A Quantitative Measurement of Codebook and its Specialized Clustering Framework for Image Representation Strategies

by

Yuki Shinomiya

Student ID Number: 1196005

A dissertation submitted to the
Engineering Course, Department of Engineering,
Graduate School of Engineering,
Kochi University of Technology,
Kochi, Japan

in partial fulfillment of the requirements for the degree of
Doctor of Engineering

Assessment Committee:
Supervisor: Yukinobu Hoshino
Co-Supervisor: Shinichi Yoshida
Co-Supervisor: Kiminori Matsuzaki
Yoshiaki Takata
Tomoharu Ugawa

September 2018

Abstract

Over the years, image recognition, which is a kind of study fields in artificial intelligence, is an attractive research due to growing a number of our familiar image contents. In general, images have no any constraints to a shooting environment and viewpoint changes, so that the typical difficulty is to adapt to deformation of objects appeared in the images. Understanding images without any constraints has many important aspects in science, engineering, and technology. To ignore objects deformation, a popular strategy is a codebook-based image representation framework. This framework is closely related to document representation in natural language processing, which represents a document as a frequency histogram by counting the number of words that correspond to a common dictionary. Similarly, codebook-based image representation frameworks treat an image as a set of local feature vectors extracted from regions of interest and a local feature as a visual-word. The earliest codebook-based approach is the Bag-of-Visual-Words (BoVW), which counts the number of visual-words which correspond to a dictionary. Here, a dictionary in image recognition is usually constructed by clustering local features extracted from various images. Variants of the BoVW, such as the Fisher Vector and the Vector of Locally Aggregated Descriptors (VLAD), have recently achieved state-of-the-art performances in several image recognition tasks and domains. This dissertation focusses on recent codebook-based approaches and provides our three research articles below.

The first article provides a low-space complexity codebook using the fuzzy clustering and its encoding approach. The fuzzy clustering has been extended from the k-means by fuzzy logic, where the k-means has a disadvantage that a quantization error is larger because it assigns a sample located equidistant to two or more cluster centers to either one. In this article, we

used the fuzzy c-means (FCM) algorithm, which is a typical fuzzy clustering and allows that a sample can belong to two or more clusters by representing belonging probabilities based on a distance metric. Few researchers have applied fuzzy clustering techniques to image recognition applications. The FCM has a problem that belonging probabilities of high-dimensional samples easily converge to a value because of the curse of dimensionality. To calculate belonging probabilities efficiently, our proposed codebook projects high-dimensional samples into low-dimensional space only when calculating the probabilities. Moreover, few related works have extended the earliest encoding strategy with the fuzzy clustering algorithm. We provide a feature encoding strategy, which is able to achieve the same level as the state-of-the-art strategies.

The second article analyzes a relationship between recognition performance and codebook parameters in recent state-of-the-art approaches and presents quantitative measurement to evaluate the quality of a codebook in image recognition. In the analysis experiment, we have evaluated recognition performances of the FV and the VLAD. For comparing under the fair condition, two additional encodings modified from the FV and the VLAD have also been evaluated. The experimental results suggest that the codebook parameters for the FV easily over-fit to training data as the codebook size increases. Here, the over-fitting means that some visual-words are disappeared and do not affect to encoding. In image recognition, the codebook size is an important parameter to decide the trade-off between recognition accuracy and computational cost. To investigate the further relationship between recognition performance and model parameters, we have parameterized the FV from the perspective of fuzzy logic and have statistically analyzed. The results show a strong negative relationship between recognition accuracy and the standard deviation of prior probabilities. These results suggest that the quality of encoded signatures can be quantitatively measured

at the training phase. For optimizing the two scaling parameters when constructing a codebook, we have also discussed the influence of the scaling parameters on recognition accuracy. The space of recognition accuracy with respect to the scaling parameters is a simple convex structure. This suggests that exhaustive optimization algorithms, such as grid search, and heuristic optimization techniques, such as particle swarm optimization, are able to optimize the scaling parameters. However, there is a limitation that they are computationally expensive because clustering is necessary for each candidate of scaling parameter values.

The third article provides a clustering framework based on the quantitative measurement to construct a codebook. This framework aims to directly construct a codebook based on the quantitative measurement to relax the computational cost at the codebook construction phase. The proposed framework has a general objective function to evaluate quantitative measurement as a minimization problem. In addition, the framework also has a sub-function that is alternative to the objective function for either the k-means or the GMM. To minimize the proposed objective, some black-box optimization algorithms have been evaluated. We first conducted an experiment to compare which black-box optimization algorithm is suitable. Then, characteristics of the proposed framework and difference to the conventional clustering techniques have been discussed with synthetic clustering datasets. The experimental results suggest that the proposed framework which is an alternative to the k-means showed similar results to k-means results. The proposed framework which is an alternative to the GMM significantly improved the over-fitting degree when the training samples are complicatedly distributed. In image recognition experiment, both alternatives applied to the VLAD and the FV encodings and were evaluated on two image datasets. For the results of the VLAD with our proposed framework, the recognition performance frequently tends to be worse when compared with the original

VLAD performance. On the other hand, the results of the FV with our proposed framework improved the performance, especially when the codebook size was larger.

Acknowledgements

I would like to thank my supervisor, Associate Professor Yukinobu Hoshino, for accepting me in his project and providing many insightful comments and encouragement, but also for the hard question which incented me to widen my research from various perspectives. In the same light, I would like to thank my co-supervisors, Associate Professor Shinichi Yoshida and Associate Professor Kiminori Matsuzaki, for their excellent observations, ideas, and creativity that inspired my dissertation. I would also like to thank the rest members of my thesis committee, Associate Professor Yoshiaki Takata and Associate Professor Tomoharu Ugawa, for serving as my committee members even at hardship. I would like to thank Dr. Cao Thang at the University of Tokyo for his constructive advice and interesting ideas. Thank you also to Dr. Dang Tuan Linh, Mr. Keita Mitani, and other members in Hoshino Laboratory at Kochi University of Technology for creating the best working conditions. In addition, I would like to thank the International Relations Division staffs at Kochi University of Technology for supporting me. Finally, my deepest appreciation goes to my family for having always encouraged me to go my own way in life.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation and Overview

Image Recognition is a kind of artificial intelligence and has a purpose to develop a computer that is able to understand and explain images in the real world. In general, the images in the real world have no constraint to viewpoints and shooting environments, there are some difficulties to adapt for deformation and occlusion of objects. Understanding images without any constraint have many important aspects in science, engineering, and technology.

The following items briefly summarize some fundamental tasks that many researchers in artificial intelligence have actively tackled, and their objectives:

- **Object recognition** [1]: The semantic labels of the objects appeared in a given image are recognized. If the task is to categorize into generic groups, such as chair, airplane, and building, it is specifically called generic object recognition. For more semantically or visually similar objects, it is called fine-grained object recognition. Both generic and fine-grained object recognition tasks aim to categorize large-scale classes and image collections.

- **Object detection** [2]: The purpose is to recognize where a specific object exists in a given image.
- **Semantic segmentation** [3, 4]: The purpose is to generate pixel-wise labels for each semantic object.
- **Image retrieval** [5, 6]: It is an information retrieval with images as queries.

Clustering is a fundamental technique for several purposes such as statistical analysis and data mining. The main purpose of clustering is to make groups called clusters. Each clustering technique has a specific objective to make groups, such as finding groups that minimize a quantization error and estimation of the appropriate distribution [7, 8]. This paper focusses on clustering in image recognition algorithms and presents an efficient objective.

In recent image recognition problems, a local feature framework is a key technique. This detects interest regions on an image and describes a discriminative feature vector from each region. The basic idea of codebook-based encodings is to capture the statistics of the distribution of local features extracted from an image. By treating local features as visual vocabularies appeared in an image, images can be processed in the same way as the natural language processing (NLP). In the NLP, specifically, the bag-of-words (BOW) model [9] expresses a document feature vector by assigning words existing in sentences to corresponding common words and counting their frequencies. For images, common visual words, called codebook, are constructed by clustering local features extracted from various images. The model in image recognition follows the same procedure as the BOW to represent image feature vectors. This approach is well-known as the bag-of-visual-words (BoVW) model [10], and its variants [5, 11–15] have achieved excellent performance on several tasks, such as object recognition [11, 12, 14] and image retrieval [5, 15].

Gosselin et al. [13] have suggested that increasing the number of common visual vocabularies is an important factor for improving recognition performance. For instance, the best recognition rate has been observed with the largest vocabulary size in their experiment. It has also been reported that saturation of the recognition performances accompanying the increase the vocabulary size has not been observed. On the other hand, a huge vocabulary size becomes a cause of high computational complexity [13] and to possibly generate not suitable vocabularies due to the over-fitting to clustering samples [12]. Our previous study [16] has considered that the distribution of prior probability can be used to measure the quality of image feature vectors in codebook-based feature encoding strategies. In addition, optimization of the distribution does not require additional computational complexity in practical applications because it is an offline step in the image recognition pipeline.

This dissertation focuses on the codebook construction step and presents some application for image recognition problems.

1.2 Contributions

This dissertation consists of our three research articles, this section introduces brief summaries and contributions of the articles.

1.2.1 Fuzzy Codebook on Image Recognition Problems

While the purpose of generic image recognition is to recognize the basic level categories [17], the codebook approach has been applied to more complex domains such as aesthetics estimation and fine-grained visual categorization (FGVC), which is difficult to humans [17]. In many cases of FGVC and aesthetics estimation, the recognition pipeline consists of multiple local feature

frameworks; According to [18, 19], the independent four local feature frameworks has been used, the extracted local features are individually converted to the global image features. Each image feature is recognized the object category. After that, final result is determined by combining the results of each local feature framework. In such the system including multiple local feature frameworks, the codebook is generated for each local feature framework. Therefore, the codebook with small memory footprints is required for the large-scale and fine-grained image recognition.

Our objective is to reduce the space complexity of image representation step with keeping recognition performance. To reduce the space complexity, our approach uses fuzzy clustering to generate the codebook, this codebook is called fuzzy codebook. To keep recognition performance, an image is represented to a high dimensional vector in the same way as recent image representation.

1.2.2 A Quantitative Measurement for Codebook-based Strategies

Codebook-based image representation has been widely used in many image recognition applications. While each of the applications has an individual purpose, the pipeline of image representation follows the same approach. The second article analyzes a relationship between recognition performance and codebook parameters in recent state-of-the-art approaches and presents quantitative measurement to evaluate the quality of a codebook in image recognition. In the analysis experiment, we have evaluated recognition performances of the FV and the VLAD. For comparing under the fair condition, two additional encodings modified from the FV and the VLAD have also been evaluated. The experimental results suggest that the codebook parameters for the FV easily over-fit to training data as the codebook size increases.

Here, the over-fitting means that some visual-words are disappeared and do not affect to encoding. In image recognition, the codebook size is an important parameter to decide the trade-off between recognition accuracy and computational cost. To investigate the further relationship between recognition performance and model parameters, we have parameterized the FV from the perspective of fuzzy logic and have statistically analyzed. The results show a strong negative relationship between recognition accuracy and the standard deviation of prior probabilities. These results suggest that the quality of encoded signatures can be quantitatively measured at the training phase. For optimizing the two scaling parameters when constructing a codebook, we have also discussed the influence of the scaling parameters on recognition accuracy. The space of recognition accuracy with respect to the scaling parameters is a simple convex structure. This suggests that exhaustive optimization algorithms, such as grid search, and heuristic optimization techniques, such as particle swarm optimization, are able to optimize the scaling parameters. However, there is a limitation that they are computationally expensive because clustering is necessary for each candidate of scaling parameter values.

1.2.3 Optimization Frameworks Based on the Quantitative Measurement

The second article requires the much expensive computattional cost, so that the third article provides a clustering framework based on the quantitative measurement to construct a codebook. This framework aims to directly construct a codebook based on the quantitative measurement to relax the computational cost at the codebook construction phase. The proposed framework has a general objective function to evaluate quantitative measurement as a minimization problem. In addition, the framework also has a sub-function

that is alternative to the objective function for either the k-means or the GMM. To minimize the proposed objective, some black-box optimization algorithms have been evaluated. We first conducted an experiment to compare which black-box optimization algorithm is suitable. Then, characteristics of the proposed framework and difference to the conventional clustering techniques have been discussed with synthetic clustering datasets. The experimental results suggest that the proposed framework which is an alternative to the k-means showed similar results to k-means results. The proposed framework which is an alternative to the GMM significantly improved the overfitting degree when the training samples are complicatedly distributed. In image recognition experiment, both alternatives applied to the VLAD and the FV encodings and were evaluated on two image datasets. For the results of the VLAD with our proposed framework, the recognition performance frequently tends to be worse when compared with the original VLAD performance. On the other hand, the results of the FV with our proposed framework improved the performance, especially when the codebook size was larger. The results suggest that the proposed framework is able to improve recognition performance in other domains.

1.3 Organization

This dissertation consists of six chapters. The contents is organized as

- Chapter 1 introduces the general overview of this research.
- Chapter 2 reviews recetn approaches in image recognition problems.
- Chapter 3 focuses on a clustering using fuzzy theory. We first discuss the behavior on high-dimensional space and slightly modify to efficient calculation. After that, it is applied to image recognition problems.

- Chapter 4 analyzes codebook-based image representation approaches. The results suggest the strong relationship between the model parameters of a codebook and its recognition performance in the object recognition task. A quantitative measurement is also presented.
- Chapter 5 proposes a clustering framework developed from the perspective of the quantitative measurement.
- Chapter 6 concludes this dissertation.

References

[1] Li Fei-Fei, Rob Fergus, and Pietro Perona. "Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories". In: *Comput. Vis. Image Underst.* 106.1 (Apr. 2007), pp. 59–70.

[2] Navneet Dalal and Bill Triggs. "Histograms of Oriented Gradients for Human Detection". In: *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*. CVPR '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 886–893.

[3] João Carreira et al. "Semantic Segmentation with Second-order Pooling". In: *Proceedings of the 12th European Conference on Computer Vision - Volume Part VII*. ECCV'12. Florence, Italy: Springer-Verlag, 2012, pp. 430–443.

[4] J. Carreira et al. "Free-Form Region Description with Second-Order Pooling". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2015).

[5] Relja Arandjelovic and Andrew Zisserman. "All About VLAD". In: *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*. IEEE, 2013, pp. 1578–1585.

[6] Jonathan Delhumeau et al. "Revisiting the VLAD Image Representation". In: *Proceedings of the 21st ACM International Conference on Multimedia*. MM '13. Barcelona, Spain: ACM, 2013, pp. 653–656.

[7] S. Lloyd. "Least Squares Quantization in PCM". In: *IEEE Trans. Inf. Theor.* 28.2 (Sept. 2006), pp. 129–137.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the Royal Statistical Society: Series B* 39 (1977), pp. 1–38.

[9] Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. Cambridge, MA, USA: MIT Press, 1999.

[10] Gabriella Csurka et al. "Visual categorization with bags of keypoints". In: *In Workshop on Statistical Learning in Computer Vision, ECCV*. 2004, pp. 1–22.

[11] Florent Perronnin and Christopher R. Dance. "Fisher Kernels on Visual Vocabularies for Image Categorization." In: *CVPR*. IEEE Computer Society, 2007.

[12] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. "Improving the Fisher Kernel for Large-scale Image Classification". In: *Proceedings of the 11th European Conference on Computer Vision: Part IV*. ECCV'10. Heraklion, Crete, Greece: Springer-Verlag, 2010, pp. 143–156.

[13] Philippe-Henri Gosselin et al. "Revisiting the Fisher vector for fine-grained classification". In: *Pattern Recognition Letters* 49 (Nov. 2014), pp. 92–98.

[14] Xi Zhou et al. "Image Classification Using Super-vector Coding of Local Image Descriptors". In: *Proceedings of the 11th European Conference on Computer Vision: Part V*. ECCV'10. Heraklion, Crete, Greece: Springer-Verlag, 2010, pp. 141–154.

[15] R. Arandjelović and A. Zisserman. "Three things everyone should know to improve object retrieval". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2012.

[16] Yuki Shinomiya and Yukinobu Hoshino. "An Analysis of Dependency of Prior Probability for Codebook-Based Image Representation". In:

2016 Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS), Sapporo, Japan, August 25-28, 2016. 2016, pp. 103–108.

[17] Thomas Berg et al. "Birdsnap: Large-Scale Fine-Grained Visual Categorization of Birds". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*. IEEE, 2014, pp. 2019–2026.

[18] Hideki Nakayama. "Augmenting descriptors for fine-grained visual categorization using polynomial embedding." In: *ICME*. IEEE Computer Society, 2013, pp. 1–6.

[19] Keiji Yanai, Takuma Maruyama, and Yoshiyuki Kawano. "A Cooking Recipe Recommendation System with Visual Recognition of Food Ingredients". In: *iJIM* 8.2 (2014), pp. 28–34.

Chapter 2

Literature Review

2.1 Introduction to Image Representation Approaches

2.1.1 General Procedure on Image Recognition

In the early researches in image recognition, primitive information, such as color, texture, and shape, has been used to describe feature vectors from an image. Such image features are treated as *visual-words* appeared in the image. To encode these features into a single vector as a global image signature, a general theory follows the vector quantization, which is inspired by document representation in the natural language processing. In the theory, a *codebook*, which is a set of some basis vectors, *vocabularies*, are constructed from the image features extracted from a lot of images before feature encoding. At the encoding phase, each image feature is assigned to its closest basis vector. The global image signature represents the frequency of assigned features for each vocabulary. This approach is called the Bag-of-Visual-Words (BoVW) [1].

The primitive image features have some problems due to their low-level information to describe object deformations. To extract more discriminative and robust image features, several local feature frameworks [2–4] have been presented, where the Scale-Invariant Feature Transform (SIFT) framework [2, 3] is well-known as the de-facto standard in image recognition experiments.

The local feature frameworks generally consist of the detection and the description steps. In the case of the SIFT, the detection step explores interest regions that are invariant to scale and location of objects on images, and then remove unstable points that are on edge and flat areas. At the description step, local descriptors are described from each interest region. Each interest region is first normalized by rotating its direction according to the strongest gradient direction, where this process gives the invariance to the orientation change. The local descriptors are then represented by aggregating the quantized relative directions with weighting by corresponding gradient magnitudes and Gaussian weight. Image signatures encoded from invariant local descriptors have achieved good recognition performances in image recognition tasks, such as object categorization and image retrieval. To recognize labels and categories of encoded global image signatures, discriminant models are well used.

As further improvements for the BoVW variants, the sampling strategy [5] is the critical factor for recognition performance. Their evaluation of sampling approaches has experimentally showed that the detection step in the local feature frameworks is not always effective in practical applications.

The basic pipeline for recognizing objects consists of the following steps.

1. *Extract local features*. A given image is first converted to a set of d-dimensional local features. The local features [2–4] have the robustness to some deformations, such as scale, rotation, occlusion.

2. *Encode to an image feature*. The above set is then encoded to a single feature vector based on a codebook, which is a set of basis vectors.

3. *Recognize object labels*. A discriminant model is used to predict object labels. Typically, the support vector machine (SVM) with a linear kernel

is used because of its computational efficiency. The computational complexity at the model construction phase is a linear order with respect to the number of training samples [6, 7].

Here, the codebook is constructed in advance in an offline step. This section reviews the codebook construction step and the feature encoding step. Figure 2.1 illustrates an example of the above procedure.

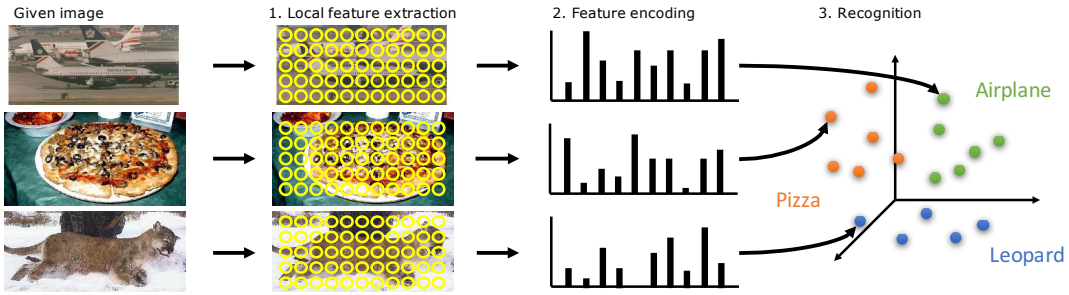


FIGURE 2.1: Illustrative example of a general procedure in object recognition. The given images are taken from the Caltech-101 dataset [8].

2.1.2 Codebook construction

The basic clustering algorithms are the k-means [9] and the Gaussian mixture model (GMM) [10]. The aim of the k-means algorithm is to find the clusters that minimize the quantization error between given samples and the corresponding mean vector. The GMM constructs Gaussians that well represents the normal distribution of given samples. In general, clustering algorithms cannot directly find global optimal by any analysis. To find an optimal solution, the above algorithms follow an iterative procedure, called the expectation and maximization (EM) algorithm, for exploring local minima. This algorithm consists of the following two steps: the expectation step and the maximization step.

In the case of the k-means, let $X = \{x_t \in \mathbb{R}^d\}_{t=1}^T$ and $\Theta = \{\mu_k \in \mathbb{R}^d\}_{k=1}^K$ respectively be the clustering samples and the model parameters, the objective function is defined as follows:

$$J_{\text{k-means}} = \sum_{t=1}^{T} \sum_{k=1}^{K} p\left(x_t; \mu_k\right) \left\|x_t - \mu_k\right\|^2, \tag{2.1}$$

where $J_{k-means}$ is the objective value, which measures the quantization error between the samples and the clusters, $p(x_t; \mu_k)$ is a probability function that becomes 1 if μ_k is the nearest cluster to x_t and 0 otherwise, and $\|\cdot\|$ is the Euclidean norm operator. To minimize the quantization error, the k-means algorithm iteratively optimizes the model parameters with Eq. (2.2) for the expectation step and Eq. (2.3) for the maximization step.

$$q_{t,k} = \begin{cases} 1 & \text{if } k = \arg\min_j \left\|\boldsymbol{x}_t - \boldsymbol{\mu}_j\right\|^2 \\ 0 & \text{otherwise} \end{cases}, \tag{2.2}$$

$$\widehat{\boldsymbol{\mu}}_k = \frac{\sum_{t=1}^{T} q_{t,k} \boldsymbol{x}_t}{\sum_{t=1}^{T} q_{t,k}}, \tag{2.3}$$

In the expectation step, the probabilities $q_{t,k}$ of a sample x_t are computed using the current mean vectors. Then, the maximization step updates the positions. The EM algorithm iterates the above two steps until termination criteria, such as a designated maximum number of iterations and the convergence of the moves, are satisfied. Fig. 2.2 shows an example of the cluster centers estimated by the k-means and their areas.

The GMM also follows the EM algorithm. The GMM model contains $\{w_k \in \mathbb{R}, \boldsymbol{\mu}_k \in \mathbb{R}^d, \boldsymbol{\Sigma}_k \in \mathbb{R}^{d \times d}\}_{k=1}^K$, where $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ denote the mean position vector and the covariance matrix to represent k-th Gaussian and w_k is a mixing weight to linear combine K Gaussians. The mixing weight w_k is also called "prior probability", which means ease of assignment to the k-th

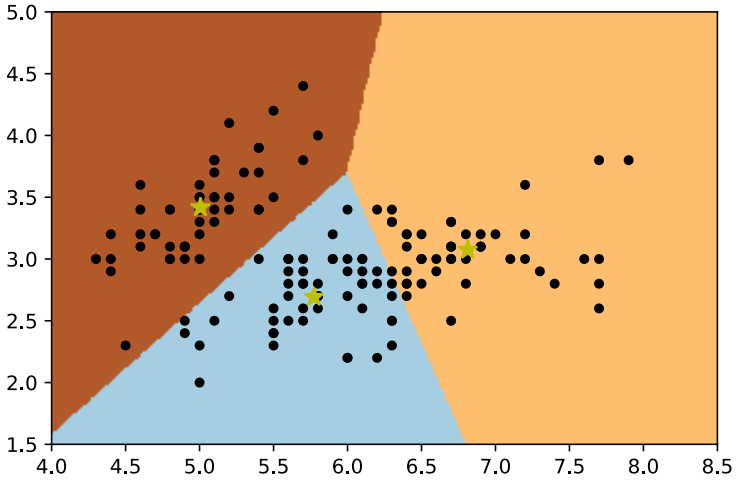


FIGURE 2.2: Example of the k-means clusters and their boundaries. The clustering samples, denoted by the black circles, are taken from the iris dataset. The yellow stars denote the cluster centers constructed by the k-means clustering. Each colored region shows the area that corresponding cluster covers.

Gaussian.

In the expectation step of the EM procedure, the assignment probability of a sample $\boldsymbol{x}_t$ to the k-th Gaussian, represented by $\boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$, is estimated as:

$$p(\boldsymbol{x}_t; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}_k|}} \exp\left[-\frac{1}{2}(\boldsymbol{x}_t - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_t - \boldsymbol{\mu}_k)\right], \tag{2.4}$$

$$q_{t,k} = \frac{w_k p(\boldsymbol{x}_t; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^{K} w_j p(\boldsymbol{x}_t; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}. \tag{2.5}$$

The maximization step of the EM procedure updates the Gaussians with the predicted assignment probabilities at the expectation step as:

$$\widehat{w}_k = \frac{\sum_{t=1}^{T} q_{t,k}}{\sum_{t=1}^{T}\sum_{k=1}^{K} q_{t,k}}, \tag{2.6}$$

$$\widehat{\boldsymbol{\mu}}_k = \frac{\sum_{t=1}^{T} q_{t,k}\boldsymbol{x}_t}{\sum_{t=1}^{T} q_{t,k}}, \tag{2.7}$$

$$\widehat{\boldsymbol{\Sigma}}_k = \frac{\sum_{t=1}^{T} q_{t,k}(\boldsymbol{x}_t - \widehat{\boldsymbol{\mu}}_k)(\boldsymbol{x}_t - \widehat{\boldsymbol{\mu}}_k)^\top}{\sum_{t=1}^{T} q_{t,k}}. \tag{2.8}$$

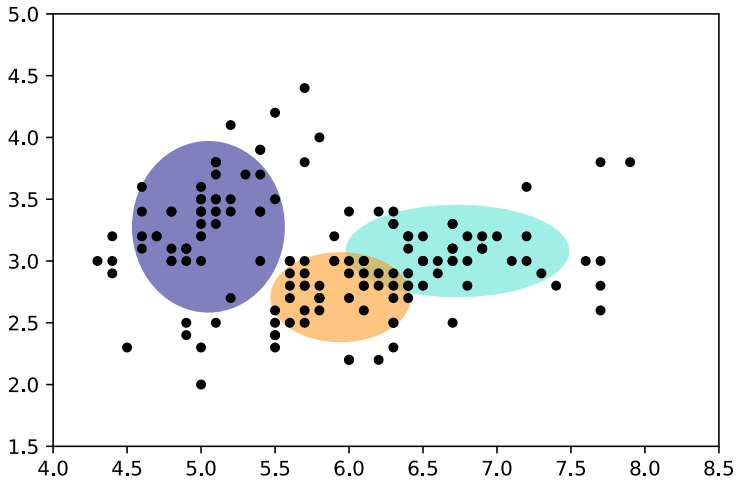


FIGURE 2.3: Example of the Gaussians estimated by the GMM. The clustering samples, denoted by the black circles, are taken from the iris dataset. The colored ellipses show the Gaussians.

Fig. 2.3 shows an example of the estimated Gaussian mixture on a toy example.

2.1.3 Feature encoding

As introduced in the previous section, the BoVW is the simplest approach to represent image features and well performs in image recognition issues. The BoVW usually uses the k-means codebook. Let $I = \{x_i \in \mathbb{R}^d\}_{i=1}^N$ be a set of d-dimensional local descriptors extracted from an image, the BoVW feature is defined as:

$$\mathcal{F}_{\text{BoVW}} = [\cdots f_k \cdots]^\top, \quad f_k = \sum_{i=1}^{N} p(x_i; \mu_k), \tag{2.9}$$

where $f_k \in \mathbb{R}^1$ is the frequency of the local descriptors assigned to the k-th visual vocabulary. For precisely capture image information, the BoVW requires a huge codebook, because the dimensionality of the BoVW is equal to a codebook size K, and it increases the computational cost, such as the finding nearest neighbors as in Eq. (2.2). Recently developed approaches

[7, 11] relax this issue by capturing higher order statistics on d-dimensional local feature space with a smaller codebook. In recent reports, the Fisher Vector (FV) [7, 12] and the Vector of Locally Aggregated Descriptors (VLAD) [11, 13] encodings are well known as state-of-the-art approaches.

The FV supplements two higher-order statistics with the GMM codebook, in addition to the frequency as follows:

$$\mathcal{F}_{\mathrm{FV}} = \left[\cdots \mathcal{F}_k^{(w)} \cdots \mathcal{F}_k^{(\mu)} \cdots \mathcal{F}_k^{(\sigma)} \cdots \right], \tag{2.10}$$

where $\mathcal{F}^{(w)} \in \mathbb{R}^1$, $\mathcal{F}^{(\mu)} \in \mathbb{R}^d$, and $\mathcal{F}^{(\sigma)} \in \mathbb{R}^d$ respectively denote frequency, mean, and covariance. These are captured as:

$$\mathcal{F}_k^{(w)} = \frac{1}{N\sqrt{w_k}} \sum_{i=1}^{N} \left(\gamma_i (k) - w_k \right), \tag{2.11}$$

$$\mathcal{F}_k^{(\mu)} = \frac{1}{N\sqrt{w_k}} \sum_{i=1}^{N} \gamma_i \frac{x_i - \mu_k}{\sigma_k}, \tag{2.12}$$

$$\mathcal{F}_k^{(\sigma)} = \frac{1}{N\sqrt{2w_k}} \sum_{i=1}^{N} \gamma_i \left[\left(\frac{x_i - \mu_k}{\sigma_k} \right)^2 - 1 \right], \tag{2.13}$$

where the Gaussians are assumed to have diagonal covariances because of the derivation [7] and computational reasons [12, 14]. So that, a FV signature have $K(2d+1)$-dimensions. The VLAD captures only mean statistics by aggregating the residuals between the local features and the mean vectors of the codebook as follows:

$$\mathcal{F}_{\mathrm{VLAD}} = \left[\cdots \mathcal{F}_k^{(\mu)} \cdots \right], \tag{2.14}$$

$$\mathcal{F}_k^{(\mu)} = \sum_{i=1}^{N} p(x_i, \mu_k)(x_i - \mu_k), \tag{2.15}$$

where the dimensionality of a VLAD signature is Kd.

Table 2.1 summarizes the dimensionality of represented signatures and the space complexity of model parameters.

TABLE 2.1: The signature dimensionality and the space complexity of model parameters regarding to the local feature dimensionality D and the codebook size K.

Method	**Signature dimensionality**	**Space complexity of model parameters**
BoVW	K	Kd
FV	$K(2d+1)$	$K(2d+1)$
VLAD	Kd	Kd

References

[1] Gabriella Csurka et al. "Visual categorization with bags of keypoints". In: *In Workshop on Statistical Learning in Computer Vision, ECCV*. 2004, pp. 1–22.

[2] David G. Lowe. "Object Recognition from Local Scale-Invariant Features". In: *Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2*. ICCV '99. Washington, DC, USA: IEEE Computer Society, 1999, pp. 1150–.

[3] David G. Lowe. "Distinctive Image Features from Scale-Invariant Keypoints". In: *Int. J. Comput. Vision* 60.2 (Nov. 2004), pp. 91–110.

[4] Herbert Bay et al. "Speeded-Up Robust Features (SURF)". In: *Comput. Vis. Image Underst.* 110.3 (June 2008), pp. 346–359.

[5] Eric Nowak, Frédéric Jurie, and Bill Triggs. "Sampling strategies for bag-of-features image classification". In: *European Conference on Computer Vision*. Springer, 2006.

[6] S. Maji and A. C. Berg. "Max-margin additive classifiers for detection". In: *2009 IEEE 12th International Conference on Computer Vision*. 2009, pp. 40–47.

[7] Florent Perronnin and Christopher R. Dance. "Fisher Kernels on Visual Vocabularies for Image Categorization." In: *CVPR*. IEEE Computer Society, 2007.

[8] Li Fei-Fei, Rob Fergus, and Pietro Perona. "Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories". In: *Comput. Vis. Image Underst.* 106.1 (Apr. 2007), pp. 59–70.

[9] S. Lloyd. "Least Squares Quantization in PCM". In: *IEEE Trans. Inf. Theor.* 28.2 (Sept. 2006), pp. 129–137.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm". In: *Journal of the Royal Statistical Society: Series B* 39 (1977), pp. 1–38.

[11] Relja Arandjelovic and Andrew Zisserman. "All About VLAD". In: *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*. IEEE, 2013, pp. 1578–1585.

[12] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. "Improving the Fisher Kernel for Large-scale Image Classification". In: *Proceedings of the 11th European Conference on Computer Vision: Part IV*. ECCV'10. Heraklion, Crete, Greece: Springer-Verlag, 2010, pp. 143–156.

[13] R. Arandjelović and A. Zisserman. "Three things everyone should know to improve object retrieval". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2012.

[14] Philippe-Henri Gosselin et al. "Revisiting the Fisher vector for fine-grained classification". In: *Pattern Recognition Letters* 49 (Nov. 2014), pp. 92–98.

Chapter 3

Fuzzy Clustering for Codebook Construction and Its Application to Image Representation

3.1 Introduction

While the purpose of generic image recognition is to recognize the basic level categories [1], the codebook approach has been applied to more complex domains such as aesthetics estimation and fine-grained visual categorization (FGVC), which is difficult to humans [1]. In many cases of FGVC and aesthetics estimation, the recognition pipeline consists of multiple local feature frameworks; According to [2, 3], the independent four local feature frameworks has been used, the extracted local features are individually converted to the global image features. Each image feature is recognized the object category. After that, final result is determined by combining the results of each local feature framework. In such the system including multiple local feature frameworks, the codebook is generated for each local feature framework. Therefore, the codebook with small memory footprints is required for more complex domains.

Our objective is to reduce the space complexity of image representation

step with keeping recognition performance. To reduce the space complexity, our approach uses fuzzy clustering to generate the codebook, this codebook is called fuzzy codebook. To keep recognition performance, an image is represented to a high dimensional vector in the same way as recent image representation.

3.2 Fuzzy Clustering Principle

A fuzzy c-means (FCM) is a basic fuzzy clustering technique and extend the k-means by fuzzy theory [4]. In particular, the FCM is derived by adding a fuzzifier parameter m to the objective function of the k-means. The objective of the FCM is defined as:

$$J_{\mathrm{FCM}} = \sum_{k=1}^{K} \sum_{t=1}^{T} (q_{t,k})^m \|\boldsymbol{x}_t - \boldsymbol{\mu}_k\|^2, \tag{3.1}$$

where $q_{t,k}$ denotes a belonging probability of the t-th sample $\boldsymbol{x}_t$ for the k-th cluster $\boldsymbol{\mu}_k$. The belonging level $q_{t,k}$ is given by:

$$q_{t,k} = \left[\sum_{j=1}^{K} \left(\frac{\|\boldsymbol{x}_t - \boldsymbol{\mu}_k\|}{\|\boldsymbol{x}_t - \boldsymbol{\mu}_j\|} \right)^{\frac{2}{m-1}} \right]^{-1}, \tag{3.2}$$

The fuzzy codebook is generated by iterative optimization by s-step, the center of each vocabulary of the codebook at s-step is:

$$\widehat{\boldsymbol{\mu}_k} = \frac{\sum_{t=1}^{T} q_{t,k} \boldsymbol{x}_t}{\sum_{t=1}^{T} q_{t,k}}. \tag{3.3}$$

In our implementation of the fuzzy clustering, the termination criterion is defined as:

$$\max \|\boldsymbol{Q}_t^{(s+1)} - \boldsymbol{Q}_t^{(s)}\| < \epsilon, \tag{3.4}$$

where $\boldsymbol{Q}_t^{(s)} = [q_{t,1} \dots q_{t,k}]^\top$ is a distribution of belonging levels of the sample

$\boldsymbol{x}_i$ to all vocabulary of the codebook $\boldsymbol{\mu}_k$ at s-step. The distribution of belonging levels is a probability distribution, the termination criterion is treated as a minimization of KL divergence which is optimization to local solution. ϵ is a threshold for termination criterion. The FCM cannot effectively calculate the belonging level of high dimensional sample such as the local feature. Fig. 3.1 shows the frequency distribution of the Euclidean distance between each vectors in the D-dimensional uniform distribution. The following is the property of the D-dimensional vectors:

3.3 Problems on High-dimensional Samples

The FCM cannot effectively calculate the belonging level of high dimensional sample such as the local feature. Fig. 3.1 shows the frequency distribution of the Euclidean distance between each vectors in the D-dimensional uniform distribution. The following is the property of the D-dimensional vectors:

- In the case of small D, the frequency distribution has spread.
- The frequency distribution is narrowed with increasing of D.
- The frequency distribution has a limitation of the distance of most frequent.

By these properties, belonging levels are easily converged to $\lim_{\|\boldsymbol{x}_i\| \to \infty} = 1/K$ in high-dimensional space. Therefore, the FCM has a problem that some of vocabularies of the fuzzy codebook are converged to the same point.

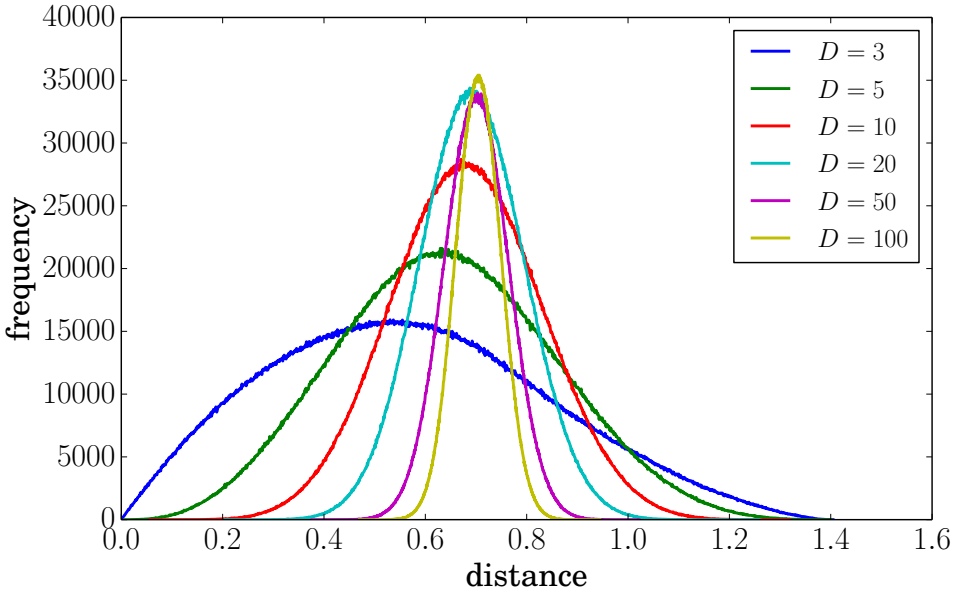


FIGURE 3.1: Example of the curse of dimensionality.

3.4 Modification of Fuzzy Codebook and its Application to Feature Encoding

In our approach, both local features and means of the fuzzy codebook are dimensionality reduced by the PCA in the only function of belonging probabilities. In addition, the fuzzy codebook is updated 1-step to the local descriptors for each image, an image is represented by capturing the frequency f_k, mean $\boldsymbol{u}_k^{(1)}$ and gradient $\boldsymbol{v}_k^{(2)}$ components as follows:

$$f_k = \sum_{\boldsymbol{x}_i \in \boldsymbol{\mu}_k} q_{ik}, \tag{3.5}$$

$$\boldsymbol{v}_k^{(1)} = \sum_{\boldsymbol{x}_i \in \boldsymbol{\mu}_k} q_{ik} \left(\boldsymbol{x}_i - \boldsymbol{\mu}_k \right), \tag{3.6}$$

$$\boldsymbol{v}_k^{(2)} = \sum_{\boldsymbol{x}_i \in \boldsymbol{\mu}_k} q_{ik} \frac{\boldsymbol{x}_i - \boldsymbol{\mu}_k}{KL\left(\boldsymbol{Q}_i^{(s+1)} \| \boldsymbol{Q}_i^{(s)} \right)}, \tag{3.7}$$

where $KL(\cdot\|\cdot)$ is a function of KL divergence that is a measure of the diference between two probabilistic distributions. KL divergence is given by:

$$KL(\boldsymbol{P}\|\boldsymbol{Q}) = \sum_i P_i \log \frac{P_i}{Q_i}. \tag{3.8}$$

An image is represented to $K(2D+1)$-dimensional image feature. This image feature consists of various statistics components, each component of vocabularies of the fuzzy codebook is applied the component-wise L2-normalization. After that, the image feature is applied the power normalization and global L2-normalization. The function of KL divergence includes the logarithmic function that is high computational cost. To reduce the computational cost, in this paper, we use the L2-norm as a measure of the difference between two probabilistic distributions for gradient component as follows:

$$\boldsymbol{v}_k = \sum_{\boldsymbol{x}_i \in \boldsymbol{\mu}_k} q_{ik} \frac{\boldsymbol{x}_i - \boldsymbol{\mu}_k}{\left\| \boldsymbol{Q}_{\boldsymbol{i}}^{(s+1)} - \boldsymbol{Q}_{\boldsymbol{i}}^{(s)} \right\|}. \tag{3.9}$$

3.5 Experiments

Our approach was compared with the recent feature encoding techniques (the BOVW and FV encodings). As the experimental setup, Caltech-101 dataset [5] was used, which consists of 9,145 images from 101 object categories and background categories, and each category contains about 40 to 800 images. Standard SIFT was used as the local feature framework. SIFT descriptors were described from four scale levels (16, 22, 31 and 44 pixels) on the intersection of regular grid of 6 pixel spacing. For creating codebook, the SIFT descriptors extracted from 510 images that were randomly selected 5 images from each category were used and the termination criterion was 30-step iterations. The dimensionality of local feature in Eq. 3.2 was reduced to 16-dimensional local feature. As the online classifier, one-versus-rest online

passive-aggressive (PA) classifier [6] was used. In addition, hinge loss was used as a loss function. The number of training image features were used 15 and 30 images per category and training image features were not divided. The hyper parameter was optimized by 5-fold cross validation. The recognition performance was evaluated over 5 trials of independent training and test images.

For the parameters of the fuzzy codebook, the dimensionality of the reduced SIFT descriptors was set to $D' = 16$ because the cumulative contribution ratio over the 60 % as shown in Fig. 3.2, and the fuzzifier parameter was set to $m = 1.4$ empirically.

Table 3.1 shows the experimental results. Our approach was recorded 0.94% (at 15 training images) and 1.00% (at 30 training images) higher recognition rates than recent feature encoding techniques. However, the difference between KL divergence and L2-norm are not clear. To clear the difference, we should evaluate on the other number of training images. Additionally, the memory footprint of image feature is important for the application on the real world, the sparsity of image features should be compared with recent feature encodings.

Table 3.2 shows the results of our approach and the VLAD encoding with $K = 16, 32, 64$. At all of the numbers of training samples, our approach significantly outperformed to the results of VLAD.

TABLE 3.1: A comparison of recognition performance (%) with respect to the number of training images

Method	Codebook size	The number of training images per category 15	30
BOVW [7]	4,000	—	50.21 (—)
FV [7]	256	—	70.48 (—)
Ours (KL) with SVM	256	64.26 (0.35)	71.52 (0.62)
Ours (L2) with SVM	256	65.20 (0.29)	—
Ours (L2) with PA	256	65.20 (0.40)	71.48 (0.62)

TABLE 3.2: A comparison of recognition performance (accuracy% and standard deviation) with respect to the number of training images.

Method	Codebook size	Dimensionality	The number of training images per category 5	10	15	25	30
VLAD	16	2,048	44.73 (0.49)	50.59 (0.97)	55.08 (0.43)	61.35 (0.27)	64.02 (0.51)
VLAD	32	4,096	44.82 (0.78)	54.02 (1.31)	57.32 (0.95)	62.88 (0.30)	65.20 (0.26)
VLAD	64	8,192	47.62 (1.01)	55.09 (0.56)	59.41 (0.40)	64.95 (0.39)	67.17 (0.19)
Ours	16	4,112	47.74 (0.95)	55.35 (0.51)	59.46 (0.30)	65.18 (0.43)	67.60 (0.55)
Ours	32	8,224	49.29 (1.20)	56.94 (0.79)	60.87 (0.37)	66.94 (0.44)	68.72 (0.74)
Ours	64	16,448	50.71 (0.71)	58.65 (0.76)	62.05 (0.39)	68.03 (0.41)	70.45 (0.45)

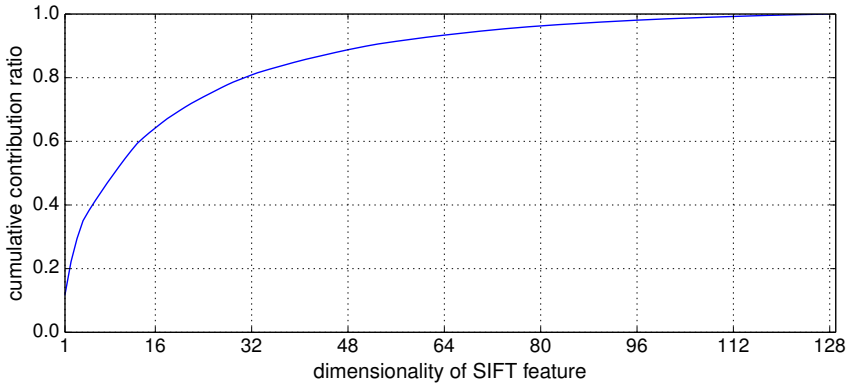


FIGURE 3.2: Cumulative contribution ratio of the eigen values of the PCA model fitted to the local descriptors.

We verified the robustness of our approach with the dimensionality reduction of the SIFT descriptors. The dimensionalities of the reduced SIFT feature were used 4, 8, 12, 16, 20, 24, 28 and 32. The codebook sizes were used 32, 64 and 256. For evaluation of the recognition performance, we used randomly selected 30 images from all the categories for training images, the rest images were used for test. The recognition rate was an average of 10 trials. Fig. 3.3 shows the experimental result. The best performance in each codebook size was recorded at 32-dimensional SIFT feature. Overall the recognition rate was increased with increasing the dimensionality of SIFT descriptors. In addition, on the whole, the image feature generated by the large codebook recorded higher recognition rate in the same dimensionality of the SIFT feature. In the case of same dimensionality image feature, with the small codebook and the high-dimensionality SIFT feature, the recognition rate became high. However, the case of 576- dimensionality image feature ($K = 32$, $D' = 4$) achieved higher the recognition rate than 2,304-dimensionality image feature ($K = 256$, $D' = 4$). The best performance of our approach ($K = 256$, $D' = 32$) achieved almost the same recognition rate as the FV encoding ($K = 256$, $D' = 80$). In comparison of our approach and the FV encoding, the difference between the recognition rates was 0.08% in less than the half dimensionality of image feature.

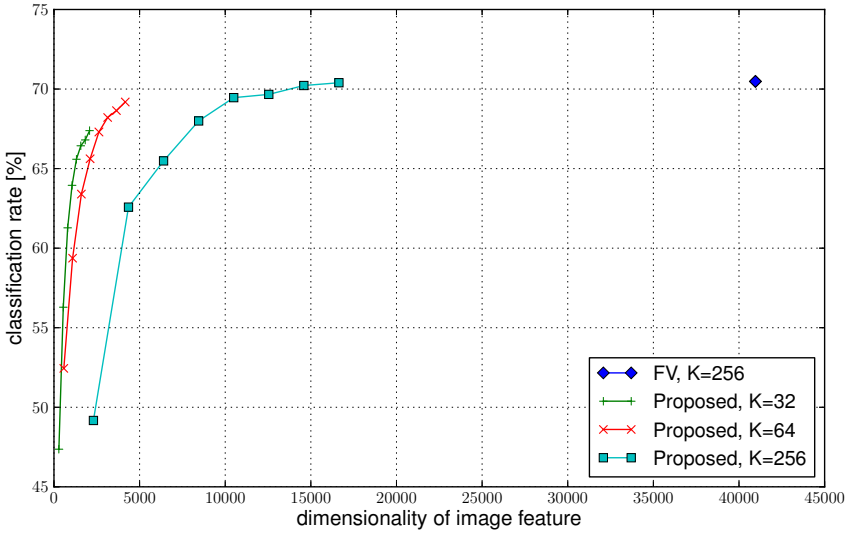


FIGURE 3.3: Evaluation of robustness of our approach for the dimensionality reduction of SIFT features. The baseline is the FV performance, denoted as "FV, K=256", reported in [7]. It is evaluated with 80-dimensional SIFT features and a codebook of $K = 256$ vocabularies, where the dimensionality of image features is $40,960$. Our proposed approach was evaluated with the three codebook sizes of $K = \{32, 64, 256\}$, respectively denoted as "Proposed, K=32", "Proposed, K=64", and "Proposed, K=256".

References

[1] Thomas Berg et al. "Birdsnap: Large-Scale Fine-Grained Visual Categorization of Birds". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*. IEEE, 2014, pp. 2019–2026.

[2] Hideki Nakayama. "Augmenting descriptors for fine-grained visual categorization using polynomial embedding." In: *ICME*. IEEE Computer Society, 2013, pp. 1–6.

[3] Keiji Yanai, Takuma Maruyama, and Yoshiyuki Kawano. "A Cooking Recipe Recommendation System with Visual Recognition of Food Ingredients". In: *iJIM* 8.2 (2014), pp. 28–34.

[4] James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Norwell, MA, USA: Kluwer Academic Publishers, 1981.

[5] Li Fei-Fei, Rob Fergus, and Pietro Perona. "Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories". In: *Comput. Vis. Image Underst.* 106.1 (Apr. 2007), pp. 59–70.

[6] Koby Crammer et al. "Online Passive-Aggressive Algorithms". In: *J. Mach. Learn. Res.* 7 (Dec. 2006), pp. 551–585.

[7] Lorenzo Seidenari et al. "Local Pyramidal Descriptors for Image Recognition". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 36.5 (2014), pp. 1033–1040.

Chapter 4

Analysis of Characteristics of Codebook-based Approaches

4.1 Introduction

Codebook-based image representations are useful approaches to recent image recognition problems, such as generic objects, more complex domains, and image retrieval. The most advantage is that it is easy to control recognition performance and practical computational costs by manipulating the codebook size. As introduced in the previous chapter, the dimensionality of image signatures is usually defined by the local feature dimensionality D and the codebook size K. The local feature dimensionality depends on the algorithm, for example, 128-dimensional for the original SIFT framework [1]. On the other hand, since the codebook size is a hyper-parameter in codebook-based image representation approaches, it can be increased as much as the limitation of computational cost. Moreover, it has been reported that increasing the codebook size frequently leads to improve recognition performance.

The FV, which is the state-of-the-art in the codebook-based image representation uses a mixture of Gaussians as the codebook. It is empirically known that the Gaussians are easy to overfit to training samples. Due to the overfitting, some Gaussians represent the same distribution or disappear.

This is an inappropriate property in image representation.

This chapter first analyzes the influence of the overfitting on recognition performance. After that, we show how to measure the quality of codebook from the model parameters. The main contribution of this chapter is to statistically analyze the relationship between codebook and recognition performance and present an indicator to quantitatively measure the quality of codebook.

4.2 Dependency of Model Parameters in Image Representation Approaches

4.2.1 Comparison of Codebook-based Image Representation Approaches

Here, we evaluate the following codebook-based image representation approaches, the FV and the VLAD. As mentioned in chapter 2, these approaches encode different statistics: frequency, mean, and variance in the FV, and only mean in the VLAD. In this section, these are notated as follows:

- **FV($\gamma + \mu + \sigma$)**: The FV signatures including full components.
- **VLAD**: The original VLAD signatures.

In order to compare these approaches under fair conditions, the following additional approaches are also evaluated:

- **FV(μ)**: The FV signatures represented by only mean components. Their dimensionality is also the same as VLAD signatures.
- **VLAD+PN**: The VLAD signatures to which the power normalization [2] is applied.

The above approaches were evaluated on the Caltech101 datasets [3], which is the most popular dataset in generic object recognition. The Caltech101 consists of 9,145 images from 101 generic object categories and one background category. Each category contains about 40 to 800 images.

To extract local descriptors, the SIFT [1] framework was used with the dense sampling strategy and PCA whitening. For each image, SIFT descriptors were extracted from the four scales 16, 22, 33, 44 pixels from the intersection of the dense grid with 8 pixels interval. Then, the extracted SIFT descriptors were reduced to 80-dimensional descriptors.

To construct a codebook, training samples were SIFT descriptors extracted from 510 images that were a set of randomly selected five images from each category. The termination criterion of the EM algorithm was that the number of iterations reaches to 30 times. The following codebook sizes were evaluated: 16, 32, 64, 128, and 256. In the GMM, each covariance matrix was assumed to be a diagonal matrix.

As a linear discriminant model, we used the SVM was used. The recognition performance was measured as an average of five trials of independent training and testing sets. The following numbers of images per category were used as the training set: 5, 10, 15, 20, 30. The rest was used as the testing set. The hyper-parameter C was optimized by 5-fold cross validation with the training set.

Figure 4.1 shows the recognition performances of the FV($\gamma + \mu + \sigma$), VLAD, FV(μ), and VLAD+PN regarding the number of training images per category.

FV($\gamma + \mu + \sigma$) achieved the best performance because it captures rich information compared with the others. For example, when the codebook size is $K = 256$, FV($\gamma + \mu + \sigma$) signatures have 41,216 dimensions, and the others have 20,480 dimensions. According to the results of VLAD and VLAD+PN, the power normalization improved about 3% recognition accuracy. In the case of comparing FV(μ) and VLAD+PN, where both approaches captured

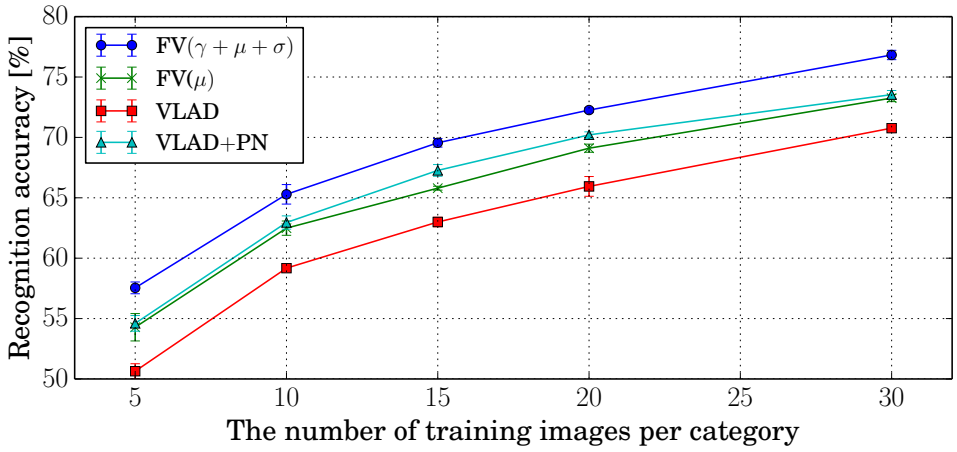


FIGURE 4.1: Comparison of recognition performances with respect to the statistics. "**FV**($\gamma + \mu + \sigma$)" is the fully encoded FV signature; "**FV**(μ)" denotes the FV signatures with only mean component; "**VLAD**" is the original VLAD signature without the power-normalization; "**VLAD**+**PN**" denotes the VLAD signature with the power-normalization.

the same statistics and were applied the power normalization, VLAD+PN significantly outperformed FV(μ).

We next focus on their model parameters. Figure 4.2 shows the distribution of the prior probabilities. Each prior probability value indicates the probability that a given sample is assigned to the corresponding cluster. The k-means does not have the prior probability, they were estimated as:

$$w_k = \frac{1}{T} \sum_{t=1}^{T} q_{t,k}, \tag{4.1}$$

which is the same way as (2.4).

The average $\overline{w}$ of prior probabilities is constant regarding the codebook size K. So that, the standard deviation of prior probabilities possibly be used to measure how much the codebook overfits to training samples. Table 4.1 shows the standard deviation of prior probabilities regarding the codebook size.

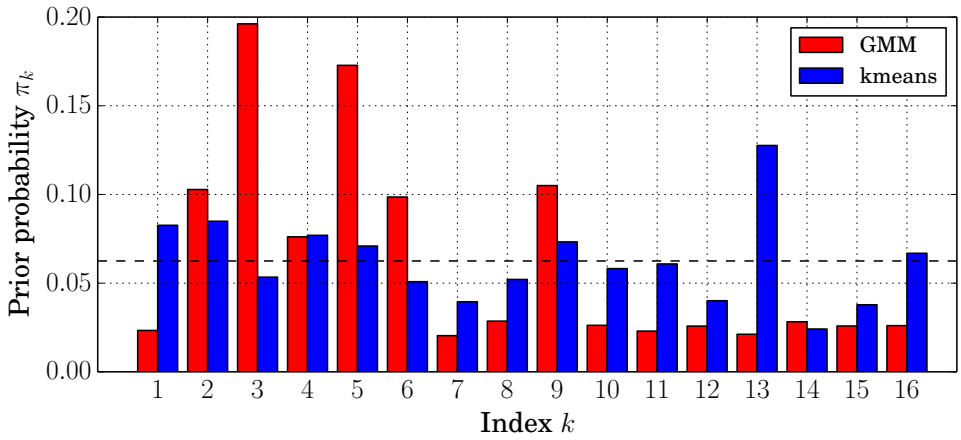


FIGURE 4.2: Comparison of prior probabitlity distribution of the k-means and the GMM with $K = 16$. The dashed line denotes the mean of prior probabilities $\overline{w} = 1/16 = 0.0625$, where the mean is always equal to $1/K$ because of the probabilistic constraint $\sum_{k=1}^{K} w_k = 1$.

TABLE 4.1: Comparison of the standard deviation of the prior probabilities regarding the codebook size. The term of "Relative ratio" indicates the relative spread of the GMM to the k-means.

	Codebook size K				
Method	**16**	**32**	**64**	**128**	**256**
GMM	0.03830	0.02257	0.01665	0.00940	0.00779
k-means	0.02921	0.01511	0.00800	0.00391	0.00188
Relative ratio	1.31119	1.49371	2.08125	2.40409	4.14362

4.3 Statistical Analysis of Relationship of Model Parameters and Recognition Performances

The two scaling parameters, γ and ν, are added to the probability density function, which is to control the standard deviation of prior probabilities, in the same manner as in [4] as:

$$p^*(\boldsymbol{x}_t; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}_k|^{1/\gamma}}} \exp\left[-\frac{1}{\nu}(\boldsymbol{x}_t - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1}(\boldsymbol{x}_t - \boldsymbol{\mu}_k)\right], \quad (4.2)$$

and the posterior probabilities are difined as:

$$q^*_{t,k} = \frac{w_k p^*(\boldsymbol{x}_t; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_{j=1}^K w_j p^*(\boldsymbol{x}_t; \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}. \quad (4.3)$$

To construct a codebook, these are used with the GMM miximization step.

The range of the scaling parameters was defined as $\{2^i : i = -5, -3, 1, 1, 3, 5\}$, where the case of $\gamma = 2$ and $\nu = 2$ is the same as the original probability density funciton. So that, we used the previous experiment results as the case of $\gamma = 2$ and $\nu = 2$. The other parameters were set to the same as the previous experiment.

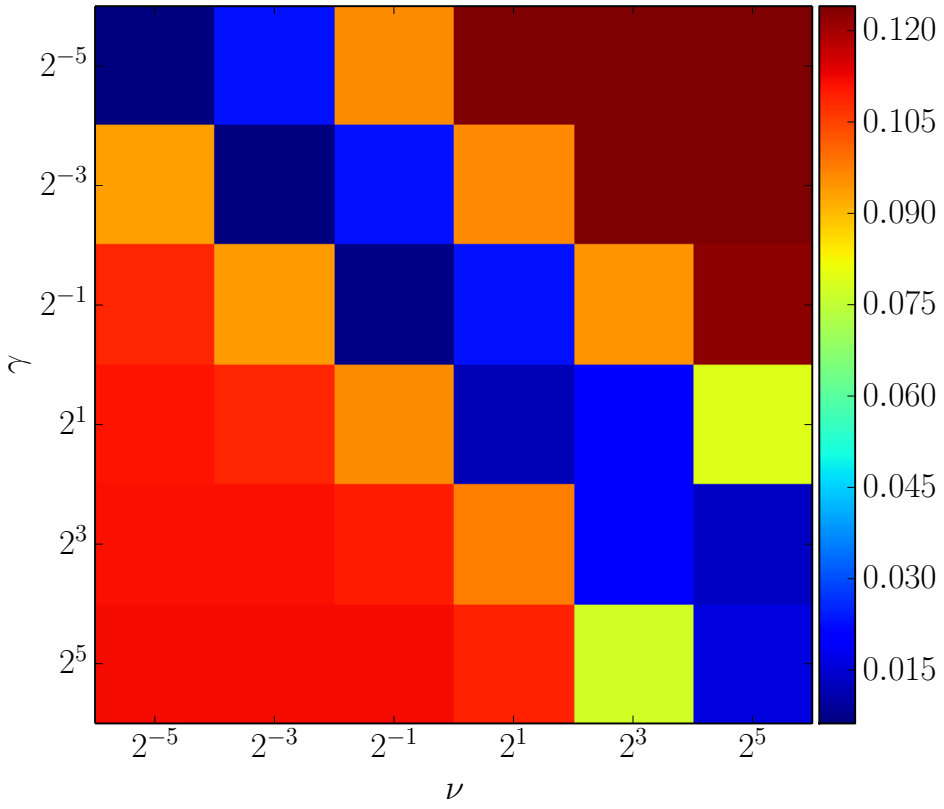
Fig. 4.3 shows the relationship with the scaling parameters. The relationships were measured by the Pearson's correlation analysis and the blue line indicates the line of best fit by a least square method. Fig. 4.3(A) shows the relationship between the standard deviation of prior probabilities and recognition performance, the correlation coefficient was strong negative -0.62. Table 4.2 shows the correlation coefficient with respect to the number of training images. Each correlation coefficient was evaluated with 36 samples, which are 6 times 6 variations for the two scaling parameter combinations. Each sample is the average recognition accuracy over 5 trials using different training and testing images. According to the correlation coefficient table, we

TABLE 4.2: Relationship between the prior probability distribution and recognition performance.

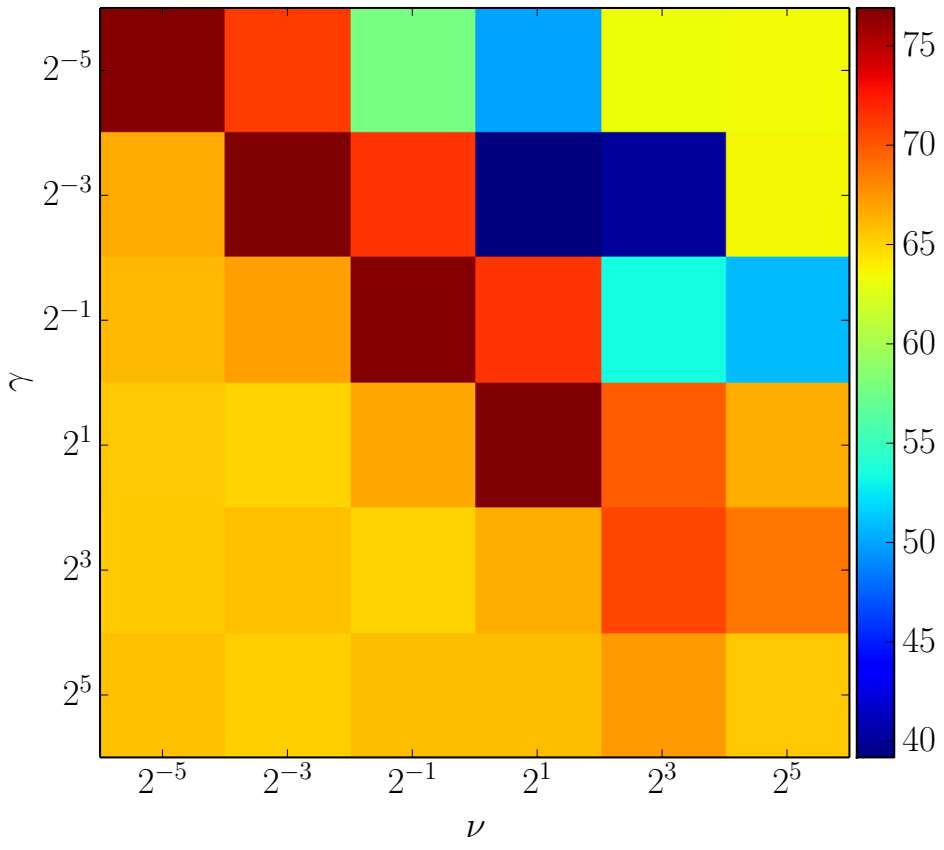
	The number of training images per category				
	5	**10**	**15**	**20**	**30**
Correlation coeficient	-0.58	-0.60	-0.61	-0.62	-0.62

observed $|-0.58| > 0.42$ for significance level $p < 0.01$ at least, where the number of training images is 5 which is the smallest correlation in Table 4.2. Therefore, the relationship between the standard deviation of prior probabilities and recognition accuracies has a significant correlation. Fig. 4.4(B) shows the relationship between the standard deviation of prior probabilities and the sparseness of parameterized FV signatures. The correlation coefficient was especially strong negative -0.88.

When using GMM, it is hard to increase image recognition rate because it does not have a regularization term. However, the results described above suggest that the parmeterized FV has a potential to improve image recognition accuracy.

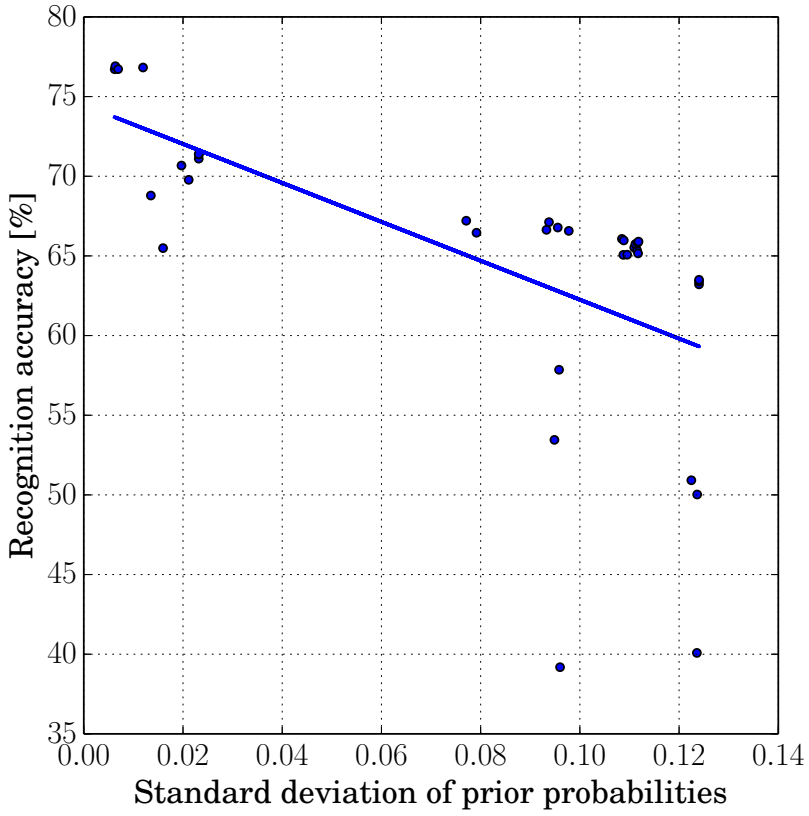


(A) Standard deviation of prior probabilities.

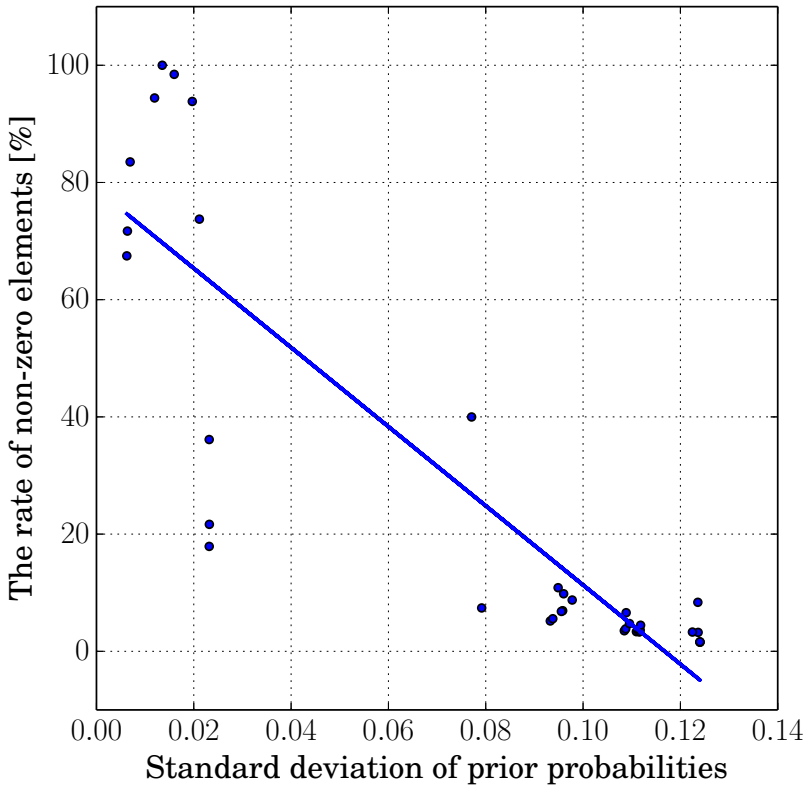


(B) Recognition accuracy with the parameterized FV.

FIGURE 4.3: Effect of the scaling parameters.



(A) Recognition accuracy.



(B) Sparseness of the parameterized FV signatures.

FIGURE 4.4: Relationship regarding the standard deviation of prior probabililties.

As an additional analysis, we used Birds dataset published by Ponce Group, other than Caltech101. This dataset consists of 6,000 images from 6 bird species. To analyze the recognition performances on this dataset, we used some different conditions for local descriptors and codebook sizes as follows:

- SIFT descriptors were extracted from four scale levels {16, 20, 24 and 28 pixels}.
- Four kinds of Color-SIFTs [5]: Gray, RGB, L*a*b, and Opponent.
- The codebook sizes: $K \in \{16, 32, 64, 128\}$.

The scale levels are set up other parameters than Caltech101 by those concepts. While Caltech101 consists of generic object images, which each image probably be a set of primitive parts. The images of Ponce Group Birds are visually similar because all categories are bird species. For those target images, we empirically used relatively smaller scale levels than the setup in Caltech101 to focus more local patterns of the bird images. Additionally, for the same reason, we used variants of color SIFT. The other parameters except above were the same as the previous experiment.

Fig. 4.5 shows the parameter space contours of the recognition performances and their maximum relatively improved accuracies compared with the baselines are shown in Table 4.3.

Each contour has an individual scale. For example, the recognition accuracies increase from the left to the right column because the image signatures have precise information as the codebook size increases. Adding color information also has a potential to improve recognition accuracies, showed in the second and subsequent rows, compared with the Gray-SIFT in the first row. To find optimal values of the two scaling parameters for setup and in a

TABLE 4.3: The relative improvements of the best accuracies compared with the corresponding baselines ($\gamma = 2$, $\lambda = 2$).

	Codebook size K			
Color	16	32	64	128
Gray	+0.14	+3.53	+2.06	+1.60
L*a*b	+3.47	+0.60	+3.26	+0.53
RGB	+4.00	+1.06	+0.47	+1.40
Opponent	+1.87	+2.33	+1.60	+1.00

specific application, such as codebook size, local feature type and characteristics of target domains, an optimization algorithm is needed to explore the corresponding parameter space.

All of the matrices in Fig. 4.5 have a similar trend to the result on Caltech101 dataset as in Fig. 4.3(B). The parameter spaces are not simple concave or convex. The diagonal part with smaller values than the baseline ($\gamma < 2$ and $\lambda < 2$) shows high recognition accuracies. Most interesting point is that these trends are not depended on the kinds of colors for SIFT descriptors and target domains (generic objects and bird species). These trends may be a constraint for restricting a searching space of optimization algorithms.

Table 4.4 shows the relationships between the standard deviation of prior probability and recognition accuracy for colors, codebook sizes and the numbers of training images {30, 40 and 50 images per category}. The relationships still showed strong negative correlations.

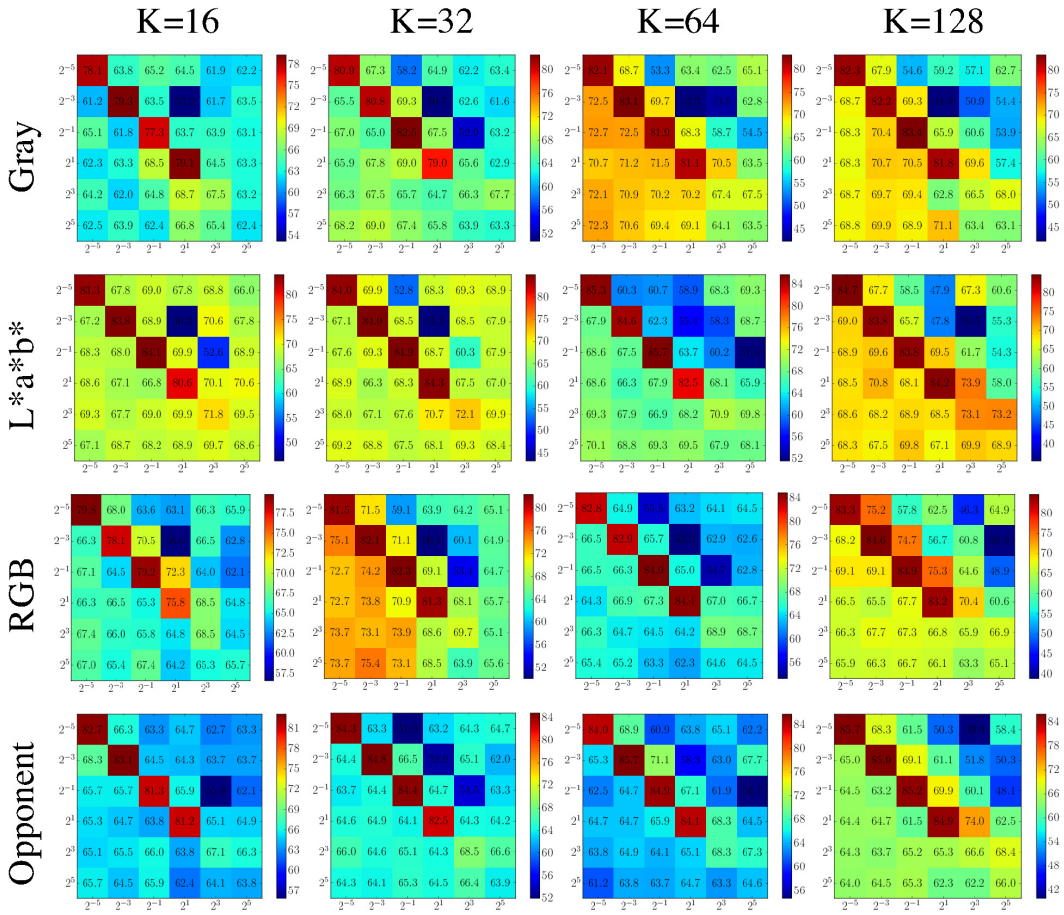


FIGURE 4.5: Parameter space of the recognition accuracies (%) on the Ponce Group Birds dataset for colors (in each row) and codebook sizes (from the top to the bottom column) with the 50 training images per category, in the same manner as Fig. 4.3(B). Here, horizontal and vertical axes are the scaling parameters (γ and ν) respectively.

TABLE 4.4: The correlation coeficients for colors, codebook sizes and training images per category.

Color	K	The number of training images 30	40	50
Gray	16	-0.66	-0.62	-0.60
	32	-0.70	-0.66	-0.65
	64	-0.61	-0.58	-0.57
RGB	16	-0.61	-0.61	-0.61
	32	-0.67	-0.66	-0.69
	64	-0.60	-0.60	-0.61
Lab	16	-0.64	-0.63	-0.63
	32	-0.61	-0.62	-0.58
	64	-0.52	-0.53	-0.52
Opponent	16	-0.73	-0.73	-0.76
	32	-0.61	-0.64	-0.67
	64	-0.52	-0.56	-0.59

References

[1] David G. Lowe. "Distinctive Image Features from Scale-Invariant Keypoints". In: *Int. J. Comput. Vision* 60.2 (Nov. 2004), pp. 91–110.

[2] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. "Improving the Fisher Kernel for Large-scale Image Classification". In: *Proceedings of the 11th European Conference on Computer Vision: Part IV*. ECCV'10. Heraklion, Crete, Greece: Springer-Verlag, 2010, pp. 143–156.

[3] Li Fei-Fei, Rob Fergus, and Pietro Perona. "Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories". In: *Comput. Vis. Image Underst.* 106.1 (Apr. 2007), pp. 59–70.

[4] Hidetomo Ichihashi et al. "Fuzzy c-means classifier with particle swarm optimization". In: *2008 IEEE International Conference on Fuzzy Systems (IEEE World Congress on Computational Intelligence)*. 2008, pp. 207–215.

[5] Koen van de Sande, Theo Gevers, and Cees Snoek. "Evaluating Color Descriptors for Object and Scene Recognition". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 32.9 (Sept. 2010), pp. 1582–1596.

Chapter 5

Optimization Framework for Codebook Construction with the Quantitative Measurement

5.1 Introduction

Chapter 4 showed the quantitative measurement from the perspective of prior probabilities for codebook quality and conducted the analysis with the PDF function parameterized by the two scaling factors. To validate the scaling parameters, the grid searching algorithm was used. Despite the fact that the scaling parameters are continuous, the grid searching algorithm exhaustively explores a discrete parameter space to construct an appropriate codebook.

This chapter presents a clustering framework, which directly optimizes an objective based on the quantitative measurement. We first describe the detail of the proposal frameworks, named *Prior Probability-Oriented Clustering* (PPOC), and the definitions of the objectives, which are alternatives to the k-means and the GMM. Then, the characteristics of the proposal frameworks are evaluated with synthetic clustering datasets. After that, the proposal frameworks are applied to image recognition problems.

5.2 The Prior Probability-Oriented Clustering

The general objective function of the proposal is defined as the following equation containing the two terms:

$$J_{\mathrm{PPOC}} = \sum_{k=1}^{K} |w_k - \overline{w}| + \lambda \frac{1}{T} \sum_{t=1}^{T} d(\boldsymbol{x}_t; \boldsymbol{\Theta})^2, \tag{5.1}$$

where the first term is the main objective of our proposal, which measures the approximate variance of prior probabilities. $\overline{w}$ denotes the average of prior probabilities, where it equals to $1/K$ due to the probabilistic constraint. The second term is a regularization term that gives unique solution for the main objective (5.1). When clustering a small set of training samples, the solution space might be discrete, in other words, small changes of candidates do not effect to the objective value and give the same objective values. The second term also serves to smooth a parameter space. λ is a weighting factor to decide the importance of the second term. In our concepts, the weighting factor is set to a small value because the main objective must be emphasized.

To optimize the objective (5.1), a black-box optimization framework that does not require any additional information, such as derivation, is used. The general procedure is as follows:

Step 1: initialize mean vectors by k-means++

Step 2: repeat

- the other parameters except to the mean vectors are estimated if needed
- evaluate the objective value of current model parameters by Eq. (5.1)
- update mean vectors by a black-box optimizer

Step 3: until the maximum number of the evaluations reaches

5.2.1 Hard Objbective

In this situation, our proposal aims to minimize the main objective while reducing the quantization error between training samples and the cluster candidates. The k-means model does not have prior probabilities as described in the section 4.2, these are calculated as in (4.1).

In the hard objective of our proposal, the regularization term measures the quantization error between the clustering samples and the cluster candidates. The quantization error is defined in the same way to the k-means objective. So that, $d(\boldsymbol{x}_t; \Theta)$ of the regularization term is defined as:

$$d(\boldsymbol{x}_t; \boldsymbol{\Theta}) = \sum_{k=1}^{K} q_{t,k} \|\boldsymbol{x}_t - \boldsymbol{\mu}_k\|. \tag{5.2}$$

The overall procedure of the proposed clustering framework with hard objective is as follows:

Step 1: initialize mean vectors by k-means++

Step 2: repeat

(a) predict assignment probabilities $q_{t,k}$ by Eq. (2.2)

(b) compute prior probabilities by $w_k = \frac{1}{T} \sum_{t=1}^{T} q_{t,k}$

(c) evaluate main objective term shown in Eq. (5.1)

(d) compute regularization term shown in Eq. (5.2)

(e) update mean vectors by a black-box optimizer

Step 3: until the maximum number of the evaluations reaches

5.2.2 Soft Objective

In order to estimate Gaussians from the mean vectors, the parameter estimation procedure is approximated. The assignment probabilities for each sample are first estimated in the same way as the k-means prediction (2.2) with

cluster candidates. Then, the other model parameters (the prior probabilities w_k and the covariance matrices $\mathbf{\Sigma}_k$) are estimated from the mean candidates and the clustering samples. This approximated procedure can be seen as the EM iteration of only one step in the GMM with the k-means initialization.

The regularization term is defined as:

$$d(\boldsymbol{x}_t; \mathbf{\Theta}) = \sum_{k=1}^{K} q_{t,k} (\boldsymbol{x}_t - \boldsymbol{\mu}_k)^\top \Sigma_k^{-1} (\boldsymbol{x}_t - \boldsymbol{\mu}_k), \tag{5.3}$$

where $\mathbf{\Sigma}_k$ is the k-th estimated covariance matrix. The overall procedure of the proposed clustering framework with soft objective is as follows:

Step 1: initialize mean vectors by k-means++

Step 2: repeat

(a) predict hard assignment probabilities $q_{t,k}$ by Eq. (2.2)

(b) estimate the other parameters, w_k and $\mathbf{\Sigma}_k$ in the same manner as Eq. (2.6) and Eq. (2.8)

(c) evaluate main objective term shown in Eq. (5.1)

(d) compute regularization term by Eq. (5.3)

(e) update mean vectors by a black-box optimizer

Step 3: until the maximum number of the evaluations reaches

5.3 Numerical Analysis on Synthetic Datasets

To evaluate quantitative and qualitative characteristics of our proposal against to the popular clustering approaches, the synthetic clustering datasets [1], the A-sets [2] and the S-sets [2, 3], are used. Both datasets have three sub-sets of clustering samples, named A1, A2, and A3 for the A-sets, and S1, S2, and S3 for the S-sets. Table 5.1 shows the statistics of these sub-sets.

TABLE 5.1: Statistics of the A-sets and the S-sets.

Dataset	The number of total samples	The number of clusters
A-sets (A1)	3,000	20
A-sets (A2)	5,250	35
A-sets (A3)	7,500	50
S-sets (S1)	5,000	15
S-sets (S2)	5,000	15
S-sets (S3)	5,000	15

TABLE 5.2: Comparison of the optimized objective values regarding the optimization algorithms on the A-sets.

Dataset	k-means	NM	SBPLX	COBYLA	NEWUOA	AUGLAG
A-sets (A1)	0.0167	0.0020	**0.0007**	0.0040	0.0060	0.0040
A-sets (A2)	0.0114	0.0038	**0.0015**	0.0038	0.0038	0.0042
A-sets (A3)	0.0396	0.0045	**0.0019**	0.0029	0.0053	0.0037

5.3.1 Comparison of Optimization Algorithms

We first explore which optimization framework is better to our objective (5.1) with the A-sets and the S-sets. As black-box optimization frameworks, the following algorithms are evaluated: the Nelder-Mead (NM) [4], the Subplex (SBPLX) [5], the COnstrained BY Linear Approximation (COBYLA) [6], the NEWUOA [7], and the AUGmented LAGrangian algorithm (AUGLAG) [8, 9], which have been implemented in the NLOPT library [10].

For all the optimization algorithms with our proposal, the optimized objective values were significantly better than the baseline results. The SBPLX

TABLE 5.3: Comparison of the optimized objective values regarding the optimization algorithms on the S-sets.

Dataset	GMM	NM	SBPLX	COBYLA	NEWUOA	AUGLAG
S-sets (S1)	0.5424	0.0220	0.0107	**0.0105**	0.0276	0.0131
S-sets (S2)	0.5636	0.0185	**0.0089**	0.0104	0.0217	0.0107
S-sets (S3)	0.4632	0.0088	**0.0016**	0.0104	0.0031	0.0092

showed the minimum values in most cases. For the soft clustering on the S1, the COBYLA gave the best value (0.0105), but it is close to the SBPLX result (0.0107).

If the optimization frameworks ideally optimize, a large number of clusters is considered to lead to decreasing our objective value (5.1) because the mean of prior probabilities is constant with respect to the number K of clusters.

In the results of the hard clustering on the A-sets shown in Table 5.2, the COBYLA and the AUGLAG show this trend, but the others construct more deviating clusters as the number of clusters increases. These suggest that our proposal with the hard objective might not effective for a large set of clustering samples or large cluster sizes.

In the S-sets, the results suggest that our proposal with the soft objective has effectively constructed Gaussians for samples spatially complicatedly distributed. For the S2 and the S3, the SBPLX showed totally better objective value compared to the other optimization algorithms.

5.3.2 Qualitative Comparison of Constructed Clusters

Figure 5.1 shows the mean vectors optimized by the SBPLX on the A-sets. The red cross and the yellow cross respectively indicate the mean positions constructed by our proposal and the k-means. The black circles show the sample distribution. Despite that our proposal mainly aims to equalize the prior probability distribution, many positions of our proposal were close to the k-means clusters. One of that reasons is that the number of samples in each cluster is 150 in all the sub-sets of the A-sets; minimizing the distribution of prior probabilities is similar to minimizing the quantization error. Our proposal with hard objective constructs the clusters that are similar to the k-means, while finely tuning their positions based on the main objective term

of (5.1).

Figure 5.2 shows the estimated Gaussians on the S-sets. The GMM constructed the fewer Gaussians than the designated cluster size; three Gaussians on the S1 and the S2, and seven Gaussians on the S3 were converged to the same positions of other Gaussians or were lost. It is due to the over-fitting and is cause to increase the deviation of the prior probabilities. As the distribution of clustering samples becomes more complicated, this tendency is considered to be noticeable. On the other hand, the results of our proposal show the fully distributed 15 Gaussians for the S-sets. For the S1 and the S2, some Gaussians were not appropriate to the sparsely distributed samples. For the complicatedly distributed samples such as the S3 in the figure, the Gaussians seem to appropriately express the sample distributions. These characteristics are consistent with the discussion in the optimization framework comparison shown in Table 5.3; our proposal possibly construct the better Gaussians compared to the GMM as the sample distribution becomes more complicated.

In the codebook-based image representation, a codebook is constructed by clustering a lot of spatially complicatedly distributed samples, usually hundreds of thousands or more. In addition, the codebook size is an important factor to decide the trade-off between computational complexity and recognition performance. Our proposal with the soft objective is expected to avoid over-fitting of the GMM and to effectively use all the constructed Gaussians.

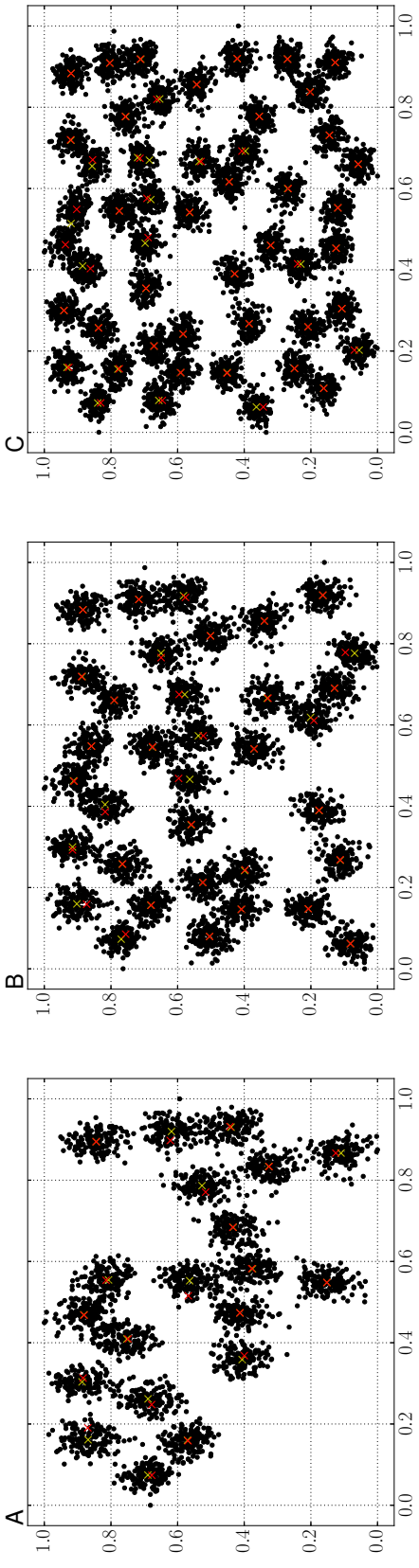


FIGURE 5.1: Qualitative comparison of generated clusters on the A-sets. The yello crosses denote the cluster positions of the k-means and the red crosses denote the cluster positions of the PPOC with the hard objective.

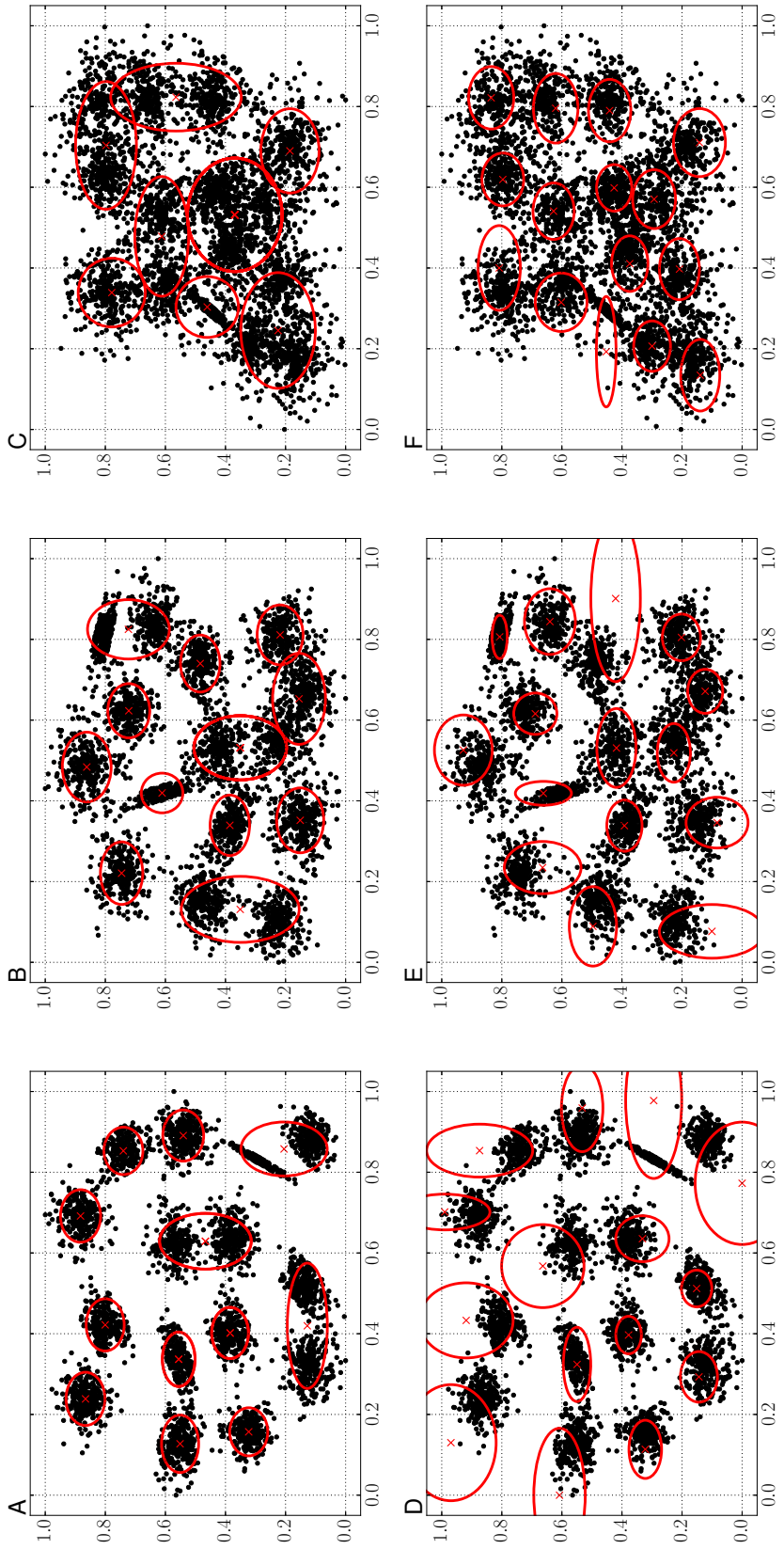


FIGURE 5.2: Qualitative comparison of generated clusters on the S-sets. The yello crosses denote the cluster positions of the GMM and the red crosses denote the cluster positions of the PPOC with the soft objective.

5.3.3 Effect of Weighting Factor

Figure 5.3 shows the trends of objective values with respect to the weighting factor λ on the A-sets and the S-sets. For the results on the A-sets in figure 5.3 (A–C), the values of the regularization term decrease as the number of clusters increase because the dispersion of samples in each cluster is small in order to A1, A2, and A3 in figure 5.2. For the S-sets in figure 5.3 (D–F), the values of the regularization term increase in order to S1, S2, and S3 because of the increase of the spatial complexity.

On the whole trends in figure 5.3 (A–F), there was no clear trend of the main objective regarding the weighting factor. The tendency to the regularization term is relatively intuitive, in particular for the soft objective, the quantization error decreases as the weighting coefficient increases.

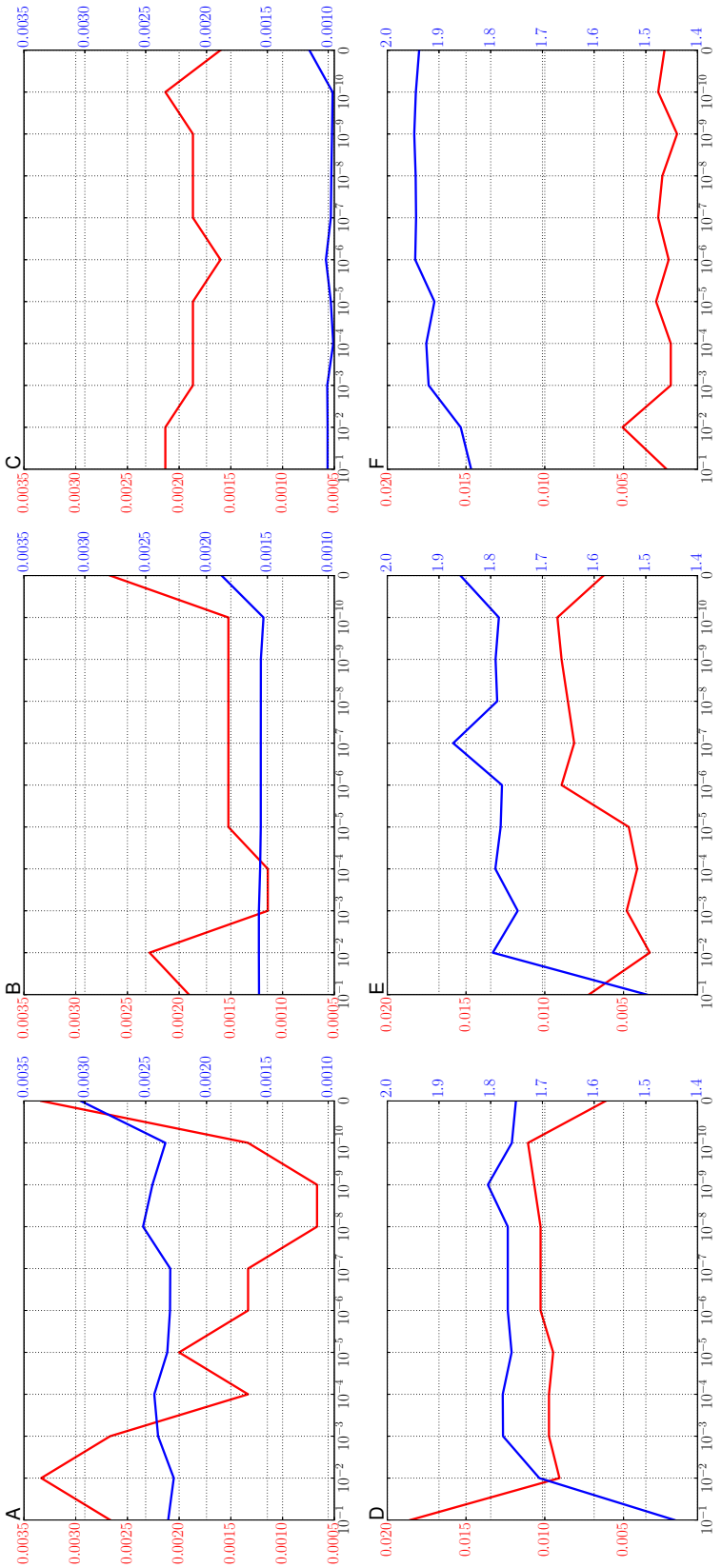


FIGURE 5.3: Trend of the optimized objective values regarding the weighting factor. (A–C) The trend of the hard objective on the A1, A2, and A3. (D–F) The trend of the soft objective on the S1, S2, and S3. The red line shows the main objective values and the blue line shows the regularization value without applying weight factor λ.

5.4 Appliation to Image Recognition

This section evaluates our proposal on image recognition tasks with the following image datasets: *Birds* [11] and *Butterflies* [12] provided by Ponce Group.

The Birds dataset consists of 600 images categorized into six bird species, where each category has 100 images. The Butterflies dataset has 619 images of seven different butterflies. Each category has about 40 to 130 images. The above two datasets are composed of visually similar images.

In the experiments with the above datasets, we used the same parameter setup except for numbers of training images to construct a codebook and a discriminant model.

We used SURF [13] as the local feature framework. To extract SURF features, we followed the dense sampling strategy [14], which SURF features were described from the intersection points of the lattice of six pixels intervals, with multiple scale regions, 16, 20, 24, and 28 pixels for each point, where each image was resized so that the long side was 300 pixels. Each SURF feature was projected to 8-dimensional space by the Principle Component Analysis before constructing a codebook and encoding an image feature [15].

To construct a codebook, clustering samples were the SURF features extracted from 10 images from each category for the Birds and 5 images from each category for the Butterflies, where we decided about 10% of the smallest number of images of their categories. The codebook sizes of the five different patterns $K = \{16, 32, 64, 128, 256\}$ were used. The termination criterion for the k-means and the GMM was set to 30 iterations because they do not converge sometimes. For our proposal, the termination criterion was set to 2,000 evaluations of the objective function. Gaussians of the GMM and our proposal with soft objective were assumed to diagonal covariance. The weighting factor of our objective was set to $\lambda = 10^{-9}$. The k-means and ours with

hard objective were used for the VLAD encoding, and the GMM and ours with soft objective were used for the FV encoding.

The SVM with the linear kernel, implemented in [16], was used as a discriminant model. The number of training images for each category was $\{30, 40, 50\}$ for the Birds and $\{20, 30, 40\}$ for the Butterflies. The training images were randomly selected, and the rest images were used for the test. The recognition accuracy was the ratio of the number of correctly recognized images for the number of test images. We measured by the average over five different training and test images.

Fig. 5.4 and Fig. 5.5 respectively show the average recognition accuracies of the VLAD and the FV on the Birds dataset. For the results of Fig. 5.4, the baseline, the VLAD with the k-means codebook, and the VLAD with our hard objective showed similar performances regardless of the parameters such as the number of training images and the codebook sizes. As discussed in the numerical analysis section, the hard objective mainly performs to finely tune mean positions, the k-means and our hard objective clustering potentially construct similar codebooks. Table 5.4 shows the objective values of the codebooks used in Fig. 5.4. When the codebook size is not greater than 64, the hard objective showed significantly better objectives compared with the k-means objectives. However, when the codebook size is greater than or equal to 64, they showed almost the same objectives. The k-means is possible to construct suitable clusters from the perspective of the variance of prior probabilities, regardless of the size of the clustering sample set or the codebook size, as shown in Fig. 5.4. The hard objective might difficult to effectively optimize codebook for large clustering sample set or large codebook sizes, as discussed in the qualitative comparison in the numerical section. On the other hand, the FV with our soft objective often showed better performances compared with the FV with the GMM codebook, especially when the codebook size is 128. When the codebook size was small, $K = 16$ and

$K = 32$, there is no significant difference of the recognition performances of the baseline and the FV with the soft objective. For the larger codebook size, the FV with the soft objective performed better accuracies. Moreover, our soft objective with a relatively larger codebook size was more effective for the case that training image set is smaller compared with the test image set. The highest mean recognition accuracy was achieved when the codebook size was 64, 128, and 128 respectively for 30, 40, and 50 training images per category. So that, an increase in the codebook size does not necessarily lead to improving recognition performance, the codebook size $K = 64$ or $K = 128$ might be enough for the Birds dataset. Table 5.5 shows the objective values of the codebooks used in Fig. 5.5. In contrast to the trend of the objective values of the hard objective, the soft objective could maintain the better values, shown in Table 5.9, even when the codebook size is increased. As with the discussions in numerical analysis, the soft objective is able to construct a suitable codebook, from the perspective of the variance of prior probability, even in image recognition tasks. When comprehensively comparing the results of the VLADs in Table 5.6 and the FVs in Table 5.7, the FV with our soft objective ($K = 64$) showed the best accuracy of 68.71 when the training images were 30 for each category. The FV with ours ($K = 128$) also showed the best accuracy as follows: 71.56 for 40 training images and 74.13 for 50 training images.

TABLE 5.4: The objective values of the k-means and ours with the hard objective with respect to the codebook size on the Birds.

	Codebook size				
Method	16	32	64	128	256
k-means	0.2210	0.1940	0.1923	0.1995	0.2082
ours (hard)	0.0071	0.0205	0.0895	0.1759	0.2126

TABLE 5.5: The objective values of the GMM and ours with the soft objective with respect to the codebook size on the Birds.

	Codebook size				
Method	16	32	64	128	256
GMM	0.4139	0.3248	0.2773	0.3240	0.3171
ours (soft)	0.0014	0.0297	0.0580	0.1268	0.1508

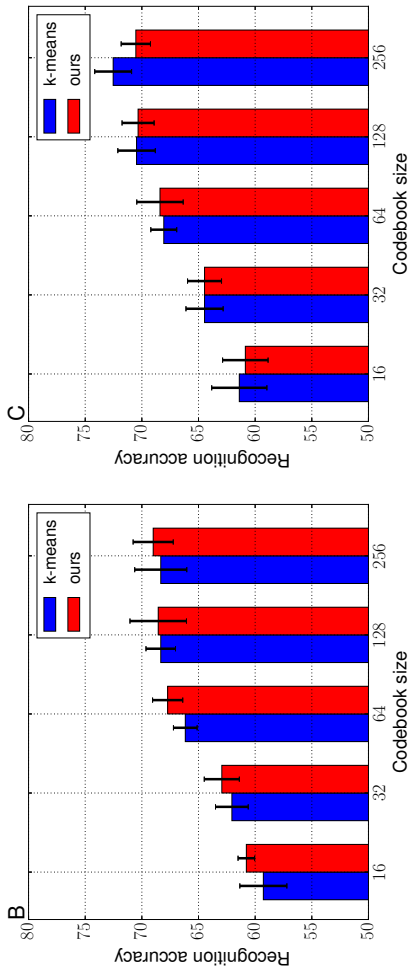


FIGURE 5.4: Recognition accuracies of the VLAD signatures with the k-means and the PPOC-hard codebooks on the PonceGroupBirds.

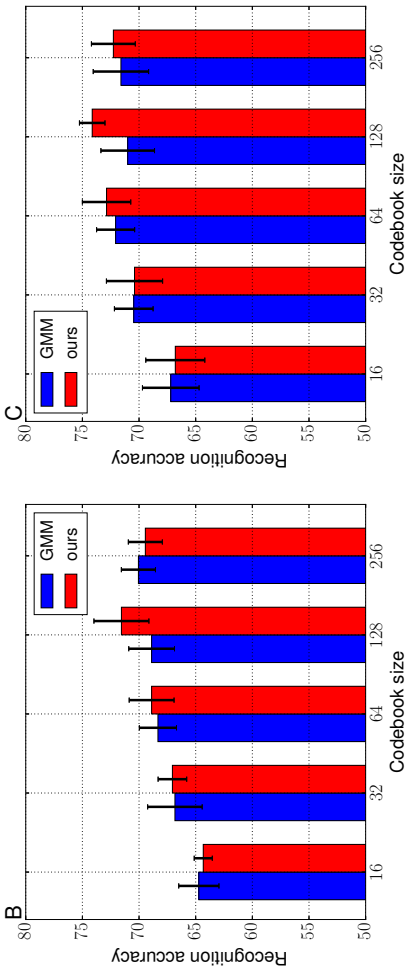


FIGURE 5.5: Recognition accuracies of the FV signatures with the GMM and the PPOC-soft codebooks on the PonceGroup-Birds.

TABLE 5.6: Recognition performance (mean accuracy ± standard deviation) of the VLADs with the k-means and ours (hard objective) codebooks on the Birds, corresponding to the Fig. 5.4.

	Codebook size K				
Method	16	32	64	128	256
30 images from each category, corresponding to Fig. 5.4 (A)					
k-means	56.43 ± 2.52	60.52 ± 0.98	64.62 ± 1.76	65.76 ± 1.97	65.62 ± 2.11
ours	55.67 ± 1.69	60.86 ± 1.43	65.90 ± 1.47	65.62 ± 1.19	65.33 ± 2.26
40 images from each category, corresponding to Fig. 5.4 (B)					
k-means	59.28 ± 2.08	62.06 1.44	66.17 ± 1.05	68.33 ± 1.30	68.33 ± 2.30
ours	60.78 ± 0.73	62.94 ± 1.55	67.72 ± 1.33	68.56 ± 2.49	69.00 ± 1.77
50 images from each category, corresponding to Fig. 5.4 (C)					
k-means	61.40 ± 2.43	64.47 ± 1.64	68.07 ± 1.14	70.47 ± 1.65	72.53 ± 1.63
ours	60.87 ± 2.01	64.47 ± 1.50	68.40 ± 2.05	70.33 ± 1.41	70.53 ± 1.29

TABLE 5.7: Recognition performance (mean accuracy ± standard deviation) of the FVs with the GMM and ours (soft objective) codebooks on the Birds, corresponding to the Fig. 5.5.

	Codebook size K				
Method	16	32	64	128	256
30 images from each category, corresponding to Fig. 5.5 (A)					
GMM	61.24 ± 1.99	64.24 ± 1.50	65.90 ± 1.39	63.90 ± 2.05	65.71 ± 1.03
ours	62.24 ± 1.92	63.14 ± 1.62	68.71 ± 2.63	67.00 ± 0.91	67.19 ± 1.85
40 images from each category, corresponding to Fig. 5.5 (B)					
GMM	64.72 ± 1.77	66.83 ± 2.41	68.33 ± 1.64	68.89 ± 2.02	70.06 ± 1.51
ours	64.33 ± 0.80	67.06 ± 1.26	68.89 ± 1.98	71.56 ± 2.43	69.44 ± 1.50
50 images from each category, corresponding to Fig. 5.5 (C)					
GMM	67.20 ± 2.50	70.47 ± 1.71	72.07 ± 1.68	71.00 ± 2.37	71.60 ± 2.44
ours	66.80 ± 2.60	70.40 ± 2.48	72.87 ± 2.14	74.13 ± 1.13	72.27 ± 1.94

Fig. 5.6 and Fig. 5.7 respectively show the average recognition accuracies of the VLAD and the FV on the Butterflies dataset. From the results in Fig. 5.6, the hard objective may deteriorate recognition performance when codebook size is smaller than or equal to 64. In addition, the objective values of the hard objective, shown in Table 5.8, were not enough optimized as with the case of the Birds dataset. For the results with the FV, the GMM and the soft objective showed similar performances when the codebook size is small. As with the numerical analysis, a smaller codebook size has less influence on the convergence of the Gaussians, and the GMM makes it easier to converge Gaussians to the same positions when the clustering samples is spatially complicatedly distributed and a codebook size is large. However, it improved recognition performances clearly when the codebook size is larger than 32, in all of the training images per category and lead to improve recognition performances when the codebook size was 256. In the case of comparing the results of the VLADs in Table 5.10 and the FVs in Table 5.11, the VLAD with the k-means ($K = 256$) showed best accuracy: 87.93 for 20 training images and 90.27 for 30 training images. On the other hand, for the 40 training images, the FV with ours ($K = 256$) showed the best accuracy of 91.33.

TABLE 5.8: The objective values of the k-means and ours with the hard objective with respect to the codebook size on the Butterflies.

	Codebook size				
Method	16	32	64	128	256
kmeans	0.1962	0.1580	0.1966	0.1828	0.2106
ours (hard)	0.0011	0.0201	0.1030	0.1915	0.1883

TABLE 5.9: The objective values of the GMM and ours with the soft objective with respect to the codebook size on the Butterflies.

	Codebook size				
Method	16	32	64	128	256
GMM	0.3810	0.3322	0.3407	0.2981	0.2961
ours (soft)	0.0018	0.0191	0.0557	0.1017	0.1352

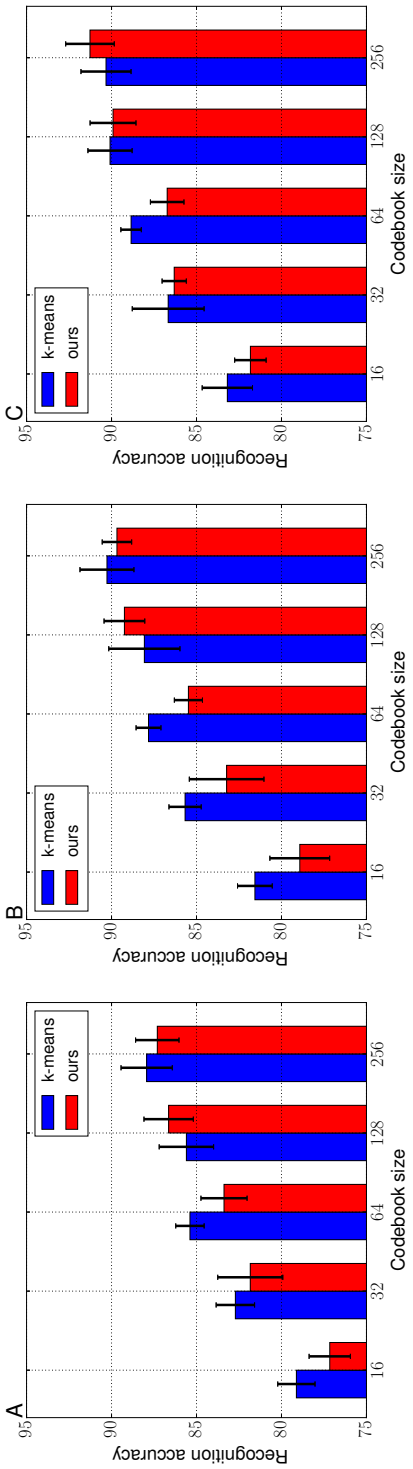


FIGURE 5.6: Recognition accuracies of the VLAD signatures with the k-means and the PPOC-hard codebooks on the PonceGroupButterfly.

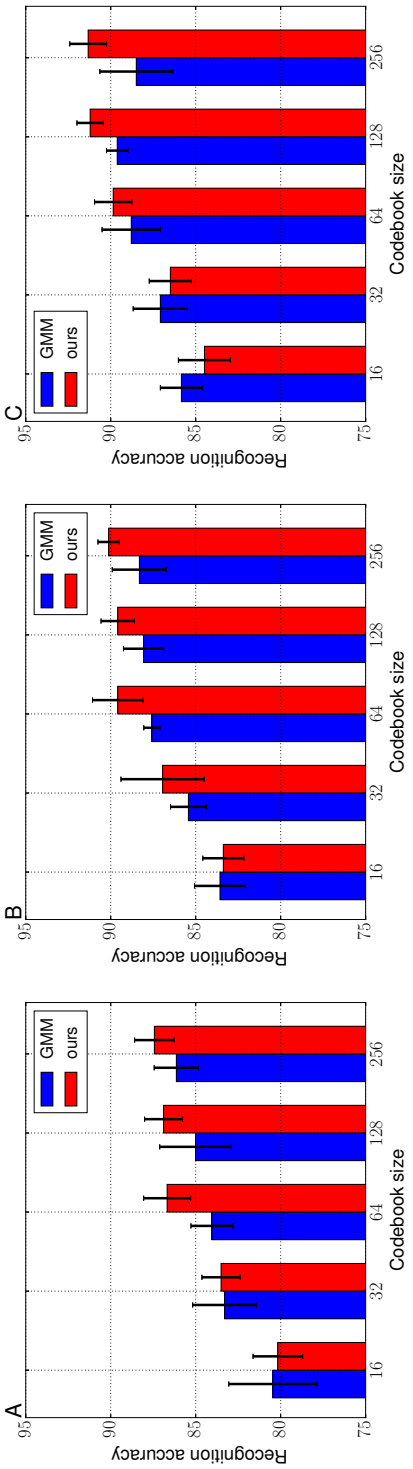


FIGURE 5.7: Recognition accuracies of the FV signatures with the GMM and the PPOC-soft codebooks on the PonceGroup-Butterfly.

TABLE 5.10: Recognition performance (mean accuracy ± standard deviation) of the VLADs with the k-means and ours (hard objective) codebooks on the Butterflies, corresponding to the Fig. 5.6.

	Codebook size K				
Method	16	32	64	128	256
20 images from each category corresponding to Fig. 5.6 (A)					
k-means	79.12 ± 1.10	82.71 ± 1.13	85.39 ± 0.84	85.59 ± 1.60	87.93 ± 1.51
ours	77.16 ± 1.21	81.84 ± 1.91	83.38 ± 1.36	86.64 ± 1.45	87.31 ± 1.28
30 images from each category corresponding to Fig. 5.6 (B)					
k-means	81.56 ± 1.02	85.67 ± 0.95	87.82 ± 0.73	88.07 ± 2.09	90.27 ± 1.59
ours	78.92 ± 1.76	83.23 ± 2.20	85.48 ± 0.83	89.24 ± 1.20	89.68 ± 0.87
40 images from each category, corresponding to Fig. 5.6 (C)					
k-means	83.19 ± 1.48	86.67 ± 2.11	88.85 ± 0.60	90.09 ± 1.30	90.32 ± 1.47
ours	81.83 ± 0.93	86.31 ± 0.71	86.73 ± 0.99	89.91 ± 1.35	91.27 ± 1.43

TABLE 5.11: Recognition performance (mean accuracy ± standard deviation) of the FVs with the GMM and ours (soft objective) codebooks on the Butterflies, corresponding to the Fig. 5.7.

	Codebook size K				
Method	16	32	64	128	256
20 images from each category corresponding to Fig. 5.7 (A)					
GMM	80.46 ± 2.59	83.30 ± 1.88	84.05 ± 1.23	85.01 ± 2.11	86.14 ± 1.30
ours	80.17 ± 1.46	83.51 ± 1.13	86.68 ± 1.38	86.89 ± 1.12	87.43 ± 1.16
30 images from each category corresponding to Fig. 5.7 (B)					
GMM	83.57 ± 1.50	85.43 ± 1.06	87.58 ± 0.47	88.07 ± 1.18	88.31 ± 1.60
ours	83.37 ± 1.21	86.94 ± 2.45	89.58 ± 1.49	89.58 ± 0.99	90.12 ± 0.63
40 images from each category, corresponding to Fig. 5.7 (C)					
GMM	85.84 ± 1.24	87.08 ± 1.60	88.79 ± 1.72	89.62 ± 0.63	88.50 ± 2.14
ours	84.48 ± 1.53	86.49 ± 1.24	89.85 ± 1.10	91.21 ± 0.78	91.33 ± 1.08

References

[1] *Clustering datasets*. Accessed: 2017-10-27. 2015.

[2] I. Kärkkäinen and P. Fränti. *Dynamic local search algorithm for the clustering problem*. Tech. rep. A-2002-6. Joensuu, Finland: Department of Computer Science, University of Joensuu, 2002.

[3] P. Fränti and O. Virmajoki. "Iterative shrinking method for clustering problems". In: *Pattern Recognition* 39.5 (2006), pp. 761–765.

[4] J. A. Nelder and R. Mead. "A Simplex Method for Function Minimization". In: *The Computer Journal* 7.4 (Jan. 1965), pp. 308–313.

[5] Thomas Harvey Rowan. *Functional Stability Analysis Of Numerical Algorithms*. Tech. rep. 1990.

[6] M. J. D. Powell. "A Direct Search Optimization Method That Models the Objective and Constraint Functions by Linear Interpolation". In: *Advances in Optimization and Numerical Analysis, Proceedings of the 6th Workshop on Optimization and Numerical Analysis, Oaxaca, Mexico*. Ed. by Susana Gomez and Jean-Pierre Hennart. Vol. 275. Dordrecht: Kluwer Academic Publishers, 1994, pp. 51–67.

[7] Alexander Zaslavski and M. Powell. "The NEWUOA software for unconstrained optimization without derivatives". In: *Large-Scale Nonlinear Optimization*. Ed. by Panos Pardalos, G. Pillo, and M. Roma. Vol. 83. Nonconvex Optimization and Its Applications. Boston: Springer US, 2006. Chap. 16, pp. 255–297.

[8] A. R. Conn, N. I. M. Gould, and Ph Toint. "A Globally Convergent Augmented Lagrangian Algorithm for Optimization With General Constraints and Simple Bounds". In: *SIAM Journal on Numerical Analysis* 28.2 (Apr. 1991), pp. 545–572.

[9] E. G. Birgin and J. M. Martínez. "Improving Ultimate Convergence of an Augmented Lagrangian Method". In: *Optimization Methods Software* 23.2 (Apr. 2008), pp. 177–195.

[10] Steven G. Johnson. *The NLopt nonlinear-optimization package*. Accessed: 2017-9-10.

[11] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. "A Maximum Entropy Framework for Part-Based Texture and Object Recognition". In: *10th International Conference on Computer Vision (ICCV '05)*. Vol. 1. Beijing, China: IEEE Computer Society, Oct. 2005, pp. 832 –838.

[12] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. "Semi-local Affine Parts for Object Recognition". In: *British Machine Vision Conference (BMVC '04)*. Ed. by Andreas Hoppe, Sarah Barman, and Tim Ellis. Kingston, United Kingdom: The British Machine Vision Association (BMVA), Sept. 2004, pp. 779–788.

[13] Herbert Bay et al. "Speeded-Up Robust Features (SURF)". In: *Comput. Vis. Image Underst.* 110.3 (June 2008), pp. 346–359.

[14] Eric Nowak, Frédéric Jurie, and Bill Triggs. "Sampling strategies for bag-of-features image classification". In: *European Conference on Computer Vision*. Springer, 2006.

[15] Yan Ke and Rahul Sukthankar. "PCA-SIFT: A more distinctive representation for local image descriptors". In: 2004, pp. 506–513.

[16] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

Chapter 6

Conclusions

In this dissertation, I have described the three codebook-based approaches and evaluations.

In chapter 2, some recent codebook-based approaches have been overviewed.

Chapter 3 described the fuzzy codebook and it has been applied to the image recognition problems.

This paper has presented an approach of reducing computational complexity in feature encoding based on the fuzzy codebook, and presented its performance with the online classifier for image recognition. In experimental results, our approach has a potential on the same level as recent approaches on the Caltech-101 dataset.

Chapter 4 has reported the relationship between the prior probabilities and recognition performance in codebook-based image representation, there is a possible to improve the recognition accuracy and the sparseness of image signatures by controling the standard deviation of the prior probabilities of codebook.

Chapter 5 focussed on clustering from the perspective of the variance prior probabilities and presented the clustering frameworks, namely hard and soft objectives, that are respectively alternative to basic approaches such as the k-means and the GMM. In the numerical analysis, four optimization frameworks were evaluated with synthetic clustering datasets. The results of all of the frameworks were better than the basic clusterings. Especially, it

showed that the Subplex optimizer is able to give better objective values from the perspective of the variance of prior probabilities and to construct intuitively appropriate clusters for complicatedly distributed clustering samples. In the experiment with image datasets, the hard objective was probably not effective for the VLAD encoding because the objective values became worse compared with the k-means results as the number of clusters increase. On the other hand, the FV encoding with the soft objective showed improvements in recognition performance regardless of some parameters such as the codebook size and the ratio of training and test images.

Publications

Journal

1. Yuki Shinomiya and Yukinobu Hoshino. "A Quantitative Quality Measurement for Codebook in Feature Encoding Strategies". In: *Journal of Advanced Computational Intelligence and Intelligent Informatics*. 21.7, 2017, pp. 1232–1239.

2. Yuto Yasuoka, Yuki Shinomiya, and Yukinobu Hoshino. "Simulation of Human Detection System using BRIEF and Neural Network". In: *Journal of Advanced Computational Intelligence and Intelligent Informatics*. 20.7, 2016, pp. 1159–1164.

3. Yuki Shinomiya and Yukinobu Hoshino. "A Feature Encoding based on Low Space Complexity Codebook called Fuzzy Codebook for Image Recognition". In: *International Journal of Fuzzy Systems*. (to appear)

4. Yuki Shinomiya and Yukinobu Hoshino. "A proposal of prior probability-oriented clustering in feature encoding strategies". *PLosONE*. (in revision)

International Conference

1. Yuki Shinomiya and Yukinobu Hoshino. "An Analysis of Dependency of Prior Probability for Codebook-Based Image Representation". In: *Proceedings of the Joint 8th International Conference on Soft Computing and*

Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS), 2016, pp. 103–108.

2. Yuto Yasuoka, Yuki Shinomiya, and Yukinobu Hoshino. "Evaluation of Optimization methods for Neural Network". In: *Proceedings of the Joint 8th International Conference on Soft Computing and Intelligent Systems (SCIS) and 17th International Symposium on Advanced Intelligent Systems (ISIS)*, 2016, pp. 92–96.

3. Yuki Shinomiya and Yukinobu Hoshino. "A Validation of Feature Encoding based on Fuzzy Codebook with Online Classifier". In: *Proceedings of the 16th International Symposium on Advanced Intelligent Systems (ISIS)*, 2015.

4. Yuto Yasuoka, Yuki Shinomiya, and Yukinobu Hoshino. "Development of human detection system by BRIEF". In: *Proceedings of the 16th International Symposium on Advanced Intelligent Systems (ISIS)*, 2015.

5. Yuki Shinomiya and Yukinobu Hoshino. "A Feature Encoding based on Fuzzy Codebook for Large-Scale Image Recognition". In: *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, 2015, pp. 2908–2913.

Domestic Conference

1. 四宮 友貴, 星野 孝総. "画像認識におけるFisher Vectorのパラメータ化の検証". 第31回ファジィシステムシンポジウム, 2015.

2. 安岡 優斗 四宮 友貴, 星野 孝総. "Neural Networkを用いた学習・識別性能テスト". 第31回ファジィシステムシンポジウム, 2015.

3. 安岡 優斗, 四宮 友貴, 星野 孝総. “局所特徴量を用いた人検出システムのFPGA化に向けたシミュレーション”. SOFT九州支部 中国・四国支部合同支部大会, 2015.

Award

1. 2016年 IEEE Computational Intelligence Society Japan Chapter Young Researcher Award