Towards Optimal Naive Bayes Nearest Neighbor

Régis Behmo[1], Paul Marcombes[1,2], Arnak Dalalyan[2], and Véronique Prinet[1]

[1] NLPR / LIAMA, Institute of Automation, Chinese Academy of Sciences*
[2] IMAGINE, LIGM, Université Paris-Est

Abstract. Naive Bayes Nearest Neighbor (NBNN) is a feature-based image classifier that achieves impressive degree of accuracy [1] by exploiting 'Image-to-Class' distances and by avoiding quantization of local image descriptors. It is based on the hypothesis that each local descriptor is drawn from a class-dependent probability measure. The density of the latter is estimated by the non-parametric kernel estimator, which is further simplified under the assumption that the normalization factor is class-independent. While leading to significant simplification, the assumption underlying the original NBNN is too restrictive and considerably degrades its generalization ability. The goal of this paper is to address this issue.

As we relax the incriminated assumption we are faced with a parameter selection problem that we solve by hinge-loss minimization. We also show that our modified formulation naturally generalizes to optimal combinations of feature types. Experiments conducted on several datasets show that the gain over the original NBNN may attain up to 20 percentage points. We also take advantage of the linearity of optimal NBNN to perform classification by detection through efficient sub-window search [2], with yet another performance gain. As a result, our classifier outperforms — in terms of misclassification error — methods based on support vector machine and bags of quantized features on some datasets.

1 Introduction

With the advent in recent years of powerful blob and corner detectors and descriptors, the orderless bag of quantized features — also called bag of words (BoW) — has been the preferred image representation for image classification. The BoW owes its popularity to its relative simplicity and its ability to produce a compact, finite-dimensional representation that can be used as input of a state-of-the-art classifier such as support vector machine (SVM) or Adaboost. One can cite several highly competitive approaches that are essentially based on the BoW/SVM combination [3,4,5,6]. In this paper, we propose an alternative to mainstream methods based on parameter-optimized version of the NBNN.

In BoW representations, the quantization step results in a substantial loss of discriminative power of the visual features [6,1]. This loss was quantitatively measured in [1] and it is argued that the popularity enjoyed by the BoW/SVM combination is due to the efficiency of the SVM classifier, not to the representation itself. In simple words, most, but not all, of the information discarded by the feature quantization step is offset

* The first author is supported by a INRIA-Cordi grant. This work was partially supported by the Chinese Ministry of Science and Technology.

K. Daniilidis, P. Maragos, N. Paragios (Eds.): ECCV 2010, Part IV, LNCS 6314, pp. 171–184, 2010.

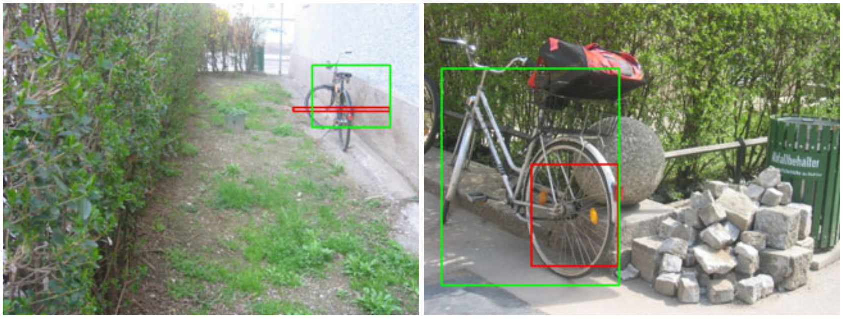

Fig. 1. Subwindow detection for the original NBNN (red) and for our version of NBNN (green). Since the background class is more densely sampled than the object class, the original NBNN tends to select an object window that is too small relatively to the object instance. As show these examples, our approach addresses this issue.

by the efficiency of the classifier. Naive Bayes Nearest Neighbor (NBNN) is a classifier introduced in [1] that was designed to address this issue: NBNN is non-parametric, does not require any feature quantization step and thus uses to advantage the full discriminative power of visual features. However, in practice, we observe that NBNN performs relatively well on certain datasets, but not on others. To remedy this, we start by analyzing the theoretical foundations of the NBNN. We show that this performance variability could stem from the assumption that the normalization factor involved in the kernel estimator of the conditional density of features is class-independent. We relax this assumption and provide a new formulation of the NBNN which is richer than the original one. In particular, our approach is well suited for optimal, multi-channel image classification and object detection. The main argument of NBNN is that the log-likelihood of a visual feature can be approximated by the distance to its nearest neighbor. In our formulation, this log-likelihood is approximately equal to an affine function of the nearest neighbor distance. The latter involves two affine coefficients that, in general, depend on properties of the training feature set. Our first contribution consists in a method to optimize these parameters by solving a linear problem that minimizes the cross-validated hinge loss. In addition, this new formulation generalizes well to optimal combinations of features of different types, here referred to as channels. The distance correction parameters also serve to balance each feature channel according to its relative relevance, as not all feature channels are equally useful to the problem at hand. As our last contribution, we show how to reformulate our classifier to perform object detection and classification by detection. In classification by detection (cf. [7] and the references therein), the aim is to classify images that contain an object embedded in a strongly cluttered background. Our solution consists in finding the image subwindow that maximizes a function that is linear in the image features. Due to this linearity, the optimal object location can be found by branch and bound subwindow search [2].

We conducted some experiments that reveal that affine distance correction improves NBNN performance by up to 20 percentage points. This indicates that our modified formulation is not merely a theoretical improvement, but is also of practical interest. Moreover, this gain is obtained with little computational overhead, compared to the original NBNN formulation. Interesting results are also given concerning the relative efficiency of radiometry invariant SIFT features [8]: Opponent SIFT is the descriptor that performed worst in NBNN, but it becomes the most efficient descriptor in our formulation.

The idea of designing optimal combinations of different feature channels by cross-validation has been studied by several authors. In the present context, the most relevant reference is [9]. While the method in [9] was conceived with the idea of having just one descriptor per image (either a global texture descriptor or a bag of words), our method works best when the number of descriptors per image is large. In [4,10], an image is subdivided into a pyramid of regions at different scales, and each region represents a channel. This fundamentally differs from our work in that they use bags of words to represent each image subregion. The idea of considering each image region as a channel can be applied in our context without any modification. With respect to the sub-window search, the idea is that in a cluttered background, classification performs best when first locating the most likely object position. This is close to the concept of region of interest developed in [11]. The detection scheme we use is inspired by [2,12].

The remainder of this paper is organized as follows. Original NBNN as well as the modification we propose are summarized in Section 2. In section 3, the adaptation of the optimal NBNN formulation to the problem of object detection is presented. Experimental results on three real datasets are reported in section 4.

2 Parametric NBNN Classification

2.1 Initial Formulation of NBNN

In this section, we briefly recall the main arguments of NBNN described by Boiman *et al.* [1] and introduce some necessary notation.

In an image I with hidden class label c_I, we extract K_I features $(d_k^I)_k \in \mathbb{R}^D$. Under the naive Bayes assumption, and assuming all image labels are equally probable ($P(c) \sim cte$) the optimal prediction $\hat{c}_I$ of the class label of image I maximizes the product of the feature probabilities relatively to the class label:

$$\hat{c}_I = \arg\max_c \prod_{k=1}^{K_I} P(d_k^I|c). \tag{1}$$

The feature probability conditioned on the image class $P(d_k^I|c)$ can be estimated by a non-parametric kernel estimator, also called Parzen-Rosenblatt estimator. If we note $\chi^c = \left\{d_k^J | c_J = c, 1 \leq k \leq K_J\right\}$ the set of all features from all training images that belong to class c, we can write:

$$P(d_k^I|c) = \frac{1}{Z} \sum_{d \in \chi^c} \exp\left(\frac{- \parallel d_k^I - d \parallel^2}{2\sigma^2}\right), \tag{2}$$

where σ is the bandwidth of the density estimator. In [1], this estimator is further approximated by the largest term from the sum on the RHS. This leads to a quite simple expression:

$$\forall d,\ \forall c,\ -\log\left(P(d|c)\right) \simeq \min_{d' \in \chi^c} \parallel d - d' \parallel^2 . \tag{3}$$

The decision rule for image I is thus:

$$\hat{c}_I = \arg\max_c P(I|c) = \arg\min_c \sum_k \min_{d \in \chi^c} \parallel d_k^I - d \parallel^2 . \tag{4}$$

This classifier is shown to outperform the usual nearest neighbor classifier. Moreover, it does not require any feature quantization step, and the descriptive power of image features is thus preserved.

The reasoning above proceeds in three distinct steps: the naive Bayes assumption considers that image features are independent identically distributed given the image class c_I (equation 1). Then, the estimation of a feature probability density is obtained by a non-parametric density estimation process like the Parzen-Rosenblatt estimator (equation 2). NBNN is based on the assumption that the logarithm of this value, which is a sum of distances, can be approximated by its largest term (equation 3). In the following section, we will show that the implicit simplification that consists in removing the normalization parameter from the density estimator is invalid in most practical cases.

Along with the notation introduced in this section, we will also need the notion of point-to-set distance, which is simply the squared Euclidean distance of a point to its nearest neighbor in the set: $\forall \Omega \subset \mathbb{R}^D$, $\forall x \in \mathbb{R}^D$, $\tau(x, \Omega) = \inf_{y \in \Omega} \parallel x - y \parallel^2$. In what follows, $\tau(x, \chi^c)$ will be abbreviated as $\tau^c(x)$.

2.2 Affine Correction of Nearest Neighbor Distance for NBNN

The most important theoretical limitation of NBNN is that in order to obtain a simple approximation of the log-likelihood, the normalization factor $1/Z$ of the kernel estimator is assumed to be the same for all classes. Yet, there is no *a priori* reason to believe that this assumption is satisfied in practice. If this factor significantly varies from one class to another, then the approximation of the maximum a posteriori class label $\hat{c}_I$ by equation 4 becomes unreliable.

It should be noted that the objection that we raise does not concern the core hypothesis of NBNN, namely the naive Bayes hypothesis and the approximation of the sum of exponentials of equation 2 by its largest term. In fact, in the following we will essentially follow and extend the arguments presented in [1] using the same starting hypothesis.

Non-parametric kernel density estimation requires the definition of a smoothing parameter σ, also called bandwidth. We consider the general case of a sample of K points $\{x_k | 1 \leq k \leq K\}$ drawn from a probability measure defined on some D-dimensional feature space Ω. The density of this probability measure can be estimated by:

$$\forall x \in \Omega, f(x) = \frac{1}{Z} \sum_{k=1}^{K} \exp\left(-\frac{||x - x_k||^2}{2\sigma^2}\right) . \tag{5}$$

The value of Z is obtained by normalization of the density function: $\int_\Omega f(x)dx = 1 \Leftrightarrow Z = K(2\pi)^{\frac{D}{2}}\sigma^D$. We retain the NBNN assumption that the likelihood of a feature is approximately equal to the value of the largest term from the sum on the right hand side of equation 5. Here we provide an argument that supports this assumption: it is known that the convergence speed of the Parzen-Rosenblatt (PR) estimator is $K^{-4/(4+D)}$ [13]. This means that in the case of a 128-dimensional feature space, such as the SIFT feature space, in order to reach an approximation bounded by $1/2$ we need to sample 2^{33} points. In practice, the PR estimator does not converge and there is little sense in keeping more than just the first term of the sum.

Thus, the log-likelihood of a visual feature d relatively to an image label c is:

$$-\log\big(P(d|c)\big) = -\log\left\{\frac{1}{Z^c}\exp\left(-\frac{\tau^c(d)}{2(\sigma^c)^2}\right)\right\} = \frac{\tau^c(d)}{2(\sigma^c)^2} + \log(Z^c), \tag{6}$$

where $Z^c = |\chi^c|(2\pi)^{\frac{D}{2}}(\sigma^c)^D$. Recall that $\tau^c(d)$ is the squared Euclidean distance of d to its nearest neighbor in χ^c. In the above equations, we have replaced the class-independent notation σ, Z by σ^c, Z^c since, in general, there is no reason to believe that parameters should be equal across classes. For instance, both parameters are functions of the number of training features of class c in the training set.

Returning to the naive Bayes formulation, we obtain:

$$\forall c, -\log\left(P(I|c)\right) = \sum_{k=1}^{K_I}\left(\frac{\tau^c(d_k^I)}{2(\sigma^c)^2} + \log(Z^c)\right) = \alpha^c\sum_{k=1}^{K_I}\tau^c(d_k^I) + K_I\beta^c, \tag{7}$$

where $\alpha^c = 1/(2(\sigma^c)^2)$ and $\beta^c = \log(Z^c)$ is a re-parametrization of the log-likelihood 6 that has the advantage of being linear in the model parameters. The image label is then decided according to a criterion that is slightly different from equation 4:

$$\hat{c}_I = \arg\min_c\left(\alpha^c\sum_{k=1}^{K_I}\tau^c(d_k^I) + K_I\beta^c\right). \tag{8}$$

We note that this modified decision criterion can be interpreted in two different ways: it can either be interpreted as the consequence of a density estimator to which a multiplicative factor was added, or as an unmodified NBNN in which an affine correction has been added to the squared Euclidean distance. In the former, the resulting formulation can be considered different from the initial NBNN. In the latter, equation 8 can be obtained from equation 4 simply by replacing $\tau^c(d)$ by $\alpha^c\tau^c(d) + \beta^c$ (since α^c is positive, the nearest neighbor distance itself does not change). This formulation differs from [1] only in the evaluation of the distance function, leaving us with two parameters per class to be evaluated.

At this point, it is important to recall that the introduction of parameters α^c and β^c does not violate the naive Bayes assumption, nor the assumption of equiprobability of classes. In fact, the density estimation correction can be seen precisely as an enforcement of these assumptions. If a class is more densely sampled than others (i.e: its feature space contains more training samples), then the NBNN estimator will have a bias towards that class, even though it made the assumption that all classes are equally probable. The purpose of setting appropriate values for α^c and β^c is to correct this bias.

It might be noted that deciding on a suitable value for α^c and β^c reduces to defining an appropriate bandwidth σ^c. Indeed, the dimensionality D of the feature space and the number $|\chi^c|$ of training feature points are known parameters. However, in practice, choosing a good value for the bandwidth parameter is time-consuming and inefficient. To cope with this issue, we designed an optimization scheme to find the optimal values of parameters α^c, β^c with respect to cross-validation.

2.3 Multi-channel Image Classification

In the most general case, an image is described by different features coming from different sources or sampling methods. For example, we can sample SIFT features and local color histograms from an image. We observe that the classification criterion of equation 1 copes well with the introduction of multiple feature sources. The only difference should be the parameters for density estimation, since feature types correspond, in general, to different feature spaces.

In order to handle different feature types, we need to introduce a few definitions and adapt our notation. In particular, we define the concept of *channel*: a channel χ is a function that associates a set of finite-dimensional characteristics to an image I: $\forall I, \chi(I) \subset \mathbb{R}^{d_\chi}$. Channels can be defined arbitrarily: a channel can be associated to a particular detector/descriptor pair, but can also represent global image characteristics. For instance, an image channel can consist in a single element, such as the global color histogram.

Let us assume we have defined a certain number of channels $(\chi_n)_{1\leq n\leq N}$, that are expected to be particularly relevant to the problem at hand. Adapting the framework of our modified NBNN to multiple channels is just a matter of changing notation. Similarly to the single-channel case, we aim here at estimating the class label of an image I:

$$\hat{c}_I = \arg\max_c P(I|c), \quad \text{with} \quad P(I|c) = \prod_n \prod_{d\in\chi_n(I)} P(d|c). \tag{9}$$

Since different channels have different features spaces, the density correction parameters should depend on the channel index: α^c, β^c will thus be noted α^c_n, β^c_n. The notation from the previous section are adapted in a similar way: we call $\chi^c_n = \bigcup_{J|c_J=c} \chi_n(J)$ the set of all features from class c and channel n and define the distance function of a feature d to χ^c_n by: $\forall d,\ \tau^c_n(d) = \tau(d, \chi^c_n)$. This leads to the classification criterion:

$$\hat{c}_I = \arg\min_c \sum_n \left(\alpha^c_n \sum_{d\in\chi_n(I)} \tau^c_n(d) + \beta^c_n |\chi_n(I)| \right). \tag{10}$$

Naturally, when adding feature channels to our decision criterion, we wish to balance the importance of each channel relatively to its relevance to the problem at hand. Equation 10 shows us that the function of relevance weighting can be assigned to the distance correction parameters. The problems of adequate channel balancing and nearest neighbor distance correction should thus be addressed in one single step. In the following section, we present a method to find the optimal values of these parameters.

2.4 Parameter Estimation

We now turn to the problem of estimating values of α^c_n and β^c_n that are optimal for classification. To simplify mathematical derivations, let us denote by $\mathbf{X}^c(I)$ the vector in $\mathbb{R}^{2N}$ defined by

$$X_n^c(I) = \sum_{d \in \chi_n(I)} \tau_n^c(d), \qquad X_{N+n}^c(I) = |\chi_n(I)|, \quad \forall n = 1, \dots, N. \tag{11}$$

For every c, the vector $\mathbf{X}^c(I)$ can be considered as a global descriptor of image I. We also denote by $\boldsymbol{\omega}^c$ the $(2N)$-vector $(\alpha_1^c, \dots, \alpha_N^c, \beta_1^c, \dots, \beta_N^c)$ and by W the matrix that results from concatenation of vectors w^c for different values of c. Using these notation, the classifier we propose can be rewritten as:

$$\hat{c}_I = \arg\min\nolimits_c \; (\boldsymbol{\omega}^c)^\top \mathbf{X}^c(I), \tag{12}$$

where $(\boldsymbol{\omega}^c)^\top$ stands for the transpose of $\boldsymbol{\omega}^c$. This is close in spirit to the winner-takes-all classifier widely used for the multiclass classification.

Given a labeled sample $(I_i, c_i)_{i=1,\dots,K}$ independent of the sample used for computing the sets χ_n^c, we can define a constrained linear energy optimization problem that minimizes the hinge loss of a multi-channel NBNN classifier:

$$E(W) = \sum_{i=1}^{K} \max_{c:c \neq c_i} \big(1 + (\boldsymbol{\omega}^{c_i})^\top \mathbf{X}^{c_i}(I_i) - (\boldsymbol{\omega}^c)^\top \mathbf{X}^c(I_i)\big)_+, \tag{13}$$

where $(x)_+$ stands for the positive part of a real x. The minimization of $E(W)$ can be recast as a linear program since it is equivalent to minimizing $\sum_i \xi_i$ subject to constraints:

$$\xi_i \geq 1 + (\boldsymbol{\omega}^{c_i})^\top \mathbf{X}^{c_i}(I_i) - (\boldsymbol{\omega}^c)^\top \mathbf{X}^c(I_i), \quad \forall i = 1, \dots, K, \; \forall c \neq c_i, \tag{14}$$

$$\xi_i \geq 0 \quad \forall i = 1, \dots, K, \tag{15}$$

$$(\boldsymbol{\omega}^c)^\top \mathbf{e}_n \geq 0, \quad \forall n = 1, \dots, N, \tag{16}$$

where $\mathbf{e}_n$ stands for the vector of $\mathbb{R}^{2N}$ having all coordinates equal to zero, except for the nth coordinate, which is equal to 1. This linear program can be solved quickly for a relatively large number of channels and images[1]. In practice, the number of channels should be kept small relatively to the number of training samples to avoid overfitting. The computational complexity of solving the aforementioned linear program is negligible w.r.t. the complexity of computing the global descriptors $\mathbf{X}^c$ based on the nearest neighbor search.

Our contribution at this point is two-fold. We have proposed a natural parametric version of NBNN that is designed to improve the predictive performance of NBNN. We have also integrated the possibility to optimally combine multiple feature channels in the classifier. Due to the fact that we estimate the distance correcting weights through the optimization of the hinge loss, the parameters α_n^c, β_n^c up-weight channels that are most relevant to classification.

3 Multi-channel Classification by Detection

Optimal NBNN was designed in the goal of classifying images with little background clutter, and it is bound to fail on images containing too many background features.

[1] Our implementation makes use of the GNU linear programming kit
`http://www.gnu.org/software/glpk/`

Classification by detection (*cf.* [7] and the references therein) is tailored to this kind of situations. It consists in selecting the image region that is the most likely to contain the object instance. In this section, we adapt our optimal NBNN to the problems of classification by detection. We will see that the final formulation is mostly identical to the formulation we adopted for general image classification.

We shall adopt the experimental framework given by the annotated Graz-02 dataset [14]: object instances from class c are surrounded by background clutter, denoted $\overline{c}$. Keeping the initial naive Bayes as well as the class equiprobability assumptions, our goal is to maximize the probability of the joint object label c and position π inside the image conditioned on the image content. We further assume object positions are equiprobable ($P(\pi|c) = P(\pi) = cte$). The image class estimate now takes the following form: $(\hat{c}_I, \hat{\pi}_I) = \arg\max_{c,\pi} P(I|c,\pi)$. Following the same line of thought as in NBNN, we can expand the likelihood term under the naive Bayes assumption: $P(I|c,\pi) = \prod_n \prod_{d \in \chi_n(I)} P(d|c,\pi)$ for all c and π.

At this point, we make the additional assumption that a feature probability knowing the object class and position only depends on the point belonging or not to the object:

$$\forall n,\ d,\ c,\ -\log\left(P(d|c,\pi)\right) = \begin{cases} \tau_n^c(d) \text{ if } d \in \pi \\ \tau_n^{\overline{c}}(d) \text{ if } d \notin \pi. \end{cases} \tag{17}$$

In the above equation, we have written the feature-to-set distance functions τ_n^c and $\tau_n^{\overline{c}}$ without apparent density correction in order to alleviate the notation. We leave to the reader the task of replacing τ_n^c by $\alpha_n^c \tau_n^c + \beta_n^c$ in the equations of this section.

The image log-likelihood function is now decomposed over all features inside and outside the object: $E(I,c,\pi) \triangleq -\log\big(P(I|c,\pi)\big) = \sum_n \big(\sum_{d\in\pi} \tau_n^c(d) + \sum_{d\notin\pi} \tau_n^{\overline{c}}(d)\big)$. The term on the RHS can be rewritten:

$$E(I,c,\pi) = \sum_n \Big\{ \sum_{d\in\pi} (\tau_n^c(d) - \tau_n^{\overline{c}}(d)) + \sum_d \tau_n^{\overline{c}}(d) \Big\}. \tag{18}$$

Observing that the second sum on the RHS does not depend on π, we get $E(I,c,\pi) = E_1(I,c,\pi) + E_2(I,c)$, where $E_1(I,c,\pi) = \sum_n \sum_{d\in\pi} \big(\tau_n^c(d) - \tau_n^{\overline{c}}(d)\big)$ and $E_2(I,c) = \sum_n \sum_d \tau_n^{\overline{c}}(d)$. Let us define the optimal object position $\hat{\pi}^c$ relatively to class c as the position that minimizes the first energy term: $\hat{\pi}^c = \arg\min_\pi E_1(I,c,\pi)$ for all c. Then, we can obtain the most likely image class and object position by:

$$\hat{c}_I = \arg\min_c \left(E_1(I,c,\hat{\pi}^c) + E_2(I,c)\right), \qquad \hat{\pi}_I = \hat{\pi}^{\hat{c}_I}. \tag{19}$$

For any class c, finding the rectangular window $\hat{\pi}^c$ that is the most likely candidate can be done naively by exhaustive search, but it proves prohibitive. Instead, we make use of fast branch and bound subwindow search [2]. The method used to search for the image window that maximizes the prediction of a linear SVM can be generalized to any classifier that is linear in the image features, such as our optimal multi-channel NBNN.

In short, the most likely class label and object position for a test image I are found by the following algorithm:

```
declare variables $\hat{c}$, $\hat{\pi}$
$\hat{E} = +\infty$
for each class label $c$ do
  find $\hat{\pi}^c$ by efficient branch and bound subwindow search
  $\hat{\pi}^c = \arg\min_\pi E_1(I,c,\pi)$
  if $E_1(I,c,\hat{\pi}^c) + E_2(I,c) < \hat{E}$ then
    $\hat{E} = E_1(I,c,\hat{\pi}^c) + E_2(I,c)$
    $\hat{c} = c$
    $\hat{\pi} = \hat{\pi}^c$
  end if
end for
return $\hat{c}$, $\hat{\pi}$
```

4 Experiments

Our optimal NBNN classifier was tested on three datasets: Caltech-101 [15], SceneClass 13 [16] and Graz-02 [14]. In each case, the training set was divided into two equal parts for parameter selection. Classification results are expressed in percent and reflect the rate of good classification, per class or averaged over all classes.

A major practical limitation of NBNN and of our approach is the computational time necessary to nearest neighbor search, since the sets of potential nearest neighbors to explore can contain of the order of 10^5 to 10^6 points. We thus need to implement an appropriate search method. However, the dimensionality of the descriptor space can also be quite large and traditional exact search methods, such as kd-trees or vantage point trees [17] are inefficient. We chose Locality Sensitive Hashing (LSH) and addressed the thorny issue of parameter tuning by multi-probe LSH[2] [18] with a recall rate of 0.8. We observed that resulting classification performance are not overly sensitive to small variations in the required recall rate. However, computations speed is: compared to exhaustive naive search, the observed speed increase was more than ten-fold. Further improvement in the execution times can be achieved using recent approximate NN-search methods [19,20].

Let us describe the databases used in our experiments.

Caltech-101 (5 classes). This dataset includes the five most populated classes of the Caltech-101 dataset: faces, airplanes, cars-side, motorbikes and background. These images present relatively little clutter and variation in object pose. Images were resized to a maximum of 300 × 300 pixels prior to processing. The training and testing sets both contain 30 randomly chosen image per class. Each experiment was repeated 20 times and we report the average results over all experiments.

SceneClass 13. Each image of this dataset belongs to one of 13 indoor and outdoor scenes. We employed 150 training images per class and assigned the rest to the testing set.

Graz-02. This manually segmented dataset contains instances of three classes: bike, people or car. Each image belongs to just one class. The training and testing sets

[2] For which an open source implementation exists: `http://lshkit.sourceforge.net/`

are both composed of 100 images per class. This database is considered as challenging [21] since the objects of interest are not necessarily central or dominant. Furthermore, they are subject to significant pose variation and partial occlusion.

4.1 Single-Channel Classification

The impact of optimal parameter selection on NBNN is measured by performing image classification with just one feature channel. We chose SIFT features [22] for their relative popularity. Results are summarized in Tables 1 and 2.

Table 1. Performance comparison between the bag of words classified by linear and χ^2-kernel SVM, the NBNN classifier and our optimal NBNN

Datasets	BoW/SVM	BoW/χ^2-SVM	NBNN [1]	Optimal NBNN
SceneClass13 [16]	67.85 ±0.78	76.7 ±0.60	48.52 ±1.53	75.35 ±0.79
Graz02 [14]	68.18 ±4.21	77.91 ±2.43	61.13 ±5.61	78.98 ±2.37
Caltech101 [15]	59.2 ±11.89	89.13 ±2.53	73.07 ±4.02	89.77 ±2.31

In Table 1, the first two columns refer to the classification of bags of words by linear SVM and by χ^2-kernel SVM. In all three experiments we selected the most efficient codebook size (between 500 and 3000) and feature histograms were normalized by their L^1 norm. Furthermore, only the results for the χ^2-kernel SVM with the best possible value (in a finite grid) of the smoothing parameter are reported. In Table 2, we omitted the results of BoW/SVM because of their clear inferiority w.r.t. BoW/χ^2-SVM.

Table 2. Performance comparison between the bag of words classified by χ^2-kernel SVM, the NBNN classifier and our optimal NBNN. Per class results for Caltech-101 (5 classes) dataset.

Class	BoW/χ^2-SVM	NBNN [1]	Optimal NBNN
Airplanes	91.99 ± 4.87	34.17 ±11.35	95.00 ± 3.25
Car-side	96.16 ± 3.84	97.67 ± 2.38	94.00 ± 4.29
Faces	82.67 ± 9.10	85.83 ± 9.02	89.00 ± 7.16
Motorbikes	87.80 ± 6.28	71.33 ±19.13	91.00 ± 5.69
Background-google	87.50 ± 6.22	76.33 ±22.08	79.83 ± 10.67

There are two lessons to be learned from these experiments: the first is that correcting the NBNN formulation proves to be an absolute necessity if we want use unquantized features to advantage. Indeed the gain produced by parameter selection is almost systematic and exceeds 15 percentage points (in average) for the SceneClass and Graz-02 datasets. Secondly, we observe that the accuracy of NBNN is comparable to the state-of-the-art classification procedures such as BoW/χ^2-SVM. It should also be noted unlike NBNN, BoW/χ^2-SVM involves a tuning parameter the choice of which is a delicate issue.

To our knowledge, the state-of-the-art reported in the literature are 73.4% [23] for SceneClass13 (with an experimental setting however different from ours, since the authors use half of the dataset for training and the other half for testing), and 82.7% [24] for Graz-02 (using 150 positive and 150 negative images for training, for each non-background class). Given the relatively small training set that we use, our results compare favorably.

4.2 Radiometry Invariance

In this experiment, results highlight the necessity of parametric density estimation to make best use of visual features. In [8], different radiometry invariants of SIFT are presented and their relative performances are evaluated. Our own experiments made with the initial formulation of NBNN concur with the conclusions of [8],

As we find that the most efficient descriptors are: rgSIFT, followed by cSIFT and transformed color SIFT (cf. Table 3). The order of these descriptors roughly corresponds to the conclusions of [8]. Experiments revealed that the performance exhibited by optimal NBNN reverse this sequence: opponentSIFT becomes one of the best descriptors, with 91.10% good classification rate, while rgSIFT performs worst, with 85.17%. Thus, a wrong evaluation of the feature space properties undermines the descriptor performance.

4.3 Multi-channel Classification

The notion of channel is sufficiently versatile to be adapted to a variety of different contexts. In this experiment, we borrow the idea developed in [4] to subdivide the image in different spatial regions. We consider that an image channel associated to a certain

Table 3. Caltech101 (5classes): Influence of various radiometry invariant features. Best and worst SIFT invariants are highlighted in blue and red, respectively.

Feature	BoW/χ^2-SVM	NBNN [1]	Optimal NBNN
SIFT	88.90 ±2.59	73.07 ±4.02	89.77 ±2.31
OpponentSIFT	89.90 ±2.18	72.73 ±6.01	91.10 ±2.45
rgSIFT	86.03 ±2.63	80.17 ±3.73	85.17 ±4.86
cSIFT	86.13 ±2.76	75.43 ±3.86	86.87 ±3.23
Transf. color SIFT	89.40 ±2.48	73.03 ±5.52	90.01 ±3.03

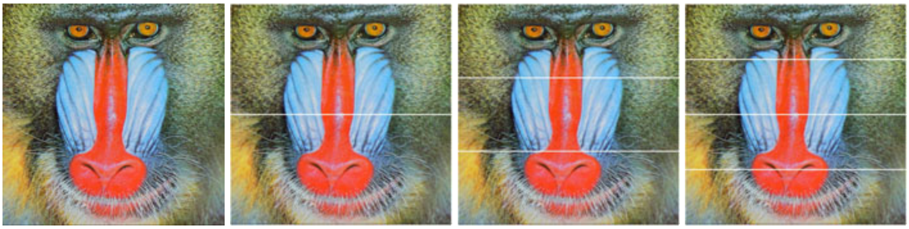

Fig. 2. Feature channels as image subregions: 1×1, 1×2, 1×3, 1×4

Table 4. Multi-channel classification, SceneClass13 dataset

Channels	#channels	NBNN	Optimal NBNN
1×1	1	48.52	75.35
$1 \times 1 + 1 \times 2$	3	53.59	76.10
$1 \times 1 + 1 \times 3$	4	55.24	76.54
$1 \times 1 + 1 \times 4$	5	55.37	78.26

image region is composed of all features that are located inside this region. In practice, image regions are regular grids of fixed size. We conducted experiments on the SceneClass13 dataset with 1 (1×1), 3 ($1 \times 1 + 1 \times 2$), 4 ($1 \times 1 + 1 \times 3$) and 5 ($1 \times 1 + 1 \times 4$) channels (see Fig. 2 for an illustration). Results are summarized in table 4. As we can see by comparing the first line with the subsequent lines, adding channels increases the rate of correct classification. Best performances are recorded in the experiment with the largest number of channels.

4.4 Classification by Detection

The Graz-02 dataset is a good example of the necessity of classification by detection for diminishing the importance of background clutter. In this set of experiments, the dataset is divided into just two classes: the positive class contains images of bicycles, while the negative class contains all other images. In this context, the estimated label of a test image I is given by:

$$\hat{c}_I = \text{sign}\left(E_2(I, back) - E(I, bike, \hat{\pi}^{bike})\right), \tag{20}$$

where we have retained notation from Section 3. The distance correction parameters that have to be determined for this problem are the α_n^c, β_n^c where c is in $\{bike, \overline{bike}, back\}$. For the sake of parameter selection, the sets of images from classes $bike$ and $\overline{bike}$ are obtained by decomposing each positive image in two complementary parts: the points located on a bicycle instance are in $bike$ while others are in $\overline{bike}$. Density estimation parameters were learned using the procedure described in Section 2.4.

We combined all five SIFT radiometry invariants already employed in Section 4.2. With classification by detection, we raised the classification rate of optimal NBNN from 78.70% to 83.60%, while classification by detection with NBNN achieved just 68.35%. Detection examples are shown in Fig. 1 and 3.

Our results by optimal NBNN (both for classification and classification by detection) are close to the results reported in [25,21] and [26], where the rate of classification is 77%, 80% and 84% respectively (see Table 5 for details).

Table 5. Per-class classification rate for the Graz-02 database

Class	NBNN	Optimal NBNN	Optimal NBNN	[21]	[25]	[26]
			by detection			
bike	68.35 ± 10.66	78.70 ± 4.67	83.6	80.5	77.8	84.4
people	45.10 ± 12.30	76.20 ± 5.85	–	81.7	81.2	–
car	42.40 ± 15.41	82.05 ± 4.88	–	70.1	70.5	79.9

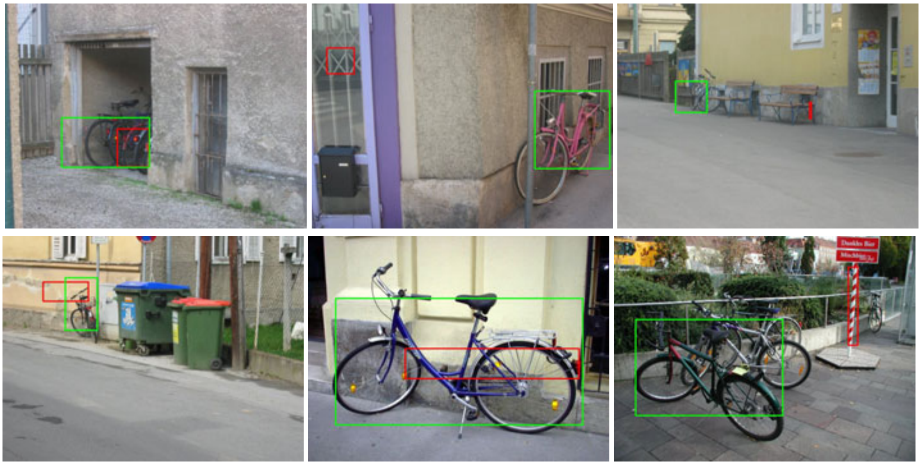

Fig. 3. Subwindow detection for NBNN (red) and optimal NBNN (green). For this experiment, all five SIFT radiometry invariants were combined. (see Section 4.4)

It can be observed that the non-parametric NBNN usually converges towards an optimal object window that is too small relatively to the object instance. This is due to the fact that the background class is more densely sampled. Consequently, the nearest neighbor distance gives an estimate of the probability density that is too large. It was precisely to address this issue that optimal NBNN was designed.

5 Conclusion

In this paper, we proposed a parametric version of the NBNN classifier as well as a method for learning the parameters from a labeled set of images. The flexibility of this new classifier is exploited for defining its multi-channel counterpart and for adapting it to the task of object localization and classification by detection. Both in theory and in practice, it is shown that the new approach is much more powerful than the original NBNN in the case where the number of features per class is strongly class-dependent. Furthermore, the experiments carried out on some standard databases demonstrate that parametric NBNN can compete with other state-of-the-art approaches to object classification. The C++ implementation of the optimal NBNN is made publicly available at `http://code.google.com/p/optimal-nbnn/`.

Testing alternative strategies for parameter optimization step [27] and combining our approach with approximate nearest-neighbor search [19] are interesting avenues for future research.

References

1. Boiman, O., Shechtman, E., Irani, M.: In defense of nearest-neighbor based image classification. In: CVPR (2008)
2. Lampert, C., Blaschko, M., Hofmann, T.: Beyond sliding windows: Object localization by efficient subwindow search. In: CVPR (2008)

3. Marszałek, M., Schmid, C., Harzallah, H., van de Weijer, J.: Learning object representations for visual object class recognition. In: Visual Recognition Challange workshop (2007)
4. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: CVPR (2006)
5. Zhang, H., Berg, A.C., Maire, M., Malik, J.: SVM-KNN: Discriminative nearest neighbor classification for visual category recognition. In: CVPR (2006)
6. Jégou, H., Douze, M., Schmid, C.: Improving bag-of-features for large scale image search. International Journal of Computer Vision 87, 316–336 (2010)
7. Harzallah, H., Jurie, F., Schmid, C.: Combining efficient object localization and image classification. In: International Conference on Computer Vision (2009)
8. van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.: Evaluating color descriptors for object and scene recognition. In: T-PAMI (2010)
9. Varma, M., Ray, D.: Learning the discriminative power-invariance trade-off. In: ICCV (2007)
10. Bosch, A., Zisserman, A., Munoz, X.: Representing shape with a spatial pyramid kernel. In: International Conference on Image and Video Retrieval, ICIVR (2007)
11. Bosch, A., Zisserman, A., Munoz, X.: Image classification using random forests and ferns. In: ICCV (2007)
12. Yuan, J., Liu, Z., Wu, Y.: Discriminative subvolume search for efficient action detection. In: CVPR (2009)
13. Stone, C.: Optimal uniform rate of convergence for nonparametric estimators of a density function or its derivatives. Recent advances in statistics (1983)
14. Marszałek, M., Schmid, C.: Accurate object localization with shape masks. In: CVPR (2007)
15. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. T-PAMI 28, 594–611 (2006)
16. Fei-Fei, L., Perona, P.: A bayesian hierarchical model for learning natural scene categories. In: CVPR (2005)
17. Yianilos, P.N.: Data structures and algorithms for nearest neighbor search in general metric spaces. In: SODA: ACM-SIAM Symposium on Discrete Algorithms (1993)
18. Dong, W., Wang, Z., Josephson, W., Charikar, M., Li, K.: Modeling LSH for performance tuning. In: CIKM, pp. 669–678. ACM, New York (2008)
19. Muja, M., Lowe, D.: Fast approximate nearest neighbors with automatic algorithm configuration. In: VISAPP (2009)
20. Jégou, H., Douze, M., Schmid, C.: Product quantization for nearest neighbor search. IEEE Transactions on Pattern Analysis & Machine Intelligence (to appear 2010)
21. Mutch, J., Lowe, D.G.: Object class recognition and localization using sparse features with limited receptive fields. Int. J. Comput. Vision 80, 45–57 (2008)
22. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV (2003)
23. Bosch, A., Zisserman, A., Munoz, X.: Scene classification via pLSA. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 517–530. Springer, Heidelberg (2006)
24. Ling, H., Soatto, S.: Proximity distribution kernels for geometric context in category recognition. In: ICCV (2007)
25. Opelt, A., Pinz, A., Fussenegger, M., Auer, P.: Generic object recognition with boosting. PAMI 28 (2004/2006)
26. Moosmann, F., Triggs, B., Jurie, F.: Fast discriminative visual codebooks using randomized clustering forests. In: Neural Information Processing Systems, NIPS (2006)
27. Lee, Y., Lin, Y., Wahba, G.: Multicategory support vector machines, theory, and application to the classification of microarray data and satellite radiance data. JASA 99, 67–81 (2004)