Supervised Learning of Quantizer Codebooks by Information Loss Minimization

Svetlana Lazebnik, *Member, IEEE,* Maxim Raginsky, *Member, IEEE*

***Abstract*—This paper proposes a technique for jointly quantizing continuous features and the posterior distributions of their class labels based on minimizing empirical information loss, such that the quantizer index of a given feature vector approximates a *sufficient statistic* for its class label. Informally, the quantized representation retains as much information as possible for classifying the feature vector correctly. We derive an alternating minimization procedure for simultaneously learning codebooks in the Euclidean feature space and in the simplex of posterior class distributions. The resulting quantizer can be used to encode unlabeled points outside the training set and to predict their posterior class distributions, and has an elegant interpretation in terms of lossless source coding. The proposed method is validated on synthetic and real datasets, and is applied to two diverse problems: learning discriminative visual vocabularies for bag-of-features image classification, and image segmentation.**

***Index Terms*—Pattern recognition, information theory, quantization, clustering, computer vision, scene analysis, segmentation.**

I. Introduction

Many computational tasks involving continuous signals such as speech or images can be made significantly easier by converting the high-dimensional feature vectors describing these signals into a series of discrete "tokens." *Nearest-neighbor quantization*, where a finite *codebook* is formed in the feature space, and then each feature vector is encoded by the index of its nearest codevector, is one of the most commonly used ways of discretizing continuous feature spaces [11]. In modern image and signal processing literature, codebooks are often formed not only for the sake of compressing high-dimensional data (the traditional goal of quantization), but also for the sake of facilitating the subsequent step of learning a statistical model for classification or inference. For example, *bag-of-features* models for image classification [8], [39], [43] work by quantizing high-dimensional descriptors of local image patches into discrete *visual codewords*, representing images by frequency counts of the codeword indices contained in them, and then learning classifiers based on these frequency histograms.

Quantizer design is typically viewed as an unsupervised task and the standard objective function is to minimize the expected distortion (i.e., squared Euclidean distance) between the original features and the respective codevectors [11]. However, in order to work well for the end goal of predicting a high-level category or attribute, the quantizer should be learned discriminatively. Generally speaking, the "ideal" discriminative quantizer is the one that retains *all* the information that is useful for predicting the attribute. Such a quantizer may be said to compute a *sufficient statistic* of the features for the attribute labels [5], [20]. Informally, for any statistical decision procedure about the attribute that uses the original features, we can find another one that performs just as well using the sufficient statistic.

This paper presents a novel method for learning codebooks for nearest-neighbor quantization such that the quantized representation of a feature approximates a sufficient statistic for its attribute label. The learning scheme is derived from information-theoretic properties of sufficient statistics [7], [20] and is based on minimizing the loss of information *about the attribute* that is incurred by the quantization operation (in general, quantization is compression, and some information will inevitably be lost). The objective function for information loss minimization involves both the feature vector positions and their class labels (and thus the quantizer must be trained in a supervised fashion, using labeled data), but the resulting nearest-neighbor codebook functions the same way as if it was produced by an unsupervised method such as k-means, and can be used to encode test data with unknown labels. Moreover, our training procedure also outputs the posterior distribution over class labels associated with each codevector. Thus, after encoding a new unlabeled feature to its nearest codevector, we can then use the learned class distribution for that codevector to predict the label of the original feature. Figure 1 schematically represents the sequence of processing in our method, where we go from the original continuous feature vector to its quantized representation, which in turn allows us to infer a class label.

The rest of the paper is organized as follows. Section II puts our method in the context of related work in the clustering and vector quantization literature. Section III first outlines the basics of information loss minimization and then presents our novel method for codebook construction together with the associated iterative minimization algorithm. Section IV shows a validation of our method on both synthetic and real data, and an application to producing effective codebooks for bag-of-features image classification. Section V gives a "bonus" application to segmenting images while using pixel attributes as supervisory information. This application, which is quite different from patch-based image classification, shows the versatility of our proposed technique, and points out interesting connections between information loss minimization and segmentation objectives based on minimum description

Svetlana Lazebnik is with the Department of Computer Science, University of North Carolina at Chapel Hill, NC, 27599.

Maxim Raginsky is with the Department of Electrical and Computer Engineering, Duke University, Durham, NC, 27708.

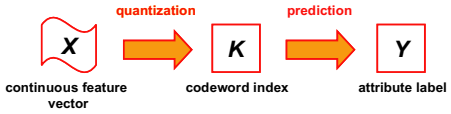


Fig. 1. Quantization for the sake of classification. X is a continuous random variable representing the features, K is a discrete random variable representing the index of the codevector nearest to X with respect to some codebook in the feature space, and Y is a discrete random variable representing the "class" or "attribute" of X that we want to predict. In an ideal case, K would be a sufficient statistic of X for Y, i.e., we would be able to predict Y based on K as well as we could have from X itself. However, some information is bound to be lost in going from the continuous feature space to the finite space of quantizer indices, and our goal is to learn a codebook that minimizes the loss of information *about* Y incurred by this operation.

length [15]. Finally, Section VI concludes the presentation with a summary of our contributions and an outline of possible future directions. We also include appendices interpreting information loss minimization in terms of lossless source coding [34] and proving generalization bounds on the performance of our empirical objective function. A preliminary version of this work has appeared in AISTATS 2007 [23].

II. Previous Work

The main concern of our paper is *empirical quantizer design* [11], [24]: given a representative training sequence drawn from the signal space, the goal is to learn a quantization rule that performs well not only on the specific training examples, but also on arbitrary, previously unseen test examples. In the field of quantizer design, there exist a number of approaches for using supervisory information to jointly learn quantizers and classifiers. Our work relies on a few techniques that are common in these approaches, in particular, forming Voronoi partitions of the feature space, but our goal is different in that we don't want to learn a classifier per se, but a quantized representation of the data that preserves the relevant information for a given classification task, regardless of the actual classifier used. Learning Vector Quantization [17], [18] is an early heuristic approach for supervised quantizer design using Voronoi partitions, based on self-organizing maps [19]. An approach more directly related to ours is Generalized Vector Quantization (GVQ) by Rao et al. [32]. GVQ is designed for regression type problems where the goal is to encode or estimate a random variable $Y \in \mathcal{Y}$ based on features $X \in \mathcal{X}$. This approach assumes a particular distortion or loss function on $\mathcal{Y}$ and uses expected distortion between the estimated and the actual values of Y on the training set as an objective function. The mapping is found by breaking up the space of $\mathcal{X}$ into Voronoi regions defined by a codebook in $\mathcal{X}$, and mapping each of these regions to a constant y. The codebook minimizing the objective function is learned using soft assignments and deterministic annealing. Inspired in part by GVQ, we also use soft Voronoi partitions to make the optimization problem more tractable. In other aspects, though, our approach is completely different. In GVQ the distortion measure on $\mathcal{Y}$ is assumed to be given a priori and the objective function is derived from this distortion measure. By contrast, we start from the statistical notion of sufficiency, which leads to the relative entropy as a natural distortion measure on the simplex of probability distributions over the attribute labels, and to information loss as the objective criterion. Another supervised quantizer design approach is the work of Oehler and Gray [29]. However, this work is tailored for use with Maximum A Posteriori (MAP) classification, whereas we use a much more general information-theoretic formulation which produces discriminative quantized representations that are (in principle) effective for any classifier or statistical model at the decision stage.

Quantizer design is conceptually related to the problem of *clustering*, or partitioning a discrete space of some entities into subsets, or clusters, such that the entities in the same cluster are in some sense "similar" to one another. Clustering is often used to train quantizers (i.e., k-means is a standard method for learning codebooks for nearest-neighbor quantization), but the majority of clustering methods make no distinction between "training" and "testing," and are not concerned with finding partitions that can be extended outside the original input set. There are several clustering methods motivated by an information-theoretic interpretation of sufficient statistics in terms of *mutual information* [10], [38], [40], [42]. The *information bottleneck* (IB) method [40], [42] is a general theoretical framework for clustering problems where the goal is to find a compressed representation K of the data X under the constraint on the mutual information $I(K;Y)$ between K and another random variable Y, which is correlated with X and is assumed to provide *relevant information* about X. For example, X is a word and Y is the topic of the document that contains it. The compressed representation K is found by minimizing an objective function of the form

$$I(X;K) - \beta I(K;Y) \tag{1}$$

over all randomized encodings of X into K. This objective function is motivated by rate-distortion theory [3] and seeks to trade off the number of bits needed to describe X via K and the number of bits of information K contains about Y (β is a variational parameter). In IB, in order to compute the encoding of X to K, one must have full knowledge of $P(y|x)$, the conditional distribution of Y given $X = x$. By contrast, the encoding rule learned by our method does not involve $P(y|x)$, and, in fact, can be used to predict this distribution for points outside the training set. Moreover, IB does not make any specific assumptions regarding the structure on $\mathcal{X}$, which can be either continuous or discrete, and can in principle result in highly complex partitions, while we admit only Voronoi partitions in order to make the operation of our classifier outside the training data as simple as possible.

A few other information-theoretic clustering algorithms [10], [38] have objective functions based on the general principles set forth by the information bottleneck framework. Our own approach is inspired in part by the divisive information-theoretic algorithm of Dhillon et al. [10], which is based on minimizing the information loss

$$I(X;Y) - I(K;Y)\,. \tag{2}$$

This objective function may be viewed as a special case of (1) where cluster assignment is deterministic and the number of clusters is fixed [10]. Eq. (2) can be interpreted as the difference between the amount of information provided by X about

Y and the amount of information provided by K about Y, and minimizing it leads to a clustering that throws away as little information about Y as possible.[1] Unfortunately, the algorithm of [10] is not suitable for our target application of quantizer design. For one, quantization requires an encoding rule that works on continuous data, does not depend on the labels other than those of the training examples, and can be applied outside the training set. In addition, the algorithm for learning the quantizer codebooks must simultaneously operate in two spaces, the vector space where the natural distance measure is Euclidean, and the space of attribute distributions that are naturally compared using the KL-divergence (relative entropy). Even though we also use information loss minimization as the objective criterion, our algorithm introduced in Section III-B significantly differs from that of [10] and successfully deals with the added challenges of quantizer design.

III. The Approach

We begin in Section III-A by giving a self-contained presentation of the empirical loss minimization framework of Dhillon et al. [10], who use it to design an iterative descent algorithm similar to k-means. This algorithm is suitable for clustering of discrete data, but not for our target problem of quantizer design. Nevertheless, it provides a starting point for our method of learning codebooks for nearest-neighbor encoding, which is developed in Section III-B. Finally, Section III-C discusses an optional modification to our objective function to trade off information loss and distortion in the feature space.

A. Background: Minimization of Empirical Information Loss

Consider a pair (X, Y) of jointly distributed random variables, where $X \in \mathcal{X}$ is a continuous *feature* and $Y \in \mathcal{Y}$ is a discrete *class label*. In the classification setting, given a training sequence $\{(X_i, Y_i)\}_{i=1}^N$ of i.i.d. samples drawn from the joint distribution of (X, Y), one typically seeks to minimize the probability of classification error $\Pr[\hat{Y}(X) \neq Y]$ over some family of classifiers $\hat{Y} : \mathcal{X} \to \mathcal{Y}$, such as k-nearest-neighbor classifiers, decision trees or support vector machines [14]. A more general approach is based on the notion of *sufficient statistics*. Informally, a sufficient statistic of X for Y contains as much information about Y as X itself. Hence an optimal hypothesis testing procedure operating on the sufficient statistic will perform as well as an optimal predictor of Y directly from X [5], [20]. This framework in principle allows us to learn compressed representations of X that retain as much discriminative power as possible without having to commit to any particular classifier.

We seek a partitioning of $\mathcal{X}$ into C disjoint subsets, such that the random variable $K \in \{1, \ldots, C\}$ giving the subset index of X would be a sufficient statistic of X for Y. By definition, a function K of X is a sufficient statistic for Y if X and Y are conditionally independent given K [5], [20]. In terms of *mutual information* [7], [20], this condition is equivalent to $I(K;Y) = I(X;Y)$. In general, going from the continuous data X to a quantized version K is bound to lose some discriminative information, so K cannot a be sufficient statistic in the strict mathematical sense. Instead, we would like to minimize the *information loss* $I(X;Y) - I(K;Y)$ over all partitions of $\mathcal{X}$ into C disjoint subsets. Because our interest here is in joint compression and classification of the features X, the number of partition elements C will, in general, be much larger than the number of class labels $|\mathcal{Y}|$.

Let P_x denote the conditional distribution $P(y|X = x)$ of Y given $X = x$, μ the marginal distribution of X, and $P = \int_{\mathcal{X}} P_x \, d\mu(x)$ the marginal distribution of Y. Then the mutual information between X and Y can be written as $I(X;Y) = \int_{\mathcal{X}} D(P_x \| P) d\mu(x)$, where $D(\cdot\|\cdot)$ is the relative entropy or the *Kullback–Leibler divergence* [20] given by $D(P_x\|P) = \sum_{y\in\mathcal{Y}} P_x(y) \log \frac{P_x(y)}{P(y)}$. Similarly, the mutual information between the two discrete variables K and Y can be written as $I(K;Y) = \sum_{k=1}^{C} P(k) D(P_k\|P)$, where P_k is the conditional distribution of Y given $K = k$.

Since the underlying distribution of (X, Y) is unknown, we have to minimize an *empirical* version of the loss for a finite training sequence $\{(X_i, Y_i)\}_{i=1}^N$. We can use the training sequence to approximate μ by the empirical distribution $\hat{\mu} = \frac{1}{N}\sum_{i=1}^N \delta_{X_i}$ and to estimate each P_{X_i} using any consistent nonparametric estimator, such as the k-nearest-neighbor rule $\hat{P}_{X_i} = \frac{1}{k}\sum_{j:X_j\in\mathcal{N}_k(X_i)} \delta_{Y_j}$, where $\mathcal{N}_k(X_i)$ is the set of k nearest neighbors of X_i (including X_i itself), or Parzen windows [31]. Also, let $\hat{P} = \frac{1}{N}\sum_{i=1}^N \hat{P}_{X_i}$ denote the corresponding estimate of P. Then the empirical version of the mutual information between X and Y is given by

$$\hat{I}(X;Y) = \frac{1}{N}\sum_{i=1}^{N} D(\hat{P}_{X_i}\|\hat{P})\,. \tag{3}$$

Now let $\mathcal{R}_1, \ldots, \mathcal{R}_C$ be a partitioning of the training set $\{X_i\}_{i=1}^N$ into C disjoint subsets, and define the map $K(X_i) = k$ if $X_i \in \mathcal{R}_k$. Then the empirical version of $I(K;Y)$ is

$$\hat{I}(K;Y) = \sum_{k=1}^{C} \frac{N_k}{N} D(\pi_k\|\hat{P})\,, \tag{4}$$

where $N_k = |\mathcal{R}_k|$ and

$$\pi_k = \frac{1}{N_k}\sum_{X_i\in\mathcal{R}_k} \hat{P}_{X_i}\,. \tag{5}$$

We can rewrite (3) as

$$\hat{I}(X;Y) = \sum_{k=1}^{C} \frac{N_k}{N} \sum_{X_i\in\mathcal{R}_k} \frac{1}{N_k} D(\hat{P}_{X_i}\|\hat{P}). \tag{6}$$

Now, using the definition of $D(\cdot\|\cdot)$ and (5), for each $k = 1, \ldots, C$ we can expand the second summation in (6) as

$$\begin{aligned}
&\frac{1}{N_k}\sum_{X_i\in\mathcal{R}_k}\sum_{y\in\mathcal{Y}} \hat{P}_{X_i}(y)\log\frac{\hat{P}_{X_i}(y)}{\hat{P}(y)} - \sum_{y\in\mathcal{Y}}\pi_k(y)\log\frac{\pi_k(y)}{\hat{P}(y)} \\
&= \frac{1}{N_k}\sum_{X_i\in\mathcal{R}_k}\sum_{y\in\mathcal{Y}} \hat{P}_{X_i}(y)\log\frac{\hat{P}_{X_i}(y)}{\pi_k(y)} \\
&= \frac{1}{N_k}\sum_{X_i\in\mathcal{R}_k} D(\hat{P}_{X_i}\|\pi_k).
\end{aligned}$$

[1] Note that $I(X;Y)$ is fixed, so minimizing (2) is equivalent to maximizing $I(K;Y)$, but we use (2) because it leads to a convenient computational solution in terms of iterative minimization [10].

Using this together with (6) and (4), we obtain the following expression for the empirical information loss:[2]

$$\hat{I}(X;Y) - \hat{I}(K;Y) = \frac{1}{N}\sum_{k=1}^{C}\sum_{X_i \in \mathcal{R}_k} D(\hat{P}_{X_i}\|\pi_k). \quad (7)$$

It is not hard to show, either directly or using the fact that the relative entropy $D(\cdot\|\cdot)$ as a Bregman divergence on the probability simplex $\mathcal{P}(\mathcal{Y})$ over $\mathcal{Y}$ [2], that

$$\pi_k = \arg\min_{\pi} \sum_{X_i \in \mathcal{R}_k} D(\hat{P}_{X_i}\|\pi), \qquad \forall k = 1, 2, \ldots, C \quad (8)$$

where the minimization is over all π in the interior of $\mathcal{P}(\mathcal{Y})$ (i.e., $\pi(y) > 0$ for all $y \in \mathcal{Y}$)[3]. π_k is the unique minimizer in (8), referred to as the *Bregman centroid* of $\mathcal{R}_k$ [2].

Given a disjoint partition $\mathcal{R}_1, \ldots, \mathcal{R}_C$ of $\{X_i\}_{i=1}^n$ and C probability distributions $q_1, \ldots, q_C$ in the interior of $\mathcal{P}(\mathcal{Y})$, define the objective function

$$\frac{1}{N}\sum_{k=1}^{C}\sum_{X_i \in \mathcal{R}_k} D(\hat{P}_{X_i}\|q_k). \quad (9)$$

From the preceding discussion we see that, for a fixed partition $\mathcal{R}_1, \ldots, \mathcal{R}_C$, this objective function is minimized by the choice $q_k = \pi_k$, $1 \le k \le C$. Moreover, it is not hard to show that, for fixed $q_1, \ldots, q_C$, the objective function is minimized by the partition

$$\mathcal{R}_k \triangleq \{X_i : D(\hat{P}_{X_i}\|q_k) \le D(\hat{P}_{X_i}\|q_j), j \ne k\}, k = 1, \ldots, C.$$

The optimization of (9) can therefore be performed by an iterative descent algorithm initialized by some choice of $\{\pi_k\} \subset \mathrm{Int}(\mathcal{P}(\mathcal{Y}))$, where each X_i is assigned to $\mathcal{R}_k$ with the smallest $D(\hat{P}_{X_i}\|\pi_k)$, and the class distribution centroids π_k are then recomputed by averaging the $\hat{P}_{X_i}$'s over each $\mathcal{R}_k$, as in (5). This descent algorithm has the same structure as the well-known k-means algorithm; in fact, as pointed out in [2], both of them are special cases of a general descent algorithm for minimizing the Bregman information loss with respect to a suitable Bregman divergence, which is given by the squared Euclidean distance in the case of k-means.

A top-down "divisive" version of the above algorithm has been used by [10] to cluster words for text document classification. However, this algorithm is unsuited for our goal of quantizer design for several reasons. First, it produces an arbitrary partition of the discrete input set without any regard to spatial coherence, whereas we need a method that takes advantage of the continuous structure of the feature space. Second, data points are assigned to clusters by nearest-neighbor with respect to KL-divergence between P_X and π_k, which requires the knowledge of P_X and thus cannot be extended to unlabeled test points. By contrast, we actually want the quantized representation of X to help us estimate the attribute distribution for previously unseen and unlabeled points, which means that at the very least, the encoding must not depend on P_X. As discussed in the next section, we propose to resolve these difficulties by placing structural constraints on the partitions and the encoding rule.

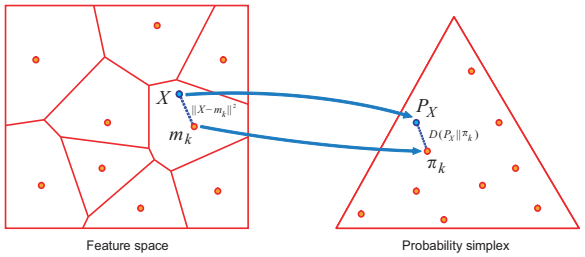


Fig. 2. Schematic illustration of our optimization problem. Encoding takes place in the feature space using the nearest-neighbor rule w.r.t. Euclidean distance, but the value of the objective function is computed in the probability space using KL-divergence. The goal is to position the codevectors m_k in the feature space to minimize the value of the objective function.

Note: in the subsequent sections, which are concerned with deriving practical algorithms for empirical loss minimization, we will be dealing only with the estimates of P_{X_i}. For notational simplicity we shall drop the hats from these quantities.

B. Constraining the Encoder

In the setting of this paper, we assume that the data X comes from a compact subset $\mathcal{X}$ of the Euclidean space $\mathbb{R}^d$. We need a way to specify a partitioning of $\mathcal{X}$ that can be extended from the training set to the whole feature space, such that the encoding rule does not depend on the attribute distribution of a given point. We can obtain a suitable scheme by considering *Voronoi partitions* of $\mathcal{X}$ with respect to a codebook of C centers or prototypes in the feature space. Now the quantizer design problem can be phrased as follows: we seek a codebook $\mathcal{M}^* = \{m_1^*, \ldots, m_C^*\} \subset \mathcal{X}$ and a set of associated posterior class distributions $\Pi^* = \{\pi_1^*, \ldots, \pi_C^*\} \subset \mathrm{Int}(\mathcal{P}(\mathcal{Y}))$ that jointly minimize the empirical information loss[4]

$$\sum_{k=1}^{C}\sum_{X_i \in \mathcal{R}(m_k)} D(P_{X_i}\|\pi_k), \quad (10)$$

where $\mathcal{R}(m_k) \triangleq \{x \in \mathcal{X} : \|x - m_k\| \le \|x - m_j\|, \forall j \ne k\}$ is the Voronoi cell of m_k. Figure 2 schematically illustrates the structure of this optimization problem, involving two corresponding codebooks in the feature space and in the simplex of posterior class probabilities. The resulting encoding rule is very simple: the partition (quantizer) index of a point $X \in \mathcal{X}$ is the index of its nearest codevector $m_k \in \mathcal{M}^*$. Note that this rule does not involve the (possibly unknown) label of X, and is thus suitable for encoding unlabeled data. Moreover, nearest-neighbor quantization naturally leads to classification: having mapped X onto its partition index k, we can predict

[2]Banerjee et al. [2] give a more general derivation of this expression for empirical information loss in the context of clustering with *Bregman divergences* [6], of which the relative entropy is a special case; the derivation included in our paper is meant to make it self-contained.

[3]The requirement that π lie in the interior of the probability simplex is a technical condition dictated by the properties of Bregman divergences [2]. It is not a restrictive condition in practice, since one can always perturb π by a small amount to force it into the interior.

[4]At this point, it is appropriate to ask about the generatization performance of a minimizer of this empirical objective function: does it give us a good approximation to a minimizer of the actual information loss for the underlying distribution over all choices of $(\mathcal{M}, \Pi)$? It is possible to show that, under a mild regularity condition on the allowed codebooks in the probability simplex over the class labels, an empirical minimizer of (10) minimizes the actual information loss $I(X;Y) - I(K;Y)$ over all Voronoi partitions of $\mathcal{X}$ with C cells, with high probability. Appendix B contains a sketch of the proof.

the label $\hat{Y}$ of X by the maximum a posteriori probability (MAP) criterion:

$$\hat{Y} = \arg\max_{y \in \mathcal{Y}} \pi_k(y)\,. \tag{11}$$

In practice, the performance of this simple decision rule will be bounded from above by a local classifier that makes MAP decisions using the "uncompressed" probabilities P_X estimated from the training data. This is not surprising, since we are effectively replacing the full training set $\{X_i\}_{i=1}^N$ by a much smaller set of codebook entries $\{m_k\}_{k=1}^C$ and the local probability estimates P_X by the quantized estimates π_k. Of course, even then, our quantization procedure may realize a significant savings in terms of search time and space complexity for the purpose of nearest-neighbor classification. In the experiments of Section IV-A, we use the MAP classification rule (11) because of its simplicity. However, in Section IV-B we also show that the applicability of our learned codebooks goes beyond simple nearest-neighbor classification of individual features. For example, the codebooks may be used to combine multiple features to make aggregate decisions, e.g., assigning a single class label to a collection of features representing an entire image.

To summarize, the objective function defined by (10) is a big improvement over (9), since it gives us a simple encoding rule that extends to unlabeled data. Unfortunately, it is still unsatisfactory for computational reasons: while the optimal choice of $\Pi = \{\pi_k\}$ for a given $\mathcal{M} = \{m_k\}$ is given by (5), optimizing the codebook $\mathcal{M}$ for a given Π is a difficult combinatorial problem. Therefore we opt for a suboptimal design procedure suitable for designing vector quantizers with structurally constrained encoders [11], [32]. Namely, we introduce a differentiable relaxation of the objective function by allowing "soft" partitions of the feature space. Let $w_k(x)$ denote the "weight" of assignment of a point $x \in \mathcal{X}$ to $\mathcal{R}(m_k)$, with $\sum_{k=1}^C w_k(x) = 1$. As suggested by Rao et al. [32], a natural choice for these weights is the Gibbs distribution

$$w_k(x) = \frac{e^{-\beta\|x-m_k\|^2/2}}{\sum_j e^{-\beta\|x-m_j\|^2/2}}, \tag{12}$$

where $\beta > 0$ is the parameter that controls the "fuzziness" of the assignments, such that smaller β's correspond to softer cluster assignments, and the limit of infinite β yields hard clustering. While in principle it is possible to use annealing techniques to pass to the limit of infinite β [36], we have found that a fixed value of β works well in practice (our method for selecting this value in the experiments will be discussed in Section IV). Note also that we deliberately avoid any probabilistic interpretation of (12) even though it has the form of the posterior probability for a Gaussian distribution with $\beta = \frac{1}{\sigma^2}$. As will be further discussed in Section IV-A, ours is not a generative approach and does not assume any specific probabilistic model underlying the data.

We are now ready to write down the relaxed form of our objective function for a *fixed* β:

$$E(\mathcal{M}, \Pi) = \sum_{i=1}^N \sum_{k=1}^C w_k(X_i) D(P_{X_i} \| \pi_k)\,. \tag{13}$$

A local optimum of this function can be found via alternating minimization, where we first hold Π fixed and update $\mathcal{M}$ to reduce the objective function, and then hold $\mathcal{M}$ fixed and update Π. For a fixed $\Pi = \{\pi_k\}$, $E(\mathcal{M}, \Pi)$ is reduced by gradient descent over the m_k's. The update for each m_k has the form

$$m_k^{(t+1)} = m_k^{(t)} - \alpha \sum_{i=1}^N \sum_{j=1}^C D(P_{X_i} \| \pi_j^{(t)}) \frac{\partial w_j^{(t)}(X_i)}{\partial m_k^{(t)}}, \tag{14}$$

where $\alpha > 0$ is the *learning rate* shared by all the centers and found using line search [4], and

$$\frac{\partial w_j(x)}{\partial m_k} = \beta[\delta_{jk} w_k(x) - w_k(x) w_j(x)](x - m_k) \tag{15}$$

where δ_{jk} is 1 if $j = k$ and 0 otherwise. For a fixed codebook $\mathcal{M}$, the minimization over Π is accomplished in closed form by setting the derivatives of the Lagrangian $E(\mathcal{M}, \Pi) + \sum_k \lambda_k \sum_y \pi_k(y)$ w.r.t. $\pi_k(y)$ to zero for all k and all $y \in \mathcal{Y}$ and solving for $\pi_k(y)$ and for the Lagrange multipliers λ_k. The resulting update is

$$\pi_k^{(t+1)}(y) = \frac{\sum_{i=1}^N w_k^{(t+1)}(X_i) P_{X_i}(y)}{\sum_{y'} \sum_{i=1}^N w_k^{(t+1)}(X_i) P_{X_i}(y')}, \ \forall y \in \mathcal{Y}. \tag{16}$$

The two updates (14) and (16) are alternated for a fixed number of iterations or until the reduction in the value of the objective function falls below a specified threshold (this is guaranteed to happen in a finite number of iterations, because the sequence of objective function values produced by the updates is monotonically decreasing and bounded from below by 0, and therefore has a limit). This alternating minimization can then be embedded in an annealing procedure if a better approximation to the original combinatorial objective function is sought.

C. Trading off Information Loss and Distortion

The loss minimization approach presented in the previous section does not pay any attention to the distortion $\|X - m_k\|^2$ incurred by encoding some data point to its nearest centroid. In practice, the regions produced by the above optimization procedure may be arbitrarily large or elongated, as some centroids either come too closely together or migrate far outside the smallest convex polytope enclosing the training set. However, for problems that combine the objectives of faithful compression with accurate classification, it is desirable to avoid such artifacts and to make sure that the codebook represents the data with relatively low distortion. To help meet this objective, we propose in this section an optional variant of our basic objective function (13) to trade off information loss and mean squared distortion in a Lagrangian formulation:

$$\tilde{E}(\mathcal{M}, \Pi) = E(\mathcal{M}, \Pi) + \lambda F(\mathcal{M}, \Pi)\,, \tag{17}$$

where λ is a tradeoff parameter, and

$$F(\mathcal{M}, \Pi) = \sum_{i=1}^N \sum_{k=1}^C w_k(X_i) \|X_i - m_k\|^2 \tag{18}$$

is the standard distortion function for soft clustering [11]. An analogous Lagrangian approach has been used by Oehler and

Gray [29] for joint compression and classification of images, where the objective function is a sum of a Bayes weighted risk term and a mean squared error term. The updates for the m_k are given by

$$\begin{aligned} m_k^{(t+1)} &= m_k^{(t)} - \alpha \sum_{i=1}^{N}\sum_{j=1}^{C} D(P_{X_i}\|\pi_j^{(t)}) \frac{\partial w_j^{(t)}(X_i)}{\partial m_k^{(t)}} \\ &\quad -\alpha\lambda \sum_{i=1}^{N}\sum_{j=1}^{C} \left[\|X_i - m_j\|^2 \frac{\partial w_j^{(t)}(X_i)}{\partial m_k^{(t)}} \right. \\ &\quad \left. +2\delta_{jk} w_j^{(t)}(X_i)(m_j - X_i) \right]. \end{aligned}$$

The updates for π_k are given by (16) as before.

The behavior of the modified objective function (17) is demonstrated experimentally in Figure 6 in Section IV-A; in all the other experiments we stick with the original objective function (13).

IV. Experimental Evaluation

This section presents an experimental evaluation on several synthetic and real datasets. Section IV-A validates the basic behaviour of our approach using nearest-neighbor classification of quantized features as a sample task, and Section IV-B applies our framework to the task of bag-of-features image classification.

A. Synthetic and Real Data

Table 3 is a summary of the datasets used in the experiments of this section. For each dataset, the table lists the average performance of a k-nearest-neighbor classifier trained on random subsets consisting of half the samples. We use a "nominal" value of $k = 10$, which worked well for all our experiments. Recall from Section III-B that the performance of a k-nearest-neighbor classifier is an effective upper bound on the performance of MAP classification with our codebook using eq. (11). For the three synthetic datasets, the table also lists the theoretically computed optimal Bayes upper bound. Note that for these datasets, the performance of the 10-nearest-neighbor (10NN) classifier comes quite close to the Bayes bound.

The two main implementation issues for our method are estimation of posterior probabilities P_{X_i} and the choice of the "softness" constant β. For all the experiments described in this section, we estimate P_{X_i} by averaging the point masses associated with the labels of the ten nearest neighbors of X_i and its own label Y_i, but we have also found the point mass estimate $P_{X_i} = \delta_{Y_i}$ to produce very similar performance. We set β to $\frac{d}{\hat{\sigma}^2}$, where d is the dimensionality of the data and $\hat{\sigma}^2$ is the mean squared error of the k-means clustering that we use to initialize the loss minimization procedure.

A good "floor" or a baseline for our method is provided by standard k-means quantization, where the data centers $m_1, \ldots m_C$ are learned without taking class labels into account, and the posterior distributions $P(y|k) = \pi_k$ are obtained afterwards by the averaging rule (5). As an alternative baseline that does take advantage of class information for learning the data centers, but does not directly minimize information loss, we chose a generative framework where each class conditional density $P(x|y)$ is modeled as a mixture of C Gaussians, and mixture components are shared between all the classes:

$$P(x|y) = \sum_{k=1}^{C} P(x|k)\, P(k|y)\,. \tag{19}$$

$P(x|k)$ is a Gaussian with mean m_k and a spherical covariance matrix σ^2I, $\sigma^2 = \frac{1}{\beta}$. The parameters of this model, i.e., the means m_k and the class-specific mixture weights $P(k|y)$, are learned using the EM algorithm [4]. (Alternatively, one could use GMVQ, a hard clustering algorithm for Gauss mixture modeling [1].) Instead of fixing a global value of σ^2, we also experimented with including the variances σ_k^2 as parameters in the optimization, but this had little effect on classification performance, or even resulted in overfitting for the more high-dimensional datasets.

Figure 4 shows results on three two-dimensional two-class synthetic datasets. Part (a) shows the centers and partitions produced by k-means and used to initialize both EM and info-loss optimizations. Part (b) shows the resulting info-loss partitions. In all three cases, our method partitions the data space in such a way as to separate the two classes as much as possible. For example, the "concentric" dataset (left column) consists of uniformly sampled points, such that the "red" class is contained inside a circle and the "blue" class forms a ring around it. The regions produced by k-means do not respect the circular class boundary, whereas the regions produced by the info-loss method conform to it quite well. It is important to keep in mind, however, that separating classes is not the primary goal of information loss minimization. Instead, the criterion given by (13) is more general, seeking to partition the data into regions where the posterior distributions P_{X_i} of the individual data points are as homogeneous as possible, measured in terms of their similarity to the "prototype" distribution π_k. When the classes in the dataset are separable, this criterion naturally leads to regions whose prototype distributions are nearly "pure," i.e., dominated by a single class.

Figure 4 (c) compares the classification performance of the three clustering methods. For k-means and info-loss, MAP classification is performed using Eq. (11) while for EM, it is derived from the probabilistic model (19). For the "concentric" dataset, the info-loss classification rate falls somewhat as the codebook size increases from 16 to 128. This is because the decision regions in this case are simple enough to be approximated well even with $C = 8$, and increasing C causes the method to overfit. Finally, Figure 4 (d) compares the performance of the three methods w.r.t. minimizing information loss or equivalently, maximizing the mutual information $I(K;Y)$ between the region index and the class label. Again, info-loss outperforms both k-means and EM.

Figure 5 shows analogous results for the three real datasets in our study. As in Figure 4, info-loss outperforms the two baseline methods. Recall from Table 3 that these datasets have as many as 11 classes and 256 dimensions, so our method appears to scale quite well as the number of classes

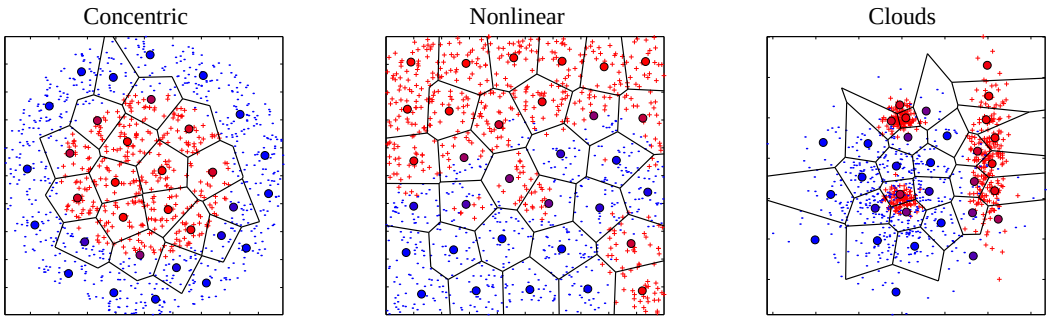
Dataset		# classes	# samples	dim.	10NN rate	Bayes rate
Concentric[1]	(synthetic)	2	2,500	2	98.01 ± 0.44	100
Nonlinear	(synthetic)	2	10,000	2	95.65 ± 0.19	96.32
Clouds[1]	(synthetic)	2	5,000	2	88.32 ± 0.43	90.33
Texture[1]	(real)	11	5,500	40	97.35 ± 0.27	-
Satimage[2]	(real)	6	6,435	36	89.18 ± 0.45	-
USPS[3]	(real)	10	9,298	256	94.46 ± 0.29	-

[1] http://www.dice.ucl.ac.be/neural-nets/Research/Projects/ELENA/databases/

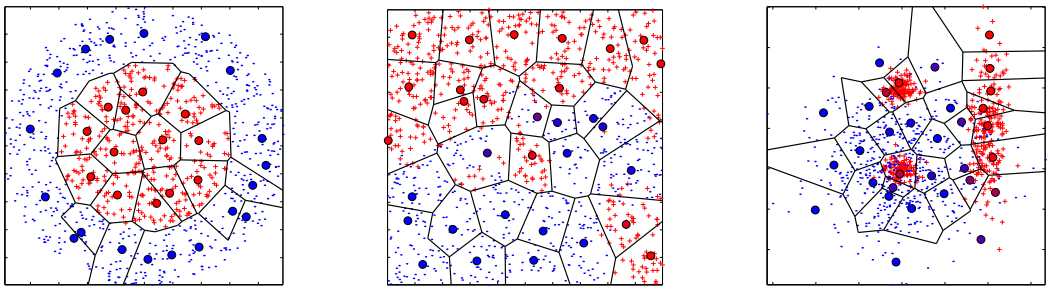
[2] http://www.ics.uci.edu/~mlearn/MLSummary.html

[3] ftp://ftp.kyb.tuebingen.mpg.de/pub/bs/data/

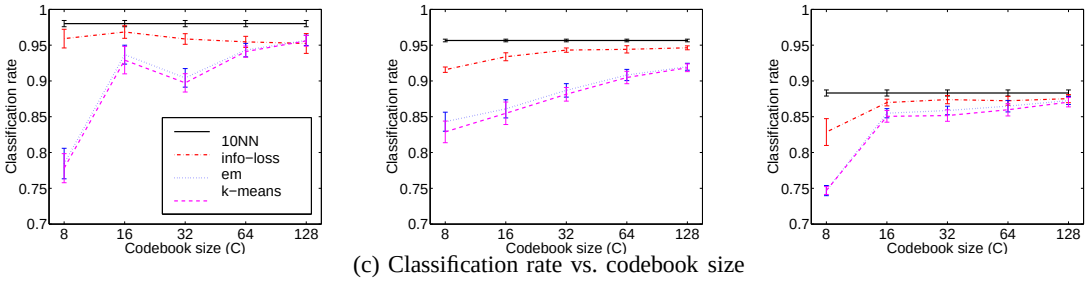
Fig. 3. Summary of the datasets used in our experiments. The *nonlinear* dataset was generated by us, and the rest were downloaded from the corresponding URLs. *Texture* contains features of small image patches taken from 11 classes from the Brodatz album. *Satimage* is Landsat satellite measurements for 6 classes of soil. The features in *USPS* are grayscale pixel values for 16×16 images of 10 digits from postal envelopes.



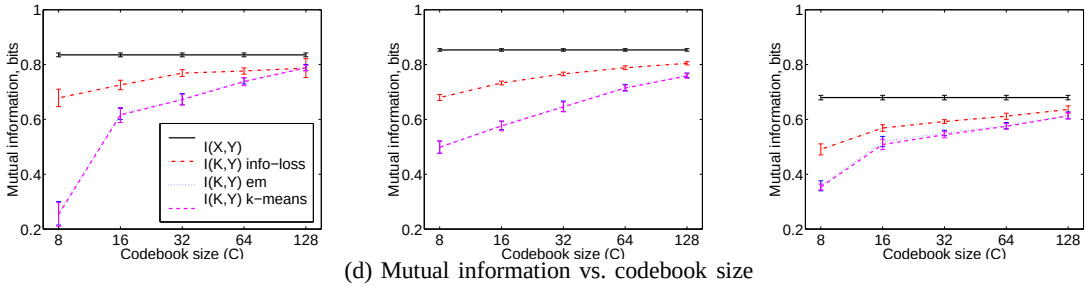
(a) Centers and partitions produced by k-means ($C = 32$)



(b) Centers and partitions produced by our info-loss method ($C = 32$)



(c) Classification rate vs. codebook size



(d) Mutual information vs. codebook size

Fig. 4. Results on synthetic data (best viewed in color). For (b), our method is initialized with the cluster centers produced by k-means in (a). For (c) and (d), we have performed 10 runs with different random subsets of half the samples used to train the models and the rest used as test data for reporting classification accuracy and mutual information. The height of the error bars is twice the standard deviation for each measurement. In (d), information loss is given by the vertical distance between $I(X;Y)$ and $I(K;Y)$.

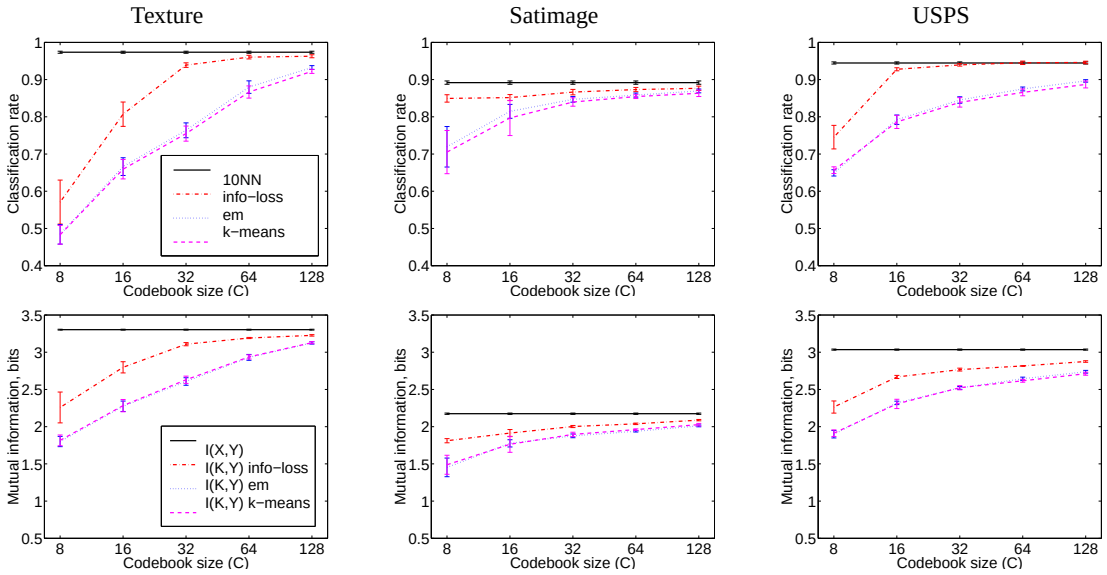


Fig. 5. Results for three real image datasets. First row: classification rate vs. codebook size. Second row: mutual information vs. codebook size. As in Figure 4, means and standard deviations are reported over 10 runs with half the dataset randomly selected for training and half for testing.

and the dimensionality of the feature space increase. It is worth noting that in all our experiments, EM achieves only a small improvement over k-means, so it seems to be relatively ineffective as a way of incorporating class information into clustering. This weakness may be due to the fact that the generative model (19) encodes a strong relationship between the density of the data and its class structure. By contrast, our info-loss framework is much more flexible, because it makes minimal assumptions about the data density, approximating it by the empirical distribution, and does not require any correspondence between the modes of this density and the posterior class distribution. As far as our method is concerned, the data can be generated using one process, such as a mixture of Gaussians, and the class distribution can be "painted on" by a completely different process.

Finally, Figure 6 demonstrates the tradeoff between quantizer distortion and information loss for the Lagrangian objective function (17) of Section III-C. We can see that for the *texture* dataset, it is possible to achieve "the best of both worlds": for intermediate values of the tradeoff parameter (i.e., $\lambda = 1$), classification accuracy is not significantly affected, while the mean squared Euclidean distortion in the feature space is almost as low as for the pure k-means algorithm.

B. Constructing Codebooks for Bag-of-Features Image Classification

Section IV-A has used classification as an example task to validate the basic behavior of our information loss minimization framework. However, it is important to emphasize that learning stand-alone classifiers is *not* our primary intended goal. Instead, we propose information loss minimization as a method for producing discriminative quantized representations of continuous data that can be incorporated into more complex statistical models. Such models may not even be aimed at classifying the individual features directly, but at combining them into higher-level representations, e.g., combining multiple phonemes to form an utterance or multiple local image patches to form a global image model. Accordingly, we demonstrate in this section the use of our method to build effective discrete visual vocabularies for *bag-of-features* image classification [8], [39], [43]. Analogously to *bag-of-words* document classification [37], this framework represents images by histograms of discrete indices of the "visual words" contained in them. Despite the extreme simplicity of this model — in particular, its lack of information about the spatial layout of the patches — it is currently one of the leading state-of-the-art approaches to image classification [43]. The performance of bag-of-features methods depends in a fundamental way on the visual vocabulary or codebook that is used to quantize the image features into descrete visual words. In recent literature, the problem of effective design of these codebooks has been gaining increasing attention (see, e.g., [21], [27] and references therein).

Figure 7 shows the dataset that we use to investigate the performance of our quantization method for forming bag-of-features representations. This dataset consists of 4485 images taken from fifteen different scene categories, and is quite challenging — for example, it is difficult to distinguish indoor categories such as bedroom and living room. This dataset has been used by Lazebnik et al. [22], who report a bag-of-features classification rate of 72.2% with a k-means vocabulary of size 200 and training sets consisting of 100 images per class.[5] In the present experiments, we follow the setup of [22] for feature extraction and training. Namely, the image features are 128-dimensional SIFT descriptors [25] of 16×16 patches sampled on a regular 8×8 grid. Let us underscore that classifying individual image patches or features is *not* our goal in this section. In fact, this task is quite difficult because small image windows are inherently ambiguous. For example, a uniform white patch may belong to a cloud in any outdoor class, or to a white wall in an indoor class. Not surprisingly, the 10NN classification rate for the individual image features in

[5]Another reference performance figure for a 13-class subset of this dataset is 65.2% by Fei-Fei and Perona [13].

λ	distortion	info. loss	class. rate
0	0.424 ± 0.04	0.282 ± 0.04	94.0 ± 1.1
0.1	0.386 ± 0.02	0.273 ± 0.03	94.5 ± 0.8
0.5	0.276 ± 0.02	0.329 ± 0.07	92.7 ± 2.6
1.0	0.247 ± 0.01	0.375 ± 0.04	90.7 ± 2.3
5.0	0.201 ± 0.01	0.479 ± 0.08	87.0 ± 3.1
10.0	0.192 ± 0.01	0.561 ± 0.06	84.2 ± 2.2
∞	0.184 ± 0.01	0.705 ± 0.05	75.6 ± 1.9

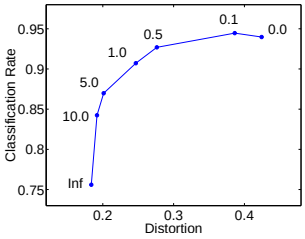


Fig. 6. Trading off quantizer distortion and information loss for the *texture* dataset with $C = 32$. Left: mean squared distortion, information loss, and classification rate as a function of λ, where $\lambda = 0$ corresponds to pure info-loss clustering and $\lambda = \infty$ corresponds to k-means. Right: classification error plotted as a function of distortion. The values of λ corresponding to each data point are shown on the plot.

Fig. 7. Example images from the scene category database. The starred categories originate from Oliva and Torralba [30]. The entire dataset is publicly available at http://www-cvr.ai.uiuc.edu/ponce_grp/data.

	$C = 32$	$C = 64$	$C = 128$	$C = 256$
NB k-means	56.2 ± 0.9	60.3 ± 1.7	62.7 ± 0.7	64.9 ± 0.4
NB info-loss	60.8 ± 0.9	62.9 ± 1.4	64.8 ± 0.8	66.6 ± 0.7
SVM k-means	59.5 ± 0.6	65.8 ± 0.5	70.4 ± 0.8	73.3 ± 0.3
SVM info-loss	63.9 ± 0.4	68.0 ± 0.5	71.6 ± 0.7	74.7 ± 0.4

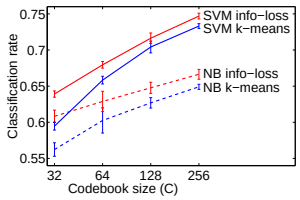


Fig. 8. Performance of bag-of-features classification for four different dictionary sizes (32, 64, 128, 256), two different methods of dictionary formation (k-means, info-loss), and two different classification methods (NB, SVM). For ease of visualization, the plot on the right reproduces the same information as the table on the left. The results are averaged over five runs with different random training/test splits. On the plot, the height of each error bar is twice the corresponding standard deviation.

this dataset is only 16%. However, even though a single small patch has only a limited predictive power about the class of the image that it comes from, a "signature" vector of frequency counts of such patches over an entire image contains a lot more information.

To create a bag-of-features representation, we first form a visual codebook or vocabulary by running either k-means or our info-loss algorithm on $22,500$ patches randomly sampled from all the classes in the training set, which constists of 100 images per class. For the info-loss algorithm, each training patch is given the class label of the image that it was extracted from. Finally, we encode the patches in each image I into the index of its closest codebook center or "vocabulary word," and represent the image as a vector of frequency counts $N_k(I)$ of each index k. Figure 8 shows results of classifying histograms based on the two types of codebooks. We use two different classifiers, Naive Bayes (NB) and support vector machines (SVM). Naive Bayes performs maximum likelihood

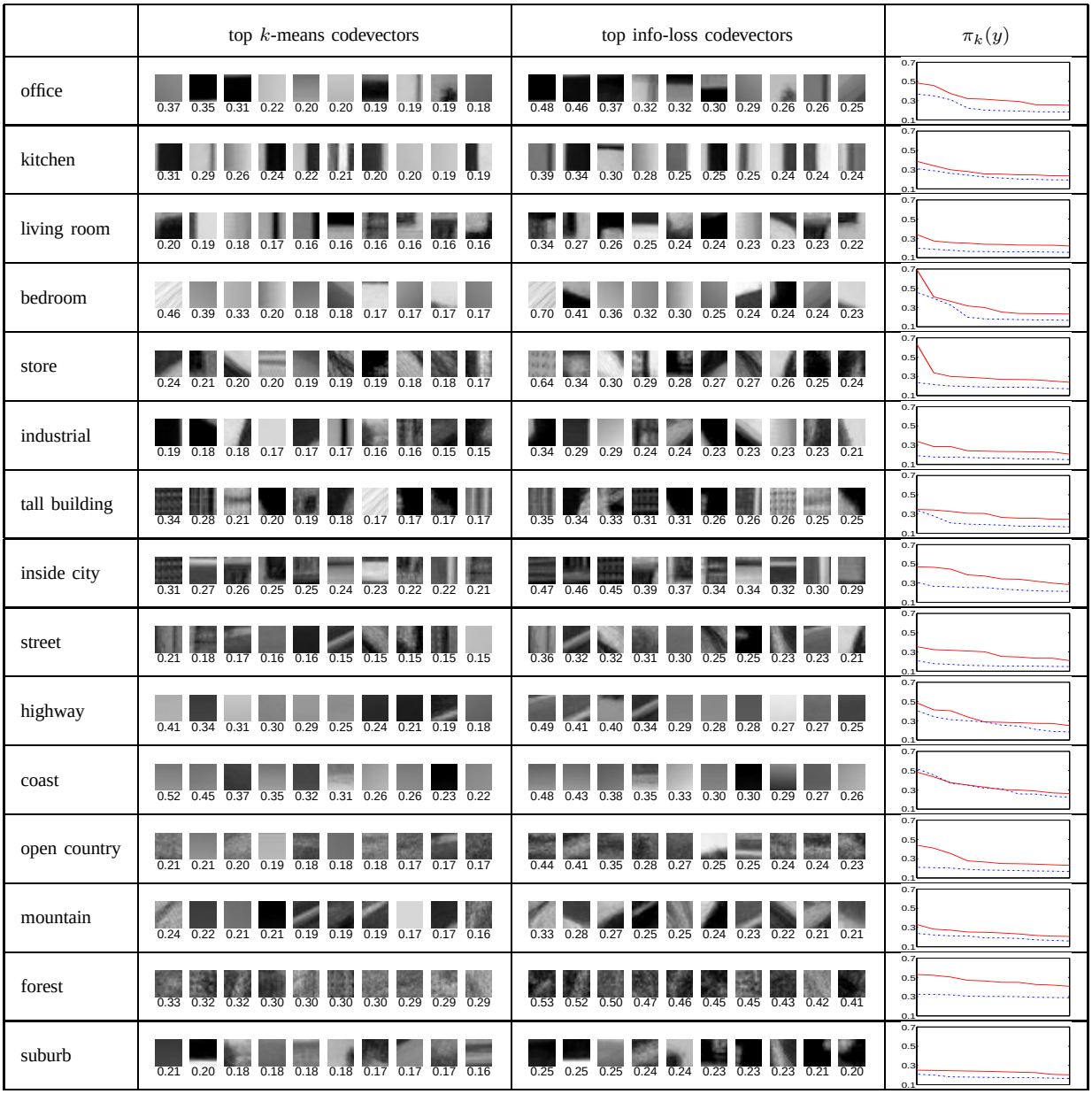


Fig. 9. Left and middle columns: top ten k-means and info-loss codewords for each class, with the corresponding posterior class probabilities indicated below. Right column: plots of the top ten probabilities for both codebooks shown on the same axes for easy comparison. The solid red line is for the info-loss codebook, and the dash-dot blue line is for the k-means codebook.

classification according to the *multinomial event model* [26]:

$$P(I|y) = \prod_k P(k|y)^{N_k(I)}\,, \tag{20}$$

where in the case of a codebook output by our method, $P(k|y)$ is obtained directly by Bayes rule from the centroid π_k. For support vector machines, we use the histogram intersection kernel [28], [41] defined by

$$\mathcal{K}\left(N(I_1), N(I_2)\right) = \sum_{k=1}^{C} \min\left(N_k(I_1), N_k(I_2)\right)\,.$$

As seen from Figure 8, codebooks produced by our method yield an improvement over k-means, which, though not large in absolute terms (2% to 4%) is consistent and statistically significant, given the extremely small variation of classification rates over multiple runs. Moreover, the improvement is higher for smaller vocabulary sizes and for Naive Bayes, which is a weaker classification method that relies more directly on the quality of the probability estimates output by the quantizer. As a caveat, we should note that this figure only considers codebook sizes up to 256, at which point the performance of both the info-loss and the k-means codebooks continues to increase, and, in fact, the k-means codebook shows a trend of "catching up" to the info-loss one. This is not surprising: once the compression rate becomes high enough, both the info-loss and the k-means codebooks will have sufficiently many codevectors to capture all the relevant class information. However, the info-loss codebook has the potential of achieving similar levels of classification performance at much lower rates than standard k-means, which can result in considerable computational savings in practice.

Figure 9 shows a more detailed visualization of the code-

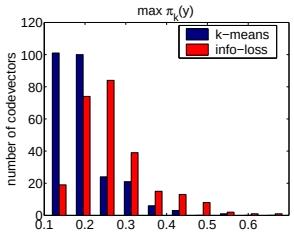


Fig. 10. Histograms of $\max_y \pi_k(y)$, or the maximum posterior probability of observing any class given a codeword k.

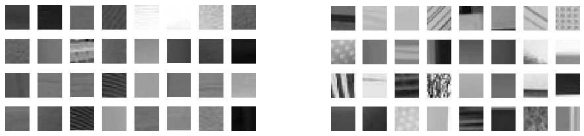


Fig. 11. Codebooks for $C = 32$.

books produced by k-means and information loss minimization for vocabulary size 256. This figure shows the top ten codewords for each class, i.e., the codewords with the highest posterior probabilities $\pi_k(y)$ for a given class y. This probability value is shown below each codeword. The leftmost column shows a plot of these values for both k-means and info-loss codebook on the same axis for easier comparison. We can see that the posterior probabilities for the info-loss keywords tend to be higher than those for the k-means codebooks. Intuitively, k-means codewords are more "mixed" and info-loss codewords are more "pure," as we have observed earlier in the synthetic examples of Section IV-A, Figure 4. The increased "purity" of info-loss codewords is also reflected in Figure 10, which shows histograms of maximum posterior probability values for the two codebooks (a similar plot was used by Larlus and Jurie [21] to demonstrate the effectiveness of their *latent mixture vocabularies* for object classification).

We can also observe the improved quality of the info-loss codebook by examining the appearance of individual codewords. For example, top "mountain" codewords for the k-means codebook include some generic uniform patches, while all the top info-loss codewords have diagonal edges that are very characteristic of mountain slopes. Note, however, that the discriminativeness of a given codeword depends primarily not on its appearance, but on the shape of its Voronoi cell, which is jointly determined by the positioning of multiple codewords. Thus, for example, the top "bedroom" codeword for both codebooks has a very similar appearance, but the posterior probability of "bedroom" for the k-means codeword is only 0.46, whereas for the info-loss codeword it is 0.70. This said, there does exist a perceptual difference in the two types of codebooks, and it is especially apparent for small codebook sizes, as seen in Figure 11 for $C = 32$. The info-loss codebook tends to contain more high-contrast patches with salient edges or texture patterns. Intuitively, such patterns are more informative about the image category than more generic, low-contrast patches that make up the standard k-means codebook.

V. Bonus Application: Image Segmentation

This section sketches an additional application of our approach to image segmentation. This application, which (at least, on the surface) seems significantly different from patch-based image classification, illustrates the potentially broad applicability of the information loss minimization framework. Moreover, it serves to point out interesting theoretical connections between information loss minimization and recently introduced objective functions for segmentation [15] that are motivated by the minimum description length principle [34].

In the segmentation setting, the feature space $\mathcal{X}$ is the space of two-dimensional coordinates of all the pixels in an image, and the label space $\mathcal{Y}$ consists of discretized appearance attributes such as intensity, color, or texture. The interpretation of the objective function (13) for segmentation is as follows: we seek a Voronoi partitioning of the image induced by a set of two-dimensional centers $\mathcal{M} = \{m_1, \ldots, m_C\}$, such that if m_k is the center closest to some pixel X (i.e., X falls into the kth Voronoi cell), then the local appearance distribution P_X in the neighborhood of X is predicted as well as possible by the appearance "centroid" π_k associated with m_k. Note that in image segmentation, there is no distinct testing regime, i.e., no image pixels with unknown appearance attributes. Instead, we are interested in compressing the known attributes of all the image pixels using a much smaller set of appearance centroids. Note that the Voronoi regions into which our procedure partitions the image can be thought of as *superpixels* [33], or coherent and relatively homogeneous units of image description.

In our implementation, the appearance attribute or label Y of each pixel X is its color or grayscale value discretized to 100 levels (for color images, minimum variance quantization is used[6], and for grayscale images, uniform quantization is used). Next, we obtain the label distribution P_X by taking a histogram of the labels Y over the 3×3 pixel neighborhood centered at X. The Voronoi centers are initialized on a regular 20×20 grid, resulting in a codebook of size $C = 400$, and the optimization is run for 50 iterations (even though the objective function continues to decrease slowly during further iterations, this does not produce any significant perceptual improvement).

Figure 12 shows results of our segmentation algorithm applied to six different images. Overall, the results are very intuitive, with the centers arranging themselves to partition each image into approximately uniform regions. There are occasional artifacts, such as in the third image from the top, where the cell boundaries do a poor job of following the smooth curve of the top of the dog's head. Such artifacts are due to the optimization process getting trapped in local minima, or to the difficulty of fitting curved image edges with piecewise-linear segmentation boundaries. They can be alleviated relatively easily through the use of an adaptive or hierarchical framework, which would work by introducing additional centers into regions where the value of the error function is above an acceptable threshold. For the sake of this paper, however, our results are not meant to compete

[6] In Matlab, this is accomplished by the command `rgb2ind(I,N)` where `I` is the image and `N` is the number of color levels.

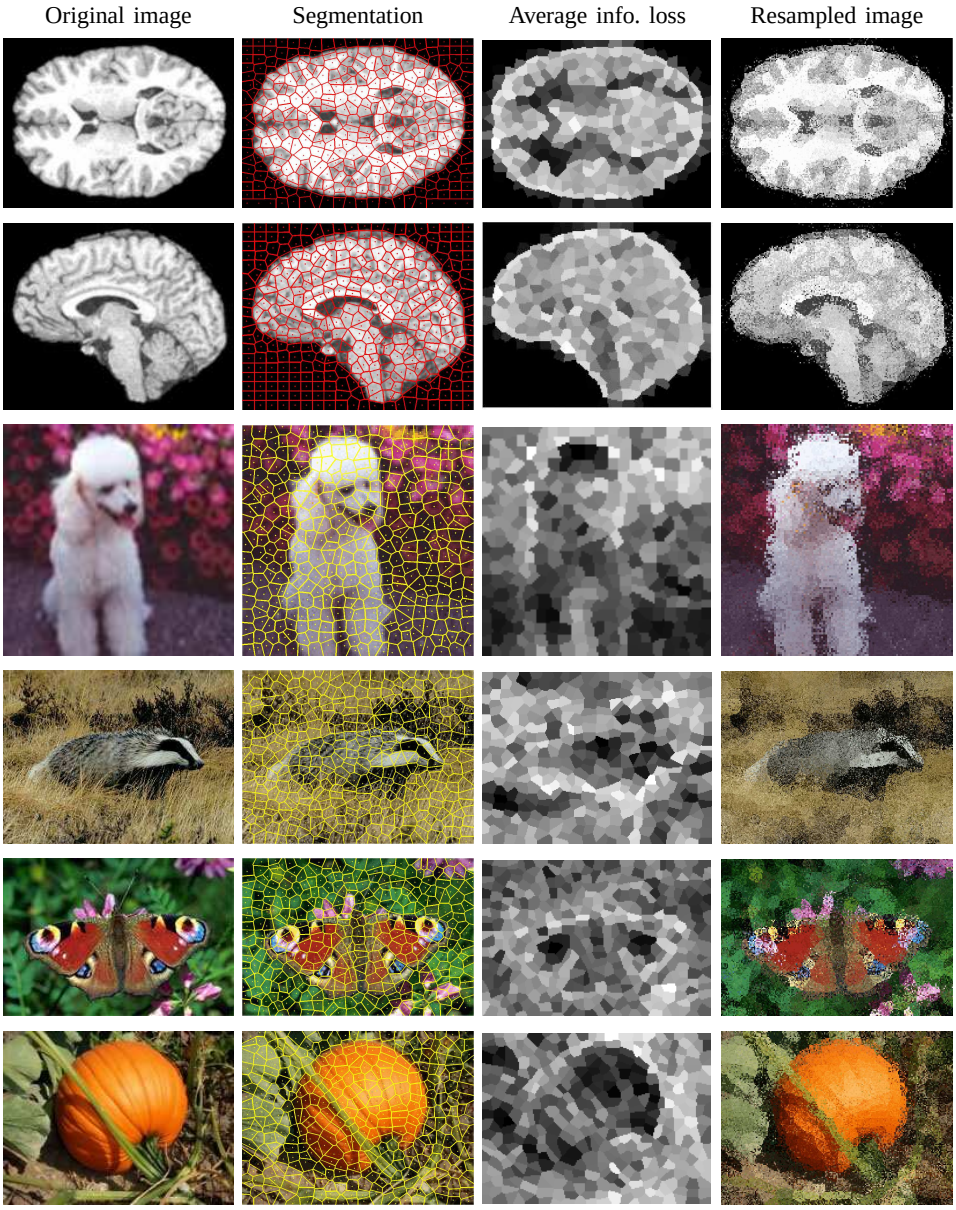


Fig. 12. Segmentation results for six images. First column: original image. Second column: centers m_k and the induced Voronoi partition after 50 iterations. Third column: map of average information loss inside each cell (higher intensity corresponds to higher loss). Note that higher loss occurs in parts of the image that are less homogeneous and have a higher level of detail. Fourth column: image created by sampling each pixel from the appearance distribution π_k of its Voronoi region.

with state-of-the-art segmentation algorithms, but to serve as a proof of concept and another demonstration of our information loss minimization framework in action.

In existing literature, KL-divergence has been used for segmentation by Heiler and Schnörr [15], who have proposed a variational framework to partition an image into two regions, Ω_{in} and Ω_{out}, by a smooth curve $\mathcal{C}$. Their objective function is as follows:

$$\int_{\mathcal{C}} ds + \int_{\Omega_{\text{in}}} D(P_x \| P_{\text{in}}) dx + \int_{\Omega_{\text{out}}} D(P_x \| P_{\text{out}}) dx \,,$$

where P_{in} and P_{out} are the prototype appearance distributions of Ω_{in} and Ω_{out}, respectively. Apart from the initial term, $\int_{\mathcal{C}} ds$, which controls the complexity of the separating boundary, note the similarity of this objective function to ours, given by eq. (13). Heiler and Schnörr motivate their objective function in terms of minimum description length [34], [12]. Namely, the quantity $D(P_X \| P_{\text{in/out}})$ represents the excess description length of encoding a pixel with true distribution P_X using a code that is optimal for the distribution $P_{\text{in/out}}$. We develop this interpretation further in Appendix A, where we show connections between information loss minimization and lossless source coding. Apart from similarly motivated objective functions, however, our approach is completely different from that of [15]: we solve for positions of multiple image centers, not for a single smooth curve, and are thus not limited to two regions (although a downside of our approach is that we cannot obtain curved boundaries); we use gradient descent instead of variational optimization; our appearance attributes are discrete color histograms, instead of closed-form parametric distributions of natural image statistics [16]. A final crucial difference is that our objective function is not tailored

exclusively to image segmentation, but is derived for the much more general problem of supervised learning of quantizer codebooks.

VI. Discussion

This paper has considered the problem of quantizing continuous feature spaces while preserving structure that is necessary for predicting a given target attribute. The basic idea behind our method is that the compressed representation of the data should be a sufficient statistic for the attribute, i.e., it should preserve *all* information about that attribute. By definition of sufficient statistics, this means that the data X and the attribute label Y should be conditionally independent given the quantizer index K. Accordingly, encoding and classification in our method follow the Markov chain $X \to K \to Y$, so that assigning a point to its nearest codevector in feature space immediately leads to an estimate of its posterior class distribution. In the realistic setting of non-ideal or lossy compression, we can only obtain an *approximate* sufficient statistic, which leads to an objective function based on information loss minimization (this function also has an alternative motivation in terms of lossless source coding, as explained in Appendix A). In designing our method, we have drawn on techniques from the fields of supervised quantizer design and information-theoretic clustering. However, unlike existing quantizer design methods, ours incorporates a generic information-theoretic criterion that does not need to assume specific classification rules and/or loss functions; and, unlike existing approaches to information-theoretic clustering, ours takes advantage of spatial coherence of vector space data and can be used to encode previously unseen test data with unknown attribute labels.

Let us make a few additional observations about our approach. The learning step (Section III-B) simultaneously solves for codebooks in the feature space and in the simplex of probability distributions. The feature space codebook $\mathcal{M} = \{m_1, \ldots, m_C\}$ can be thought of as a compressed version of the training set that still provides approximately the same performance in terms of nearest-neighbor classification. In turn, the codebook $\Pi = \{\pi_1, \ldots, \pi_C\}$ can be thought of as a piecewise-constant estimate of the posterior class distribution as a function of X. Starting with some approximate "local" estimates P_{X_i}, which can even be point masses, we find a constant estimate π_k by averaging these local estimates over a region of $\mathcal{X}$ carefully selected to minimize the loss of information in going from P_{X_i} to π_k. The estimate π_k can be used directly for MAP classification as in Section IV-A, or incorporated into a more complex statistical modeling framework, as demonstrated in Section IV-B for bag-of-features image classification and for image segmentation. Finally, note that we can gain additional flexibility at the learning stage by modifying the objective function to control the tradeoff between the supervised criterion of information loss and the unsupervised criterion of squared Euclidean distortion (Section III-C).

We close by outlining some directions for future work. First of all, constraining the encoder to a nearest-neighbor one is overly restrictive in some situations: the resulting partition cells are convex polytopes, even though cells with curved boundaries may perform better. Therefore, it would be of interest to relax the nearest-neighbor constraint and allow more general encoders. For instance, one could use Gauss Mixture Vector Quantization (GMVQ) [1] to model the distribution of the features as a Gauss mixture, thus allowing for partition cells with quadratic boundaries. Going beyond information loss minimization, our approach readily extends to any Bregman divergence [6], [2], not just the relative entropy. In fact, the Bregman centroid (5) is the unique solution to an optimization problem of the type of (8) with any other Bregman divergence [2]. For the application of image segmentation, KL-divergence may not be the most effective way to compare appearance attributes, and a different divergence may be more suitable.

In the longer term, we are interested in considering a wider class of problems of task-specific compression. In this paper, we have only addressed the relatively simple scenario where the compression is accomplished by nearest-neighbor quantization and the target task is to predict a discrete label. We can imagine more complex tasks, such as compressing video streams for the purpose of sending them over a network and performing stereo reconstruction on the other end. Clearly, it is desirable to perform compression in a way that does not destroy any relevant information for the reconstruction task — otherwise, the compression artifacts may show up as spurious structure in the 3D reconstruction. While this example is obviously much more challenging than the basic setting of this paper, the principle of information loss minimization should be powerful and general enough to serve as a guide towards effective solutions for these kinds of real-world problems.

Appendix A
Lossless Source Coding Interpretation

If we formulate the problem of inferring the class label Y from the observed feature X in the Bayesian decision-theoretic framework [35], then the main object of interest is the posterior distribution, i.e., the conditional distribution P_x of Y given $X = x$. Let us consider the problem of using the training sequence $\{(X_i, Y_i)\}_{i=1}^N$ to learn the posterior distribution P_x for every $x \in \mathcal{X}$.

This learning scheme would output a mapping $\pi : \mathcal{X} \to \mathcal{P}(\mathcal{Y})$, such that for every $x \in \mathcal{X}$, $\pi = \pi(x)$ is an approximation to P_x. The quality of this approximation can be judged in terms of the relative entropy $D(P_x \| \pi)$. The goal is to choose the mapping π to minimize the average

$$\int_{\mathcal{X}} D(P_x \| \pi(x)) d\mu(x). \tag{21}$$

The above quantity admits an information-theoretic interpretation in terms of lossless source coding [7] of the class labels.

Based on the standard correspondence between discrete probability distributions and lossless codes [7], knowing P_x is equivalent to knowing the optimal lossless code for Y given $X = x$. This code encodes each $Y = y$ with a codeword of length $-\log P_x(y)$ and has average codeword length equal to $H(P_x)$, the entropy of P_x. Suppose we observe $X = x$ and are asked to supply a lossless code for Y. We do not

know the true distribution P_x, but rather approximate it by $\pi = \pi(x)$ and then encode Y with a lossless code optimized for π. Given each $y \in \mathcal{Y}$, this code produces a binary codeword of the length $-\log \pi(y)$. The excess average codeword length or *redundancy* of this code relative to the optimal code for P_x is given by $\mathbb{E}_{P_x}[-\log \pi(Y)] - H(P_x) = D(P_x \| \pi(x))$, and the expression in (21) is then equal to the average redundancy with respect to X.

Suppose that, in choosing the map π, we are constrained to having only C possible codes for Y associated with a partition of the feature space into C cells $\mathcal{R}_1, \ldots, \mathcal{R}_C$ and the corresponding probability distributions $\{\pi_k\}_{k=1}^C$. In this scenario, the optimal set of codes (or, equivalently, the distributions π_k) is the one with minimal average redundancy $\sum_{k=1}^C \int_{\mathcal{R}_k} D(P_x \| \pi_k) d\mu(x)$. When we restrict the quantizers to nearest-neighbor ones and when we do not possess full knowledge of the distribution of (X, Y), but instead have access to a training sequence $\{(X_i, Y_i)\}_{i=1}^N$, this problem reduces to minimizing the objective function in (10). Finally, when the probabilities P_{X_i} are estimated by point masses δ_{Y_i}, the objective function simplifies to $-\sum_{k=1}^C \sum_{X_i \in \mathcal{R}_k} \log \pi_k(Y_i)$. This has the interpretation of minimizing the sum of total description lengths of the labels Y_i corresponding to the X_i's in each partition cell $\mathcal{R}_k$.

Appendix B
A uniform deviation bound for empirical information loss

In this appendix, we sketch the derivation of a uniform bound on the absolute deviation of the empirical information loss from the actual information loss over all choices of Voronoi partitions of the feature space and the corresponding codebooks of posteriors over class labels, for a fixed number of codevectors. Let $\{(X_i, Y_i)\}_{i=1}^N$ be a training sequence of independent samples from the joint distribution of X and Y. For a fixed C, let $\mathcal{M} = \{m_1, \ldots, m_C\} \subset \mathcal{X}$ be a codebook in the feature space, and let $\Pi = \{\pi_1, \ldots, \pi_C\} \subset \mathrm{Int}(\mathcal{P}(\mathcal{Y}))$ be a codebook in the probability simplex over the class label space. Denote by $\mathcal{R}_k$ the Voronoi cell of m_k and define

$$L(\mathcal{M}, \Pi) \triangleq \sum_{k=1}^C \int_{\mathcal{X}} I_{\{x \in \mathcal{R}_k\}} D(P_x \| \pi_k) d\mu(x), \tag{22}$$

where $I_{\{\cdot\}}$ is the indicator function, and

$$\hat{L}(\mathcal{M}, \Pi) \triangleq \frac{1}{N} \sum_{k=1}^C \sum_{i=1}^N I_{\{X_i \in \mathcal{R}_k\}} D(\hat{P}_{X_i} \| \pi_k). \tag{23}$$

Observe that (22) is precisely the information loss due to the partitioning of the feature space $\mathcal{X}$ into Voronoi cells $\mathcal{R}_1, \ldots, \mathcal{R}_C$ and then assigning the posterior π_k to all features in $\mathcal{R}_k$. Similarly, (23) is an empirical version of this information loss. We assume that the estimator $\hat{P}_x$ of P_x is such that, for any $\delta > 0$, the probability

$$P(N, \delta) \triangleq \Pr\left(\sup_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} |P_x(y) - \hat{P}_x(y)| > \delta \right) \to 0 \tag{24}$$

as $N \to \infty$ (this assumption can be weakened, but the resulting proof will be quite technical).

We now derive a uniform bound on the absolute deviation between (22) and (23) over all choices of $\mathcal{M}$ and Π, provided that the components of Π are not too close to the boundary of the probability simplex over $\mathcal{Y}$. Namely, given some $\theta > 0$, let us consider only those Π for which $\pi_k(y) \geq \theta$ for all $1 \leq k \leq C$ and all $y \in \mathcal{Y}$. Let us define $U_k(x) \triangleq I_{\{x \in \mathcal{R}_k\}} D(P_x \| \pi_k)$ and $\Delta_k(x) \triangleq D(P_x \| \pi_k) - D(\hat{P}_x \| \pi_k)$. Then

$$\begin{aligned}
&\sup_{\mathcal{M}, \Pi} \left| L(\mathcal{M}, \Pi) - \hat{L}(\mathcal{M}, \Pi) \right| \\
&= \sup_{\mathcal{M}, \Pi} \left| \sum_{k=1}^C \left\{ \frac{1}{N} \sum_{i=1}^N \mathbb{E}[U_k(X_i)] - I_{\{X_i \in \mathcal{R}_k\}} D(\hat{P}_{X_i} \| \pi_k) \right\} \right| \\
&\leq \sup_{\mathcal{M}, \Pi} \left| \sum_{k=1}^C \left\{ \frac{1}{N} \sum_{i=1}^N \mathbb{E}[U_k(X_i)] - U_k(X_i) \right\} \right| \\
&\quad + \sup_{\mathcal{M}, \Pi} \left| \sum_{k=1}^C \frac{1}{N} \sum_{i=1}^N I_{\{X_i \in \mathcal{R}_k\}} \Delta_k(X_i) \right| \\
&\leq \sqrt{2} \log \frac{1}{\theta} \sum_{k=1}^C \sqrt{\sup_{\mathcal{M}} \left| \frac{1}{N} \sum_{i=1}^N I_{\{X_i \in \mathcal{R}_k\}} - \mu(\mathcal{R}_k) \right|} \qquad (25) \\
&\quad + \sum_{k=1}^C \sup_{\Pi} \left| \frac{1}{N} \sum_{i=1}^N \Delta_k(X_i) \right|, \qquad (26)
\end{aligned}$$

where we have used the Cauchy–Schwarz inequality to get (25). Now fix some $\delta \in (0, 1/2)$. The supremum in (25) is over the collection of all Voronoi cells in $\mathbb{R}^d$ induced by C points. This collection is contained in the collection of all sets in $\mathbb{R}^d$ that are bounded by at most $C + 1$ hyperplanes. The shatter coefficient of the latter collection is bounded by $\left(\frac{Ne}{d+1}\right)^{(d+1)(C-1)}$ (see, e.g., Section 19.1 in [9]). Therefore, using the Vapnik–Chervonenkis inequality [9, Theorem 12.5] and the union bound, we have

$$\begin{aligned}
&\Pr\left(\sup_{\mathcal{M}} \left| \frac{1}{N} \sum_{i=1}^N I_{\{X_i \in \mathcal{R}_k\}} - \mu(\mathcal{R}_k) \right| \leq \frac{\delta^2}{2}, \forall k \right) \\
&\qquad \geq 1 - 8C \left(\frac{Ne}{d+1} \right)^{(d+1)(C-1)} e^{-N\delta^4/128}. \qquad (27)
\end{aligned}$$

Turning to (26), we first write for a fixed k

$$\begin{aligned}
\left| \frac{1}{N} \sum_{i=1}^N \Delta_k(X_i) \right| &\leq \left| \frac{1}{N} \sum_{i=1}^N \left[H(\hat{P}_{X_i}) - H(P_{X_i}) \right] \right| \\
&\quad + \log(1/\theta) \frac{1}{N} \sum_{i=1}^N \sum_{y \in \mathcal{Y}} |P_{X_i}(y) - \hat{P}_{X_i}(y)|,
\end{aligned}$$

where $H(\cdot)$ denotes the Shannon entropy [7]. Because $\delta < 1/2$, we can show that, on the event that

$$\frac{1}{N} \sum_{i=1}^N \sum_{y \in \mathcal{Y}} |P_{X_i}(y) - \hat{P}_{X_i}(y)| \leq \delta \tag{28}$$

we will have

$$\frac{1}{N} \left| \sum_{i=1}^N \left[H(P_{X_i}) - H(\hat{P}_{X_i}) \right] \right| \leq -\delta \log(\delta / |\mathcal{Y}|)$$

(see, e.g., [7, Theorem 17.3.2]). Hence, provided (28) holds,

$$\sum_{k=1}^{C} \sup_{\Pi} \left| \frac{1}{N} \sum_{i=1}^{N} \Delta_k(X_i) \right| \leq -C\delta \log(\theta\delta/|\mathcal{Y}|).$$

From (24) it follows that (28) will happen with probability at least $1 - P(N, \delta)$, which, along with (27), implies that, for any $\delta \in (0, 1/2)$,

$$\sup_{\mathcal{M},\Pi} |L(\mathcal{M}, \Pi) - \hat{L}(\mathcal{M}, \Pi)| \leq -C\delta \log(\theta^2\delta/|\mathcal{Y}|) \qquad (29)$$

with probability at least $1 - 8C[Ne/(d+1)]^{(d+1)(C-1)} e^{-N\delta^4/128} - P(N, \delta)$. In particular, (29) implies that if $(\mathcal{M}^*, \Pi^*)$ minimizes $\hat{L}(\mathcal{M}, \Pi)$, then

$$\left| L(\mathcal{M}^*, \Pi^*) - \inf_{\mathcal{M},\Pi} L(\mathcal{M}, \Pi) \right| \leq -2C\delta \log(\theta^2\delta/|\mathcal{Y}|).$$

In other words, the pair $(\mathcal{M}^*, \Pi^*)$ that minimizes empirical information loss performs close to the actual optimum with high probability. Note that the right-hand side of (29) tends to zero as $\delta \to 0$, but increases with C. This is to be expected because increasing C will result in smaller information loss (i.e., smaller bias), but this will be accompanied by an increase in the difference between the true and the empirical information loss (i.e., larger variance). By properly choosing C as a function of the sample size N to control the bias-variance trade-off, it will be possible to use the quantized representation of the features to learn a consistent classifier, where consistency is understood in the sense of asymptotically approaching the Bayes rate $\inf_{\hat{Y}} \Pr[\hat{Y}(X) \neq Y]$.

ACKNOWLEDGMENTS

We thank the anonymous reviewers and the editor for comments that helped us to improve this paper. At the time of completing this work, Svetlana Lazebnik was supported by France Telecom and the National Science Foundation under grant *Toward Category-Level Object Recognition*, J. Ponce (PI) and Y. LeCun, IIS-0535152/0535166. Maxim Raginsky was supported by the Beckman Foundation Fellowship.

REFERENCES

[1] A. Aiyer, K. Pyun, Y. Huang, D.B. O'Brien and R.M. Gray. Lloyd clustering of Gauss mixture models for image compression and classification. *Signal Processing: Image Commun.* 20: 459–485 (2005).

[2] A. Banerjee, S. Merugu, I.S. Dhillon and J. Ghosh. Clustering with Bregman divergences. *J. Mach. Learning Res.* 6:1705–1749 (2005).

[3] T. Berger. *Rate Distortion Theory: A Mathematical Basis for Data Compression.* Prentice-Hall, Englewood Cliffs, NJ, 1971.

[4] C. Bishop. *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford, 1995.

[5] D. Blackwell and M.A. Girshick. *Theory of Games and Statistical Decisions*, Wiley, New York, 1954.

[6] L.M. Bregman. The relaxation method of finding the common points of convex sets and its application to the solution of problems in convex programming. *USSR Comput. Math. and Math. Phys.*, 7:200–217 (1967).

[7] T.M. Cover and J.A. Thomas. *Elements of Information Theory*, 2nd ed., Wiley, New York, 2006.

[8] G. Csurka, C. Dance, L. Fan, J. Willamowski and C. Bray. Visual categorization with bags of keypoints. *ECCV Workshop on Statistical Learning in Computer Vision* (2004).

[9] L. Devroye, L. Györfi and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*, Springer, New York, 1996.

[10] I. Dhillon, S. Mallela, and R. Kumar. A divisive information-theoretic feature clustering algorithm for text classification. *J. Mach. Learning Res.* 3:1265-1287 (2003).

[11] A. Gersho and R.M. Gray. *Vector Quantization and Signal Compression*, Kluwer, Boston, 1992.

[12] P.D. Grünwald. *The Minimum Description Length Principle*, MIT Press, Cambridge, MA, 2007.

[13] L. Fei-Fei and P. Perona. A Bayesian hierarchical model for learning natural scene categories. *Proc. CVPR*, 2005.

[14] T. Hastie, R. Tibshirani and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, Springer, New York, 2001.

[15] M. Heiler and C. Schnörr. Natural Image Statistics for Natural Image Segmentation. *IJCV* 63(1):5–19 (2005).

[16] J. Huang and D. Mumford. Statistics of Natural Images and Models. *Proc. ICCV* (1), pp. 541–547, 1999.

[17] T. Kohonen. Learning vector quantization for pattern recognition. Tech. Rep. TKK-F-A601, Helsinki Institute of Technology, Finland (1986).

[18] T. Kohonen. Improved versions of learning vector quantization. *Proc. IEEE Int. Joint Conf. on Neural Networks*, vol. I, pp. 545–550 (1990).

[19] T. Kohonen. *Self-Organizing Maps*, 3rd ed., Springer-Verlag, Berlin (2000).

[20] S. Kullback. *Information Theory and Statistics*, Dover, New York, 1968.

[21] D. Larlus and F. Jurie. Latent mixture vocabularies for object characterization. *Proc. BMVC*, 2005.

[22] S. Lazebnik, C. Schmid and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. *Proc. IEEE Conf. Computer Vision and Pattern Recognition* (2006).

[23] S. Lazebnik and M. Raginsky. Learning nearest-neighbor quantizers from labeled data by information loss minimization. *International Conference on Artificial Intelligence and Statistics*, 2007.

[24] T. Linder. Learning-theoretic methods in vector quantization. In *Principles of Nonparametric Learning*, L. Györfi, ed., Springer-Verlag, New York (2001).

[25] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110 (2004).

[26] A. McCallum and K. Nigam. A comparison of event models for naive Bayes text classification. *AAAI-98 Workshop on Learning for Text Categorization*, pp. 41–48 (1998).

[27] F. Moosmann, B. Triggs, and F. Jurie. Randomized clustering forests for building fast and discriminative visual vocabularies. *NIPS* (2006).

[28] F. Odone, A. Barla and A. Verri. Building kernels from binary strings for image matching. *IEEE Trans. Pattern Analysis Mach. Intel.*, 14(2):169–180 (2005).

[29] K.L. Oehler and R.M. Gray. Combining image compression and classification using vector quantization. *IEEE Trans. Pattern Analysis Mach. Intel.*, 17:461–473 (1995).

[30] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *IJCV*, 42(3):145–175 (2001).

[31] E. Parzen. On estimation of a probability density function and mode. *Annals Math. Statist.*, 33(3):1065–1076 (1962).

[32] A. Rao, D. Miller, K. Rose and A. Gersho. A generalized VQ method for combined compression and estimation. *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, pp. 2032–2035 (1996).

[33] X. Ren and J. Malik. Learning a Classification Model for Segmentation. *Proc. ICCV*, vol. 1, pp. 10–17 (2003).

[34] J. Rissanen. Modelling by the shortest data description. *Automatica* 14:465-471 (1978).

[35] C.P. Robert. *The Bayesian Choice*, 2nd ed. Springer-Verlag, New York (2001).

[36] K. Rose. Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. *Proc. IEEE*, 86(11):2210-2239 (1998).

[37] G. Salton and M. McGill. *Introduction to modern information retrieval*. McGraw-Hill, New York (1986).

[38] N. Slonim, G.S. Atwal, G. Tkačik and W. Bialek. Information-based clustering. *Proc. Nat'l Acad. Sci.*, 102:18297–18302 (2005).

[39] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. *Proc. ICCV* vol. 2, pp. 1470-1477 (2003).

[40] N. Slonim and N. Tishby. The power of word clusters for text classification. *Proc. 23rd European Colloquium on Information Retrieval Research (ECIR)* (2001).

[41] M. Swain and D. Ballard. Color indexing. *IJCV* 7(1):11–32 (1991).

[42] N. Tishby, F.C. Pereira and W. Bialek. The information bottleneck method. *Proc. 37th Annual Allerton Conf. on Communication, Control and Computing*, pp. 368–377 (1999).

[43] J. Zhang, M. Marszalek, S. Lazebnik and C. Schmid. Local features and kernels for classification of texture and object categories: a comprehensive study. *IJCV* 73(2):213–238 (2007).