Optimizing Content-Based Image Retrieval for Geolocation Estimation

By

YIRAN LIU

Department of Data Science and Technology
DELFT UNIVERSITY OF TECHNOLOGY

A dissertation submitted to the Delft University of Technology in accordance with the requirements of the degree of MASTER OF DST in the faculty of EEMCS

MARCH 2017

Supervisor: Martha Larson

Abstract

The prediction of geo-graphical location at which an image is taken is drawing increasing attention in recent research. However, one major limitation of most current research is that it focuses mostly on improving the geolocation prediction performance while ignoring the problem of index size, which is helpful in saving storage space. Traditional image retrieval index reduction approach can be achieved at the cost of losing retrieval performance, e.g., by using low-level features. This thesis investigates how to optimize the content-based image retrieval for geolocation estimation, by reducing the large-scale image retrieval index size without losing geo-prediction performance. More specifically, it focuses on the challenge of trade-off between index size and geo-prediction performance.

The aim of this research is to propose an approach to investigate the possibilities to reduce the index size and improve the geo-prediction performance based on *'Large Scale Image Retrieval for Location Estimation'*. To solve the research challenge, Common Concepts Removal (CCR) is proposed, which is built based on the SSD deep learning framework. In this approach we believe that some common concepts (e.g., cars, persons, buses, etc) in restricted scenario cannot contribute to the geolocation prediction performance and the index size can be considerably reduced by removing them. These kinds of common concepts exist everywhere in the city streets and look similar, which means that they can hardly contribute to the geolocation prediction performance and even harm the prediction result in some special circumstances. We manually defined eight common concepts in San Francisco and analyzed their different influence on the geolocation prediction. We implement CCR for three different geo-prediction approaches, 1-Nearest Neighbor, Geo-Visual Ranking, and Geo-Distinctive Visual Element Matching for the geo-constrained scenario-San Francisco street view dataset. The experiment results illustrate that using this approach, the index size can be reduced by 30.6% while the performance is improved by approximately 6.0%.

Based on the findings presented in this thesis, we make recommendations for future research directions, which we argue are substantial and promising for further reducing the index size as well as improving content-based geolocation prediction performance.

Dedication and Acknowledgements

Two years ago, when I first stood on this unfamiliar land, everything was so different from my homeland: a different culture, a different lifestyle, and different weather. With all my enthusiasm and curiosity, I started my new career here in Delft for my Master degree. Two years is a long time; long enough for me to learn new knowledge and sharpen my research skills. However, two years is also short, which makes me reluctant to leave my beloved campus. The two years journey contains a large amount of difficulties and hard work. Luckily, I am not alone on my way pursuing new knowledge and challenges with so many kind and excellent people to be with me. I sincerely acknowledgement all of their help.

First and foremost, I would like to thank my supervisor, Prof. Martha Larson, for the patient guidance, encouragement, and suggestions you have provided throughout my thesis time as your student. I have been extremely lucky to have a supervisor who cared so much about my work, and who responded to my questions so promptly. Thank you for providing me with the remote computer cluster on SURFsara that I required to proceed working in the right direction and successfully complete my thesis. I enjoyed the time discussing with you about the research questions and basic strategies in this thesis, and your passion always motivate me keep moving forward. It was definitely a great pleasure for me to work on my thesis under your guidance.

Many people have helped and taught me immensely in the process of working on my thesis. Xinchao Li, thank you so much for helping me understand your remarkable work on *'Large Scale Image Retrieval for Location Estimation'*. You are like a big brother to me, whenever I met with difficulties, you always squeezed your precious time and answered my questions with great patience, even though you were busy in the night. Jaehun Kim, thank you for showing me the usage of SURFsara remote server and helping me with the storage problem. Jaeyoung Choi, thank you for discussing with me in the process of re-implementing Li's work and offering me your precious suggestions of my thesis. Finally, I would like to acknowledge SURFsara, the national high-performance computing and e-science support center in Netherlands. With all your supports, I can overcame difficulties and finish my research.

There is an ancient Chinese saying - 'There must be one out of three who can be my teacher'. I would like to thank all my friends, Bo Wang, Yun Liu, Xiao Wang, Bairong Wu, and all my friends in Delft Basketball Team. From you, I have learned courage, confidence, and optimistic, which are the faith that leads to achievement. I am so grateful to have your accompany in Netherlands.

Particularly, I would like to thank my family, especially Mom and Dad, for the continuous support you have given me throughout my time in Netherlands. You are always the source of energy in my life that motivate me to realize my ideal.

TABLE OF CONTENTS

LIST OF TABLES

TABLE **Page**

List of Figures

Figure **Page**

CHAPTER 1

INTRODUCTION

The geographical location at which an image or video was taken is a key piece of multimedia information. Such geo-information has become an indispensable component of systems enabling personalized and context-aware multimedia services. Although nowadays, images taken by people may be automatically tagged with geolocation, there still exist a large number of images that are not labeled with such kind of geo-information. For example, the images uploaded to platform *Flickr* usually do not contain GPS-based latitude/longitude coordinates. Obtaining such geographic information is beneficial for a variety of applications including the E-learning domain and remote sensing.

In this thesis, we focus on reducing the large-scale image retrieval index while improving the geolocation prediction performance. The geolocation prediction algorithm is based on *'Large Scale Image Retrieval for Location Estimation'* [1–3]. These research proposed several novel approaches to estimate the geolocation globally based on the content of images. For example, [3] an approach called Geo-Distinctive Visual Element Matching (DVEM) is put forward to maximize the influence of visual elements that are geo-distinctive. There are currently many available methods to deal with the CBIR indexing process, such as Multidimensional Indexing Method, Dimension Reducing Method, Approximate Nearest Neighbor, etc. However, almost all these methods face the same unfortunate situation, the unavoidable trade-off between index dimension and image retrieval performance. The retrieval performance can reach a high level with high index dimension. On the contrary, the retrieval performance will be sacrificed with low index dimension. Many researchers in recent years put the effort in finding out a better solution to solve this problem, such as [4, 5].

Inspired mostly by *'Large Scale Image Retrieval for Location Estimation'*, we realize that there are usually a lot of common concepts, which are not geo-distinctive and cannot contribute

to estimate the geolocation that existing in images taken from a certain area, such as city San Francisco. For example, cars, persons, and buses, these kinds of common concepts exist everywhere in the city streets and look similar, which means that they can hardly contribute to the geolocation prediction performance and even harm the prediction result in some special circumstances, for example, cars and buses may block the distinctive architectures. In other words, we believe that stationary concepts like architectures in city scale can mostly contribute to the geolocation prediction process due to the reason that they are unique and geo-distinctive. For example, there is only one *Eiffel Tower* all over the world, which means that the coordinate is unique once the tower exists in the query image.

In view of above, instead of aiming to find out better solutions to deal with the problem of indexing dimensions, we proposed a simple but effective way to reduce the retrieval index, based on the idea that some common concepts cannot contribute to the geolocation estimation process. First, through applying deep learning framework SSD, we detect the coordinates and class names of the 8 manually defined concepts (car, person, bus, plant, train, motorbike, bicycle, boat) in images. Then, we produce the retrieval index based on *'Large Scale Image Retrieval for Location Estimation'* and avoid the detected common concept areas by removing them. Comparing with the original approaches in *'Large Scale Image Retrieval for Location Estimation'*, which have to index full image area, our proposed method only needs to index some special parts of images, like buildings, which can easily produce more representative features and considerably reduce the size of retrieval index. Currently, the retrieval index size of San Francisco street view dataset produced by *'Large Scale Image Retrieval for Location Estimation'* pipeline is 73.8GB, by applying the method-CCR that is proposed in this thesis, the retrieval index size can be reduced by approximately 30.9% to 51.0GB, and the geolocation prediction performance is improved by approximately 6%.

1.1 Background

The research work reported in this thesis can be considered as an extension of Li's work in *Large Scale Image Retrieval for Location Estimation*, reducing the retrieval index and improving the geolocation prediction performance. The goal of *'Large Scale Image Retrieval for Location Estimation'* is to develop a scalable visual content-based location estimation system for images and to investigate the possibilities to improve its accuracy and reliability. This system is applicable to both the geo-constrained scenario, in which the multimedia item is taken at one of a previously defined set of locations, and the geo-unconstrained scenario, in which the multimedia item could have been taken anywhere in the world. The system structure of *'Large Scale Image Retrieval for Location Estimation'* is shown in Fig. 1.1.

Currently, two general approaches can be followed to infer the location information from the visual content of images: classification-based approach and search-based approach. Classification-

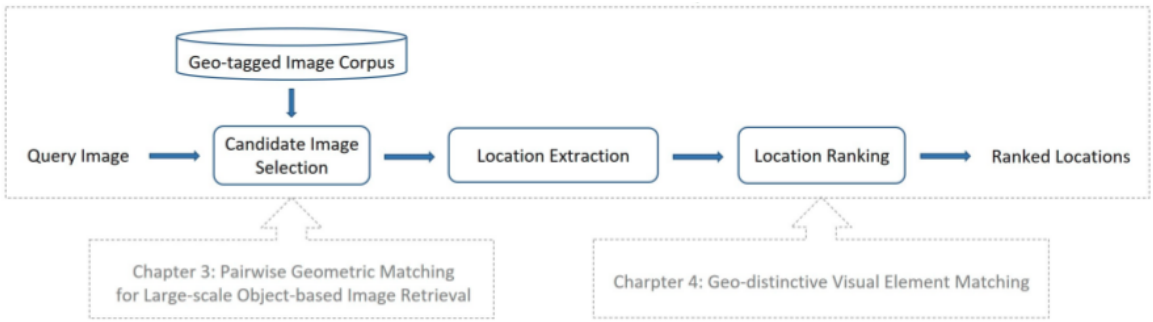

FIGURE 1.1. *Large Scale Image Retrieval for Location Estimation* structure [1–3].

based approaches collect location-related visual clues from different training images, and make a compact representation for individual locations. However, their main disadvantage is that it is highly problematic to formulate a location as a single class. Thus, *'Large Scale Image Retrieval for Location Estimation'* chose to apply a search-based framework for location estimation, which requires only a single relevant photo for a given query, if does not matter whether this photo captures a frequently photographed visual scene or a rarely photographed one.

The work of *'Large Scale Image Retrieval for Location Estimation'* is composed of three parts. Firstly, this research [1] unravels the problem of location inference from visual content by introducing the search-based approach and proposes a novel way of implementing it, namely in the form of a *Geo-Visual Ranking (GVR) method* that takes into account the ambiguity of how visual content describes a location. The rationale underlying the *GVR* method is that, compared to the images from a wrong location, more images from the true location will likely contain more elements of the query image's visual content. Then, to improve the scalability and robustness of object-based image retrieval in *GVR* framework, this research presents a novel *Pairwise Geometric Matching* method [2] for the spatial verification stage. It uses the global scale and rotation relations to enforce the local consistency of geometric relations derived from the locations of pairwise correspondence. Since some objects may be common to different visual scenes (e.g., common static objects and mobile objects), an additional adaptation of the framework is required to make it focus on the scene-distinctive objects only. Thus, this research also presents a novel *Geo-Distinctive Visual Element Matching* method [3] to further improve the robustness of the location estimation framework. It explores and exploits geographical distinctiveness of visual elements found in the query image, and it further strengthens the support for finding the true location by devising an aggregated visual representation of a location that combines all visual elements from the query found in the images of that location.

1.2 Motivation

Why do we need to reduce the image retrieval index? What can content-based geolocation prediction systems benefit from reducing the retrieval index size? These are the very first two questions that motivate us to start discovering and researching in this thesis. With the rapid development of high capacity computation platforms, such as Amazon and SURFsara, which make cloud computing easier, researchers can deal with increasingly large amount of data size. However, due to the widely used mobile devices like smart phones and cameras, the data size online also grows explosively. Thus, although we can process millions of image dataset in very short time, it is still necessary to research in reducing the image retrieval index. Content-based geolocation prediction systems can benefit from reducing the index size through several aspects: Firstly, the systems' computational complexity can be reduced, which means we can deal with a larger size of dataset using the same computational capacity. Secondly, by reducing the image retrieval index size, we can save considerable storage space. Finally, researchers can save research funding, because the image retrieval program can be processed in shorter time.

The unavoidable trade-off between index size and image retrieval performance also motivate us to find a better solution to reduce the image retrieval index and improve the image retrieval performance. Recently, researchers contribute a lot of work in image retrieval indexing field [4–6]. However, most of the research cannot avoid the research bottleneck - the trade-off between index size and image retrieval performance. Thus, it is not surprising that most research focuses on applying low-dimensional indexing techniques to achieve efficient and effective retrieval performance, such as color, shape, etc. Meanwhile, some of the image retrieval indexing approaches have been only tested on the small size of dataset. For instance, Prasad in [4] only tested his approach with 200 images of flags and 120 images of fruits, flowers and simulated objects (squares, rectangles, triangles, circles, etc). We firmly believe that the index size can indeed be reduced by adopting these approaches, but with geolocation prediction performance sacrificed.

1.3 Objective - Improving Geo-Prediction Accuracy with Reduced Retrieval Index

The challenge addressed in this thesis is illustrated in Fig. 1.2 and can be formulated as follows: "*Adapting 'Large Scale Image Retrieval for Location Estimation', reducing the image retrieval index size and improving the geolocation prediction performance*".

This challenge is substantial for two reasons. First, re-implementing *'Large Scale Image Retrieval for Location Estimation'* is extremely complex and time-consuming, because there are several crucial algorithms in this pipeline and a large amount of code needs to be modified. Meanwhile, many techniques and platforms are needed to be learned and implemented, such as *Eclipse, Hadoop, MapReduce, Yarn, etc*. Second, retrieval index reduction usually results in sacrificing the performance. For example, traditional approaches to reduce the image retrieval

FIGURE 1.2. Illustration of the challenge of reducing the index size and improving the geolocation prediction performance

.

index usually focusing on applying low-level features such as color, shape, etc. These approaches can help reduce the index size, however, the corresponding consequence is sacrificing the retrieval performance [4–6].

In view of these considerations, we focus our research on reducing retrieval index size without applying lower dimensions. Our expectation is that the solution arising from our research should work well in both scenarios, that is for both reducing the index size and improving the geolocation prediction performance.

1.4 Research Questions

The research questions of this thesis are listed below:

- **RQ1:** Does removing concepts from query images improve the geolocation estimation? (Sec. 4.2.3)
- **RQ2:** Does removing concepts from training images help with reducing the index? (Sec. 4.3.3)
- **RQ3:** Does removing concepts from query and training images help with improving the performance and reducing the index? (Sec. 4.3.3)
- **RQ4:** What is the influence of different concepts on location prediction performance? (Sec. 4.2.3)
- **RQ5:** Does the pre-trained model performs better comparing with the self trained model?
- **RQ6:** How to evaluate the SSD detection result? (Sec. 4.1.2)

1.5 Thesis Contribution and Layout

This thesis makes the following contributions.

- This thesis proposes the approach CCR (Common Concept Removal) and solves the problem of trading-off between index size and geolocation prediction performance.

- The index size of *Large Scale Image Retrieval for Location Estimation* has been reduced from 73.8GB to 51.0GB.
- The geolocation prediction performance is improved by approximately 6.1% under geo-constrained scenario.
- Three different geolocation prediction approaches have been applied in this thesis to test the performance of CCR.
- Eight manually defined common concepts have been removed separately from query images and their different performances have been analyzed.

Although the proposed methods in *'Large Scale Image Retrieval for Location Estimation'* delivers a considerable performance improvement compared to the state of art, the index size is still large, which is 73.8GB when using San Francisco street view dataset. This thesis mainly focuses on resolving this problem and improving the geolocation prediction performance by putting forward alternative algorithms. Building on recent breakthroughs in semantic segmentation and fine-grained localization using deep learning techniques, attempts have been made to simultaneously detect and segment objects contained in an image, from which the pixel-level object class knowledge is available. This knowledge can then serve as the context information in object retrieval and provide guidance for making geometric constraints when building correspondences between images.

This thesis proposes a novel, straight, and effective approach to reduce the image retrieval index and improve the geolocation prediction performance, by applying the deep learning framework SSD, the common concepts in images can be detected, such as cars, buses, and persons. The basic structure of this approach is shown in Fig. 1.2. Then by recording and storing the coordinates and class names of detected common concepts in the database, we read the detected areas on a pixel level and make these areas transparent, which means give value $(255, 255, 255, 0)$ to each pixel. There are two advantages of this approach: Firstly, the origin San Francisco street view dataset size can be reduced by 19.9% from 32.2GB to 25.8GB. Secondly, in the process of predicting the geolocation, the system can avoid extracting features from these common areas, which can help reduce the retrieval index size. After successfully implementing (Fig. 1.3) our proposed approach in *'Large Scale Image Retrieval for Location Estimation'*, the index size is reduced from 73.8GB to 51GB, and the geolocation prediction performance is improved by approximately 6.0%.

In addition, we implement our proposed common concept removal method on 1-Nearest-Neighbor classifier for the geolocation estimation task, which adopts the geolocation of the image that is visually most similar to the query image as the predicted location. Once this image has been identified, its geo-coordinates are propagated to the query image. 1-NN classifier is an easy and stable image classification algorithm for which no learning is necessary. This algorithm relies on the distance between feature vectors. In [1], Li compares the behavior between his proposed

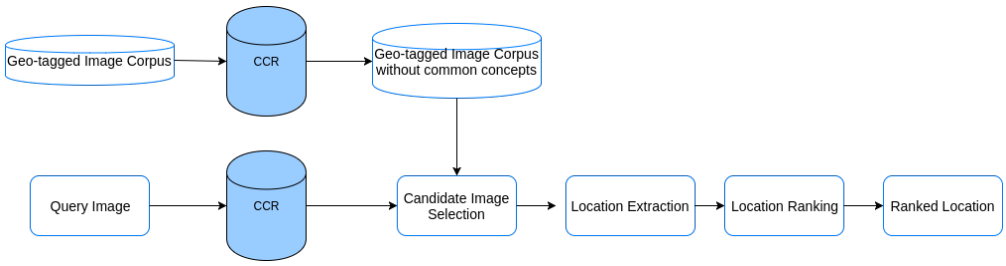

FIGURE 1.3. Pipeline of this thesis.

method GVR with 1-NN classifier, which shows that the 1-NN classifier is simple, but in practice has been proven difficult to beat. Last, we analyze the different influences of 8 manually defined concepts to the geolocation prediction performance, we re-run the geolocation prediction pipeline 9 more times by removing the concepts separately.

The rest of this thesis is composed of 4 chapters. In **Chapter 2**, the related work and recent research are described. In **Chapter 3**, we briefly illustrate the experiment framework and introduce the related techniques that have been adopted to this thesis. The implementation of our proposed approach is indicated in **Chapter 4**, in which the geolocation prediction performances and index reduction size are compared among 1-NN classifier, GVR, and DVEM approaches. Finally, conclusion and future work are presented in **Chapter 5**.

CHAPTER 2

RELATED WORK

The related challenges in estimating the geolocation of an image using only its visual content has drawn increasing research attention in the past years. Work addressing this challenge has been pursued along several major directions. Geo-constrained prediction, where the possible locations at which the target image could have been taken are limited to a defined geographic range or a set of predefined locations. Geo-unconstrained prediction, assumes that the target image could have been taken anywhere around the globe. Concept detection, can be better understand as the purpose of detecting the existing objects in images. Retrieval index reduction, which can help with the retrieval process, reducing the computation complexity and make the retrieval process more affordable. We briefly elaborate on the reported achievements in these four directions.

2.1 Geo-Constrained Location Estimation

Early work on geolocation estimation focuses on small areas, such as street level or certain cities, and can be regarded as Geo-Constrained Location Estimation.

Currently, most research about geolocation estimation focuses on the direction of relying on SIFT algorithm to find out the best matched location. For instance, in [7], authors aim to find out the accurate image location based on Google Maps Street View by indexing the SIFT descriptors of the detected SIFT interest points in the reference image using a tree. The tree is queried using the detected SIFT descriptors in the query image. A novel GPS-tag-based pruning method removes the less reliable descriptors. Then, a smoothing step with an associated voting scheme is utilized; this allows each query descriptor to vote for the location where its nearest neighbor belongs to. Other research applied the SIFT algorithm to select the closest views in the database and then deal with large percentage of outliers based on an efficient robust estimation

technique, which is also accompanied by a model selection step among the fundamental matrix and the homograph [8]. Once the motion between the closest reference views is estimated, the location of the query view is then obtained by triangulation of translation directions.

In the scope of Geo-Constrained Location Estimation, Landmark Identification is also an attractive and popular research direction, which keeps drawing increasing attention and can be applied to find out the landmark in a certain area, for example, Eiffel Tower in Paris. A new method of landmark recognition is proposed in [9] by fusing two popular representations of street level image data-facade-aligned and viewpoint-aligned and show that they contain complementary information that can be exploited to significantly improve the recall rates on the city scale. This paper also improves feature detection in low contrast parts of the street level data, and discusses how to incorporate priors on a user's position, which previous approaches often ignore. Landmark recognition technique is also expanded to mobile robot navigation field, for example in [10] and [11]. Other research propose the concept of *PREACTE* to predict the appearance and disappearance of objects [10], thereby reducing computational complexity and locational uncertainty and [11] introduces the recognition model based on the affine moment invariant (AMIs), whose ability regarding the particular landmark shape is investigated in the presence of additive random noise and in the case of various viewing angles.

Location recognition is often cast as an image retrieval problem and recent research has almost exclusively focused on improving the chance that a relevant database image is ranked high enough after retrieval, like in [8] and [7]. The implicit assumption is that the number of inliers found by spatial verification can be used to distinguish between a related and an unrelated database photo with high precision. However, this assumption does not hold for large dataset due to the appearance of geometric bursts, such as sets of visual elements appearing in similar geometric configurations in unrelated database photos. [12] proposes algorithms for detecting and handling geometric bursts by using the weighting schemes, which dramatically improve the recall that can be achieved when high precision is required compared to the standard re-ranking based on the inlier count. While, on the contrary, [13] argues that repeated visual elements such as structures are not a nuisance but, when appropriately represented, they form an important distinguishing feature for many places. It describes a representation of repeated structures suitable for scalable retrieval based on robust detection of repeated image structures and a simple modification of weights in the bag-of-visual-word model.

Content-based geolocation estimation can also be realized by applying classifiers, such as *K-NN* and *SVMs*. [14] casts the place recognition problem as a classification task and uses available geo-tags to train a classifier for each location in the database in a similar manner to per-exemplar *SVMs* in object recognition and proposes a new approach to calibrate all the per-location *SVM* classifiers using only the negative examples.

2.2 Geo-Unconstrained Location Estimation

Most approaches concerning content-based geolocation estimation typically attempt to narrow the domain of estimation and tackling the task in a geo-constrained way. They either estimate location within a geographically constrained area, within a set of predefined regions, or by reducing the task to specific landmark recognition ([10], [11]). Due to the difficulty of the challenge, there have only been a few attempts to tackle the geolocation estimation problem in a geo-unconstrained way, that is, where the target location can be any place around the world, for example [1] and [15].

[1], [2] and [3] are 3 main parts in *Large Scale Image Retrieval for Location Estimation* of Li's work. In [1], Li proposes an automatic method that addresses the challenge of predicting the geolocation of social images using only the visual content of those images. This method can generate a geolocation prediction for an image globally. For a given query image a geolocation is recommended based on the evidence collected from images that are not only geographically close to this geolocation, but also have sufficient visual similarity to the query image within the considered image collection. [2] considers the pairwise geometric relations between correspondences and proposes a strategy to incorporate these relations at significantly reduced computational cost, which makes it suitable for large-scale object retrieval. In addition, Li combines the information on geometric relations from both the individual correspondences and pairs of correspondences to further improve the verification accuracy. [3] proposes a novel approach called Geo-Distinctive Visual Element Matching (DVEM), using representations that are specific to the query image whose location is being predicted. These representations are based on visual element clouds, which robustly capture the connection between the query and visual evidence from candidate locations, then maximize the influence of visual elements that are geo-distinctive because they do not occur in images taken at many other locations.

In addition to Li's work, there is also some other research that deals with the content-based geolocation estimation on global scale, such as [16] and [17]. [17] proposes a simple algorithm for estimating a distribution over geographic locations from a single image using a purely data-driven scene matching approach and shows that geolocation estimation can provide the basis for numerous other image understanding tasks such as population density estimation, land cover estimation or urban/rural classification. Although [16] also focuses on finding out the content-based geolocation on global scale, it works out only the landmark recognition problem globally. Thus, this job is less challenging compare with the work addressed in [1], [2] and [3].

2.3 Concept Detection Algorithms

Concept detection is one of the most popular methods applied to find the matching images in content-based geolocation estimation. Under concept detection based image retrieval, we understand that the problem of finding images that contain the same concepts or scene elements as in the query images, however, possibly captured under different conditions in terms of rotation,

viewpoint, zoom level, occlusion or blur. Many concept detection approaches and methods ([18], [19], [20]) have been proposed recently, largely built on the Bag-Of-Features (BOF) principle for image representation.

Concepts in images can be extracted in many ways, such as indicated in [21] and [22]. [23] proposes a method for extracting distinctive invariant features from images that can be used to perform reliable matching between different views of an object or scene, which is realized by the detection of scale-space extrema and key point localization. This paper also claims that the extracted concepts are highly distinctive that a single concept can be correctly matched with high probability against a large database of features from many images. To optimize the concepts extraction result, improving the detection scores of concepts is usually adopted, such as [19]. This paper introduces a novel contextual fusion method to improve the detection scores of semantic concepts in images and videos. Method proposed in this paper mainly consists of three phases, for each individual concept, the prior probability of the concept is incorporated with detection score of an individual SVM detector. Then probabilistic estimates of the target concept are computed using all of the individual SVM detectors. Finally, these estimates are linearly combined using weights learned from the training set. While existing object retrieval methods perform well in many cases, they may still fail to return satisfactory results if the ROI (region of interest) specified by the user is inaccurate or if the object captured there is too small to be represented using discriminative features and consequently to be matched with similar objects in the image collection. To improve the object retrieval performance also in these difficult cases, [24] proposes an object retrieval method that exploits the information about the visual context of the query object and employs it to compensate for possible uncertainty in feature-based query object representation. Contextual information is drawn from the visual elements surrounding the query object in the query image, and the ROI is considered as an uncertain observation of the latent search intent and the saliency map detected for the query image as a prior. Comparing with traditional methods applied to extract distinctive invariant features from images, deep learning method is extremely popular and well developed in recent years, such as SSD and MXNET. Based on [22], SSD discretizes the output space of bounding boxes into a set of default boxes over different aspect ratios and scales per feature map location. At prediction time, the network generates scores for the presence of each concept category in each default box and produces adjustments to the box to better match the object shape. Additionally, the network combines predictions from multiple feature maps with different resolutions to naturally handle objects of various sizes. SSD is simple relative to methods that require object proposals because it eliminates proposal generation and subsequent pixel or feature re-sampling stages and encapsulates all computation in a single network. This makes SSD easy to train and straightforward to integrate into systems that require a detection component. Experimental results on the *PASCALVOC*, *COCO*, and *ILSVRC* dataset confirm that SSD has competitive accuracy to methods that utilize an additional object proposal step and is much faster.

Since concepts in images can contribute a lot to content-based geolocation estimation, recently much work has been done in this direction, for example, realizing geolocation estimation by applying concepts classification. According to [25], a novel approach called Semantic Concept Mapping (SCM) is used to classify entities occurring in the text to a custom-defined set of concepts, which is applied to efficiently exploiting free-text annotations as a complementary resource to image classification. While [18] focuses on finding out the concept categories depicted in a set of unlabeled images by using a model developed in the statistical text literature: probabilistic Latent Semantic Analysis (pLSA). The concepts categories are treated as topics, so that an image containing instances of several categories is modeled as a mixture of topics. Comparing with [25] and [18], the accuracy of visual concept classifier can also be improved by combing social data and low-level content-based descriptors according to [26].

Concepts detection can also be realized using learning-based approach, [20] successfully develop such kind of approach by making use of a sparse, part-based representation in still, gray-scale images. A vocabulary of distinctive object parts is automatically constructed from a set of sample images of the object class of interest, images are then represented using parts from this vocabulary, together with spatial relations observed among the parts.

The state of the art in visual object retrieval from large databases is achieved by systems that are inspired by text retrieval. A key component of these approaches is that local regions of images are characterized using high-dimensional descriptors which are then mapped to visual words selected from a discrete vocabulary, such as in [27]. [28] explores techniques to map each visual region to a weighted set of words, allowing the inclusion of features which were lost in the quantization stage of previous systems. The set of visual words is obtained by selecting words based on proximity in descriptor space, then describe how this representation may be incorporated into a standard *tf-idf* architecture, and how spatial verification is modified in the case of this soft-assignment. On the other hand, [29] believes that the performance of *Bag-of-Words (BoW)* features in semantic concept detection for large-scale multimedia databases is subject to various representation choices. Thus, it conducts a comprehensive study on the representation choices of *BoW*, including vocabulary size, weighting scheme, stop word removal, feature selection, spatial information, and visual bi-gram. This paper also offers practical insights in how to optimize the performance of *BoW* by choosing appropriate representation choices. Sometimes concepts detection and quantization are noisy processes and this can result in variation in the particular visual words that appear in different images of the same object, leading to missed results.

2.4 Index Reduction

In recent years, large-scale image retrieval for location estimation has shown remarkable potential in real-life applications. The current applications are mostly based on the foundation of extracting features from query images as the retrieval index and then finding the best matched

images in database, for example, in [1] and [2]. Thanks to the recent development of distributed storage and processing algorithms such as Hadoop, these methods usually build based on training large amount of dataset (over millions of images) and only consider about improving the performance of geolocation estimation as far as possible without considering the computation complexity. However, they will be no longer suitable in two specific situations: (i), when the dataset grows to hundreds of millions, which is far beyond the capability of current algorithms. (ii), when executing such a geolocation estimation task on computational limited devices. In these two specific conditions, it will become crucial and necessary to develop corresponding algorithms to reduce the retrieval index and make the computation process more affordable, which is also the goal of this thesis.

In content-based image retrieval process for geolocation estimation, [3] and [23] both hold the opinion that not all features are necessary for image retrieval. That is, distinctive features have stronger discrimination power than commonly observed features. An importance measure representing both robustness and distinctiveness of a local feature based on diverse density is presented by [23], which claims that the number of local features related to each database entry can be reduced. In [3], a new approach is proposed and named *DVEM*, which uses representations that are specific to the query image whose location is being predicted. These representations are based on visual element clouds, which robustly capture the connection between the query and visual evidence from candidate locations. Then the influence of visual elements that are geo-distinctive is maximized because they do not occur in images taken at many other locations. Classification is often performed based on data from measurements or ratings of objects or events. These data are called features or retrieval index. Since the number of training samples needed to design a classifier grows with the dimension of the features, a way to reduce the dimension of the features without losing any essential information is needed. [30] presents a new method, *linear transformations*, for feature reduction and claims that their method have a more stable and predictable performance than other methods.

To decrease the computational time, a new strategy *Quasi-Gabor filter*, is presented to extract an image concepts with high retrieval accuracy in [31]. Then this paper also proposes how to reduce the image feature dimension using the reward-punishment algorithm, so any robust indexing methods can be used. Since content-based visual matching can also be applied for fast concepts detection in video images, thus it is meaningful to also have a look about concepts detection work for video. A two-step method to speed-up concept detection systems in computer vision that use *Support Vector Machine* as classifiers is proposed in [32]. In the first step, feature reduction is performed by choosing relevant image features according to a measure derived from statistical learning theory. In the second step a hierarchy of classifiers is built. [33] presents an application of rough sets and statistical methods to index reduction and pattern recognition. The presented description of rough sets theory emphasizes the role of rough sets reduction in feature selection and data reduction in pattern recognition. The overview of methods of feature

selection emphasizes feature selection criteria, including rough set-based methods. This paper also contains a description of the algorithm for feature selection and reduction based on the rough sets method proposed jointly with Principal Component Analysis.

To reduce retrieval time as the database being searched may contain thousands of images, Inverted Indexing is the basic technique, given images are represented by Bag-of-Words model. However, one major limitation of both standard Inverted Index and Bag-of-Words model is that they ignore spatial information of the visual words in images. This might reduce retrieval accuracy. In [34], the author introduces an approach to integrate spatial information into inverted index to improve accuracy while maintaining short retrieval time. To solve the problem that inverted index requires gigabytes of memory, which significantly slows down the database server. Authors in [35] develops and compares techniques for inverted index compression for image-based retrieval. The work proposed in this paper includes fast decoding methods, an off-line database reordering scheme that exploits the similarity between images for additional memory savings, and a generalized coding scheme for soft-binned feature descriptor histograms. The experiment results in his work indicates that these techniques significantly reduce memory usage, by as much as 5 times, without a loss in recognition accuracy. In web-scale image retrieval, the most effective strategy is to aggregate local descriptors into a high dimensionality signature and then reduce it to a small dimensionality. However, the computation of this index has a very high complexity, because of the high dimensionality of signature projectors. [36] proposes a new efficient method to greatly reduce the signature dimensionality with low computational and storage costs by applying the linear projection of the signature onto a small subspace using a sparse projection matrix.

CHAPTER 3

CCR FRAMEWORK

In this chapter, we introduce our experiment frameworks with more details, because re-implementing *'Large Scale Image Retrieval for Location Estimation'* pipeline and realizing our proposed method CCR are extremely complex and a lot of corresponding techniques and frameworks are needed (e.g., SSD deep learning framework, Hadoop, MapReduce, etc.). Due to the large size of our San Francisco dataset, which is over 1.06 million images, we need to deal with the dataset not only on local PC, but also on remote server as well. First of all, we need to set up the necessary environment on remote sever (SURFsara) to apply the SSD deep learning framework, including the NVIDIA driver, CUDA, and CUDNN. Then, after environment setting, we build our own SSD deep learning framework and store the common concept detection result as *JSON* format. Next, we retrieve the detection result from server to local PC and transfer the detected common concept areas in our dataset to transparent regions by adopting the *Python Imaging Library*. After the dataset with common concepts removed is ready, we re-implement *'Large Scale Image Retrieval for Location Estimation'* pipeline and wrap the dataset as MapFile, set up *Hathi Client* on local PC, and submit the Hadoop job to SURFsara cluster.

Our work in this thesis mainly consist of 3 parts, Common Concept Removal, 1-NN GeoLocation Prediction, and *'Large Scale Image Retrieval for Location Estimation'* pipeline. Thus, to show our experiment framework structure clearer, we describe all the related techniques and frameworks based on these 3 experimental sections.

3.1 Common Concept Removal

Common Concept Removal is the fundamental algorithm of our experiment in this thesis to reduce the index size and improve the geolocation prediction performance. We believe that in

the process of geolocation prediction, a lot of common concepts (e.g., cars, persons, buses, plants, etc.) cannot contribute to the prediction performance, and can even harm the prediction result. Because these common concepts look similar in street view images of a certain region, like San Francisco, they are not unique neither representative to help predict the geolocation. Thus, by removing these common concepts from query and training images, the index size can be reduced and the geolocation prediction performance can be considerably improved.

3.1.1 SSD Deep Learning Framework

Thanks to the recent development of deep learning technology in object detection direction, a lot of corresponding frameworks have been developed, which increase both the accuracy and efficiency in the process of object detection. One of the most representative deep learning framework for object detection is proposed in 2015 [22]. [22] presents the first deep network based object detector that does not re-sample pixels or features for bounding box hypotheses and is as accurate as approaches that do. By eliminating bounding box proposals and the subsequent pixel or feature re-sampling stage, the speed is greatly improved.

The SSD approach is based on a feed-forward convolutional network that produces a fixed-size collection of bounding boxes and scores for the presence of object class instances in those boxes, followed by a non-maximum suppression step to produce the final detections. The early network layers are based on a standard architecture used for high quality image classification, which is called as the base network. Then Multi-scale feature maps and convolutional predictors are added for the detection work. The comparison between two single shot detection models: SSD and YOLO is shown in Fig. 3.1.

The key difference between training SSD and training a typical detector that uses region proposals, is that ground truth information needs to be assigned to specific outputs in the fixed set of detector outputs. During the training process, the SSD framework needs to decide which default boxes corresponding to a ground truth detection and train the network accordingly. Then by matching each ground truth box to the default box with the best *jaccard* overlap, the network is able to predict high scores for multiple overlapping default boxes. Instead of processing the image at different sizes and combining the results afterwards for the purpose of handling the different object scales, this framework utilizes feature maps from several different layers in a single network for prediction. Finally, to make the model more robust to various input object sizes and shapes, each training image is randomly sampled by using the entire original input image or randomly sample a patch.

SSD is not only highly efficient, but also an accurate deep learning framework for object-detection. Once set up the environment and train the model, the object detection speed can reach as fast as 0.1s/image, and almost all objects in query images can be detected as long as the object visualize score threshold and the non-maximum suppression threshold are correctly set. By applying SSD framework on remote server SURFsara, the detection example is shown in Fig.

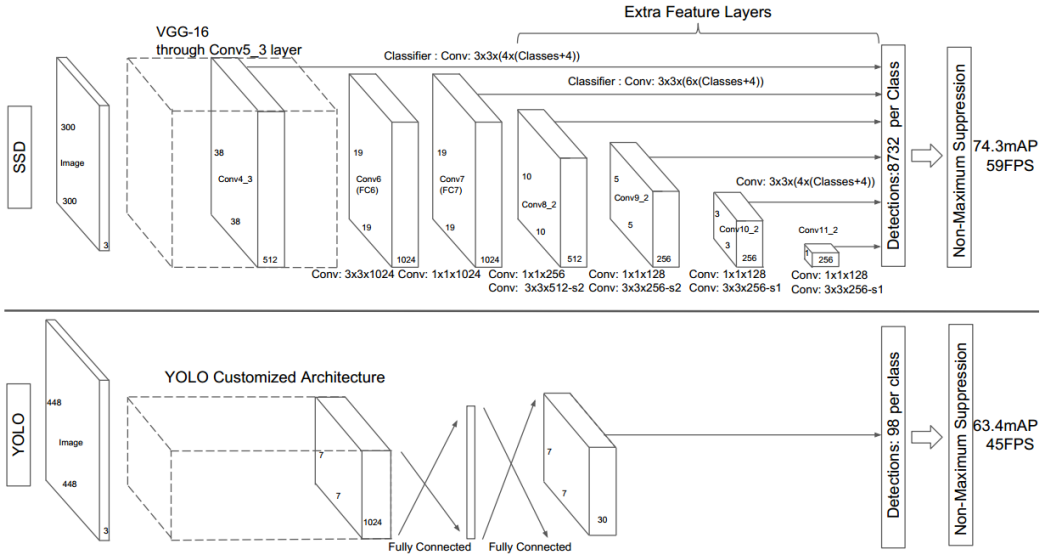

FIGURE 3.1. A comparison between two single shot detection models: SSD and YOLO [22]

Modes	Time Efficiency
CPU	0.541 seconds/image
GPU	0.112 seconds/image

Table 3.1: *SSD* deep learning framework behavior comparing under CPU and GPU mode

3.2

SSD is both applicable for running on CPU mode and GPU mode, however, our experiments shows that their detection efficiency are greatly different. We test detecting objects on 100 images for both CPU mode and GPU mode, the average detection speed results is shown in Table. 3.1. It is obvious that the detecting speed is insanely slow under CPU mode and much faster under GPU mode due to the fact of adopting both *CUDA*, a parallel computing platform and programming model invented by *NVIDIA*, and *CUDNN*, a GPU-accelerated library of primitives for deep neural networks. The GPU we adopt to run SSD framework on SURFsara is 'GRID K2', which has 4GB memory. For the purpose of making this GPU functional, we set the NVIDA driver version as 367.57, adopt the CUDA 8.0 and CUDNN v5. By applying the GPU mode on remote server SURFsara, the total time consuming for detecting 1.06 million images from San Francisco dataset is approximately 36 hours.

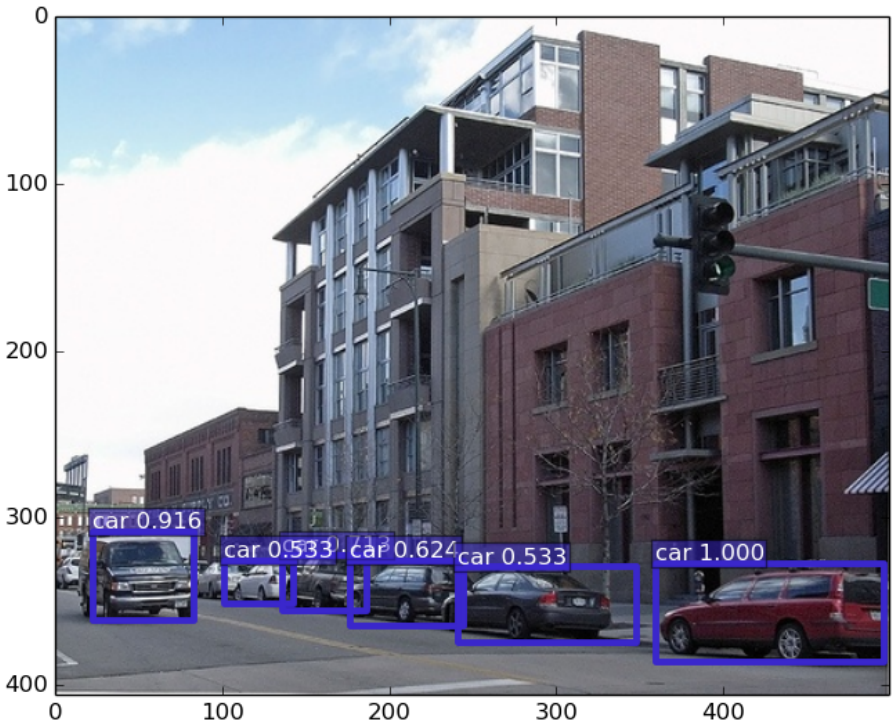

FIGURE 3.2. *SSD* deep learning framework detection result

3.1.2 Python Imaging Library (PIL)

After the detection process on remote server by adopting SSD deep learning framework, we apply *Python Imaging Library* to read the dataset images and transfer the detected areas to transparent. The *Python Imaging Library* adds image processing capabilities to the Python interpreter. This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities. The core image library is designed for fast access to data stored in a few basic pixel formats. It provides a solid foundation for a general image processing tool. Meanwhile, the PIL contains basic image processing functionality, including point operations, filtering with a set of built-in convolution kernels, and color space conversions. The library also supports image resizing, rotation and arbitrary affine transforms.

To be more specifically, by applying the PIL model, we read pixel by pixel in each image as RGBA format, which stands for red green blue and alpha. The alpha channel is normally used as an opacity channel. If a pixel has a value of 0% in its alpha channel, it is fully transparent, whereas a value of 100% in the alpha channel gives a fully opaque pixel. In this thesis, all the pixels in common concept areas are given the value of $(255, 255, 255, 0)$, in which $(255, 255, 255)$ represents the white and alpha number 0 represents the transparent. Finally, we save the

common concepts removed images as tar files and wrap them as MapFile by applying *'Large Scale Image Retrieval for Location Estimation'* pipeline.

3.2 1-Nearest Neighbor GeoLocation Prediction Classifier

Before formally test our proposed CCR algorithm in *'Large Scale Image Retrieval for Location Estimation'*, we firstly re-build the 1-NN classifier to predict the geolocation based only on visual similarity. The implementation details is described in Sec. 4.2.2. In this chapter, we briefly introduce one of the most important algorithms SIFT, which is applied in 1-NN geolocation prediction classifier.

3.2.1 Scale-Invariant Feature Transform Algorithm

Matching features across different images is a common problem in computer vision field. The scale-invariant feature transform (SIFT) is such kind of algorithm to detect and describe local features in images, which is firstly propose by *David Lowe* in 1999 [37]. There are mainly several steps in SIFT and we will briefly introduce them below.

Since we cannot use the same window to detect the key points with different scale, the scale-space filtering is needed. Thus, *LoG (Laplacian of Gaussian)* is applied in SIFT as a blob detector which detects blobs in various sizes due to change in various values σ. However, the *LoG* is costly, so SIFT algorithm uses *DoG (Difference of Gaussians)* which is an approximation of *LoG*. Difference of Gaussian is obtained as the difference of Gaussian blurring of an image with two different σ. Once the *DoG* are found, images are searched for local extrema over scale and space. For example, one pixel in an image is compared with its 8 neighbors as well as 9 pixels in next scale and 9 pixels in previous scales. If it is a local extrema, it is a potential key point, which basically means that key point is best represented in that scale. The basic structure is shown in Fig. 3.3

Once potential key points locations are found, Taylor series expansion of scale space is then adopted to get more accurate location of extrema, and if the intensity at this extrema is less than a threshold value, it is rejected. Then the edges will be removed because *DoG* has higher response for edges.

Since an orientation is assigned to each key point to achieve invariance to image rotation, then a neighborhood is taken around the key point location depending on the scale, and the gradient magnitude and direction is calculated in that region. Then key point descriptor is created, a 16×16 neighborhood around the key point is taken and divided into 16 sub-blocks of 4×4 size. For each sub-block, 8 bin orientation histogram is created, so a total of 128 bin values are available. In addition, several measures are taken to achieve robustness against illumination changes, rotation, etc. Finally, key points between two images are matched by identifying their nearest neighbors.

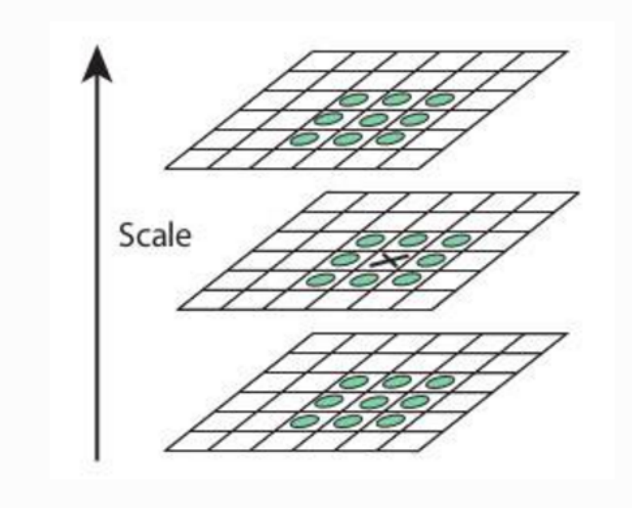

FIGURE 3.3. Structure of images searching for local extrema over scale and space

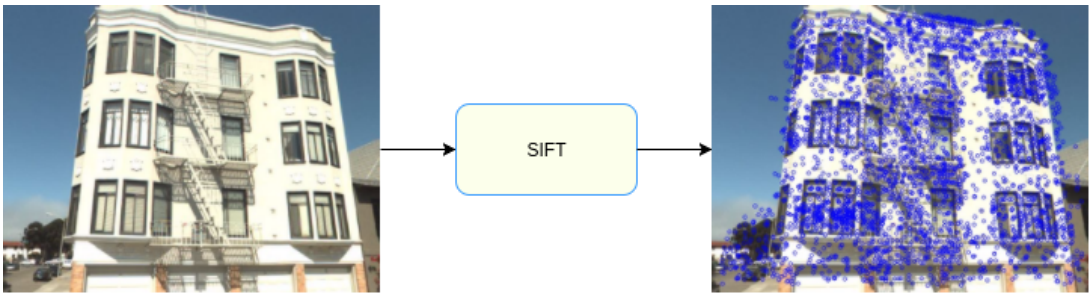

FIGURE 3.4. *SIFT* salient points detection result by adopting *OpenCV*

For the purpose of finding out the behavior of SIFT, we have tested images from San Francisco using *OpenCV*. The detection result is shown in Fig. 3.4, all the detected salient points are represented as small blue circles in the right image.

Then we are also curious to find out if it is possible to find the best matched candidate image only based on the detected salient points. For example, in Fig. 3.5, the left one is the query image and the right one is one of the selected matching images. We firstly hold the opinion that the fire stair is an obvious and representative concept, which can help the query image find the best matched candidate image. However, the experiment result in Fig. 3.5 clearly illustrates that the right image is matched to the left one not only because of the fire stair, but also because of the edges in the bottom. Thus, we draw the conclusion that only counting on SIFT algorithm to find the best matched candidate images is almost impossible in the process of image retrieval. This experiment helps us better understand the SIFT descriptor, which contributes in [1] to find and describe invariant regions in image.

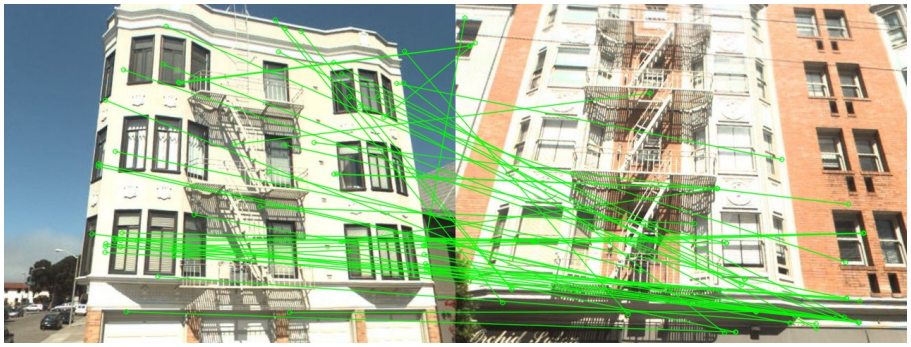

FIGURE 3.5. Image matching based on *SIFT* salient points detection

3.3 Large Scale Image Retrieval for Location Estimation

'Large Scale Image Retrieval for Location Estimation' pipeline is built on Eclipse platform and written in Java, the experimental results are very impressive, especially for the proposed method GVR (Geo Visual Ranking) and DEVM (Geo-Distinctive Visual Element Matching), which have been proved that significantly outperform the other reference methods, such as [6, 38]. In the process of re-implementing this pipeline, we firstly need to write our dataset in MapFile format and submit it to the SURFsara cluster, then all the corresponding code need to be generated as jar files from Eclipse platform and submit the job to the yarn framework. In this section, we briefly introduce the Hadoop, MapReduce, and Hadoop Distributed File System (HDFS) for the purpose of better understanding the whole structure of *'Large Scale Image Retrieval for Location Estimation'* pipeline.

3.3.1 Hadoop

Hadoop is an Apache open source framework written in Java that allows distributed processing of large dataset across clusters of computers using simple programming models. A Hadoop frameworked application works in an environment that provides distributed storage and computation across clusters of computers. Hadoop is designed to scale up from single server to thousands of machines, each offering local computation and storage. Hadoop framework includes following four modules:

- Hadoop Common: These are Java libraries and utilities required by other Hadoop modules. These libraries provide file system and OS level abstractions and contain the necessary Java files and scripts required to start Hadoop.
- Hadoop YARN: A framework for job scheduling and cluster resource management.
- Hadoop Distributed File System (HDFS): A distributed file system that provides high-throughput access to application data.

- Hadoop MapReduce: This is YARN-based system for parallel processing of large dataset.

There are many advantages of Hadoop, which convince us that it is a good choice to realize the large scale image retrieval for location estimation. Firstly, Hadoop framework allows the user to quickly write and test distributed systems. It is efficient, and it automatically distribute the data and work across the machines and in turn, utilizes the underlying parallelism of the CPU cores. Secondly, Hadoop does not rely on hardware to provide fault-tolerance and high availability (FTHA), rather Hadoop library itself has been designed to detect and handle failures at the application layer. Then, servers can be added or removed from the cluster dynamically and Hadoop continues to operate without interruption. Apart from being open source, it is compatible on all the platforms since it is Java based.

3.3.2 MapReduce

Hadoop MapReduce is a software framework for easily writing applications which process big amounts of data in-parallel on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner. The term MapReduce actually refers to the following two different tasks that Hadoop programs perform:

- The Map Task: This is the first task, which takes input data and converts it into a set of data, where individual elements are broken down into tuples (key/value pairs).
- The Reduce Task: This task takes the output from a map task as input and combines those data tuples into a smaller set of tuples. The reduce task is always performed after the map task.

Typically, both the input and the output are stored in a file-system. The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks. The MapReduce framework consists of a single master JobTracker and one slave TaskTracker per cluster-node. The master is responsible for resource management, tracking resource consumption/availability and scheduling the jobs component tasks on the slaves, monitoring them and re-executing the failed tasks. The slaves TaskTracker execute the tasks as directed by the master and provide task-status information to the master periodically. The JobTracker is a single point of failure for the Hadoop MapReduce service which means if JobTracker goes down, all running jobs are halted.

3.3.3 Hadoop Distributed File System (HDFS)

Hadoop can work directly with any mountable distributed file system such as Local FS, HFTP FS, and others, but the most common file system used by Hadoop is the Hadoop Distributed File System (HDFS). The Hadoop Distributed File System (HDFS) is based on the Google File System

(GFS) and provides a distributed file system that is designed to run on large clusters (thousands of computers) of small computer machines in a reliable, fault-tolerant manner. HDFS uses a master-slave architecture, where master consists of a single NameNode that manages the file system metadata and one or more slave DataNodes that store the actual data. A file in an HDFS namespace is split into several blocks and those blocks are stored in a set of DataNodes. The NameNode determines the mapping of blocks to the DataNodes. The DataNodes takes care of reading and writing operation with the file system. They also take care of block creation, deletion and replication based on instruction given by NameNode. HDFS provides a shell like any other file system and a list of commands are available to interact with the file system.

3.4 Conclusion

In this chapter, we briefly introduce the related techniques and frameworks, which are crucial in the process of realizing our propose approach CCR. The work in this thesis was represented as three parts, Common Concept Removal, 1-NN Geolocation Prediction Classifier, and *Large Scale Image Retrieval for Location Estimation* pipeline. In **Sec. 3.1**, we introduced more implementation details of SSD deep learning framework, including network of SSD, hardware environment on remote server, and brief detection speed comparing. Meanwhile, the Python Imaging Library (PIL) was introduced and the RGBA value we have adopted is indicated. In **Sec. 3.2**, we focused on introducing the scale-invariant feature transform algorithm, because we consider this algorithm as the fundamental of implementing the 1-NN geolocation prediction classifier. Finally, in **Sec. 3.3**, we briefly introduce the Hadoop and other related techniques, which were built on SURFsara cluster and help us realize the geolocation prediction process in acceptable time.

CHAPTER 4

CCR EXPERIMENT RESULT

Through rapid development and widespread usage of capture devices such as cameras, phones and tablets, generation of social images in recent years has exploded. Although smart phones nowadays make it possible and easy to tag geo-coordinates to images during capturing process, most social images are still uploaded without such kind of geo information. To solve this problem, increasing research attention has been devoted to techniques that can automatically estimate geolocations of social images. Such approaches are commonly referred to as geolocation prediction techniques. For example, one popular method is called textual metadata, which often accompany social images may include place names and other location-specific terms and in this way help inform the geolocation prediction process. However, the drawback of textual annotation is that it needs to be manually created firstly by the users, which is hard to guarantee that useful location-related information is provided. Other possible approaches rely on the content of images. Images taken at a location demonstrate a high degree of visual variability. The challenge addressing this problem is shown in Fig. 4.1. Thus, it is not surprising that the majority of such approaches usually narrow the domain of prediction and tackle the task within a geographically constrained area.

Currently, related research for geolocation estimation based on content-based-image-retrieval mostly focuses on improving the prediction accuracy as far as possible (e.g., in recent content-based geolocation estimation work [1, 3, 16]), but few of them concentrate on solving the problem of decreasing the retrieval index, which could make it possible to run the geolocation estimation process on low computation capability platforms.

In this chapter, we test our proposed method Common Concept Removal, which is adopted in this thesis to reduce the retrieval index size and improve the geolocation prediction performance in the process of content-based geolocation estimation. Our proposed approach is tested in geo-

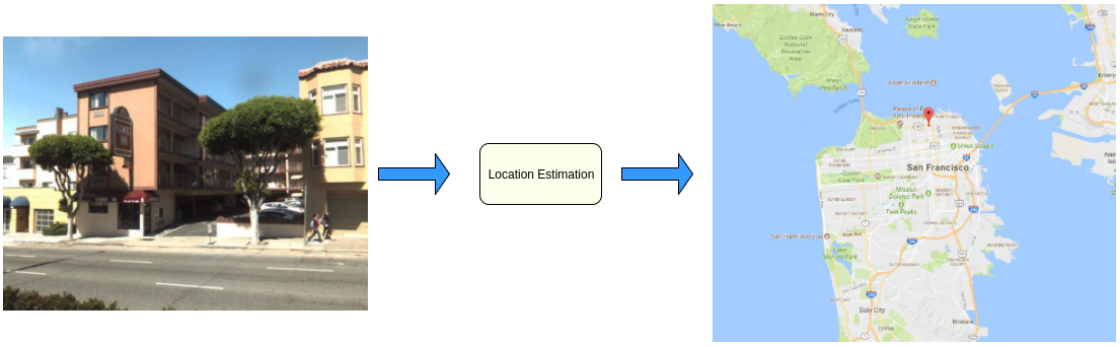

FIGURE 4.1. Estimate the geolocation of an image solely using its visual content

constrained area *San Francisco*, and the street view images from *Stanford Digital Repository* are chosen as the dataset for our experiment.

'Large Scale Image Retrieval for Location Estimation' ([1–3]) has shown that the proposed methods can successfully improve the content-based geolocation estimation behavior comparing to the state of the art in search-based approaches. This research also claims in [1] that in practice, 1-NN approach [17], which uses the location of the image visually most similar to the query image as the predicted location is difficult to beat. Thus, in addition to rebuilding *'Large Scale Image Retrieval for Location Estimation'*, we also choose to re-implement the 1-NN classifier in this chapter, which can be adopted as the framework to realize the content-based geolocation prediction for the purpose of testing the behavior of our propose method in this thesis. Moreover, comparing with *'Large Scale Image Retrieval for Location Estimation'*, which is built based on Hadoop and MapReduce, the 1-NN classifier is less complicated to realize and helpful to make our experiment results more reliable.

4.1 CCR Design

With the well development of mobile picture capturing devices, the number of images all around the world is exploding and most of them are still untagged with geo-information. Thus, analyzing the content of such huge number of images and making it possible to predict their geolocation has kept drawing increasing attention in recent years. There is plenty of corresponding work in this direction, such as [1, 3, 7], e.g. However, most of the work focuses mainly on solving the problem of geo-prediction accuracy and building their work based on high-level computational devices such as remote server and distributed systems such as Hadoop. Currently, the object-based geolocation estimation systems mainly consist of two steps: collect the candidate ranking images based on visual similarity (shown in Fig. 4.2) and re-rank the candidates by applying geometric constraints to assess the reliability of visual correspondences between images.

Although the spatial verification stage is the key to achieve a high precision for object-based image retrieval, the initial ranking stage is also crucial to reduce the index and retrieve the

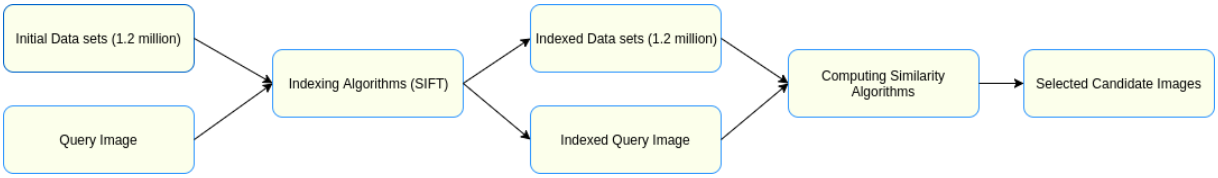

FIGURE 4.2. Initial image retrieval candidate selection process

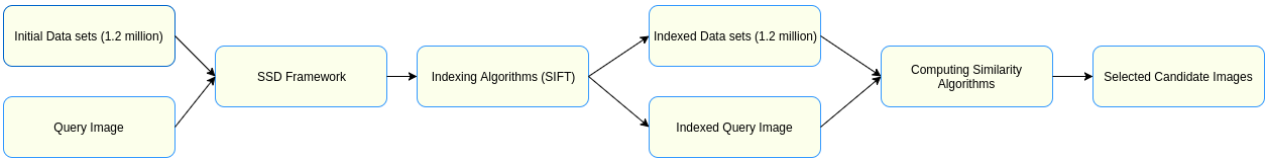

FIGURE 4.3. Image retrieval candidate selection with *SSD*

candidate corresponding images much faster. Inspire much by [1, 22], firstly we build the deep learning framework based on SSD and detect the manually select common concepts in San Francisco dataset, such as *car, person, bus, plant*, get their rectangular coordinates, class names, and store them in database. Then, we extract the features from both training dataset and query images, but avoid extracting the detected common areas by transferring the detected common areas to transparent. We organize the candidate ranking and collection pipeline as illustrated in Fig. 4.3. We firmly believe that our proposed algorithm can greatly reduce the index size in the image retrieval process and can contribute to retrieve the corresponding images more accurately.

4.1.1 Experiment Setup

Before re-implementing 1-NN geolocation prediction classifier and *'Large Scale Image Retrieval for Location Estimation'*, we firstly need to detect the manually defined common concepts by adopting the SSD deep learning framework. The SSD can be both implement on CPU mode and GPU mode, however, the experiment result shows that the detection speed of GPU with CUDA and CUDNN is almost 5 times faster than CPU. Limited by the computational ability of local PC and 1.06 million images dataset, it is almost impossible to implement the SSD locally. By adopting SURFsara, we can work on the necessary hardware for running the SSD. SURFsara offers a NVIDIA GPU-Grid K2, with 4GB memory, which is capable to detect the 1.06M images in approximately 36 hours. We adopt the pre-trained mode Resnet-50, which is trained based on VOC07+12 dataset. The SSD approach is based on a feed-forward convolutional network that produces a fixed-size collection of bounding boxes and scores for the presence of object class instances in those boxes, followed by a non-maximum suppression step to produce the final detections (for bounding boxes with most overlap keep the one with highest score). Thus, to

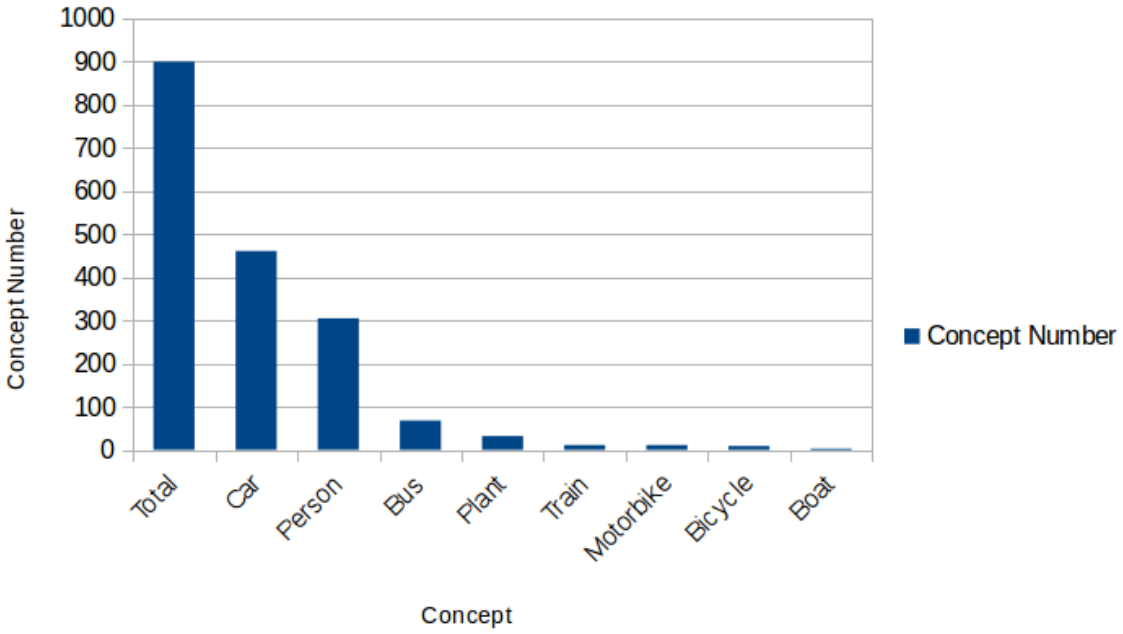

FIGURE 4.4. 803 query images detection result distribution

ensure the common concept detection accuracy and avoid much mis-detection, the parameter of object visualize score threshold is set as 0.8 and the non-maximum suppression threshold is set as 0.5 for both the 1-NN geolocation estimation classifier and *Large Scale Image Retrieval for Location Estimation* pipeline.

4.1.2 Experiment Result and Evaluation

After the necessary environment settings, we apply the SSD framework to detect the common concepts from 803 query images, the detection distribution is shown in Fig. 4.4. By setting object visualize score threshold as 0.8, totally, we successfully detect 900 objects from 803 query images, including 461 cars, 305 persons, 68 buses, 32 plants, 11 trains, 11 motorbikes, 9 bicycles, and 3 boats. Are all the common concepts correctly detected? To answer this question, we decide to evaluate the SSD detection results manually one by one because evaluation time is affordable for 803 query images. The total detection number of common concepts and the correctly detected common concepts number is calculated and compared in Fig. 4.5. Totally, 814 concepts are correctly detected, comparing with the initially detected 900 concepts, the SSD common concept detection accuracy is 90.4%. Thus, **RQ5** is answered, the SSD detection results are successfully evaluated manually.

To be more intuitive and convincing, the example of initial query images and the concept removed query images are compared in Fig. 4.6. As mentioned before, we set the object visualize

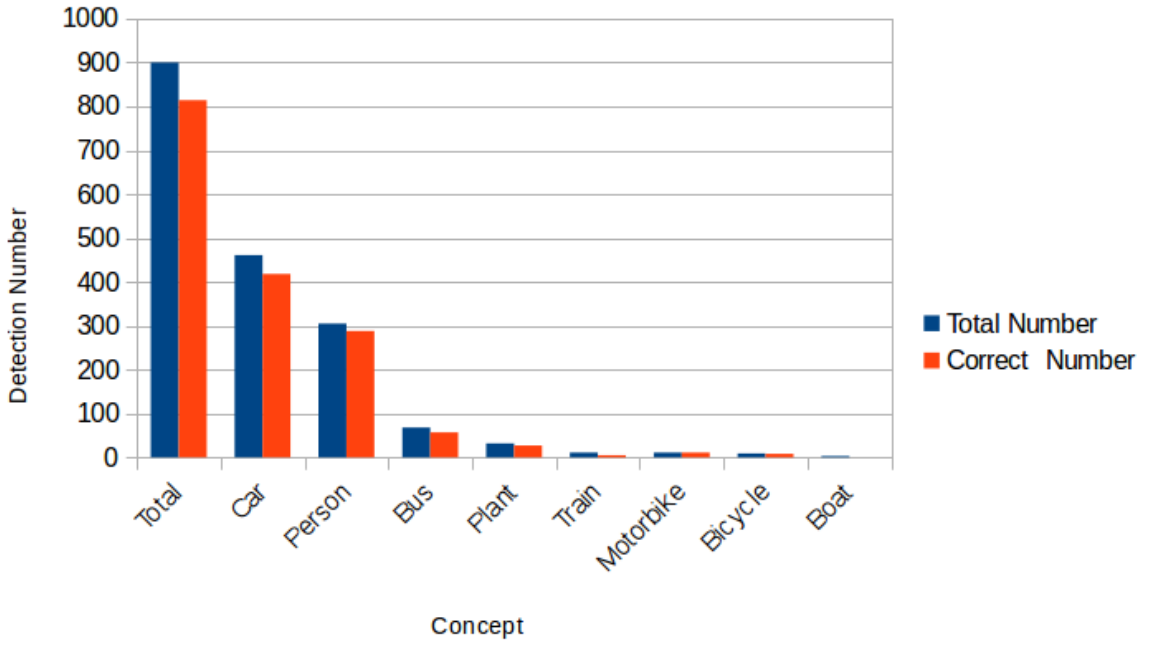

FIGURE 4.5. 803 query images evaluation result

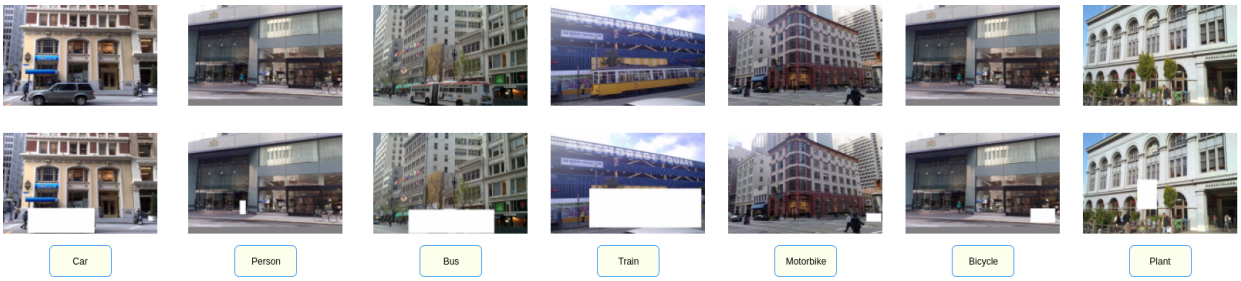

FIGURE 4.6. 803 query images detection and removal results

score threshold as 0.8 to avoid mis-detection and sacrificing too much prediction accuracy. Thus, it is easy to understand that not all objects in an image can be detected with such a high threshold, for example, the detection result of plant in Fig. 4.6. There are several plants exist in front of the building, but only one of them is successfully detected and removed.

4.2 1-Nearest Neighbor GeoLocation Prediction Classifier

4.2.1 Details of 1-NN GeoLocation Prediction Classifier

There are plenty of related work focusing on dealing with image classification and retrieval jobs by using different classifiers, such as *SVM*, *K-NN*, *1-NN*. For instance, [39–41] all describe the usage of *Support Vector Machine* classifier in content-based image retrieval process. Although *SVM* is popular in dealing with image classification and retrieval tasks, *1-NN* also performs some good properties, which can help outperform *SVM* in some special situations.

In this thesis, we firstly choose to apply *1-NN Classifier* for geolocation estimation and test our proposed approach-Common Concept Removal. The 1-NN classifier is also built in *'Large Scale Image Retrieval for Location Estimation'* and mostly inspired by the work of [17], which proposes a simple algorithm of *1-NN Classifier* for estimating a distribution over geographic locations from a single image using purely data-driven scene matching approach.

In [17], the authors consider the text labeled image is ambiguous and the geo-located images are most likely to be visually irrelevant. Thus, for the purpose of looking for a large number of useful images, the authors propose an idea to increase the likelihood of finding accurately geo-located and visually useful data by taking the intersection of groups, images with both GPS coordinates and geographic keywords. Meanwhile, the images tagged with labels like 'birthday', 'concert', 'abstract' and 'camera phone' are excluded from the selected dataset. However, in our thesis, we do not need to worry about the quality of dataset, because we only test our proposed common concept removal methods in a certain city San Francisco. All the 1.06M training images and the 803 query images are formulated as the same size 480×640 from [9], and the corresponding geolocation information is included in the name of each image.

Is it feasible to extract the geolocation information only based on the content of images? Humans can easily estimate the location of a given image even if he or she has never seen it before. If we want to assign this task to computers, we will need to extract features from images as representations for the retrieval process. [17] has evaluated and compared an assortment of popular features, such as color histogram, texton histogram, line features, etc. In this section, the same strategy and some of the features are adopted as proposed in [17].

In this section, for comparing the different prediction performance of 1-NN classifier, in addition to applying the 1-NN classifier for the purpose of geolocation estimation, we also apply the *GVR* method that is proposed by Li in [1].

4.2.2 Experiment Setup

In this subsection, we describe the setup of our experimental framework for assessing the performance of the 1-NN classifier method. In the following subsections, we will elaborate on the details regarding all aspects of this framework, including the dataset we used, features we

selected to measure visual similarity of images, reference methods that we deploy for comparative analysis and the assessment criteria, etc.

To assess the performance of the 1-NN classifier method, we carry out experiments on an image collection that is based on the dataset released by [9]. Here, 1.06M images serve as training images, which are taken by a vehicle-mounted camera moving around downtown San Francisco, and 803 images serve as query images, which are taken by various people using a variety of mobile photo-capturing devices.

In the 1-NN classification method, the following features are adopted as described in [17].

- Tiny images: The most trivial way to match scenes is to compare them directly in color image space. Reducing the image dimensions drastically makes this approach more computationally feasible and less sensitive to exact alignment. This method of image matching has been examined thoroughly in [42] for the purpose of object recognition and scene classification.

- Color histogram: A color histogram is a representation of the distribution of colors in an image. In the spirit of most image retrieval literature, the joint histograms of color in CIE L*a*b* color space for each image are built. We have fewer bins in the intensity dimension because other descriptors will measure the intensity distribution of each image. Then the distance between these histograms is computed by using χ^2 distance.

- Line features: The statistics of straight lines in images are also useful for distinguishing between natural and man-made scenes and for finding scenes with similar vanishing points. For each image two histograms are built based on the statistics of detected lines-one with bins corresponding to line angles and one with bins corresponding to line lengths. L1 distance is used to compare these histograms.

We assess the 1-NN classifier in this section through a comparative experiment analysis, which we perform in comparing with the GVR approach. The GVR approach consists of 3 main steps [1]. In the first, candidate image selection step, for a given query image, retrieve from the collection of geo-tagged images as a ranked list C of candidate images that are most visually similar to the query. Then, in the location extraction step, based on geo-distribution of these candidate images, candidate locations are extracted that form the set G. Each location from G is represented by images from the list C that form the corresponding location cluster. Finally, parameter 'score' is modeled by the visual proximity between the sets C and the query q and is used to rank the candidate locations for the purpose of selecting the most likely one to be adopted for the query image. This last step is referred to as location ranking.

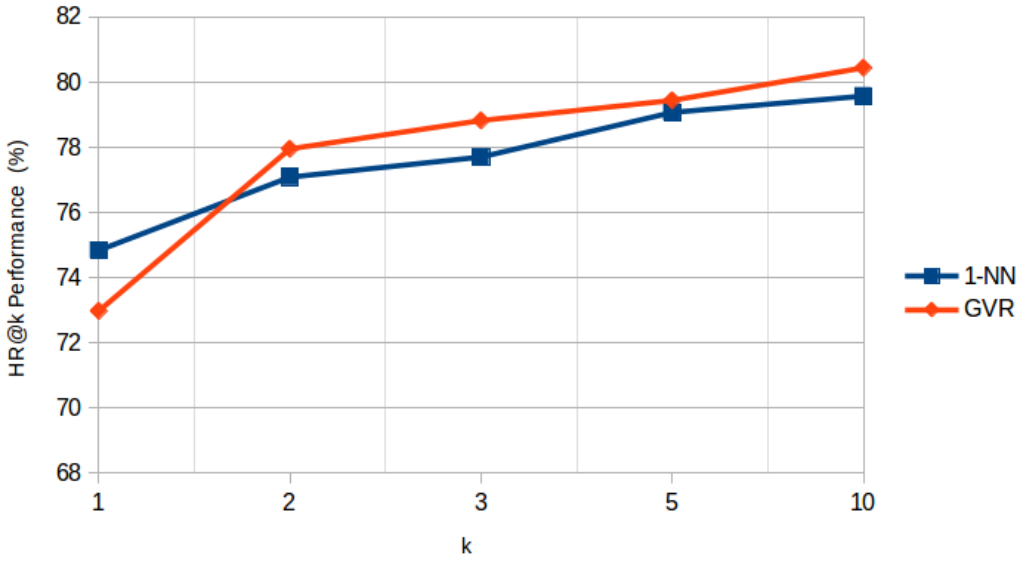

FIGURE 4.7. Performance comparing between 1-NN and GVR

4.2.3 Experiment Result

In this subsection, we introduce our experimental result with respect to two aspects, geolocation prediction performance and index size reduction, by removing the common concepts only from the 803 query images and keep the 1.06M training images initially.

In the beginning of implementing the 1-NN classifier, we are interested in finding out the different performance between the 1-NN classifier and the GVR approach. We adopt the same evaluation method as introduced in section. 4.2.4, same dataset from San Francisco, and set the same evaluation radius as 1km to fairly compare their behavior. As shown in Fig. 4.7, GVR can always outperform the 1-NN classifier with the HR@k evaluation approach adopted except for HR@1.

Does removing the common concepts from 803 query images help improve the geolocation estimation? This is our first research question proposed in Chapter 1. After successfully detecting and removing the common concepts, we wrap the common concepts removed 803 query images and the origin 1.06M training images together as MapFile and submit the job to a Hadoop cluster. For each concept, we need 13 hours to write them as MapFile on local laptop, 3 hours to upload the MapFile (approximately 35.6GB) to the Hadoop cluster, and 4 hours to run the 1-NN geolocation prediction classifier. Since we re-run the 1-NN classifier for 9 times, the total time consuming is approximately $(13+3+4)\times 9 = 180(h)$. Then, the 1-NN geolocation estimation classifier performance with different common concepts is shown in Fig. 4.8. After applying our proposed common concept removal approach, the geo-prediction performance is considerably improved by about 6% for HR@1, HR@2, HR@3, HR@5, HR@10. So, **RQ1** is well answered. What

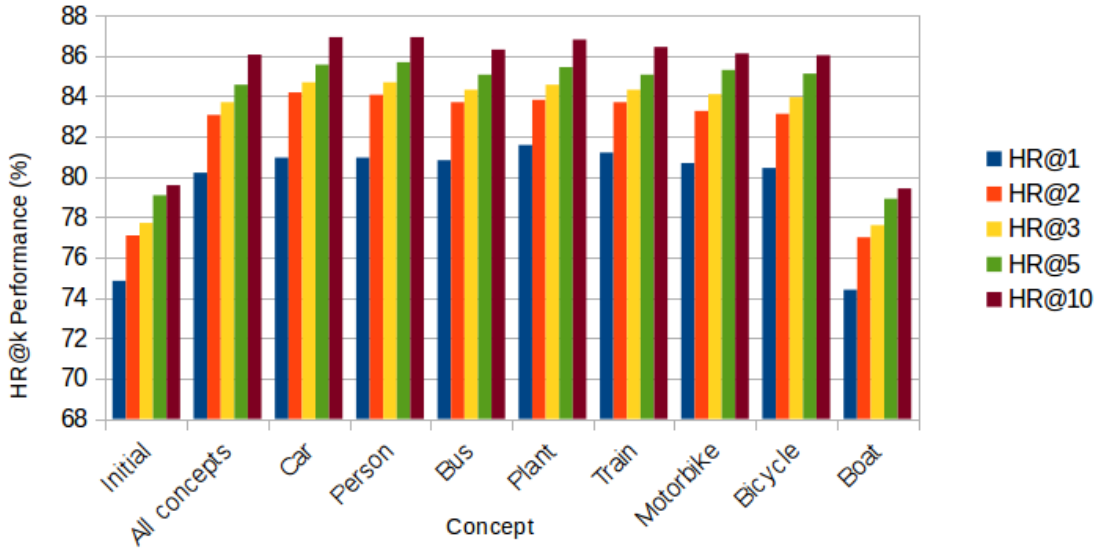

FIGURE 4.8. Common concept removal influence on 1-NN performance

is more, from Fig. 4.8, we can figure out that the 1-NN geolocation prediction performance is largely influenced by the SSD detection performance and the shape of the detected concepts. When the detected common concepts number is high, and the shape of the concepts fit rectangular (because we remove the common concepts in the shape of rectangular), the geolocation prediction performance after removal can beat the performance of initial dataset. On the contrary, when the detected common concepts number is small, and the shape of concepts do not fit the rectangular, for example, concept boat (3 detected and 0 correct), the geolocation prediction performance is almost the same as initial performance without the common concepts removed. Thus, the **RQ4** is answered, the influence of different concepts on location prediction performance has been analyzed.

Then, we aim to find the influence of concepts removed 803 query images on the index size. Even though the index size reduction may be small, because the influence of 803 query images to index size is almost ignorable comparing with the 1.06M training images. Thus, we compare the dataset size, feature size, and index size in Fig. 4.9, with different concepts as the variable. The results proved that, the index size and feature size changing are almost impossible to observe. To further prove that the index size is slightly reduced, we compare the index size changing as bytes in Table. 4.1. From this table, the index size changing is more intuitive, the results indicate that the feature size is slightly reduced, but the index size basically keeps the same for almost all concepts, except for concept bus, which is only 212 bytes reduced.

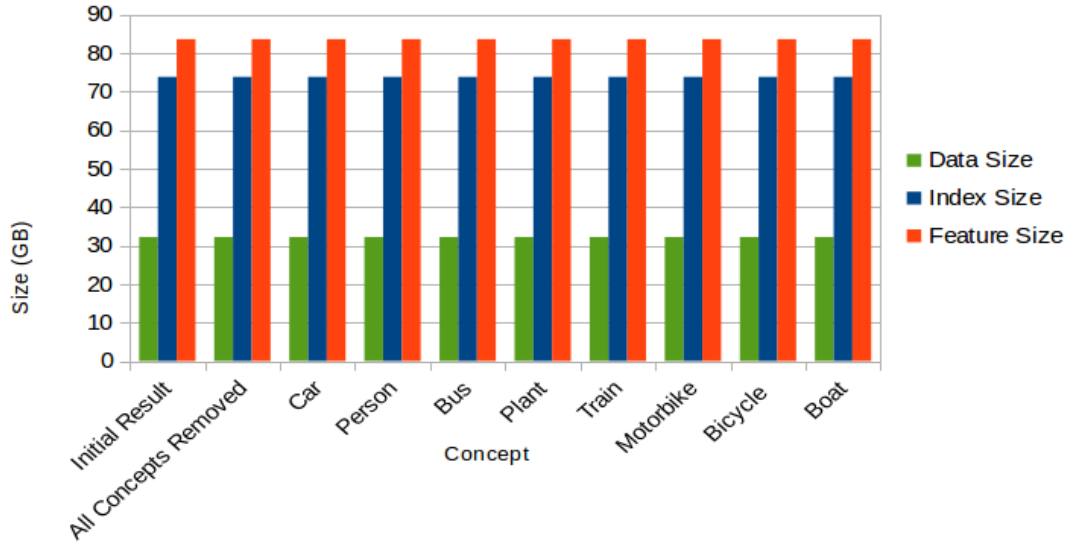

FIGURE 4.9. Influence of query image concept removal on index size

Concepts	MapFile Size (Bytes)	Feature Size (Bytes)	Index Size (Bytes)
No Concepts Removed	34610112070	89756393776	79230407047
Total Concepts	34588180269	89754387646	79230407047
Car	34589273397	89768804495	79230407047
Person	34590008990	89780673029	79230407046
Bus	34589678195	89777875815	79230406835
Plant	34590108401	89784314669	79230407047
Train	34592104283	89795324460	79230405126
Motorbike	34592109348	89798827659	79230407047
Bicycle	34595106421	89795314334	79230407047
Boat	34610112058	89806393744	79230406955

Table 4.1: Index size comparing between initial dataset and concepts removed dataset

4.2.4 Evaluation

To evaluate the performance of the 1-NN classifier for geolocation estimation, we adopt the procedure standardly used in *'Large Scale Image Retrieval for Location Estimation'*. We start by defining an evaluation radius r_{eval}. This radius controls the evaluation precision and the tolerance to data noise in the ground truth, which is generated by a GPS device or through manual labeling. An image is considered to be correctly predicted if its predicted geo-coordinates fall within r_{eval} around the ground truth location. Formally expressed, the correctness of an image with respect to an evaluation radius r_{eval} is calculated by the evaluation function $f_{r_{eval}}$,

$$f_{r_{eval}}(g,\widetilde{g}) = \begin{cases} right, & geoDist(g,\widetilde{g}) \leq r_{eval} \\ wrong, & otherwise \end{cases}$$

where $geoDist(g,\widetilde{g})$ indicates the geographical distance between g and $\widetilde{g}$.

As the same strategy applied in *'Large Scale Image Retrieval for Location Estimation'*, we use the *Hit Rate* at top k (HR@k) as the criterion to assess the quality of prediction. Given a query, the system returns a ranked list of possible locations. Then, HR@k measures the proportion of queries that are correctly located in the top k locations. More specifically, HR@1 represents the ability of the system to output a single accurate prediction.

4.3 Large Scale Image Retrieval for Location Estimation

4.3.1 Details of Large Scale Image Retrieval for Location Estimation

The challenge addressed in *'Large Scale Image Retrieval for Location Estimation'* can be formulated as follows: *given the visual content of an image, determine the geo-coordinates of the location of the depicted scene*. The challenge of this research is substantial due to several reasons. First, one and the same visual scene can be captured under strongly varying conditions determined by the level or type of light, distortions, zoom or occlusion. Second, depending on the capture angle and direction, different scenes can be captured at one and the same location. For instance, standing on a particular spot on a beach, one can take a photo of the sea, but also of the beach or of the street running in parallel with the shore. This means that there is no unique link between the visual scene and a location. Third, the number of different unique scenes and locations worldwide is effectively infinite. Due to these reasons, most of the work in this direction has reported attempts which first make the challenge tractable before performing location estimation. These approaches typically attempt to narrow the domain of estimation and tackling the task in a geo-constrained way. *'Large Scale Image Retrieval for Location Estimation'* is mainly organized into three parts, *Geo Visual Ranking*, *Pairwise Geometric Matching* and *Geo-Distinctive Visual Element Matching*.

In *Geo Visual Ranking*, the problem of location inference from visual content is unraveled, introduce the search-based approach and propose a novel way of implementing it, namely in the form of a Geo-Visual Ranking (GVR) method that considers the ambiguity in how visual content reflects a location. The rationale underlying the *GVR* method is that, comparing to the images from a wrong location, more images from the true location will likely contain more elements of the visual content of the query image. For this reason, *'Large Scale Image Retrieval for Location Estimation'* hypothesizes that the evidence from the set of visually similar images from a wrong location is too weak to compete with the set captured at the true location, independently of the set size. Basically, the *GVR* lead us to focusing on deriving location information from the objects captured in the images, or in other words using the object-based image retrieval approach.

With object-based image retrieval we understand the problem of finding images that contain the same objects or scene elements as in the query image. The object-based image retrieval systems generally consist of two main stages:

- Initial ranking stage, where the ranking of images from the collection is based on visual similarity computed on visual feature statistics measured in different images.
- Spatial verification stage, where the initial ranked list is re-ranked by applying geometric constraints to assess the reliability of visual correspondences between images.

To improve the scalability and robustness of object-based image retrieval in the *GVR* framework, a novel *Pairwise Geometric Matching Method* is proposed for the spatial verification stage. It uses global scale and rotation relations to enforce the local consistency of geometric relations derived from the locations of pairwise correspondences. The results presented for this method indicate the suitability of the proposed pairwise geometric matching method as a solution for large-scale object retrieval at an acceptable computational cost.

Since some objects may be common to different visual scenes, e.g., common static objects and mobile objects, an additional adaptation of *'Large Scale Image Retrieval for Location Estimation'* is required to make it focus on the scene distinctive objects only. Thus, *Geo-distinctive Visual Element Matching* is proposed to further improve the robustness of the location estimation framework. It explores and exploits geographical distinctiveness of visual elements found in the query image, and it further strengthens the support for finding the true location by devising an aggregated visual representation of a location that combines all visual elements from the query found in the images of that location. The proposed method makes the location estimation more tractable in case of a large image collection, but also more reliable, which leads to an overall significant improvement of the location estimation performance and redefines the state-of-the-art in both geo-constrained and geo-unconstrained location estimation.

4.3.2 Experimental Setup

In this sub-section, we describe the setup of our experimental framework for assessing the performance of our propose method Common Concept Removal in *'Large Scale Image Retrieval for Location Estimation'* pipeline. This provides the background for comparing with the experimental results of 1-NN geolocation estimation approach, which has been specifically described in Sec. 4.2.

We still carry out experiments on San Francisco Landmark dataset [9] that are commonly used in location estimation. This dataset is designed for city-scale location estimation, i.e., geo-constrained location estimation. The database images (background collection) are taken by a vehicle-mounted camera moving around downtown San Francisco, and query images are taken randomly from a pedestrian's perspective at street level by various people using a variety of

```
"PCI_sp_8310_37.795662_-122.416958_937789214_21_671094996_18.7458_4.23059.jpg": {
    "car": [
        261,
        358,
        424,
        461
    ]
},
"PCI_sp_7458_37.797994_-122.405552_937788093_3_671148275_343.652_56.7488.jpg": {},
"PCI_sp_9285_37.794459_-122.425912_937789866_12_718819621_184.23_-7.7657.jpg": {},
"PCI_sp_8042_37.797185_-122.40458_937788695_17_671148492_17.9227_-15.5658.jpg": {},
"PCI_sp_8016_37.797342_-122.40338_937788669_26_671148545_181.36_41.9475.jpg": {},
"PCI_sp_8105_37.796829_-122.407409_937788958_11_671148449_113.993_0.451809.jpg": {
    "car": [
        37,
        317,
        134,
        398
    ],
    "person": [
        334,
        348,
        369,
        413
    ]
},
"PCI_sp_8298_37.795732_-122.416418_937789202_1_671162484_195.796_9.51228.jpg": {},
"PCI_sp_7563_37.798593_-122.400641_937788179_1_671148323_298.492_19.4266.jpg": {
    "bus": [
        3,
        411,
        127,
        480
    ]
},
"PCI_sp_8069_37.79703_-122.405824_937788773_6_671148472_214.426_10.8867.jpg": {},
"PCI_sp_8239_37.796089_-122.413571_937789162_2_671162505_220.068_32.9697.jpg": {},
"PCI_sp_9357_37.79408_-122.423126_937789938_4_718481900_296.01_37.9329.jpg": {},
"PCI_sp_8048_37.797152_-122.404845_937788759_0_671148481_241.615_-1.8755.jpg": {
    "car": [
        493,
        239,
        548,
        287
    ]
},
```


FIGURE 4.10. Detection result save as JSON format

mobile photo-capturing devices. We use 1.06M perspective central images (PCI) derived from panoramas as the database photos, and the original 803 test images as queries. The ground truth for this dataset consists of building IDs. The geolocation of an image is considered correctly predicted if the building ID is correctly predicted.

As the same as the parameters and pre-trained mode indicated before in Sec. 4.2.2, we keep working on SURFsara platform to detect the common concepts from the 1.06M training images. Since the SSD framework is mostly build based on *Python*, we choose the easiest and the most reliable storage method to store our detection result-*JSON* format. Within the file of detection result, the image ID, detected objects names, and the detected object coordinates are stored (shown in Fig. 4.10).

After successfully detect the objects in 1.06M images and saving the corresponding coordinates in file, we apply *Python Image Library* to locate the detected objects and transfer the detected areas to transparent. The detection result and the transfer process is shown in Fig. 4.11. After the detection and removal process, we save the image as the same format of initial dataset and then submit the common concepts removed dataset to *'Large Scale Image Retrieval for Location Estimation'* pipeline.

Hadoop is applied in our experiments to deal with 1.06M dataset from San Francisco, because it is able to store, manage, process and analyze data at petabyte scale and process the local data to each node in a cluster. By adopting SURFsara Hadoop cluster, we are able to build our large framework on it without waiting much time for the experiments results.

FIGURE 4.11. SSD detection and common concept removal result

To fairly compare our propose approach under 1-NN classifier situation and *'Large Scale Image Retrieval for Location Estimation'* pipeline, we set the parameters as same as in Sec. 4.2. In the first stage, we set the object visualize score threshold as 0.8 and non-maximum suppression threshold as 0.5 in the process of SSD detection. Then, we set the evaluation radius as 1km as same as applied in 1-NN classifier, and evaluate the experiments result under different value of HR@k.

4.3.3 Experimental Result

In this section, we report the experiments results and compare the results with the initial output from *'Large Scale Image Retrieval for Location Estimation'* pipeline, including the dataset size, extracted feature size, index size, time efficiency, and the geolocation prediction performance. We run *'Large Scale Image Retrieval for Location Estimation'* pipeline (represent as GVR and DVEM) 9 times and compare the outputs with 1-NN geolocation prediction classifier as described in Sec. 4.2.

First, we would like to find out the different behavior of DVEM, GVR, and 1-NN approach in geolocation prediction. We apply the same evaluation method HR@k, use the initial dataset without common concepts removed, and set the evaluation radius to 1km. The experiment result is compared in Fig. 4.12. The experiment result is almost as same as indicated in *'Large Scale Image Retrieval for Location Estimation'*, except for the 1-NN outperforms the GVR when HR@k is 1, and the GVEM can outperform both the GVR and the 1-NN approach.

As indicated in the Sec. 4.2, the Common Concept Removal approach can efficiently improve the geolocation prediction performance and slightly reduce the index size, because the SSD detection accuracy can reach to 90% high. In this section, by re-building *'Large Scale Image Retrieval for Location Estimation'*, we would like to find out the geolocation prediction performance of 1-NN, GVR, and DVEM after applying our proposed method-Common Concept Removal to both 803 query images and the 1.06M training images. First, we apply SSD framework to both 803 query images and 1.06M training images, calculate the detection result, and draw the distribution in Fig. 4.13. In total, we successfully detect 314229 common concepts, including 250772 cars, 28977 persons, 10496 buses, 3232 plants, 12044 trains, 1913 motorbikes, 2727 bicycles, and 4068 boats. The initial dataset size is reduced from 32.2GB to 31.2GB, which is approximately 3.1% reduction. The reduction of dataset size helps to ease the burden of the storage system. Although

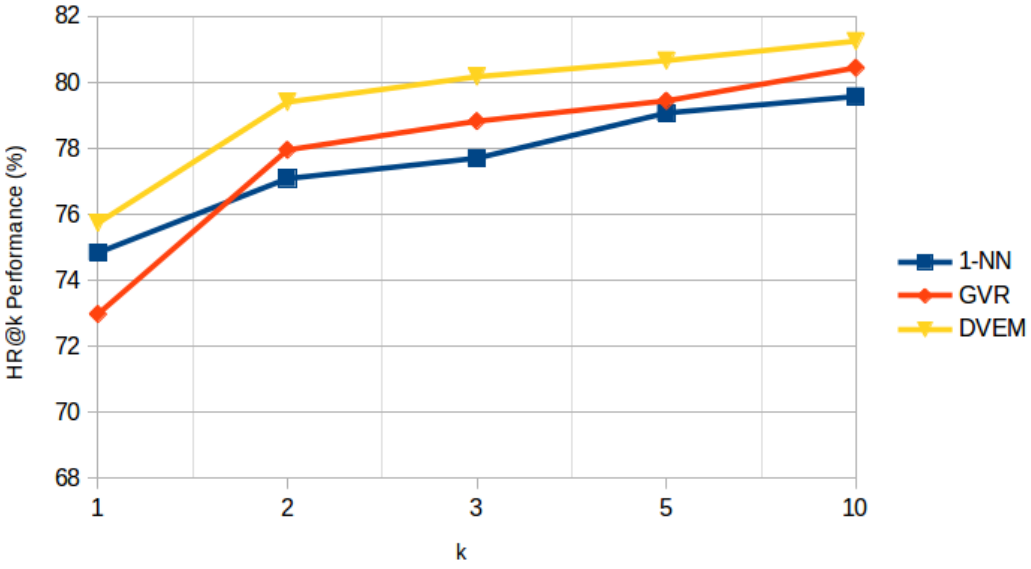

FIGURE 4.12. HR@k performance for varying k on the San Francisco street view dataset

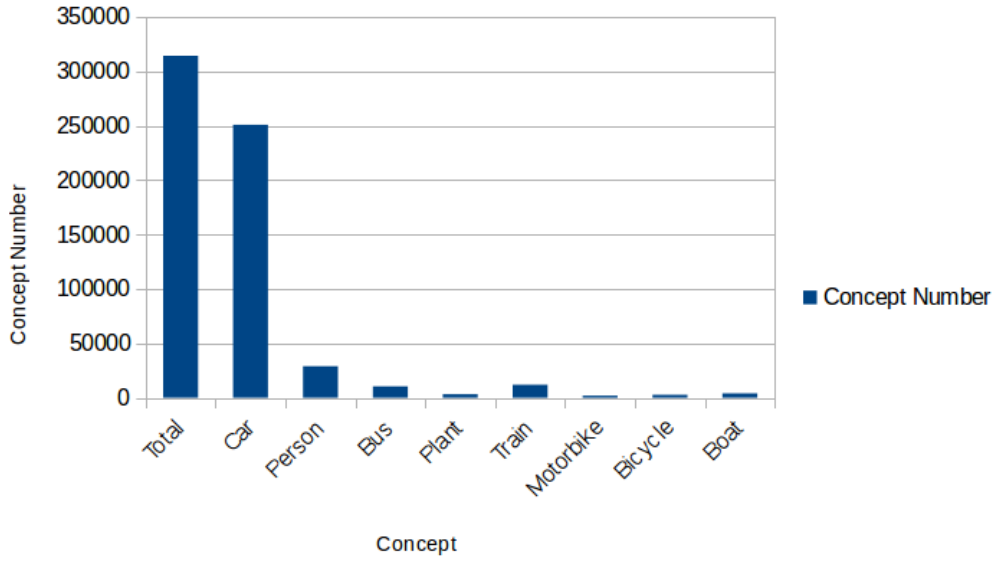

FIGURE 4.13. Detected concepts distribution with removing from both query and training dataset

the dataset size is only reduced by 1GB, consider other large dataset like *Flikr* or *MediaEval'15 Placing Task dataset*, the reduction extent can be impressive. Since we have evaluated the SSD detection performance in Sec. 4.3 on 803 query images, it is not necessary to evaluate the SSD performance again in this section, and because of the unaffordable time consuming to evaluate the whole 314229 common concepts manually.

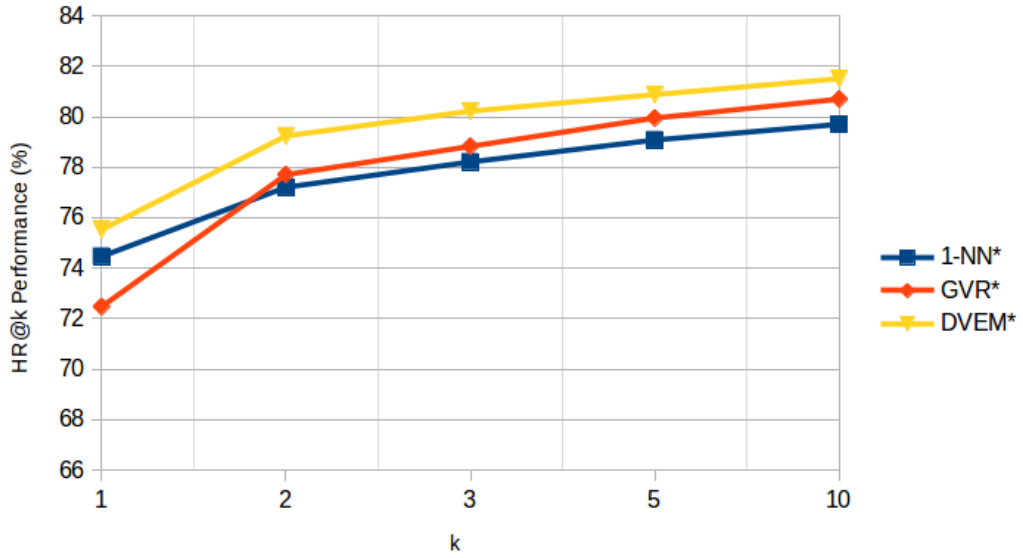

FIGURE 4.14. HR@k performance for varying k on the SanFrancisco street view dataset after removing concepts from training images

Before applying our proposed common concept removal approach to both query and training dataset, we firstly remove the detected common concepts only from 1.06M training dataset and keep the query images original for finding out the influence of our proposed approach to the geolocation prediction performance. We adopt the same parameters as indicated in Fig. 4.12 except for only removing the detected concepts from 1.06M training images. The result is shown in Fig. 4.14. By comparing the different geolocation prediction performance from Fig. 4.12 and Fig. 4.14, we find out that the geolocation prediction performance is almost the same as using the initial dataset with only tiny performance sacrifice. The performance details are shown in Table. 4.2, the performance of removing the common concepts from the 1.06M training dataset are represented as 1-NN*, GVR*, and DVEM*. Meanwhile, the index size is reduced by 10.16% from 73.8GB to 66.3GB, the **RQ2** is answered, removing concepts from training images help with reducing the index.

Then, we would like to find out the influence of our proposed method to geolocation estimation performance by removing common concepts from both query and training dataset. To be more specifically, for each common concept (e.g., car, person, bus, etc.), we choose to remove all the detected common concepts from the 1.06M training images and remove the specific common concept from the 803 query images. To make the experiment result more readable, we plot 3 graphs to show the different behavior of our proposed method to 1-NN, GVR and DVEM.

Approach	HR@1(%)	HR@2(%)	HR@3(%)	HR@5(%)	HR@10(%)
1-NN	74.84	77.09	77.71	79.08	79.58
1-NN*	74.47	77.21	78.21	79.08	79.70
GVR	72.98	77.96	78.83	79.45	80.45
GVR*	72.48	77.71	78.83	79.95	80.70
DVEM	75.74	79.41	80.18	80.67	81.26
DVEM*	75.53	79.24	80.22	80.88	81.51

Table 4.2: 1-NN, GVR, DVEM performance comparing before and after concepts removed from training dataset

Firstly, after applying SSD for both the query and training dataset, we adopt *Python Image Library* to transfer the detected common areas to transparent, and then wrap them to MapFile format as the input to 1-NN approach. Next, we re-run the 1-NN geolocation prediction classifier 9 more times and draw the performance in Fig. 4.15. The original geolocation prediction performance without applying our proposed method is shown as 'initial' for the purpose of comparing with other performances with different common concepts removed. Comparing with the original experiment output of 1-NN geolocation prediction classifier, after applying our proposed method, most removed concepts can help improve the geolocation prediction performance by average 6% except for concept 'boat'. As indicated before, there are only 3 boats detected from 803 query images (all wrongly detected) and 4068 boats detected from 1.06M images. However, comparing with the SSD detection result in Fig. 4.13, we find out that the geolocation prediction performance of 'All concepts' is beaten by common concept 'car', and 'car' is beaten by 'person', although there are more concepts removed from 'All concepts' than 'car' and more concepts removed from 'car' than 'person'. We guess that this strange situation may be caused by the fact that we choose to remove the detected common concepts as the shape of rectangular, which are not exactly the shape of concepts. More specifically, for larger concepts like cars, additional districts are also removed, which belong to the rectangular area and do not belong to the detected car district. These removed additional districts usually contain some components of architectures on the street, which are crucial to predict the geolocation. On the other hand, for smaller concepts like persons, the removed rectangular regions can almost perfectly match the shape of concepts, which means sacrificing less contributable components in street view images comparing with cars. To prove our guess, we implement an additional experiment to draw the SIFT detected salient points in common concepts removed images, which is shown in Fig. 4.16. The first image shows the SIFT salient point detection result of initial image, the middle one indicates the salient points after the person being removed, the right image illustrates the salient points after the car being removed. By comparing the different salient points detection results, our guess is proved.

The result in Fig. 4.15 shows that our proposed Common Concept Removal approach can indeed help improve the geolocation prediction performance whether by removing concepts only from query dataset or removing concepts both from query and training dataset.

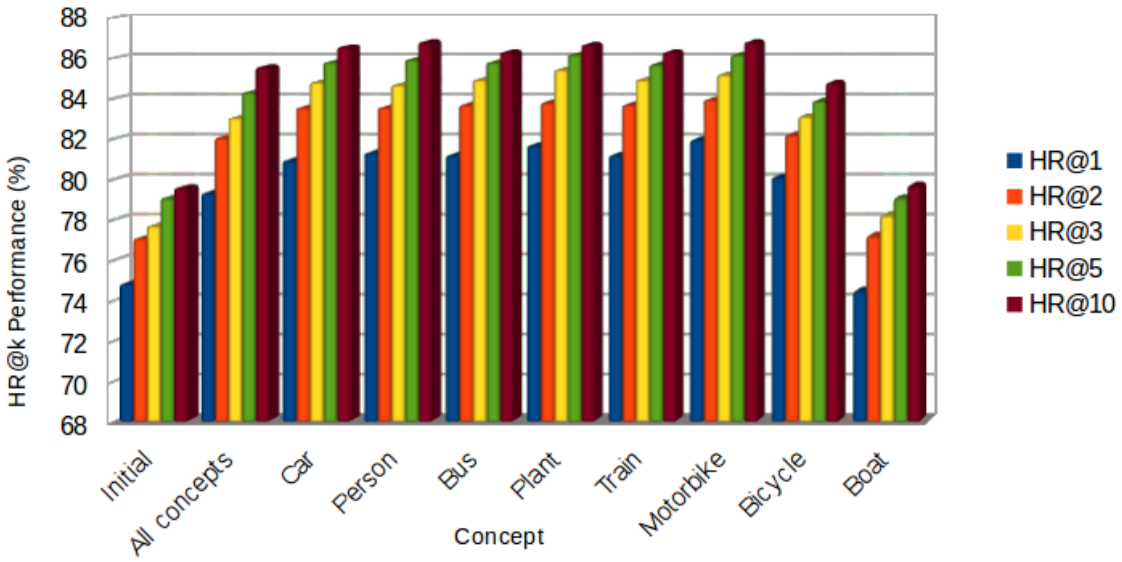

FIGURE 4.15. 1-NN performance with all concepts removed from query and training dataset

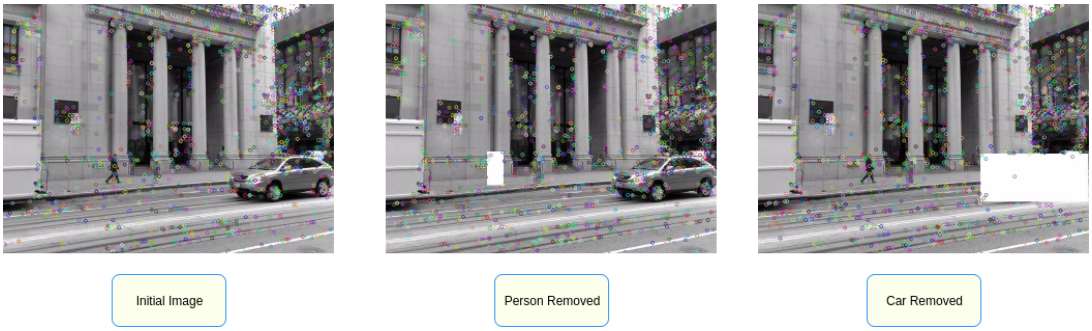

FIGURE 4.16. SIFT salient points detection analysis after applying CCR

Next, we analyze the experiment result by applying our proposed method to GVR, which is proposed by Li in [1]. The experiment result that combine our proposed method and GVR is shown in Fig. 4.17. Comparing with Fig. 4.16, the reason that caused the GVR concentrates more than 1-NN is that GVR believes there will be more elements of visual content of the query image in true location than wrong location. That means, compare with wrong location, candidate images from the true location contain more visual element (like some landmark buildings) that are also contained in query image. For example, there are only slight differences comparing HR@1 with HR@10, because HR@1 uses the nearby candidate images to contribute to the match score.

Then, we also apply our propose approach to DVEM, which is proposed by Li in [3]. The performance of DVEM is shown in Fig. 4.18. The trend of lines in Fig. 4.18 are almost as same as

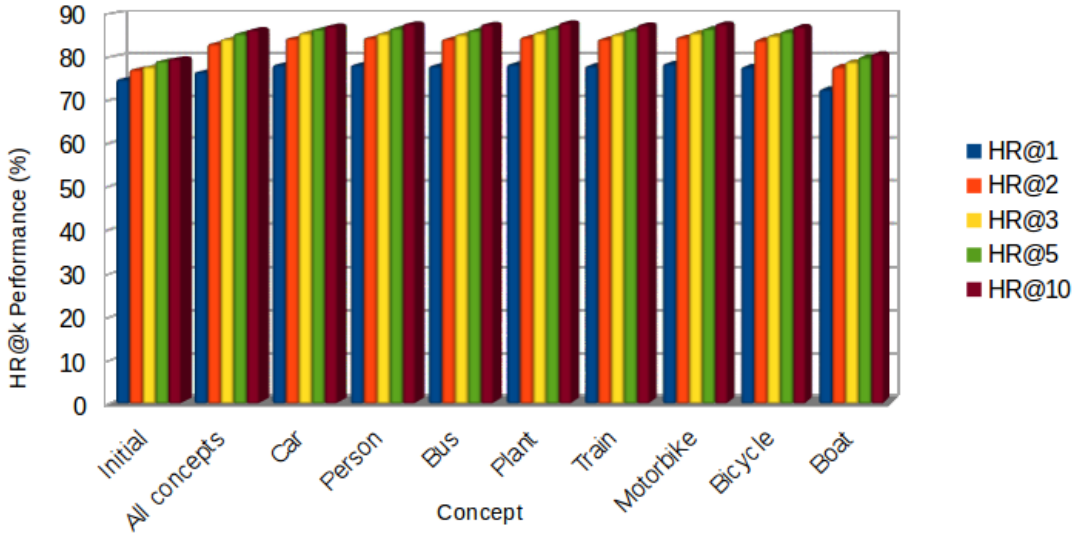

FIGURE 4.17. GVR performance with all concepts removed from query and training dataset

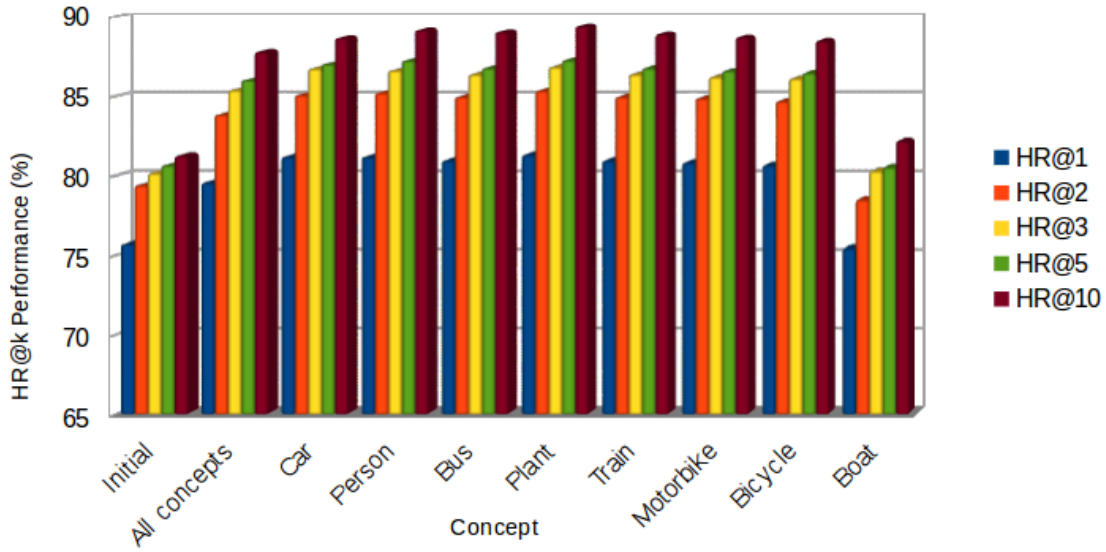

FIGURE 4.18. DVEM performance with all concepts removed from query and training dataset

in Fig. 4.16, except that the geolocation prediction performance in Fig. 4.18 all outperform the performance in Fig. 4.16.

Finally, as all the concepts have been removed from both the query and training images, we suppose that with such large number of common concepts removed, the index size and feature size

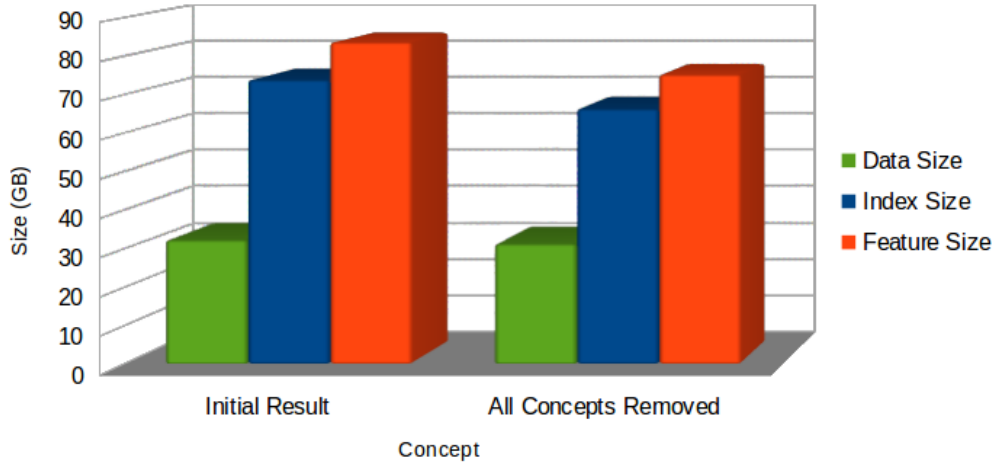

FIGURE 4.19. Influence to index size with all concepts removed from query and training images

can be considerably reduced. We compare feature size, index size, and initial dataset size in Fig. 4.19. The 'Initial Result' represents the original output of *'Large Scale Image Retrieval for Location Estimation'* without applying our proposed approach, and 'All Concepts Removed' represents the experiment output after applying our propose approach. In this graph, we choose not to compare the index size of different common concepts, because we believe that the geolocation prediction performance is mainly influenced by the query images and the index size is mainly influenced by the 1.06M training images. Since it has been proved previously that removing common concepts from the query images can indeed help improve the performance, thus, we mainly focus on reducing the index size as large as possible and choose to ignore the influence of different common concepts to the index size. From Fig. 4.19, the dataset size is reduced from 32.2GB to 31.2GB, the feature size is reduced from 83.6GB to 75.2GB, and the index size is reduced from 73.8GB to 66.3GB. Thus, **RQ3** is successfully answered, removing concepts from query and training images can help with improving the performance and reducing the index size.

4.4 Further Reduce Index Size Based on OVST

Can we further reduce the image retrieval index and keep outperforming the initial geolocation prediction performance? To answer this question, we choose to adjust the object visualize score threshold from 0.8 to lower values, as 0.5 and 0.2. As discussed before, in order to avoid sacrificing too much geolocation prediction performance, we set the visualize score threshold as 0.8 for both the query images and the 1.06M training images. With the high threshold as 0.8, the SSD common concepts detection accuracy can reach over 90% as indicated in Fig. 4.5. However, the drawback of such high value of threshold is that the total SSD common concept detection number

FIGURE 4.20. SSD deep learning detection result example in San Francisco

is largely limited. For example, we implement the SSD common concept detection with one San Francisco street view image and draw the detection result in Fig. 4.20, in which all the manually defined common concepts can be successfully detected with different object visualize score. The score of the car in the left and bottom corner is 0.326 and the score of the person in frond of the building is 0.312. All the experiment results in Sec. 4.2 and Sec. 4.3 are built based on object visualize score threshold as 0.8, which means there are more potential available common concepts are ignored. Thus, we believe if we can set the object visualize score threshold score smaller than 0.8, there should be more common concepts detected.

For the purpose of further proving our speculation, we choose a representative San Francisco street view image from the 803 query images, comparing the different detection result by setting the object visualize score as 0.8, 0.5, and 0.2, which is shown in Fig. 4.21. When the OVST (object visualize score threshold) value is 0.8, there are only three persons detected, other common concepts existing in this image are automatically ignored. When we set the OVST as 0.5, there are 5 persons and 1 car detected. When the OVST is set as 0.2, there are 8 persons and 2 cars detected. The experiment in Fig. 4.21 proved our speculation, we can detect more common concepts by adjusting the object visualize score threshold.

4.4.1 Experiment Setup

In this sub-section, we describe the setup of our experiment framework for assessing the geolocation prediction performance of 1-NN classifier and GVR with different object visualize score threshold.

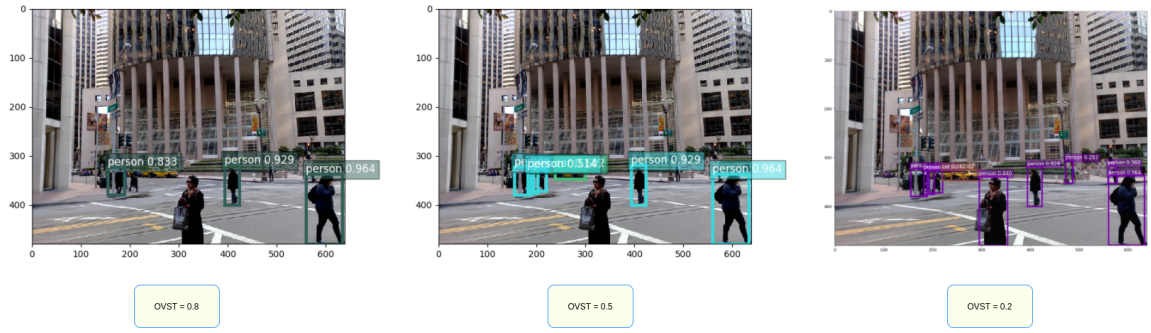

FIGURE 4.21. Different object visualize score threshold detection result comparing

We keep using the same San Francisco street view images as our experiment dataset, and setting the object visualize score threshold as 0.5 and 0.2 respectively in the process of SSD common concepts detection. After detecting the common concept for both the query images and the training images, we save the coordinates and class names of the common concepts in database and retrieve the generated data to local PC. Then, by applying the PIL library, we transfer all the detected common concepts areas to transparent and generate the concepts removed images as MapFile. With the help of previous work, we submit the job to SURFsara Hadoop cluster, retrieve the geolocation prediction performance and count the size of index, feature size, and initial data. Meanwhile, we keep using the same SSD pre-trained model - Resnet-50 and same evaluation approach HR@k, which is proposed by Li and introduced in Sec. 4.2.4.

4.4.2 Experiment Result

As indicated in Fig. 4.21, we know that higher object visualize score threshold can lead to larger number of common concepts detected. We re-implement the SSD deep learning framework two more times for OVST value of 0.5 and 0.2 respectively, calculate the detected common concepts number and distribution for each concepts, which is shown in Fig. 4.22. Totally, 314229 concepts are detected from 1.06M training images and 803 query images with OVST = 0.8, 428428 concepts are detected with OVST = 0.5, and 800223 concepts are detected with OVST = 0.2. Comparing with the detected common concepts number of OVST = 0.8, the number is improved by 36.3% with OVST = 0.5, and 154.7% with OVST = 0.2. This relationship also works for other common concepts, like car, person, bus, etc. We believe that the index size and feature size can be further reduced with the increasing detected common concept number.

The large improvement of common concepts number is amazing and greatly motivate us to find out the influence of OVST to geolocation prediction performance. Then, we propose the following two interesting questions:

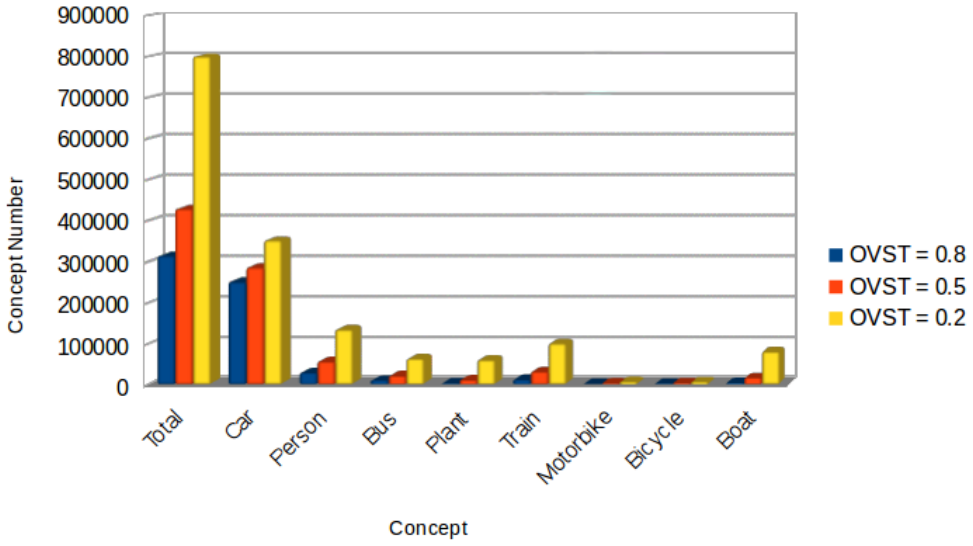

FIGURE 4.22. SSD detection number counting with different OVST (Object Visualize Score Threshold)

- Will our propose approach CCR still outperform the initial geolocation performance of 1-NN with lower OVST?

- Will the performance of larger OVST outperform the smaller OVST in 1-NN geolocation prediction classifier and GVR geolocation prediction approach?

To answer these two questions, we re-implement the 1-NN classifier in *'Large Scale Image Retrieval for Location Estimation'* pipeline and adopt the same strategy as indicated in Sec. 4.4.1. The geolocation prediction performance comparing of different OVST is shown in Fig. 4.23. The initial geolocation prediction performance in *'Large Scale Image Retrieval for Location Estimation'* without implementing our propose method is represented as '1-NN' and green line. For convenience, we represent the OVST of 0.8, 0.5, 0.2 as '1-NN(0.8)', '1-NN(0.5)', and '1-NN(0.2)' respectively. The experimental result in Fig. 4.23 illustrates that our propose approach CCR can considerably improve the geolocation prediction performance for all the three different OVST values. By comparing the different behavior of different OVST values, we also find out that the '1-NN(0.8)' slightly outperforms the '1-NN(0.5)', and '1-NN(0.5)' beats the '1-NN(0.2)' by approximately 1%. This result answered our second proposed question, the performance of larger OVST can slightly outperform the smaller OVST in 1-NN geolocation prediction classifier. This situation is easy to understand, as we adopting the smaller OVST, the detection accuracy will also decrease, because there will be more wrongly removed common concepts from dataset, which may contain useful elements to the geolocation prediction process.

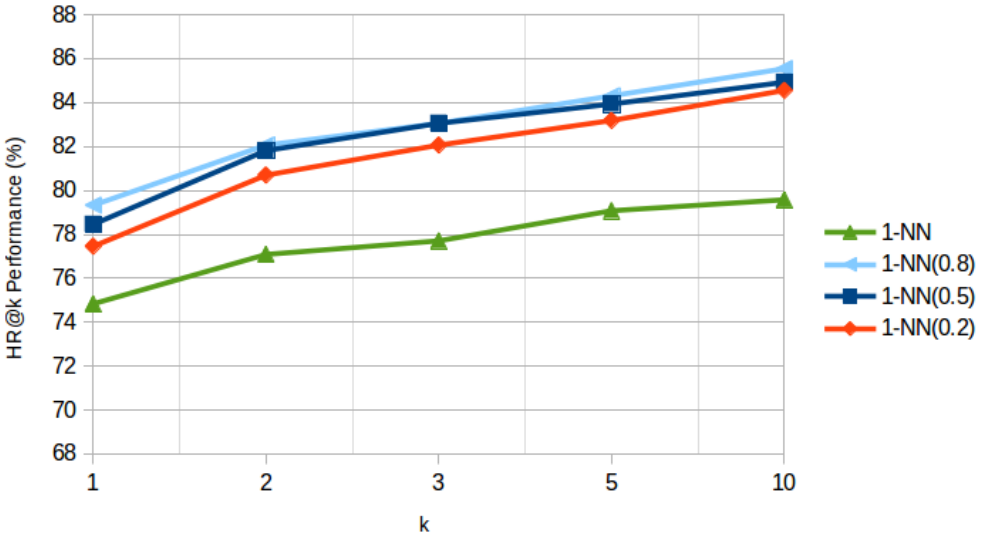

FIGURE 4.23. 1-NN performance comparing with different value of OVST

What is the influence of different OVST to GVR geolocation prediction performance? To answer this question, we re-implement GVR in *'Large Scale Image Retrieval for Location Estimation'* pipeline and adopt the same evaluation method HR@k. The geolocation prediction performance comparing is illustrated in Fig. 4.24. The initial performance of GVR is represented as 'GVR' and blue line, the performance of GVR after applying the CCR with different OVST is represented as 'GVR(0.8)', 'GVR(0.5)' and 'GVR(0.2)' respectively. The experiment result indicates that our propose method CCR can significantly outperform the initial GVR performance for all different OVST values. Meanwhile, the performance of larger OVST can slightly outperform the smaller OVST in GVR, for example, in all the three different OVST, 'GVR(0.8)' performs the best and 'GVR(0.2)' performs the worst. However, comparing with the considerably performance improvement to initial GVR, we consider the slightly performance difference among 'GVR(0.8)', 'GVR(0.5)', 'GVR(0.2)' are acceptable.

Since we have proved before in Fig. 4.23 and Fig. 4.24, our propose method CCR can always outperform the initial performance of 1-NN and GVR, even with smaller OVST. Then, we would like to find out the influence of different OVST to data size, index size, and feature size. The experiment output is indicated in Fig. 4.25, in which the data size is represented as green bar, the index size is represented as blue bar, and the feature size is represented as orange bar. By comparing the size of different OVST with the initial size in *'Large Scale Image Retrieval for Location Estimation'* pipeline, we find out that the lower value of OVST lead to smaller size of data size, index size, and feature size. To be more specifically, we only analyze the index size changing, the index size of initial result is 73.8GB, the index size with OVST = 0.8 is 66.3GB, the index size with OVST = 0.5 is 63GB, and the index size with OVST = 0.2 is 51GB. Thus, we can

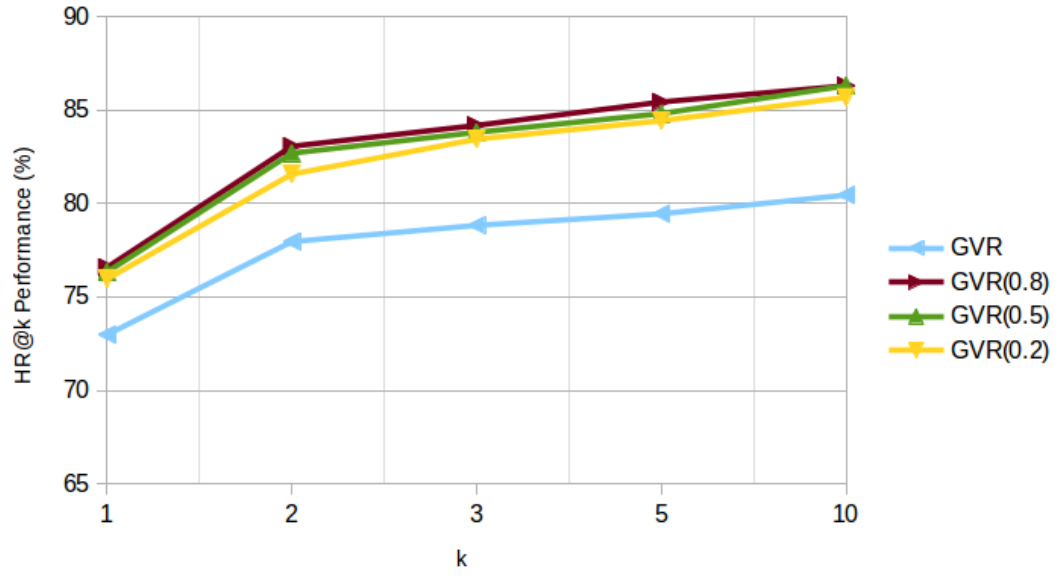

FIGURE 4.24. GVR performance comparing with different value of OVST

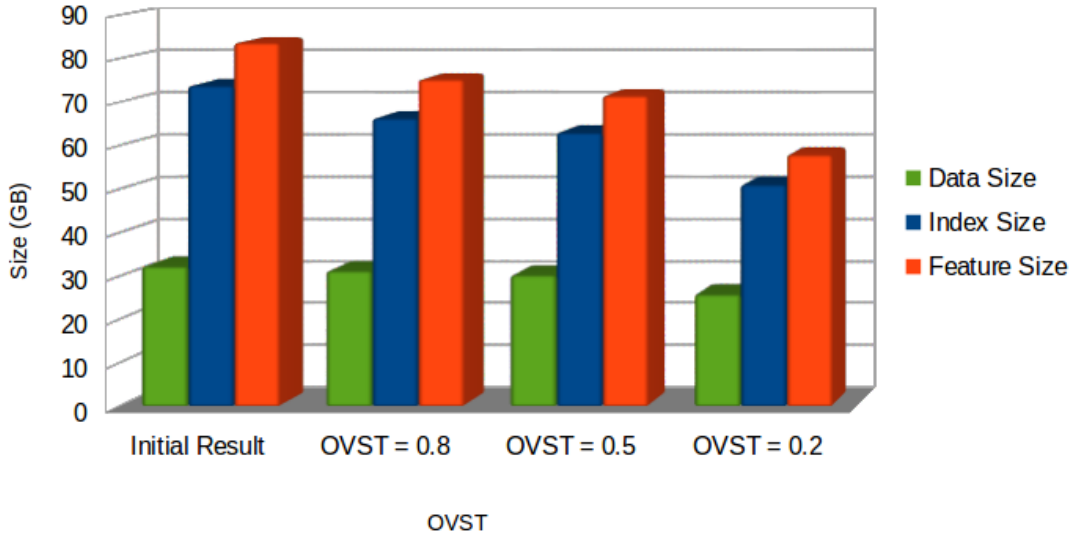

FIGURE 4.25. Index size, feature size, and data size comparing with different OVST values

draw the conclusion that smaller OVST can lead to smaller index size. By applying the OVST = 0.2, the index size is reduced from 73.8GB to 51GB, which is a 30.9% decrease.

4.5 Conclusion

In this chapter, we firstly introduced our propose approach in this thesis CCR (Common Concept Removal) in **Sec. 4.1**, and analyzed the SSD common concept detection results. Then we tested the CCR performance in 1-NN geolocation prediction classifier before formally start testing with GVR and DVEM, which is indicated in **Sec. 4.2**. By only removing the different common concepts from the 803 query images, the geolocation prediction performance was considerably improved for all common concepts except for concept 'boat', because all the 3 'boat' were wrongly detected. After applying our propose approach to only the query images and proving that CCR can improve the geolocation prediction performance, we applied the CCR to both the query images and the 1.06M training images in **Sec. 4.3**, and compared the different concepts influences to 1-NN, GVR, and DVEM. The experimental results illustrated that our propose approach CCR can not only improve the geolocation performance for 1-NN, GVR, and DVEM, but also reduce the index size by approximately 10.16%. For the purpose of investigating the possible solution to further reduce the index size without sacrificing geolocation prediction performance, we compared the influence of different OVST (Object Visualize Score Threshold) to geolocation prediction performance in **Sec. 4.4**. By applying lower OVST as 0.2, the index size can be significantly reduced by 30.9% comparing with initial index size. Although the geolocation prediction performance of lower OVST(0.2) was worse than higher OVST(0.8) by approximately 0.7%, comparing with the considerably improvement with the initial performance and significantly further reduced index size, we consider this slight sacrifice to be acceptable.

CHAPTER 5

DISCUSSION AND FUTURE WORK

5.1 Discussion

The objective of the research reported in this thesis was to put forward an approach to reduce the index size and improve the geolocation prediction. To investigate the possibilities to reduce the index size and improve the geolocation prediction performance to a substantial extent, we proposed the algorithm-Common Concept Removal, tested on geo-constrained scenario with San Francisco street view dataset. We organized these perspectives as the four main chapters in this thesis. While Chapter 1 introduced the background of our research and described the research questions. Chapter 2 covered the general introduction of related work in the content-based geolocation prediction field. Chapter 3 and Chapter 4 focused on describing the implementation details of our proposed approach and the experimental results. Specifically, Chapter 3 covered the related frameworks and techniques, which were crucial to the final success of our research. In this section, we reflected on the algorithmic solution that we developed and the results that we obtained.

Our research can be considered to have been motivated by the two underlying questions:

- Does removing common concepts from query images improve the geolocation estimation?
- Does removing concepts from query and training images help with improving the geolocation prediction performance and reducing the index size?

Following the research questions in Sec. 1.4 of **Chapter 1**, we firstly introduce the related research of geolocation prediction in **Chapter 2**, in which all the related papers are classified as four different categories, geo-constrained location estimation, geo-unconstrained location estimation, concept detection, and image retrieval index reduction. Due to the large amount of

engineering work in this thesis, we briefly introduce the related techniques and frameworks in **Chapter 3**. Because they played crucial roles in the process of realizing our proposed method and re-implementing *'Large Scale Image Retrieval for Location Estimation'*. Mostly inspired by related research, we realized that most current index reducing approaches are implemented at the cost of using lower-level of features, which result in sacrificing the geolocation prediction performance. Thus, in **Chapter 4**, the rationale approach - CCR is proposed in Sec. 4.1. Before formally test the CCR approach in GVR (Geo Visual Ranking) and DVEM (Geo-Distinctive Visual Element Matching), we firstly choose to implement it in 1-NN (1-Nearest Neighbor) geolocation prediction classifier, which is built based solely on visual similarity. By removing common concepts only from 803 query images, we analyzed the influences of different concepts to the geolocation prediction performance, and draw the conclusion that our proposed approach CCR can considerably improve the geolocation prediction performance by applying the 1-NN geolocation prediction classifier. The experiment result in Sec. 4.2 greatly encouraged us to further implement CCR to both the query and training images by applying GVR and DVEM, to improve the performance and reduce the index size. In Sec. 4.3, we applied CCR to both the query and training images and tested the performance in GVR and DVEM. The experiment results indicated that the performance is improved by approximately 6.0% and the index size is reduced by 10.2%, which answered the research question that CCR can help with reducing the index size and improving the geolocation prediction performance. With the research question answered in Sec. 4.3, we moved forward towards our overall goal in this thesis, that is, further reduce the index size and keep CCR outperform the initial output of 1-NN and GVR. In Sec. 4.4, through revising the object visualize score threshold from 0.8 to 0.5 and 0.2, more common concepts were detected from the 803 query images and the 1.06M training images. After applying CCR with smaller OVST, the index size is further reduced to 51GB, comparing with the initial index size of 73.8GB, we consider the reduction size of index is significant. Although the experiment results also illustrated that lower OVST lead to worse performance comparing with higher OVST, comparing with the initial performance and the benefit of index reduction size, we believe this slightly performance sacrificing (approximately 0.7%) is acceptable.

5.2 Future Work

Based on the findings presented in this thesis, we would like to make the following recommendations for future work which we think are substantial and promising for reducing the index size of large scale image retrieval and improving the geolocation estimation performance.

5.2.1 Detect the Boundary of Concepts

As indicated in **Chapter 4**, we met the problem that the large common concept lead to worse geolocation prediction performance, even the detected number of large common concepts (e.g.,

car) is overwhelming the small common concepts (e.g., person). This situation is caused by the fact that we have adopted the rectangular area to represent the detected common concepts, inside which exist some special small regions that belong to the street view (e.g., buildings) and do not belong to the common concepts. We consider these sacrificed regions being 'innocent' because they can indeed help with the geolocation prediction process. In this thesis, we have applied the SSD deep learning framework to detect the common concepts from millions of street view images, however, the dilemma exist in this approach is that the detected common concepts areas can only be represented as default boxes.

In the future work, we hope to find a way to detect the exact boundary of common concepts by combining the SSD deep learning framework and SIFT. Since SSD is efficient and accurate to detect the box of common concepts, and SIFT algorithm can help with detecting the exact boundary of common concept in the default box. The exact boundary detection of common concept can obviously benefit the geolocation prediction performance and the index size reduction in the future work.

5.2.2 Train SSD Model For San Francisco

In the process of implementing our propose approach CCR, we choose the pre-trained model Resnet-50 (Residual Network with 50 layer), which is trained based on VOC07+12. Based on this model, the experiment implementing results indicated that most of the manually defined concepts can be well detected for both the query and training images. However, we believe that the detection number and accuracy can be further improved by training SSD model based on street view images in San Francisco in addition to query and training images. This proposal can be realized by the following strategy: First, choose a pre-trained mode (like Resnet-50) and record all the correctly detected common concepts from the additional street view images from San Francisco, including class names and coordinates. Then, create text files for each concept, which are labeled by class names and coordinates. Finally, by adopting the training model offered by SSD, new model for San Francisco will be generated. Due to the limited period of working on this thesis, we didn't have enough time to label new concepts and generate new models. Hope this direction can further help with improving the geolocation prediction performance and reducing the index size.

5.2.3 Test CCR on Global Scale

'Large Scale Image Retrieval for Location Estimation' focus on predicting the geolocation based on visual content in global scale while we solely test our propose approach CCR in geo-constrained area - San Francisco dataset. Thus, we recommend that further research can work on testing the CCR on global scale dataset. We consider the CCR is promising in improving the global scale geolocation prediction performance and reducing the image retrieval index size. However, the CCR approach is proposed based on restricted areas, in which some common concepts like cars

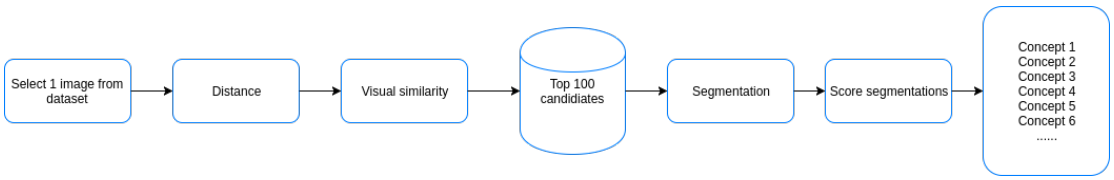

FIGURE 5.1. Index size, feature size, and data size comparing with different OVST values

and buses looks similar to each other. When applying CCR on global scale, the performance will possibly be sacrificed, because common concepts from different areas may vary a lot. For example, taxis from San Francisco and London are totally different, in some special case, the taxi on the street can also contribute to the geolocation prediction process. Based on all the discussion above, we consider it is necessary to further test our proposed approach on global scale.

5.2.4 Automatic Concept Selection in Data-Driven Way

In this thesis, we have manually defined 8 common concepts - *car, person, bus, plant, train, motorbike, bicycle, boat* and removed them from both the query and training images to reduce the index size and improve the geolocation prediction performance. However, the potential problem that exists in our proposed method CCR is that all the common concepts are selected manually based on San Francisco dataset. These concepts applied in CCR may not help improve the geolocation prediction performance neither reduce the index size for other scenarios. For example, by comparing the buses and cars from London and San Francisco, we can easily find out that they look totally different. Thus, it is not reliable and convincing to use the same common concepts in this thesis to other scenarios, like global geolocation prediction or geolocation prediction for other cities.

In view of above, we consider that the automatic concept selection in data-driven way is a better choice to define common concept rather than manually definition. There are two obvious advantages that make us believe the automatic concept selection can outperform the manually concept selection: First, the automatic concept selection can avoid the subjective influence that is caused by human in the process of manually defining common concept. For example, we believe that the concept 'tree' is not geo-distinctive and removed it from query and training images. However, 'tree' can also contribute to the geolocation prediction in some special scenarios. Second, automatic concept selection can help find out more common concepts than manual concept selection from dataset.

Thus, in the end of this research, we proposed a potential approach to help select concept automatically from dataset, the pipeline is shown in Fig. 5.1. Firstly, we randomly choose one street view image from San Francisco dataset. Then, by calculating the distances of rest images to the selected image, we choose the distances that are larger than 5KM as candidate images.

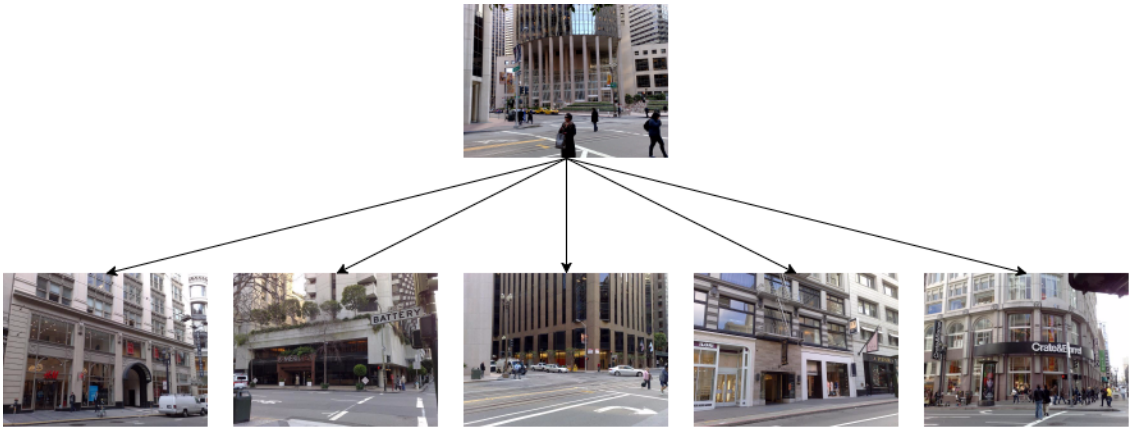

FIGURE 5.2. Performance of image retrieval system based only on color descriptor

Because we believe that the images nearby the selected image may contain common concepts that cannot be found from other regions of this city. Next, we choose the 100 most visually similar images from the candidates of previous step. To realize this, we built a simple image retrieval system based only on color descriptor, the performance of this system is shown in Fig. 5.2. Then, we divide each of the 100 candidate images to 100 small segmentations as the size of 48×64. We consider that the small regions from the candidate images can be represented as potential concepts. Finally, by iterative calculating the similarity among the 10000 small images, we select the top 10 small images as the automatically selected concepts, which have the highest score. We define the score of each small image as the number of other images that match to it with threshold larger than 7 (represented as the similarity between vectors).

The automatically select common concept process and results are shown in Fig. 5.3. We list the selected common concepts in the bottom of this figure, in which tree, car, bus and even billboard are chosen as the common concepts. Even though the automatically selected common concepts are not representative enough to be applied in CCR approach, we consider this direction promising in the future research.

FIGURE 5.3. Automatically select common concept result

BIBLIOGRAPHY

[1] X. Li, M. Larson, and A. Hanjalic, "Global-scale location prediction for social images using geo-visual ranking," *IEEE Transactions on Multimedia*, 2015.

[2] X. Li, M. Larson, and A. Hanjalic, "Pairwise geometric matching for large-scale object retrieval," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2015.

[3] X. Li, M. Larson, and A. Hanjalic, "Geo-distinctive Visual Element Matching for Location Estimation of Images,"

[4] B. G. Prasad, S. K. Gupta, and K. K. Biswas, "Color and Shape Index for Region-Based Image Retrieval,"

[5] P. Kinnaree, S. Pattanasethanon, S. Thanaputtiwirot, and S. Boontho, "RGB color correlation index for image retrieval," in *Procedia Engineering*, 2011.

[6] G. Tolias, Y. Avrithis, and H. Jegou, "Image Search with Selective Match Kernels: Aggregation Across Single and Multiple Images," *International Journal of Computer Vision*, 2016.

[7] A. Zamir and M. Shah, "Accurate Image Localization Based on GoogleMaps Street View," *in Proc. ECCV*, pp. 255–268, 2010.

[8] W. Zhang and J. Kosecka, "Image Based Localization in Urban Environments," *Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT'06)*, pp. 33–40, 2006.

[9] D. M. Chen, G. Baatz, and S. S. Tsai, "City-Scale Landmark Identification on Mobile Devices,"

[10] N. E. W. Species and H. Giglioli, "LANDMARK RECOGNITION FOR AUTONOMOUS MOBILE ROBOT," *Genus*, 1830.

[11] J. F. Barbara Zitova, "Landmark Recognition using invariant features," *Elsevier*, vol. 20, pp. 541–547, 1999.

[12] T. Sattler, M. Havlena, K. Schindler, and M. Pollefeys, "Large-Scale Location Recognition and the Geometric Burstiness Problem," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1582–1590, 2016.

[13] A. Torii, J. Sivic, M. Okutomi, and T. Pajdla, "Visual Place Recognition with Repetitive Structures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 11, pp. 2346–2359, 2015.

[14] P. Gronat, J. Sivic, G. Obozinski, and T. Pajdla, "Learning and Calibrating Per-Location Classifiers for Visual Place Recognition," *International Journal of Computer Vision*, pp. 1–18, 2016.

[15] J. Choi and G. Friedland, *Multimodal Location Estimation of Videos and Images*.

[16] Y. T. Zheng, M. Zhao, Y. Song, H. Adam, U. Buddemeier, A. Bissacco, F. Brucher, T. S. Chua, and H. Neven, "Tour the World: Building a web-scale landmark recognition engine," *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*, pp. 1085–1092, 2009.

[17] J. Hays and A. A. Efros, "IM2GPS: Estimating geographic information from a single image," *26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, vol. 05, 2008.

[18] J. Sivic, B. C. Russell, A. a. Efros, A. Zisserman, and W. T. Freeman, "Discovering Objects and their Localization in Images," *IEEE Int'l Conf. on Computer Vision (ICCV'05)*, pp. 370–377, 2005.

[19] B. Orhan and M. Shah, "IMPROVING SEMANTIC CONCEPT DETECTION AND RETRIEVAL USING CONTEXTUAL ESTIMATES," pp. 536–539, 2007.

[20] S. Agarwal, A. Awan, and D. Roth, "Learning to detect objects in images via a sparse, part-based representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1475–1490, 2004.

[21] R. Arandjelovic and A. Zisserman, "Three things everyone should know to improve object retrieval," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2911–2918, 2012.

[22] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single Shot MultiBox Detector,"

[23] S.-i. Keypoints and D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[24] L. Yang, B. Geng, Y. Cai, A. Hanjalic, and X. S. Hua, "Object retrieval using visual query context," *IEEE Transactions on Multimedia*, vol. 13, no. 6, pp. 1295–1307, 2011.

[25] T. Kliegr, K. Chandramouli, J. Nemrava, V. Svatek, and E. Izquierdo, "Combining Image Captions and Visual Analysis for Image Concept Classification,"

[26] M. J. Huiskes, B. Thomee, and M. S. Lew, "New Trends and Ideas in Visual Concept Detection The MIR Flickr Retrieval Evaluation Initiative," pp. 527–536, 2010.

[27] K. E. A. V. D. Sande and T. Gevers, "University of Amsterdam at the Visual Concept Detection and Annotation Tasks," vol. 32, pp. 343–358, 2010.

[28] J. Philbin, O. Chum, M. Isard, J. Sivic, and a. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[29] Y. G. Jiang, J. Yang, C. W. Ngo, and A. G. Hauptmann, "Representations of keypoint-based semantic concept detection: A comprehensive study," *IEEE Transactions on Multimedia*, 2010.

[30] H. Brunzell and J. Eriksson, "Feature reduction for classification of multidimensional data," *Pattern Recognition*, vol. 33, no. 10, pp. 1741–1748, 2000.

[31] M. Park, J. Jin, and L. Wilson, "Fast Content-Based Image Retrieval Using Quasi-Gabor Filter and Reduction of Image Feature Dimension," *Proceedings Fifth IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 0–4, 2002.

[32] B. Heisele, T. Serre, S. Mukherjee, and T. Poggio, "Feature reduction and hierarchy of classifiers for fast object detection in video images," *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 2, p. 7, 2001.

[33] R. W. Swiniarski and R. W. Swiniarski, "Rough sets methods in feature reduction and classification," *International-Journal-of-Applied-Mathematics-and-Computer-Science*, vol. 11, no. 3, pp. 565–582, 2001.

[34] B. V. Nguyen, D. Pham, T. D. Ngo, D. D. Le, and D. A. Duong, "Integrating spatial information into inverted index for large-scale image retrieval," in *Proceedings - 2014 IEEE International Symposium on Multimedia, ISM 2014*, 2015.

[35] D. M. Chen, S. S. Tsai, V. Chandrasekhar, G. Takacs, R. Vedantham, R. Grzeszczuk, and B. Girod, "Inverted index compression for scalable image matching," in *Data Compression Conference Proceedings*, 2010.

[36] R. Negrel, D. Picard, and P.-H. Gosselin, "DIMENSIONALITY REDUCTION OF VISUAL FEATURES USING SPARSE PROJECTORS FOR CONTENT-BASED IMAGE RETRIEVAL,"

[37] D. G. Lowe, "Object recognition from local scale-invariant features," *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, no. [8, pp. 1150–1157, 1999.

[38] R. Gopalan, "Hierarchical Sparse Coding With Geometric Prior For Visual Geo-location,"

[39] M. M. Rahman, B. C. Desai, and P. Bhattacharya, "Medical image retrieval with probabilistic multi-class support vector machine classifiers and adaptive similarity fusion," *Computerized Medical Imaging and Graphics*, vol. 32, no. 2, pp. 95–108, 2008.

[40] J. Li, N. Allinson, D. Tao, and X. Li, "Multitraining support vector machine for image retrieval.," *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, vol. 15, no. 11, pp. 3597–601, 2006.

[41] R. Datta, D. Joshi, J. Li, and J. Z. Wang, "Image Retrieval: Ideas, Influences, and Trends of the New Age," *ACM Computing Surveys*, vol. 40, no. 2, pp. 1–60, 2008.

[42] A. Torralba, R. Fergus, and W. T. Freeman, "Tiny images," 2007.