Visualizing High-dimensional Image Feature Space using Graph Drawing Techniques

by

Yi Gao

高 毅

A Master Thesis

修士論文

Submitted to
the Graduate School of Frontier Sciences
The University of Tokyo
February 3, 2015

Thesis Supervisor: Shigeo Takahash

指導教員：高橋 成雄　准教授

Abstract

The bag-of-features model is one of the most popular and promising approaches for extracting the underlying semantics from image databases. However, the associated image categorization based on machine learning techniques may not convince us of its validity since we cannot visually verify how images have been classified in the high-dimensional image feature space. This thesis aims at visually rearranging the images in the projected feature space by taking advantage of a set of representative features called visual words obtained using the bag-of-features model. The main idea is to associate each image with a specific number of visual words to compose a bipartite graph, and then lay out the overall images using anchored map representation in which the ordering of anchor nodes is optimized through a genetic algorithm. For handling relatively large image datasets, A pair of similar images is merged one by one to conduct the hierarchical clustering through the similarity measure based on the weighted Jaccard coefficient. Voronoi partitioning has been also incorporated into the present approach so that we can visually identify the image categorization based on support vector machine. Experimental results are finally presented to demonstrate that the present visualization framework can effectively elucidate the underlying relationships between images and visual words through the anchored map representation.

Acknowledgements

I am grateful to the Associate Professor Shigeo Takahashi for his suggestion and guidance. I also would like to thank Associate Professor Kazuo Misue for his useful advices and discussions. Moreover, I would like to thank all the members of Takahashi Laboratory, as they helped me a lot not just on research, but in life as general. Special thanks to Hsiang-Yun Wu and Kazuyo Mizuno who gave me some good insights and guidance during the research. This research has been partially supported by the MEXT KAKENHI under Grants-in-Aid for Scientific Research on In-novative Areas No. 25120014.

Contents

Chapter 1

Introduction

Sophisticating tools for image categorization becomes more crucial due to the rapid development of Internet and the increasing size of image databases. Searching relevant images based on user preferences is not a trivial task if images are annotated manually, however, the manual annotating process is time-consuming and subjective. It's desirable to have content-based image categorization. While the associated techniques have been improved until recently, it is still a hard work to sufficiently infer the underlying semantics from images. This problem primarily arises from the fact that we cannot precisely identify specific objects embedded in the images regardless of possible variations in their view, lighting, and occlusion conditions. The *bag-of-features* (BoF) model [22, 3] successfully alleviates the above problem for effective image retrieval. A main idea behind the BoF model is to seek an analogy of methods for inferring text categorization based on the bag-of-words model, where each document is represented as a sparse vector of representative words by referring to their occurrence without worrying about their associated orders. In practice, the BoF model allows us to associate an individual image with a small weighted set of *visual words*, each of which stands for a group of local features in the high-dimensional feature space and thus corresponds to some specific image content in the image.

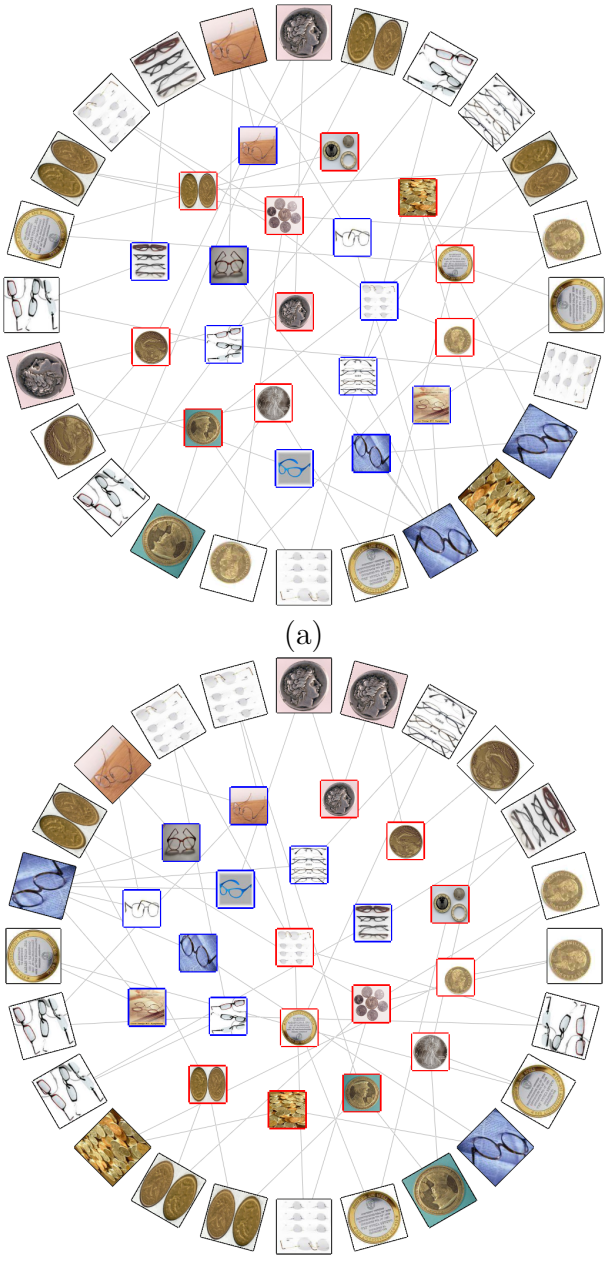
Since images are represented as histograms by using BoF model, visual

image categorization problem can be simplified as multi-class supervised learning problem, which can be solved by employing machine learning techniques. Machine learning techniques include two separate steps in order to category unlabeled input images: training and testing. In the training step, classifiers are trained using training images that are labeled by hand, while in the testing step, images are annotated with their corresponding categories.

Nonetheless, the correctness of the image categorization is not always convincing even with the help of classification methods based on machine learning algorithms, since the actual mechanism for the associated image categorization has not been fully visualized due to the high-dimensionality of the image feature space. In this thesis, I solve this problem by encoding the relationship between images and visual words as a bipartite graph first, and then employing *anchored map* representation [16] to rearrange the image set on the 2D screen space, as shown in Figure 1.1(a). Genetic algorithms have also been employed to optimize the circular ordering of visual words around the image feature space, so that I can visually elucidate the underlying relationship between images and visual words (as shown in Figure 1.1(b)). Furthermore, I employ a hierarchical visualization framework by merging and splitting images through referring to values of the weighted Jaccard similarity between sets of visual words associated to their corresponding images, so that I can achieve a better understanding through global and local exploration. In the present prototype system, an image will be temporarily merged when its Jaccard similarity coefficient is larger than a particular threshold, which can be defined by users. The background of images is drawn by assigning different colors to different categories while I partition the screen space into several Voronoi cells by referring to the center coordinate of each image.

The remainder of this thesis is organized as follows: Chapter 2 provides a review on conventional techniques for bag-of-features and high-dimensional visualization. Chapter 3 describes how to extract image features and con-

struct the dictionary of visual words by extracting low-level image features. Chapter 4 presents my approach to transforming the high-dimensional image feature space to an anchored map representation by referring to the bipartite relationships between images and visual words. After having presenting several experimental results to demonstrate the feasibility of my prototype system in Chapter 5, Chapter 6 concludes this thesis and refer to future work.



(a)

(b)

Figure 1.1: Using anchored maps to visualize bag-of-features image categorization. (a) Original layout. (b) Enhanced layout with an optimized circular ordering of visual words annotated with representative images. Images in same category are brought closer to each other. (#{visual words} = 24.)

Chapter 2

Related Work

In this Chapter, I describe related work on image categorization, and also provide a survey on existing high-dimensional visualization techniques.

2.1 Image Categorization Based on Machine Learning Techniques

Content-based image retrieval has been a hot topic in the fields like image processing field, computer graphics, and multimedia. For effective search for specific contents, it is important to classify images into several categories by inferring semantics of visual features embedded in them. The *bag-of-features* (BoF) model is a well-known approach for such image representation and helps us categorize images by computing the number of occurrence of particular visual features contained in each image [22, 3]. This idea originates from the concept of *bag-of-words* that naturally allows us to classify documents by counting the number of particular words defined in the dictionary [11]. The bag-of-words model represents documents or sentences as bags of words, where the word order and even grammar are disregarded, so that documents or sentences can be represented as vectors of words, where each entry of these vectors refers to the frequence of corresponding word. After representing documents as vectors, these vectors are used for training classifiers and

document classification problem is solved. Indeed, this concept has been extended to the image databases where an image is represented as a vector of features, here features called *visual words* are employed as words and a set of features is employed as the dictionary in the bag-of-words model. Thus, image categorization is usually considered as a two-step approach, including representing images as vectors of extracted features and recognizing images based on simulating human behavior on classifying types of images.

About the feature extraction and image representation, in the early stage of approaches of feature extraction, several studies focused on detecting global image features for encoding the image as a whole. Nonetheless, these features appeared to be incorporated for the purpose of categorizing images because they are too sensitive to image transformations such as translation, scaling, and rotation together with lighting conditions and occlusions. Lowe presented a feature detection technique called *scale-invariant feature transform* (SIFT) [14], which extracts local image features in a way that they are robust enough to the prescribed conditions. In the BoF model, the visual words were obtained by collecting the SIFT features from a set of training images and employing the conventional k-means clustering [22, 3] to identify the corresponding cluster centers as the visual words, and then represent images as vectors of these visual words. In recent year, sparse coding [25] instead of k-means clustering is used for clustering features in order to get better performance in image representations.

As for practical approaches for image categorization, After encoding each image in the database as a weighted sum of the relevant visual words. Indeed, BoF models facilitates us to assign a sparse vector representation of visual words to each image by quantizing it in terms of its associated visual words [22, 3] or L_1-norm regularization [25]. *Support vector machine* (SVM) has been often employed as a standard classifier since it produces high accuracy in image categorization [3, 5]. As an extension, Bosch et al. [1] revisited

the recognition scheme and apply it to the video by employing probabilistic latent semantic analysis (pLSA) followed by k-nearest neighbor (k-NN) classification.

Over the years, a wide range of methods has been developed to improve the quality of the image categorization. A state-of-the-art technique is spatial pyramid matching (SPM) proposed by Lazebnik et al. [13], where they incorporated spatial gradient information of images at multiple scales into the BoF model. More studies also focused on improving the discriminative power of the visual words dictionary. For example, Winn et al. [24] introduced a statistical measure for the optimization framework to make the dictionary of the visual words more compact, while Perronnin [20] combined local and global feature detection frameworks to exhibit higher performance. However, the space of image features extracted by these approaches is always high-dimensional and too abstract to understand the meaningful structures hidden behind that space.

2.2 Visualizing High-dimensional Image Feature Space

Visualizing high-dimensional feature space often successfully elucidates the image classification obtained through machine learning techniques. A dimensionality reduction technique called *multidimensional scaling* (MDS) [23, 12] is one of the common techniques to project the high-dimensional space onto a 2D screen space for better readability. Kyle Heath et al. [8] built graph structures called Image Webs where collections can be highly interconnected through implicit links between image pairs viewing the same or similar objects. Recently, Paulovich et al. [19] and Mamani et al. [15] developed dimensionality reduction frameworks that allows us to interactively edit the underlying structures of the high-dimensional space through screen-space

manipulations. Furthermore, Mizuno et al. [18] presented a framework for interactively exploring feature space that is specific to the BoF models, by referring to the relationships between images and visual words. In my approach, I also focus on such relationships specific to the BoF models and encode them as anchored map representations [16, 21] for visualization purposes. Technical details of the present approach will be detailed later in Chapter 4.

Chapter 3

Bag-of-features Model for Image Categorization

This chapter first provides a brief overview of the BoF model for encoding images as feature vectors, then describes how images are categorized using machine learning techniques.

3.1 Image Representation Based on the Bag-of-Features Model

The bag-of-features model (BoF) is a well-known approach for image representation and helps us categorize images by computing the number of occurrence of particular visual features contained in each image [19, 3]. This idea originates from the concept of bag-of-words that naturally allows us to classify documents by counting the number of particular words defined in the dictionary [8]. Indeed, this concept has been extended to the image databases where a set of local features called visual words is employed as the dictionary for the analysis of image contents. In general, the BoF model consists of three steps: *feature extraction*, *visual words dictionary formation* and *image-histogram representation*.

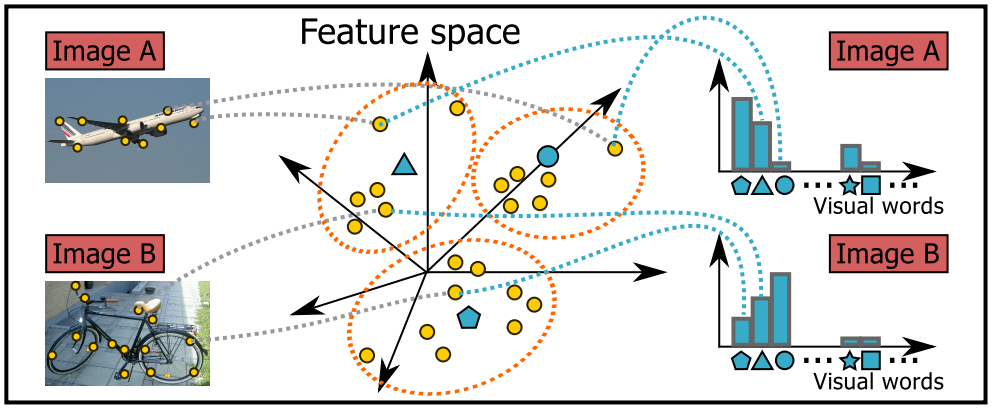


Figure 3.1: The BoF model. (a) SIFT feature vectors extracted from images are plotted in the 128 dimensional feature space. (b) The k-means clustering algorithm is employed to identify a visual word as the center of each cluster. (c) Each image is encoded as normalized histogram coordinates in terms of the visual words.

3.1.1 Feature Extraction

The first step of the BoF construction is the *feature extraction*, where we extract SIFT features from the respective images. A feature here is a piece of information that is relevant for solving the computational task related to image categorization. Features may be image color information, corners, edges, shape and texture features, nonetheless, these features appeared to be inappropriate for image categorization because they are not robust to translation, scaling and rotation. Lighting conditions and occlusions and affect the result of image categorization. Lowe presented a feature detection technique called scale-invariant feature transform (SIFT) [11], which extracts image features in a way that they are robust enough to the prescribed conditions. The SIFT features are described as 128-dimensional feature vectors and after extracting SIFT features from images we then plot these features within the 128-dimensional feature space as shown in Figure 3.1(a).

3.1.2 Visual Words Dictionary Formation

For conducting the second step as the *visual words dictionary formation*, all the SIFT features are grouped into a specific number of clusters. The simplest technique for this purpose is the conventional k-means clustering algorithm, where the number of clusters k is predefined. Now we are ready to identify the center of each cluster as a representative feature called a *visual word*, and compose the list of k visual words as the dictionary as exhibited in Figure 3.1(b).

3.1.3 Image Histogram Representation

The last step is *image histogram representation*, where we encode each image as a histogram coordinates in terms of the visual words. This is accomplished by quantizing each SIFT feature vector contained in the image to its closest visual word in the 128-dimensional feature space first, and then counting the occurrence of each visual word to construct the histogram. Finally, each image is represented as a sparse vector of visual words by normalizing the bins of the histogram to compose the normalized histogram coordinates, as shown in Figure 3.1(c).

3.2 Image Categorization using Support Vector Machine

After representing images as feature vectors by employing BoF model, the support vector machine (SVM) is employed as the simplest learning models for classifying images by partitioning the high-dimensional space spanned by the extracted visual words [3]. In practice, the classifier finds the maximum marginal hypersurfaces that separates positive and negative samples in the training dataset, and further classifies each of the unknown samples by referring to the separating hypersurfaces. In this thesis, I introduce the

SVM-based image categorization process proposed by Csurka et al. [3] and visualize how the bounding hypersurfaces enclose the images of specific type according to the input training samples provided by users. In my approach, I employed radial basis functions (RBFs) kernels for representing such separating hyperplanes to better classify the complicated configuration of images in the high-dimensional space, and visualize the associated image classification in the screen space for more convincing representation.

Chapter 4

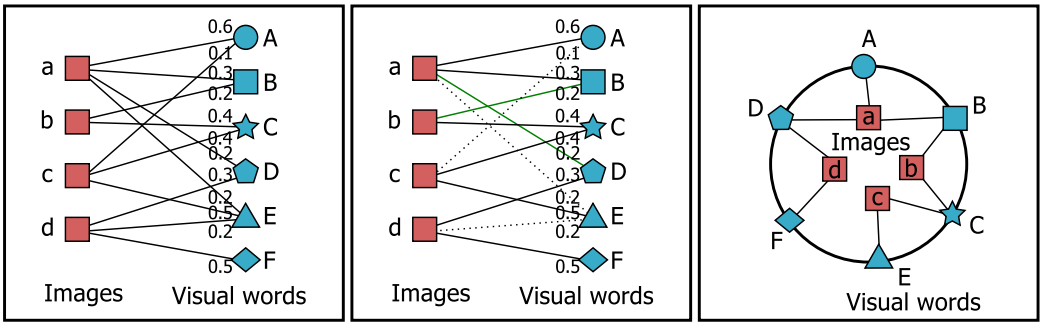
Hierarchical Bipartite Graph Visualization

In this chapter, I describe how to visualize image categorization via an anchored map representation by referring to the bipartite relationships between images and visual words. I also introduce the weighted Jaccard similarity index for adaptively clustering images so that we can hierarchically represent large scale image sets within the framework of anchored maps. Moreover, a voronoi partitioning has been incorporated into my approach to help us visually identify the image categorization based on support vector machine.

4.1 Bipartite Network Composition

The most common way of visualizing the high-dimensional image feature space is to employ dimensionality reduction techniques. Nonetheless, it is often the case that we still cannot fully discriminate each image category from others if the images are simply projected onto the low-dimensional space. My original idea for alleviating this problem is to extract bipartite relationships between images and visual words from the BoF model first, and then transform them into a network structure so that I can takes advantage of existing graph drawing techniques for better visualization.

For this purpose, I first establish edge connections between each image



(a) (b) (c)

Figure 4.1: Bipartite relationships between images and visual words in the BoF model. (a) An original bipartite graph. (b) A sparse bipartite graph after edge pruning. (c) The corresponding anchored map representation.

and its relevant visual words if they correspond to non-zero histogram coordinates of that image. Note that here I represents images and visual words as nodes of the bipartite graph, while I associate each normalized histogram coordinate value with the corresponding edge as its weight value as shown in Figure 4.1(a). Furthermore, I would like to make the bipartite graph as sparse as possible for better readability of the resulting graph visualization. Thus, I sort the edges in an ascending order according to the weight values, and prune the edge having the minimum weight one by one until I cannot remove edges any more without decomposing the graph into multiple connected components, as shown in Figure 4.1(b). In this way, I construct a sparse representation of the bipartite graph over the image and visual word nodes [26].

4.2 Anchored Map Representation

As for the visualization of the bipartite relationships, I investigate more graph drawing techniques, since the graph is a data structure that is used for representing mutual relationship between attributes. Large numbers of techniques

are studied for drawing general graph while here I focus on drawing bipartite graph because the mutual relationship between images and their represented visual words is exactly a bipartite structure. Bipartite graph is a specific subset in graphs, where the attributes can be exactly distributed into two independent subsets. Misue presented an approach called anchored maps, which fixes one set of attributes in a circular ordering and allow remaining attributes to move by employing conventional force-directed placement [16, 17]. An energy-based spherical embedding method is also introduced to adjust positioning vector of each attribute, while the method limits the placement of attributes on two concentric circles [6]. In this thesis, I choose anchored maps for visualizing the relationship between images and visual words because the technique provides more feasibility to the placement of images.

In the anchored map representation, nodes in one of the two disjoint sets of the bipartite graph are spaced along the boundary of a disk region, while nodes of the other set are free to move within the disk, as shown in Figure 4.1(c). The restricted nodes are fixed like anchors, hence the terms "anchors"and "free nodes".
The drawing procedure of anchored maps includes two steps:

- Space anchors equally along the boundary.
- Arrange free nodes at the appropriately positions by using spring embedder algorithm [4]. Edges connected the two disjoint nodes are drawn as straight line segments.

In my system, I release the image nodes within the central disk region as free nodes of the anchored map and fixed the visual word nodes along its circular boundary as anchors. The conventional spring embedder algorithm is also applied to the free nodes to avoid unnecessary overlap among images in the central region, where I also incorporates edge weights into my formulation so that each image will be brought closer to its relevant visual words according

to their corresponding normalized histogram coordinates. In spring embedder algorithm, the initial positions of the free nodes has influence on the layout of the anchored map representations, in order to get better layout, the positions of free nodes are initialized to the weighted average of the adjacent visual words.

In the anchored map representation, the length of the circular boundary only influence the size of the drawing, doesn't influence the readability of the layout; While the order of the anchors plays an important role in the readability of the layout. By optimizing the order of anchors based on some rules, I can improve the readability of anchored maps. And in my sparse representation of the bipartite graph, each image usually depends on a small number of visual words. This means that my scheme is more likely to bring image of the same category close to each other in the anchored map representation since they usually share almost the same set of visual words in their histogram representation. Furthermore, this visual readability of the image categorization can be enhanced if I carefully reorder the visual word nodes along the circular boundary of the disk to make each image node have its neighbor visual word nodes within its vicinity. This is accomplished by devising genetic-based algorithms for optimizing the circular ordering of visual words, where I define a chromosome as a value-encoding sequence of visual word IDs as showed in Figure 4.2. For fully discriminating between image categories, I optimize the chromosome sequence by defining the cost function. So that, for each image node, every pair of its adjacent visual word nodes become closer to each other. This amounts to calculating the circular distance between adjacent visual word nodes for each image node, and summing up the squared distances except for the largest one [16]. The cost function can be expressed by using the following formula:

$$\mathrm{C} = \sum_{i=1}^{k-1} D_I^q, \tag{4.1}$$

where k represents the number of visual words connected with image I, and parameter q is the power of circular distance D. Parameter q is set to be 2 in the experiments of this thesis. As in the example showed in Figure 4.2, the image a is connected with three visual words: A, B and D. The circular distance between each pair of adjacent visual word is 1, 1 and 4. And calculate the cost function by summing up the squared distances except for the largest one:

$$\mathrm{C} = 1^2 + 1^2 = 2$$

I optimize the chromosome sequence by finding local minimum solution of the cost function. This genetic-based optimization provides us with better anchored maps in the sense that images in the same category will be closer to each other in the central disk region as shown in Figure 1.1(b).

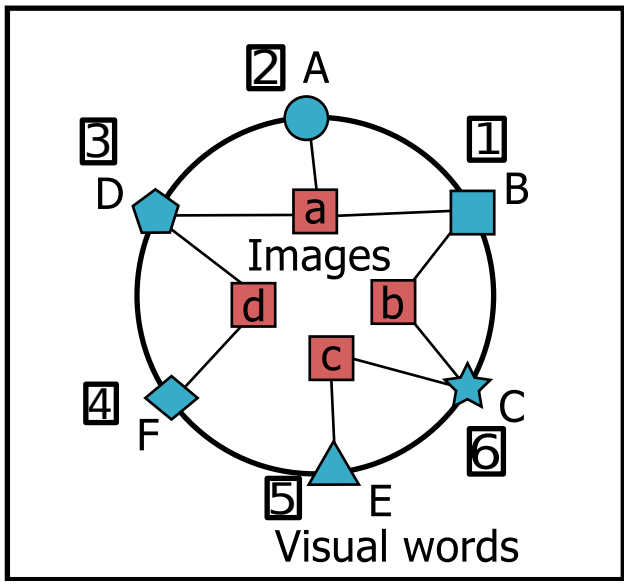
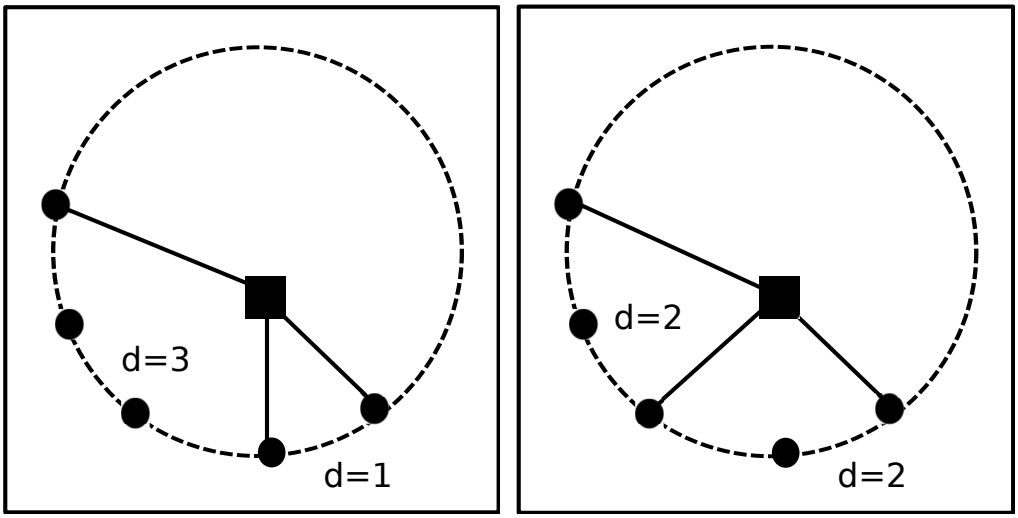


Figure 4.2: A genetic-based optimization is employed to improve the readability of the anchored map representation. The chromosome is defined as a value-encoding sequence of visual word IDs.

The reason why I employed parameter q to the cost function is illustrated

in the Figure 4.3. When q is set to be 1, the cost function $C = 1^1 + 3^1 = 4$ in the case of Figure 4.3(a), and $C = 2^1 + 2^1 = 4$ in the case of case Figure 4.3(b), since the same value of cost function in either case, I cannot decide which layout is better. In order to solve this problem as well as take account of the symmetry and balance q is set to be $q > 1$ [16]. In all the experiments of this thesis, parameter q is 2, when $q = 2$, cost function $C = 1^2 + 3^2 = 10$ in the case of Figure 4.3(a), and $C = 2^2 + 2^2 = 8$ in the case of Figure 4.3(b). Thus, the layout in Figure 4.3(a) is better than the layout in Figure 4.3(a).



(a) (b)

Figure 4.3: Parameter q in the cost function.

4.3 Hierarchical Clustering of Images

As the number of input images increases, the central disk region of the anchored map will be more crowded with the images. For improving the scalability of the anchored map representation, I also introduced hierarchical representation of the anchored map by adaptively clustering images accord-

ing to their image similarities. More specifically, I compose a dendrogram tree structure of images by merging a pair of the most similar images one by one iteratively [21]. For evaluating the similarity among images, we employ the conventional Jaccard similarity index, which is the most popular similarity measure between a pair of sets [2].

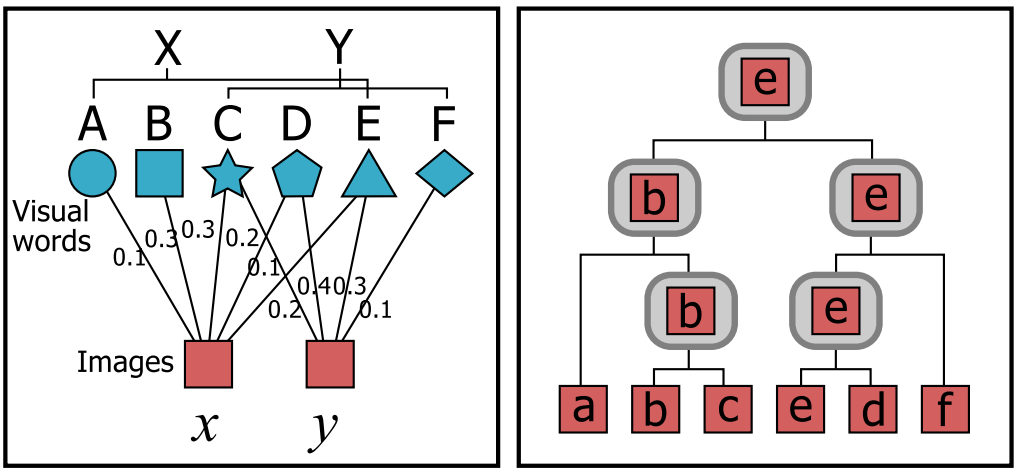
Let us consider two sets X and Y for example. The conventional Jaccard index is defined as $J(X,Y) = |X \cap Y|/|X \cup Y|$, where $|Z|$ represents the number of elements contained in the set Z. However, in my case, the weighted Jaccard similarity index [10, 2] is more appropriate in the sense that we can incorporate the importance of each relevant visual word when evaluating the image similarities, rather than simply counting the number of relevant visual words in the union and intersection of the two sets.

As described earlier, my bipartite graph is composed by connecting an image with its relevant visual words, and the weight of each edge is equivalent to the normalized histogram coordinate value of the corresponding visual word with respect to that image. Thus I can easily compute the weighted Jaccard similarity index between a pair of images by referring to their corresponding sets of visual words X and Y, together with their corresponding edge weights, as follows:

$$\mathrm{WJ}(X,Y) = \frac{\sum_{i=1}^{n} \min(X_i, Y_i)}{\sum_{i=1}^{n} \max(X_i, Y_i)}, \tag{4.2}$$

where n denotes the total number of visual words contained in the union of X and Y. Note that the numerator is obtained by summing up the minimum values between two weights of the edges emanating from visual words in X and Y, while the denominator is the sum of the maximum values.

Figure 4.4(a) shows an example, where the X_i and Y_i are defined as normalized histogram coordinates for the image nodes x and y, and thus we can set $(X_i) = (0.1, 0.3, 0.3, 0.2, 0.1, 0.0)$ and $(Y_i) = (0.0, 0.0, 0.2, 0.4, 0.3, 0.1)$. This means that we can compute the weighted Jaccard similarity index be-



(a) (b)

Figure 4.4: Hierarchical structure of bipartite graph visualization. (a) An example bipartite graph between images and visual words. (b) Dendrogram-based representation of clustered images.

tween the image nodes x and y as

$$\mathrm{WJ}(X,Y) = \frac{0.0+0.0+0.2+0.2+0.1+0.0}{0.1+0.3+0.3+0.4+0.3+0.1} = \frac{1}{3}.$$

Using the weighted Jaccard measure, I can iteratively merge a pair of the most similar images into a group one by one, and encode the clustering process as a dendrogram tree representation as shown in Figure 4.4(b). As illustrated in this figure, I incorporate an image node having a smaller number of child nodes into the other image node representing more child nodes in my implementation.

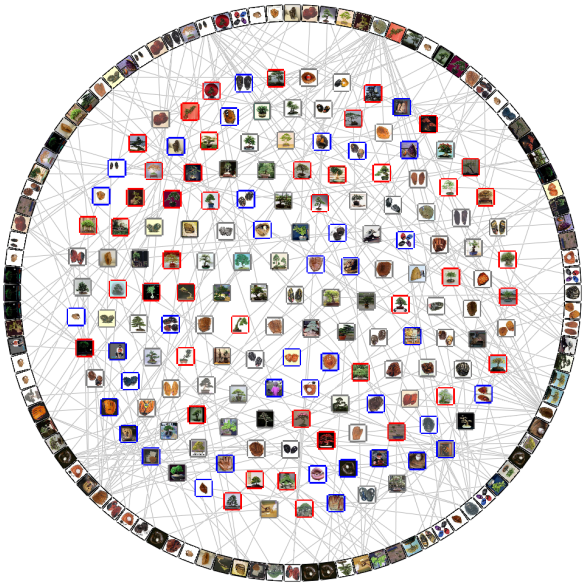
In the example of Figure 4.5, the image set contains images of two different objects, i.e., bonsais and baseball gloves. Bonsai images are outlined by red and baseball glove images are outlined by blue. The central disk region of the anchored map is crowded with images in the original layout as showed in Figure 4.5(a). After clustering images according to their similarities, the readability of the anchored map representation will be improved as showed in Figure 4.5(b). My system also allows users to adjust the number of clusters

by preference, experiment results will be shown in Chapter 5.

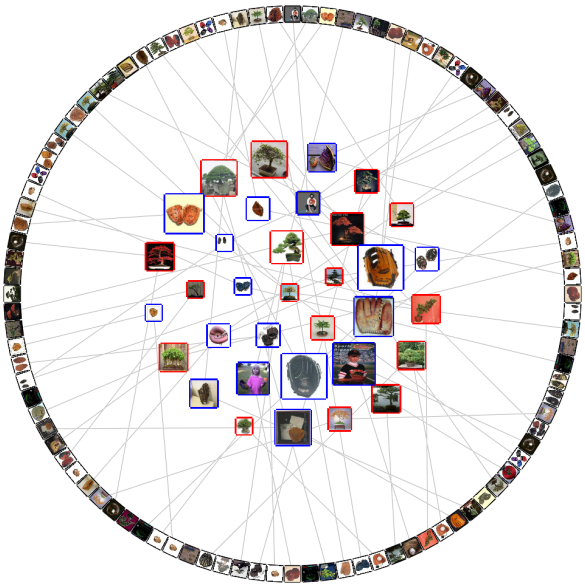
4.4 Visualizing SVM-based Image Classification

I also equip my prototype system with an interface for classifying images using support vector marching (SVM). In practice, users are allowed to interactively specify a subset of images as a training set for SVM-based classifier together with the tags that represent whether the corresponding images are classified into a specific category or not. Nonetheless, conventional BoF models just present the classification results only and do not provide us with any information about how the images are classified in the high-dimensional image feature space. I projected the high-dimensional image categorization onto the central disk region within the anchored map, and introduced the Voronoi tessellation technique in order to clarify how the region is partitioned according to the image categorization. Here, I employ the position of each image node as a seed point for the Voronoi cell, and assign a specific color to that cell according to its image category obtained through the SVM classification. This successfully makes us convinced with the image categorization provided by the SVM-based classifier by visualizing the associated image categorization within the anchored map representation. As shown in Figure 4.6, coin image category is assigned to be yellow so that we can visualize how the coin images are categorized in this representation.

Note that, in my implementation, I incorporated a hardware-assisted algorithm for computing Voronoi diagrams [9] and restrict the drawing area to the central disk region of the anchored map using the stencil buffer.

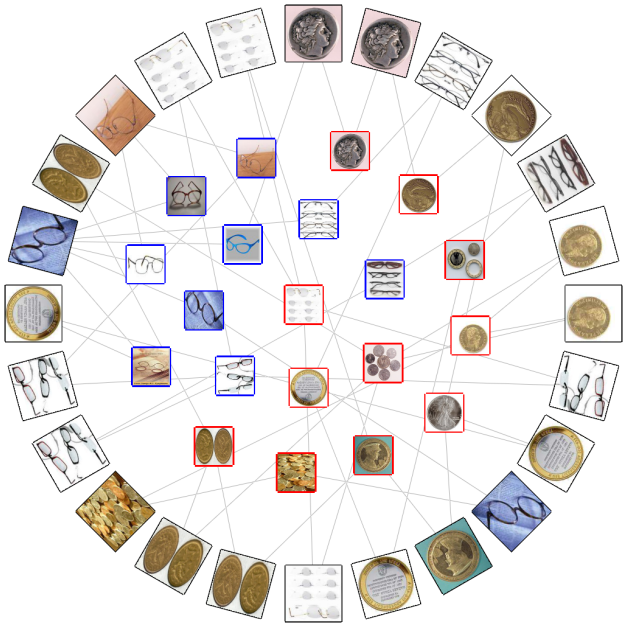


(a)

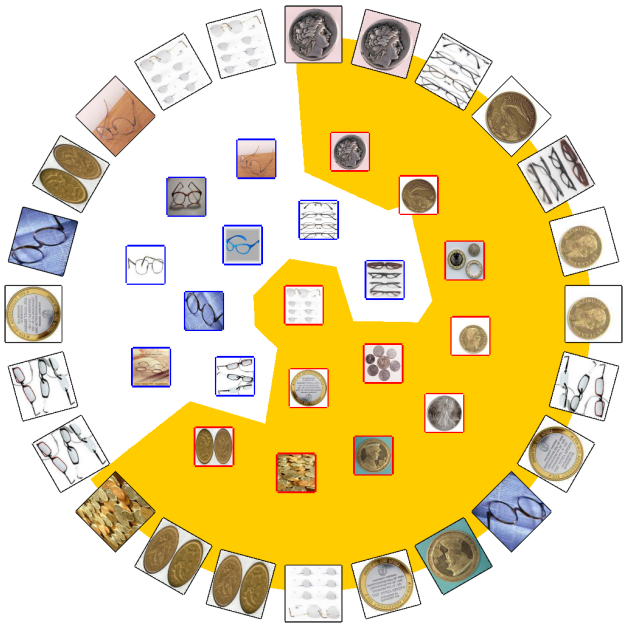


(b)

Figure 4.5: Hierarchical Clustering of Images. (a) Original layout. (b) Enhanced layout by adaptively clustering images according to their similarities. Each image stands for a cluster of images and its size indicates how many images are contained in that cluster.



(a)



(b)

Figure 4.6: Visualizing SVM-based image classification by employing voronoi tessellation technique. (a) Enhanced layout with an optimized circular ordering of visual words annotated with representative images. (b) Region is partitioned according to the image categorization using Voronoi tessellation technique. Coin image category is assigned to be yellow. (#{visual words} = 24.)

Chapter 5

Experimental Results

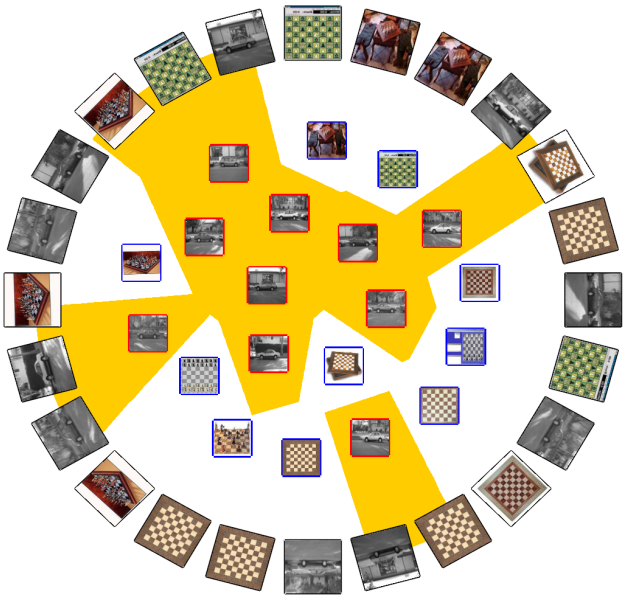
The present system has been implemented on a laptop PC with an Intel Core i7 CPU (2GHz, 4MB cache) and 8GB RAM, and the source code has been written in C++ using the OpenGL library for drawing graph layouts, OpenCV library for SIFT feature extraction and SVM learning models, and GAlib library for the implementation of the genetic-based algorithm. The images datasets used in this thesis were collected from Caltech256 [7].

Figure 5.1 exemplifies how the underlying image categorization can be better visualized by taking advantage of the optimal ordering of visual words around the circular boundary of the anchored map representation. The image set contains images of two different objects: cars and chessboards from which we try to discriminate one against another, while the yellow polygons represent the region dominated by car images. Here, Figure 5.1(a) shows the initial ordering of visual words and layout of images in the dataset where images of cars and chessboards are intricately mixed. On the other hand, images of two categories are sufficiently discriminated in Figure 5.1(b) when we rearrange the ordering of the visual words using genetic-based optimization.

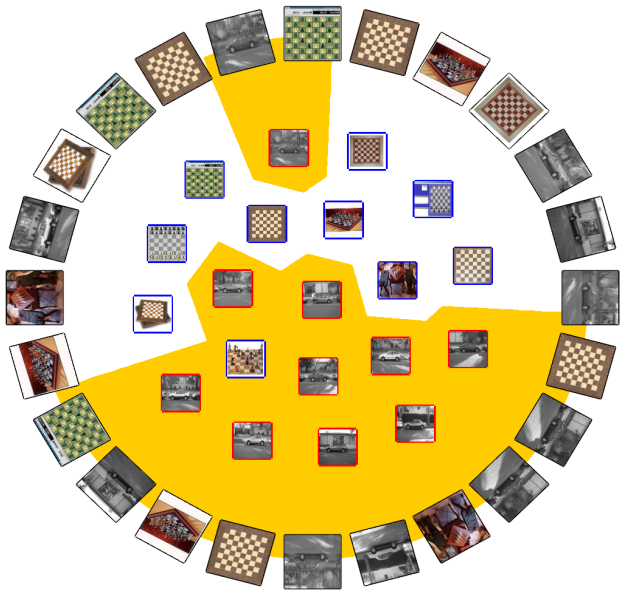
The image set exhibited in Figure 5.2 contains images of three different objects, i.e., cars, tomatoes and grapes from which we try to discriminate car images specifically from the others. For effectively handling a large number of images, I first compute a small number of image clusters through hier-

archical grouping of images, and distinguish car images from the others as our target using the SVM-based image categorization. Note that here the images outlined in red are labeled as example images within the specific category (i.e. car images), while those in blue are images that are out of our target. I then gradually decompose each image cluster into smaller clusters, and adjust the image categorization by interactively labeling a small number of images as the training set according to their categories. This successfully allows us to enclose car images within yellow background region from the coarsest level to the finest (i.e. original) level as shown in Figure 5.2.

Figure 5.3 demonstrates how we can categorize images of a specific category even when we train our image classifier indirectly with similar looking images. In this case, I represent each image in terms of visual words obtained from training images containing tomatoes, coins, and cars and try to collect images of round shapes. However, I also takes as input images of additional categories such as CDs and glasses in this example, while we still can categorize images of round objects into our target category using the SVM-based classifier, and clearly visualize the associated image categorization both at coarse and fine levels through the anchored map representation as shown in the Figure 5.3.



(a)



(b)

Figure 5.1: Discriminating images from tow different categories (cars and chessboards). Car images outlined in red and chessboard images outlined in blue. (a) Original layout. (b) Enhanced layout with an optimized circular ordering of visual words annotated with representative images. (#{visual words} = 24.)

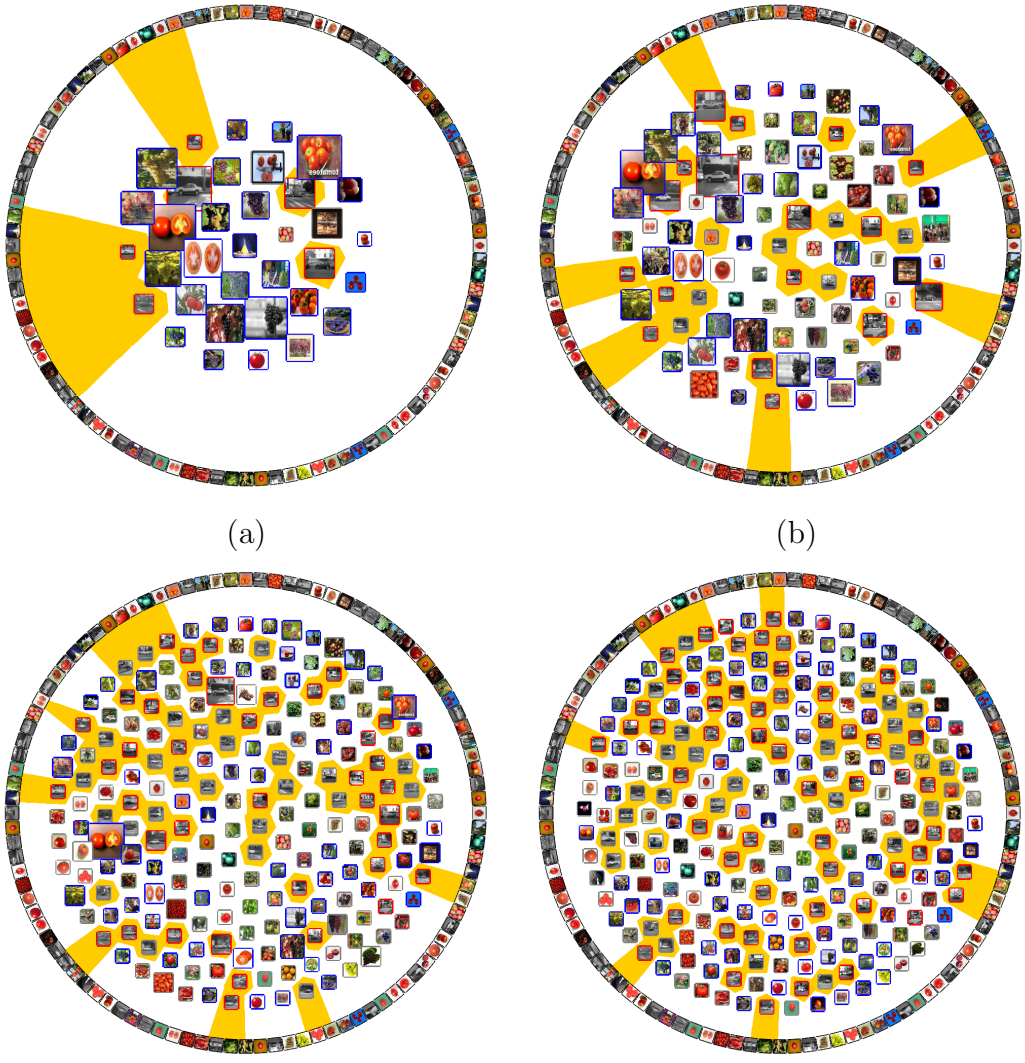


Figure 5.2: Discriminating car images using the support vector machine at multiple hierarchical levels. Images of the training set are labeled as red (car images) and blue (others). The inferred region of the car images is rendered in yellow through the Voronoi tessellation. (a) 10%, (b) 30%, (c) 40%, and (d) 100% of images. (#{visual words} = 100.)

(a)

(b)

Figure 5.3: Categorizing images of round objects from images of five different categories (tomatoes, coins, cars, CDs, and glasses). (a) Coarse level. (b) Fine level. (#{visual words} = 100.)

Chapter 6

Conclusion and Future Work

In this thesis, I have presented an approach to visualize image categorization within the high-dimensional feature space by taking advantage of the characteristics of the BoF model and graph drawing techniques. The idea behind my approach is to extract the bipartite relationships between the input images and visual words first and then visualize them as a network using the anchored map representation. This new type of dimensionality reduction framework successfully convinces us of the plausibility of resulting image categorization based on the BoF model. The readability of the anchored map representations have been further enhanced by seeking the optimal circular ordering of visual words and dendrogram-based hierarchical representation of images. Voronoi-based partitioning has been also incorporated into the central disk region of the anchored map to visualize the border of some specific image category.

As future work, fully classifying images of multiple categories according to users' preference remains to be tackled. The readability of the anchored map representations also depends on the quality of the sparse vector representations of the images in terms of the extracted visual words. Improving the sparse coding of such images in the BoF model is left as a topic of future research. Enhancing the interactivity of the present image retrieval system is also left as a future research theme.

Reference

[1] Anna Bosch, Andrew Zisserman, and Xavier Muñoz. Scene classification via pLSA. In *Proceedings 9th European Conference on Computer Vision (ECCV 2006)*, volume 3954 of *Springer Lecture Notes in Computer Science*, pages 517–530, 2006.

[2] Flavio Chierichetti, Ravi Kumar, Sandeep Pandey, and Sergei Vassilvitskii. Finding the Jaccard median. In *Proceedings 21st Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 293–311, 2010.

[3] Gabriella Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. Visual categorization with bags of keypoints. In *ECCV'04 Workshop on Statistical Learning in Computer Vision*, pages 1–22, 2004.

[4] Peter Eades. A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160, 1984.

[5] Jan Eichhorn and Olivier Chapelle. Object categorization with SVM: Kernels for local features. In *Advances in Neural Information Processing Systems (NIPS)*, 2004.

[6] Takayasu Fushimi, Yamato Kubota, Kazumi Saito, Masahiro Kimura, Kouzou Ohara, and Hiroshi Motoda. Speeding up bipartite graph visualization method. In *Proceedings Advances in Artificial Intelligence*,

volume 7106 of *Springer Lecture Notes in Computer Science*, pages 697–706, 2011.

[7] Gregory Griffin, Alex Holub, and Pietro Perona. Caltech-256 object category dataset. Technical report, California Institute of Technology, 2007.

[8] Kyle Heath, Natasha Gelfand, Maks Ovsjanikov, Mridul Aanjaneya, and Leonidas J Guibas. Image webs: Computing and exploiting connectivity in image collections. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010*, pages 3432–3439, 2010.

[9] K. E. Hoff III, T. Culver, J. Keyser, M. Lin, and D. Manocha. Fast computation of generalized voronoi diagrams using graphics hardware. In *Proceedings SIGGRAPH '99*, pages 277–286, 1999.

[10] Sergey Ioffe. Improved consistent sampling, weighted minhash and l1 sketching. In *Proceedings 10th IEEE International Conference on Data Mining 2010*, pages 246–255, 2010.

[11] Thorsten Joachims. *Text categorization with support vector machines: Learning with many relevant features.* Springer, 1998.

[12] J. B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.

[13] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178, 2006.

[14] D. G. Lowe. Object recognition from local scale-invariant features. In *Proceedings 7th IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157, 1999.

[15] G. M. H. Mamani, F. M. Fatore, L. G. Nonato, and F. V. Paulovich. User-driven feature space transformation. *Computer Graphics Forum*, 32(3):291–299, 2013.

[16] Kazuo Misue. Drawing bipartite graphs as anchored maps. In *Proceedings Asia-Pacific Symposium on Information Visualisation 2006 (APVis '06)*, pages 169–177, 2006.

[17] Kazuo Misue. Anchored map: graph drawing technique to support network mining. *IEICE Transactions*, 91-D(11):2599–2606, 2008.

[18] Kazuyo Mizuno, Hsiang-Yun Wu, and Shigeo Takahashi. Manipulating bilevel feature space for category-aware image exploration. In *Proceedings of the 7th IEEE Pacific Visualization Symposium (PacificVis 2014)*, pages 217–224, 2014.

[19] F. V. Paulovich, D. M. Eler, J. Poco, C. P. Botha, R. Minghim, and L. G. Nonato. Piecewise laplacian-based projection for interactive data exploration and organization. *Computer Graphics Forum*, 30(3):1091–1100, 2011.

[20] F. Perronnin. Universal and adapted vocabularies for generic visual categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(7):1243–1256, 2008.

[21] Shuji Sato, Kazuo Misue, and Jiro Tanaka. Readable representations for large-scale bipartite graphs. In *Knowledge-Based Intelligent Information and Engineering Systems*, pages 831–838, 2008.

[22] J. Sivic and A. Zisserman. Video google: a text retrieval approach to object matching in videos. In *Proceedings 9th IEEE International Conference on Computer Vision*, pages 1470–1477, 2003.

[23] W. S. Torgerson. Multidimensional scaling: I. theory and method. *Psychometrika*, 17(4):401–419, 1952.

[24] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. In *Proceedings 10th IEEE International Conference on Computer Vision*, volume 2, pages 1800–1807, 2005.

[25] Jianchao Yang, Kai Yu, Yihong Gong, and Thomas Huang. Linear spatial pyramid matching using sparse coding for image classification. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[26] Gao Yi, Hsiang-Yun Wu, Kazuo Misue, Kazuyo Mizuno, and Shigeo Takahashi. Visualizing bag-of-features image categorization using anchored maps. In *Proceedings of the 7th International Symposium on Visual Information Communication and Interaction*, pages 39–48, 2014.