifier). Note that TP + FN = Sequence length because the system either detects or does not detect the visual object. However, the number of false positives (FP) can be anything greater than or equal to zero since a single image can have more than one matching of the same object.

Kevin Sequence	Sequence length	TP	FP	FN
with NN classifier	120	120	0	0
without NN classifier	120	108	5	12
Dudek Sequence	Sequence length	TP	FP	FN
with NN classifier	140	133	0	7
without NN classifier	140	106	0	34

Table 3

Average time performance per frame (in seconds) of each step of the recognition algorithm with and without the neural net (NN) classifier.

	with NN classifier	*without* NN classifier
Database search	1	40
Outlier rejection	2	120
Verification	5	600
Total	8	760

recognition time, but they also allow for a more accurate recognition.

References

[1] S. Agarwal, D. Roth, Learning a sparse representation for object detection, in: European Conference on Computer Vision, 2002, pp. 113–130.

[2] Y. Amit, D. Geman, A computational model for visual selection, Neural Computation 11 (1999) 1691–1715.

[3] A. Bosch, A. Zisserman, X. Munos, Image classification using random forests, in: Proc. Interntaional Conference on computer Vision, Rio de Janeiro, Brazil, 2007.

[4] G. Carneiro and D. Lowe, Sparse flexible models of local features, in European conference on Computer Vision, Graz, Austria, 2006, Vol. 3, pp. 29–43.

[5] G. Carneiro, A. Jepson, Phase-based local features, in: European Conference on Computer Vision, Copenhagen, Denmark, 2002, pp. 282–296.

[6] G. Carneiro, A. Jepson, Multi-scale phase-based local features, in: IEEE Conference on Computer Vision and Pattern Recognition, Madison, Wisconsin, USA, 2003.

[7] G. Carneiro, A. Jepson, Flexible Spatial Configuration of Local Image Features, in: IEEE Transactions on Pattern analysis nad Machine Intelligence, 29, 12 ,2007

[8] G. Carneiro, A. Jepson, The distinctiveness, detectability, and robustness of local image features, in: IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 2005.

[9] G. Csurka, c. Dance, L. Fan, J. Willamowski, C. Bray, Visual characterization with bags of keypoints, in: Workshop on Statistical Learning in computer Vision, ECCV. 2004.

[10] A. Davison, Real-time simultaneous localisation and mapping with a single camera, in: Internationa Conference on Computer Vision, Nice, France, 2003.

[11] L. Ding and A. Martinez, Precise Detailed Detection of Faces and Facial Features, in: IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 2008.

[12] G. Dorko, C. Schmid, Selection of scale-invariant parts for object class recognition, in: International Conference on Computer Vision, Nice, France, 2003.

[13] L. Fei-Fei, R. Fergus, and P. Perona, A Bayesian Approach to Unsupervised One-Shot Learning of Object Categories, Proc. Ninth Int. Conf. Computer Vision, pp. 1134-1141, Oct. 2003.

[14] R. Fergus, P. Perona, A. Zisserman, Object class recognition by unsupervised scale-invariant learning, in: IEEE Conference on Computer Vision and Pattern Recognition, 2003.

[15] D. Fleet, Measurement of Image Velocity, Kluwer Academic Publishers, 1992.

[16] W. Freeman, E. Adelson, The design and use of steerable filters, IEEE Transactions on Pattern Analysis and Machine Intelligence 13 (9) (1991) 891–906.

[17] K. Fukunaga, Introduction to statistical pattern recognition. Academic press. 1990.

[18] C. Harris, M. Stephens, A combined corner and edge detector, in: Alvey Vision Conference, 1988.

[19] X. He, R. Zemel, V. Mnih, Learning landmarks for localization via manifolds, in: Workshop on Machine Learning Based Robotics in Unstructured Environments (NIPS), Vancouver, Canada, 2005.

[20] Y. Ke, R. Sukthankar, PCA-SIFT: A more distinctive representation for local image descriptors, in: IEEE CVPR. 2004.

[21] I. Laptev, Improvemens of object detection using boosted histograms, in: Proc. British Machine Vision Conference, Edinburgh, UK, 2006.

[22] D. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer VisionPaper accepted for publication.

[23] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, L. van Gool., A comparison of affine region detectors, International Journal of Computer Vision 65 (7) (2005.) 43–72.

[24] K. Mikolajczyk, C. Schmid, Indexing based on scale invariant interest points, in: IEEE International Conference on Computer Vision, Vancouver, Canada, 2001, pp. 525–531.

[25] H. Murase, S. Nayar, Visual learning and recognition of 3-d objects from appearance, International Journal of Computer Vision 14 (1) (1995) 5–24.

[26] I. Nabney, C. Bishop, Netlab neural network software, http://www.ncrg.aston.ac.uk/netlab/. (2003).

[27] R. Nelson, Memory-based recognition for 3-d objects, in: ARPA Image Understanding Workshop, Palm Springs, USA, 1996, pp. 1305–1310.

[28] K. Ohba, K. Ikeuchi, Detectability, uniqueness, and reliability of eigen windows for stable verification of partially occluded objects, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (9) (1997) 1043–1048.

[29] A. Pope, D. Lowe, Probabilistic models of appearance for 3d object recognition, International Journal of Computer Vision 40 (2) (2000) 149–167.

[30] P. Sala, R. Sim, A. Shokoufandeh, S. Dickinson, Landmark selection for vision-based navigation, IEEE Transactions on Robotics 22 (2) (2006) 334–349.

[31] F. Schaffalitzky, A. Zisserman, Automated scene matching in movies, in: Proceedings of the Challenge of Image and Video Retrieval, London, LNCS 2383, Springer-Verlag, 2002, pp. 186–197.

[32] C. Schmid, R. Mohr, Local grayvalue invariants for image retrieval, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (5) (1997) 530–535.

[33] S. Se, D. Lowe, J. Little, Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks, International Journal of Robotics Research 21 (8).

[34] R. Sim, G. Dudek, Learning generative models of scene features, in: IEEE Conference on Computer Vision and Pattern Recognition, Hawaii,USA, 2001, pp. 646–655.

[35] J. Sivic, F. Schaffalitzky, A. Zisserman, Object level grouping for video shots, in: European Conference on Computer Vision, Prague, Czech Republic, 2004.

[36] P. Torr, D. Murray, The development and comparison of robust methods for estimating the fundamental matrix, International Journal of Computer Vision 24 (3) (1997) 271–300.

[37] M. Weber, M. Welling, P. Perona, Unsupervised learning of models for recognition, in: ECCV (1), 2000, pp. 18–32.

[38] J. Wu, Bayesian estimation of stereo disparity from phase-based measurements, Master's thesis, Queen's Univeristy, Kingston, Ontario, Canada (2000).

[39] A. Yuille, J. Coughlan, High-level and generic models for visual search: When does high level knowledge help?, in: IEEE Conference on Computer Vision and Pattern Recognition, 1999.

[40] W. Zhang and J. Kosecka, Hierarchical building recognition, in: Image and Vision Computing, 25(5), pp. 704-716, 2007.

[41] Http://www.cs.utoronto.ca/ carneiro/databases.pdf.

A Image Deformations Studied

The image deformations described in this section are used to evaluate the robustness to perturbations of the interest point detector and the local feature extractor. The set of image deformations $\mathcal{DF} = \{d\}$ considered here are: a) two types of global brightness changes, b) non-uniform local brightness variations, c) additive noise, d) scale changes, e) 2D rotation, f) shear and g) sub-pixel translation. The non-uniform global brightness changes are implemented by adding a constant to the brightness value, taking into account the gamma correction non-linearity: $\tilde{I}_d(\mathbf{x}) = 255 * \left[\max\left(0, \left(\frac{I(\mathbf{x})}{255}\right)^{\gamma} + k\right)\right]^{\frac{1}{\gamma}}$, where $\gamma = 2.2$, I is the original image, and $k \in \{-0.5, -0.25, 0, 0.25, 0.5\}$ controls the changes in brightness. The resulting image is linearly mapped to values between 0 and 255, and then quantized. The uniform brightness change is simply based on the division of gray values by a constant $c \in \{1, 2, 3\}$.

For the non-uniform local brightness variations, highlights are simulated at specific locations of the image $\{\mathbf{x}_i | i = 1, ..., N\}$, where the positions $\mathbf{x}_i$ are selected at regular intervals of 15 pixels both in the horizontal and vertical directions. The highlights are simulated by adding the following image of Gaussian blobs:

$$I_g(\mathbf{x}) = \sum_{i=1}^{N} r_i g(\mathbf{x} - \mathbf{x}_i; \sigma), \tag{A.1}$$

where $\sigma = 15$, r_i is a normally distributed random variable with mean zero and standard deviation one, and $g(\mathbf{x}; \sigma) = \exp\left(-x^2/(2\sigma^2)\right)$. The deformed image is then computed as $\tilde{I}_d(\mathbf{x}) = I(\mathbf{x}) + pI_g(\mathbf{x})$, where $p \in \{5, 10, 15, 20, 25, 30\}$. Again, the resulting image is mapped to values between 0 and 255, and then quantized. For noise deformations, we simply add Gaussian noise with varying standard deviation ($\sigma = 255 * \{10^{-3}, 10^{-2}, 10^{-1}\}$), followed by normalization and quantization, as

Fig. A.1. Model image for deformations in Fig. A.2.

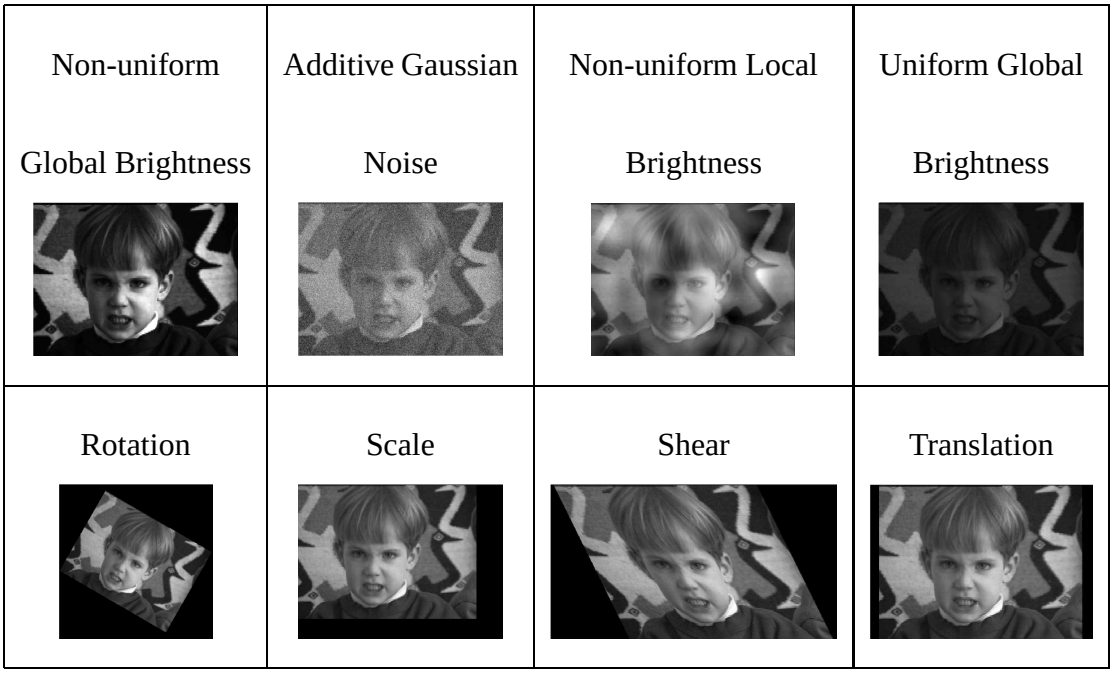


Fig. A.2. Image deformations studied.

above. The geometric deformations are 2D rotations (from -90^o to $+90^o$ in intervals of 15^o), uniform scale changes (with expansion factors in the range $[0.25, 1]$ with steps of 0.125), shear in the horizontal direction (so that a vertical line is perturbed by $\pm 26^o$), and sub-pixel translation (in the range [0,1] in steps of 0.2) pixel. The geometrically deformed images are quantized to $[0, 255]$ without normalization. All the deformations described above are depicted in Fig. A.2, which shows several deformed versions of the image in Fig. A.1.

B Database of Images used in the Quantitative Evaluations

The images used for the quantitative evaluation consist of general pictures of landscape, people, animals, and texture. We use a pool of 270 images and randomly sample 30 to form the foreground database (see Figures B.1), and the remaining 240 images form the background database (Figure B.2) The full database is available in [41].

Fig. B.1. Subset of database of images $\mathcal{T}$.

Fig. B.2. Subset of database of images $\mathcal{R}$.