One can't show high ideals without simple living; One can't have lofty aspirations without a peaceful state of mind.

– Liang Zhuge, 181-234.

University of Alberta

A MEASURE OF PERCEPTUAL ALIASING IN IMAGE DESCRIPTORS

by

Jiemin Wang

A thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

Fall 2013
Edmonton, Alberta

To my beloved family

Abstract

This thesis is concerned with a measure of perceptual aliasing in image descriptors. Perceptual aliasing occurs when the one-to-one mapping relations between world states (objects) and their representation (descriptors) are not maintained. Our method measures the discriminating power of an image descriptor in terms of its ability to distinguish between images of different objects and to match images of the same object. Specifically, our method runs spectral clustering on the similarity matrix computed with descriptors of known image clusters and measures the performance of an image descriptor by its ability to maintain the original clusters, using two indices, MRI-1 and MRI-2, that are based on the Rand index. Experiments on MRI versus precision and recall show that our proposed metrics are more appropriate for applications such as content-based image retrieval in which image clustering is a critical step.

Acknowledgements

Foremost, I would like to express my deepest appreciation to my supervisor Prof. Hong Zhang for the continuous support of my master study. His guidance, useful comments, remarks and persistent help through the learning process and research.

I would like to thank my committee members, Csaba Szepesvari and Nilanjan Ray for spending time on reading and commenting my thesis.

In addition, I would like to thank all my colleagues and friends who helped and supported me for the two years and made my life bright and colorful.

Last but not least, I would like to thank my beloved family for giving birth to me and showing love and support throughout my life.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Perception is a concept in psychology indicating the recognition and interpretation of the sensory information to understand the environment. The perceptual system in our brain makes it possible to identify the world around us even when the sensory information is changing, unstable or incomplete. Despite the complex and strong brain we have, it is still possible to misrecognize the environment, such as a place, a building, a plant, etc.

In computer vision, the machine (e.g. a robot) perceives the environment based on the images taken by a camera. To recognize the world, image matching has become a fundamental problem in many computer vision tasks, such as object recognition. Perceptual aliasing exists when two images are incorrectly matched or unmatched. To improve the matching rate and efficiency, people have introduced image descriptors to characterize an image. Existing performance metrics are almost exclusively dependent on the k nearest neighbor list of a query, i.e. the top retrieved images, which might not be appropriate to capture perceptual aliasing between other image pairs. The lack of measure of perceptual aliasing in image descriptors motivates us to design a new metric based on its original definition [47].

In this thesis, we propose a novel method to quantify perceptual aliasing in image descriptors. Perceptual aliasing within different applications can be different. Our method can serve as a reference when selecting image descriptor for a specific application. Section 1.1 defines the problem in detail and Section 1.2 describes the objective and contribution of this thesis. In Section 1.3, we overview the thesis.

1.1 Perceptual Aliasing

Perceptual aliasing used in characterizing a sensing process was proposed in 1991 [47]. In computer vision, the sensory information consists of images captured by a machine (e.g. a robot) and the decision is made by processing, analyzing and understanding the images. Perceptual aliasing arises in image matching as the machine understands the world and it occurs when the one-to-one mapping relations between the world states (objects) and the internal representation (descriptors) are not

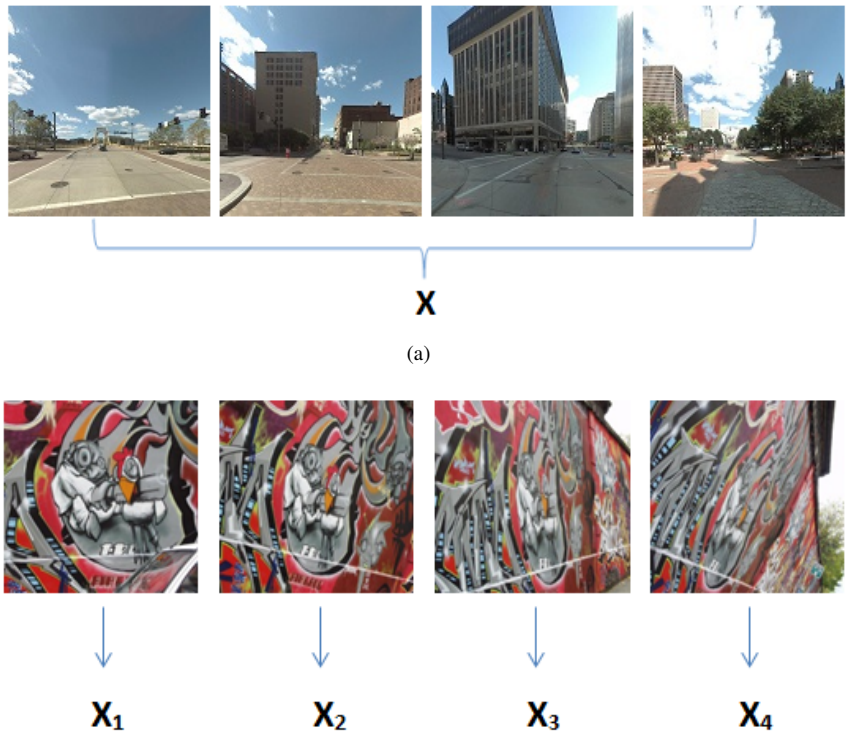


Figure 1.1: Examples of two types of perceptual aliasing. (a) Type 1, images of multiple world states are perceived the same with descriptor vector X. (b) Type 2, one world state has multiple internal representation, i.e., the descriptor vectors X_i, where $X_i \neq X_j$ if $i \neq j$.

maintained. There are two types of perceptual aliasing [47]. First, multiple world states share the same internal representation, i.e., images of different objects are perceived the same according to their image descriptors. Second, one world state has more than one internal representation, i.e., images of the same object are described by different image descriptors. Simple examples are shown in Figure 1.1. The first type of perceptual aliasing is illustrated in Figure 1.1(a) where different world states, i.e., different places, are perceived the same with the descriptor X. Figure 1.1(b) offers an example of the second type of perceptual aliasing. In this case, images of one world state, the graffiti, have four different descriptors. Ideally, we expect one world state corresponds to one internal representation and vice versa. These two types of perceptual aliasing characterize an image descriptor in terms of whether it retains the inherit similarities and differences between images.

People who are interested in image matching related works, such as image retrieval, object recognition and robot localization, face the curse of perceptual aliasing. It is important to take the issue into account when selecting an appropriate image descriptor within an application. We focus on designing a measure in quantifying these two types of perceptual aliasing in this thesis.

1.2 Thesis Objective and Contribution

The work presented in this thesis offers a simple and novel measure of perceptual aliasing. Extensive literature exists on comparison of image descriptors. Most of the works make use of precision and recall or their variants to evaluate the performance, such as mean average precision (mAP), recall@R and average normalized rank of relevant images [39]. These works mainly depends on the k nearest neighbor list of a query. The way these performance metrics are computed ignores the similarities and differences between each image pair but only focuses on the similarities between the query image and the top retrieved ones, therefore leading them to be insensitive to the two types of perceptual aliasing. The similarity information between all image pairs is important in performance evaluation within clustering-based image applications to avoid overlap between image clusters. For instance, with the proliferation of web and digital images, there are millions to billions of images, and clustering has been taken as a pre- or post-processing step to organize the data. Overlap between two image clusters, which can either hurt the efficiency or deteriorate the accuracy of the system, is not expected. Specifically, image search results are returned with clusters to facilitate users' browsing [6, 3, 20]; distinctive visual words need to be clustered to build a big dictionary [39, 41, 37]; similar images are clustered into scenes or views to improve the efficiency and accuracy in place recognition applications [18, 42, 21]; images are organized into clusters to distribute into different machines as a pre-processing step to improve the scalability of an image processing system [25]. Existing performance metrics, e.g. precision and recall, might not be appropriate in such applications since it can be non-trivial to do the linear search in a huge dataset and the k nearest neighbor list cannot capture the overlap information between each two image clusters. The goal of our work is to define the indices for the two types of perceptual aliasing based on its original definition and to propose a method using spectral clustering to quantify perceptual aliasing in image descriptors. The usage of clustering algorithm gives us the opportunity to take into account the similarity between all image pairs and the overlap between image clusters. Our method can assist in the process of descriptor selection in both clustering-based and non-clustering-based applications.

In this thesis, we make two contributions:

- **We define indices for perceptual aliasing.** An optimal image descriptor should minimize perceptual aliasing when it is used to compare images, i.e., similar images remain similar in the descriptor space and vice versa. This leads to the idea of constructing a dataset consisting of clusters of images. Images in the same cluster belong to the same object. Images in different clusters belong to different objects. We expect the same similarity and dissimilarity properties in the corresponding descriptor space. To quantify the comparative analysis, we borrow the technique in clustering analysis of using the Rand index [38] to evaluate the similarity between the set of clusters in the original image space and the set of clusters in descriptor space and define the indices for perceptual aliasing, i.e., modified Rand indices MRI-1 and MRI-2.

- **We offer a method to measure perceptual aliasing in image descriptors.** In order to measure perceptual aliasing in image descriptors for a specific application, we provide a method using our proposed perceptual aliasing indices to compare image descriptors. Given the dataset within a specific application, the performance can be evaluated if the similarity can be computed between image pairs. Clusters of images are collected for the application scenario and the similarity matrix is computed correspondingly. The usefulness of spectral clustering [28] makes it possible to do the clustering on the similarity matrix to generate the clusters in descriptor space even when the descriptors are of no fixed length or not available explicitly. With the clusters in image space and in descriptor space, we are then able to evaluate the mapping relations between the world states (objects) and the internal representation (descriptors) by computing the indices of perceptual aliasing.

1.3 Organization

The thesis is organized as follows. Chapter 2 reviews the common image descriptors and the performance metrics used in image descriptors comparison, followed by the description of the faultiness of these metrics in evaluating perceptual aliasing. Chapter 3 introduces the indices for the two types of perceptual aliasing in detail. Then, Chapter 4 presents the performance comparison method in regard to perceptual aliasing with the proposed indices. To illustrate the utility of the method, Chapter 5 provides the experimental results on different datasets and gives the related explanations. Finally, the thesis is concluded with a discussion in Chapter 6.

Chapter 2

Background and Related Work

In this chapter we present the background and related work of the thesis. The topics include: image descriptors and performance metrics. The motivation of proposing a measure of perceptual aliasing in image descriptors is also illustrated.

2.1 Introduction

Image description can be a critical step in many computer vision applications, such as content-based image retrieval (CBIR), location recognition, robot mapping and localization. Generally, there are two strategies to create an image descriptor. First, an image descriptor can rely on the detection of local keypoint feature, and the description of the keypoint in terms of textural information around the keypoint, which is called local keypoint descriptor. Alternatively, one can construct an image descriptor without detecting keypoints but by characterizing a whole image, which we refer to as whole image descriptor or global image descriptor in this thesis. Perceptual aliasing in image descriptors is a key issue in image matching tasks where an optimal image descriptor will minimize the problem when used to match two images. However, the existing performance metrics are not sensitive to the two types of perceptual aliasing and are not appropriate to measure the problem.

Extensive literature exists on comparison of keypoint descriptors [31, 12, 10, 14, 43, 24, 5]. Precision and recall are the most common performance metrics, and the computation is based on the matched detected keypoints. Other evaluation techniques are usually used in the context of an application where each competing alternative is substituted in turn and a performance metric is defined as the basis for choosing the optimal image descriptor for this application. The way these performance metrics are computed relies on the k nearest neighbor list of a query, therefore ignoring the similarities and differences between other image pairs but only focuses on the similarities between the query image and the top retrieved ones, and making them improper to measure perceptual aliasing.

We offer a novel measure of perceptual aliasing which originates from the definition [47]. With the evaluation of the two types of perceptual aliasing, one can select an optimal image descriptor within an application.

2.2 Image Descriptors

Image descriptor characterizes an image so that the image matching can be executed effectively and efficiently. Local keypoint descriptors characterize the keypoint patches detected by the keypoint detector, e.g. Harris Corner detector [13]. In accordance with different ways of computing local keypoint descriptors, there are following categories: gradient-based descriptors, binary descriptors and bag-of-words (BOW) descriptors [39]. Recently, people found that image descriptors can also be created for the whole image, which attain comparable performance and achieve efficiency [1]. The computation of gradient-based descriptors and binary descriptors can also be applied to generate whole image descriptors. In this case, the whole image is taken as a "keypoint", and the descriptor is used to characterize the whole image. Besides, filter-based descriptors, e.g. GIST [35], are developed to capture the structural information of a whole image. An overview for each category in local keypoint descriptors and whole image descriptors is presented in the following sections.

2.2.1 Local Keypoint Descriptors

To use local keypoint descriptors to describe an image, a feature detector is needed to detect the interest points in the image. Harris Corner [13], Hessian Affine [30], SIFT [26, 27], SURF [2] and MSER [29] are some of the most common feature detectors. After detecting the keypoints, local keypoint descriptors are computed to characterize the neighborhood around these keypoints. We are interested in the performance of different local keypoint descriptors in perceptual aliasing with respect to different changes in image conditions, such as scale, rotation, viewpoint and illumination.

Gradient-based Descriptors

The gradient property of pixels or a patch of pixels around the keypoint is usually made use of to capture the texture information of the interest point. The common gradient-based descriptors include: SIFT [27] and SURF [2].

- SIFT: Scale Invariant Feature Transform. Each image patch corresponding to a keypoint is rotated in accordance with the dominant orientation of the patch, which results in the rotation invariance property of the descriptor, and then is divided into 4×4 subregions. A histogram with eight bins indicating eight gradient orientations is created for each subregion as shown in Figure 2.1. The gradient orientation is added to the histogram after weighting by the gradient magnitude, and the gradient magnitude and orientation are computed in Equations (2.1) and (2.2), where L is the Gaussian smoothed image at a specific scale, $m(x, y)$ and $\theta(x, y)$ indicate magnitude and orientation at point (x, y) respectively. The final $4 \times 4 \times 8 = 128$ dimensional SIFT descriptor is normalized to a unit vector. SIFT descriptor is proposed together with its scale invariant detector, leading to its well-known scale invariance property. In addition, the computation of gradient information gives it viewpoint and illumination invariance. With the

Figure 2.1: An image patch is divided into 4×4 subregions and weighted gradient orientation is added to the eight bins/orientations histogram in each subregion. The final descriptor consists of $4 \times 4 \times 8$ dimensions.

invariance to common image condition changes, SIFT descriptor has been applied to image retrieval, place recognition, image segmentation, etc. However, the inefficient computation impedes its usefulness in real-time applications, such as SLAM (Simultaneous Localization and Mapping). It triggers the interest in finding a comparable image descriptor with SIFT descriptor but achieving more efficiency. SIFT descriptor is usually taken as the baseline to compare the performance of other image descriptors.

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \tag{2.1}$$

$$\theta(x,y) = \tan^{-1}((L(x,y+1) - L(x,y-1))/(L(x+1,y) - L(x-1,y))) \tag{2.2}$$

- SURF: Speeded Up Robust Features. The intent of SURF descriptor is to speed up the computation of SIFT descriptor at the cost of some performance. It divides the neighborhood of a keypoint into 4×4 subregions and computes the Haar wavelet response along x and y directions respectively. Instead of computing gradient based on the intensity of pixels, Haar wavelet calculates the gradient in terms of the intensity of pixel patches, which obviates the burden in pixel level computation by relying on the integral images. The descriptor consists of the summation of Haar wavelet response and their absolute values along x and y directions as shown

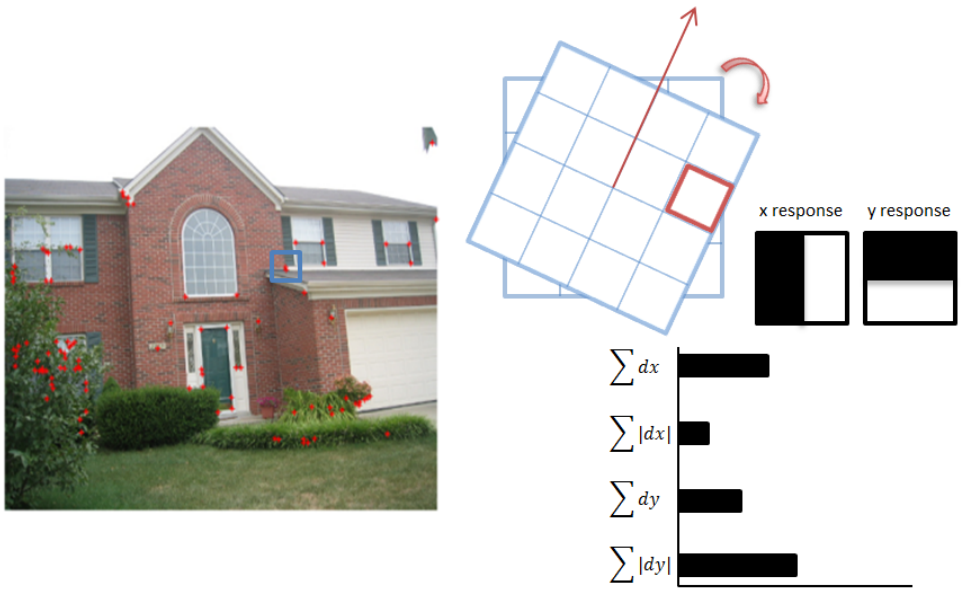


Figure 2.2: For each detected interest point, its neighborhood is divided into 4×4 subregions. The Haar wavelet response and its absolute value is summed vertically and horizontally in each subregion. The final SURF descriptor is the concatenation of $v = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|)$ in all subregions.

in Figure 2.2. SURF descriptor preserves the invariance property of SIFT descriptor to some extent. Moreover, in applications like SLAM, the camera seldom rotates so that the computation of SURF descriptor can be further optimized without including rotation information [2]. It is interesting to investigate the performance of SURF descriptor within an application since SURF descriptor is a common substitute to SIFT descriptor when the efficiency matters.

Binary Descriptors

Recently, binary image descriptors have been proposed such as BRIEF [4], BRISK [23] and FREAK [36]. The computation of binary image descriptors is usually based on intensity comparison with predefined binary tests defined in Equation 2.3, where $p(x)$ denotes the intensity of pixel x and τ indicates one bit in the binary image descriptor. The comparison of pixel intensity is somewhat independent of absolute pixel values, which makes the descriptor insensitive to illumination changes. The difference between different binary descriptors exists in the way that the pixel sampling pattern is defined. BRIEF makes use of Gaussian distribution to find point pairs as shown in Figure 2.3. BRISK and FREAK define their explicit sampling patterns to capture meaningful texture information. The binary image descriptor is compact and easy-computed to facilitate huge gains in efficiency. The disadvantage is their sensitivity to image distortions and transformations, such as viewpoint and rotation changes, which hinders their application in some scenarios.

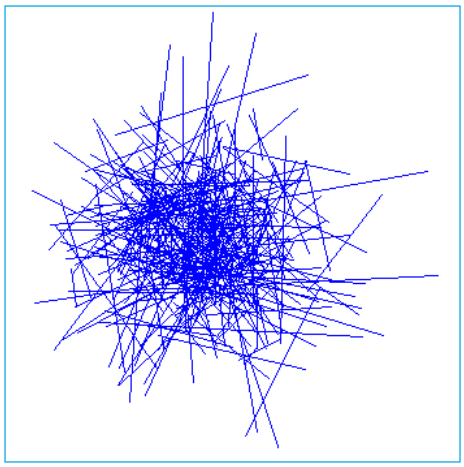


Figure 2.3: Illustration of sampling pattern of BRIEF. Point pair $(x, y) \sim i.i.d.Gaussian\,(0, \frac{1}{25}S^2)$. S is the size of the keypoint patch $S \times S$.

$$\tau(p; x, y) := \begin{cases} 1 & if\ \ p(x) < p(y) \\ 0 & otherwise \end{cases} \tag{2.3}$$

BOW: Bag-of-words

Because an image may contain hundreds or even thousands of keypoints, bag-of-words (BOW) is introduced from text search techniques to generate concise frequency histogram of the detected keypoints [39] as shown in Figure 2.4. The performance of BOW depends on the discrimination of the visual words vocabulary, as well as the number of words used to bin the local feature descriptors [48]. It is interesting to investigate the parameters and find an optimal setting. To improve the performance of BOW method, pyramid BOW has recently been proposed to preserve more information of the image by generating an image descriptor with the concatenation of the weighted BOW descriptors computed from increasingly fined subregions of the original image [22]. The performance of BOW based method is highly associated with the performance of the chosen local feature descriptors. Parameters have to be carefully tuned to optimize the performance.

Summary

Local keypoint descriptors can be used to describe an image in terms of its characteristic regions and detailed information. There could be hundreds or thousands of keypoint descriptors within an image so that image matching using local keypoint descriptors can be computation and memory costly. The performance of the descriptors is affected by both of the keypoint detector and keypoint descriptor. Different keypoint detectors or descriptors need to be selected for different application scenarios. There is no general solution to the problem of performance comparison of image descriptors and selection of optimal ones. The lack of work on the evaluation and comparison of local keypoint

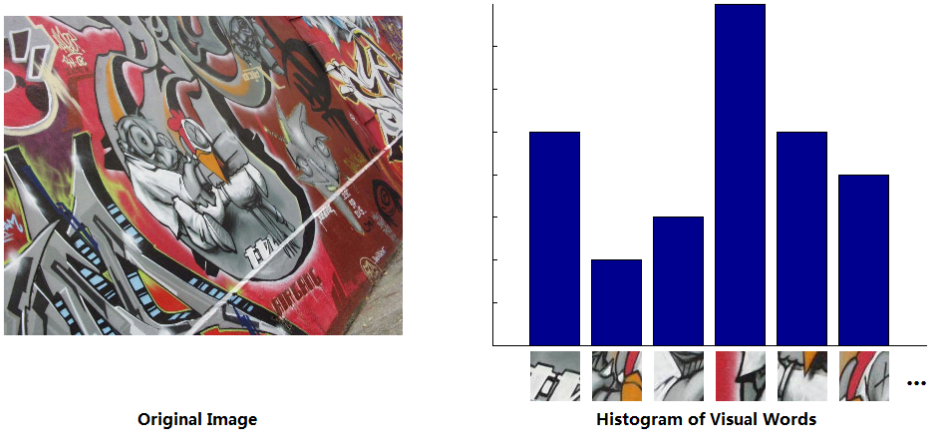


Figure 2.4: Illustration of bag of words method. Hundreds of visual words are extracted from the original image. The final image descriptor consists of a frequency histogram accumulated by the visual words.

descriptors in terms of perceptual aliasing interests us to design a proper method to quantify the problem and compare the popular ones under different changes in image conditions.

2.2.2 Whole Image Descriptors

[1] found that by taking the whole image as a keypoint and creating an image descriptor which describes the neighborhood around the center point of the image, we can obtain a whole image descriptor that is more compact than local keypoint descriptors. Meanwhile, the whole image descriptor achieves higher efficiency, requires less memory storage, as well as presents competing distinctiveness. In the following sections, we overview the common whole image descriptors and discusses their properties.

Gradient-based Descriptors

WI-SIFT and WI-SURF [1] are the global versions of the corresponding local normalized keypoint descriptors SIFT and SURF. "WI-" stands for "whole image". In this case, we downsample the image to a smaller image patch, e.g. 128×128 pixels, take the center point of the image as a keypoint and create a SIFT or SURF image descriptor in the same way as we compute the local descriptor. The whole image descriptors do not require keypoint detection and is promising because of their comparable performance with the local version. Another attractive property of WI-SIFT and WI-SURF is the very concise representation of a whole image, which facilitates efficiency in real time applications.

HOG (Histograms of Oriented Gradients) is another global image descriptor based on gradient information over a dense grid of overlapping spatial blocks in the original image [9]. The locally

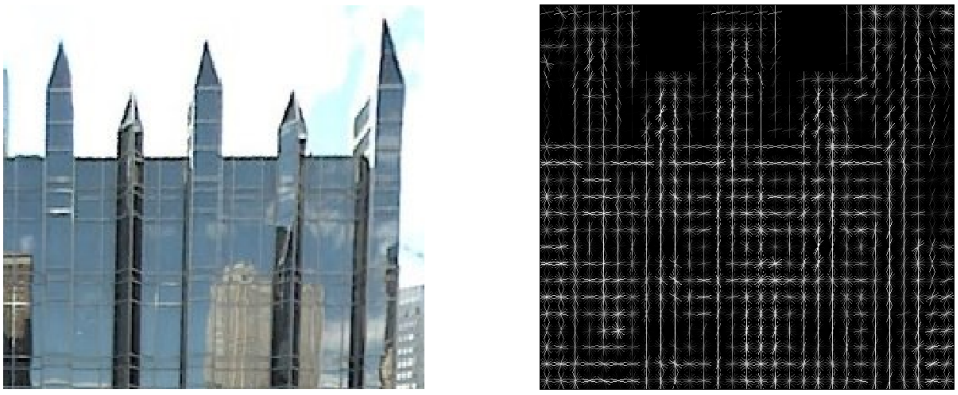


Figure 2.5: Example of Histograms of Oriented Gradients descriptor with a cell size of eight pixels. HOG is an array of cells. Each cell contains the feature components. The size of HOG in this case is $23 \times 29 \times 31$. The dimension of HOG feature in each cell is 31.

normalized histogram of gradient orientations features are similar to SIFT feature. HOG is able to capture the structural information, which gives good performance in applications like people or car detection. Figure 2.5 shows an example of the original image and its corresponding HOG features.

Binary Descriptors

BRIEF-Gist [40] is a global image descriptor based on the idea of the local feature descriptor BRIEF. Rather than extracting local binary image descriptors by comparing intensity of local keypoint pairs, BRIEF-Gist downsamples the original image to proper descriptor patch size and builds the binary image descriptor for a whole image. To preserve information of the original image and create distinctive image descriptor, BRIEF_$n \times n$ divides the image into $n \times n$ tiles, builds descriptor for each tile and concatenates them into one image descriptor. BRIEF-Gist is designed to be applied to appearance-based place recognition system and proves to perform comparatively in large scale SLAM with the state-of-the-art techniques [40].

Filter-based Descriptors

Gist descriptor [35] characterizes an image in terms of its response to a Gabor filter bank. The property of Gabor filters gives them the advantage in detecting the structural information of the image content, so as to determining the degree of naturalness, openness, roughness, etc. of an image. The structural information is a useful feature in representing an image so that many applications have adopted GIST descriptor in image description step. Figure 2.6 shows an example of GIST descriptor of a street view. In this case, the image is divided into 4×4 tiles, and the final descriptor consists the average response of each tile. 32 Gabor filters are used and the dimension of the descriptor is therefore $4 \times 4 \times 32 = 512$. The performance of GIST descriptor is highly dependent on the content of the image - GIST works well with outdoor environment and works poorly in other situations -

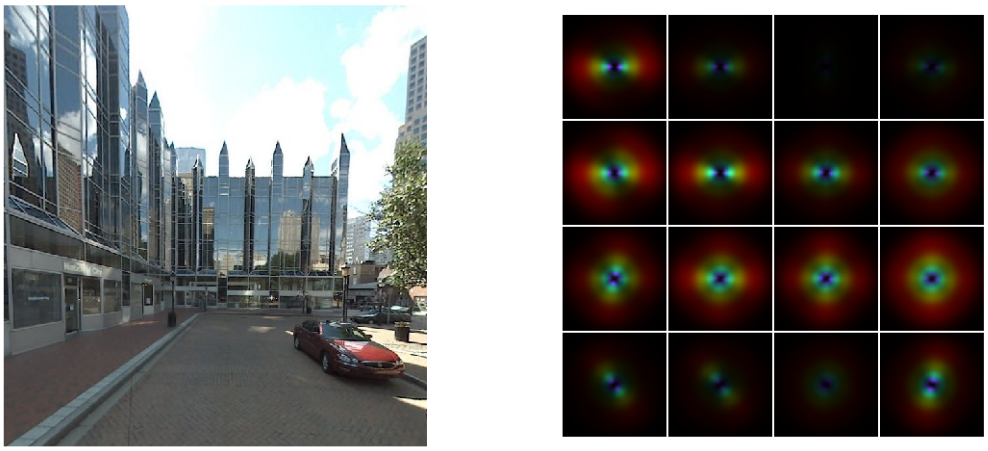


Figure 2.6: Example of GIST descriptor. 32 Gabor filters are used with eight orientations and four frequencies. The response image is divided into 4×4 tiles and the final descriptor consists of the average response of each tile. The final descriptor is $4 \times 4 \times 32 = 512$ dimensions.

which restricts its use for different applications [49].

Summary

The extraction and computation of whole image descriptors require much less time and memory than the local versions, therefore alleviating the load of the CPU and making it possible to be applied in real-time tasks. However, it is inevitable to lose detailed information by describing the whole image only, which results in performance decrease. The investigation and comparison of performance of whole image descriptors becomes an important task since it can help find a promising global image descriptor which has the potential to replace local keypoint descriptors within some application.

2.3 Performance Metrics

A measure of perceptual aliasing in image descriptors is proposed in this thesis. The discriminating power of the image descriptors in terms of their ability to distinguish images of different objects and match images of the same object is evaluated. The existing performance metrics in comparing image descriptors are either based on precision and recall or related with image retrieval application, which only focuses on the relations between the query keypoint/image and the top retrieved ones, while ignoring the similarity and dissimilarity between the other keypoint/image pairs. The motivation and methodologies of common performance metrics of image descriptors are analyzed in this section. Toy example has been provided to illustrate the weakness of existing performance metrics in terms of measuring perceptual aliasing.

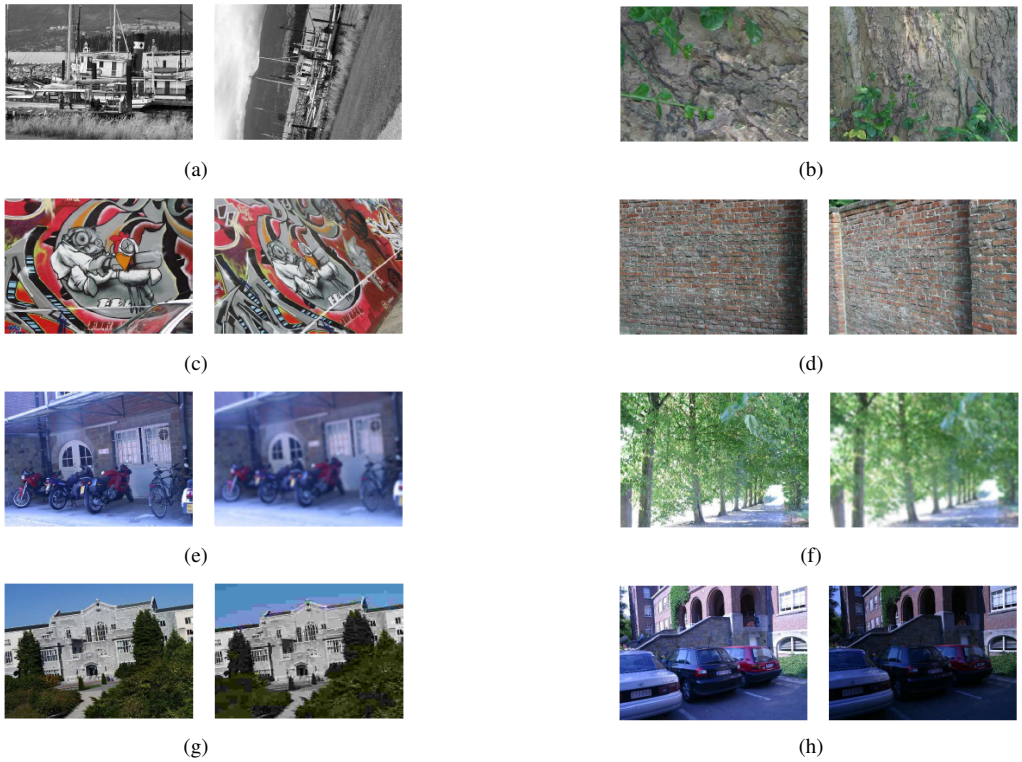


Figure 2.7: Dataset for keypoint descriptors evaluation. (a)(f): Zoom+rotation. (d)(d): Viewpoint. (e)(f): Image blur. (g): JPEG Compression. (h): Illumination.

2.3.1 Precision and Recall

When an image is described, perceptual aliasing introduced by the description step can be indirectly determined by the precision and the recall computed from matching the image descriptors. Extensive literature exists on comparison of keypoint descriptors [31, 12, 10, 14, 43, 24, 5] which include SIFT [27], SURF [2] and the more recent BRIEF [4], and the focus of these studies is often on the invariance properties in regard to scale, rotation, illumination, etc. Performance is measured in terms of repeatability of the keypoints, and the precision and the recall of matching detected keypoints. On the other hand, precision and recall can also be applied in the evaluation of performance of image descriptors in the context of image retrieval [8, 50]. By tuning the threshold on the similarity values, precision and recall are computed by counting the true and false returned images and the ground truth matched images based on the query one. Unfortunately, it is inappropriate to quantify perceptual aliasing using precision and recall because of their lack of ability to measure the discriminating power of image descriptors on distinguishing images of different objects and matching images of the same object but only focus on the similarities between the query image and the correct retrieved ones.

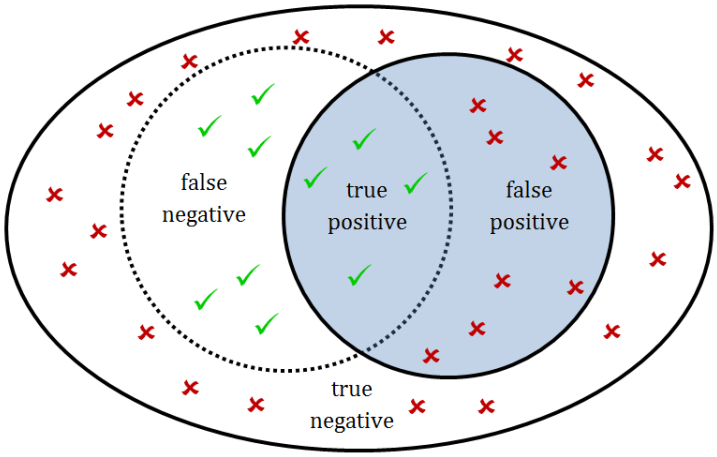


Figure 2.8: Venn diagram for true positive, false positive, true negative and false negative.

Precision and Recall in Local Keypoint Descriptor Matching

The seminal work of Mikolajczyk and Schmid [31] presented perhaps the most popular framework for comparing the performance of keypoint descriptors. They carefully constructed image datasets under various image condition changes including affine, compression, blur, scale, rotation and illumination as shown in Figure 2.7, and measured the performance of keypoint detectors in terms of repeatability and that of keypoint descriptors in terms of 1-precision and recall as defined in Equation (2.4, 2.5),

$$1 - precision = \frac{\text{FP}}{\text{TP} + \text{FP}} \tag{2.4}$$

$$recall = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2.5}$$

where FP, false positive, is the number of incorrectly matched keypoint descriptors, TP, true positive, is the number of correctly matched keypoint descriptors and FN, false negative, is the number of missing matched keypoint descriptors. Figure 2.8 shows the Venn diagram to illustrate the relation of the above variables.

For each detected keypoint, only one matched keypoint is found by computing the distance between the keypoint descriptors, therefore ignoring the distance from the other keypoints in which perceptual aliasing can occur if the distance is still small and several keypoints can be misperceived as the same.

Precision and Recall in Object Tracking

Gauglitz et al. [12] more recently compared local keypoint descriptors in the context of object tracking in video. Their work included up-to-date descriptor algorithms and examined their keypoint matching performance in consecutive video frames, which due subject to motion continuity involve

more predictable transformations than the data sets in [31]. As in [31], precision of 1NN (the first nearest neighbor) of matched keypoints, i.e., the descriptor with the smallest distance is identified as the matched keypoint descriptor, are used as the metrics of performance. In this study, the comparison is for keypoint descriptor performance evaluation in video tracking application. But precision of 1NN suffers from the same problem as previous [31].

Precision and Recall in Image Retrieval

Mark and Paul [8] develops a probabilistic approach to reduce perceptual aliasing in place recognition problems. To evaluate the performance of their approach, they make use of precision and recall in the context of loop closure detection in SLAM as the performance metrics. By tuning the probability at which a loop closure is detected, precision and recall curves are generated for their proposed method. Instead of computing precision and recall by matching the local keypoint descriptors, Mark and Paul's comparison is conducted on the image matching tasks. Plenty of researches take advantage of precision and recall in an specific application scenario to compare the performance of different image descriptors. To do the comparison within an application, complex experiment environment or framework needs to be constructed, which is inefficient and non-trivial.

Toy Example in Perceptual Aliasing

Precision and recall are usually used to compare the performance of image descriptors. However, they are insensitive to the perceptual aliasing problem, which will be explained by the following toy examples.

Figure 2.9 shows a toy example in which 6 objects are indicated by different shapes and each of them has 5 images. The dataset consists of images of the 6 objects. Figure 2.10 illustrates two image retrieval cases to explain the insensitivity of precision in measuring perceptual aliasing. A "star" is used to start a query and 10 images are retrieved correspondingly. In both cases, precision equals $\frac{5}{10}$. But the perceptual aliasing in case 2 is more serious than that in case 1 since the similarity between images of different objects is higher and multiple objects can share the same internal representation. Precision can not capture the different levels of perceptual aliasing in these cases. On the other hand, Figure 2.11 shows another two image retrieval results based on the query "star". Recall in both cases are the same, i.e., 1, since all relevant "star" images have been retrieved. However, the "square" in case 2 is highly possible to be misperceived as a "star", which means case 2 suffers from more serious perceptual aliasing. Recall is insensitive to perceptual aliasing problem.

The inaccuracy of precision and recall in capturing perceptual aliasing is due to their computation based on the k nearest neighbor list of a query. The similarity between other image pairs where perceptual aliasing can occur is ignored. This disadvantage makes them inappropriate to be applied to the evaluation of image descriptors in clustering-based image applications where similarity between all image pairs needs to be considered to avoid clusters' overlap. Clustering technique

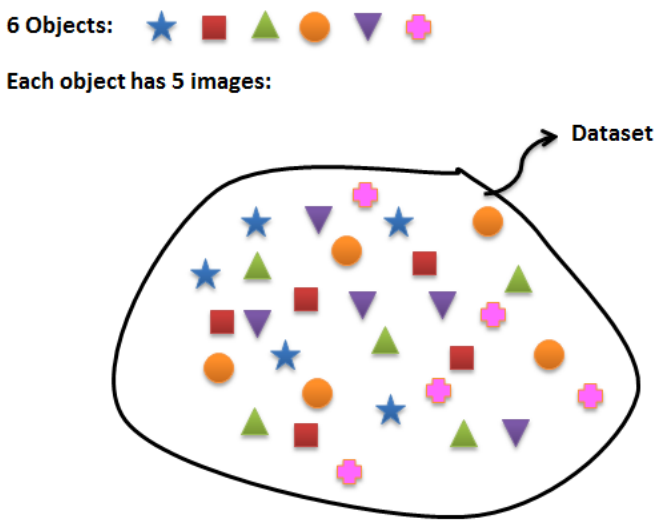


Figure 2.9: Toy example: 6 objects are indicated by different shapes and each object has 5 images.

is common in computer vision and image processing, especially with large scale dataset. One example is that big dictionary needs to be constructed to express different objects in computer vision applications using bag-of-words method [39, 41, 37]. To differentiate objects, the visual words are required to be distinctive and representative. It is important to find an image descriptor suffering from less perceptual aliasing when building a dictionary. Another example is that image clustering is executed to cluster images into scenes or views in place recognition applications to improve accuracy and achieve efficiency [18, 42, 21]. In addition, some large scale image retrieval system searches results by image clusters to facilitate users' browsing [6, 3, 20]. In such application, we expect images in the same cluster to be similar and images in different clusters to be dissimilar. A performance metric which measures the ability to remain the similarity and dissimilarity between image pairs in descriptor space is needed. Moreover, clustering play an important role in data organization as a pre-processing step for large scale dataset [25]. In this case, data, i.e., images, need to be clustered and distributed to several machines to improve the scalability and efficiency of the image processing system. The overlap between image clusters can be a serious problem since it can place a heavy burden in computation and destroy the parallel processing if more machines have to be touched. The lack of ability in measuring the discriminating power of image descriptors between image pairs by existing performance metrics motivates us to define a novel measure of perceptual aliasing.

2.3.2 Other Related Metrics

Rather than precision and recall, other metrics have been developed for evaluating the performance of an image descriptor, and they are almost exclusively always in the context of specific applications such as content-based image retrieval or place recognition and dependent on the k nearest neighbor

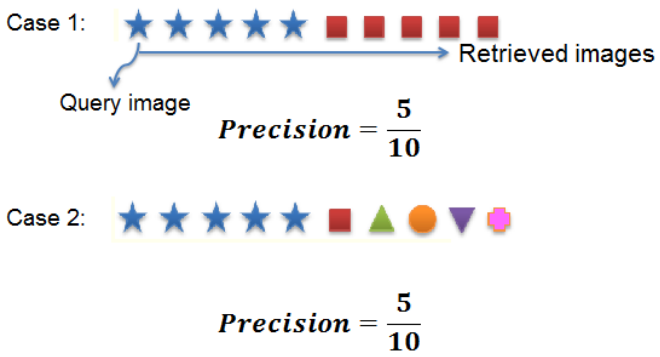


Figure 2.10: Two image retrieval results. The "star" is used to start a query. The precision in these two cases are the same.

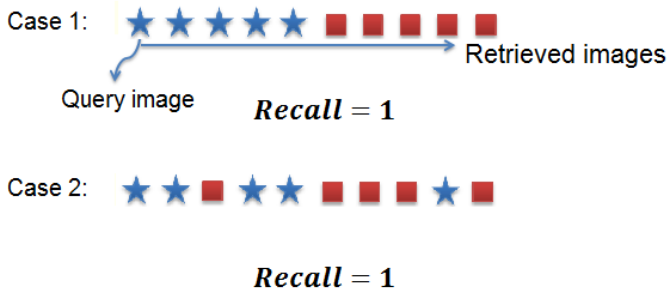


Figure 2.11: Two image retrieval results. The "star" is used to start a query. The recall in these two cases are the same.

list of a query as well. These metrics are designed to measure different aspects of image descriptors. They have the same disadvantage as precision and recall and might not be appropriate to measure perceptual aliasing. In addition, metrics proposed in information retrieval are used to measure the performance of any information retrieval method, which can also be applied to compare the performance of image descriptors within image retrieval application. The evaluation is based on the retrieval list of a query, which either only focuses on the correct matches or needs a ground truth ranking list explicitly. However, in perceptual aliasing measurement, similarities between all image pairs need to be considered, and no ground truth ranking list exists. The metrics in information retrieval are not appropriate to measure perceptual aliasing.

Generalizing Precision and Recall: mAP, recall@R

Generalizing precision and recall, [17] proposed mean average precision (mAP) and recall@R to compare the performance of the proposed compact global image descriptor with the state-of-the-art image descriptors in large scale image retrieval. Intuitively, mAP computes the average area under the precision-recall curve for a set of queries, thereby eliminating the need for choosing recall levels. mAP is defined as,

$$mAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q} \tag{2.6}$$

where Q is the number of queries, $AveP(q)$ is the average precision of a single query q, i.e., the mean of the precision scores after each relevant image retrieved.

Recall@R, on the other hand, computes the recall for the first R returned images, in order to overcome the difficulty in setting different recall levels for the high number of returned images in large scale image retrieval.

Average Normalized Rank of Relevant Images

To measure the quality of the image retrieval result, average normalized rank of relevant images, with values ranging between 0 and 1, was proposed in [33]. This metric was later used in the video Google work [39] by Sivic and Zisserman to analyze the performance of their visual BoW image representation. The query images have to be substituted each time and the retrieval has to be run repeatedly to evaluate the performance of image descriptors. The computation of average normalized rank of relevant images is defined as,

$$\widetilde{Rank} = \frac{1}{NN_{rel}}(\sum_{i=1}^{N_{rel}} R_i - \frac{N_{rel}(N_{rel}+1)}{2}) \tag{2.7}$$

where N is the total number of images in the dataset, N_{rel} is the number of relevant images, R_i is the rank of the ith relevant image. The $\widetilde{Rank}$ takes only the rank of relevant images into account without considering the similarity between other image pairs, leading it inappropriate to measure perceptual aliasing.

Correct Matched Images in Top x-percent

Another possible performance metric for image retrieval was based on top relevant images [34], motivated by the intuition that the correct images have to be at the top of the matched list for the image descriptor and the retrieval algorithm in general to be effective. In this case, the performance is defined in terms of the percentage of ground truth images that are among the top x percent of the returned images. Again, this performance metric pays attention on correct matched images while ignoring the possible perceptual aliasing in other image pairs.

Mean Reciprocal Rank

Mean reciprocal rank is a index in information retrieval evaluating the performance of any process that produces a list based on a query [45]. It computes the average reciprocal ranks of correct matches of queries,

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \tag{2.8}$$

where Q is a set of queries, $|\cdot|$ indicates the cardinality of a set, and $rank_i$ is the rank of the correct matches of the ith query. The mean reciprocal rank only takes into account the rank of the correct matches, therefore ignoring the similarities between other image pairs. The index values for both cases in Figure 2.10 are equal, which means this index is insensitive to perceptual aliasing.

Cumulative Gain

Cumulative gain measures the effectiveness of an information retrieval method [16]. It computes the sum of the relevance values of all results in a returned list based on a query. The cumulative gain (CG) at rank position p is defined as,

$$CG_p = \sum_{i=1}^{p} rel_i \tag{2.9}$$

where rel_i is the relevance value of the ith returned result. There are two other versions of CG: discounted cumulative gain (DCG) and normalized DCG [46, 7]. DCG penalizes the highly relevant results that appear lower in the returned list by reducing the relevance values logarithmically proportional to the rank of the results, which is defined as,

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2(i)} \tag{2.10}$$

Since the length of the result lists can vary based on different queries, normalized DCG uses the maximum DCG till p in the sorted resulted list by relevance, which is called ideal DCG (IDCG) to normalize the DCG across all queries. The normalized DCG can be computed as,

$$nDCG_p = \frac{DCG_p}{IDCG_p} \tag{2.11}$$

The CG, DCG and normalized DCG need explicit relevance value for each returned result. Their computation is still based on the returned results list. In measuring perceptual aliasing, we do not have the ground truth relevance values and we are interested in the relations between all image pairs. The above metrics are not appropriate in our case.

Kendall Tau Rank Distance

Kendall tau rank distance measures the distance between two ranking lists based on the number of pairwise disagreements [19, 11]. The distance between two ranking lists L_1 and L_2 is defined as,

$$K(\tau_1, \tau_2) = |\{(i,j) : i < j, (\tau_1(i) < \tau_1(j) \wedge \tau_2(i) > \tau_2(j)) \vee (\tau_1(i) > \tau_1(j) \wedge \tau_2(i) < \tau_2(j))\}| \tag{2.12}$$

where τ_1 and τ_2 are the rankings of results in lists L_1 and L_2. The smaller the distance, the more similar the two ranking lists are. In our case, the ground truth ranking list of all images does not exist. We are not able to borrow the Kendall tau distance to measure perceptual aliasing.

Purity

Purity is a measure for clustering quality [51]. It assigns the class of a cluster with the class of the most frequent elements in the cluster and counts the number of the dominant element for each cluster. Purity can be defined as,

$$Purity(\Omega, \mathrm{C}) = \frac{1}{N} \sum_{k} \max_{j} |\omega_k \cap c_j| \tag{2.13}$$

where $\Omega = \omega_1, \omega_2, \ldots, \omega_K$ indicates a set of clusters, $\mathrm{C} = c_1, c_2, \ldots, c_J$ indicates a set of classes, N is the number of elements in all clusters. The purity values of the two cases in Figure 2.10 are equal, which is $\frac{5}{10}$, if we take the two retrieved lists as two clusters. However, the severity of perceptual aliasing in these two cases is different. The purity is computed based on element count, which ignores the relation between images. Hence, it is inappropriate to measure perceptual aliasing.

In this section, we overview three performance metrics in image retrieval. These metrics are based on the k nearest neighbor list of a query, therefore ignoring the similarities in all image pairs and being insensitive to perceptual aliasing. We also introduce three metrics in information retrieval, which either focuses only on correct matches or needs ground truth ranking lists. In perceptual aliasing measurement, relation between all image pairs need to be considered and no ground truth ranking lists exist. The last metric purity is used to measure the clustering quality. However, its computation is based on element count and neglects the relation between images. As far as we know, the existing performance metrics are inappropriate to measure perceptual aliasing.

2.4 Summary

In this chapter, we briefly introduce some of the state of the art local and global image descriptors. By describing local keypoints, local image descriptors preserve more information about the original image than global image descriptors. Global image descriptors are developed with the intent to improve efficiency for real-time applications. The selection of image descriptors becomes an important and interesting topic in most image matching related tasks with the consideration in distinctiveness, which can be defined in terms of the two types of perceptual aliasing. As far as we know, the existing performance metrics for image descriptors are either based on precision and recall and their generalized versions or in the context of a specific application, e.g., image retrieval, which may not be appropriate to measure perceptual aliasing because their lack of ability in measuring the discriminating power of image descriptors on distinguishing images of different objects and matching images of the same object. This motivates us to design a novel performance metric to measure the two types of perceptual aliasing in a proper way, where spectral clustering is run on the similarity matrix computed from image descriptors of known image clusters and two indices (MRI-1 and MRI-2) which are based on the Rand index are defined to measure the performance of an image descriptor by its ability to obtain clusters in descriptor space with the same clusters distribution in

image space. Further details of MRI-1 and MRI-2, together with our proposed procedure to evaluate the performance of image descriptors is presented in Chapter 3 and Chapter 4.

Chapter 3

Indices for Perceptual Aliasing

The previous chapter introduced the common local and global image descriptors and their performance metrics. The existing performance metrics may not be appropriate to evaluate perceptual aliasing problem since they are designed to measure different aspects of image descriptors. To design a measure of perceptual aliasing in image descriptors, we borrow the technique in clustering analysis of Rand index to define the indices of perceptual aliasing. In this chapter, we present the detailed information of the modified Rand index, MRI-1 and MRI-2.

3.1 Introduction

Image description is a process of transforming an image from intensity space to descriptor space, and perceptual aliasing happens when dissimilar images are mapped to similar descriptors, or when images of the same object or place under various image condition changes such as view point, scale, or illumination changes are mapped to different descriptors. We can quantify perceptual aliasing by clustering analysis and computing the ratio of image pairs belonging to different objects falling into the same image cluster and the ratio of image pairs belonging to the same object dropping into different image clusters. We define the modified Rand indices (MRI-1 and MRI-2) for the two types of perceptual aliasing based on the definition [47].

3.2 Perceptual Aliasing Indices

In Section 3.2.1, we introduce the Rand index. We then define our own indices to measure perceptual aliasing in image descriptors, which will be presented in greater detail in Section 3.2.2.

3.2.1 Rand Index

Rand index [38] was originally introduced to measure the performance of a clustering algorithm. Specifically, given two partitions of a set S, $X = \{X_1, \ldots, X_r\}$ and $Y = \{Y_1, \ldots, Y_s\}$, Rand index is defined as,

$$R = \frac{a+b}{a+b+c+d} \tag{3.1}$$

where a refers to the number of element pairs in S that are in the same subset of X and in the same subset of Y, b refers to the number of element pairs in S that are in different subsets of X and in different subsets of Y, c indicates the number of element pairs in S that are in the same subset of X but in different subsets of Y, d indicates the number of element pairs in S that are in different subsets of X but in the same subset of Y. Intuitively, $a + b$ denotes the number of agreements between X and Y; $c + d$ denotes the number of disagreements between X and Y; $a + b + c + d$ denotes the total number of element pairs in set S.

We can also create a contingency table to represent the overlapping between these two partitions $X = \{X_1, \ldots, X_r\}$ and $Y = \{Y_1, \ldots, Y_s\}$. The table is shown in Table 3.1.

Table 3.1: Contigency table of X and Y.

X/Y	Y_1	Y_2	$\cdots$	Y_s	Sums
X_1	n_{11}	n_{12}	$\cdots$	n_{1s}	a_1
X_2	n_{21}	n_{22}	$\cdots$	n_{2s}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
X_r	n_{r1}	n_{r2}	$\cdots$	n_{rs}	a_r
Sums	b_1	b_2	$\cdots$	b_s	

where n_{ij}, a_i and b_j are defined by the following equations. $|\cdot|$ indicates the cardinality of a set.

$$n_{ij} = |X_i \cap Y_j| \tag{3.2}$$

$$a_i = |X_i|, \quad i = 1, 2, \ldots, r \tag{3.3}$$

$$b_j = |Y_j|, \quad j = 1, 2, \ldots, s \tag{3.4}$$

Based on the contingency table, Rand index can be further defined as,

$$R = \frac{\mathbf{C}(n,2) - \sum_{i=1}^{r} \mathbf{C}(a_i,2) - \sum_{j=1}^{s} \mathbf{C}(b_j,2) + 2 \cdot \sum \mathbf{C}(n_{ij},2)}{\mathbf{C}(n,2)} \tag{3.5}$$

where $n = \sum n_{ij}$ and $\mathbf{C}(n,2)$ is the number of 2-combinations from a set of n elements.

The range of Rand index is $[0, 1]$ with lower value indicating less similarity between two partitions. However, it suffers from the problem that the Rand index value for randomly generated distribution of a set seldom equal to zero. Random distribution has no purpose to reproduce the original distribution of a set, therefore hardly sharing the distribution information and being less similar to the original distribution. The adjusted Rand index [38, 15, 44] overcomes this problem with the Rand index by using the expected Rand index of a random distribution in the similarity measure and offers a corrected-for-chance version of the Rand index. Based on the contingency table, the adjusted Rand index (ARI) is then defined as,

$$
\begin{aligned}
ARI &= \frac{Index - ExpectedIndex}{MaxIndex - ExpectedIndex} \\
&= \frac{\sum_{ij} \mathbf{C}(n_{ij}, 2) - [\sum_i \mathbf{C}(a_i, 2) \sum_j \mathbf{C}(b_j, 2)]/\mathbf{C}(n, 2)}{\frac{1}{2}[\sum_i \mathbf{C}(a_i, 2) + \sum_j \mathbf{C}(b_j, 2)] - [\sum_i \mathbf{C}(a_i, 2) \sum_j \mathbf{C}(b_j, 2)]/\mathbf{C}(n, 2)}
\end{aligned} \tag{3.6}
$$

The adjusted Rand index ranges from -1 to 1. The larger the index, the higher the similarity between two partitions.

3.2.2 MRI-1 and MRI-2

In our case, we have two partitions of a set S, i.e. the dataset, from image space and descriptor space respectively. One partition consists of image clusters of different world states (objects), i.e., images of the same world state belong to one cluster. On the other hand, in the corresponding image descriptor space, we can obtain another partition according to the similarity between descriptors, i.e., similar descriptors make up of one cluster. We borrow the toy example in Figure 2.9 to illustrate the idea behind the definition of the indices for the two types of perceptual aliasing.

In the original image space, the image cluster structure is shown in Figure 3.1, with which a ground truth partition Y has been obtained accordingly in Figure 3.2.

In the image descriptor space, another cluster structure can be achieved based on the similarity between image descriptor pairs. An example set of clusters has been illustrated in Figure 3.3. Correspondingly, we can then get another partition X in the descriptor space as shown in Figure 3.4.

To minimize perceptual aliasing, we expect high similarity between these two partitions X and Y, which means the image descriptor with strong discriminating power is able to maintain the similar and dissimilar relations between each image pair in descriptor space. Fortunately, the property of Rand index gives us the opportunity to define indices for the two types of perceptual aliasing so that the performance of image descriptors can be compared based on the quantified perceptual aliasing values.

MRI-1: The first type of perceptual aliasing occurs when multiple world states share one internal representation. We define the modified Rand index (MRI) for the first type of perceptual aliasing as,

$$
\text{MRI-1} = \frac{1}{r} \sum_{i=1}^{r} \frac{d_i}{\mathbf{C}(|X_i|, 2)} \tag{3.7}
$$

where $\mathbf{C}(|X_i|, 2)$ indicates the number of 2-combinations of the set X_i, $|X_i|$ denotes the cardinality of the set X_i and d_i is the number of image pairs that are in different subsets of Y but in the same subset of X. MRI-1 takes the average over the ratios computed on each subset $X_i \in X$ in the range [0, 1]. In our case, we do not have explicit weights for $X_i \in X$, therefore simply taking the average number. For specific applications, max or weighted average number can also be used. The smaller the MRI-1 value of an image descriptor, the superior the ability of this descriptor against

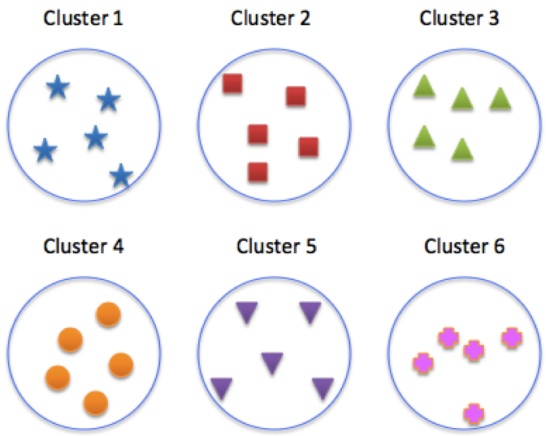


Figure 3.1: Toy example: The image clusters in the original image space.

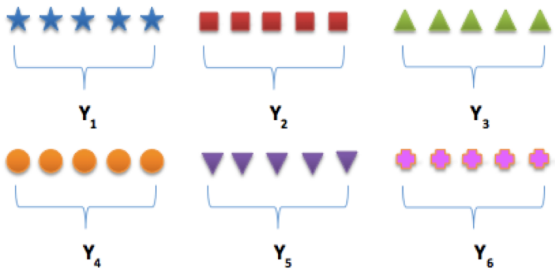


Figure 3.2: Toy example: Ground truth partition $Y = \{Y_1, Y_2, Y_3, Y_4, Y_5, Y_6\}$ of the dataset.

the first type of perceptual aliasing. The MRI-1 value of the example in Figure 3.4 is computed in Equation 3.8.

$$\text{MRI-1} = \frac{1}{6}\left(\frac{6}{\mathbf{C}(5,2)} + \frac{7}{\mathbf{C}(5,2)} + \frac{8}{\mathbf{C}(5,2)} + \frac{0}{\mathbf{C}(5,2)} + \frac{6}{\mathbf{C}(5,2)} + \frac{6}{\mathbf{C}(5,2)}\right) = 0.55 \tag{3.8}$$

MRI-2: The second type of perceptual aliasing occurs when images of the same object are mapped to dissimilar image descriptors, i.e., images that belong to Y_j are clustered into different subsets of X. Figure 3.5 shows the mapping between world states and internal representation in Figure 3.4. MRI-2 is defined to characterize this type of perceptual aliasing as,

$$\text{MRI-2} = \frac{1}{s}\sum_{j=1}^{s}\frac{c_j}{\mathbf{C}(|Y_j|,2)} \tag{3.9}$$

where c_j is the number of image pairs that are in the same subset of Y but different subsets of X. Similar to MRI-1, MRI-2 then takes the average over the ratios computed on each subset $Y_j \in Y$ and lies in the range [0, 1]. Similarly, max or weighted average number can be computed with different purposes in specific applications. The smaller the MRI-2, the less severe the second type of perceptual aliasing in the image descriptor. Equation 3.10 shows the MRI-2 value of the example

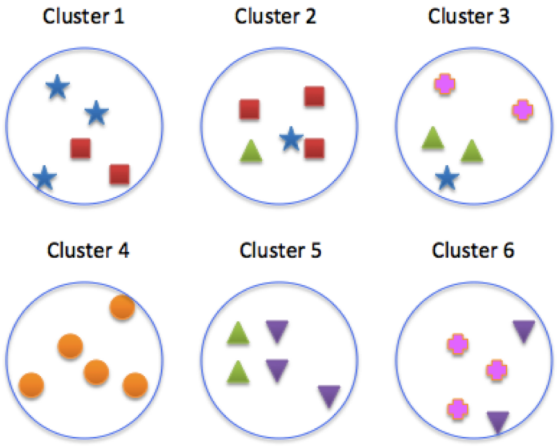


Figure 3.3: Toy example: An example set of clusters in the image descriptor space. This set of clusters is achieved by the similarity between pairs of image descriptors.

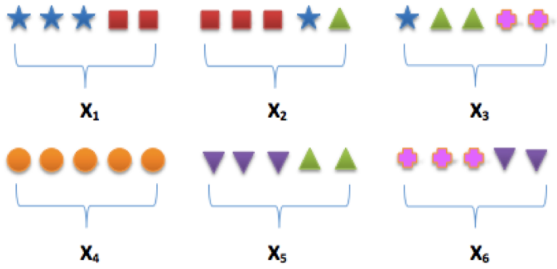


Figure 3.4: Toy example: Example partition $X = \{X_1, X_2, X_3, X_4, X_5, X_6\}$ in image descriptor space.

in Figure 3.5.

$$\text{MRI-2} = \frac{1}{6}\left(\frac{7}{\mathbf{C}(5,2)} + \frac{6}{\mathbf{C}(5,2)} + \frac{8}{\mathbf{C}(5,2)} + \frac{0}{\mathbf{C}(5,2)} + \frac{6}{\mathbf{C}(5,2)} + \frac{6}{\mathbf{C}(5,2)}\right) = 0.55 \quad (3.10)$$

3.2.3 Discussion

The modified Rand index (MRI) is originated from the definition of the two types of perceptual aliasing. Intuitively, perceptual aliasing can be measured in terms of image descriptor's ability to minimize the ratio of intraclass and interclass variances of clusters of images whose distance/similarities are determined by the image descriptor. There are many indices to measure clustering quality, such as Calinski and Harabasz index, Je(2)/Je(1) index and C-index [32]. However, we cannot quantify the two types of perceptual aliasing by borrowing these indices directly. These indices are exclusively based on element count and ignore the relation between all image pairs. In addition, we cannot use one index value to measure two types of perceptual aliasing. We define two separate indices, MRI-1 and MRI-2, based on the Rand index, with MRI-1 quantifying perceptual aliasing

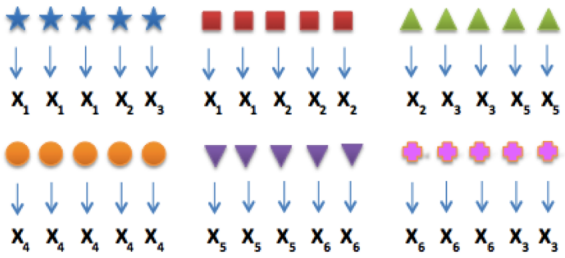
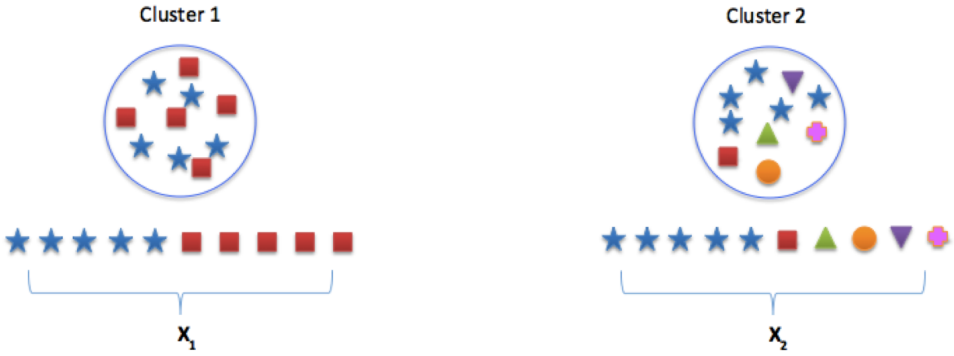


Figure 3.5: Toy example: Mapping relations from Figure 3.4



(a) (b)

Figure 3.6: Mapping relation corresponding to Figure 2.10. (a) Mapping relation for case 1. (b) Mapping relation for case 2.

between image clusters and MRI-2 quantifying perceptual aliasing within an image cluster. Unlike existing performance metrics which mainly focus on the similarity between query image and the top retrieved ones, our indices evaluate the similarity and dissimilarity between all image pairs where perceptual aliasing can occur. In Section 2.3.1, we discussed the insensitivity of precision and recall in measuring perceptual aliasing. We will evaluate perceptual aliasing using our proposed indices based on the same example in Section 2.3.1 to demonstrate the utility of our indices.

Figure 3.6 shows the corresponding mapping relations from Figure 2.10, and Table 3.2 illustrates the precision and MRI-1 values computed for the two cases. Precision measures how many retrieved images are correct, therefore ignoring the similarity between other image pairs. The precision values for the two cases both are $\frac{1}{2}$ which show the insensitivity of precision in the first type of perceptual aliasing. Our MRI-1 is able to capture the different levels of perceptual aliasing and provide detailed information about the problem.

To explain the difference between recall and MRI-2, we borrow the toy example in Figure 2.11. Recall is computed by setting the recall levels with different thresholds on similarity values or by picking the top n retrieved results, which focuses on only the similarity between the query and the top retrieved ones. For case 1 in Figure 2.11, two different example sets of clusters can be found in descriptor space as shown in Figure 3.7 and Figure 3.8. Although the recall of case 1 is 1 which

Table 3.2: Precision and MRI-1 values for Figure 3.6.

	precision	MRI-1
minimum	0	0
maximum	1	1
value for Figure 3.6(a)	$\frac{1}{2}$	$\frac{25}{\mathbf{C}(10,2)} = \frac{5}{9}$
value for Figure 3.6(b)	$\frac{1}{2}$	$\frac{35}{\mathbf{C}(10,2)} = \frac{7}{9}$

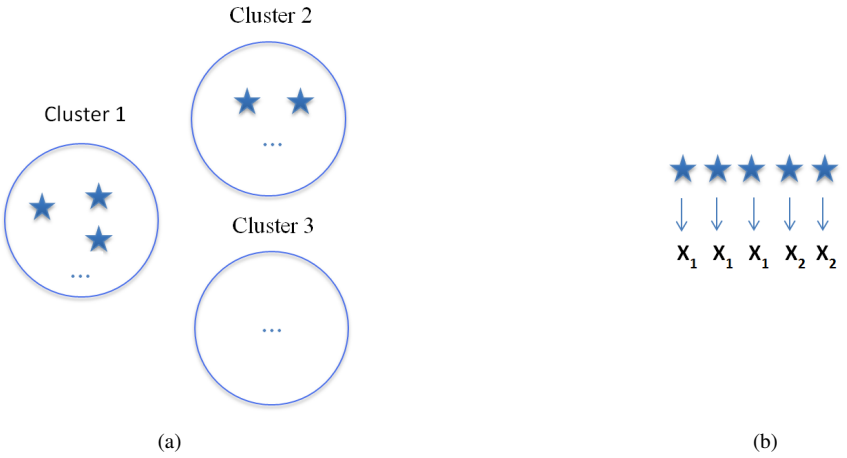


Figure 3.7: One set of clusters corresponding to Figure 2.11. (a) Five images drop into two clusters. (b) Mapping relations.

means all the related images have been retrieved, the potential relations among them can be very different. One possible set of clusters show that the five "stars" drop into two clusters in descriptor space. Another possible set of clusters shows these images belong to three clusters instead. The recall cannot capture the difference and is insensitive to the second type of perceptual aliasing since its computation does not take other image pairs into account. The property of clustering gives us the opportunity to explore the relations between all image pairs, therefore leading to different MRI-2 values as shown in Table 3.3.

Table 3.3: Recall and MRI-2 values for Figure 3.7 and Figure 3.8.

	recall	MRI-2
minimum	0	0
maximum	1	1
value for Figure 3.7	1	$\frac{6}{\mathbf{C}(5,2)} = \frac{3}{5}$
value for Figure 3.8	1	$\frac{7}{\mathbf{C}(5,2)} = \frac{7}{10}$

Different performance metrics aim at evaluating different aspects of image descriptors. The insensitivity of precision and recall in measuring perceptual aliasing motivates us to design novel indices for the two types of perceptual aliasing, i.e., MRI-1 and MRI-2. Our indices are computed based on the clusters in the original image space and the corresponding image descriptor space, making it possible to consider relations between all image pairs and giving accurate measurement

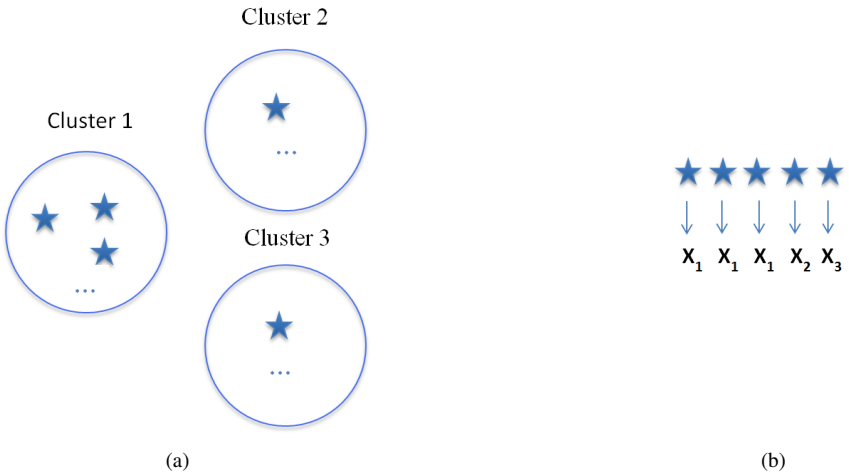


Figure 3.8: Another set of clusters corresponding to Figure 2.11. (a) Five images drop into three clusters. (b) Mapping relations.

of perceptual aliasing.

3.3 Summary

In this chapter, indices for the two types of perceptual aliasing are presented. The indices rely on the similarity between each pair of images to characterize different levels of perceptual aliasing in image matching tasks. Simple examples have been given and discussed to illustrate the different consideration of existing performance metrics, e.g. precision and recall, and our proposed indices. Our indices are designed according to the definition of perceptual aliasing. As the examples show, the measurement results from our indices are more sensitive to the two types of perceptual aliasing. A method using MRI-1 and MRI-2 to evaluate the performance of image descriptors will be presented in the next chapter.

Chapter 4

Perceptual Aliasing Measurement

In the previous chapter we introduced the indices for the two types of perceptual aliasing, i.e., MRI-1 and MRI-2 and discussed the usefulness in measuring the problem by comparing the results with precision and recall. Our indices can provide accurate quantification of perceptual aliasing based on the clusters in the original image space and the corresponding image descriptor space. To evaluate the image descriptor performance, we introduce a perceptual aliasing measurement using MRI-1 and MRI-2 in detail in this chapter. The measurement can be applied to different applications and the results can be taken as a reference to select an appropriate image descriptor.

4.1 Introduction

In image matching related tasks, image descriptors are usually extracted to characterize images. The process of generating image descriptors is presented in Figure 4.1. Local keypoint descriptor is used to characterize the property of an image patch which is detected by a keypoint detector, e.g. Harris corner detector. On the other hand, whole image descriptor describes the property of a whole image, which is more compact and computationally efficient. The performance of either local keypoint descriptors or whole image descriptors can be evaluated by following the proposed procedures in the next sections.

To investigate the extent of perceptual aliasing caused by a particular image descriptor, we can construct a similarity matrix with respect to many clusters of images such that images within a cluster are similar and images between clusters are dissimilar. With a similarity matrix, the performance of image descriptors can be evaluated even when the descriptors are not explicitly available or of no fixed length. The clusters can be collected easily by grouping consecutive images of a video into clusters – assuming the camera captures the images continuously. The similarity matrix can be computed by matching local feature/keypoint descriptors like SIFT [27] and SURF [2] or global image descriptors like GIST [35] and HOG [9]. Similarly, similarity matrix can also be constructed by computing the similarity between each pair of detected image patches to evaluate the performance of local keypoint descriptors only in regard to different image condition changes like scale, view

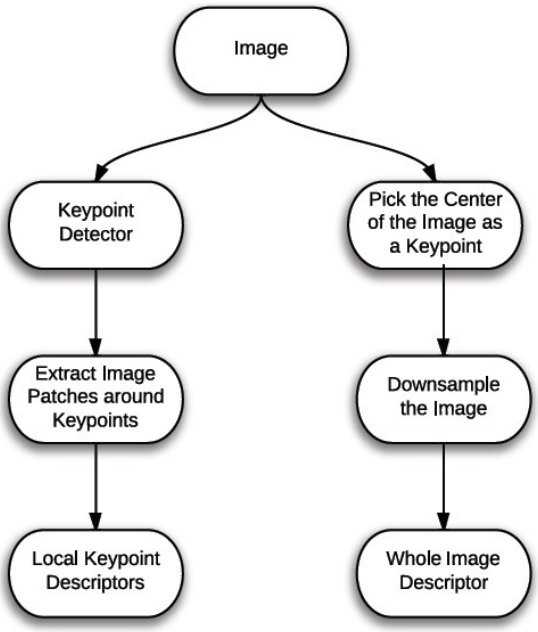


Figure 4.1: Generation of local keypoint descriptors or whole image descriptors.

point, illumination, rotation, etc. In the following sections, we will first introduce the construction of a similarity matrix in detail and then explain the clusters exploration in both image space and the corresponding image descriptor space. Finally, we will make use of the proposed indices to measure the two types of perceptual aliasing in image descriptors.

4.2 Similarity Matrix Construction

Each element s_{ij} in similarity matrix S indicates the similarity between two data points. To construct a similarity matrix, we collect images to create a dataset such as those shown in Figure 4.2 that includes street views of the city of Pittsburgh. We include c adjacent images of the video in a cluster and select m clusters in total. Images in the same cluster contain similar scenes and images in different clusters do not overlap by requiring each cluster every $n \gg c$ images.

The similarity can be computed by matching the local feature/keypoint descriptors or calculating the distance between two global image descriptors if the image descriptors are explicit and of the same length such as GIST. As a result, the similarity matrix S is of size $mc \times mc$. An example similarity matrix, computed from the GIST image descriptors with 20 clusters and 5 images per cluster, is plotted in Figure 4.3. Since images of the same cluster contain similar scenes, elements near the diagonal form $c \times c$ bright blocks indicating much higher similarity within each block than between blocks. Due to perceptual aliasing, however, many off-diagonal elements of the similarity matrix have high values. This common weakness is shared by all image descriptors to various extents.

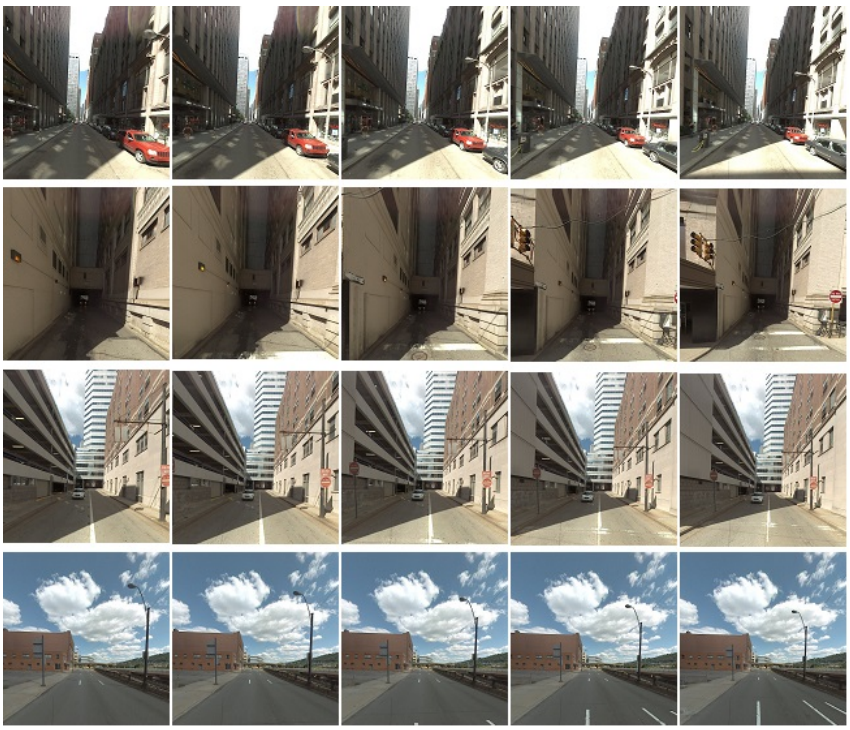


Figure 4.2: Four image clusters from the dataset. Each cluster contains five adjacent images taken from the back perspective of a camera car. Images in the same cluster contain the same scene and there is no overlapping between two clusters.

In addition, to evaluate the performance of local keypoint descriptors under different image condition changes, we can also construct a similarity matrix based on the similarity between pairs of local keypoint patches. With respect to different changes in image conditions, such as scale, view point, illumination, rotation, etc., sets of images containing the same scene but under a specific image condition change to different degrees are collected as shown in Figure 2.7. Each set has six images under one type of image condition change. For a specific image condition change, e.g. scale change, keypoints are first detected from one image in the set by using a keypoint detector, e.g., Harris corner detector. The corresponding keypoints in the other images of the set are found by using RANSAC-based robust matching. Clusters of keypoint patches are then extracted accordingly with six keypoint patches per cluster. A similarity matrix is computed on clusters of keypoint patches. One example has been shown in Figure 4.4.

4.3 Spectral Clustering

Spectral clustering [28, 42] is widely used in image processing, computer vision, computational biology, statistical data analysis and machine learning. It works on the spectrum of the similarity matrix and is superior in solving general clustering problems even when the clusters are not convex sets. With the similarity matrix computed on image descriptors, we are then able to group images

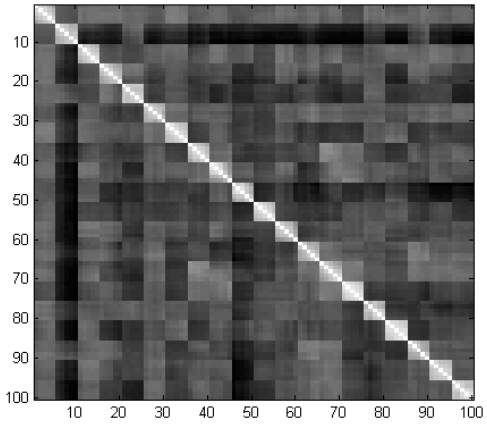


Figure 4.3: Similarity matrix computed with GIST descriptors. $m = 20$ and $c = 5$. Similarity matrix is of size $mc \times mc = 100 \times 100$.

according to their descriptors using spectral clustering and perform an analysis of perceptual aliasing based on the clustering results. Since the ground truth clusters in the original image space are known in our dataset, we can measure the performance of an image descriptor by examining how well the clusters have been retained in image descriptor space.

The advantage of spectral clustering comes from its graph-partitioning-based clustering without any assumption on the structure of the clusters, and its usage of eigenvectors derived from the data, i.e., the data mapped to a low-dimensional space that is more separable. Specifically, given data points $x_1, \ldots, x_N$ and the pairwise similarity (or affinity) $s_{ij} = s(x_i, x_j)$ where similarity can encode the local structure in the data, a graph G can be constructed which consists of a vertex set $V(G)$ corresponding to the data points and an edge set $E(G)$ related to the similarity (or affinity) between data points as shown in Figure 4.5. In spectral clustering, we always use undirected graph with the edges representing the similarity between data points. The object of spectral clustering is to find a cut through the graph. Graph is an important component of spectral clustering since many datasets have natural graph structure, such as protein structures, citation graphs and webpage links. In our case, with the help of spectral clustering, we are able to construct a graph taking similarity between each pair of images into account to measure the two types of perceptual aliasing without assuming the form of the clusters.

In addition, a low-dimensional embedding can be found to project the data points to a new space as shown in Figure 4.6. We are then able to cluster the data points using a clustering algorithm. Particularly, we need first compute the adjacency matrix W of the constructed undirected graph G. The adjacency matrix W is an $N \times N$ symmetric binary matrix with rows and columns representing the vertices and entries indicating the edges of the graph G. Second, an affinity matrix A, the weighted adjacency matrix, is created whose edges are weighted by pairwise vertex affinity (or similarity). Affinity matrix in our case is also called similarity matrix. Third, we also need to define the degree

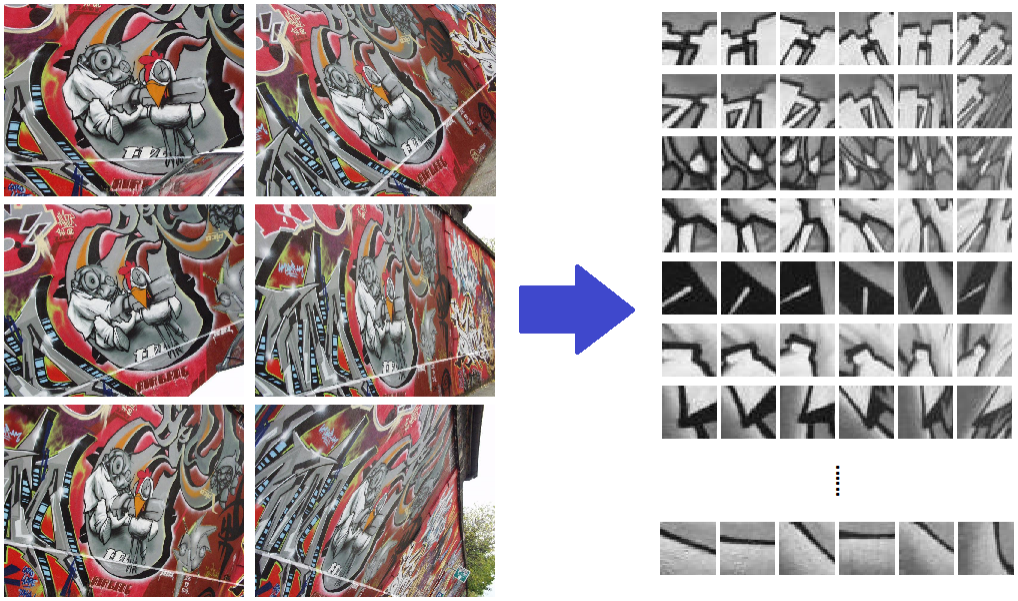


Figure 4.4: A set of graffiti images of different view point changes. Rows of keypoint patches are extracted. Keypoints are detected using Harris corner detector from the first image (top left one). The corresponding keypoint patches are extracted from the other 5 images using the Homography matrices H1to2, H1to3, . . ., H1to6 indicating the transformation from the first image to the corresponding image. Each row contains 6 keypoint patches from the 6 images.

matrix D of the undirected graph, which is an $N \times N$ diagonal matrix containing the degrees of vertices in graph G. Based on the above matrices, we can finally compute a Laplacian matrix L of the simple undirected graph, and find its eigenvectors to do the clustering. There are many strategies to calculate a Laplacian matrix, while L_{rw} achieves better clustering performance [28], which is computed in Equation 4.2.

$$L = D - A \tag{4.1}$$

$$L_{rw} = D^{-1}L = I - D^{-1}A \tag{4.2}$$

where I is the identity matrix of size $N \times N$. 0 is the smallest eigenvalue of L_{rw} with the constant one vector $\mathbb{1}$. L_{rw} is positive semi-definite with n non-negative real-valued eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$. The eigenvectors $u_1, \ldots, u_k$ corresponding to the k smallest eigenvalues are taken to construct $U \in \mathbb{R}^{n \times k}$. K-means is then used to do clustering on U row-wisely to find the k clusters. Intuitively, the original data points have been mapped to a new low-dimensional space, i.e., k-dimensional space, where the data is more separable.

In the graph theoretic point of view, spectral clustering is solving a graph cut problem with the affinity matrix A. The simplest solution to this problem is to find a min-cut through the graph. However, clustering results may not be reasonable when the connected components are not balanced as shown in Figure 4.7, which can deteriorate the performance of image descriptors in our case. It

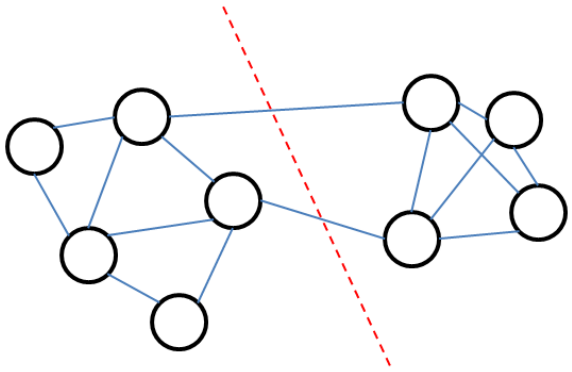
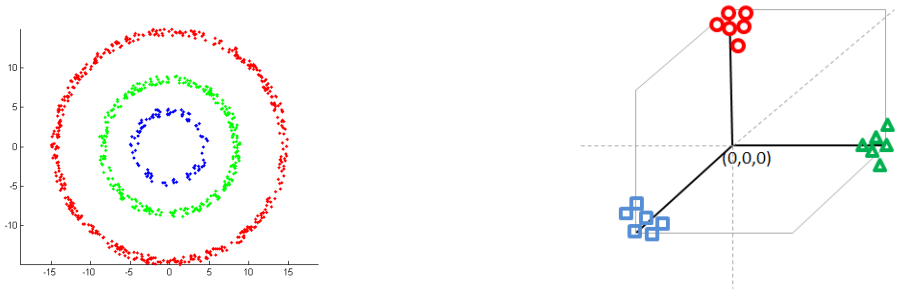


Figure 4.5: Spectral clustering: graph theoretic view.



(a) Original data (b) Projected data

Figure 4.6: Spectral clustering: low dimension embedding view.

is necessary to consider the size of each cluster to generate more balanced results. In general, there are three ways to balance the clustering results: ratio-cut, normalized cut and min-max cut. Each is defined as,

$$\text{Ratiocut}(Z_1, \ldots, Z_k) = \frac{1}{2}\sum_{i=1}^{k}\frac{A(Z_i, \bar{Z}_i)}{|Z_i|} = \sum_{i=1}^{k}\frac{\text{cut}(Z_i, \bar{Z}_i)}{|Z_i|} \tag{4.3}$$

$$\text{NCut}(Z_1, \ldots, Z_k) = \frac{1}{2}\sum_{i=1}^{k}\frac{A(Z_i, \bar{Z}_i)}{\text{vol}(Z_i)} = \sum_{i=1}^{k}\frac{\text{cut}(Z_i, \bar{Z}_i)}{\text{vol}(Z_i)} \tag{4.4}$$

$$\text{MinMaxCut}(Z_1, \ldots, Z_k) = \frac{1}{2}\sum_{i=1}^{k}\frac{A(Z_i, \bar{Z}_i)}{A(Z_i, Z_i)} = \sum_{i=1}^{k}\frac{\text{cut}(Z_i, \bar{Z}_i)}{A(Z_i, Z_i)} \tag{4.5}$$

where $|Z_i|$ indicates the number of vertices in the subset Z_i, $A(Z_i, \bar{Z}_i)$ is the minimum similarity between the subset Z_i and its complementary set $\bar{Z}_i$, $\text{cut}(Z_i, \bar{Z}_i) = \frac{1}{2}\sum_{i\in Z_i, j\in \bar{Z}_i}\text{Affinity}(Z_i, \bar{Z}_i)$, $\text{vol}(Z_i) = \sum_{i\in Z_i} D_i$, $A(Z_i, Z_i)$ indicates the maximum similarity between subset Z_i. The problem is NP-hard. Relaxing RatioCut leads to unnormalized spectral clustering. Specifically, we can rewrite the problem as,

$$\min \text{Ratiocut}(Z_1, \ldots, Z_k) = \min_{Z_1,\ldots,Z_k} \text{Tr}(U'LU) \text{ subject to } U'U = I \tag{4.6}$$

where $U \in \mathbb{R}^{n \times k}$ contains k indicator vectors as columns. The element u_{ij} is defined as,

$$u_{ij} = \begin{cases} \dfrac{1}{\sqrt{|Z_j|}}, \text{ if } v_i \in Z_j \\ 0, \text{ otherwise} \end{cases} \quad i = 1, \ldots, n; \; j = 1, \ldots, k \tag{4.7}$$

The problem can be relaxed by allowing the matrix U to take arbitrary real values,

$$\min_{U \in \mathbb{R}^{n \times k}} \operatorname{Tr}(U'LU) \text{ subject to } U'U = I \tag{4.8}$$

Optimal U is the first k eigenvectors of L as columns.

The relaxation of NCut and MinMaxCut leads to normalized spectral clustering which computes the normalized Laplacian L, and the eigenvectors are normalized before being clustered by Kmeans [28]. Similarly, we rewrite the problem as,

$$\min \text{NCut}(Z_1, \ldots, Z_k) = \min_{Z_1, \ldots, Z_k} \operatorname{Tr}(U'LU) \text{ subject to } U'U = I \tag{4.9}$$

In this case, the element u_{ij} is defined as,

$$u_{ij} = \begin{cases} \dfrac{1}{\sqrt{\text{vol}(Z_j)}}, \text{ if } v_i \in Z_j \\ 0, \text{ otherwise} \end{cases} \quad i = 1, \ldots, n; \; j = 1, \ldots, k \tag{4.10}$$

Similar to above, we relax the problem by allowing the matrix T to take arbitrary real values,

$$\min_{T \in \mathbb{R}^{n \times k}} \operatorname{Tr}(T'D^{-1/2}LD^{-1/2}T) \text{ subject to } T'T = I \tag{4.11}$$

$$U = D^{-1/2}T \tag{4.12}$$

Solution U contains the first k eigenvectors of L_{rw}. Relaxing MinMaxCut leads to exactly the same solution as relaxing NCut. Better clustering performance is achieved with normalized spectral clustering based on the eigenvectors of L_{rw} [28]. Our method makes use of L_{rw}. The time complexity of spectral clustering is $\mathcal{O}(n^3)$

With spectral clustering, we are able to evaluate the perceptual aliasing in image descriptors if similarity can be computed between each pair of images, even when the image descriptors are not available explicitly or not of fixed length. We set the number of image clusters to the ground truth number k as input to spectral clustering. We tried to find natural cluster number with clustering algorithms and no method exists to do it reliably. Besides, since we are interested in the relative performance, it is still fair to set the known cluster number when we measure perceptual aliasing for different image descriptors. To illustrate the usefulness of spectral clustering in our case, we compare the performance of Kmeans and spectral clustering with the Adjusted Rand index. The ground truth clusters are collected as discussed in Section 4.2 and the image descriptor we used is GIST descriptor. Figure 4.8(a) shows that spectral clustering can in general generate superior clustering

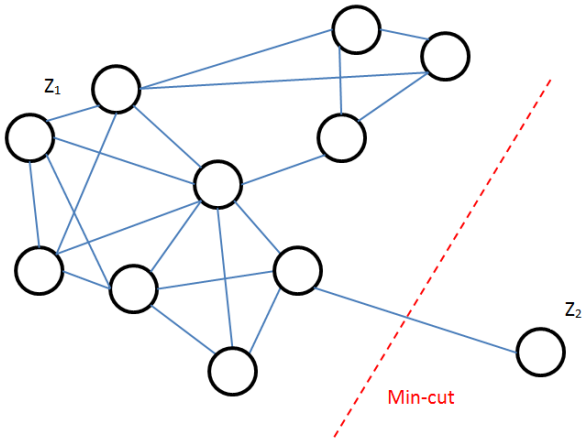


Figure 4.7: Unbalanced clusters generated by min-cut on the graph.

results in our case. Meanwhile, spectral clustering runs Kmeans row-wisely on $U \in \mathbb{R}^{n \times k}$ which is constructed by the first k eigenvectors of the Laplacian matrix corresponding to the first k smallest eigenvalues. The clustering on k dimensional data can achieve more efficiency than traditional clustering algorithms, e.g. Kmeans, on the original high dimensional data as shown in Figure 4.8(b). Since the data in our case is of relatively small scale – 100 image clusters with five images per cluster and the descriptors are around tens to hundreds of dimensions, the efficiency of clustering algorithm is not that important. We obtain the ground truth partition/clusters $Y = \{Y_1, \ldots, Y_s\}$ in image space. With the help of spectral clustering, we then get another partition/clusters $X = \{X_1, \ldots, X_r\}$ in descriptor space. Based on these two partitions, we can make use of the proposed indices to analyze the two types of perceptual aliasing.

4.4 Computing Perceptual Aliasing Indices

Indices for the two types of perceptual aliasing (MRI-1 and MRI-2) are then computed for the two sets of clusters $X = \{X_1, \ldots, X_r\}$ and $Y = \{Y_1, \ldots, Y_s\}$. The overall process is summarized in Algorithm 1. The number of image clusters k is set from 2 to K, where $K \leq m$, so that we are able to observe the trend of perceptual aliasing with more image clusters. For each k, the corresponding similarity matrix $S^{kc \times kc}$ is extracted from the input similarity matrix $S^{mc \times mc}$. C_x and C_y are vectors of the result cluster labels and the ground truth cluster labels of the images. PA is the method to compute MRI-1 and MRI-2 according to Equations (3.7, 3.9) given the cluster labels. The experimental steps are then repeated with each k for max_iter times since perceptual aliasing between different image clusters can be different and the final results are the average over the results of all iterations.

To compare the performance of different image descriptors within an application, image clusters containing images of the specific application can first be collected and similarity matrices on different image descriptors can then be computed. By following the steps, comparison results with

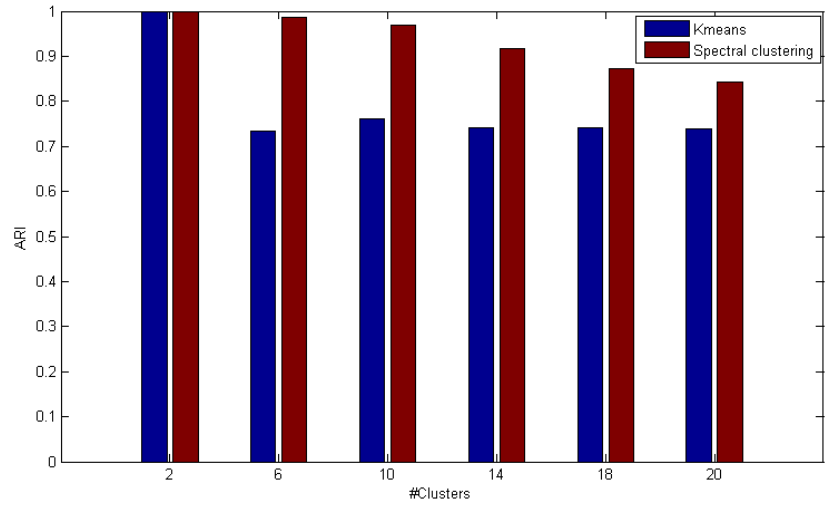
respect to perceptual aliasing are generated to help image descriptor selection.

Require: $S \leftarrow$ similarity_matrix, $K \leftarrow$ #image_cluseters, $t \leftarrow 1$
1: MRI-1$_table = zeros(max_iter,\ k)$;
2: MRI-2$_table = zeros(max_iter,\ k)$;
3: **for** $k = 2 \rightarrow K$ **do**
4: **while** $t \leq max\_iter$ **do**
5: $S^{kc \times kc} =$ sub similarity matrix corresponds to randomly selected k image clusters;
6: $C_Y = Spectral\_clustering(S^{kc \times kc}, k)$;
7: $[\text{MRI-1}, \text{MRI-2}] = PA(C_X, C_Y)$;
8: MRI-1$_table(t,\ k) =$ MRI-1;
9: MRI-2$_table(t,\ k) =$ MRI-2;
10: **end while**
11: **end for**
12: MRI-1$_result = mean(\text{MRI-1}_table, 1)$;
13: MRI-2$_result = mean(\text{MRI-2}_table, 1)$

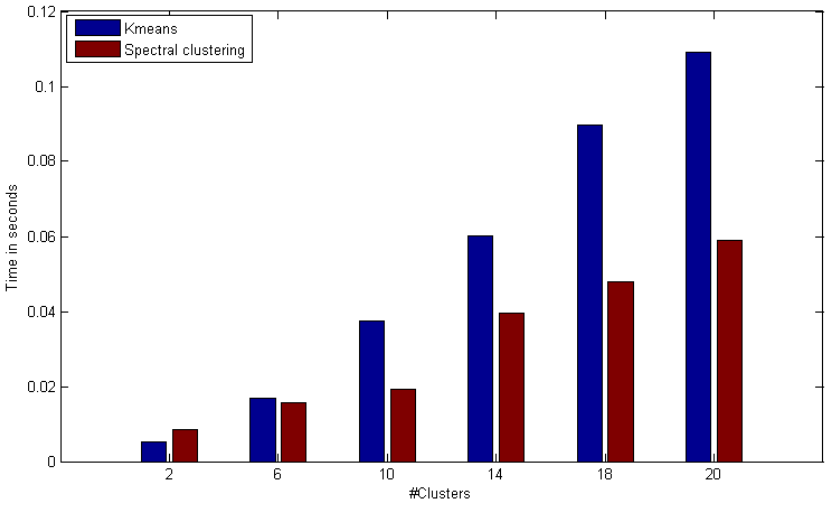
Algorithm 1: A method to measure perceptual aliasing

4.5 Summary

In this chapter, a procedure to measure perceptual aliasing in image descriptors based on the proposed indices MRI-1 and MRI-2 is introduced. By constructing a similarity matrix, we can evaluate the performance of image descriptors even when the descriptors are of no fixed length or not available explicitly. The similarity matrix can be created on either global image descriptors describing the characteristic of whole images or local feature descriptors capturing the properties of keypoint image patches. With the help of spectral clustering, we are then able to capture the clusters in descriptor space which corresponds to the original image space so that the mapping relations can be found between world states and internal representation. We briefly investigate the spectral clustering in terms of its graph-partitioning-based point of view and its low-dimensional embedding point of view. Normalized spectral clustering with Laplacian matrix L_{rw} produces the best clustering results and we make use of this version of spectral clustering in our case. In summary, spectral clustering uses the spectrum of the similarity matrix and generates superior clustering results without the assumption in the form of the data. After clustering, the indices for the two types of perceptual aliasing are computed for different numbers of image clusters to investigate the trend of the problem. To demonstrate the usefulness of our measure, experimental results on comparing different global image descriptors, as well as local feature descriptors, are provided in Chapter 5.



(a)



(b)

Figure 4.8: Comparison of Kmeans and spectral clustering. The clusters are collected as discussed in Section 4.2. (a) Adjusted Rand index on the clustering results. (b) Computational complexity of the two clustering algorithms.

Chapter 5

Experimental Results

In this chapter, we conduct four sets of experiments. The first set involves evaluation of different versions of the global BRIEF descriptor, i.e., BRIEF-Gist, to demonstrate the conclusion of our method is consistent with the literature. To illustrate the utility of our method in image descriptor comparison, we run the second set of experiments on two image datasets, the Pittsburgh Street View dataset and the Recognition Benchmark dataset[1], to examine image descriptors within place recognition and image retrieval applications. The first two sets of experiments are run to evaluate and compare the performance of the common global image descriptors in terms of MRI-1 and MRI-2 defined in the previous chapter. In the third set of experiments, we investigate the performance of local keypoint descriptors against perceptual aliasing with respect to different changes in image conditions, such as viewpoint, scale, rotation and illumination changes. In these experiments, keypoint patches are collected from the Affine Covariant Regions dataset[2] and local keypoint descriptors are created to characterize the keypoint patches. In the last set of experiments, we compare the evaluation results using our proposed method and precision and recall to further demonstrate the utility of our method in clustering-based image applications. The empirical results show the usefulness of our method in real applications.

5.1 Datasets

We use three datasets in our experiments. The first two datasets are used to evaluate the performance of global image descriptors in the first two sets of experiments. The third dataset containing images under different changes in image conditions is used to compare the performance of local keypoint descriptors. The different utility between our method and precision and recall on evaluation of perceptual aliasing is investigated using global image descriptors on the first dataset.

Pittsburgh Street View dataset: The dataset contains consecutive images taken from four perspectives, front, back, left and right, along the Pittsburgh streets. The images are taken by Google for the street view in Google Maps. The back and front views are more suitable for our work since

[1]Recognition Benchmark dataset: http://www.vis.uky.edu/ stewe/ukbench/
[2]Affine Covariant Regions dataset: http://www.robots.ox.ac.uk/˜vgg/data/data-aff.html

Figure 5.1: Four image clusters from Benchmark dataset. Each cluster contains four images of different changes in scale, rotation, illumination, blur, etc.

consecutive images share overlap so that we are able to collect image clusters. We pick the back views and select 100 image clusters with each cluster containing five consecutive images. To exclude overlap between two image clusters, each cluster is chosen by an interval of 100 images. We use $100 \times 5 = 500$ images. Figure 4.2 shows example image clusters selected from the Pittsburgh Street View dataset.

Recognition Benchmark dataset: This dataset is different from the Pittsburgh Street View dataset and contains image clusters of different objects, such as shoes, CD covers, books, clocks, etc. Most images are indoor scenes with various blur, illumination, view point and rotations changes, which can affect the performance of some image descriptors. We select 100 image clusters and four images per cluster from the dataset. Figure 5.1 illustrates the sample images from this dataset.

Affine Covariant Regions dataset: To evaluate the performance of keypoint detectors and keypoint descriptors, Mikolajczyk and Schmid collected sets of images under different changes in image conditions [31]. Each set has six images under different changes in image conditions as shown in Figure 2.7. For each set of images, homography matrices recoding the transformation information between the first image and the other images in the same set are also included. We use this dataset to collect keypoint patches and measure perceptual aliasing in local keypoint descriptors. With homography matrices, we are able to find corresponding keypoint patches among the six images in a set and conduct experiments to compare keypoint descriptors with respect to a specific type of image condition change.

In our experiments, we extract image descriptors to create a similarity matrix of size $mc \times mc$,

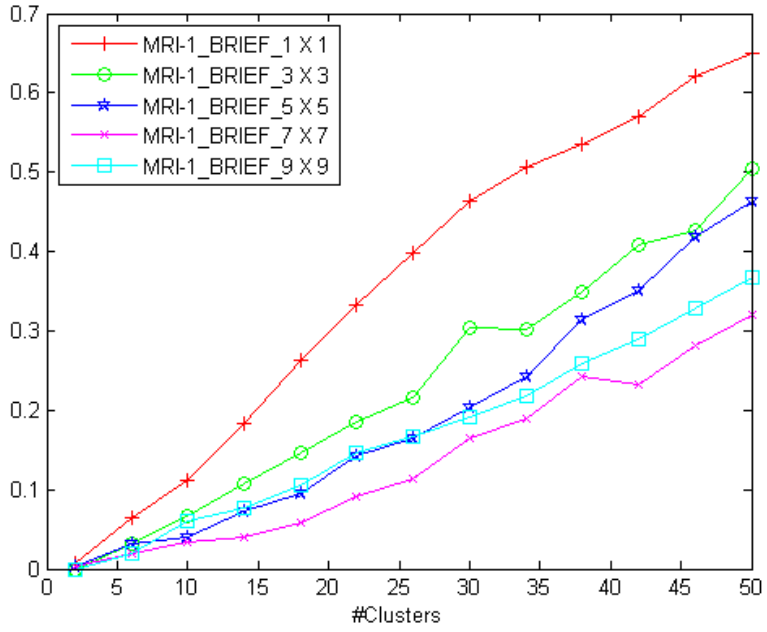


Figure 5.2: Comparison of the performance against perceptual aliasing of global BRIEF descriptors with MRI-1 on Pittsburgh Street View dataset.

where m is the number of image clusters, c is the number of images per cluster. Given the similarity matrix, we set the maximum number of image clusters K to 50 so that k is from 2 to 50. Practically, it is difficult to determine the range of k. However, since we are interested in measuring the relative performance against perceptual aliasing of different image descriptors, we can just set k from 2 to 50 and observe the comparison results. The maximum number of iterations max_iter has been set to 100 since perceptual aliasing can be different within different combinations of image clusters and the results can be averaged to show the overall performance.

5.2 Results on BRIEF-Gist

To examine our method, we ran the experiments on the Pittsburgh Street View dataset to evaluate perceptual aliasing using different versions of global BRIEF descriptors.

The BRIEF-Gist descriptor is proposed and applied to SLAM problem where the scenario is similar to that of the Pittsburgh Street View. [40] proves that BRIEF with 7×7 tiles achieves the better performance than a smaller number of tiles. With a larger number of tiles, the performance will also not improve. Our results of the relative performance of different versions of BRIEF are shown in Figure 5.2 and Figure 5.3 where one can observe that they are consistent with that reported in [40]. However, our method provides specific performance evaluation in regard to the two types of perceptual aliasing.

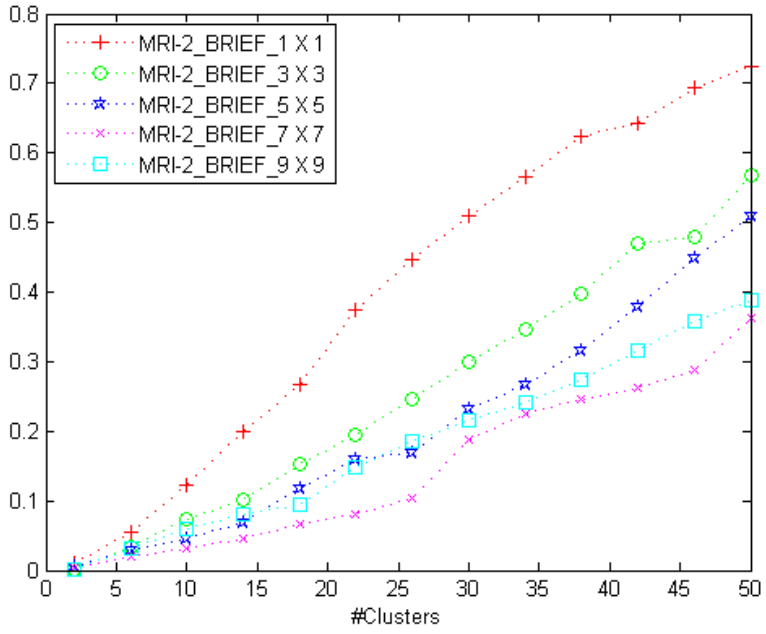


Figure 5.3: Comparison of the performance against perceptual aliasing of global BRIEF descriptors with MRI-2 on Pittsburgh Street View dataset.

5.3 Results on Different Datasets

Experiments of this section are conducted to illustrate the utility of our method in image descriptor comparison within different application scenarios in terms of perceptual aliasing.

We evaluate and compare the performance of the common global image descriptors that do not require keypoint detection, including: BRIEF-Gist, GIST, WI-SIFT and WI-SURF, on Pittsburgh Street View dataset and Recognition Benchmark dataset. We also ran experiments with the local SIFT descriptor, since it shows outstanding performance and serves as a baseline to observe the performance of global image descriptors intuitively and objectively. The similarity matrix for SIFT descriptors is created by matching the local feature/keypoint descriptors.

5.3.1 Results on Pittsburgh Dataset

Figure 5.4 and Figure 5.5 show the experimental results on Pittsburgh dataset. Local SIFT shows the best performance among the tested image descriptors, which is not unexpected since a local keypoint descriptor is capable of preserving more information in general. BRIEF-Gist was recently designed and applied in SLAM problem where the dataset is similar to the dataset we used in our experiments. There are different versions of BRIEF-Gist and BRIEF_7×7 is superior than the other versions [40]. Figure 5.4 and Figure 5.5 show that BRIEF_7×7 achieves better performance than the other global image descriptors. Following BRIEF_7×7 are WI-SIFT and WI-SURF, which are quite

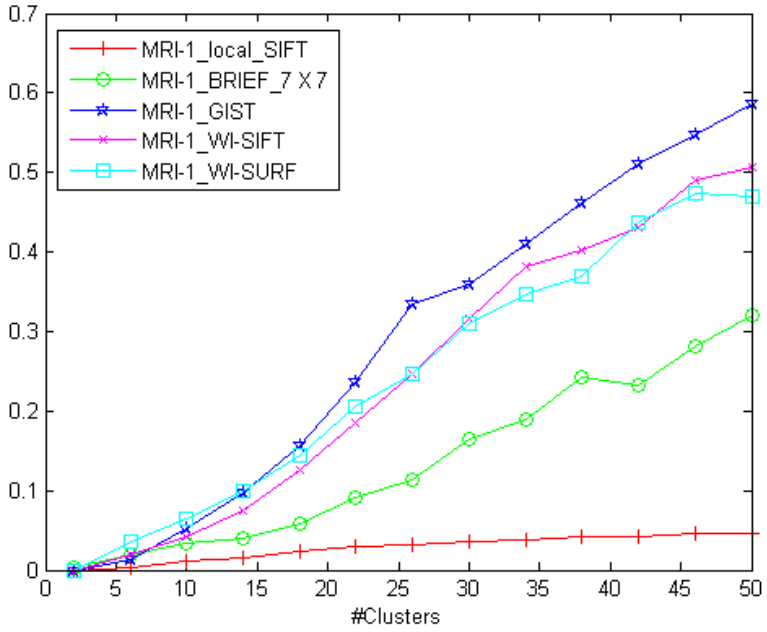


Figure 5.4: Comparison of the performance against perceptual aliasing of the common image descriptors with MRI-1 on Pittsburgh Street View dataset.

similar in performance in terms of perceptual aliasing, perhaps due to the similar way in which they describe an image with SURF aiming to speed up the computation. GIST was originally proposed to represent the global structure of the scene and was used to describe the naturalness of the image content, such as buildings, mountains, trees, etc. In this case, the Street View dataset contains images of an urban city environment, which a similar degree of naturalness that GIST captures. Therefore, it is difficult for GIST to distinguish and match images in this dataset.

5.3.2 Results on Benchmark Dataset

Figure 5.6 and Figure 5.7 show the experimental results on Benchmark dataset that contains image clusters of different objects, such as shoes, CD covers, clocks, etc., which are different from the Pittsburgh Street View dataset. The local feature descriptor again achieves the best performance as expected. We notice that the MRI-1 curve of local SIFT goes down with larger number of image clusters. Figure 5.8 shows the standard deviation of MRI-1 values of three selected image descriptors in Figure 5.6. The variance of MRI-1 values converges with larger number of image clusters. For local SIFT, the curve is not guaranteed to go down since the variance on smaller number of image clusters is larger than that on larger number of image clusters and some index values are smaller than those of larger number of image clusters. The first type of perceptual aliasing might be serious between some randomly selected image clusters with description of local SIFT. Among the global image descriptors, WI-SURF gains the best performance, followed by WI-SIFT, GIST and

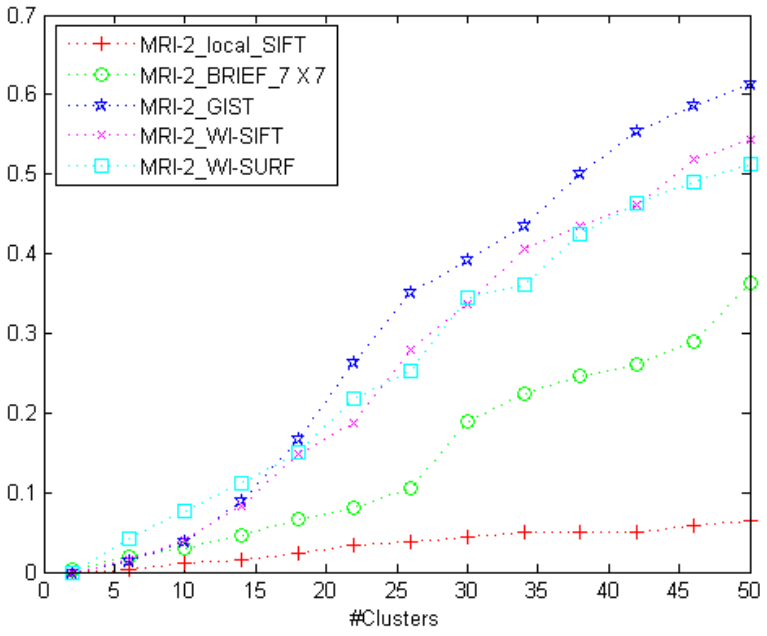


Figure 5.5: Comparison of the performance against perceptual aliasing of the common image descriptors with MRI-2 on Pittsburgh Street View dataset.

BRIEF_7 $\times$ 7. It is interesting to observe that the rank of image descriptors is different from that on the Street View dataset, and there is a simple explanation. Since this dataset contains many indoor images with varying amounts of blur and under different lighting conditions, WI-SURF is expected to be robust to these variations. BRIEF_7 $\times$ 7 however is not invariant to rotation changes and unable to handle many of the images in the dataset that experience rotational changes. GIST also seems have difficulty with this dataset. It performs better than BRIEF_7 $\times$ 7 in MRI-1 but worse with large number of image clusters in MRI-2.

The results in this section show that our performance indices, MRI-1 and MRI-2, can be used to select image descriptors in different applications. For example, in SLAM, the robot will easily lose itself if images of multiple locations look similar to each other in the descriptor space, i.e., if MRI-1 is very serious with an image descriptor, the localization mission will fail. Better ability of an image descriptor against MRI-1 is preferred in this situation. On the other hand, in image retrieval, it is important for the matching images to be at or near the top in the ranked retrieved results, i.e., the robustness against the second type of perceptual aliasing should instead be favoured.

5.4 Results on Local Keypoint Descriptors

In this section, we compare local keypoint descriptors by measuring perceptual aliasing under different changes in image conditions, including: blur, viewpoint, scale, rotation, illumination and JPEG

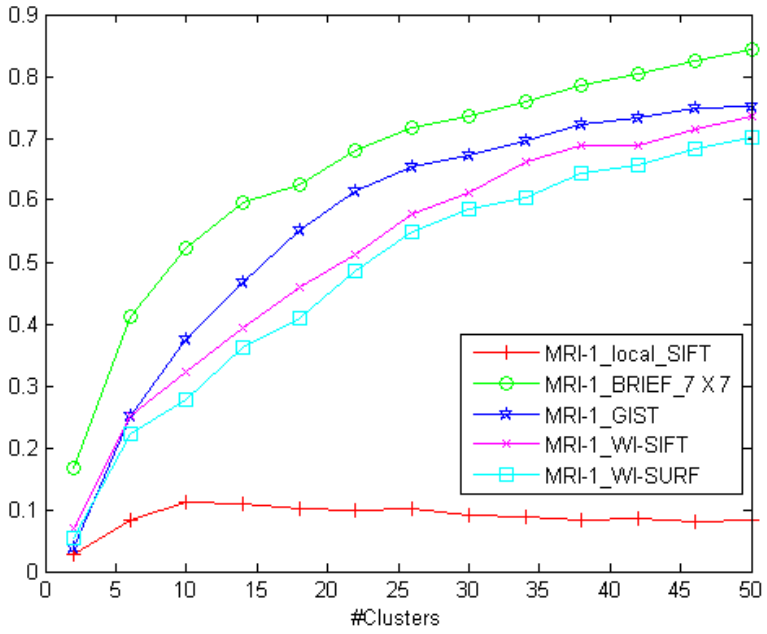


Figure 5.6: Comparison of the performance against perceptual aliasing of the common image descriptors with MRI-1 on Benchmark dataset.

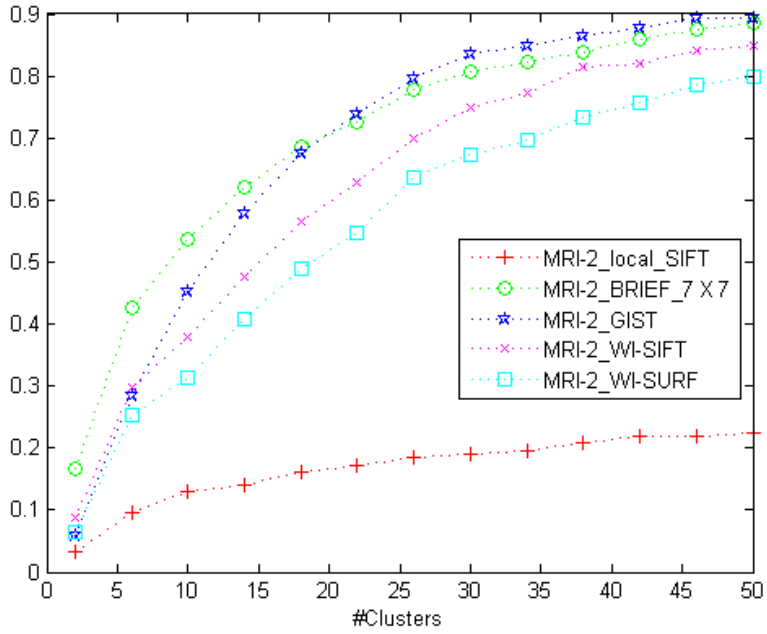


Figure 5.7: Comparison of the performance against perceptual aliasing of the common image descriptors with MRI-2 on Benchmark dataset.

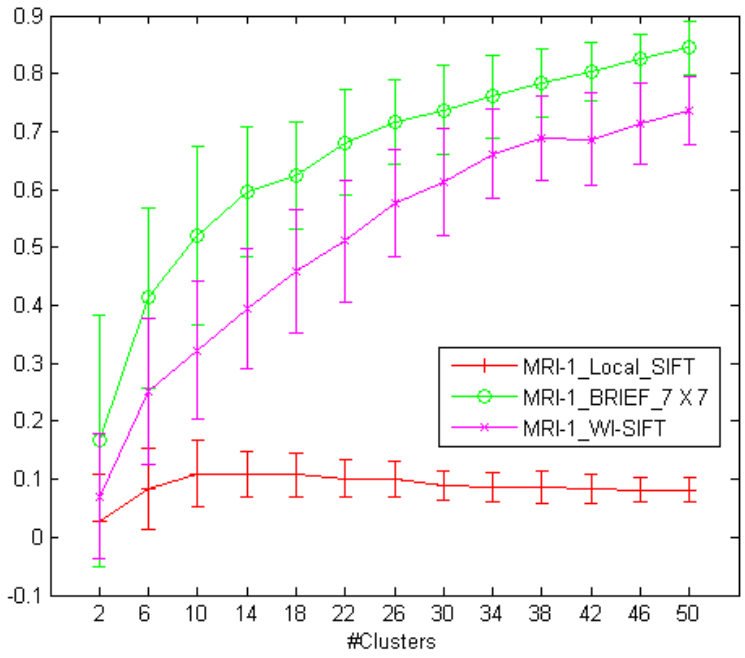


Figure 5.8: Standard deviation of MRI-1 values of local SIFT, BRIEF_7 $\times$ 7 and WI-SIFT.

compression, to demonstrate that our method can also be applied to local keypoint descriptors.

The evaluation of local keypoint descriptors also depends on the performance of keypoint detectors. For different detectors, the rank can be different since different detectors capture keypoints based on different properties. People who want to find out the optimal descriptor within an application can do experiments to evaluate the performance with their preferred keypoint detector. In our case, we choose Harris corner detector as the keypoint detector. In terms of the keypoint patch size, we do the experiments with different keypoint patch sizes and the results show that the rank of different local keypoint descriptors is independent of the patch size. For the following experiments, we set the patch size to 32×32. We then evaluate the performance of three popular local keypoint descriptors: BRIEF, SIFT and SURF. The similarity matrices are computed based on the keypoint patches collected as stated in Section 4.2.

Figure 5.9 illustrates the results of the experiments. For blur, viewpoint and zoom+rotation changes, there are two corresponding sets of images respectively. The "bikes" and the "trees" sets contain images of blur changes. Images in the "trees" set are of repeated textures. Similarly, The "graffiti" and the "wall" sets contain images of viewpoint changes with images in the "wall" set of repeated textures. The "boat" and the "bark" sets are of images of zoom+rotation changes. The "bark" has images of repeated textures. Figure 5.9(a) and Figure 5.9(c) show that SIFT achieves better performance under blur and viewpoint changes. However, with repeated textures, BRIEF is able to capture the property of keypoints better, leading to higher discriminating power as shown

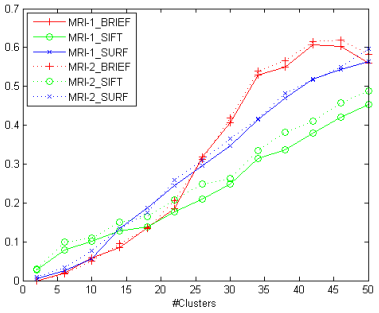
in Figure 5.9(b) and Figure 5.9(d). For zoom+rotation and illumination (the "cars" set) changes, Figure 5.9(e), Figure 5.9(f) and Figure 5.9(g) show that SIFT is always good in these cases, followed by SURF and BRIEF. Finally, under JPEG compression, BRIEF and SIFT behave similar.

Our experiments in this section demonstrate that our method can be applied to evaluate and compare the performance of local keypoint descriptors with respect to different changes in image conditions. The perceptual aliasing in keypoint descriptors is sensitive to different image condition changes. To compare keypoint descriptors, people can use their preferred keypoint detector to detect keypoints and then evaluate the performance of local keypoint descriptors of interest.

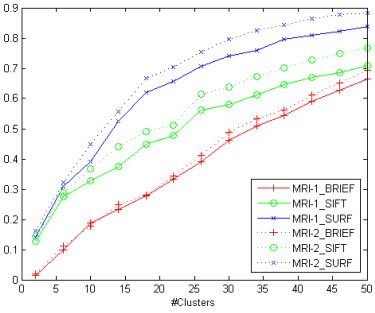
5.5 MRI vs. Precision and Recall

In this section, we discuss the different utility between our method and precision and recall in comparing image descriptors in clustering-based image applications by examining the evaluation results. This set of experiments is conducted on the Pittsburgh Street View dataset. We compare the performance of BRIEF_7 $\times$ 7 and GIST using both precision and recall and our proposed method. We create 100 image clusters with five images per cluster, extract image descriptors and compute the similarity matrix as before. MRI values have been calculated in Section 5.3.1. To measure precision and recall in terms of different numbers of image clusters, we set the ground truth number of image clusters from 2 to 50 and compute the precision and recall values given the corresponding sub similarity matrices. In our case, each image is taken as a query and the evaluation is executed in the context of image retrieval. Since each query has five ground truth matches (if the query image itself is used in image retrieval), precision and recall are computed on the top five retrieved images, leading the precision and the recall values to be the same in our case. For each number of image clusters, the experiment steps are repeated for 100 times and the average precision or recall are computed and plotted. The parameters are set the same way as what we did for computing MRI indices.

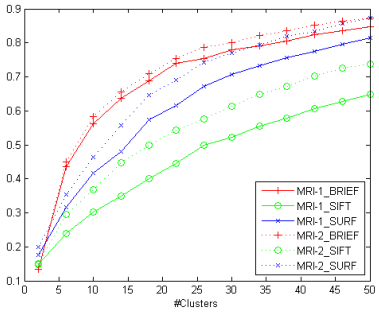
Figure 5.10 shows the experimental results on the common global image descriptors: BRIEF_7$\times$ 7 and GIST. The curves of precision and recall are almost flat, which means the results are insensitive to cluster count, and the interval between the curve of BRIEF_7 $\times$ 7 and the curve of GIST are too insignificant to tell the difference between the discriminating power of the image descriptors. These disadvantages result from their computation based on the top matched results. Specifically, to compute precision and recall, we simply find k nearest neighbor list for each query, which is essentially insensitive to the dataset size, i.e., when the number of images in the dataset increases, the k nearest neighbor list is most likely the same. Therefore, precision and recall values do not change with the cluster count. On the other hand, the metrics based on the k nearest neighbor list might only capture local similarity information and ignore the potential perceptual aliasing between other pairs of images, leading a minor difference between two curves if the k nearest neighbor lists are similar using different image descriptors. In contrast, to select an image descriptor for a clustering-based image application, global similarity information, i.e. the similarity between all



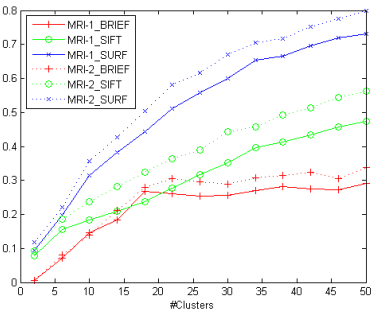
(a) MRI on bikes



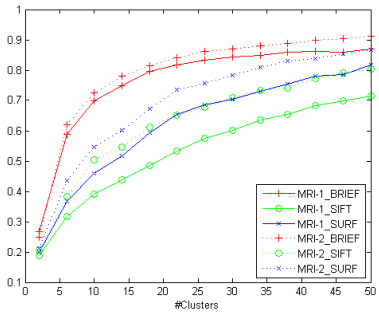
(b) MRI on trees



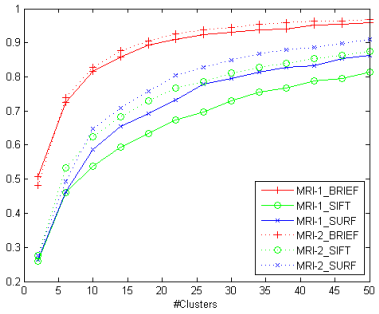
(c) MRI on graffiti



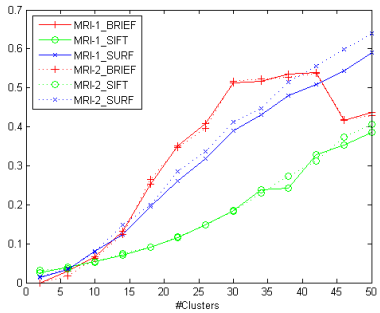
(d) MRI on wall



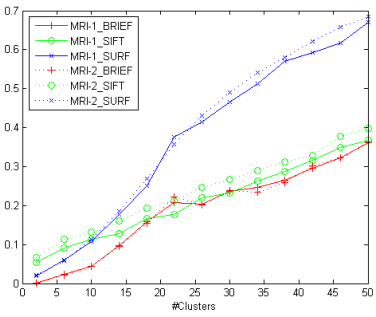
(e) MRI on boats



(f) MRI on bark



(g) MRI on cars



(h) MRI on ubc

Figure 5.9: MRI values of different local keypoint descriptors on eight sets of images under different image condition changes. (a)(b): Results under blur changes. (c)(d): Results under viewpoint changes. (e)(f): Results under zoom+rotation changes. (g): Results under illumination changes. (h): Results under JPEG compression.

image pairs, need to be taken into account to avoid overlap between two clusters. For example, in large scale image retrieval based on the bag-of-words method, we build a dictionary and expect few synonyms; In applications that image organization acts as a pre- or post-processing step, non-overlap between two image clusters is important to achieve efficiency or scalability. Performance metrics that use the k nearest neighbor list are not involved in these cases. The usage of clustering algorithm in our method can provide an alternative performance metric in such applications since clustering is based on the global similarity information, i.e. the similarity between all image pairs. As we can see in Figure 5.10, our measure can provide detailed information and the trend in terms of perceptual aliasing in image descriptors with different numbers of image clusters, therefore assisting in finding an optimal image descriptor for different application scenarios, especially for clustering-based image applications where a problem of grouping images exist.

In addition, the measure based on the top matched results may not capture the quality of the similarity matrix. Figure 5.11 and Figure 5.12 show the similarity matrix computed with BRIEF_7 $\times$ 7 and GIST descriptors on Pittsburgh Street View dataset respectively. The similarity matrix of BRIEF_7 $\times$ 7 descriptor is visually of higher quality than that of GIST descriptor, in terms of measuring image similarity. The ambiguous off-diagonal areas with large values are where perceptual aliasing can occur. However, the precision/recall curves indicate that GIST performs better than BRIEF_7 $\times$ 7, which contradicts visual observation. Our results in Figure 5.4 and Figure 5.5 provide the intuitively matched rank. The advantage of our method comes from the usage of clustering algorithm, which makes it possible to take into account the relations between all pairs of images, in measuring perceptual aliasing in image descriptors rather than the commonly used k nearest neighbor list for each query.

The time complexity of retrieval based performance measurement is $\mathcal{O}(mnk)$, where m is the number of test images, n is the number of images in the dataset, k is the length of the retrieved list. For each query, it takes $\mathcal{O}(n)$ to retrieve a matched image from the dataset by linear search. Therefore, the total time to retrieve k results based on m queries is $\mathcal{O}(mnk)$. On the other hand, the time complexity of our method depends on the complexity of spectral clustering algorithm. In our case, the spectral clustering is executed on the similarity matrix of k' image clusters, where k' is from 2 to 50. For each k', if there is c images per cluster, the time complexity of spectral clustering is $\mathcal{O}(k'c)^3$. Unlike retrieval based performance measurement which conducts retrieval from the whole dataset, our evaluation is based on the randomly selected image clusters from the dataset. Therefore, it will be less time-consuming than the retrieval based performance measurement.

In general, a measure of image descriptors sensitive to different number of image clusters is important in large scale image applications. our method can serve as a reference in image descriptor selection within these applications. Meanwhile, the clustering algorithm provides us the opportunity to explore the relations between all pairs of images in a dataset and evaluate the performance of image descriptions with respect to perceptual aliasing.

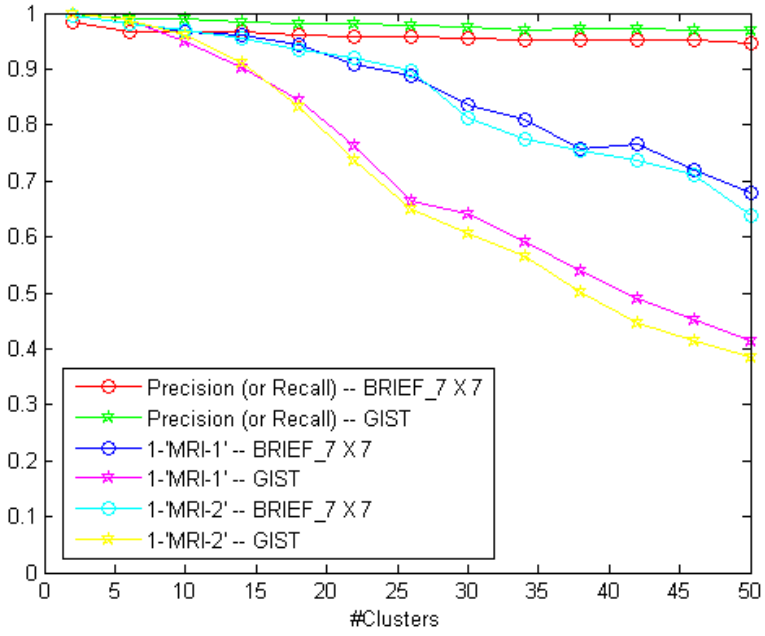


Figure 5.10: Precision/recall for different number of image clusters. y axis shows precision or recall values. x axis indicates the number of image clusters. We use $1-$MRI values to draw the curves. Note: The computation is based on the top five returned results, which means the precision and the recall are equal in our case.

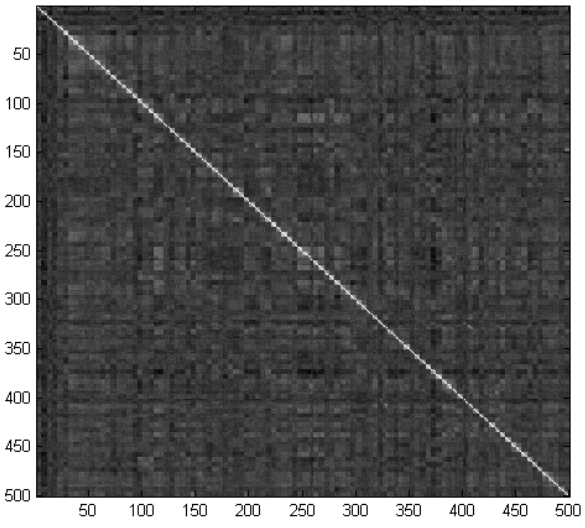


Figure 5.11: Similarity matrix computed with BRIEF_7 $\times$ 7 descriptors on Pittsburgh Street View dataset. There are 100 image clusters with five images per cluster.

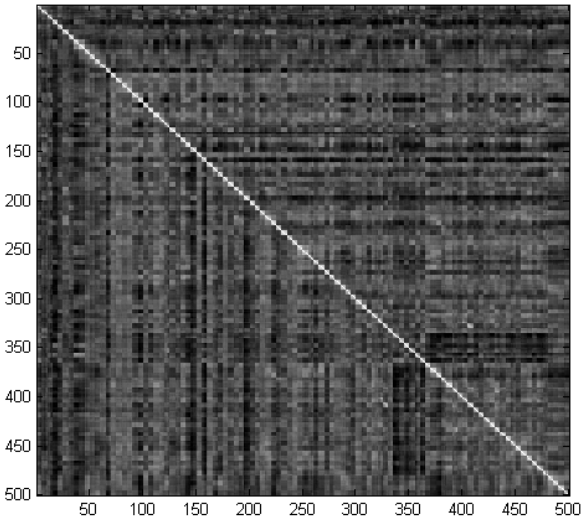


Figure 5.12: Similarity matrix computed with GIST descriptors on Pittsburgh Street View dataset. There are 100 image clusters with five images per cluster.

5.6 Summary

In this chapter, experimental results using our proposed method to measure perceptual aliasing in image descriptors are presented. Four sets of experiments have been conducted. The first set of experiments in evaluating and comparing different versions of BRIEF-Gist descriptor demonstrates that the results from our method are consistent with the literature. In the second set of experiments, we measure perceptual aliasing in the common global image descriptors which do not require local keypoint detectors on two different types of datasets. The results show that perceptual aliasing in image descriptors can be different within place recognition and image retrieval. Our method can serve as a reference to image descriptor selection. Furthermore, we conduct the third set of experiments to compare local keypoint descriptors under different changes in image conditions by collecting clusters of keypoint patches from Affine Covariant dataset. The comparison of local keypoint descriptors have been an interesting topic since various keypoint descriptors have been proposed and it is important to select an appropriate in different applications, such as object tracking, place recognition, people counting, etc. The performance of local keypoint descriptors with respect to perceptual aliasing is sensitive to different image condition changes and can be evaluated by our method. The last set of experiments is conducted to explain the different utility between our method and the other existing performance metrics, for example, precision and recall. Unlike the existing performance metrics which mainly rely on the top retrieved results, the property of clustering algorithm gives us the opportunity to measure perceptual aliasing existing in all pairs of images, which is important for clustering-based image applications.

Chapter 6

Conclusion and Future Work

In this thesis, we have presented a pair of novel performance metrics for comparing and measuring perceptual aliasing in image descriptors. The metrics are formulated directly from the definition of perceptual aliasing and calculated in a similar way to the Rand index, used popularly in comparing clustering results. We have also introduced a procedure in which images in the intended application are selected and organized into clusters and a similarity matrix constructed using descriptors of these images. In terms of local keypoint descriptors, the procedure can also be applied to keypoint patches collected from images under different image condition changes and the similarity matrix can be constructed correspondingly. With the help of spectral clustering and comparison with the ground truth clusters, we are thus able to evaluate the performance against perceptual aliasing of any descriptors. We should add that our method is applicable even when image descriptor is of no fixed length or explicitly available.

We have demonstrated the reliability and usefulness of our method by using different common global image descriptors, as well as local keypoint descriptors. The comparison results from our method are consistent with the literature but provide more detailed conclusion concerning perceptual aliasing. Furthermore, the proposed method is efficient as it does not involve an application. The results can be taken as an important reference when choosing the appropriate global or local image descriptor for a specific application. The comparison of our MRI metrics with other existing performance metrics for image descriptor evaluation demonstrates that our method is more intuitively sensitive to different number of clusters and potential perceptual aliasing among image pairs. The advantage comes from the clustering based performance measurement which takes into account the relations between all pairs of images besides the top retrieved ones. The proposed method is able to capture important descriptor properties that other common metrics such as precision and recall fail to capture.

Our future work includes the verification of the usefulness of our method in real applications. In the thesis, we conduct several experiments to compare both local and global image descriptors, as well as compare our method with other existing performance metrics. It is important to apply the proposed method to real application scenarios and evaluate the performance to further demonstrate

the utility.

Another aspect of our work that can be improved is that we compare and evaluate the performance of the common image descriptors by selecting 2 to 50 image clusters. The cluster count is relatively smaller than that in real world. We are also interested in investigating perceptual aliasing in image descriptors for much larger scale image applications.

Bibliography

[1] Hernn Badino, Daniel F. Huber, and Takeo Kanade. Real-time topometric localization. In *ICRA*, pages 1635–1642, 2012.

[2] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded-up robust features. *Computer Vision and Image Understanding*, 110(3):346–359, June 2008.

[3] Deng Cai, Xiaofei He, Zhiwei Li, Wei-Ying Ma, and Ji-Rong Wen. Hierarchical clustering of www image search results using visual, textual and link information. In *Proceedings of the 12th annual ACM international conference on Multimedia*, pages 952–959, 2004.

[4] Michael Calonder, Vincent Lepetit, Christoph Strecha, and Pascal Fua. Brief: binary robust independent elementary features. In *ECCV*, pages 778–792, 2010.

[5] Vijay Chandrasekhar, David M. Chen, Andy Lin, Gabriel Takacs, Sam S. Tsai, Ngai-Man Cheung, Yuriy A. Reznik, Radek Grzeszczuk, and Bernd Girod. Comparison of local feature descriptors for mobile visual search. In *ICIP*, pages 3885–3888. IEEE, 2010.

[6] Yixin Chen, James Z. Wang, and Robert Krovetz. Content-based image retrieval by clustering. In *Multimedia Information Retrieval*, pages 193–200, 2003.

[7] Bruce Croft, Donald Metzler, and Trevor Strohman. *Search Engines: Information Retrieval in Practice*. 2009.

[8] M. Cummins and P. Newman. Invited Applications Paper FAB-MAP: Appearance-Based Place Recognition and Mapping using a Learned Visual Vocabulary Model. In *27th Intl Conf. on Machine Learning (ICML2010)*, 2010.

[9] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893, 2005.

[10] Thomas Deselaers, Daniel Keysers, and Hermann Ney. Features for image retrieval: an experimental comparison. *Information Retrieval*, 11(2):77–107, April 2008.

[11] Ronald Fagin, Ravi Kumar, and D. Sivakumar. Comparing top k lists. In *In Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, pages 28–36, 2003.

[12] Steffen Gauglitz, Tobias Höllerer, and Matthew Turk. Evaluation of interest point detectors and feature descriptors for visual tracking. *International Journal of Computer Vision*, 94(3):335–360, September 2011.

[13] Chris Harris and Mike Stephens. A combined corner and edge detector. In *In Proceeding of Fourth Alvey Vision Conference*, pages 147–151, 1988.

[14] Eva Hörster, Thomas Greif, Rainer Lienhart, and Malcolm Slaney. Comparing local feature descriptors in plsa-based image models. In *Proceedings of the 30th DAGM symposium on Pattern Recognition*, pages 446–455, Berlin, Heidelberg, 2008. Springer-Verlag.

[15] L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.

[16] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems*, 20:2002, 2002.

[17] Hervé Jégou, Matthijs Douze, Cordelia Schmid, and Patrick Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, pages 3304–3311, Jun 2010.

[18] Edward Johns and Guang-Zhong Yang. From images to scenes: Compressing an image cluster into a single scene model for place recognition. In *ICCV*, pages 874–881. IEEE, 2011.

[19] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30:81–93, 1938.

[20] Santhana Krishnamachari and Mohamed Abdel-mottaleb. Hierarchical clustering algorithm for fast image retrieval. In *In IS&T/SPIE Conference on Storage and Retrieval for Image and Video databases VII*, pages 427–435, 1999.

[21] Benjamin Kuipers and Patrick Beeson. Bootstrap learning for place recognition. In *AAAI/IAAI*, pages 174–180, 2002.

[22] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, pages 2169–2178, 2006.

[23] Stefan Leutenegger, Margarita Chli, and Roland Siegwart. Brisk: Binary robust invariant scalable keypoints. In *ICCV 2011*, pages 2548–2555, 2011.

[24] Jing Li and Nigel M. Allinson. A comprehensive review of current local features for computer vision. *Neurocomputing*, 71(10-12):1771–1787, June 2008.

[25] Ting Liu, Charles Rosenberg, and Henry A. Rowley. Clustering billions of images with large scale nearest neighbor search. In *Proceedings of the Eighth IEEE Workshop on Applications of Computer Vision*, WACV 2007, 2007.

[26] David G. Lowe. Object recognition from local scale-invariant features. In *ICCV*, pages 1150–1157, 1999.

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

[28] Ulrike Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, December 2007.

[29] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *Proceedings of the British Machine Vision Conference*, pages 36.1–36.10, 2002.

[30] Krystian Mikolajczyk and Cordelia Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

[31] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 27(10):1615–1630, 2005.

[32] Glenn Milligan and Martha Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2):159–179, 1985.

[33] Henning Müller, Stéphane Marchand-Maillet, and Thierry Pun. The truth about corel - evaluation in image retrieval. In *Proceedings of the International Conference on Image and Video Retrieval*, pages 38–49, 2002.

[34] David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *CVPR*, pages 2161–2168, 2006.

[35] Aude Oliva and Antonio Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42:145–175, 2001.

[36] Raphael Ortiz. Freak: Fast retina keypoint. In *CVPR*, pages 510–517, 2012.

[37] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *CVPR*, 2007.

[38] William M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

[39] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *ICCV*, volume 2, pages 1470–1477, October 2003.

[40] Niko Snderhauf and Peter Protzel. Brief-gist - closing the loop by simple means. In *IROS*, pages 1234–1241. IEEE, 2011.

[41] Qi Tian, Shiliang Zhang, Wengang Zhou, Rongrong Ji, Bingbing Ni, and Nicu Sebe. Building descriptive and discriminative visual codebook for large-scale image applications. *Multimedia Tools Appl.*, 51(2):441–477, 2011.

[42] Christoffer Valgren and Achim Lilienthal. Incremental spectral clustering and seasons: Appearance-based localization in outdoor environments. In *ICRA*, 2008.

[43] Fredrik Viksten, Per-Erik Forssén, Björn Johansson, and Anders Moe. Comparison of local image descriptors for full 6 degree-of-freedom pose estimation. In *ICRA*, pages 1139–1146, 2009.

[44] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: is a correction for chance necessary? In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1073–1080, 2009.

[45] E. Voorhees. The TREC-8 question answering track report. In *Proceedings of the 8th Text REtrieval Conference (*TREC-*8)*, pages 77–82, 1999.

[46] Yining Wang, Liwei Wang, Yuanzhi Li, Di He, Tie-Yan Liu, and Wei Chen. A theoretical analysis of ndcg type ranking measures. *CoRR*, 2013.

[47] Steven D. Whitehead and Dana H. Ballard. Learning to perceive and act by trial and error. *Machine Learning*, 7(1):45–83, July 1991.

[48] J. Winn, A. Criminisi, and T. Minka. Object categorization by learned universal visual dictionary. In *ICCV*, pages 1800–1807, 2005.

[49] Jianxin Wu and James M. Rehg. Centrist: A visual descriptor for scene categorization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(8):1489–1501, 2011.

[50] Yimeng Zhang, Zhaoyin Jia, and Tsuhan Chen. Image retrieval with geometry-preserving visual phrases. In *CVPR*, pages 809–816. IEEE, 2011.

[51] Y. Zhao and G. Karypis. Criterion functions for document clustering: Experiments and analysis, 2001.