

# **FAST EMBEDDING FOR IMAGE CLASSIFICATION & RETRIEVAL AND ITS APPLICATION TO THE HOSTEL INDUSTRY**

A thesis submitted for the degree of Brunel Integrated Doctor of

Philosophy by

Chanattra Ammatmanee

Department of Electronic and Computer Engineering,  
Brunel University London

## **Abstract**

Content-based image classification and retrieval are the automatic processes of taking an unseen image input and extracting its features representing the input image. Then, for the classification task, this mathematically measured input is categorized according to established criteria in the server and consequently shows the output as a result. On the other hand, for the retrieval task, the extracted features of an unseen query image are sent to the server to search for the most visually similar images to a given image and retrieve these images as a result. Despite image features could be represented by classical features, artificial intelligence-based features, Convolutional Neural Networks (CNN) to be precise, have become powerful tools in the field. Nonetheless, the high dimensional CNN features have been a challenge in particular for applications on mobile or Internet of Things devices. Therefore, in this thesis, several fast embeddings are explored and proposed to overcome the constraints of low memory, bandwidth, and power. Furthermore, the first hostel image database is created with three datasets, hostel image dataset containing 13,908 interior and exterior images of hostels across the world, and Hostels-900 dataset and Hostels-2K dataset containing 972 images and 2,380 images, respectively, of 20 London hostel buildings. The results demonstrate that the proposed fast embeddings such as the application of GHM-Rand operator, GHM-Fix operator, and binary feature vectors are able to outperform or give competitive results to those state-of-the-art methods with a lot less computational resource. Additionally, the findings from a ten-year literature review of CBIR study in the tourism industry could picturize the relevant research activities in the past decade which are not only beneficial to the hostel industry or tourism sector but also to the computer science and engineering research communities for the potential real-life applications of the existing and developing technologies in the field.

## Acknowledgements

### **My Supervisory team**

I would like to acknowledge and thank Dr Lu Gan, principal supervisor, for her energetic support, kindness and great mentoring throughout my PhD (Doctor of Philosophy) study and prior, Dr Tatiana Kalganova for her continuous support and being an inspiration, and Dr Hongying Meng for his guidance and valuable advice.

### **My Family**

I would like to thank my family for their caring upbringing, enthusiastic support, and being understanding of all my decisions.

### **My Friends and Colleagues**

I appreciate all friends and colleagues for their willingness to support and useful discussion.

## Abbreviations

ACC	Auto Colour Correlogram
ACCC	Auto Colour Correlogram and Correlation
ACM	Association for Computing Machinery
AI	Artificial Intelligence
ANIRs	Affine Noisy Invariant Region
ANN	Artificial Neural Network
ANOVA	Analysis of Variance
ASMK	Aggregated Selective Match Kernels
BMP	Bitmap image file
BoVP	Bag of Visual Phrases
BRISK	Binary Robust Invariant Scalable Keypoints
CASP	Critical Appraisal Skills Programme
CBE	Circular Binary Embedding
CBIR	Content-Based Image Retrieval
CLD	Colour Layout Descriptor
CMY	Cyan, Magenta, and Yellow
CMYK	Cyan, Magenta, Yellow, and Black
CNN	Convolutional Neural Network
CPU	Central Processing Unit
DBPSP	Difference Between Pixels of Scan Pattern
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DCT	Discrete Cosine Transforms
DenseNet	Densely connected convolutional Network



## ABBREVIATIONS

EASC	Embedding and Aggregation on Selective Convolution features
EHD	Edge Histogram Descriptor
FBI	Federal Bureau of Investigation
GB	Gigabyte
GCMs	Gaussian circulant matrices
GDP	Gross Domestic Product
GeM	Generalised-Mean
GHM	Golay-Hadamard matrix
GHz	Gigahertz
GIF	Graphics Interchange Format
GIST	Generalized Search Tree
GMM	Gaussian Mixture Model
GP	Genetic Programming
GPU	Graphics Processing Unit
HSH	Hierarchical Deep Hashing
HSV	Hue, Saturation, and Value
IBM	International Business Machines Corporation
ICA	Independent Component Analysis
IDF	Inverse Document Frequency
IEEE	Institute of Electrical and Electronics Engineers
IET	Institution of Engineering and Technology
i.i.d.	independently identically distributed
iOS	iPhone Operating System
IoT	Internet of Things
ITQ	Iterative Quantization

## ABBREVIATIONS

JLL	Johnson-Lindenstrauss lemma
JLT	Johnson-Lindenstrauss Transform
JPEG	Joint Photographic Experts Group
kNN	k-Nearest Neighbour
Lab	Lightness, a, and b
LDA	Linear Discriminant Analysis
LLE	Locally Linear Embedding
LSH	Locality Sensitivity Hashing
LTrPs	Local tetra patterns
M	Million
mAP	mean Average Precision
MATLAB	Matrix laboratory
MB	Megabyte
MMHG	Multimodal Hypergraph
MSER	Maximally Stable Extremal Regions
NASNet-Mobile	Neural Search Architecture Network-Mobile
ORB	Oriented FAST and Rotated BRIEF
PCA	Principle Components Analysis
PDF	Probability Density Function
PNG	Portable Graphics Format
PhD	Doctor of Philosophy
P2B	Pixels to Binary
QBIC	Query By Image Content

## ABBREVIATIONS

RAM	Random Access Memory
RandS	Random-Sampling operation
ResNet	Residual Neural Network
RGB	Red, green, and Blue
SBIR	Semantic-Based Image Retrieval
SCD	Scale Colour Descriptor
SfM	Structure-from-Motion
SIFT	Scale-Invariant Feature Transform
STEM	Science, Technology, Engineering, and Math
SSDH	Supervised Semantics-preserving Deep Hashing
SURF	Speeded Up Robust Features
SVM	Support Vector Machine
TBIR	Text-Based Image Retrieval
t-SNE	t-distributed Stochastic Neighbour Embedding
UK	United Kingdom
VGG	Visual Geometry Group
WHT	Walsh Hadamard Transform
YHA	Youth Hostels Association

## Table of Contents

	Page
Abstract.....	1
Acknowledgements.....	2
Abbreviations.....	3
Table of Contents.....	7
List of Figures.....	9
List of Tables.....	12
Declaration.....	15
Chapter 1: Introduction.....	16
1.1 Content-Based Image Classification and Retrieval.....	16
1.2 Hostel Search and the Application of CBIR & Classification Systems..	18
1.3 Existing CBIR Systems, Promises, and Challenges.....	20
1.4 Motivation and Research Question.....	25
1.5 Aim and Objectives.....	26
1.6 Contributions of This Thesis.....	27
1.7 Thesis Outline.....	28
Chapter 2: Literature Review.....	29
2.1 Existing Approaches for CBIR and Classification.....	29
2.2 Features Representing Image Content and Example Hostel Images....	41
2.3 Quantization of Image Features.....	49
2.4 Dimensionality Reduction of Image Features.....	52
2.5 Similarity Measures and Performance Measurement.....	56
2.6 Summary.....	60
Chapter 3: Quantization effect on General Image Retrieval and Classification....	62
3.1 Uniform Scalar Quantization.....	62
3.2 Non-Uniform Scalar Quantization.....	65
3.3 Performance Comparison.....	66
3.4 Summary.....	76

# TABLE OF CONTENTS

Chapter 4: Fast Dimensionality Reduction for General Image Retrieval and Classification.....	77
4.1 Existing Fast Dimensionality Reduction Methods.....	77
4.2 Proposed Fast Dimensionality Reduction Methods.....	83
4.3 Performance Comparison.....	88
4.4 Summary.....	117
Chapter 5: Hostel Image Database.....	119
5.1 Hostel image dataset.....	119
5.2 London hostel building datasets.....	124
5.3 Summary.....	128
Chapter 6: Hostel Image Classification.....	130
6.1 Experiment settings.....	130
6.2 Results of Proposed Fast Embedding.....	138
6.3 Summary.....	147
Chapter 7: Hostel Image Retrieval.....	148
7.1 Experiment settings.....	148
7.2 Results of Proposed Fast Embedding.....	156
7.3 Summary.....	164
Chapter 8: Conclusions and Future Work.....	165
8.1 Thesis Summary.....	165
8.2 Main Findings and Conclusions.....	167
8.3 Future Work.....	171
References.....	174
Prior Dissemination.....	193

## List of Figures

	Page
Figure 1.1: The process of content-based image classification.....	16
Figure 1.2: The process of content-based image retrieval.....	17
Figure 1.3: The example of Google's Search By Image for hostel search.....	21
Figure 1.4: The example of Mitro's system for hostel search after cosine similarity comparison.....	23
Figure 2.1: Image samples of INRIA holidays database.....	37
Figure 2.2: An example of colour histogram in the RGB colour system.....	42
Figure 2.3: The example of hostel search after colour feature comparison.....	42
Figure 2.4: An example of co-occurrence matrices.....	43
Figure 2.5: The example of hostel search after texture feature comparison.....	43
Figure 2.6: An example of a Fourier transform.....	44
Figure 2.7: The example of hostel search after shape feature comparison.....	45
Figure 2.8: The example of hostel search after spatial feature comparison....	45
Figure 2.9: Quantization process.....	49
Figure 2.10: Quantization step in the content-based image classification process.....	51
Figure 2.11: Quantization step in the content-based image retrieval process..	51
Figure 2.12: A comparison between feature extraction and feature selection..	52
Figure 2.13: Dimensionality reduction step in the content-based image Classification process.....	55
Figure 2.14: Dimensionality reduction step in the content-based image retrieval process.....	55
Figure 2.15: Euclidean distance.....	56
Figure 2.16: Hamming distance = 4 (No. of bit difference).....	57
Figure 3.1: Uniform mid-rise staircase quantizer.....	64
Figure 3.2: Uniform mid-tread staircase quantizer.....	64

## LIST OF FIGURES

Figure 3.3: Non-uniform mid-rise staircase quantizer.....	65
Figure 3.4: SIFT feature extraction and matching result.....	66
Figure 3.5: Hostel image examples of each class.....	68
Figure 3.6: No. of matched SIFT features after image quantization on 128-dimensional vector.....	68
Figure 3.7: The elapsed time results of image quantization on 128-dimensional vector.....	69
Figure 3.8: A recall comparison of four operators on Caltech 101 dataset using SqueezeNet network with multi-bit quantization.....	72
Figure 3.9: A recall comparison of four operators on Caltech 101 dataset using VGG19 network with multi-bit quantization.....	74
Figure 4.1: The covariance matrix.....	80
Figure 4.2: An $n \times n$ circulant matrix $C$ .....	81
Figure 4.3: A diagonal matrix $D$ .....	81
Figure 4.4: Ordering schemes.....	84
Figure 4.5: The 64 basis functions of an $8 \times 8$ matrix.....	85
Figure 4.6: No. of matched SIFT features after dimensionality reduction at 16-bit quantization.....	89
Figure 4.7: The elapsed time results of dimensionality reduction at 16-bit quantization.....	89
Figure 4.8: An example of sorted Haar-wavelet magnitudes of a feature vector extracted from ResNet50.....	97
Figure 5.1: Hostel image examples of each type of hostel.....	120
Figure 5.2: Hostel image examples of the hostel image dataset.....	122
Figure 5.3: Image examples of London hostel building datasets.....	126
Figure 6.1: CNN model architecture.....	130
Figure 6.2: CNN transfer learning for image classification.....	133
Figure 6.3: Hostel image examples of each class.....	135

## LIST OF FIGURES

Figure 6.4: The implementation of transfer learning for image classification task.....	136
Figure 6.5: An accuracy comparison of 11 pretrained CNNs after transfer learning at different epoch.....	140
Figure 6.6: An accuracy and training time comparison of 11 pretrained CNNs after transfer learning.....	141
Figure 6.7: An accuracy comparison of DenseNet121 & DenseNet201 on training set & validation set.....	145
Figure 7.1: CNN model architectures in a CBIR process.....	149
Figure 7.2: The implementation of transfer learning for image retrieval task.....	154
Figure 7.3: Image examples of the worst results retrieved.....	161



## List of Tables

	Page
Table II.I: The comparative study of CBIR in the tourism discipline.....	32
Table III.I: SqueezeNet and VGG19 models information.....	70
Table III.II: Comparison of the number of random coefficients and number of operations between different operators.....	71
Table IV.I: Walsh Function Values.....	83
Table IV.II: MobileNetv2, GoogLeNet, and ResNet 50 models information.....	91
Table IV.III: A recall comparison of four operators on Caltech 101 and Caltech 256 datasets using MobileNetv2, GoogLeNet, and ResNet 50 networks .....	93
Table IV.IV: Comparison of the number of random coefficients and computation costs for different operators.....	100
Table IV.V: Comparison of mean Average Precision (mAP) for different dimensionality reduction operators. For RParis 6k and ROxford 5k, “E”, “M” and “H” represent the easy, medium, and hard subsets, respectively. Bold values indicate the best results for a given M and dataset.....	102
Table IV.VI: A mAP comparison of binary code retrieval with supervised and unsupervised hashing methods on Paris6k and Oxford 5k datasets. Bold values indicate the best results for a given M and dataset.....	104
Table IV.VII: Multi-class classification accuracy (%) on GoogLeNet features for Caltech101 and Caltech256 datasets. The best mean accuracy is highlighted in bold.....	105
Table IV.VIII: Comparison of the number of random coefficients and computation costs between the RandS operator and the state-of-the-art operators. In the case of ITQ, it is too high in complexity for this comparison.....	109

## LIST OF TABLES

Table IV.IX: Multi-class classification accuracy (%) on MobileNet-v2 features for Caltech101, Caltech256 and Places356-subset datasets. The best mean accuracy is highlighted in bold.....	110
Table IV.X: A mAP comparison of compact representations in symmetric and asymmetric settings on Oxford5K, Oxford105K, Paris6K, and Paris106K datasets using fine-tuned VGG network. The highest mAP are highlighted in bold.....	114
Table V.I: Quantitative information of hostel image dataset.....	123
Table V.II: Image quantity of Hostels-900 and Hostels-2K datasets, London hostel building datasets.....	127
Table VI.I: CNN models information.....	132
Table VI.II: An accuracy comparison of SqueezeNet1_0 on validation set & test set at different data split. Bold values indicate the highest accuracy and the optimal option is highlighted.....	138
Table VI.III: An accuracy comparison of SqueezeNet1_0 on validation set & test set at different dropouts at 90:10 and 70:30 data split. Bold values indicate the highest accuracy and the optimal option is highlighted.....	139
Table VI.IV: An accuracy and training time comparison of 11 pretrained CNNs after transfer learning. Bold values indicate the best performance and the optimal options are highlighted.....	141
Table VI.V: An accuracy comparison of 11 pretrained CNNs after transfer learning on validation set & test set. Bold values indicate the highest accuracy and the optimal options are highlighted.....	143
Table VI.VI: An accuracy comparison of 11 pretrained CNNs on original size database & resized database at 10 epochs. Bold values indicate the best performance and the optimal options are highlighted.....	144

## LIST OF TABLES

Table VII.I: Chosen CNN models information.....	151
Table VII.II: An accuracy comparison of four pretrained CNN-based retrieval systems on four datasets. Bold values indicate the highest accuracies, and the best-performing system is highlighted. Additional comparison to Radenovic, Tolias, and Chum’s results are provided.....	156
Table VII.III: An accuracy comparison of four pretrained CNN-based retrieval systems when using the default value and the binary value of feature vectors. Bold values indicate the highest accuracies, and the best-performing system is highlighted.....	158
Table VII.IV: An accuracy comparison of ResNet152 when retrieving the best results and the worst results on Hostels-900 and Hostels-2K datasets.....	160
Table VII.V: An accuracy comparison of ResNet152 when retrieving the worst results and the 2 <sup>nd</sup> worst results on Hostels-900 and Hostels-2K datasets using default value and binary value of feature vectors.....	160

## Declaration

I hereby declare that this thesis is my original research work. Sources of others' contributions are referenced within the text and appended in the bibliography. This work has not been submitted for any other degree at Brunel University or any other educational institution. Published parts of this work are identified in the prior dissemination section which all contents have been undertaken during this PhD study.

## Chapter 1: Introduction

### 1.1 Content-Based Image Classification and Retrieval

Looking at content-based image classification, it is the process of taking an unseen input which is the image contents rather than the metadata such as keywords, tags, or other ways of image descriptions and extracting its features representing the input image which could be an object or type of land. Then, this mathematically measured input is categorized according to established criteria in the server and consequently shows the output as a result. Furthermore, the image classification is often found as a supervised learning problem in which a set of target classes are firstly defined and the model is then trained to recognize them using labelled images. The classification process is illustrated in figure 1.1 below.

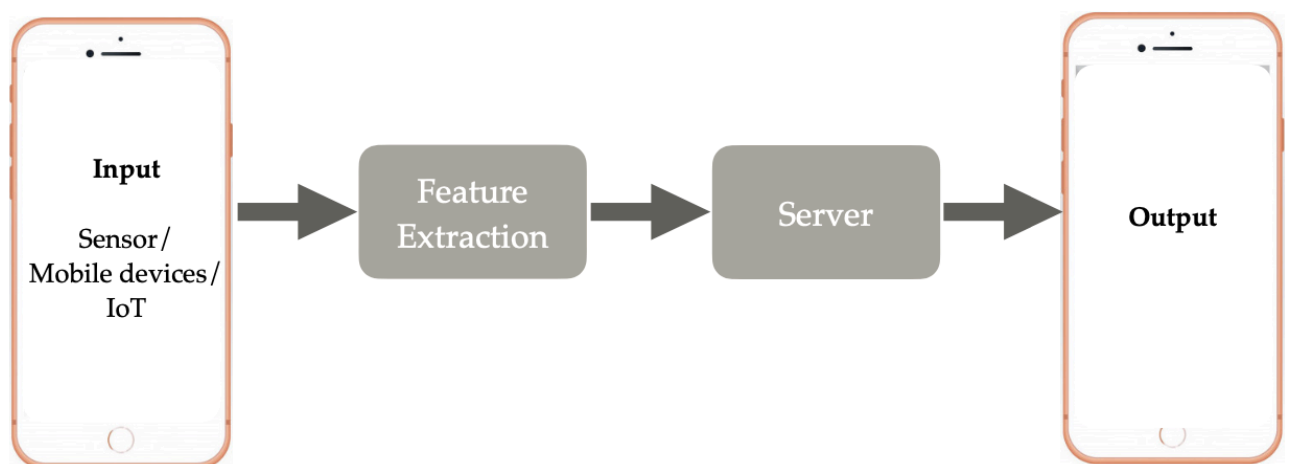


Figure 1.1: The process of content-based image classification

Moving to content-based image retrieval (CBIR), it is also known as query by image content which is the process of taking image contents as the unseen query image, extracting its features, and sending it to the server which contains a large database of digital images. Then, search for the most visually similar images to a given image and retrieve these images as a result. Moreover, in many cases, the image retrieval is an unsupervised learning problem meaning the model analyses the unseen images in the

database and discovers the patterns on its own. Then, comparing these patterns to the patterns of the given query image in order to match and retrieve the closest images. The retrieval process is shown in figure 1.2 below.

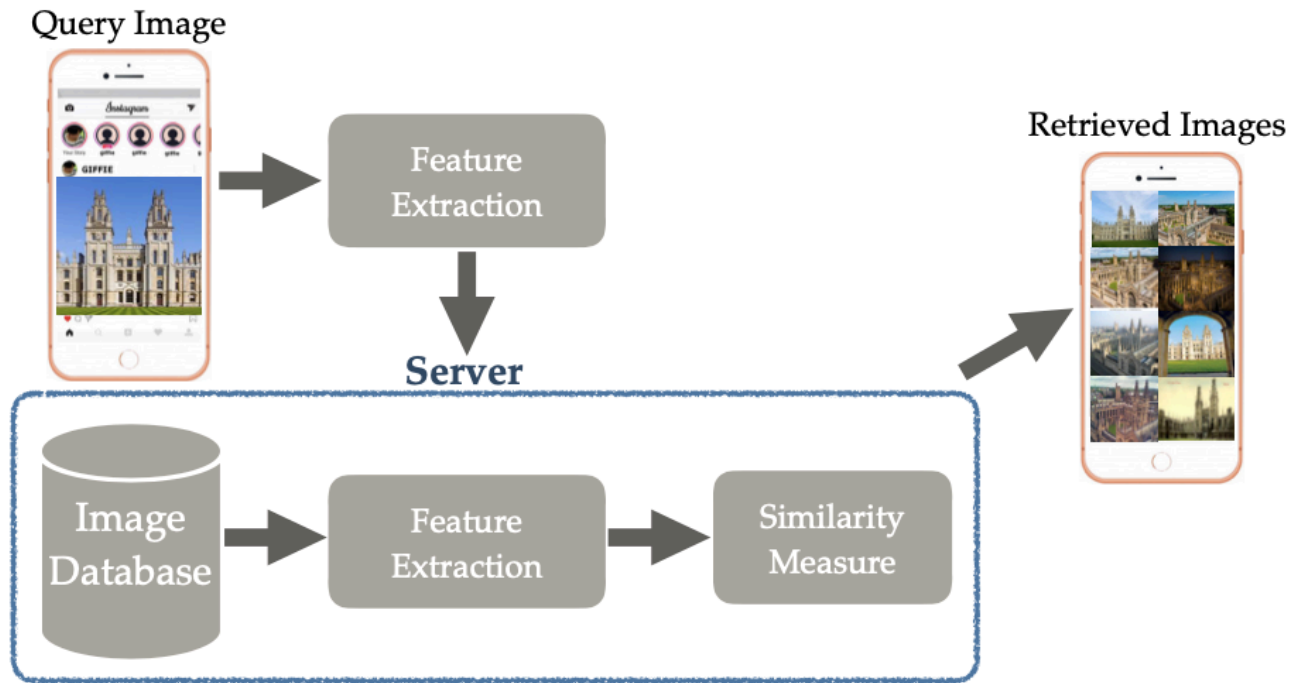


Figure 1.2: The process of content-based image retrieval

In addition, an image query/input could be in the form of sketch or colour map, however, image example would be the most appropriate image type for content-based image classification and retrieval as it has well-represented attributes of an image such as the presence of a particular combination of colour, shape, and/or texture, the presence or arrangement of the object(s) in an image, and the presence of location(s), event(s), or individual(s). It can be seen that the content-based technique could be a more challenging technique compared to the semantic-based technique and text-based technique which use words as tools of communication and are logically structured by a human [1]. However, by using the content-based technique, an image classification could achieve high accuracy and the closest image(s) could be retrieved for image retrieval with less human errors and less labour cost in these image indexing and organization.

## 1.2 Hostel Search and the Application of CBIR & Classification Systems

The tourism industry has been one of the driving forces of the economy for many countries around the world [2]. In 2019, the industry contributed 8.9 trillion US Dollars to the world's gross domestic product (GDP), accounted for 10.3 per cent of global GDP, and provided 330 million jobs, one in ten jobs, around the world. Moreover, 948 billion US Dollars in capital investment was allocated to the industry. Additionally, despite the service nature of tourism, it can be seen as a tourism product which consists of five elements including attractions, access, accommodation, amenities, and activities [3] [4] [5]. As a consequence, accommodation as a major part of the tourism product had also increased in number. Particularly, accommodations targeting on millennial generation became a focus for world-dominant hoteliers such as AccorHotels, Hilton, and Marriott [6] [7] [8] [9]. On top of this, major hostel chains such as A&O, Generator, Meininger, and St Christopher's Inns had expanded their business while there was an increasing number of independent hostels in different parts of the world [10] [11].

In terms of sources for accommodation search, online travel agencies have played major roles as a fare aggregator and a reservation portal as they facilitate and provide decision-making information about a number of major hostel chains such as A&O, Generator, Meininger, and St Christopher's Inns and countless of the independent hostels. Some of the key players are [www.hostelz.com](http://www.hostelz.com), providing a list of almost 50,000 hostels in over 7,000 cities, [www.hostelworld.com](http://www.hostelworld.com), containing a database of over 36,000 properties, [www.hostels.com](http://www.hostels.com) and [www.booking.com](http://www.booking.com), listing over 30,000 hostels on their sites. Despite the fact that the hostel market considers its images as a critical factor as images portrayed its facilities and unique atmosphere which highly influence the decision-making process of potential guests when choosing an accommodation [12] and the fact that digital image sharing on social media via smartphone has become a phenomenon in today's society, 350 million images are uploaded every day on Facebook, more than 50 billion images so far have been uploaded to Instagram, and over 1.2 billion images and videos are uploaded on Google Photos [13] [14] [15] [16], these e-intermediaries mainly rely on the text-based

method by using keywords to retrieve the results of each search and has no sign of CBIR application onto their search systems. Furthermore, their additional and crucial challenge is the management of the vast and growing collections of digital images. For example, on booking.com, travellers upload more than 150 million images onto the system in order to share their experiences during their stay [17]. Therefore, to manually or automatically classify this enormous amount of images in their database and systematically organize them for future use such as image retrieval or image analysis could be costly and time-consuming, let alone correctly classify these images. In addition, by considering a hostel image, a hostel interior image tends to have more complex clusters of image contents and a hostel building image could be less visible when compared to a hotel image. This leads to more difficulty in image classification/organization. Having said that, an image would be the appropriate tool for accommodation search as 'a picture is worth a thousand words'.



## 1.3 Existing CBIR Systems, Promises, and Challenges

There is evidence that CBIR has been applied to many industries such as for crime prevention purposes by the Federal Bureau of Investigation (FBI) [18] and immigration offices in many countries using this image retrieval technique for automatic fingerprint matching, and the Los Angeles police department [19] and immigration control across the world use software designed for face recognition.

As for museum management purposes, IBM (International Business Machines Corporation)'s QBIC (Query By Image Content) system was applied to the website of the Hermitage Museum in Russia [20]. Therefore, people are able to search and appreciate the precious artworks without being at the museum.

Additionally, the application of CBIR can be found for medical diagnosis purposes in healthcare, intellectual property protection for trademark owners, architectural or engineering design for related companies, fashion or interior design for relevant enterprise, image library in journalism and advertising, cultural heritage documentation for museums and art galleries, and self-study courses of academia and training providers [18].

Furthermore, three of the key developers have launched the search by image constructs, Search By Image by Google, TinEye by Idee, and Yandex by Yandex, for the general public. On top of these, mobile phone giants adopt an Artificial Intelligence-powered technology that uses a smartphone camera and deep machine learning for object recognition and image retrieval using only visual content on their mobile apps such as Bixby by Samsung which allows mobile users to point their cameras at an object in order to retrieve similar items with additional information such as shops that sell these items, product features, and online deals [21]. Similarly, HiVision by Huawei, and Google Lens by Google which is now compatible with many iPhones operating systems (iOS) and Android devices are also available [22] [23]. Having said this, retrieval accuracy and computational cost remain major challenges as object recognition and retrieval require more in-depth studies on smaller-scale datasets and

# CHAPTER 1: INTRODUCTION

in return contribute to more effective image search using visual features in response to a user-specified query. Therefore, there is a large room for improvement in this field within academic research communities, as well as commercial research, and its real-life applications in our society.

To briefly test existing CBIR systems on hostel images, simulation results from two systems, Google's Search By Image and Mitro's system [251], are illustrated below.

To begin with **Google's Search By Image**, the system is primarily based on colour and shape features. Additionally, it makes a text-based guess on the image query as a part of image identification which results in returned images.


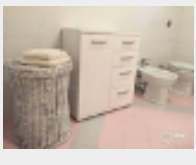
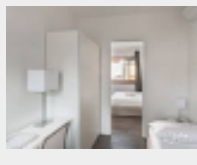
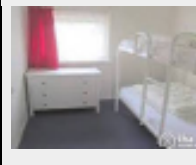
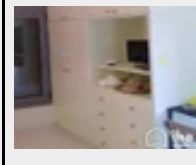
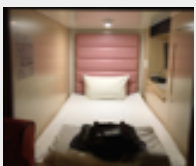
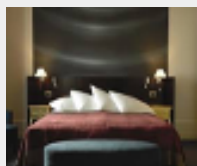

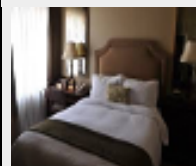
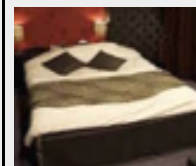
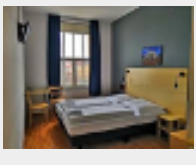
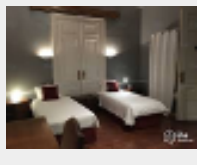
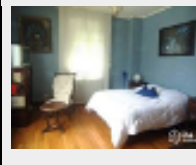
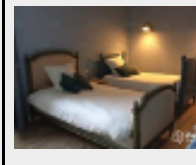
Image Query	Visually Similar Images					
						
<b>Away Hostel, Lyon, France</b>						
						
<b>Y's Cabin Hostel, Yokohama, Japan</b>						
						
<b>A&amp;O Hostel, Leipzig, Germany</b>						

Figure 1.3: The example of Google's Search By Image for hostel search

Concerning colour features, most of the returned images are visually similar to each hostel image query. However, when it comes to the shape features, in spite of similar

shapes, some objects in each image are different from each other. For instance, in the case of Away hostel image, instead of retrieving images with a two-layer bunk bed, the results show images of cupboards or chests of drawers. On top of these, despite the fact that the system has a text-based guess for image identification, two of three hostel image queries, Y's Cabin and A&O which both are chain hostels, are retrieved as ones of the returned images and the system is not able to retrieve the same image as the image query for Away hostel which is an independent hostel. Moreover, some of the images retrieved are not the bedroom images of hostels but the toilet or living room or possibly other types of accommodation instead.

Therefore, according to the results in Figure 1.3, Google's Search By Image is not yet applicable for a hostel search due to its image database that contains a variety of image categories which is not specifically designed for a unique purpose like a hostel search. Additionally, there is room to improve the system especially on its shape features extraction, image retrieval techniques used, and Google's hostel image database. Nevertheless, the retrieval speed could be considered acceptable quick for a large image database.

Mitro's CBIR system focuses on two classical features, colour and texture features. By using a subset of this PhD project's image dataset as a database, 500 hostel interior images, as well as comparing cosine similarity of the hostel image query and the hostel images in the database, the results of hostel search are shown as follow.

# CHAPTER 1: INTRODUCTION





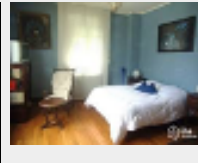
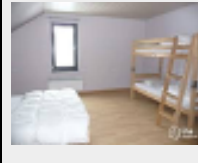
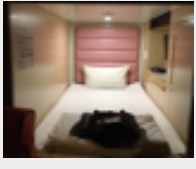
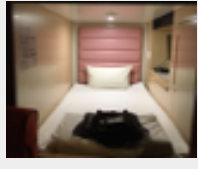
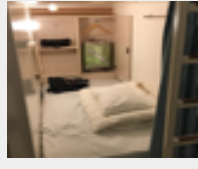


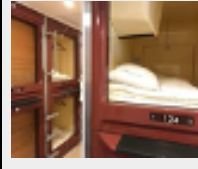
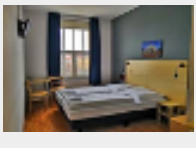
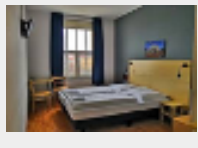


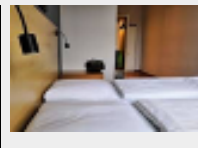
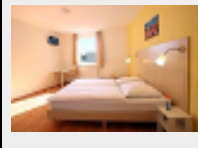
Image Query	Retrieved Images				
					
<b>Away Hostel, Lyon, France</b>					
					
<b>Y's Cabin Hostel, Yokohama, Japan</b>					
					
<b>A&amp;O Hostel, Leipzig, Germany</b>					

Figure 1.4: The example of Mitro's system for hostel search after cosine similarity comparison

Based on colour features, it can be seen that despite using classical and basic features like colour features, the retrieved images contain a variety of colours some of which do not appear on the image query. Moving to the texture features, Mitro's system could be considered quite an effective system in terms of performance as the floor textures on most retrieved images are similar to the floor textures on the image query. Moreover, the wall textures on most retrieved images are as plain as the image query which has no pattern on the wall.

Consequently, according to the results in Figure 1.4, Mitro's system has a potential for hostel search as it has strengths in the techniques used for its texture features extraction and the image queries are retrieved as ones of the returned images after comparing cosine similarity. Having said this, colour features extraction needs to be improved as colours in a hostel image reflect different moods and tones of hostel interior designs which result in different preferences of each potential hostel guest. Additionally, bunk bed images are retrieved instead of single/double bed images.

Therefore, shape features could be considered for the purpose of improving the system in particular for hostel images.

Despite the strengths of Google's Search By Image and Mitro's system and their potential for hostel search, it is undeniable that the databases remain a critical challenge, especially Google's. As it is not only about the size of the database or the types of images in the database but each image in the database also could be labelled in order to potentially increase retrieval accuracy.

## 1.4 Motivations and Research Question

The inspiration of this research project initiates from the researcher's observation on available online travel agencies for hostel search on smartphone and their lack of CBIR technique used which is considered state-of-the-art knowledge. It is a surprising result of observation, in spite of the importance of the accommodation sector for the tourism industry, especially in the age of the digital image. Furthermore, to the best of my knowledge, the CBIR technique study or application, or even text-based image retrieval (TBIR) and semantic-based image retrieval (SBIR) techniques [24] in relation to accommodation, in general, are rarely found and there is none of the hostel market, let alone content-based image classification. For these reasons, the researcher wants to pursue research that transfers the knowledge in the computer science and engineering discipline to the tourism discipline by developing the first hostel-oriented algorithm using fast computable and memory-efficient embeddings with novel techniques for content-based image classification and retrieval on low-computational devices such as mobile phone and Internet of things (IoT). On top of this, personal interest in the hostel market also contributes to the inspiration for this project.

As a result, a research question of this project is **Which fast embeddings are state-of-the-art and applied in content-based image classification and retrieval for the hostel search on low-computational devices?**

## 1.5 Aim and Objectives

According to the research gap and personal interest aforementioned, the **aim of this work** is to develop fast computable and memory-efficient embeddings for content-based image classification and retrieval on low-computational devices such as mobile phones and the Internet of things (IoT) and pioneeringly apply these novel techniques to the hostel industry.

In addition, the researcher has addressed **six research objectives** as follows.

- To carry out relevant literature review including types of features representing image content, the quantization of image features, the state-of-the-art dimensionality reduction operators, the existing CBIR and classification approaches in computer science and engineering discipline and tourism discipline, and similarity measures and performance measurement.
- To familiarize with relevant programming languages such as MATLAB and Python, as well as interactive environments such as MATLAB and Google Colab.
- To gather the first hostel image datasets, hostel image dataset containing 13,908 interior and exterior images of hostels across the world, and Hostels-900 dataset and Hostels-2K dataset containing 972 images and 2,380 images, respectively, of 20 London hostel buildings.
- To implement and evaluate available algorithms in relation to dimensionality reduction and quantization.
- To develop and evaluate the fast computable and memory-efficient operators for content-based image classification and retrieval.
- To apply and evaluate the developed techniques on the collected hostel image datasets.

By fulfilling the above objectives, the researcher hopes to achieve the aim of this research project and simplify our way of hostel search in the future.

## 1.6 Contributions of This Thesis

In response to the objectives aforementioned, the contributions of this thesis are addressed below.

- Creating the first hostel image database which contains three datasets of hostel images. The first dataset, the hostel image dataset, consists of hostel interiors, such as images of a bedroom, kitchen, and laundry room, and exterior, such as garden and facade, for image classification experiments. The other two datasets, Hostels-900 dataset and Hostels-2K dataset consisting of 20 London hostel buildings for image retrieval experiments. These images represent independent hostels which are usually privately owned and have unique designs and chain hostels which are corporately owned and usually have similar designs.
- Conducting and presenting a systematic evaluation of the quantization effect using uniform scalar quantizer on content-based image classification and retrieval.
- Conducting and presenting a systematic evaluation of the fast dimensionality reduction operators using Hadamard Projection and Discrete Cosine Transform, against Random Projection, Principal Component Analysis, and Circular Binary Embedding.
- Applying and evaluating the developed techniques on the collected hostel image datasets for image classification and retrieval.

The overall intention is to develop fast computable and memory-efficient embeddings for content-based image classification and retrieval on devices with low memory, bandwidth, and power such as mobile phones and the Internet of things (IoT) and pioneeringly apply these novel techniques to the hostel industry.



## 1.7 Thesis Outline

**Chapter Two:** This chapter presents a literature review conducted in relevant areas including existing approaches for CBIR and classification, features representing image content, dimensionality reduction and quantization of image features, similarity measures, and performance measurement.

**Chapter Three:** This chapter studies the quantization effect using uniform scalar quantizer and dithering-based scalar uniform quantizer on general image classification and retrieval.

**Chapter Four:** This chapter addresses the state-of-the-art techniques of fast dimensionality reduction including Random Projection, Principal Component Analysis, Circular Binary Embedding, Hadamard Projections, and Discrete Cosine Transforms. The performance comparison between the proposed approaches and the existing methods for general image retrieval and classification is demonstrated.

**Chapter Five:** This chapter is devoted to the first hostel database, consisting of hostel image dataset and London hostel building datasets, which is newly collected for this research project. Data collection including data sources, data selection, and data pre-processing are illustrated.

**Chapter Six:** This chapter presents the application of the proposed fast embeddings on hostel image classification task using hostel image dataset.

**Chapter Seven:** This chapter applies the proposed fast embeddings on hostel image retrieval task using Hostels-900 dataset and Hostels-2K dataset.

**Chapter Eight:** This chapter summarizes the main findings of this research, draws conclusions, and highlights the potential future work

# Chapter 2: Literature Review

## 2.1 Existing Approaches for CBIR and Classification

Due to being interdisciplinary research of this study, this literature review section has addressed relevant literature in both computer science and engineering discipline and the tourism discipline.

### I. Approaches for CBIR and classification in Computer Science and Engineering Discipline

To begin with the existing approaches for CBIR, as a nature of computer science and engineering discipline mainly focuses on understanding, designing, and developing programmes in relation to computer-oriented subjects, therefore the available literature in CBIR topic is dominantly on advancing the existing feature extraction techniques and the existing image retrieval techniques in order to create more efficient tools rather than apply the techniques to particular fields of study or particular industries [25].

For instance, the studies that propose novel methods for feature extraction [26][27][28][29], the study that propose improved method for feature extraction [30], the study that propose novel method for image retrieval [28], and the studies that propose improved methods for image retrieval [25][31][32][33][34][35][36][37][38][39].

In addition, the studies that compare the CBIR technique to the text-based or semantic-based image retrieval techniques [24][40][41][42][43] are also often found.

Nevertheless, to emphasize the validity of suggested approaches, the transparency of research experiments and the improvement on performance measurements are the crucial elements which should not be neglected.

Moving to the content-based image classification literature in the discipline, the significant contribution in this area is from ImageNet Large Scale Visual Recognition Challenge (ILSVRC) which is a competition for object detection and image classification at a large scale since 2012. Many winners developed novel CNN architectures which are widely used these days including AlexNet [44], GoogLeNet [45], and ResNet [46]. In addition, several CNN models newly developed by the runners up are also popular, particularly VGG [47]. Outside the competition, novel CNN models are created such as SqueezeNet [48] and DenseNet [49]. These models were trained on an ImageNet dataset containing over 14 million images with 1,000 object classes. To provide alternative pretrained CNN models on other large scale image datasets for image classification, Zhou *et al.* [50] created Places356 dataset containing 1.8 million images with 356 scene categories which is a scene-centric dataset, unlike ImageNet which is an object-centric dataset. Then, off-the-shelf AlexNet, off-the-shelf GoogLeNet, and off-the-shelf VGG were trained from the scratch on Place356 dataset. Furthermore, the transfer learning technique was explored by Shaha and Pawar [51] in order to make the most of transferable parameters of the pretrained CNNs. Moreover, various experiments in the computer science and engineering discipline have been focusing on improving the accuracy of image classification through improved methods for feature extraction. For example, the evaluation of Individual units in CNN via Ablation [52], the evaluation of invariance properties and distinctiveness of colour descriptors [53], the application of OverFeat image features extractor [54], the application of Scale-Invariant Feature Transform [55][56], the use of K-nearest Neighbours(NN) [39][55], the implementation of Co-Occurrence Matrix [30][37][57], Edge Histogram Descriptor (EHD) and Color Layout Descriptor (CLD) [58], and Deep CNN with Max-Pooling [59].

On top of this, an additional literature review for specific experiments is addressed in Chapter 3, Chapter 4, Chapter 6, and Chapter 7.

### II. Approaches for CBIR and classification in Tourism Discipline

**(The majority of content in this section is a peer-reviewed manuscript which is published in the Electronic Library journal)**

One of the earliest studies was conducted by Kato and Kurita in 1990 [60]. They studied visual interaction by using a visual example, a sketch, as a query in order to retrieve similar paintings in an electronic art gallery. However, their approach integrated a personal index with a pictorial index which is rather considered as semantic-based image retrieval.

In 1994 Holt and Hartwick [61] collaborated with IBM in a joint research project with the purpose of evaluating the effectiveness of QBIC (Query By Image Content), a visual query software developed by IBM. Though having a vast image database of fine art from the Art and Art History Department Slide Library and the latest knowledge at the time, many results showed inaccuracy especially when shape features were involved.

Another example of previous studies is Jain *et al.*'s work [62]. They applied the CBIR technique to access image contents of Brihadisvara temple's heritage such as scriptures, architectural drawings, and publications. However, due to the low quality of the image database as many of these materials were old and difficult to read, the data preparation process was a crucial hindrance for their image retrieval.

By considering the aforementioned projects within the tourism industry context, it can be seen that there were room for improvement in terms of the techniques used for feature extraction as well as the limited storage capacity. On top of these, the clarification of the experiments could also be available such as the technical details on feature extraction techniques and image retrieval techniques, the size and the source of image database, the comparison of different techniques used or different sets of the database used, and the performance measurement in order to assure their validity. Despite the limitations mentioned, the approaches of using a combination of features, colour and shape features, and using a combination of techniques contributed better

## CHAPTER 2: LITERATURE REVIEW

results in CBIR. Furthermore, the attempts to develop a real-life application of CBIR for public and academic domains, under limited knowledge and conditions during that time, led to more advanced attempts in following generations.

Furthermore, by focusing on the relevant literature between the year 2010 and the year 2019, a comparative study of key elements and results from each experiment are highlighted in table II.I below with critical analysis.

Author (s)	Country(s) & Publication	Image Database(s) & Format(s)	No. of Images & Classes	Extracted Feature(s)	Performance Measurement (s) & Result(s)	Summary of Findings
Premchaiswari <i>et al.</i> (2010) [63]	Thailand (IEEE)	Tourist attractions (JPEG, BMP & GIF)	3,600 (n/a)	colour correlogram	Precision (89%), recall (227), mean average precision (86%) & mean average recall (2)	The ACCC (Auto Colour Correlogram and Correlation) algorithm outperforms the ACC (Auto Colour Correlogram) algorithm for CBIR
Wenger <i>et al.</i> (2011) [64]	Switzerland & France (Proceedings of the 19th ACM international conference on Multimedia)	INRIA holidays (JPEG)	1,491(500)	colour & SIFT	Mean average precision (65.3%)	The colour SIFT descriptor outperforms the SIFT descriptor for CBIR

## CHAPTER 2: LITERATURE REVIEW

Raisi <i>et al.</i> (2011) [65]	Iran (IEEE)	own tourism database (attractions of Zahedan city and University of Sistan and Baluchistan) (n/a)	1,021 (n/a)	colour, texture, & edge	Average Normalized Modified Retrieval Rate (0.3444 for EHD) & running time (0.06s for SCD)	The EHD (Edge Histogram Descriptor) and the SCD (Scale Colour Descriptor) methods outperform the others for CBIR
Abdullahzadeh & Mohanna (2013) [66]	Iran (International Research Journal of Applied and Basic Sciences)	Building category of Corel (JPEG)	100 (n/a)	colour	Average Normalized Modified Retrieval Rate (0.0759)	The combined grey and HSV colour ANIRs (Affine Noisy Invariant Region) algorithm outperforms the others for CBIR
Raisi <i>et al.</i> (2014) [58]	Iran (International Journal of Advanced Networking and Applications)	own tourism database (attractions of Zahedan city and University of Sistan and Baluchistan) & Corel_1k (n/a)	1,000 (17) & 1,000 (20)	colour, texture, & shape	average normalized modified retrieval rate (0.2751 for EHD&CLD) & query running time (0.050s for SCD)	The combined EHD (Edge Histogram Descriptor) with CLD (Colour Layout Descriptor) and the SCD (Scale Colour Descriptor) method outperforms the others for CBIR
Zheng <i>et al.</i> (2014) [67]	China (IEEE Transactions on Image Processing)	INRIA holidays, Ukbenc, Dulmage & MIR Flickr 1M (JPEG)	1,491 (500), 10,200 (2,250), 1,104 (33) & 1M (n/a)	colour & SIFT	Mean average precision (85.2%)	The combined SIFT & colour algorithm outperforms the others for CBIR

## CHAPTER 2: LITERATURE REVIEW

Zhu <i>et al.</i> (2015) [68]	China (IEEE Transactions on Cybernetics)	Landmarks from Flickr (n/a)	5,000 (25)	colour moment, texture, shape, SIFT, & GIST	Precision (29.77%)	The MMHG (Multimodal Hypergraph) algorithm outperforms the others for CBIR
Amato <i>et al.</i> (2015) [69]	Italy (ACM Journal on Computing and Cultural Heritage)	Pisa monuments & landmarks from Flickr (n/a)	1,227 (12)	SIFT, SURF, ORB & BRISK	F1 macro (0.95)	The kNN(K-Nearest Neighbour) with SIFT algorithm outperforms the others for CBIR
Wang <i>et al.</i> (2015) [70]	Australia (Proceedings of the 23rd ACM International Conference on Multimedia)	Landmarks from Flickr, Picasa web album & Oxford building (n/a)	49,840 (55), 4,100 (16) & 5,062 (12)	shape & SIFT	Mean average precision (59.94%)	The novel method based on a multi-query expansion paradigm outperforms the others for CBIR
Makantasis <i>et al.</i> (2016) [71]	Greece & Cyprus (Multimed Tools Application)	Cultural heritage from Flickr (n/a)	31,000 (n/a)	ORB	Precision (78%), recall (92%) & F1 Score (84%)	The DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm outperforms the others for CBIR
Lacheheb & Aouat (2016) [72]	Algeria (Multimed Tools Application)	ZuBuD (PNG), WANG & Coil-100 (n/a)	1,120 (201), 1,000 (10) & 7,200 (n/a)	colour & SIFT	Precision (56%), recall (100%), F-measure (70%) & error rate (0.01)	The combined SIFT & HSV algorithm outperforms the others for CBIR

## CHAPTER 2: LITERATURE REVIEW

Elleuch & Marzouki (2017) [73]	Tunisia (Multi med Tools Application)	INRIA holidays (JPEG), Ukbenc h, MIR Flickr 1M & Flickr60K (n/a)	1,491 (500), 10,200 (2,250), 1M (n/a) & 67,714 (n/a)	colour & SIFT	Mean average precision (59.4%)	The novel multi-IDF (Inverse Document Frequency) design algorithm outperforms the others for CBIR
Lonarkar & Rao (2017) [74]	India (Proceedings of the International Conference on Inventive Computing and Informatics)	INRIA holidays (JPEG)	1,000 (500)	colour histogram	Precision (100%) & recall (33.6%)	The combined automated segmentation & region-based feature extraction algorithm outperforms the others for CBIR
Wang <i>et al.</i> (2017) [75]	Australia (IEEE Transactions on Image Processing)	Landmarks from Flickr & Picasa Web Albums (n/a)	49,840 (55) & 4,100 (16)	colour, SIFT & CNN	Mean average precision (63.62%)	The novel method based on a multi-query expansion paradigm outperforms the others for CBIR
Hung (2018) [76]	Taiwan (The Electronic Library)	Chinese paintings (n/a)	1,200 (3)	colour histogram & texture	Mean average precision (92%)	The combined colour & texture algorithm performs well for CBIR
Arun <i>et al.</i> (2019) [77]	India (Artificial Intelligence Review)	INRIA holidays (JPEG), Oxford buildings, Scene-15, GHIM-10K (JPEG), IAPR TC-12 (JPEG) & SUN-397	1,491 (500), 5,062 (11), 4,485 (15), 10,000 (20), 20,000 (n/a) & 108,754 (397)	SIFT	Mean average precision (83.6%), average R-Precision (86.7%) & discounted cumulative gain (89.9%)	The BoVP (Bag of Visual Phrases) algorithm outperforms the others for CBIR



## CHAPTER 2: LITERATURE REVIEW

Basak <i>et al.</i> (2019) [78]	India (International Research Journal of Engineering and Technology)	Monuments (n/a)	4 (2)	shape	Edge magnitude value (1.61 for 1 <sup>st</sup> image & 1.65 for 2 <sup>nd</sup> image)	The shape feature algorithm performs well for CBIR
---------------------------------	--	-----------------	-------	-------	--	--

Table II.I: The comparative study of CBIR in the tourism discipline

Looking at the recent decade, it can be seen that the trend of CBIR study in the tourism discipline is to improve existing feature extraction techniques (47%) to better represent image information and eventually accelerate the image retrieval performance with high efficiency and effectiveness. Nevertheless, some studies attempted to contribute novel techniques in the feature extraction process by fine-tuning fusion features (41%) and some attempted to improve the input of the CBIR system, image query (12%).

As for the research author(s), the majority of CBIR studies are collaborative works of two authors (24%), three authors (40%), four authors (24%), and five authors (6%). Only 6% of the CBIR study is conducted by one author. Additionally, 62% of the study is the collaboration between institutions, whereas 38% is done within an institution or most authors are from the same institution. Despite the fact that it is an interdisciplinary subject, all authors chose to publish their academic papers in journals/conferences in relation to the STEM (Science, Technology, Engineering, and Math) subjects instead of the humanities subjects such as leisure and tourism, social science, and business study. As for the origin country(s) of authors, Asia (65%) is the most active continent for tourism-related CBIR study, particularly conducted in India (17%), Iran (17%), and China (17%), Taiwan (7%), and Thailand (7%). Nonetheless, European scholars (17%), African scholars (12%), and Australian scholars (6%) also have an interest in the topic. Furthermore, even though the CBIR study in tourism has been published almost every year and there is a consistent number of published papers each year, more intensive study in the

field could have been conducted in response to the ongoing challenges in image data management/organization in the tourism industry.

Moving to the database(s) used in each experiment. Although there are five basic elements of tourism product which are attractions, accommodation, access, amenities, and activities, the tourist attraction element has been in the authors' focus, particularly in art and cultural heritage domains which are understandable as art and cultural heritage are considered the priceless treasures of humankind from generations to generations. These databases include landmark images, monument images, historical building images, beach images, mountain images, and/or painting/object images in the museum. There are fewer images of access element, bus images, amenities element, food images at restaurants, and activities element, traditional/special event images. However, images of the accommodation element were not included in these studies. On top of these, many experiments used existing tourism-related databases in which not all of them contain meaningful image labels such as INRIA holidays, Oxford buildings, and ZuBuD Zurich buildings and some experiments considered other existing databases which contain tourism-related images such as Flickr and Picasa Web Albums. Only 17% of these few experiments created their own databases by browsing images from available search engines or taking photographs at the premises.



Figure 2.1: Image samples of INRIA holidays database [79]

Furthermore, it seems there is no academic scholar who suggests a number of reliable data sizes or how to calculate the appropriate data size for research in relation to image retrieval. Therefore, the choice of data size in each experiment is subjective based on the limitations of each research, between 100 images to 1M images per existing database and between 1,000 images to 49,840 images per newly created database with various sizes of image category. Nevertheless, there is still a lack of images of accommodation element in these databases and the update of the database used is not mentioned in any study which could result in dated information when applying these studies in real-life applications.

In terms of image features which contain information of an image, as a part of metadata, it can be seen the classical features were mainly extracted and the colour and/or Scale-Invariant Feature Transform (SIFT) features were the most popular choices (53%) to represent images in the tourism context. Moreover, the approach of using a combination of features was a trend as these fusion features include significant image information on a larger scale when compared to a single feature used. Additionally, though Artificial Intelligence-based features are state-of-the-art features which are advanced and closer to human's cognitive process, their computational complexity and cost, as well as time-consuming, could be the major hindrances to apply these types of features on a small-scale study with resource and financial limitations. Having said this, there is an attempt of using Convolutional Neural Network (CNN) features to represent images.

As for the performance measurement used in the CBIR study, even though there is no single rule to evaluate the CBIR system as it depends on the user requirement for that specific task, it is undoubtedly that a combination of measures, especially precision and recall, is common in these studies. On top of this, processing time in image retrieval should not be a neglected criterion in experiments.

Considering the relevant CBIR studies above, it can be seen that there is no existing CBIR study for hostel search on a smartphone, let alone the study in hostel image classification, which reassures the research gap mentioned in the previous section.

Additionally, even though there is no evidence on content-based image classification and retrieval study for the hostel sector, to the best of my knowledge, some studies in hotel image retrieval and classification are found. For example, Stylianou *et al.* [80] suggest an end-to-end system including metric learning for hotel matching, visualization tools, and automatic report generation for sex trafficking investigation. Stylianou *et al.* [81] further conduct hotel instance recognition and hotel chain recognition experiments for human trafficking investigation which are the instance-level retrieval task but with supervised learning. ResNet (Residual Neural Network) model was implemented for hotel instance recognition which is CBIR, whereas the hotel chain recognition used VGG (Visual Geometry Group) model for content-based image classification. This work well illustrates the attempt to improve existing image retrieval and classification techniques and apply them to a real-life problem. Xuan *et al.* [82] propose an “Easy Positive” approach to create more flexible and generalizable embeddings to new unseen data. Kamath *et al.* [83] provide a new “Cross-Entropy” approach to hotel recognition for human trafficking investigation, as well as create a new hotel dataset. Tseytlin and Makarov [84] use Latent Image Embeddings on the hotel chain recognition tasks which is also an instance-level retrieval experiment with supervised learning. Another example is Kanchinadam [85] who publishes the hotel image classification results from the Kaggle InClass competition [86]. A CNN model, GoogLeNet, was implemented with transfer learning in order to classify hotel images into eight classes, bathroom, guest room, pool, gym, restaurant, lobby, aerial view, and business centre. Utilization and improvement of an existing image feature extraction technique are well-addressed in this work, though the further peer-reviewed processes could be undertaken. Furthermore, Ren *et al.* [87] conduct the hotel photo content analysis, Rath [88] investigates Kayak’s hotel image categorization with deep learning, and Tasli [17] explores the automated hotel image tagging. These studies show that machine learning with hotel image queries could be applied to real-world problems and provide meaningful results. Therefore, there is potential for its application to the hostel domain which has similar elements but greater complex clusters of image content as it is more likely that there are more than two beds in the same room in a hostel bedroom, there are more than one washing machine/dryer in a hostel laundry

room, there are more than one shower/toilet in a hostel bathroom, and there are more objects in a hostel kitchen/living room. Therefore, it is possible that hostel image features consist of vaster feature types and levels such as many colours, many textures, and many shapes which account for the higher number of features to consider when performing classification and retrieval tasks.

On top of this, in the wider tourism context, the image datasets of Oxford buildings, Paris buildings, and holidays are widely used in pretrained CNN experiments with transfer learning. For instance, Mei *et al.* [89] investigate instance-level object retrieval via the deep region CNN method in which the VGG16 model and Oxford105k dataset are tested. Alzu'bi *et al.* [90] use compact deep convolutional features on the CBIR task involving VGG models and Oxford5k, Oxford105k and holidays datasets. Tzelepi and Tefas [91] exploit supervised learning of deep features in CBIR using the CaffeNet model with Oxford5k and Paris6k datasets. Sun *et al.* [92] explore similarity metrics of CNN features on image retrieval tasks via SiameseNet on Oxford5k, Paris6k, and holidays datasets. Gordo *et al.* [93] recommend an end-to-end learning approach on image retrieval tasks using Siamese, VGG16, ResNet101 models on Oxford5k, Oxford105k, Paris6k, Paris106k, and holidays datasets. Radenovic *et al.* [94] conduct unsupervised fine-tuning CNN image retrieval using VGG, AlexNet, ResNet models on Oxford5k, Oxford105k, Paris6k, Paris106k, holidays, and holidays101k datasets. Tzelepi and Tefas [59] offer three approaches to produce compact image descriptors of CNNs for CBIR through the CaffeNet model on the Paris6k dataset. It is worth noting that each CNN model has its own advantages and disadvantages. For example, the small-size CNN models require less storage and less computational time, whereas the bigger-size CNN models could contain more deep layers or parameters which could lead to higher accuracy.

To summarize, the utilization of CBIR and image classification demonstrates that content-based systems can be specifically applied or in synchronization for diverse areas of research.

### 2.2 Features Representing Image Content and Example Hostel Images

Prior to the retrieval or classification process, the extraction of features within an image is a crucial step as these features which represent the image would be mathematical measured.

Image features could be broadly categorized into two types of features, classical features and Artificial Intelligence-based features.

#### I. Classical features

Traditionally, image features can be represented in colours, textures, shapes, spatial positions, or in the form of Scale-Invariant Feature Transform (SIFT), for example, as described below.

##### *A. Colour features*

Colour has been one of the most active features used in content-based image classification and retrieval as it is a vital element when it comes to human's visual perception. There are four colour systems can be used which are **RGB** (red, green, and blue), **CMY** (cyan, magenta, and yellow) or **CMYK** (cyan, magenta, yellow, and black), **HSV** (hue, saturation, and value), and **Lab** (lightness, a, and b). Having said this, it seems RGB is the most widely mentioned and used colour system [59][38][61][95][96] even though it is not the most corresponding colour system to the human's colour perception [97].

Next, colour features can be represented in various ways such as **colour histograms** where the proportion of each colour in an image is illustrated [30][61][98][57][99][100][101][102][103][104], **colour moments** where the probability distribution of each colour in an image is demonstrated [105][100], and **colour block-based** where each colour in an image is divided into blocks, according to the given block size in pixels [98][105][106]. Based on the relevant literature retrieved, it is clearly shown that colour histogram is the most-cited method of colour representation.

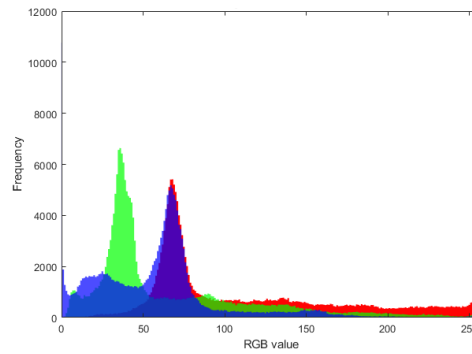


Figure 2.2: An example of colour histogram in the RGB colour system [107]

Furthermore, as shown in figure 2.3, the dominant colour in the image query is white along with the brown colour of the floor. Therefore, based on the colour features, the images with similar colours in similar proportions are expected to be retrieved.

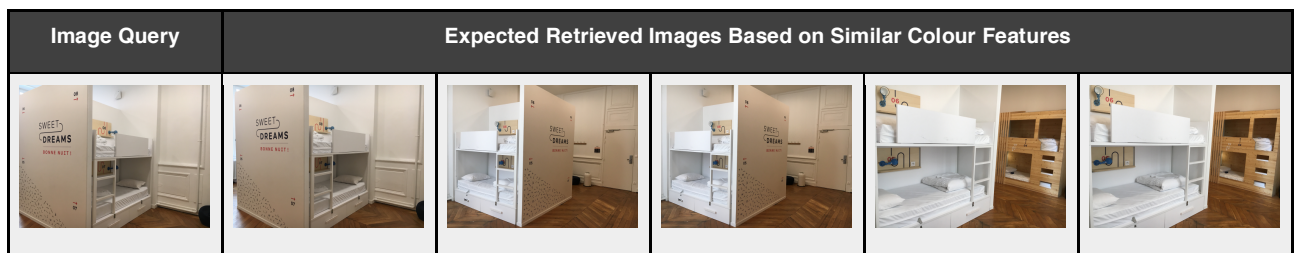


Figure 2.3: The example of hostel search after colour feature comparison

## B. Texture features

Similar to colour features, human's visual perception tends to consider the textures of objects in an image. In order to extract texture features, several texture analysis tools are available. For example, **co-occurrence matrix**, one of the earliest techniques proposed by Haralick *et al.* in 1973 [108], which show the frequency of pairs of the pixel with specific values located with respect to one another in a certain way in an image [38][95][98][96][104][109], **Tamura features**, proposed by Tamura *et al.* in 1978 [110], which group the brightness of pixels into six parameters in order to explain six texture properties, coarseness, contrast, directionality, line-likeness, regularity, and

roughness [98][104], and **wavelets** which analyze signal frequency in order to compute wavelet transform coefficients and are translated to the meaning of textures [35][96][97][98][104]. Although a selection of texture analysis techniques is introduced over the years, the co-occurrence matrix seems to remain one of the most popular techniques used for texture features extraction.

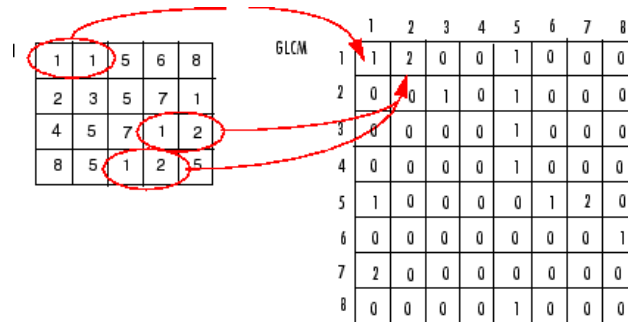


Figure 2.4: An example of co-occurrence matrices [111]

In the case of hostel images, objects in an image are likely to have different textures such as the floor texture, the patterns of wallpaper, and the bed sheet texture as seen in figure 2.5. These texture features of each hostel image would be compared in order to find hostel images with similar textures of objects.

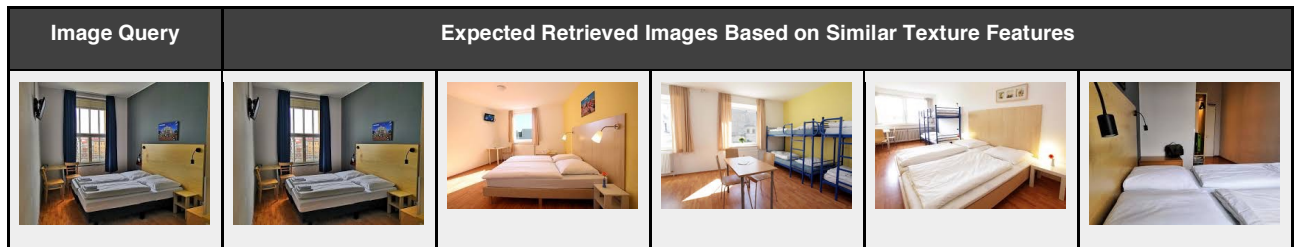


Figure 2.5: The example of hostel search after texture feature comparison

### C. Shape features

Two widely used features of shape are global features and local features. Global features such as circularity, aspect ratio, and moment invariants [112] are universal standards that can be used in order to represent the boundary of the shapes in an image. Following this, local features can be applied for the purpose of illustrating sets of consecutive boundary segments [113] in order to show the structure of objects in



an image. Therefore, particular objects in an image could be identified which contribute to a closer result of content-based image retrieval.

Looking at commonly used techniques for shape features extraction, it can be seen that three techniques receive more interest than the others. Firstly, the **Fourier transform** is a useful tool to decompose a complicated signal into simple waves which represent geometric structures in an image [97][98][104][114][115][116][117]. Secondly, **invariant moments** which illustrate the stabilization of image pixels' intensities in weighted average in order to describe objects in an image [96][97][104][118][119][120][121]. Lastly, **geometrical parameters** such as the centre of gravity, perimeter, eccentricity, rectangularity, and circularity ratio are used for the purpose of describing shapes [104][122]. Despite the fact that geometrical characteristics are the simplest technique, the Fourier transform seems to be the most popular one.

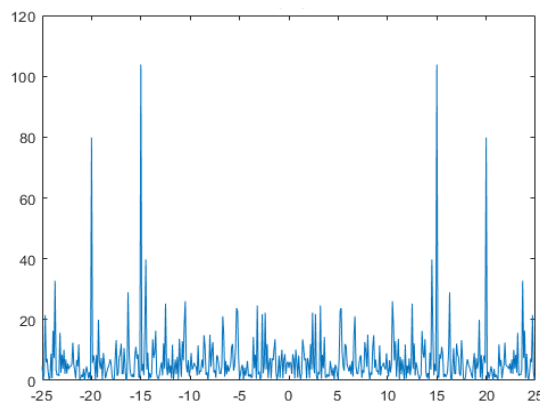


Figure 2.6: An example of a Fourier transform [123]

For hostel search, the shapes of objects in hostel images could be considered such as shapes of beds which could represent single beds, double beds, two-layer bunk beds, or capsule beds, for example, shapes of heaters, and shapes of toilet bowls. In figure 2.7, the image query represents rectangular shapes of the front entry to the capsule bed, as well as the single bed in the capsule. By comparing these shape features, the most similar hostel images could be retrieved.

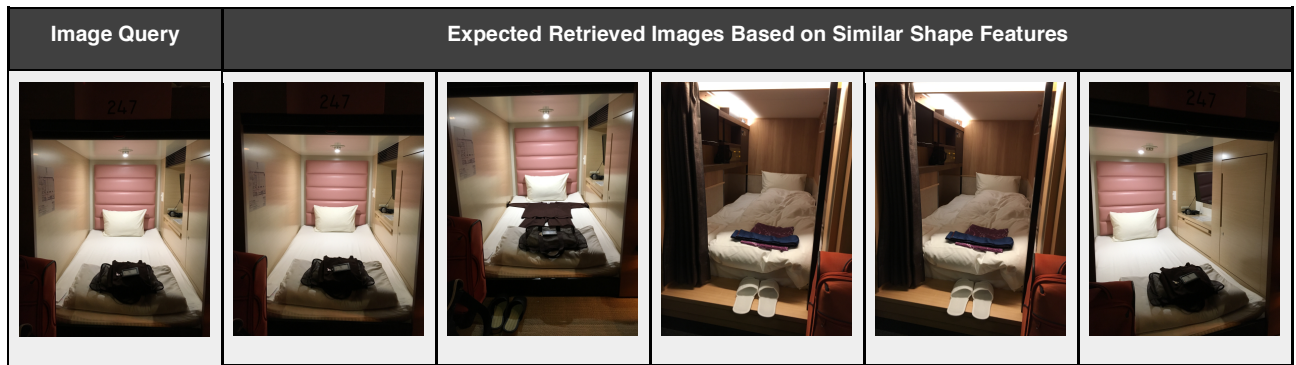


Figure 2.7: The example of hostel search after shape feature comparison

### D. Spatial features

The spatial position of an object within an image is one of the most basic and sensible features for image search as a geographical aspect can be seen in real-life examples. For instance, as shown in Figure 2.8, the window location of the hostel image query is in the middle and the two-layer bunk beds are on the left and the right. Therefore, the expected retrieved images should have a window and two-layer bunk beds in the same spatial positions as in the hostel image query. This useful feature helps the process of image search be more accurate. However, image rotation is a challenge for this type of feature.



Figure 2.8: The example of hostel search after spatial feature comparison

### E. Scale-Invariant Feature Transform (SIFT) features

The SIFT feature, which is also considered a classical or handcrafted feature, was introduced by Lowe in 2004 [64] has been studied in numerous works as it focuses on the key point(s) of interest in an image and this key point is invariant to scale, rotation,

location, and illumination, unlike the spatial features. There are four stages of computation used to generate SIFT features.

Stage one: Scale-space extrema extraction which is to identify potential interest points over all scales and image locations by using differences of Gaussian filters.

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (2.1)$$

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (2.2)$$

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \quad (2.3) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned}$$

where  $G$  is Gaussian kernel,  $L$  is scale space, and  $D$  is difference of Gaussian.

Stage two: Keypoint localization which is to identify the keypoints that are stable and resistant to image distortion by using a threshold value.

$$D(x, y, \sigma) = D(x_i, y_i, \sigma_i) + \left( \frac{\partial D(x, y, \sigma)}{\partial(x, y, \sigma)} \right)_{\substack{x-x_i \\ y-y_i \\ \sigma-\sigma_i}}^T \Delta + \frac{1}{2} \Delta^T \left( \frac{\partial^2 D(x, y, \sigma)}{\partial(x, y, \sigma)^2} \right)_{\substack{x-x_i \\ y-y_i \\ \sigma-\sigma_i}} \Delta \quad (2.4)$$

where  $\Delta = \begin{pmatrix} x-x_i \\ y-y_i \\ \sigma-\sigma_i \end{pmatrix}$  and using Taylor-series to express the difference of Gaussian function in the 3D neighbourhood around a keypoint. If the intensity at local extrema is more than a threshold value at 0.03, that potential point would be accepted as a key point.

Stage three: Orientation assignment which is to assign orientation(s) to each keypoint location based on local image gradient magnitudes and directions.

$$m(x, y) = \frac{L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)}{\sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}} \quad (2.5)$$

$$\theta(x, y) = \tan^{-1} \left( \frac{(L(x, y + 1) - L(x, y - 1))}{L(x + 1, y) - L(x - 1, y)} \right) \quad (2.6)$$

Then, an orientation histogram of all keypoints is created to identify each peak.

Stage four: Keypoint descriptor which is to measure the local image gradients at the selected scale (usually 2 x 2 cells and 4 x 4 in size for each cell) in the region around each keypoint and those above 80% peak in orientation histogram would represent the keypoint as a 128-dimensional vector containing information it describes.

Based on this key strength of the SIFT feature, it contributes to better accuracy of hostel search, especially when hostel images are photographed and posted on social media by young travellers who are likely to be non-professional photographers and the quality of hostel images taken is not in high resolution, as well as with variations in image angles.

## II. Artificial Intelligence-based features

Moving to today's society, it is undeniable that Artificial Intelligence (AI) has become one of the most discussed topics as several tasks used to be operated by human beings have now transferred to computing machines which are automatically run. Goodfellow, Bengio, and Courville [124] introduce a Venn diagram which simply defines AI terms including deep learning, a subset of representation learning, representation learning, a subset of machine learning, and machine learning, a subset of artificial intelligence. Considering these terms, CBIR and classification can be considered as deep learning as its algorithms are designed for software to train itself to recognize the image [125]. Having said that, the concept of CBIR has been discussed since 1992 by Kato [126] as his experiments involved automatic retrieval of images by matching features of image query with features of images in a database.

As AI is a powerful tool, Artificial Neural Network (ANN) has been explored in feature extraction in order to simulate the human cognition process. Firstly, the image attribute(s) will be manually defined by a human. Then, the designed system will be trained to recognize the labelled or unlabeled image(s) and this learning-based feature(s) would be used to compare to visual feature(s) of an unseen image(s) for the purpose of classifying the unseen image (s) or retrieving similar image(s). Among various types of Artificial Neural Network (ANN), Convolutional Neural Network (CNN) has been primarily applied on several image recognition and retrieval works including AlexNet, ZF Net, VGG Net, and GoogLeNet [59][127][128], for example. Nonetheless, the extraction of these Artificial Intelligence-based features remains a challenge as it is not easy to define all attributes or features for a hostel image or an image, due to the complexity of human neural network that the Artificial Neural Network should be able to simulate, and also the image retrieval and classification process are time and cost consuming with these artificial intelligence-based features.

### 2.3 Quantization of Image Features

As for quantization, it is the process of constraining an input from a continuous and/or large set of values such as the real numbers to an output in a smaller and countable set such as the integers. There are three steps in the quantization process, encoding an input, mapping the input value to the quantization index, and decoding the index to the quantized value. The process is illustrated in figure 2.9 below.

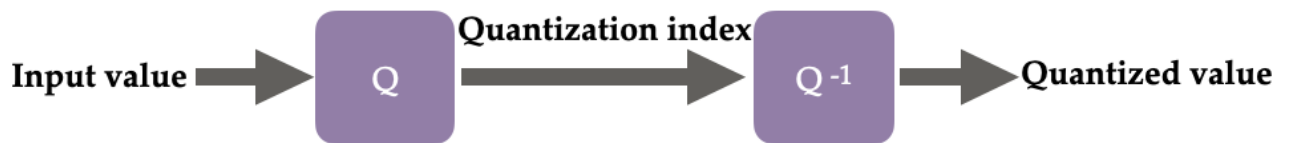


Figure 2.9: Quantization process

In more theoretical words, where  $I$  is a real random variable with the probability density function (pdf)  $p(I)$  and the representation of  $I$  is denoted as  $I$ , if we are given  $r$  bits to represent  $I$ , the value  $I$  can take on  $2^r$  values. The critical point of quantization is to find the optimum set of values for  $I$ , called the code points or partitions  $I_1, I_2, \dots$  and the regions  $S_1, S_2, \dots$ , that are associated with each code point/partition. Furthermore, based on the quantization theory [144], a  $2^r$ -rate distortion code consists of an encoding function,

$$a: R \rightarrow C, \quad (2.8)$$

where  $C$  is a subset with  $2^r$  elements of the set of all integers  $Z$  and a decoding function

$$b: C \rightarrow R. \quad (2.9)$$

The encoding function defines a partition  $\{S_1, \dots, S_{2^r}\}$  of  $R$  such that  $S \cap S = \emptyset$  for all  $i \neq j$  and  $\cup_{m=1}^{2^r} S_m = R$ .

The interval  $S_m$  is called the  $m$  –  $th$  quantization region and it is defined such that  $f(I)$  is constant for all  $I \in S_m$  and  $g(f(I)) = I_m$  for all  $I \in S_m$ .

There are two types of quantization mentioned in the literature, vector quantization and scalar quantization. Vector quantization [145][146][147][148][149] is a classical technique of data compression which divides a large set of points (vectors) into groups with approximately the same number of points closest to them and each group is represented by its centroid point. Since the modelling of probability density functions is allowed by the distribution of prototype vectors, the compressed data has errors that are inversely proportional to density. The transformation is usually done by projection or by using a codebook, a vector whose length is the same as the number of partition intervals, which guides the quantizer about the assigned value to inputs that falls into each range of the partition. Nonetheless, vector quantization might not be the most practical technique for image compression due to its computational complexity.

Next, scalar quantization [150][151][152][153] is a potential alternative as a scalar value is selected from a finite list of possible values to represent an input. It separately maps and rounds off each real input value into the nearest integer of output value, resulting smaller codebook required to be stored and significantly less computational complexity when compared to the vector quantization. Furthermore, scalar quantization can be generally categorized into two types, uniform quantization and non-uniform quantization [150]. The uniform quantization [150][151][152][154][155] is when the output levels are uniformly spaced with equal distance and the thresholds are midway between adjacent levels, whereas the non-uniform quantization [150][153][156][157] is when the intervals are not uniformly and equally spaced which provide lower average distortion but more complex design. Although images are likely to be quantized during image processing through this lossy compression technique as they are reduced the number of colours required to represent digital images which results in smaller file sizes such as JPEG and JPEG 200, it is crucial to further quantize image features and represent each pixel by a certain number of bit(s), in particular for high dimensional CNN features and/or

operations on low computable devices. As a result, it helps to preserve significant information while reducing less important information, provide a better feature representation of the image and make better discrimination, reduce datasize transmission, and become more computational efficient. Similar to dimensionality reduction, the quantization step is also executed after the high dimensional image features are extracted as demonstrated below.

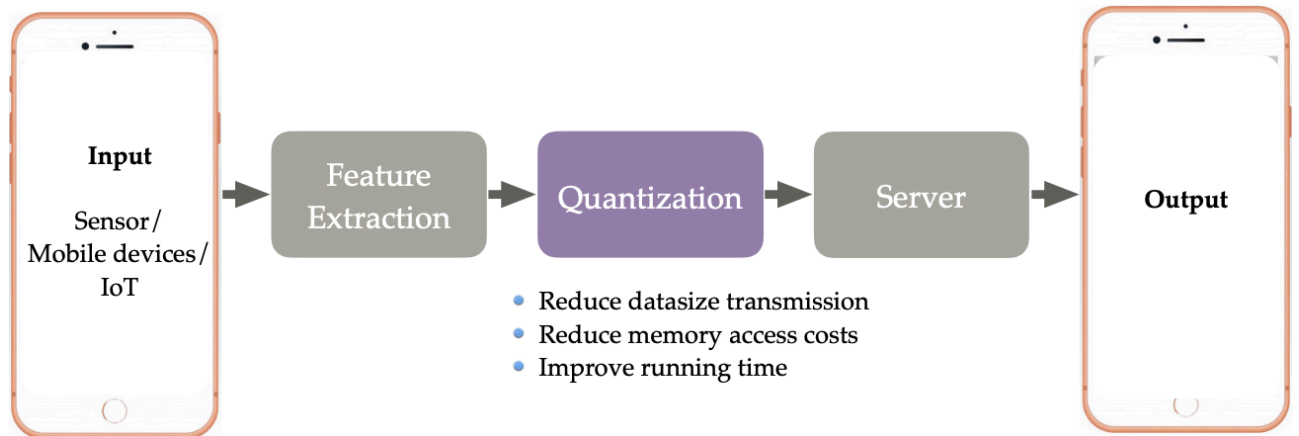


Figure 2.10: Quantization step in the content-based image classification process

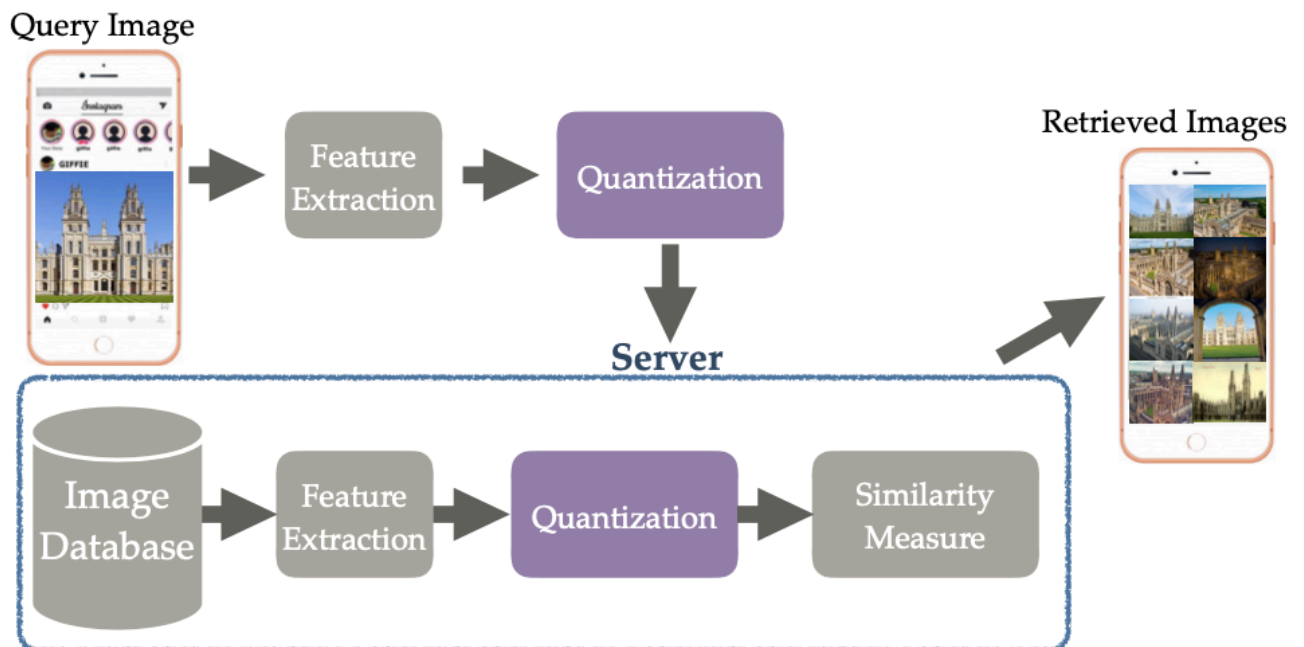


Figure 2.11: Quantization step in the content-based image retrieval process



2.4 Dimensionality Reduction of Image Features

Due to the development of multimedia technology and its heavy use in today’s society, multimedia data have been massively produced each day, especially digital images. As these data could be characterized by hundreds or thousands of features, the data transmission and storage could be the burdens of direct use of these high dimensional data which would result in inefficient performance, particularly for frequently used data. Therefore, a variety of dimensionality reduction techniques have been studied as attempts to reduce the computational complexity by compressing data with minimum loss of useful information. These low dimensional representations could be achieved by feature extraction or feature selection [129][130][131][132][133]. The key difference is that feature extraction generates synthetic attributes, whereas feature selection keeps some original features which bring benefits for classification in image processing tasks such as image recognition and retrieval if the selected features contain the most relevant information.

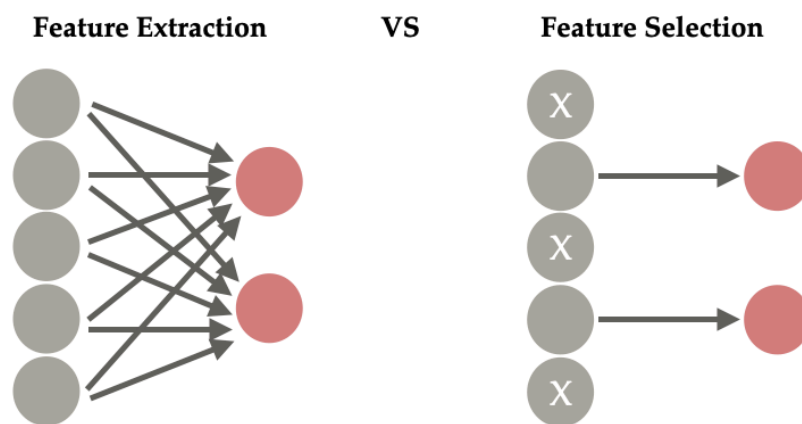


Figure 2.12: A comparison between feature extraction and feature selection

Feature selection methods could be grouped into three types, wrapper methods, filter methods, and embedded methods [131][134]. The wrapper methods consider one subset of features at a time to evaluate its relevance. Then, the next subset is iterated until the optimal subset is found. The subset selection could be implemented manually or automatically in three fashions, forward selection [134], backward selection [134],

and stepwise selection [134]. Nonetheless, these wrapper methods are time-consuming and lead to model overfitting when having a small-size dataset. Next is the filter methods. The selection of its optimal subset is to rank all features and choose the most relevant ones. Various ranking measures are found such as the Chi-square [131], Pearson correlation [129][131], analysis of variance (ANOVA) [129], and variance thresholding [129]. Even though the filter methods use less computational time than the wrapper methods, the correlations between features could be blind due to the drawback(s) of the chosen filter. Lastly, embedded methods such as Lasso regression [129][131], Ridge regression [129], and Decision tree [129][131] are often seen in literature as these methods select relevant features while tuning the model.

To avoid selection errors when using feature subsets, feature extraction is another option to consider for dimensionality reduction. Through subspace learning methods, the high dimensional data could be mapped to a low dimensional subspace via a linear or non-linear projection [135][136].

$$X_{p \times n} = A_{p \times m} X_{m \times n} \quad (2.7)$$

where  $X_{m \times n}$  is the samples in the original  $m$  dimensional feature space,  $X_{p \times n}$  is the samples in a lower  $p$  dimensional feature space, and  $A_{p \times m}$  is a data-independent projection matrix.

Some of the most cited linear projections include Principle Components Analysis (PCA) [130][135][136][137] which considers the most important input features out of all features embedded from the original data. This unsupervised learning technique linearly captures the original data distribution while maximizing variances and minimizing the reconstruction error. Unlike PCA, the Independent Component Analysis (ICA) [130][138] considers only independent input features by finding two represented vectors whose linear and non-linear dependence are equal to zero. Though this technique could separate independent sources from a mixed signal, it does not consider variances of data points nor have Gaussian distribution. Linear Discriminant Analysis (LDA) [130][135][136][139] focuses on projections which

maximize the distance between the mean of each data class, while minimizing the distance within its own class. This supervised learning dimensionality reduction technique tends to well-perform on the classification tasks. However, the poor results can be found for non-Gaussian input data and on the small-scale datasets with fewer classes. In addition, these types of linear projections could achieve high performance when different features have linear relationships. Otherwise, the non-linear approach could be considered. For example, Locally Linear Embedding (LLE) [130][140] which is an unsupervised and Manifold learning-based projection that makes an object of lower dimensions, Manifold, representable in its original dimensions by discovering non-linear structure from the local symmetries of linear reconstructions. The t-distributed Stochastic Neighbour Embedding (t-SNE) [130][141] is popular for the visualization of high dimensional data. This unsupervised learning technique minimizes the Kullback-Leibler divergence between the joint probabilities of the input features in high-dimensional space and the lower-dimensional space. Nevertheless, due to the Student-t distribution in lower space modelling, t-SNE might not be suitable for  $> 3$  dimensions as these lower representations might not preserve the local structure of the input data. Autoencoders [130][142][143] is a non-linear transformation that compresses the input data into the latent space in order to remove irrelevant information. Then, reproducing the lower-dimensional data from the encoded latent space. Though this technique could reduce dimensions, the reconstructed data could be of poor quality, especially when the data complexity is high such as digital image data and video data.

Furthermore, for the feature extraction approach, the dimensionality reduction step is typically executed after the high dimensional features are extracted for both image classification and retrieval tasks as illustrated in Figure 2.13 and Figure 2.14 below. Then, these lower-dimensional features are used in the following stage(s) in order to fasten computational time, reduce transmitted datasize, and require smaller storage space. Dimensionality reduction does not only benefit the large-scale image processing tasks but also those much smaller tasks on mobile devices or the Internet of Things (IoT) which have become today's day-to-day operations. On top of this, in the light of increasing AI-based feature use which creates millions of features and

needs a significant amount of data for training computations, it is even more crucial to reduce the size of these deep features and generate compact descriptors. Nevertheless, the curse of dimensionality remains a challenge for performance accuracy and time proficiency.

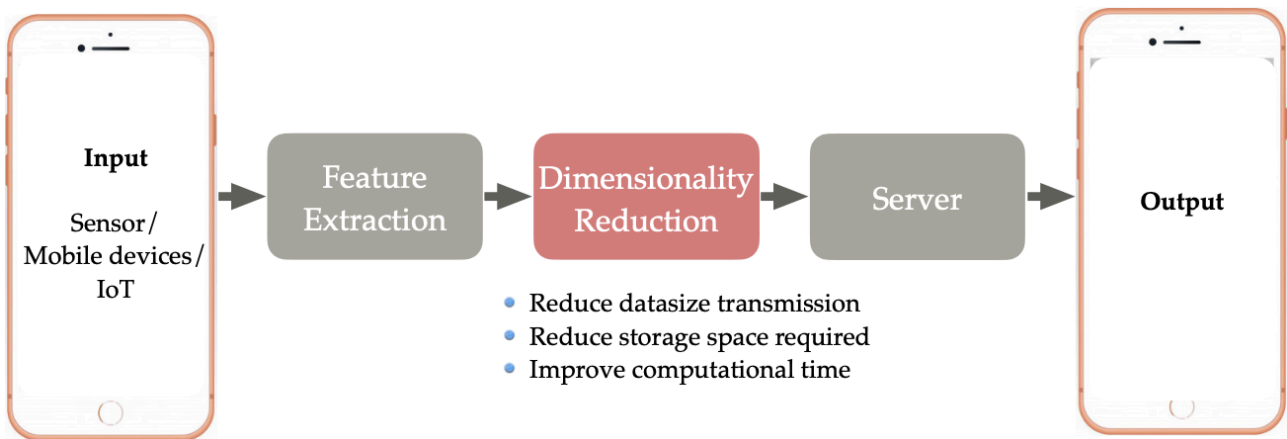


Figure 2.13: Dimensionality reduction step in the content-based image classification process

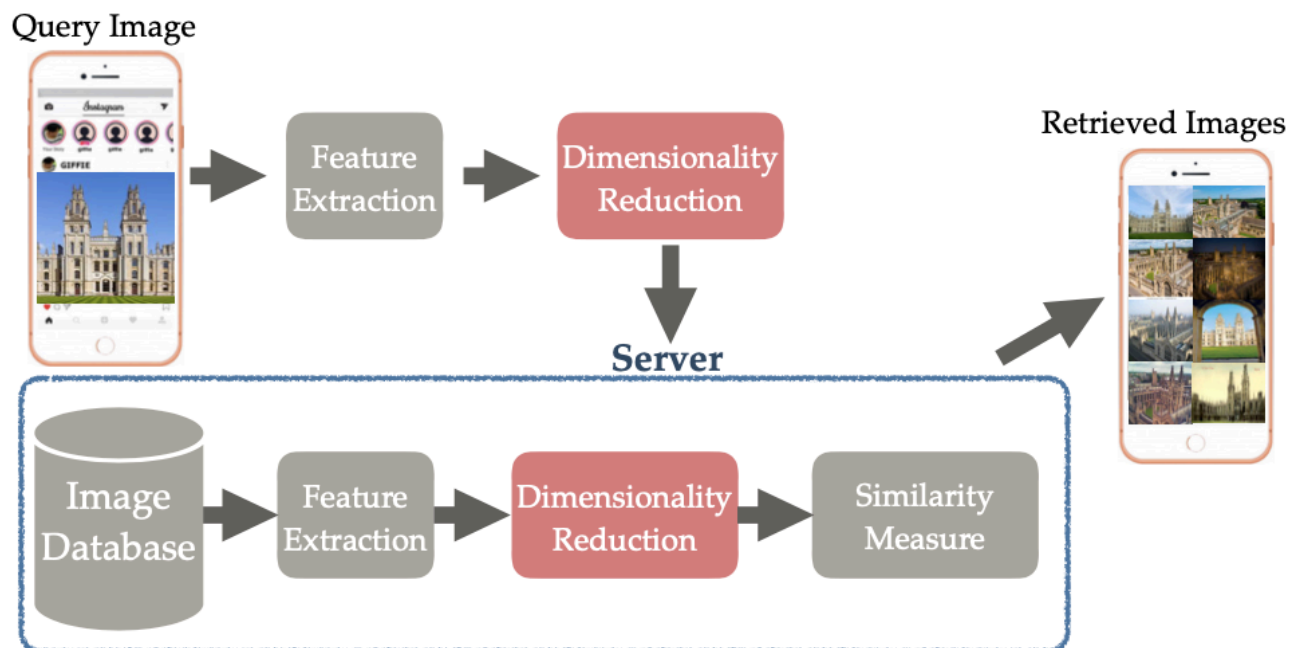


Figure 2.14: Dimensionality reduction step in the content-based image retrieval process

## 2.5 Similarity Measures and Performance Measurement

The similarity measure is vital for machine learning tasks such as content-based image classification and retrieval as it reflects the closeness between two considered images. The high similarity is found when two images are from the same class, whereas low similarity or no similarity means the two images are from different classes. Similarity can be evaluated by calculating distance of two points, origin and coordinate. A variety of distance measure can be seen in literature such as Manhattan distance [143][145][158][159], Euclidean distance [145][158][159][160][161][162], Minkowski-Form distance [163][164][165], Chi-Square distance [74][158][163], and Hamming distance [160][165][166][167][168]. Nevertheless, the types of data determine whether a distance measure is suitable for the expected resulting scores. Consequently, Euclidean distance and Hamming distance are frequently implemented in this study.

**Euclidean distance** which is also known as L2 normalization or L2 norm is represented as the shortest distance between two points in a Euclidean space,  $\mathcal{N}$ dimensional space, in a straight line.

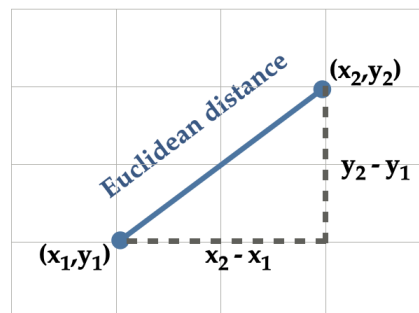


Figure 2.15: Euclidean distance

The calculation is for two rows of data with numerical values such as a floating-point or integer values. If columns have values with differing scales, it is essential to normalize the numerical values across all columns prior to calculating the Euclidean distance. Otherwise, columns with large values will dominate the distance measure. The formula is shown below.

$$ED(V_1, V_2) = \sqrt{\sum_{i=1}^N (V_{1i} - V_{2i})^2} \quad (2.10)$$

where  $V_1$  and  $V_2$  are two real-valued vectors which help to define an optimization objective that is not the arbitrary one. This distance is always a positive number. In another word, the direction of the vector is ignored or zero.

Due to its simplicity, the Euclidean distance is one of the most commonly used measures. Furthermore, it is known to provide a single, stable and analytical solution, while allowing normalized and weighted features. Nonetheless, it has less resistance to outliers and high dimensional data might lower its performance.

In terms of **Hamming distance**, it is a distance between two binary strings, also called bit strings, of the same vector length. In another word, the number of different positions between two same-length strings. For a one-hot encoded string, the calculation is simple by doing a sum of the bit differences between the strings which is a 0 or 1.

**No. of bit difference**

<b>A</b>	0	1	0	0	1	0	1	0
<b>B</b>	0	0	1	0	1	0	0	1
		1	2				3	4

Figure 2.16: Hamming distance = 4 (No. of bit difference)

$$D_H = \sum_{i=1}^k |X_i - Y_i| \quad (2.11)$$

As for bit strings which have many 1 bits, the average number of bit differences will be considered for a Hamming distance with the following formula.

$$D_H = \frac{\sum_{i=1}^k |X_i - Y_i|}{k} \quad (2.12)$$

The key advantage of the Hamming distance is its effectiveness in binary space or on networks in which the data streams are given for the single-bit errors. Moreover, it is easy and quick to compute the distance. Having said that, the application is difficult for non single-bit problems.

Moving to the performance measurement, the evaluation of a CBIR or classification system is crucial as it is the tool to measure the image retrieval or classification performance in terms of its successfulness in practical application. Generally, there are three criteria to consider its performance which are accuracy, computational efficiency, and memory cost.

Accuracy is a measure used to quantitatively check the correctness of the relevant retrieved results or classified images against data in the database.

Computational efficiency is a measure used to check the time cost in the process of visual vocabulary, feature indexing, and image querying. The visual vocabulary and the feature indexing processes are operated offline, whereas the image querying process is conducted online which is expected to perform in real-time.

Memory cost is a measure used to check the memory usage during the online stage including loading the quantizer and the index file of the database.

It could be difficult to justify all criteria due to a tradeoff hindrance, however, the attempt to achieve the best possible results would be ideal. A variety of performance measurement are available such as Precision [133][169][170], Recall [133][160][166][167], F-Measure [133][171][172], Precision-Recall curve [159][169][172], and Mean Average Precision [145][159][160][161][169][173][174]. Nevertheless, one of the most widely used measures in CBIR and classification is the **mean Average Precision (mAP)**, particularly when dealing with high dimensional data with many classes or categories. The mean Average Precision can be calculated from Precision with the following formula [175].

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (2.13)$$

where  $Precision = \frac{True\ Positive}{Total\ Positive\ results}$  and  $AP_i$  is Average Precision.

Another commonly seen measure is **Precision@k**. It is used in binary problems for retrieval or recommendation systems when considering relevant and irrelevant items. Firstly,  $k$  is chosen as a number of recommended or retrieved items. Then, the Precision@k which is the proportion of relevant item(s) in the retrieved item(s) is calculated as shown in the formula below.

$$Precision@k = \frac{\text{number of retrieved items@k that are relevant}}{\text{number of retrieved items@k}} \quad (2.14)$$



### 2.6 Summary

A systematic and comprehensive process of literature review has been conducted in order to identify relevant literature. Consequently, due to being interdisciplinary research of the study, the existing approaches of CBIR and classification application in computer science and engineering discipline and tourism discipline are highlighted. Firstly, the computer science and engineering discipline. In the light of the discipline's nature which is to focus on understanding, designing, and developing programmes in relation to computer-oriented subjects, an advancing of the existing feature extraction techniques and the existing image retrieval techniques in order to create more efficient tools rather than apply the techniques to particular fields of study or particular industries are mainly found. Furthermore, the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) has given researchers the opportunity to develop algorithms for object detection and image classification using their enormous 14 million image dataset. More importantly, several novel CNN architectures are developed for the competition, and outside the competition, and many of these CNN models still gain popularity these days. Moving to the tourism discipline, the main CBIR approaches are to improve image representation and retrieval by advancing existing feature extraction techniques, contributing novel techniques in the feature extraction process through fine-tuning fusion features, and improving image query of the CBIR system. However, despite there are five basic elements of tourism product which are attractions, accommodation, access, amenities, and activities, the tourist attraction element has been in the researchers' focus, particularly in art and cultural heritage domains. The accommodation sector is understudied and, to the best of my knowledge, there is no CBIR or classification research undertaken in relation to the hostel domain. Having said that, some studies in hotel recognition and retrieval have been found.

In terms of image features, two categories are identified as features representing image content which are classical features and Artificial Intelligence-based features. The classical features include colour, shape, texture, spatial position, and SIFT, for example. Artificial Intelligence-based features, particularly for CNN, have become

powerful tools for feature extraction on several image recognition and retrieval works in recent years. Furthermore, due to the high dimensional image features, especially deep CNN features, dimensionality reduction and quantization are essential in order to improve computational time and use less memory space, as well as reduce data size transmission for the applications on mobile devices or IoT which have a number of constraints. Even though extensive studies in dimensionality reduction and quantization techniques have been found, there is room for improvement in both areas.

Lastly, the similarity measures and the performance measurement are fundamental. Despite a variety of methods are available for similarity measurement and performance evaluation, the appropriate tools should be applied to tailor each experiment. Therefore, the results of each study would be reliable and make meaningful/significant contributions to the research community.

## Chapter 3: Quantization Effect on General Image Retrieval and Classification

It is undeniable that the benefits of quantization are significant for image classification and retrieval, especially on low computational devices, as it helps to reduce the influence of noise (quantization error) and obtain a better quality of the signal, as well as reduce computational cost. Therefore, scalar quantization is implemented in this research as it is the most simple and applicable compression technique and helps to achieve the aim which is to develop fast computable and memory efficiency operators on devices with low-bandwidth, low-powered, and low-buffered restrictions. Further details on each quantization technique and performance comparison are demonstrated in this chapter.

### 3.1 Uniform Scalar Quantization

The uniform scalar quantization is widely used in compressive data works [150][151][152] which aim to reduce data size transmission and storage. This uniform quantizer could be applied when input is either uniformly or non-uniformly distributed. Nevertheless, it is essential to additionally use an entropy coder, the lossless data compression method which represents frequently occurring patterns with few bits and rarely occurring patterns with many bits, or minimize the distortion by first writing a distortion as a function of the step size in each frequency region and then minimize the function for non-uniformly distributed input. Furthermore, as the uniform quantizer contributes the output levels with equal distance, the adjacent levels could be an infinite number or semi-infinite at the outermost cells when the number of levels is less than infinity.

Most uniform quantizers for signed input value can be categorized into mid-rise (zero is not one of the output levels and the number of levels is even) or mid-tread (zero is

## CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

one of the output levels and the number of levels is odd) [150]. A typical mid-rise uniform scalar quantizer with a quantization step size  $q > 0$  can be addressed as

$$Q_q(x) = q \cdot \left( \left\lfloor \frac{x}{q} \right\rfloor + \frac{1}{2} \right) \quad (3.1)$$

where the notation  $\lfloor x \rfloor$  corresponds to the greatest integer less than or equal to  $x$ . For simplicity, it is assumed that  $C = Z$  is not finite. The encoding function  $f(x)$  is

$$f(x) = \left\lfloor \frac{x}{q} \right\rfloor \quad (3.2)$$

and the decoding function is

$$g(k) = q \cdot \left( k + \frac{1}{2} \right), \forall k \in C \quad (3.3)$$

As for the mid-tread uniform scalar quantizer with deadzone  $\lambda > 0$ , it is defined as

$$Q_{q,\lambda}(x) = \text{sign}(x) \max \left( 0, \left\lfloor \frac{|x| - \frac{\lambda}{2}}{q} + 1 \right\rfloor \right) \times q \quad (3.4)$$

where  $\text{sign}(I)$  denotes the sign of  $I$ :  $\text{sign}(I) = 1$  if  $I > 0$ ,  $\text{sign}(I) = -1$  if  $I < 0$  and  $\text{sign}(0) = 0$ . The zero output of the quantizer is the interval  $\left[ -\frac{\lambda}{2}, \frac{\lambda}{2} \right]$  called the deadzone. The standard mid-tread quantizer corresponds to  $\lambda = q$ . The encoding function is

$$f(x) = \text{sign}(x) \cdot \max \left( 0, \left\lfloor \frac{|x| - \frac{\lambda}{2}}{q} + 1 \right\rfloor \right) \quad (3.5)$$

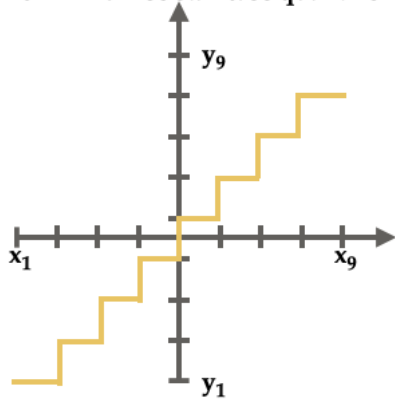
and the decoding function is

$$g(k) = \text{sign}(k) \cdot \left( \frac{\lambda}{2} + q \cdot \left( |k| - \frac{1}{2} \right) \right), \forall k \in C. \quad (3.6)$$

## CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

As a result, the output levels are uniformly spaced with equal distance and the thresholds are midway between adjacent levels represented by a finite list of possible values.

Uniform mid-rise staircase quantizer



Uniform mid-tread staircase quantizer

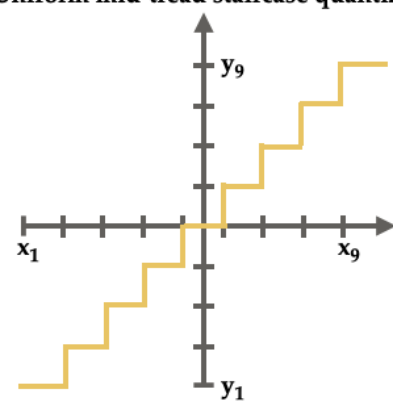


Figure 3.1: Uniform mid-rise staircase quantizer      Figure 3.2: Uniform mid-tread staircase quantizer

Furthermore, in the light of bit allocation as side information for the decoder, the re-scaling of transmitted quantized spectral values to the original values could be possible. Despite the quantization is a nonlinear map, the dithering operation could be used to smoothen. Dither is high-frequency noise intentionally added to remove the low-frequency limit cycles, while substantially reducing the system response to its sufficiently high frequency. Consequently, the dither can reduce the amplitude of limit cycle oscillation dramatically.

Let  $Q: R \rightarrow R$  be a uniform scalar quantizer with step size  $\Delta$  (giving  $Q(x) = i\Delta$  for all  $x \in (i\Delta - \frac{\Delta}{2}, i\Delta + \frac{\Delta}{2})$ ) and  $Z$  be a dither random variable uniformly distributed on  $(-\frac{\Delta}{2}, \frac{\Delta}{2})$  [176][177][178].

Theoretically, a dither random variable  $Z_1$  is a real random variable which is uniformly distributed on  $(-\frac{\Delta_1}{2}, \frac{\Delta_1}{2})$  and known to both the encoder and decoder. Therefore, a pseudo-random value can be generated at the encoder and explicitly described at the decoder. Additionally, the cost of this value description can be small when it is amortized, while being able to stabilize the impact of the uniform scalar quantization.

# CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

## 3.2 Non-Uniform Scalar Quantization

Unlike uniform quantization, non-uniform scalar quantization [150][153][156][157] produces the output levels with different sizes of space between each adjacent level. The finer regions indicate greater association to more likely values which are used for ranges of the first parameter and help to lower the average distortion. Their thresholds and intervals can be determined by using various types of distribution such as the Laplacian distribution of wavelet coefficients [156], the Lloyd Max approach [179], and the Wyner-Ziv principle [180]. By applying these non-uniform step-size schemes, it is possible to reduce bit consumption, while preserving perceptible quality. Nevertheless, the complex design of this quantizer remains challenging.

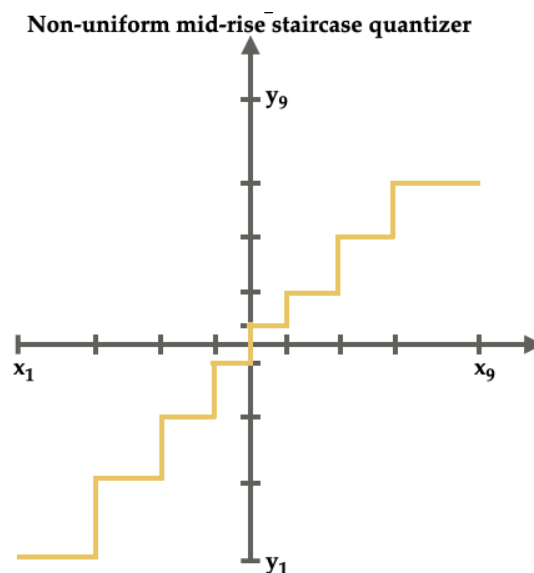


Figure 3.3: Non-uniform mid-rise staircase quantizer

# CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

## 3.3 Performance Comparison

### Stage 1: Matched SIFT features of hostel bedroom images after quantization

- *SIFT feature extraction*

The SIFT feature has been successfully used in many computer vision and image processing tasks, especially object recognition, due to its description of local gradient distribution which is invariant to scale, rotation, location, and illumination. Consequently, the extracted features of an object in a training image can be used to identify and locate the object in a test image which could contain many other objects and eventually help CBIR and classification systems perform effectively and accurately. To generate SIFT features, the fundamental process aforementioned is implemented, Scale-space extrema extraction, Keypoint localization, Orientation assignment, and Keypoint descriptor selection. Consequently, the SIFT features of two images are compared and matched to demonstrate their similarity as illustrated in figure 3.4 below.

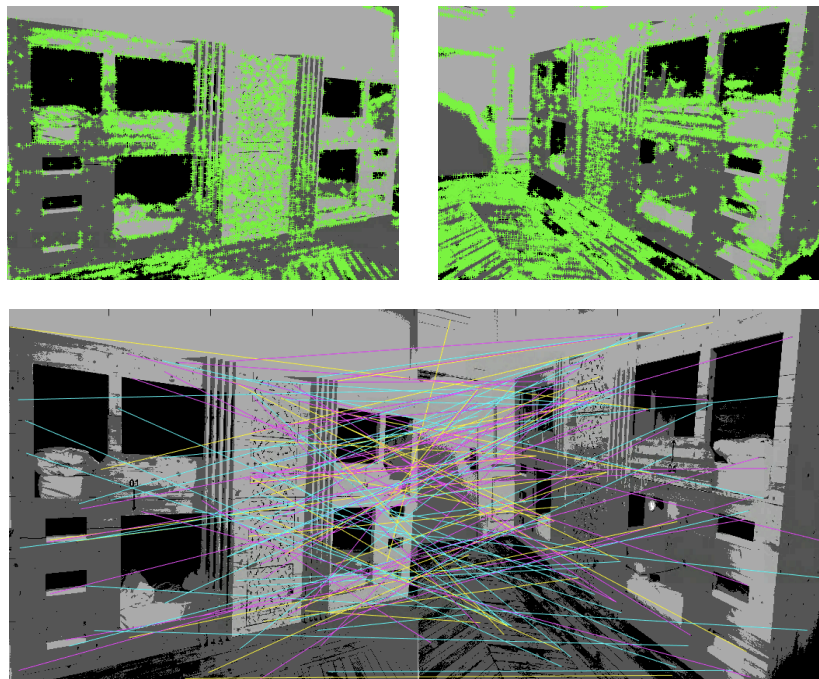


Figure 3.4: SIFT feature extraction and matching result

## CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

### • *Quantization of SIFT features*

The fact that SIFT features are high dimensional and the number of SIFT features for one image varies from tens to thousands, matching the large size of SIFT keypoints between two images could result in high memory usage and time complexity. Additionally, despite each JPEG image could already be quantized during the process of JPEG compression, further quantization is crucial for hostel search on smartphones as it helps to preserve significant information while reducing less important information and provides a better feature representation of the image, as well as makes better discrimination. Therefore, in this study, a dithering-based scalar uniform quantizer is applied as it partitions the whole space of digitized image signal in a uniform manner and represents all values in each subspace by a single value which could contribute to faster transmission in CBIR and classification systems [181].

### • *Preliminary experiment and results*

The simulations of this preliminary experiment are implemented on Intel(R) Core (TM) i7-6700 CPU @3.40GHz machine with 16GB RAM and programmed in Matlab language (version R2020a). Six classes of the dataset are explored, single bed class, two-layer bunk bed class, three-layer bunk bed class, capsule bed class, double bed class, and mixed beds class. 513 hostel bedroom images are tested, 387 independent hostel images and 126 chain hostel images. Each image size is between 389 KB to 707 KB and all images are converted from the RGB colour system to greyscale for the purpose of information minimization of each hostel image's pixel.



# CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

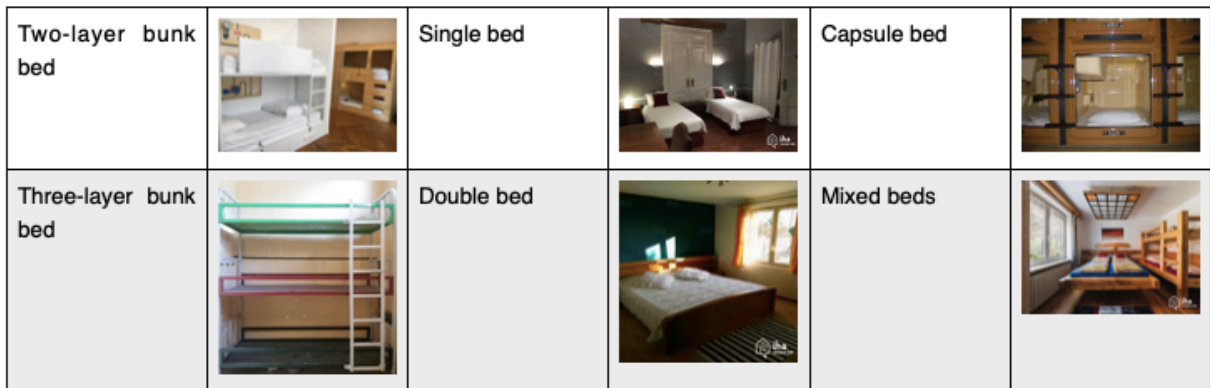


Figure 3.5: Hostel image examples of each class

Next, the quantization process. the dithering-based scalar uniform quantizer takes each subspace of digitized image signal and produces an integer quantization index which is represented by a number of bit(s) per image pixel. Seven numbers of bits are tested on 128-dimensional vector, 16-bit quantization, 12-bit quantization, 10-bit quantization, 8-bit quantization, 5-bit quantization, 4-bit quantization, and 3-bit quantization, and the SIFT feature matching result is demonstrated in figure 3.6, as well as its elapsed time in figure 3.7 below.

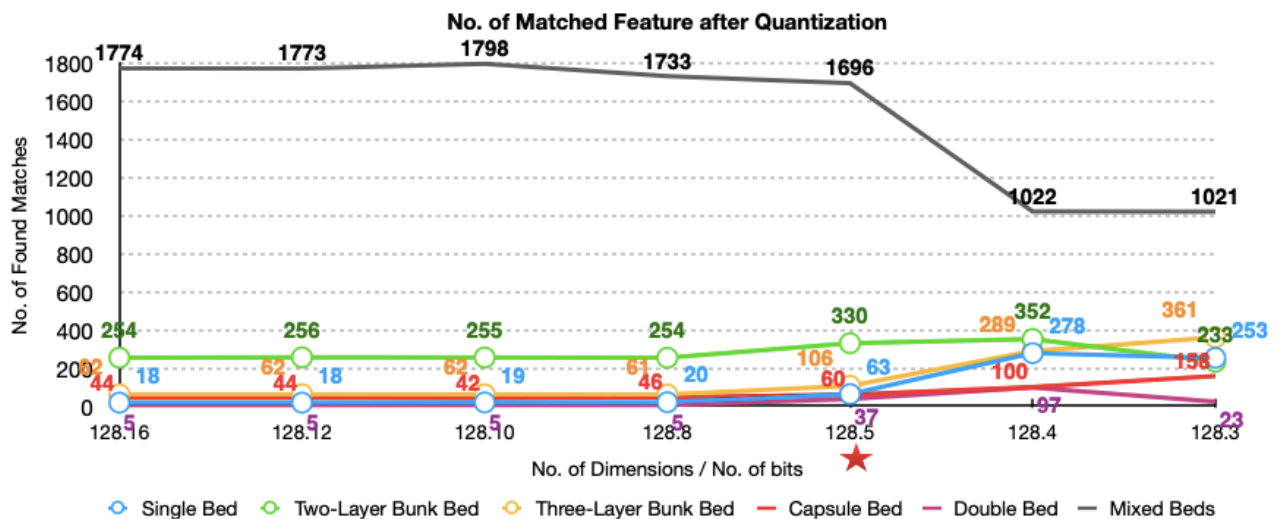


Figure 3.6: No. of matched SIFT features after image quantization on 128-dimensional vector

# CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

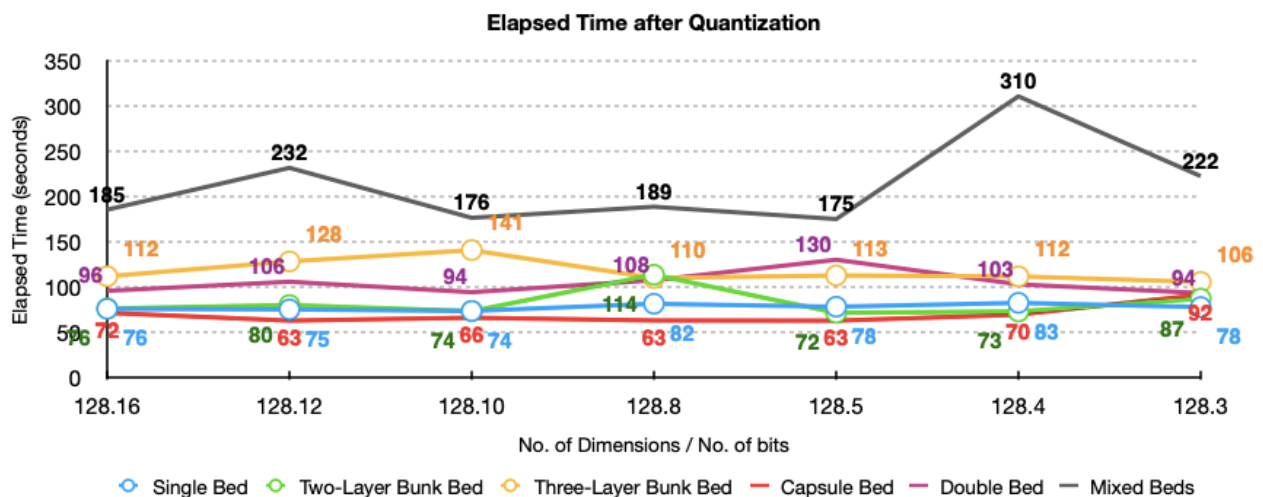


Figure 3.7: The elapsed time results of image quantization on 128-dimensional vector

After considering different levels of bits for 128-dimensional vector, figure 3.6 shows good performances in SIFT feature matching for most classes of hostel image. In spite of the number of matched features is slightly decreased at a few quantization levels, the upward trend continues for low-bit quantizations, at 5-bit quantization onwards. As for the elapsed time, figure 3.7 shows that the simulations in Matlab only consume short processing time, a 1 - 5 minute range for each simulation at different quantization levels.

## • Conclusion

Based on the performances of this preliminary experiment, the direct positive effect of image quantization shows in the SIFT feature matching results as the trend reflects a progressive performance even at low-bit quantization, especially at 5-bit quantization. Furthermore, the short elapsed time in Matlab simulations highlights the promising CBIR and classification outcomes on low-computational devices after the implementation of the proposed quantization approach.

# CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

## Stage 2: Multi-bit quantization effect on embedded CNN features

- *Pretrained CNN features*

Though several CNN models are available and already pretrained on the large-scale ImageNet dataset, two big different-in-size models are chosen, SqueezeNet and VGG19, in order to investigate the quantization effect on their deep features. The information about the selected CNN models is shown below.

CNN model	Model size	No. of deep layer	No. of parameter
SqueezeNet [237]	5.2MB	18	1.2 million
VGG19 [47]	535MB	19	144 million

Table III.1: SqueezeNet and VGG19 models information

- *Embedded CNN features with quantization*

As this project's aim is toward the application on devices with low computational resources, prior to further quantization, the chosen off-the-shelf CNNs are reduced their high dimensional features using four operators, Random Gaussian, Circular Binary Embedding (CBE), RD-rand with Discrete Cosine Transforms (DCT), and RD-Golay with Discrete Cosine Transforms (DCT). Random Gaussian and CBE are the state-of-the-art dimensionality reduction operators, whereas RD-rand and RD-Golay are newly developed  $M \times N$  operators ( $A$ ) during this project,  $M$  is number of bits in reduced dimension and  $N$  is number of dimensions for the CNN feature vectors.

- RD-rand: 
$$A = \frac{1}{\sqrt{M}} R_{\Omega} D_0 \quad (3.7)$$

*A random DCT matrix (RD – rand) is given by  $D_0$   
 $= DCT D_{\zeta}$ , where DCT is the DCT matrix and  $D_{\zeta}$  is a diagonal matrix with random sequence  $\zeta$  on its diagonal.*

## CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

- RD-Golay: 
$$A = \frac{1}{\sqrt{NM}} R_I D_1 D_0 \quad (3.8)$$

A Golay DCT matrix (RD – Golay) is given by  $D = DCT D_\zeta$ , where  $DCT$  is the DCT matrix and  $D_\zeta$  is a diagonal matrix with Golay sequence  $\zeta$  on its diagonal

where  $D_i (i = 0,1)$  are RDs,  $R_\Omega$  and  $R_I$  represent the sub-sampling operator. In particular,  $\Omega$  in (3.7) is a uniform random subset of  $\{1, 2, \dots, N\}$  with size of  $|\Omega| = M$ ; and in (3.8),  $I$  is a fixed set with  $I = \{1, 2, \dots, M\}$ .

Operators	No. of Random Coefficients		No. of Operations	
	Floats	Binaries	Multiplications	Additions
Gaussian	$\mathcal{O}(MN)$	0	$\mathcal{O}(MN)$	$\mathcal{O}(MN)$
CBE [167]	$N$	$N$	$\mathcal{O}(N \log N)$	$\mathcal{O}(N \log N)$
RD-rand (proposed)	0	$N$	0	$\mathcal{O}(M \log N)$
RD-Golay (proposed)	0	0	0	$\mathcal{O}(N \log N)$

Table III.II: Comparison of the number of random coefficients and number of operations between different operators

Next, a uniform scalar quantizer is applied to explore its effect in various bits.

# CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

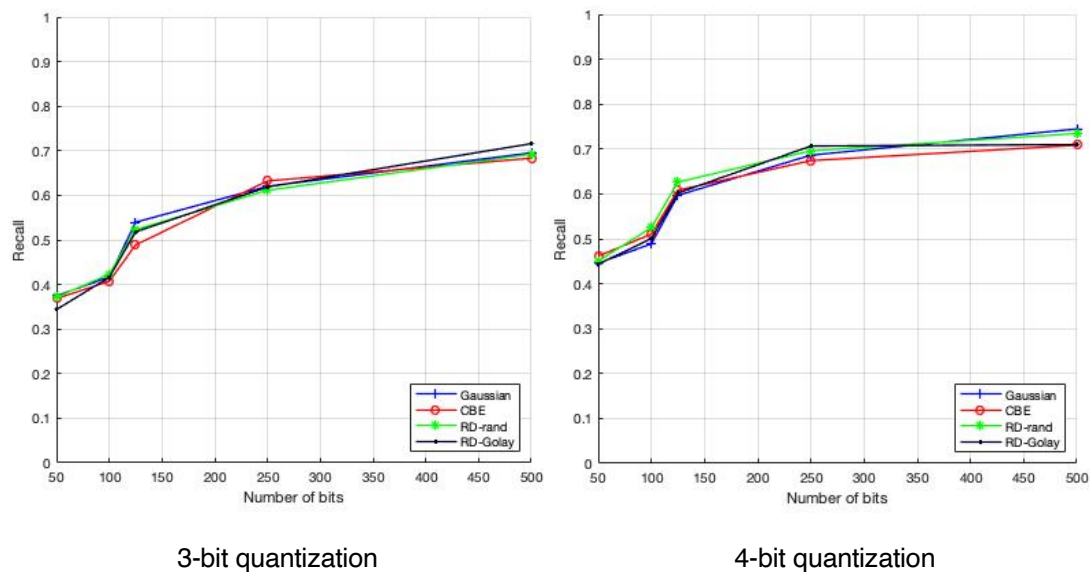
## • Experiments and results

This image classification experiment is conducted in the Matlab environment on Intel(R) Core (TM) i7-6700 CPU @3.40GHz machine with 16GB RAM. The SqueezeNet model and VGG19 model are tested on the Caltech 101 dataset containing 9,146 images of 101 object categories. The dataset is split into 70:30, 70 per cent is for model training and 30 per cent is for query testing. Additionally, 5 iterations are completed for each simulation. Recall is chosen as a performance measure since relevant items are the focus of image classification tasks, whereas Precision is a more appropriate tool for image retrieval tasks where both relevant and irrelevant items are considered in a search result. To be precise, recall is a percentage of the relevant items classified in relation to the total relevant items in the database.

$$Recall = \frac{TP}{TP + FN} \quad (3.9)$$

where  $TP$  is true positives and  $FN$  is false negatives

Figure 3.8 illustrates that for a small-scale model like SqueezeNet, a 50% recall rate can be achieved at only 100 bits per descriptor with 4-bit quantization for all four operators in a similar manner. Similarly, 5-bit quantization can reach over 60% recall at 100 bits per descriptor. Moreover, with 6-bit to 8-bit quantization, the recall rate improves slightly at 70%-80%.



# CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

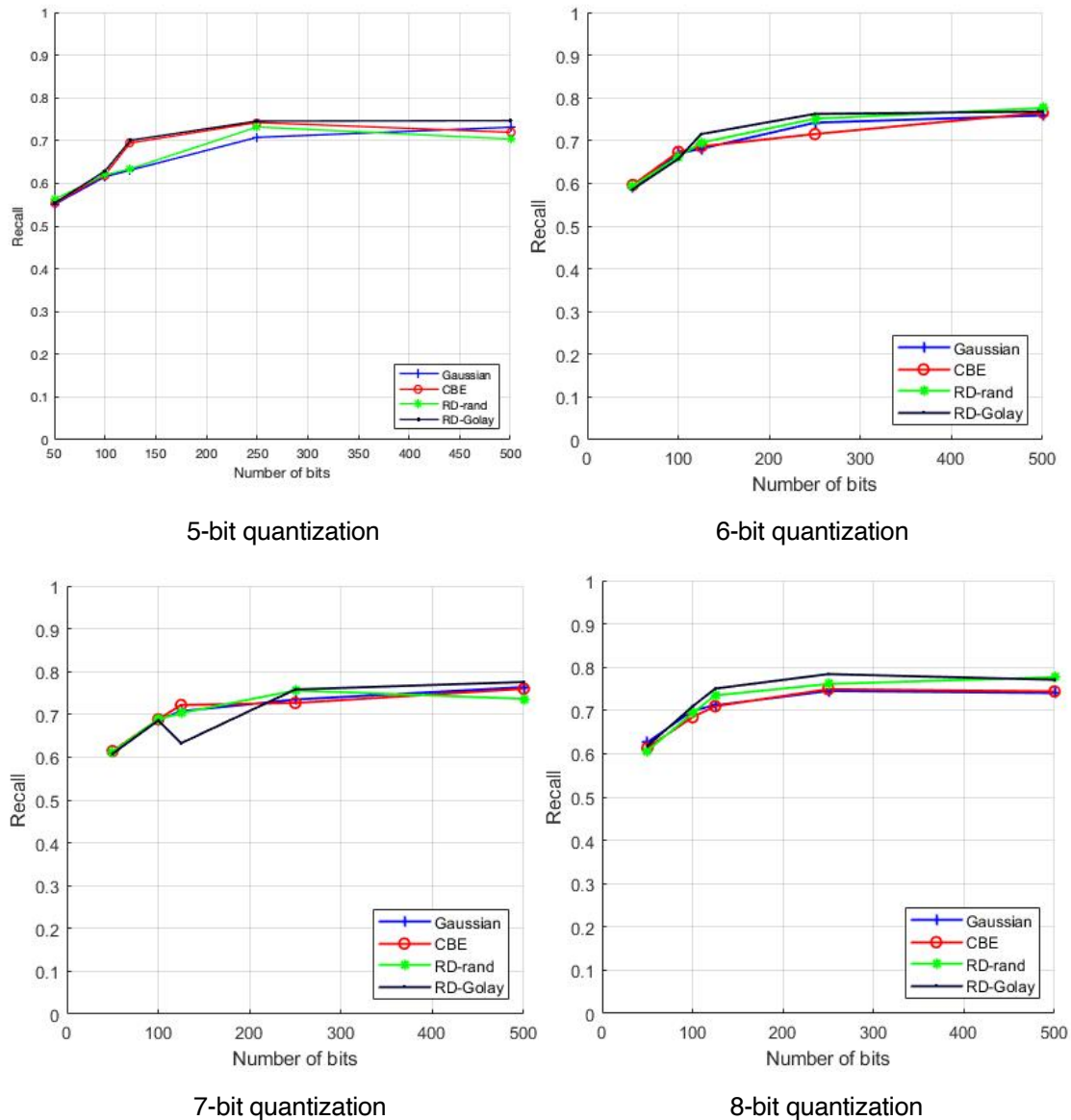
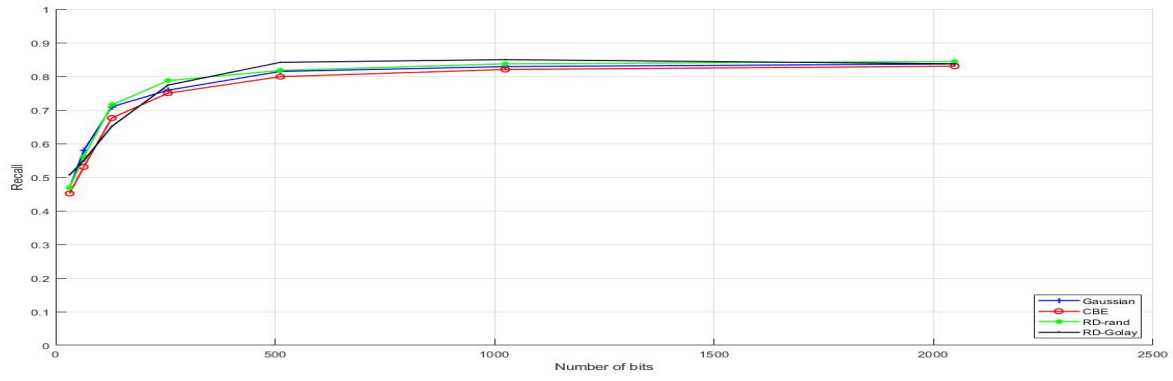


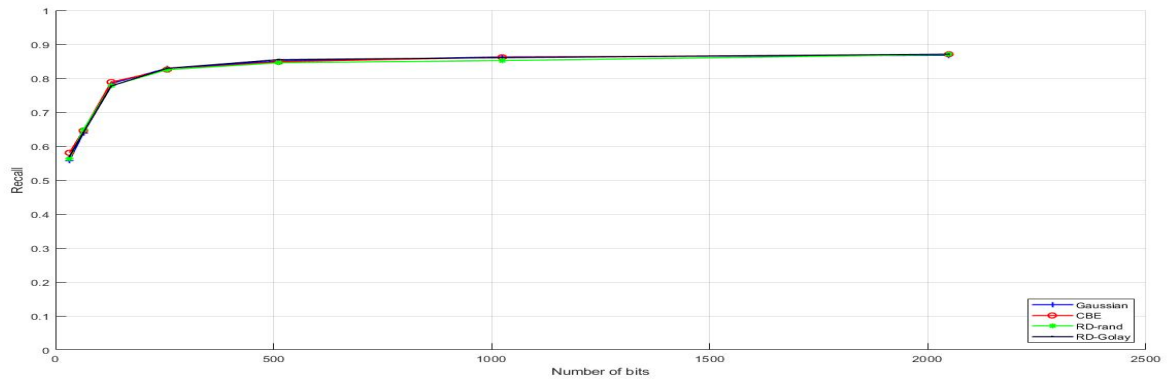
Figure 3.8: A recall comparison of four operators on Caltech 101 dataset using SqueezeNet network with multi-bit quantization

Moving to the testing of VGG19 on the Caltech 101 dataset, the results from figure 3.9 show that a big model like VGG19 is able to reach an 80% recall rate at a small number of bits per descriptor, around 128 bits per descriptor, with only 5-bit quantization for all four operators. Having said this, the recall rate improves slightly after 6-bit quantization at around 80%-88%.

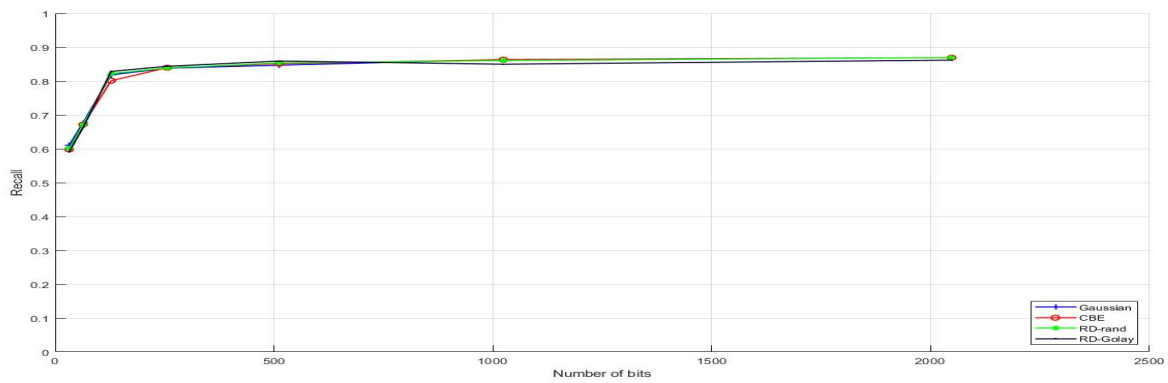
# CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION



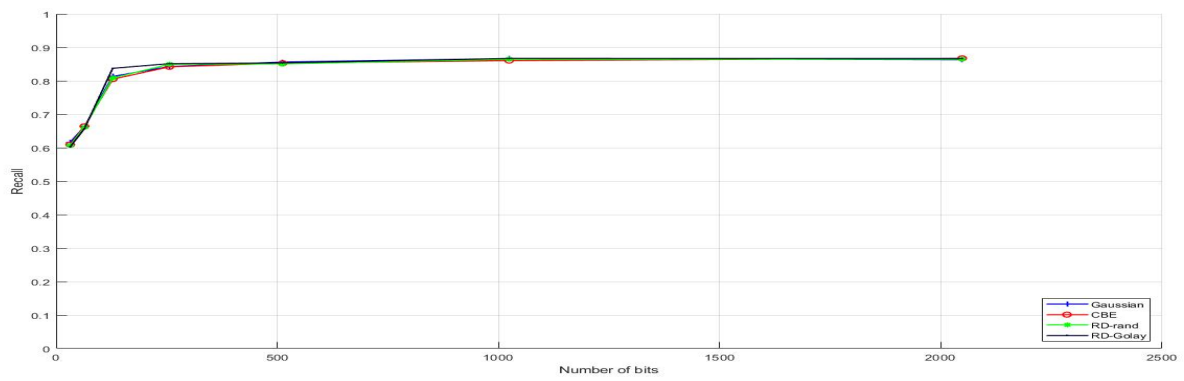
4-bit quantization



5-bit quantization



6-bit quantization



7-bit quantization

Figure 3.9: A recall comparison of four operators on Caltech 101 dataset using VGG19 network with multi-bit quantization

## CHAPTER 3: QUANTIZATION EFFECT ON GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

### • *Conclusion*

The results from this simulation highlight that with low-bit quantization positive effects can be achieved on both small-size CNN model and large-size CNN model. With around 100 bits per descriptor, SqueezeNet can reach 50% recall with only 4 or 5-bit quantization. On the other hand, with around 128 bits per descriptor, VGG19 can achieve 80% with only 5-bit quantization. Nonetheless, the recall rate increases slightly with higher-bit quantization. This means, in addition to the application of dimensionality reduction operators, further quantization using uniform scalar quantizer provides positive results on the image classification task.



### 3.4 Summary

This chapter demonstrates the effect of quantization on general image classification and potentially on image retrieval. To begin with the SIFT features, the feature matching experiment shows that a positive trend can be seen with low-bit quantization. Furthermore, elapsed time can be less which addresses the promising outcomes when applying uniform scalar quantizer on CBIR and classification task. Looking at the CNN feature experiment, after testing different sizes of CNN models, SqueezeNet and VGG19, the classification results show that low-bit quantization helps to achieve an 80% recall rate in the case of VGG19 and a 50% recall rate in the case of SqueezeNet while using only a 100 or 128 bit per descriptor. The application of the proposed quantization approaches, in addition to the dimensionality reduction operators, potentially provides efficient and effective results of CBIR and classification tasks on low computational devices such as mobile devices and IoT.

## Chapter 4: Fast Dimensionality Reduction for General Image Retrieval and Classification

For the purpose of maximizing original image features and avoiding potential error from feature selection, feature extraction is chosen to reduce high-dimensional data to the lower space. Additionally, to fasten this dimensionality reduction process with a less complex structure and preserve the correlation between variables, a linear transformation is an appropriate tool to achieve the aim of this research. The following state-of-the-art techniques are implemented in various simulations in this Chapter.

### 4.1 Existing Fast Dimensionality Reduction Methods

#### I. Random Projection

Random Projection from Gaussian or normal distribution is one of the most widely used and theoretically optimal non-adaptive techniques in dimensionality reduction [170][182][183][184]. Its theoretical foundation is from the Johnson-Lindenstrauss lemma (JLL) [167][184][185] which proves that high-dimensional data can be linearly embedded and randomly projected into much lower-dimensional space while the pairwise distances between data points are nearly preserved without dependence on the original dimensionality of the input data. Below is the calculation of the lemma.

$$p > \frac{\ln(n)}{\varepsilon^2} \quad (4.1)$$

where the  $n$  data size, desired error limit at  $\varepsilon$ , and the minimum number of  $p$  dimensions based on random projection with high probability.

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

Therefore, the Gaussian Random Projection can reduce the high-dimensional data,  $m$  dimensions, by matrix multiplication with its  $m \times p$  matrix of standard normal random values  $R_{i,j} \sim N[0,1]$

$$Y_{n \times p} = \frac{1}{\sqrt{p}} X_{n \times m} R_{m \times p} \quad (4.2)$$

where  $Y_{n \times p}$  is the low dimensional feature matrix,  $X_{n \times m}$  is the original feature matrix, and the scalar  $\frac{1}{\sqrt{p}}$  accounts for the impact on pairwise distances of working in the lower dimensional space. Furthermore, this randomness comes from atmospheric noise which is the True Random Number Generator that can be easily picked up with a normal radio. This atmospheric noise produces random numbers that pass statistical checks and for many purposes is better than the pseudo-random number algorithms as it statistically fits many natural phenomena which are unpredictable [252].

As the Random Projection is simple to implement in practice, easy to analyze, and computationally efficient, it is ubiquitous in various fields when dealing with high-dimensional data which is expensive to perform the calculation, the sparse number of samples which is difficult to reliably calculate covariances or dissimilarities, the lack of entire dataset access, and more importantly the unaffordable calculation of an orthogonal transform. Having said this, when compared to other orthogonal projections, the Random Projection could introduce more distortions, degrade the performance of the approximate solution, and increase processing time in response to the projection process.

Despite the benefits of the Gaussian Random Projection aforementioned, its non-orthogonal nature makes it less desirable compared to the orthogonal projections which are able to make the data reconstruction much simpler.

# CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

## II. Principal Component Analysis (PCA)

Despite Random Projection provides the optimal result in dimensionality reduction, its time-consuming process could be a huge hindrance in many applications, in particular on low computational devices. Therefore, faster approaches have been introduced including the famous Principal Component Analysis (PCA) [130][135][136][137][169][186]. PCA is an orthogonal transform which generates a new set of uncorrelated variables, called principal components, that maximize variance. Each principal component is a linear combination of the original variables and all principal components are orthogonal to each other. The components are ordered by the number of variances. The first principal component, a single axis in space, describes the largest possible variance which accounts for as much variability as possible. The second principal component is another axis in space which describes the largest possible remaining variance. The maximum number of principal components could be as large as the original set of the variable. However, in most cases, only the first few principal components could cover over 80% of the total variance of the original data [187].

In order to generate each PCA, there are three steps to implement, standardizing the features, calculating the standardized feature covariance matrix, and calculating the Eigen vectors and Eigen values of the covariance matrix.

The standardization of the original variable ranges helps to eliminate the dominance of those variables with larger ranges over those with smaller ranges. Therefore, each variable range would equally contribute to the analysis which leads to unbiased results. The standardization can be calculated as follows.

$$X^s = \frac{X - \bar{X}}{\sigma_X} \quad (4.3)$$

Next, the covariance matrix computation. This step identifies the correlation or uncorrelation between every two variables. The covariance matrix is a  $p \times p$

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

symmetric matrix, where  $p$  is the number of dimensions, which has entries the covariances associated with all possible pairs of the original variables.

$$C = \begin{bmatrix} p(X_1, X_1) & \cdots & p(X_1, X_m) \\ \vdots & \ddots & \vdots \\ p(X_m, X_1) & \cdots & p(X_m, X_m) \end{bmatrix} \text{ if } \sigma_j^2 = 1 \forall j = 1, \dots, m$$

Figure 4.1: The covariance matrix

The matrix can be calculated below.

$$C_{X_{m_1} X_{m_2}} = \frac{\sum_{i=1}^n (X_{m_1} - \bar{X}_{m_1})(X_{m_2} - \bar{X}_{m_2})}{(n-1)} \quad (4.4)$$

Lastly, the Eigen vector and Eigen value computation. As the Eigen vector is the direction of the axis which creates a principal component and the Eigen value is the Eigen vector's coefficient which addresses an amount of variance carried in each principal component, the computation could simply be done by ranking the highest Eigen values in order to select the most significant principal components. Furthermore, the percentage of variance or information accounted for by each component is also critical since only the significant components should be selected to form a matrix of vectors, called feature vectors. As a result, the lesser principal components kept, the lesser the feature dimension is.

Therefore, PCA is considered one of the most information preserved techniques, yet simple, in dimensionality reduction with a little accuracy trade-off. It contributes to better visualization and easier data analysis from its smaller dataset, as well as fast computation and less memory required through its feature extraction for the learning algorithm. Nevertheless, if the original variables are not a strongly correlated or weak relationship between variables, PCA might cause a significant loss of important information when dimensions are reduced. For example, when most of the correlation coefficients are smaller than 0.3, PCA should not be used. Additionally, PCA contributes to higher computational complexity when compared to Random Projection.

# CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

## III. Circular Binary Embedding (CBE)

Circulant matrix plays significant role in digital image processing regarding image reconstruction and compression [162][166][167]. The circulant matrix is a square matrix generated from a vector as the first row (or column) and circularly shifted by one element to the right in order to create a second row. The process is repeated for the second row to create the third row and so on.

$$C = \begin{pmatrix} C_0 & C_1 & C_2 & C_3 & \dots & C_{n-1} \\ C_{n-1} & C_0 & C_1 & C_2 & C_3 & \vdots \\ C_{n-2} & C_{n-1} & C_0 & C_1 & C_2 & \vdots \\ C_{n-3} & C_{n-2} & C_{n-1} & C_0 & C_1 & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ C_1 & C_2 & \dots & C_{n-2} & C_{n-1} & C_0 \end{pmatrix}$$

Figure 4.2: An  $n \times n$  circulant matrix  $C$

Construction of an  $A$  matrix from a gaussian circulant matrix can be done with the following form.

$$A = R_I C_V D_\zeta \quad (4.5)$$

where  $I$  is a fixed subset of  $\{1, 2, \dots, N\}$  with a size of  $M$  (number of bits in reduced dimension), and  $R_I$  is a subsampling operator that restricts a vector to its entries indexed by  $I$ .  $C_V$  is a circulant matrix generated by a standard Gaussian vector  $v$  and  $D_\zeta$  is a diagonal matrix with independent random sign vector  $\zeta$  on its diagonal, i.e.,  $\Pr(\zeta_i = 1) = \Pr(\zeta_i = -1) = 0.5$

$$D = \begin{bmatrix} \sigma_0 & 0 & 0 & 0 \\ 0 & \sigma_1 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \sigma_{d-1} \end{bmatrix}$$

Figure 4.3: A diagonal matrix  $D$

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

This class of matrices is attractive as its matrix-vector multiplication can be computed with fewer operations by exploiting the fast Fourier transform. Moreover, the limiting spectral distribution of a gaussian circulant matrix seems to be complex normal and bounds are given for the probability that a circulant sign matrix is singular.

Next, to create CBE [162][166][167], the sign of  $A_x$  is taken for a simple binary operation.

$$f(x) = \text{sign}(A_x) \quad (4.6)$$

where the  $\text{sign}(\cdot)$  denotes the element-wise sign operation. If each data point ( $x$ ) is on the unit-sphere, i.e.,  $\|x_i\|_2 = 1$  ( $1 \leq i \leq Q$ ) and  $Q$  denotes the total number of points in a finite dataset  $T$ , the angular distances of  $x_i$  and  $x_j$  can be evaluated or estimated using the normalized Hamming distances of  $f(x_i)$  and  $f(x_j)$ .

Consequently, in the light of circulant structure, the use of fast Fourier transformation is enabled to improve time complexity from  $O(MN)$  to  $O(N \log N)$ , and the space complexity from  $O(MN)$  to  $O(N)$  with linear projection. Nonetheless, the computational complexity could be further improved for the application on mobile devices or the Internet of Things (IoT) which have various constraints.

# CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

## 4.2 Proposed Fast Dimensionality Reduction Methods

The proposed methods in this Chapter are based on Hadamard Projections and Discrete Cosine Transforms as follow.

### I. Hadamard Projections

The Walsh-Hadamard Transform is a widely used linear transform in image and signal processing [183][184][188][189][190][191] as it requires less storage space and time consumption. This orthogonal transformation technique is a non-sinusoidal technique that decomposes an original signal into a set of basis functions, Walsh functions, which are rectangular or square waves with values of +1 or -1. Each Walsh function has a unique sequency value in which the sequency is a generalized notion of frequency and is defined as one half of the average number of zero-crossing per unit time interval. An example of the first eight Walsh functions' sequency values is below.

Index	Walsh Function Values
0	11111111
1	1111-1-1-1-1
2	11-1-1-1-111
3	11-1-111-1-1
4	1-1-111-1-11
5	1-1-11-111-1
6	1-11-1-11-11
7	1-11-11-11-1

Table IV.I: Walsh Function Values



## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

The Walsh functions could be stored in three different ordering schemes [192], sequency ordering which has the functions in order of increasing sequency value where each row has an additional zero crossing, Hadamard ordering which has the functions in normal Hadamard order, and dyadic ordering which has the functions in Gray code order where a single bit change occurs from one coefficient to the next. The sequency ordering is a default ordering scheme which is used dominantly in signal processing applications whereas Hadamard ordering and dyadic ordering are used in controls applications and mathematics, respectively.

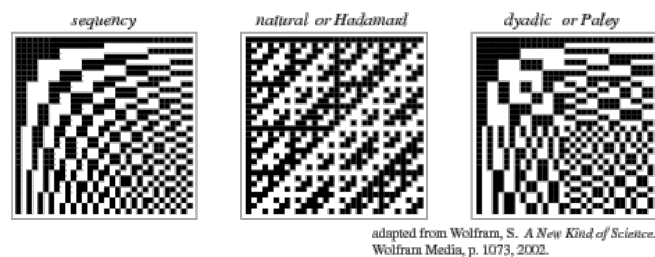


Figure 4.4: Ordering schemes [193]

After the Walsh Hadamard matrix, represented the Transform, is performed by multiplication with the feature matrix of high-dimensional data, the data containing information of the original image is concentrated at the corners of the image. As a result, the key information can be recovered and used in lower dimensions. Additionally, the Walsh Hadamard Transform (WHT) has a fast version, called the fast Walsh Hadamard Transform, which can be defined below.

$$Y_n = \frac{1}{N} \sum_{i=0}^{N-1} x_i WAL(n, i), \quad (4.7)$$

where  $x_i$  is original signal,  $i = 0, 1, \dots, N - 1$  and  $WAL(n, i)$  are Walsh functions. The  $N$  elements are decomposed into two sets of  $\frac{N}{2}$  elements, which are combined using a butterfly structure to form the fast Walsh Hadamard Transform. Its coefficients are calculated by evaluating across the rows and the columns of input signals, specified as a feature matrix or a feature vector.

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

The fact that the WHT is the real transform which only adds or subtracts the given data, it produces energy compaction for spread-spectrum analysis which contributes to higher accuracy than independent random projection like Gaussian. Moreover, its simplicity results in fast computation and less bandwidth storage requirement. Nonetheless, this technique does not consider the class label of data and might not be suitable for sparse data as it produces a dense matrix.

### II. Discrete Cosine Transforms (DCT)

The Discrete Cosine Transform is extensively used in various areas of image processing, especially image compression [194][195][196][197][198][199][200], due to its requirements in less computational complexity and less storage resource. The orthogonal linear transform represents an image of the spatial domain as the frequency domain by means of sinusoidal basis functions, Cosine functions. Firstly, an original image, high-dimensional data, is divided into 8 x 8 pixel groups. Then, each block of 8 x 8 pixel group is represented by 64 basis functions of the DCT as illustrated in figure 4.5 below.

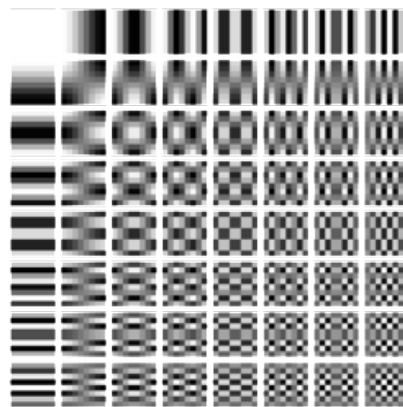


Figure 4.5: The 64 basis functions of an 8 x 8 matrix [201]

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

The transform coefficients are computed and weighted [201].

$$B_{pq} = a_p a_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos \frac{\pi(2m+1)p}{2M} \cos \frac{\pi(2n+1)q}{2N}, \quad 0 \leq p \leq M-1, \quad 0 \leq q \leq N-1 \quad (4.8)$$

$$a_p = \begin{cases} \frac{1}{\sqrt{M}} & , p = 0 \\ \sqrt{2/M} & , 1 \leq p \leq M-1 \end{cases} \quad (4.9)$$

$$a_q = \begin{cases} \frac{1}{\sqrt{N}} & , q = 0 \\ \sqrt{2/N} & , 1 \leq q \leq N-1 \end{cases} \quad (4.10)$$

The low-frequency DCT coefficients are the dominant coefficients which contain the key information of an original image, whereas the high-frequency coefficients have no or fewer effects on the quality of the output image which has lower dimensions.

Therefore, by considering only dominant coefficients which concentrate in low-frequencies mainly in the top left corners, the other coefficients in higher-frequencies can be discarded due to their low psychovisual significance. Consequently, the low-dimensional image data can be reconstructed by using the inverse Discrete Cosine Transform of each block of 8 x 8 pixel group without losing significant information.

$$A_{mn} = \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} a_p a_q B_{pq} \cos \frac{\pi(2m+1)p}{2M} \cos \frac{\pi(2n+1)q}{2N}, \quad 0 \leq m \leq M-1, \quad 0 \leq n \leq N-1 \quad (4.11)$$

$$a_p = \begin{cases} \frac{1}{\sqrt{M}} & , p = 0 \\ \sqrt{2/M} & , 1 \leq p \leq M-1 \end{cases} \quad (4.12)$$

$$a_q = \begin{cases} \frac{1}{\sqrt{N}} & , q = 0 \\ \sqrt{2/N} & , 1 \leq q \leq N-1 \end{cases} \quad (4.13)$$

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

Despite the DCT contributes to key advantages for dimensionality reduction such as maintaining a maximum amount of image information with a significantly reduced number of dimensions, providing energy compaction properties for highly correlated images, performing high accuracy and speed, and being easy to compute, there is some loss of quality in the reconstructed low-dimensional image and the appropriate determination of the dominant coefficients is not straightforward.

## 4.3 Performance Comparison

Stage 1: Matched SIFT features of hostel bedroom images after dimensionality reduction

This experiment is an extended work from the Stage 1: Matched SIFT features of hostel bedroom images after quantization in the previous chapter. SIFT features are tested with the same settings on the same dataset to see the effect of dimensionality reduction by applying the Walsh-Hadamard Transform matrix, one of the widely used transforms in signal and image processing, which decomposes a signal of high dimensional data into a set of basis functions with the values of +1 or -1 in order to reduce the complexity of the CBIR or classification problem, dampen the curse of dimensionality, and allow better generalization.

### • *Preliminary results*

Firstly, each nominal 128-dimensional vector of SIFT feature vectors is reduced to six smaller dimensions at 16-bit quantization, 100-dimensional vector, 80-dimensional vector, 64-dimensional vector, 48-dimensional vector, 32-dimensional vector, and 16-dimensional vector, by doing matrix multiplication with the Walsh-Hadamard Transform matrix. After testing several combinations of different numbers of the reduced dimensional vector with a fixed 16-bit quantization, the result in figure 4.6 shows good performances in SIFT feature matching for most pairs of hostel images, despite each vector's dimensions are reduced. On top of this, the trend of matched SIFT features is increased at 64 dimensions onwards. In terms of elapsed time, figure 4.7 highlights that different sizes of vector dimension do not greatly affect the processing time. This could be the result of not being able to fully control the network resources and the network traffic as this experiment is conducted via using a centrally managed university PC and each simulation with a different size of vector dimension is run separately. Having said that, the overall elapsed time of these simulations in

# CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

Matlab is considerably low for all classes of hostel image, a 1 - 4 minute range for each simulation.

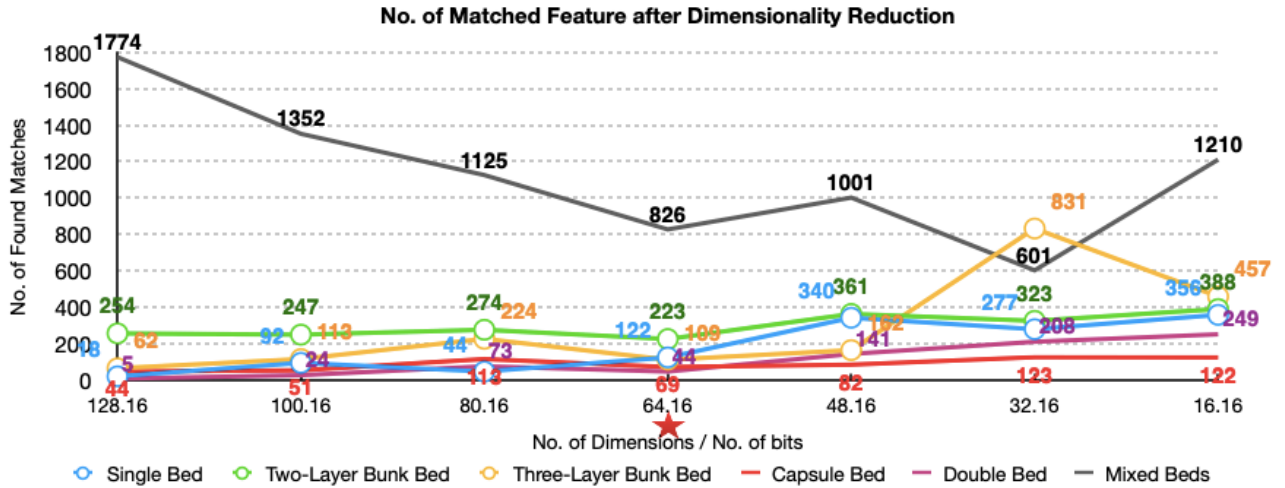


Figure 4.6: No. of matched SIFT features after dimensionality reduction at 16-bit quantization

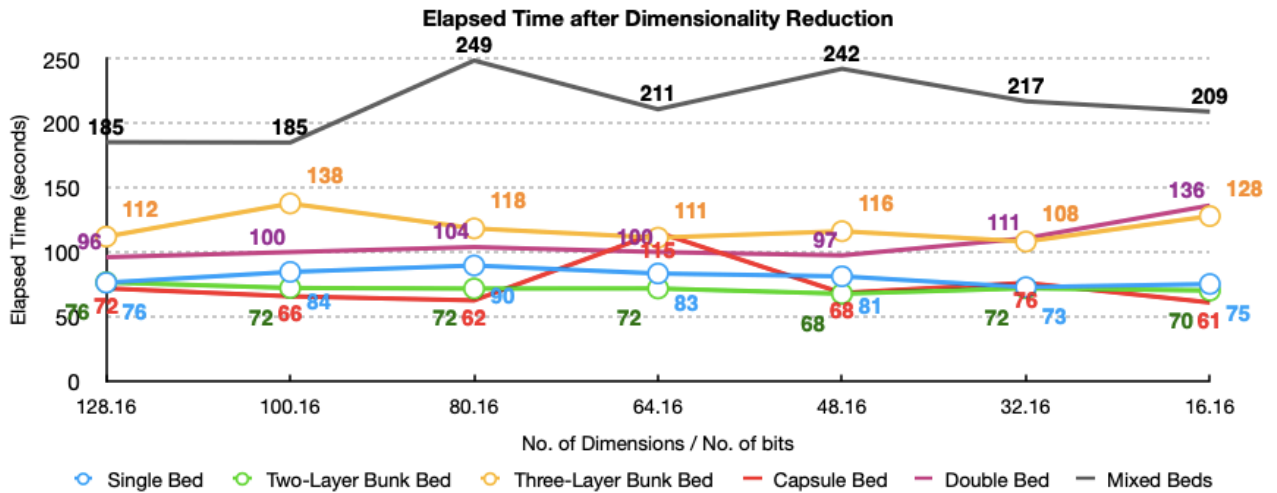


Figure 4.7: The elapsed time results of dimensionality reduction at 16-bit quantization

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

### • *Conclusion*

Based on the performances of this preliminary experiment, despite the dimensionality reduction of the SIFT 128-dimensional vector, significant information is preserved as high numbers of matched features generally continue, especially for 64-dimensional vector. Additionally, 1-4 minutes elapsed time addresses the potential application of a dimensionality reduction operator on CBIR and classification tasks on devices with memory, bandwidth, and power constraints.

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

Stage 2: Multi-size dimension of CNN features using various dimensionality reduction operators

Similar to the previous simulation, this experiment is an extended work from the Stage 2: Multi-bit quantization effect on embedded CNN features in Chapter 3. Two state-of-the-art dimensionality reduction operators, Random Gaussian and CBE, and the proposed operators, RD-rand and RD-Golay, are tested on Intel(R) Core (TM) i7-6700 CPU @3.40GHz machine with 16GB RAM with Matlab programme. In addition to Caltech 101 dataset, the Caltech 256 dataset containing 30,607 images of 256 object categories is included with a 70:30 data split, 70 per cent is for the training process and the remaining 30 per cent is for the testing process. Nonetheless, 3 different-size CNN models are implemented which are MobileNetv2, GoogLeNet, and ResNet 50 at 1-bit quantization with 5 feature lengths for each model. Furthermore, 5 iterations are completed for each simulation and recall is a selected performance measure as on classification tasks the relevant items are in focus, unlike retrieval tasks where both relevant and irrelevant items are considered in a search result using the Precision measure.

CNN model	Model size	No. of deep layer	No. of parameter
MobileNetv2 [202]	13MB	53	3.5 million
GoogLeNet [175]	27MB	22	7 million
ResNet 50 [203]	96MB	50	25.6 million

Table IV.II: MobileNetv2, GoogLeNet, and ResNet 50 models information



## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

### • *Experiment results*

From Table IV.III, it can be seen that RD-rand and RD-Golay outperform in most cases. With MobileNetv2, RD-rand gives the highest accuracy for the original 1280 bits per descriptor, 87.88% on Caltech101 and 75.20% on Caltech256, and remains the best when dimensions are reduced to 160 bits per descriptor with 69.34% on Caltech101 and 52.66% on Caltech256. Additionally, it reaches 69.51% for 640 bits per descriptor on the Caltech256 dataset. Similar to RD-rand, RD-Golay provides competitive results and achieve the highest recall rate at 86.09% for 640 bits per descriptor and 56.42% for 80 bits per descriptor on Caltech101 and 62.83% for 320 bits per descriptor on Caltech256. Looking at the GoogLeNet network, when considering the original 1024-dimensional vector, RD-Golay outperforms at 87.42% and 99.74% recall rate on Caltech101 and Caltech256, respectively. It remains the best dimensional reduction operator for 512-dimensional vector, 256-dimensional vector, and 128-dimensional vector with 97.47%, 84.96%, and 63.87% on Caltech256, as well as 79.90% for 256-dimensional vector on Caltech101. Though RD-rand can only reach the same peak as RD-Golay at 99.74% for 1024-dimensional vector on Caltech256, it still gives high accuracy across all reduced dimensions. To highlight, with original-size dimensional vector RD-Golay and RD-rand are able to achieve highly statistically significant results at 99.74% and the RD-Golay can provide statistically significant result at 97.47% with 512-dimensional vector on Caltech256 dataset [253]. Moving to ResNet50, RD-rand achieves the highest accuracy at 90.32% on Caltech101 and 79.87% on Caltech256 for the original 2048 bits per descriptor. It shows impressive performance with 76.07%, 62.08%, and 51.71% on Caltech256 for 1024, 256, and 128 bits per descriptor, respectively, whereas RD-Golay provides the highest recall rate at 78.91% and 68.22% for 256 and 128 bits per descriptor on Caltech 101 and 69.77% for 512 bits per descriptor on Caltech256.

# CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

Method	Caltech101 (9,146 images)					Caltech256 (30,607 images)				
	80 bits	160 bits	320 bits	640 bits	1280 bits	80 bits	160 bits	320 bits	640 bits	1280 bits
<b>MobileNetv2</b>										
Gaussian	54.10	67.79	77.54	83.62	87.37	<b>42.31</b>	51.96	61.83	68.86	74.46
CBE [166]	54.25	69.05	<b>79.44</b>	83.53	87.45	41.64	52.12	61.45	69.26	74.43
RD-rand (proposed)	56.06	<b>69.34</b>	79.23	85.15	<b>87.88</b>	41.74	<b>52.66</b>	62.40	<b>69.51</b>	<b>75.20</b>
RD-Golay (proposed)	<b>56.42</b>	67.14	79.31	<b>86.09</b>	87.49	40.84	52.44	<b>62.83</b>	68.87	75.11
<b>GoogLeNet</b>										
Gaussian	58.81	70.03	78.59	84.00	86.87	<b>52.72</b>	63.74	84.00	97.12	99.69
CBE [166]	<b>59.55</b>	<b>70.68</b>	78.20	<b>84.48</b>	87.24	52.24	63.07	84.49	97.14	99.64
RD-rand (proposed)	59.26	70.66	79.41	83.41	86.81	52.52	63.76	84.93	97.43	<b>99.74</b>
RD-Golay (proposed)	58.56	69.91	<b>79.90</b>	84.22	<b>87.42</b>	51.70	<b>63.87</b>	<b>84.96</b>	<b>97.47</b>	<b>99.74</b>
<b>ResNet50</b>										
Gaussian	65.72	76.97	83.19	<b>88.34</b>	90.18	50.95	<b>62.08</b>	69.22	75.50	79.14
CBE [166]	65.28	77.18	<b>84.28</b>	87.78	90.16	50.31	62.10	69.32	75.66	79.36
RD-rand (proposed)	66.68	77.66	83.73	87.99	<b>90.32</b>	<b>51.71</b>	<b>62.08</b>	69.62	<b>76.07</b>	<b>79.87</b>
RD-Golay (proposed)	<b>68.22</b>	<b>78.91</b>	83.35	87.17	89.74	50.88	61.33	<b>69.77</b>	75.97	79.67

Table IV.III: A recall comparison of four operators on Caltech 101 and Caltech 256 datasets using MobileNetv2, GoogLeNet, and ResNet 50 networks

## • Conclusion

Based on the results from this multi-size dimension of CNN features using various dimensionality reduction operators experiment, it can be seen that, when high dimensions of deep features are reduced, the proposed RD-rand and RD-Golay operators provide impressive performances with the highest accuracy in most cases on both datasets for all three CNN models. This demonstrates their potential application on low computational devices for CBIR and classification tasks.

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

Stage 3: Binary embedding using Golay-Hadamard matrices

**(This peer-reviewed manuscript is published on ICME 2021 conference proceedings)**

- *Binary embedding*

Various deep hashing methods [161][168][204][205][206][207][208][209] have also been investigated to generate binary codes directly from deep learning neural networks. Binary code is attractive as it is more memory efficient. It also allows fast query by computing Hamming distance in binary space. However, most of the existing works focused on supervised hashing. Besides, binary codes produced by unsupervised hashing methods are often inferior to real-coefficient CNN descriptors.

Considering a data set  $T \subset \mathbb{R}^N$ , the researcher wants to embed each  $x_i \in T$  into  $M \leq N$  bits so that for any two points  $x_i, x_j \in T$ , their pairwise Euclidean (or angular) distances can be preserved. A typical way is to first multiply each  $x_i$  with an  $M \times N$  data-independent projection matrix  $A$ , and then map  $Ax$  to binary bits  $\{1, -1\}^M$  using a given function. In what follows, a brief review of binary embedding using independently identically distributed (i.i.d.) Gaussian random matrices and Gaussian circulant matrices (GCMs) is provided.

Recall that the classical Johnson-Lindenstrauss Lemma [185] states that if  $A$  has independent Gaussian entries with zero-mean and unit variance, then with very high probability, the following inequality holds for  $\epsilon > 0$  and all pair of points  $x_i, x_j$  in a finite data set  $T$

$$\left| \left\| \frac{1}{\sqrt{M}} A(x_i - x_j) \right\|_2^2 - \|x_i - x_j\|_2^2 \right| \leq \epsilon \|x_i - x_j\|_2^2 \quad (4.14)$$

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

provided that  $M \leq \mathcal{O}(\epsilon^{-2} \log Q)$ , in which  $Q$  denotes the total number of points in  $T$ .

A simple binary operation is to take the sign of  $Ax$ , as given below

$$f(x) = \text{sign}(Ax) \quad (4.15)$$

where the  $\text{sign}(\cdot)$  denotes the element-wise sign operation. If each data point is on the unit-sphere, i.e,  $\|x_i\|_2 = 1 (1 \leq i \leq Q)$ , the angular distances of  $x_i$  and  $x_j$  can be evaluated or estimated using the normalized Hamming distance of  $f(x_i)$  and  $f(x_j)$ . That is, for  $\delta > 0$  when  $A$  is a standard i.i.d. Gaussian matrix with  $M \geq \mathcal{O}(\delta^{-2} \log(\frac{N}{\eta}))$  the following holds with probability  $1 - \eta$

$$\left| \frac{1}{M} d_H(f(x_i), f(x_j)) - \frac{1}{\pi} \arccos(\langle x_i, x_j \rangle) \right| \leq \delta \quad (4.16)$$

It is also known that this bound is optimal in bit complexity  $M$ [210]. It should be pointed out that the above result is for any finite dataset. The bound for  $M$  can be further improved if the datasets have a low-complexity structure. For example, when each  $x_i$  is a sparse vector with  $s$  non-zero elements, (4.16) holds when  $M \geq \mathcal{O}(\delta^{-2} s \log(\frac{eN}{s}))$  even for the infinite set.

Over the past decade, the design of fast Johnson-Lindenstrauss Transform (JLT) for binary embedding has been an active area. Several operators have been proposed such as bilinear embedding [160], fast binary embedding [211] and circulant binary embedding [166]. As this research project aims to build low-complexity operators for practical applications, a brief review on the construction of  $A$  from a Gaussian circulant matrix (GCMs) with the following form [166][212] is described below.

$$A = R_I C_\nu D_\zeta \quad (4.17)$$

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

where  $I$  is a fixed subset of  $\{1, 2, \dots, N\}$  with the size of  $M$ , and  $R_I$  is a subsampling operator that restricts a vector to its entries indexed by  $I$ .  $C_v$  is a circulant matrix generated by a standard Gaussian vector  $v$  and  $D_\zeta$  is a diagonal matrix with random sign vector  $\zeta$  on its diagonal, i.e.,  $\Pr(\zeta_i = 1) = \Pr(\zeta_i = -1) = 0.5$ . This class of matrices is attractive as its matrix-vector multiplication can be computed with only  $\mathcal{O}(N \log N)$  operations [167].

Circulant binary embedding (CBE) using (4.15) and (4.17) was investigated in [167] for vectors on the unit sphere. If the maximum magnitude of  $x_i$  is small, i.e.,  $\|x_i\|_\infty = \mathcal{O}\left(\frac{\log N}{\sqrt{N}}\right)$ , the Hamming distance  $d_H(f(x_i), f(x_j))$  can be used to estimate their angular distances. For an arbitrary dataset, one can pre-process each data vector by multiplying it with a randomly modulated Hadamard matrix. That is,  $A$  can be replaced by  $A = R_I C_v D_\zeta H D_{\ell}$  where  $H$  is an  $N \times N$  Hadamard matrix and  $\ell$  is a random sign vector.

### • Problem formulation

The goal here is to design memory efficient and fast-computable binary embedding for CNN features. Note that for fast locality sensitivity hashing (LSH), it is shown empirically [213] that a full random matrix can be approximated by  $A = R_I H D_{\zeta_2} H D_{\zeta_1} H D_{\zeta_0}$  without much performance degradation in approximate nearest neighbour search, where  $\zeta_i (i = 0, 1, 2)$  are random sign vectors. This construction is for arbitrary finite data-set. When the dataset has an intrinsic low-complexity structure, a more efficient dimension operator can be used [162].

Interestingly, CNN-based image descriptors are found compressible in the wavelet domain. As a quick demo, Figure 4.8 shows the sorted wavelet magnitudes for a CNN feature vector. Here, the off-the-shelf feature is extracted from the last fully-connected layer of Resnet-50 [203] with a dimension length of  $N = 1000$ . 8-level Haar-wavelet decomposition is then applied. As one can see, most of the signal's

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

energy concentrates only on a few large wavelet coefficients. Similar results are found for other CNN architectures. Based on this observation,  $A$  is designed by using only 1 or 2 stages of  $HD_{\zeta_i}$  with deterministic sign vectors  $\zeta_i (i = 0, 1)$ .

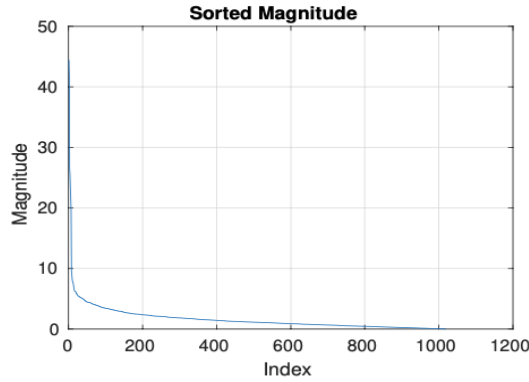


Figure 4.8: An example of sorted Haar-wavelet magnitudes of a feature vector extracted from ResNet50.

- *Golay-Hadamard matrices for binary embedding*

According to [214], a partial Hamdard matrix modulated by a Golay sequence can offer a near-optimal bound for the compressive sampling of sparse signals in the wavelet domain [214]. The definition of a Golay sequence is given below [215]

**Definition 1**

Consider two length  $- N$  binary sequences  $a = [a_0, a_1, \dots, a_{N-1}]$  and  $b = [b_0, b_1, \dots, b_{N-1}]$ . Define two polynomials  $A(z) = \sum_{n=0}^{N-1} a_n z^n$  and  $B(z) = \sum_{n=0}^{N-1} b_n z^n$ .  $a$  and  $b$  are said to be a Golay complementary pair (GCP) if

$$|A(z)|^2 + |B(z)|^2 = 2N \quad (4.18)$$

for all  $|z| = 1$ .  $a$  (or  $b$ ) is called as a Golay sequence 21 [215].

From (4.18), it is clear that  $|A(z)| \leq \sqrt{2N}$  for all  $z = e^{j\omega} (0 \leq \omega < 2\pi)$ , which means that a Golay sequence is nearly flat in the spectrum domain, i.e., it is a binary pseudo-random sequence. A GCP can be constructed directly or recursively. When

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

$N = 2^n$ , the methods in [215] can produce  $N \cdot n!$  different Golay sequences. One popular way is through the Golay-Rudin-Shapiro recursion formula [215]

$$a^{(0)} = 1 \quad (4.19)$$

$$b^{(0)} = -1 \quad (4.20)$$

$$a^{(l)} = [a^{(l-1)}, b^{(l-1)}] \quad (4.21)$$

$$b^{(l)} = [a^{(l-1)}, -b^{(l-1)}] \quad (4.22)$$

for  $l = 1, 2, \dots, n - 1$ . Unlike random sign vectors, each element  $a_i$  in a Golay sequence has an explicit form. Hence, it can be easily implemented in both software and hardware without storing the whole sequence. Next, the Golay-Hadamard matrix (GHM) is defined:

### Definition 2

A Golay Hadamard matrix (GHM) is given by  $G = HD_\zeta$ , where  $H$  is the Hadamard matrix and  $D_\zeta$  is a diagonal matrix with Golay sequence  $\zeta$  on its diagonal.

It can be shown that each row in a GHM is still a Golay sequence [214][215]. Thus,  $Gx$  calculates the inner products of  $x$  with  $N$  orthogonal binary pseudo-random sequences. Below, two classes of  $M \times N$  dimensionality reduction operators  $A$  constructed from GHMs are proposed.

- GHM-Rand:  $A = \frac{1}{\sqrt{M}} R_\Omega G_0 \quad (4.23)$

- GHM-Fix:  $A = \frac{1}{\sqrt{NM}} R_1 G_1 G_0 \quad (4.24)$

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

where  $G_i (i = 0,1)$  are GHMs,  $R_\Omega$  and  $R_I$  represent the sub-sampling operator similar as that in (4.17). In particular,  $\Omega$  in (4.23) is a uniform random subset of  $\{1, 2, \dots, N\}$  with size of  $|\Omega| = M$ ; and in (4.24),  $I$  is a fixed set with  $I = \{1, 2, \dots, M\}$ .

Assume that the feature vectors  $x_i$  are strictly sparse in the Haar-transform domain with only  $s$  non-zero coefficients each, i.e.,  $\|W_{x_i}\|_0 \leq s$  for all  $x_i \in T$ , in which  $W$  denotes the Haar-wavelet transform matrix. For GHM-Rand in (4.23), when the Golay sequence is constructed from the Golay-Rudin-Shapiro recursion using (4.19), (4.20), (4.21) and (4.22), (4.14) holds with high probability when  $M \geq \mathcal{O}(\epsilon^{-2} s \log^3(2s) \log N)$ .

This implies that GHM-Rand can be used as a fast JLT to embed wavelet-domain sparse CNN feature vectors without any binarization. At this stage, the rigorous proof with binary function  $f(x)$  seems to be addressed and the following numerical results show they can offer excellent performance for image retrieval and classification.

Note that GHM-Rand in (4.23) requires only one stage of a partial GHM. According to [216], multiplication of such a matrix requires only  $\mathcal{O}(N \log M)$  additions, which is computationally efficient. It is also memory-efficient as the randomness only comes from  $\Omega$ . In the particular case when  $M = N$ , GHM-Rand becomes a deterministic operator as well. GHM-Fix in (4.24) is completely deterministic, which includes a cascade of 2 different GHMs requiring  $\mathcal{O}(N \log N)$  operations.



## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

Table IV.IV compares the proposed GHM-Rand and GHM-Fix operators with i.i.d. Gauss and GCMs for memory requirement and computational cost. As can be seen here, the proposed systems have reduced randomness and lower computational complexity.

Operators	No. of Random Coefficients		No. of Operations	
	Floats	Binaries	Multiplications	Additions
i.i.d. Gauss	$\mathcal{O}(MN)$	0	$\mathcal{O}(MN)$	$\mathcal{O}(MN)$
GCM [167]	$N$	$N$	$\mathcal{O}(N \log N)$	$\mathcal{O}(N \log N)$
GHM-Rand	0	$N$	0	$\mathcal{O}(M \log N)$
GHM-Fix	0	0	0	$\mathcal{O}(N \log N)$

Table IV.IV: Comparison of the number of random coefficients and computation costs for different operators

### • Simulation Results

To evaluate the performance of the proposed operators, simulations are carried out for different CNN architectures and different image datasets in image retrieval and classification.

#### Results on unsupervised image retrieval

For unsupervised image retrieval, the test data-sets are the standard Oxford 5k [217] and Paris 6k data-sets [173] along with their revised versions ROxford 5k and RParis 6k [218]. The CNN architecture is based on the Resnet101-AP-GeM [193][219] trained from Resnet101[203] on Landmarks-full dataset [220]. The source codes on GitHub [221] are used for evaluation. In the original setting [221], each CNN feature has a length of  $N = 2048$  with each float coefficient stored in 32-bits, i.e.,  $2048 \times 32=65536$  bits altogether. Binary embedding using i.i.d. Gauss, GCM [167], GHM-Rand and GHM-Fix are applied for the original CNN feature vectors. Hamming distance is then calculated for the approximate nearest neighbour search. For comparison purpose, results of dimensionality reduction

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

using PCA with  $PCA_w$  are also included, where the whitening operator is also learnt from the Landmarks-full dataset [193]. To produce  $M = 32d - bits$ , the  $d \times N PCA_w$  operator is applied with whitening power of 0.25. The performance is measured via standard mean average precision (mAP) [193]. Following the convention, only the annotated region of interests is used. The results are shown in Table IV.V. As one can see from here, the performance of  $PCA_w$  drops quickly with the decrease of  $M$ . Despite their low complexity, the proposed operators offer the best performances in nearly all cases for a given  $M$  except for a couple of cases, in which GHM-Rand and GHM-Fix are slightly worse than those of i.i.d. Gauss matrix. In fact, when  $M = 2048$  bits, GHM-Rand becomes a fixed operator and its performances are still very close to those of the original one with 65536 bits (2048 floats). Although GHM-Fix is a data oblivious and deterministic operator, it offers very similar performance to those of full i.i.d. Gauss matrix and GCMs.

# CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

No. of Bit $M$	Method	Paris 6k	RParis 6k			Oxford 5k	ROxford 5k		
			E	M	H		E	M	H
65536	Original [193]	92.95	90.76	80.31	60.86	89.12	83.28	67.13	42.26
2048	$PCA_w$ [193]	88.36	86.38	75.94	54.15	79.88	71.16	56.30	27.234
	i.i.d. Gauss	91.28	<b>89.87</b>	77.34	55.93	87.05	81.72	64.31	38.56
	GCM [204]	91.87	89.70	77.51	56.70	84.94	79.30	62.82	36.78
	GHM-Rand	<b>92.20</b>	89.74	<b>78.65</b>	<b>58.46</b>	<b>87.95</b>	<b>81.95</b>	<b>65.64</b>	<b>40.73</b>
	GHM-Fix	91.81	89.78	78.49	58.01	85.89	80.83	63.41	37.61
1024	$PCA_w$ [193]	82.26	78.42	63.39	45.55	75.26	64.66	50.51	23.00
	i.i.d. Gauss	90.51	85.69	74.50	53.31	84.02	<b>78.65</b>	59.92	32.87
	GCM [204]	89.98	87.71	74.20	51.59	82.23	73.65	57.77	32.08
	GHM-Rand	<b>90.92</b>	<b>88.55</b>	<b>76.04</b>	<b>54.61</b>	<b>84.85</b>	77.75	<b>61.54</b>	35.29
	GHM-Fix	90.02	87.76	75.21	53.39	84.69	77.06	61.12	<b>35.45</b>
512	$PCA_w$ [193]	71.97	63.96	57.02	35.36	55.79	42.56	31.95	8.62
	i.i.d. Gauss	86.59	84.56	69.32	44.65	78.60	68.93	52.77	26.08
	GCM [204]	85.84	83.56	69.03	44.56	79.76	68.76	51.95	26.69
	GHM-Rand	<b>87.53</b>	<b>85.31</b>	<b>71.57</b>	46.68	<b>81.93</b>	<b>73.18</b>	<b>57.38</b>	30.42
	GHM-Fix	86.83	84.40	70.34	<b>46.74</b>	80.43	71.52	56.10	<b>31.64</b>

Table IV.V: Comparison of mean Average Precision (mAP) for different dimensionality reduction operators. For RParis 6k and ROxford 5k, “E”, “M” and “H” represent the easy, medium and hard subsets, respectively [218]. Bold values indicate the best results for a given  $M$  and dataset.

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

Table IV.VI further compares the results of the proposed systems with different state-of-the-art supervised deep hashing methods such as supervised semantics-preserving deep hashing (SSDH) [209], hierarchical deep hashing (HSH) [207] and un-supervised ones such as Deepbit [208], pixels to binary (P2B) codes [168] and embedding and aggregation on selective convolution features (EASC) [161]. One can observe the proposed GHM-Rand operator offers the best performance except for Paris 6k at 256 bits. The performance of GHM-Fix is also very close to that of GHM-Rand. These benchmark methods often require much more complicated training and/or post-processing to produce binary codes. On the other hand, our proposed system only requires simple multiplication of CNN descriptors with GHM(s) followed by a sign operation. In addition, it is worth noting that the accuracy of retrieval systems using CNN features highly depends on the scale of the training dataset. Despite all experiments presented in Table IV.VI adopt pretrained CNN models which are extensively trained on the ImageNet dataset, when new tasks are implemented on Paris 6K and Oxford 5K datasets which are different in size, the accuracy gap (10.20% - 22.20% difference) shows while using the same method and the same bit length. Having said that, the proposed operators provide smaller accuracy gaps (5.33% - 9.45% difference) demonstrating the efficiency of both proposed operators.

# CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

Method		Paris 6k (6,392 images)			Oxford 5k (5,063 images)		
		1024 bits	512 bits	256 bits	1024 bits	512 bits	256 bits
Supervised Hashing	SSDH [209]	-	83.87	-	-	63.80	-
	HDH [207]	-	87.30	<b>85.20</b>	-	70.50	69.70
Unsupervised Hashing	Deepbit [208]	-	82.90	82.50	-	62.70	60.30
	P2B [168]	-	-	-	-	74.84	69.20
	EASC [161]	-	79.10	74.10	-	68.90	58.50
Proposed	GHM-Rand	<b>90.92</b>	<b>87.53</b>	78.94	<b>84.85</b>	<b>81.93</b>	<b>70.00</b>
	GHM-Fix	90.02	86.83	78.70	84.69	80.43	69.35

Table IV.VI: A mAP comparison of binary code retrieval with supervised and unsupervised hashing methods on Paris6k and Oxford 5k datasets. Bold values indicate the best results for a given M and dataset.

# CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

## Results on image classification

The classification performance is tested on two publicly available databases, Caltech101 (9,146 images) with 101 object categories [222] and Caltech256 (30,607 images) with 256 object categories [223]. The off-the-shelf feature vectors are extracted from the last pooling layer ( $pool5 - 7 \times 7 - s1$ ) of GoogLeNet [224]. 70% of images are used during the model training process and the remaining 30% are for testing. Matlab 2020a's default linear support vector machine (SVM) classifier is used to train and classify binary bits directly. The results are documented in Table IV.VII. Here, 50 trials are performed for each dimensionality reduction operator with random coefficients. As one can see, the proposed GHM-Rand and GHM-Fix achieve similar performance on image classification for all bit length  $M$  on both datasets.

No. of bits $M$	Method	Caltech 101	Caltech 256
16384	Original	90.31	98.88
1024	i.i.d. Gauss	86.80 $\pm$ 0.32	99.69 $\pm$ 0.20
	GCM [167]	87.24 $\pm$ 0.31	99.64 $\pm$ 0.21
	GHM-Rand	86.81	<b>99.74</b>
	GHM-Fix	<b>87.75</b>	<b>99.74</b>
512	i.i.d. Gauss	84.00 $\pm$ 0.37	97.12 $\pm$ 0.20
	GCM [167]	<b>84.48</b> $\pm$ 0.36	97.14 $\pm$ 0.22
	GHM-Rand	83.40 $\pm$ 0.32	97.43 $\pm$ 0.19
	GHM-Fix	<b>84.49</b>	<b>97.47</b>
256	i.i.d. Gauss	78.59 $\pm$ 0.38	84.00 $\pm$ 0.19
	GCM [167]	78.20 $\pm$ 0.32	84.49 $\pm$ 0.21
	GHM-Rand	79.41 $\pm$ 0.28	84.93 $\pm$ 0.19
	GHM-Fix	<b>80.13</b>	<b>84.96</b>

Table IV.VII: Multi-class classification accuracy (%) on GoogLeNet features for Caltech101 and Caltech256 datasets. The best mean accuracy is highlighted in bold.

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

### • *Conclusion*

In this stage, the binary embedding of CNN feature vectors for low-powered, low-buffered devices, e.g., mobile or Internet of things (IoT) devices is studied. Two fast dimensionality reduction operators are proposed based on Golay-Hadamard matrices (GHMs). In particular, to embed an  $N$ -dimensional feature vector into  $M$  bits, GHM-Rand requires  $N$  random binary bits and  $\mathcal{O}(M \log N)$  additions along with  $N$  sign flipping operations. GHM-fix is completely deterministic with  $\mathcal{O}(N \log N)$  additions and  $2N$  sign flipping operations. To demonstrate the effectiveness of the proposed operators, simulation results are carried out for image instance retrieval and image classification on some popular image datasets. Despite their low complexity in both computation and storage, GHM-Rand and GHM-Fix offer competitive (or even better) performance to PCA, full random Gaussian matrices and Gaussian circulant operators. This indicates their promising applications in practical mobile or IoT applications.

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

### Stage 4: Fast random-sampling using Discrete Cosine Transform

- *Iterative quantization (ITQ) with PCA and random rotations*

Similar to the binary embedding using independently identically distributed (i.i.d.) Gaussian random matrices and Gaussian circulant matrices (GCMs) in the previous simulation, the unsupervised PCA is one of the most commonly used compression approaches, even though the same number of bits is allocated to the different variance of the data in each PCA direction which could lead to poor performance.

ITQ was applied in the similarity-preserving binary code experiment [169] to preserve the locality structure of the PCA-projected data by rotating it so that the quantization error could be minimized in the orthogonal Procrustes problem which aims to find the orthogonal matrix that most closely maps a given set of points to another given set of points [254]. By adopting the formulation of [225], the following continuous objective function was maximized.

$$\begin{aligned} I(W) &= \sum_k E(\|xw_k\|_2^2) = \frac{1}{n} \sum_k w_k^T X^T X w_k & (4.25) \\ &= \frac{1}{n} \text{tr}(W^T X^T X W), W^T W = I \end{aligned}$$

The constraint  $W^T W = I$  required the hashing hyperplanes to be orthogonal to each other, which is a relaxed version of the requirement that code bits be pairwise decorrelated. For a code of  $c$  bits,  $W$  was obtained by taking the top  $c$  eigenvectors of the data covariance matrix  $X^T X$ .

Given  $v \in R^c$  be a vector in the PCA-projected space. It can be seen in that if  $W$  is an optimal solution, then so is  $\tilde{W} = WR$  for any orthogonal  $c \times c$  matrix  $R$ . Consequently, the PCA-projected data can be orthogonally transformed with a minimized quantization loss



## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

$$Q(B, R) = \|B - VR\|_F^2, \quad (4.26)$$

where  $\|\cdot\|_F$  denotes the Frobenius norm. Nonetheless, the calculation of PCA is time-consuming and the storage and multiplication of the PCA projection matrix require high cost.

### • Discrete Cosine matrix with random-sampling

In this experiment, the binary embedding of convolutional neural network (CNN) features through fast Johnson-Lindenstrauss transform (JLT) followed by random-sampling operation (RandS) is proposed. This approach is constructed from the renowned discrete cosine transform (DCT) matrix [201] which produces short feature vectors and results in comparable classification and retrieval performances against the existing dimensionality reduction methods, while reducing randomness and computation complexity.

Similar to the Binary embedding using Golay-Hadamard matrices simulation, the goal here is to design memory efficient and fast computable binary embedding for deep learning features. The full random matrix can be approximated by  $A = R_1 D C D_{\zeta_2} D C D_{\zeta_1} D C D_{\zeta_0}$  without much performance degradation in the approximate nearest neighbour search, where  $DC$  is the discrete cosine matrix and  $D_{\zeta_i}$  is a diagonal vector with a random sign vector  $\zeta_i$  on its diagonal. This construction is also for *arbitrary* finite dataset and when the dataset has an intrinsic low-complexity structure, a more efficient dimension operator can be used.

By constructing the discrete cosine matrix, the proposed operator  $A$  (RandS) is defined as follow.

$$A = \frac{1}{\sqrt{M}} R_{\Omega} D C D_i \quad (4.27)$$

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

where  $DCD_i (i = 0,1)$  is discrete cosine matrix,  $R_\Omega$  represents the sub-sampling operator. In particular,  $\Omega$  in (4.27) is a uniform random subset of  $\{1, 2, \dots, N\}$  with a size of  $|\Omega| = M$ . Assume that the feature vectors  $x_i$  are strictly sparse with only  $s$  non-zero coefficients each, i.e.,  $\|Wx_i\|_0 \leq s$  for all  $x_i \in T$ , in which  $W$  denotes the Haar wavelet transform matrix. For RandS,  $\left| \left\| \frac{1}{\sqrt{M}} A(x_i - x_j) \right\|_2^2 - \|x_i - x_j\|_2^2 \right| \leq \epsilon \|x_i - x_j\|_2^2$  holds with high probability when  $M \geq \mathcal{O}(\epsilon^{-2} s \log^3(2s) \log N)$ . This means RandS can also be used as a fast JLT to embed wavelet-domain sparse CNN feature vectors.

Table IV.VIII compares the proposed RandS operator with i.i.d. Gauss and CBE for memory requirement and computational cost. As can be seen here, the proposed system has reduced randomness and lower computational complexity.

Operators	No. of Random Coefficients		No. of Operations	
	Floats	Binaries	Multiplications	Additions
i.i.d. Gauss	$\mathcal{O}(MN)$	0	$\mathcal{O}(MN)$	$\mathcal{O}(MN)$
GCM [166]	$N$	$N$	$\mathcal{O}(N \log N)$	$\mathcal{O}(N \log N)$
RandS	0	$N$	0	$\mathcal{O}(M \log N)$

Table IV.VIII: Comparison of the number of random coefficients and computation costs between the RandS operator and the state-of-the-art operators. In the case of ITQ, it is too high in complexity for this comparison.

### • Simulation Results

This image classification simulation is tested on three publicly available databases, Caltech101 (9,146 images) with 101 object categories [226], Caltech256 (30,607 images) with 256 object categories [227], and a subset of Places365-standard (35,600 images, 100 images per category) with 365 scene categories [228]. The off-the-shelf feature vectors are extracted from the last pooling layer (layer 151, global\_average\_pooling2d\_1) of MobileNet-v2, the small-size CNN with 13MB in size, 53 deep layers, and 3.5M parameters which seems to be feasible for mobile

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

applications. 70% of images are used during the training process and 30% of images are for the testing process.

Different dimensionality reduction operators using ITQ, i.i.d. Gauss, CBE, and proposed RandS are evaluated. Hamming distance is calculated for the approximate nearest neighbour search. Matlab 2020a's default linear support vector machine (SVM) classifier is used to train and classify binary bits. Additionally, 100 iterations are performed for each dimensionality reduction operator with random coefficients. The performance is measured via standard recall. Despite the super low complexity, Table IV.IX shows RandS becomes a fixed operator when  $M = 1280$  bits and their performances are close to those of the original one with 20,480 bits (1280 floats). Moreover, the data oblivious and deterministic RandS achieves comparable results to data-dependent ITQ or those full i.i.d. Gauss and CBE on image classification for all bit length  $M$  on all datasets.

No. of bits $M$	Method	Caltech 101	Caltech 256	Places365-subset
20480	Original	90.40	79.22	27.46
1280	ITQ	86.57	70.84	20.70
	i.i.d. Gauss	87.38	74.38	22.82
	CBE [166]	<b>87.40</b>	<b>74.54</b>	<b>22.89</b>
	RandS	86.76	72.47	21.27
640	ITQ	<b>87.13</b>	<b>70.76</b>	<b>21.35</b>
	i.i.d. Gauss	84.01	69.02	19.08
	CBE [166]	84.24	69.01	19.07
	RandS	82.13	66.77	17.79
320	ITQ	<b>84.46</b>	<b>69.10</b>	<b>19.84</b>
	i.i.d. Gauss	78.60	61.64	16.39
	CBE [166]	78.34	61.66	16.34
	RandS	75.60	58.30	13.77

Table IV.IX: Multi-class classification accuracy (%) on MobileNet-v2 features for Caltech101, Caltech256 and Places356-subset datasets. The best mean accuracy is highlighted in bold.

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

### • *Conclusion*

In this study, the fast random-sampling embedding of CNN feature vectors for devices with power, buffer, and bandwidth constraints is investigated. The proposed RandS is based on a discrete cosine matrix to reduce dimensions. RandS requires  $N$  random binary bits and  $\mathcal{O}(N \log N)$  additions along with  $2N$  sign flipping operations. The classification result shows that despite super low complexity in both computation and storage, the data-independent RandS operator offers competitive performance to the data-dependent ITQ, full random Gaussian, and CBE operators which indicates its promising application in mobile or IoT devices.

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

### Stage 5: Compact representations in symmetric and asymmetric settings

- *Compact representation using sub-sampling*

Due to the high dimensional nature of CNN-based feature vectors, the image features could be reduced and compactly represented by adopting a sub-sampling method which takes a subset of the original data. The subset is chosen by specifying a parameter  $n$  that every  $n$ th data point the image feature will be extracted. This uniform selection method is suitable for structured data like image data for the purpose of reducing its data size prior to the beginning of the iterations.

- *Symmetric and asymmetric settings*

In this simulation, two retrieval systems are considered, symmetric setting and asymmetric setting. The symmetric setting means both client and server settings use the binary codes with 1-bit binary coefficients in order to create compact image representation. On the other hand, the asymmetric system means only the client side uses binary codes to support low-computational devices such as mobile or IoT while on the server side is storing the default descriptors in floats with 32-bit coefficients. In addition, the implemented code is adapted from Radenovic, Tolias, and Chum's work [94] using whitened and re-normalized Generalized-Mean (GeM) descriptor and Structure-from-Motion (SfM) information for training data, as well as the Siamese learning. Moreover, Gaussian embeddings and CBE are applied to further reduce the descriptors' dimensions. Furthermore, this simulation is implemented on Intel(R) Core (TM) i7-6700 CPU @3.40GHz machine with 16GB RAM and programmed in Matlab language (version R2020a).

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

### • *Simulation Results*

This classification task performs on two popular datasets, Oxford5K (5,063 images) and Paris6K (6,392 images), and their extended versions with extra 100K images added to each dataset for the testing process making Oxford105K (105,063 images) and Paris106K (106,392 images) datasets. The fine-tuned VGG with GeM are extracted for image representation and 300 iterations are performed for each dimensionality reduction operator for various bit lengths, 512 bits, 256 bits, 128 bits, and 64 bits. Three methods are proposed, sub-sampling, GeM + Gaussian, and GeM + CBE, and compared to the state-of-the-art methods. The results in Table IV.X highlight that the sub-sampling method impressively outperforms the other methods in all bit lengths, on all datasets, and on both symmetric and asymmetric settings. For the symmetric setting, with sub-sampling, mAP can be achieved 46.04%-82.54% accuracy, whereas the compared methods reach 38.30%-79.10% accuracy. In terms of the asymmetric setting, higher accuracies show at 49.08%-86.19% for sub-sampling, whereas the others have 25.70%-63.50% accuracy. Nonetheless, the proposed GeM + Gaussian and GeM + CBE also provide great performance and outperform the state-of-the-art methods in many cases in the symmetric setting with 22.40%-81.01% and 22.45%-79.53% accuracy for GeM + Gaussian and GeM + CBE, respectively. Moreover, in an asymmetric setting, both proposed methods also achieve higher accuracies than the compared methods on all cases with 29.08%-83.48% accuracy for GeM + Gaussian and 28.75% - 81.84% accuracy for GeM + CBE. On top of this, they even give higher accuracies than the original GeM with a large margin as GeM only reaches 25.70%-63.50% accuracy,

# CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

No. of bits $M$	Method	Oxford5k (5,063 images)	Oxford105k (105,063 images)	Paris6k (6,392 images)	Paris106k (106,392 images)
<b>Symmetric</b>					
512	M <sub>MAX</sub> + ... + ITQ [161]	57.30	49.80	74.70	56.10
	siaMAC + ... + ITQ [161]	68.90	60.90	79.10	70.30
	GeM + Gaussian (proposed)	78.27	71.11	81.01	72.18
	GeM + CBE (proposed)	76.05	68.52	79.53	69.68
	Sub-sampling (proposed)	<b>80.09</b>	<b>73.38</b>	<b>82.54</b>	<b>74.07</b>
256	M <sub>MAX</sub> + ... + ITQ [161]	45.00	38.30	63.00	50.50
	siaMAC + ... + ITQ [161]	58.50	49.10	74.10	63.60
	GeM + Gaussian (proposed)	69.74	59.84	74.24	62.37
	GeM + CBE (proposed)	67.53	57.86	73.04	60.14
	Sub-sampling (proposed)	<b>75.60</b>	<b>66.97</b>	<b>81.09</b>	<b>71.23</b>
128	M <sub>MAX</sub> + ... + ITQ [161]	-	-	-	-
	siaMAC + ... + ITQ [161]	-	-	-	-
	GeM + Gaussian (proposed)	55.13	42.18	62.83	47.15
	GeM + CBE (proposed)	54.12	41.49	62.36	45.83
	Sub-sampling (proposed)	<b>69.13</b>	<b>59.38</b>	<b>77.05</b>	<b>65.40</b>
64	M <sub>MAX</sub> + ... + ITQ [161]	-	-	-	-
	siaMAC + ... + ITQ [161]	-	-	-	-
	GeM + Gaussian (proposed)	36.14	22.40	46.93	28.76
	GeM + CBE (proposed)	35.53	22.45	46.70	28.03
	Sub-sampling (proposed)	<b>56.82</b>	<b>46.04</b>	<b>65.42</b>	<b>51.82</b>

# CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

No. of bits $M$	Method	Oxford5k (5,063 images)	Oxford105k (105,063 images)	Paris6k (6,392 images)	Paris106k (106,392 images)
<b>Asymmetric</b>					
512	siaMAC+MAC [218]	56.20	45.50	57.30	43.40
	siaMAC+RMAC [218]	46.90	37.90	58.80	45.60
	GeM [174]	56.20	44.40	63.50	45.50
	GeM. +Gaussian (proposed)	81.42	75.05	83.48	75.66
	GeM + CBE (proposed)	78.97	71.90	81.84	72.77
	Sub-sampling (proposed)	<b>84.30</b>	<b>79.31</b>	<b>86.19</b>	<b>79.48</b>
256	siaMAC+MAC [218]	-	-	-	-
	siaMAC+RMAC [218]	-	-	-	-
	GeM [174]	34.10	25.70	43.90	29.00
	GeM. +Gaussian (proposed)	75.54	66.80	78.83	68.50
	GeM + CBE (proposed)	72.93	63.85	77.27	65.62
	Sub-sampling (proposed)	<b>82.00</b>	<b>75.23</b>	<b>85.18</b>	<b>77.65</b>
128	siaMAC+MAC [218]	-	-	-	-
	siaMAC+RMAC [218]	-	-	-	-
	GeM [174]	-	-	-	-
	GeM. +Gaussian (proposed)	64.20	51.53	70.10	55.43
	GeM + CBE (proposed)	62.81	49.90	69.36	53.39
	Sub-sampling (proposed)	<b>75.15</b>	<b>65.61</b>	<b>82.10</b>	<b>72.55</b>
64	siaMAC+MAC [218]	-	-	-	-
	siaMAC+RMAC [218]	-	-	-	-
	GeM [174]	-	-	-	-
	GeM. +Gaussian (proposed)	46.04	29.08	56.06	35.78
	GeM + CBE (proposed)	44.94	28.75	55.54	34.55
	Sub-sampling (proposed)	<b>64.53</b>	<b>49.08</b>	<b>71.44</b>	<b>57.51</b>

Table IV.X: A mAP comparison of compact representations in symmetric and asymmetric settings on Oxford5K, Oxford105K, Paris6K, and Paris106K datasets using fine-tuned VGG network.

The highest mAP are highlighted in bold.



## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

### • *Conclusion*

On this image classification task, three compact representations of image data are proposed in this simulation, sub-sampling, GeM + Gaussian, and GeM + CBE under two types of settings. The symmetric setting is when both client and the server settings store VGG feature vectors in binary forms with 1-bit binary coefficients and the asymmetric system is when only the client side uses binary codes to support low-computational devices such as mobile or IoT while on the server side is storing the default descriptors in floats with 32-bit coefficients. Four bit lengths are explored, 512 bits, 256 bits, 128 bits, and 64 bits on four popular datasets, Oxford5K, Oxford105K, Paris6K, and Paris106K. The results show that sub-sampling is the best-performed dimensionality reduction operator on all bit lengths, on all datasets, and in all settings with mAP of 46.04%-82.54% and 49.08%-86.19% accuracy for symmetric and asymmetric settings, respectively. Moreover, both GeM + Gaussian and GeM + CBE provide competitive performance which can outperform the state-of-the-art methods in most cases in symmetric setting and impressively outperform all cases in the asymmetric setting. These results demonstrate the promising application of the proposed methods on devices with low-computational resources.

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

### 4.4 Summary

In this chapter, various dimensionality reduction-focused simulations are carried out for general image retrieval and classification. Firstly, the investigation of matched SIFT features of hostel bedroom images after dimensionality reduction is conducted. The results of this preliminary experiments illustrate the promising implemented dimensionality reduction on low-computational devices. Significant information is preserved as high numbers of matched features generally continue while feature vectors' dimensions are being reduced. Moreover, the short elapsed time in Matlab emphasizes the potential of the proposed approach.

Next, the multi-size dimension of CNN features using various dimensionality reduction operators experiment is investigated. The results show that the proposed RD-rand and RD-Golay dimensionality reduction operators outperform the state-of-the-art operators in most cases on Caltech101 and Caltech256 datasets when implementing three CNN models, MobileNetv2, GoogLeNet, and ResNet50.

Then, the simulation highlighting the novel approach of binary embedding using Golay-Hadamard matrices is addressed. CNN feature vectors are experimented and the two proposed fast dimensionality reduction operators are implemented, GHM-Rand operator which requires  $N$  random binary bits and  $\mathcal{O}(M \log N)$  additions along with  $N$  sign flipping operations and GHM-fix operator which is completely deterministic with  $\mathcal{O}(N \log N)$  additions and  $2N$  sign flipping operations. The results demonstrate that both operators offer competitive (or even better) performance to PCA, full random Gaussian matrices and Gaussian circulant operators on instance-level image retrieval and image classification tasks, despite their low complexity in both computation and storage. This indicates their promising applications in practical mobile or IoT applications.

Moving to the simulation of fast random-sampling embedding using Discrete Cosine Transform, this RandS operator requires  $N$  random binary bits and  $\mathcal{O}(N \log N)$

## CHAPTER 4: FAST DIMENSIONALITY REDUCTION FOR GENERAL IMAGE RETRIEVAL AND CLASSIFICATION

additions along with  $2N$  sign flipping operations in order to reduce the high dimensions of CNN feature vectors. The image classification results illustrated that this data-independent RandS is able to offer competitive performance to the data-dependent ITQ, full random Gaussian, and CBE operators. Similar to the previous simulations, the fast RandS dimensionality reduction operator also has a potential application on devices with power, buffer, and bandwidth constraints.

The next simulation is on the image classification task using compact representations in symmetric and asymmetric settings. The symmetric setting is when both client and the server settings store the fine-tuned VGG with GeM feature vectors in binary codes with 1-bit binary coefficients and the asymmetric system is when only the client side uses binary forms, whereas the server side stores the default descriptors with 32-bit float coefficients. Three compact representations of image data are proposed in this simulation, sub-sampling, GeM + Gaussian, and GeM + CBE. The results highlight that sub-sampling is the best-performed dimensionality reduction operator in all cases, even with all bit lengths, 512 bits, 256 bits, 128 bits, and 64 bits. Furthermore, GeM + Gaussian, and GeM + CBE also provide competitive performance which can outperform the compared state-of-the-art methods in most cases in symmetric setting and also impressively outperform all cases in the asymmetric setting. These results indicate that the proposed methods could also be applied to devices with low-computational resources.

### Chapter 5: Hostel Image Database

Despite this research project has implemented CNN features which normally require a large number of data to achieve good performance, these deep features have already pre-trained on the ImageNet dataset. Furthermore, transfer learning with parameter fine-tuning is adopted in order to accommodate simulations on smaller scale datasets and be in a better position to reach high accuracy. Therefore, in the dataset collection process, the scales of newly collected datasets are similar to the scales of publicly available datasets used in experiments for the purpose of fair comparison.

#### 5.1 Hostel image dataset (Project Contribution)

##### I. Data sources

The hostel image dataset is created for this study's content-based image classification experiments. The dataset is collected from five freely available sources which are Google with over 50% of total images, Flickr, Places365 database, online travel agencies 'websites such as [www.booking.com](http://www.booking.com) and [www.hostelworld.com](http://www.hostelworld.com). Consequently, as various data sources are included, images in this dataset could represent different hostel designs which vary from traditional to modern, simple to eclectic, and/or budget to boutique from big chain hostels which are corporately owned, usually have a similar design, and provide similar bed types and bedroom amenities such as television, cupboard, and/or bathroom to independent hostels which are usually privately owned and have unique designs.

## CHAPTER 5: HOSTEL IMAGE DATABASE



Figure 5.1: Hostel image examples of each type of hostel

Moreover, images of hostels across the world are considered in order to reduce bias such as hostels in the United States (North America), hostels in the United Kingdom (Europe), hostels in Thailand (Asia), and hostels in Ghana (Africa). Lastly, a variety of image quality from the professional and the non-professional and the time these images are available and collected, as of December 2020, could demonstrate the real-world digital image collections on current online platforms. Additionally, some images are taken by the researcher at the premises, where possible, to create this first hostel image-only dataset.

### II. Data selection and pre-processing

There is no academic scholar who suggests a number of reliable data sizes or how to calculate the appropriate data size for research in relation to image classification. Nonetheless, the key advantage of pretrained CNNs with transfer learning and fine-tuning is to improve classification accuracy when using deep features on small scale datasets and it is evidenced in Rath's experiment [88] that for the 8,677-image dataset the classification performance can reach 95% accuracy, as well as the popular caltech101 dataset which consists of 9,146 images. Therefore, **13,908** hostel images are collected for this work's hostel image dataset. This dataset is divided into two sets, training/validation set and test set. There are **13,008** hostel images in the

training/validation set (94% of all images), the training set which is used to train the algorithm/model and the validation set which is used to unbiasedly evaluate the algorithm/model while fine-tuning parameters. The test set contains **900** images (6% of all images) which is used to unbiasedly evaluate the final algorithm/model. To solve the supervised learning problem, all images in the training/validation set are labelled and grouped into **28** categories. Seven categories belong to interior images which are bunk bedroom, normal bedroom, other bedroom, bathroom, living room, laundry room, and kitchen, totaling 8,083 images (58% of all images). Since the bedroom is the key feature for hostel accommodation, three out of seven categories are allocated to three types of bedrooms. Moreover, these interior images contain complex clusters which demonstrate the greater challenge of hostel images when compared to hotel images as it is more likely that there are more than two beds in the same room in a hostel bedroom, there are more than one washing machine/dryer in a hostel laundry room, there are more than one shower/toilet in a hostel bathroom, and there are more objects in a hostel kitchen/living room. Therefore, it is possible that hostel image features consist of vaster feature types and levels such as many colours, many textures, and many shapes which account for the higher number of features to consider when performing classification tasks. Next, 21 categories belong to exterior images of various hostels in more than 20 capital cities around the world, totalling 4,925 images (42% of all images). The numbers of images per category are ranging from 49 - 1,951 images. Moving to the test set, a mix of unlabelled images from all 28 categories, 5 - 100 images per category, is separated for model testing. Moreover, this hostel image dataset contains images in different sizes, in three colour channels which are red, green, and blue (RGB), and stored in JPEG format which is an appropriate still image format for the use of image classification purposes. For non-commercial research and educational purposes, this dataset is made available on Google Drive [229]. Hostel image examples of seven categories, interior and exterior categories, and quantitative information of all categories are demonstrated in figure 5.2 and table V.I below.

# CHAPTER 5: HOSTEL IMAGE DATABASE

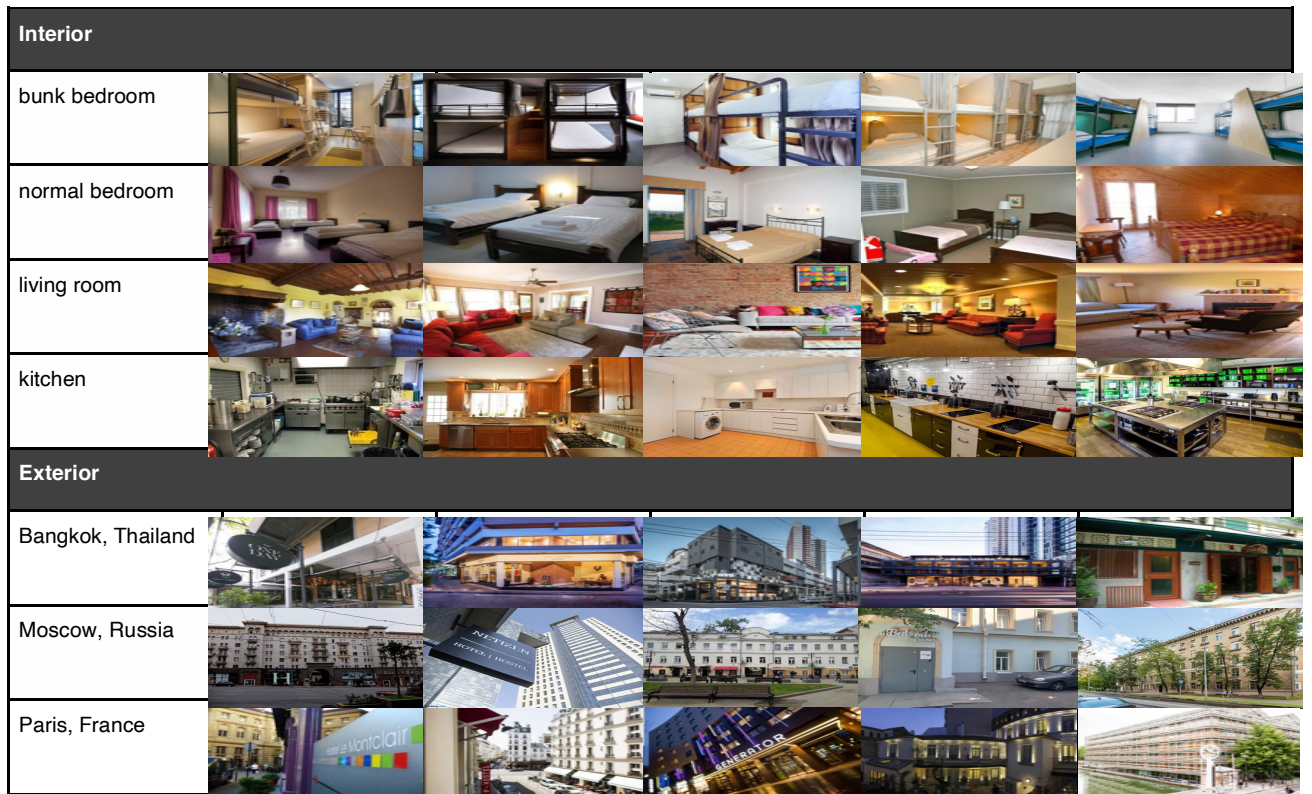


Figure 5.2: Hostel image examples of the hostel image dataset

## CHAPTER 5: HOSTEL IMAGE DATABASE

	Categories	No. of images	
		Training & Validation sets	Test set
	<b>Interior</b>		
1	bathroom	1037	100
2	bunk_bedroom	1001	100
3	kitchen	1000	100
4	laundry_room	1951	100
5	living_room	1000	100
6	normal_bedroom	1066	100
7	other_bedroom	1028	100
	<b>subtotal</b>	<b>8083</b>	<b>700</b>
	<b>Exterior</b>		
8	Abuja, Nigeria	259	5
9	Accra, Ghana	263	5
10	Ankara, Turkey	193	5
11	Bangkok, Thailand	219	5
12	Beijing, China	218	5
13	Berlin, Germany	326	5
14	Bern, Switzerland	149	5
15	Brasilia, Brazil	107	5
16	Canberra, Australia	110	5
17	Doha, Qatar	49	5
18	London, UK	305	5
19	Moscow, Russia	155	5
20	New Delhi, India	170	5
21	Others	1016	100
22	Ottawa, Canada	137	5
23	Paris, France	236	5
24	Seoul, South Korea	255	5
25	Singapore, Republic of Singapore	227	5
26	Tokyo, Japan	234	5
27	Washington, D.C., US	117	5
28	Wellington, New Zealand	180	5
	<b>subtotal</b>	<b>4925</b>	<b>200</b>
<b>Subtotal</b>		13008	900
<b>Total</b>		<b>13908</b>	

Table V.I: Quantitative information of hostel image dataset



### 5.2 London hostel building datasets\_(Project Contribution)

#### I. Data sources

In intention to create London hostel building datasets for content-based image retrieval experiments, two datasets are produced, Hostels-900 and Hostels-2K. These hostel images are gathered from three major freely available sources which are Google, with over 80% of the total images, Booking.com and Flickr to create new hostel image-only datasets, including images taken by professional and non-professional photographers who produce different image quality. Furthermore, as of April 2021 when the time of these two datasets were collected, the hostel images could be well-represented of today's real-world digital image collections.

#### II. Data selection and pre-processing

The first dataset is **Hostels-900**, consisting of 972 images of 20 London hostel buildings. The dataset is divided into two sets, training/validation set and test set. The training/validation set, 872 images (90% of all images) with 17 - 95 images for each hostel building, is comprised of the training set for model training and the validation set for unbiased model evaluation while fine-tuning parameters. The test set, 100 images (10% of all images) with 5 images for each hostel building, is for the final model evaluation with parameters fine-tuning process is completed.

Additionally, only London hostels are selected as the city is one of the world's most popular tourist destinations and where the researcher is based. A variety of both chain hostels, corporately owned and tend to have similar design within the same chain, and independent hostels, usually privately owned and tend to have unique designs, are considered. 10 of these buildings (50% of all buildings) belong to chain hostels such as Youth Hostels Association (YHA), St. Christopher's, Meininger, and Wombats and the remaining 10 buildings (50% of all buildings) belong to the independent hostels. Furthermore, images with various angles of each hostel building are included. Consequently, the Hostels-900 dataset could portray different hostel designs from

cosy to grand, traditional to modern, and/or budget to boutique without a label is given to any of these images. Furthermore, to be resemblance to Oxford5K (5,062 images) and Paris6K (6,412 images) which are one of the most widely used datasets on image retrieval experiments with unsupervised learning and they are only a few building image datasets available which were collected over ten years ago, the Hostels-900 considers only the exterior of hostel buildings. On top of this, the exterior images of hostel buildings could be more challenging for retrieval tasks when compared to that of hotel buildings since the hostel buildings tend to be smaller and could be more difficult to spot within their surroundings. As a result, it is possible that a hostel building image could have more feature types and levels such as many colours, shapes, and textures which account for a higher number of features to consider when performing retrieval tasks. In addition, all images in Hostels-900 are in 300 x 300 pixels, in RGB colour space, and stored in JPEG format.

Similar to the Hostels-900 dataset, the **Hostels-2K**, consisting of 2,380 images of 20 London hostel buildings, is an extension of the Hostels-900 where the number of images in the training/validation set is increased to 2,280 images (96% of all images) with 24 - 451 images for each hostel building, whereas the number of images in the test set remains the same, 100 images (4% of all images) with 5 images for each hostel building. Moreover, all images in Hostels-2K are also in 300 x 300 pixels, in the same RGB colour space, in JPEG format, and unlabelled or uncategorized for the purpose of unsupervised learning and instance-level image retrieval.

Hostels-900 and Hostels-2K datasets are made available on Google Drive [230][231] for non-commercial research and educational purposes. Image examples of these London hostel building datasets and further details of the image quantity of each dataset are demonstrated in figure 5.3 and are shown in Table V.II below.

# CHAPTER 5: HOSTEL IMAGE DATABASE



Figure 5.3: Image examples of London hostel building datasets

## CHAPTER 5: HOSTEL IMAGE DATABASE

	Hostel Name	Hostels-900		Hostels-2K	
		No. of images in the training/validation set	No. of images in the test set	No. of images in the training/validation set	No. of images in the test set
1	Astor Hyde Park	37	5	68	5
2	Clink 261	67	5	120	5
3	Green Rooms	17	5	24	5
4	Hyde Park Inn	31	5	81	5
5	London Backpackers Hostel	21	5	36	5
6	London Waterloo Hostel	94	5	266	5
7	Meininger London Hyde Park	32	5	52	5
8	Palmers Lodge Swiss Cottage	95	5	309	5
9	Park Villa Boutique Hostel	36	5	81	5
10	PubLove	39	5	96	5
11	Safestay London Kensington	57	5	144	5
12	SoHostel	25	5	35	5
13	St. Christopher's Inn London Bridge	23	5	71	5
14	St. Christopher's Village	25	5	44	5
15	The Birds Nest	40	5	84	5
16	The Phoenix Hostel	24	5	40	5
17	The Walrus Hostel	59	5	134	5
18	Wombat's The City Hostel London	19	5	62	5
19	YHA Central London	41	5	82	5
20	YHA London St. Pancras	90	5	451	5
	Subtotal	872	100	2,280	100
	<b>Total</b>	<b>972</b>		<b>2,380</b>	

Table V.II: Image quantity of Hostels-900 and Hostels-2K datasets, London hostel building datasets

### 5.3 Summary

Despite the fact that the digital collections of hostel images are vast and significantly growing due to the love of digital image sharing in today's society and the affordable stay in hostel accommodation, there is no publicly available database dedicated to hostel images. Therefore, it is difficult to test and/or implement the state-of-the-art technologies that could benefit this sector, especially the areas of content-based image classification and retrieval. Furthermore, hostel images tend to have more complex clusters of image contents when compared to hotel images. This means the greater challenge of hostel images could advance the current knowledge in research communities. Consequently, the first hostel image database is created in order to apply the proposed techniques for this interdisciplinary research. The database consists of three datasets, hostel image dataset and London hostel building datasets, Hostels-900 and Hostels-2K. The hostel image dataset is tested in a content-based image classification task which is a supervised learning problem. **13,908** hostel images are collected from four freely available sources, as well as some of them are taken by the researcher, that include images taken by both professional and non-professional photographers and consist of chain hostel and independent hostel images. 13,008 images are separated for the training/validation set and labelled/grouped into **28** categories with 49 - 1,951 images per category, whereas 900 images are for the test set which is a mix of unlabelled images from all 28 categories with 5 - 100 images per category for the purpose of model testing. The second and third datasets belong to London hostel building datasets which the Hostels-900 contains **972** images and the Hostels-2K contains **2,380** images of **20** hostel buildings in London with different angles for content-based image retrieval task. These images are gathered from three freely available sources with images from Google account for over 80% of total images. Similar to the hostel image dataset, some images are taken by professional photographers and some are taken by non-professional. Images from both chain hostels and independent hostels are included without labelling or categorizing them for the purpose of unsupervised learning. Their training/validation sets consist of 872 images with 17 - 95 images for each hostel building in Hostels-900 and 2,280 images with 24 - 451 images for each

## CHAPTER 5: HOSTEL IMAGE DATABASE

hostel building in Hostels-2K, while both of their test sets have 100 images which include 5 images from each building. Lastly, images in both London hostel building datasets are in different sizes and in RGB channels, as well as being stored in JPEG format.

## Chapter 6: Hostel Image Classification

### 6.1 Experiment settings

(This peer-reviewed manuscript, Transfer learning with CNNs for hostel image classification, is published in the Data Technologies and Applications journal)

#### I. Pretrained CNN models

CNN is a widely used network for image classification tasks and several other tasks in computer vision. A CNN model consists of a set of components such as convolutional layers, pooling layers, and fully connected layers. The convolutional operations traverse input matrices with convolutional kernels, which act as filters in order to automatically learn parameters such as kernel size, strides, channels, padding, and activation via the deep learning method, in the low-level feature extraction process. Then, the output is subsampled by pooling layers to learn high-level features representing the original input. The fully connected layers are placed lastly to generate numerical output or act as the classifier. Despite CNN models vary in a number of deep layers and layer types, a typical CNN architecture can be illustrated in figure 6.1 below.

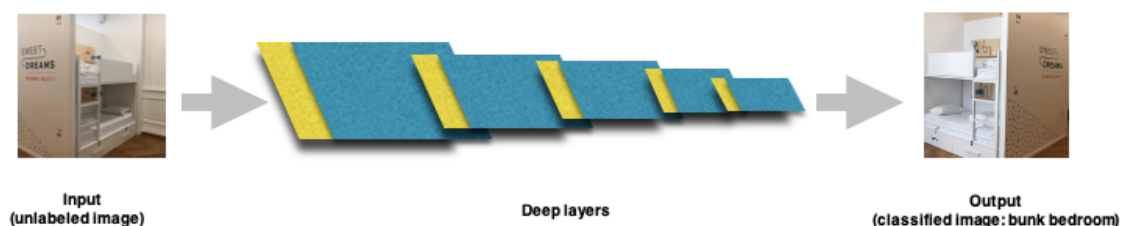


Figure 6.1: CNN model architecture

However, the accuracy of image classification using CNN highly depends on large scale database as the more information it receives, the more accurate performance results it is likely to provide, high-end computational unit to execute this compute-

intensive task in a short period of time which is costly to operate and procure, and the network depth as the deeper network could create more detailed representatives of image input and become more efficient in computation and number of parameters [51].

Several CNN models were developed and extensively trained on ImageNet, a large-scale database used in Large-Scale Visual Recognition Challenge, such as AlexNet, VGG, GoogLeNet, and ResNet. Many of these state-of-the-art neural networks are widely used in image classification tasks. For example, Dawud *et al.* [232] and Zhao *et al.* [50] applied AlexNet in their experiment, Li *et al.* [233], Mateen *et al.* [234], Shaha and Pawar [51] and Stylianou *et al.* [80] used VGG, Szegedy *et al.* [175] preferred GoogLeNet, and Mateen *et al.* [234], Stylianou *et al.* [81], and Xiao *et al.* [235] chose ResNet. Although more complex models might lead to more accurate results, to be compatible with mobile applications, in this study 11 small-size CNN models are explored which are SqueezeNet models (SqueezeNet1\_0 and SqueezeNet1\_1), ResNet models (ResNet18, ResNet34, ResNet50, ResNet101, and ResNet152), and DenseNet (Densely Connected convolutional Network) models (DenseNet121, DenseNet161, DenseNet169, and DenseNet201). SqueeneNet models are the most compact networks available which take advantage of fire module using only 1 x 1 filters in squeezed convolutional layers and a mix of 1 x 1 and 3 x 3 filters in expanded convolutional layers. ResNet models provide optimized solutions to the degradation problem, which tends to be caused by an increase of deep layers, by letting the stacked layers fit a residual mapping instead of the underlying mapping. Similar to ResNet models, the DenseNet models add shortcuts in deep layers but also concatenate the outputs from previous layers in deep dimensions. Therefore, DenseNet models use fewer parameters than ResNet to represent image features. Moreover, models with different numbers of deep layers are included for a comprehensive approach and the accuracies and relative prediction time using GPU (Graphics Processing Unit) of these models [236] are also feasible for this study. The information about the chosen CNN models is demonstrated as follow.



## CHAPTER 6: HOSTEL IMAGE CLASSIFICATION

CNN model	Model size	No. of deep layer	No. of parameter
SqueezeNet1_0 [237]	5.2MB	18	1.2 million
SqueezeNet1_1 [237]	5.2MB	18	1.2 million
ResNet18 [203]	44MB	18	11.7 million
ResNet34 [203]	87MB	34	21.2 million
ResNet50 [203]	96MB	50	25.6 million
ResNet101 [203]	167MB	101	44.6 million
ResNet152 [203]	230MB	152	58.2 million
DenseNet121 [49]	31MB	121	8 million
DenseNet161 [49]	110MB	161	29 million
DenseNet169 [49]	55MB	169	14 million
DenseNet201 [49]	77MB	201	20 million

Table VI.I: CNN models information

### II. Transfer learning technique

Similar to human judgement, the more information we receive, the better decision we make, it is essential for CNN to have sufficient information in order to achieve a good result. Therefore, large databases are created to train CNN such as ImageNet which contains over 14 million images and Place365 which contains over 1.8 million images. However, for image classification task or other tasks on a much smaller database, even with data augmentation applied by executing rotation, horizontal flipping, vertical flipping, or resizing images in the database in order to provide more information to CNN model via these artificial increase of training images, it could be difficult for the model to learn millions of its network parameters and provide good performance as it is likely to overfit the training data with low bias which lead to false

## CHAPTER 6: HOSTEL IMAGE CLASSIFICATION

performance. As a result, the transfer learning technique is introduced to small scale database tasks as an alternative to deep learning the CNN from scratch. The process of transfer learning can be explained in two steps, importing the fully pretrained CNN construction and fine-tuning its parameters, as illustrated in figure 6.2 below.

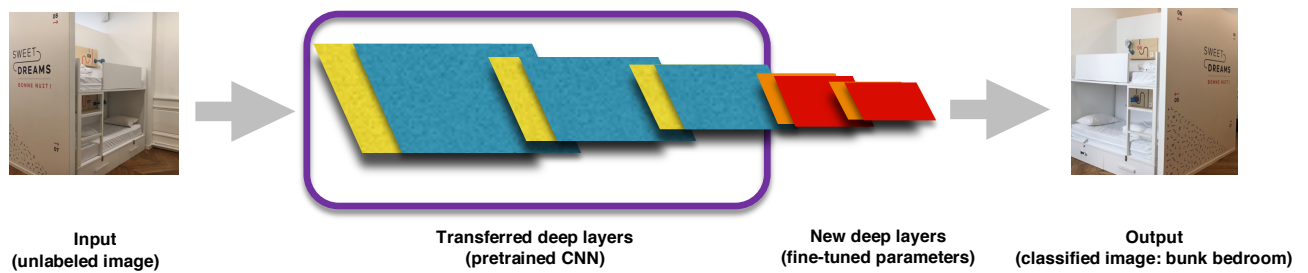


Figure 6.2: CNN transfer learning for image classification

To reduce computational cost and achieve high accuracy, transfer learning takes advantage of pretrained CNN by using the learned parameters/weights from its earlier layers which extract low-level features such as edges, patterns, corners, and gradients which are general features in any image. Despite different databases used on the pretrained CNN, these low-level features are similar and can be transferred to the new task resulting in increasing the feature extraction capability of the adapted CNN by requiring less computational time in the training process. The next step is to fine-tune parameters by making small adjustments on parameters/weights of the pretrained CNN on the latter layers which extract high-level features such as objects and events which are specific features in a given image in order to match new dataset and then, add a new classifier and train this adapted CNN for the new task. This fine-tuning step can be repeated until the desired result is achieved. Transfer learning does not only help to fasten the overall process of image classification but also reduces computational resources used such as large storage space, high-speed network connection, and multi-core processors. Furthermore, better performance could be achieved by using a lot fewer data during the training of adapted CNN.

Several previous works apply transfer learning by directly employing the pretrained CNN and fine-tuning it for their databases such as Shaha and Pawar [51], Stylianou *et al.* [81], Xiao *et al.* [235] and Zhao *et al.* [238]. Nonetheless, some experiments additionally modify the pretrained CNN by changing its layers [232][233][234][238]. It is also worth noting that it is not necessary that the high accuracy of the same pretrained CNN which used the ImageNet database is transferable to other tasks which use different datasets [236].

### III. Data pre-processing

In this experiment a subset of the hostel image dataset is tested, the dataset consists of **7,350** images of chain hostels which are corporately owned and usually have a similar design, and independent hostels which are usually privately owned and have unique designs. The dataset is divided into two sets, training/validation set and test set. The training/validation set consists of the training set which is used to train the algorithm/model and the validation set which is used to unbiasedly evaluate the algorithm/model while fine-tuning parameters, containing **7,000** images. The test set, containing **350** images, is used to unbiasedly evaluate the final algorithm/model. Images in the training/validation set are labelled and grouped into seven classes, normal bedroom, bunk bedroom, other bedroom, bathroom, kitchen, living room, and laundry room. Since the bedroom is the key feature for hostel accommodation, three out of seven classes are allocated to three types of bedrooms. Furthermore, each class contains **1,000** images equally in the training/validation set and **50** images equally in the test set in order to control this variable which might affect the test performance in this study. Additionally, these images are in different sizes, in three colour channels, and stored in JPEG format. For non-commercial research and educational purposes, this dataset is made available on Google Drive [239] and Hostel image examples of each class are demonstrated in figure 6.3 below.

## CHAPTER 6: HOSTEL IMAGE CLASSIFICATION

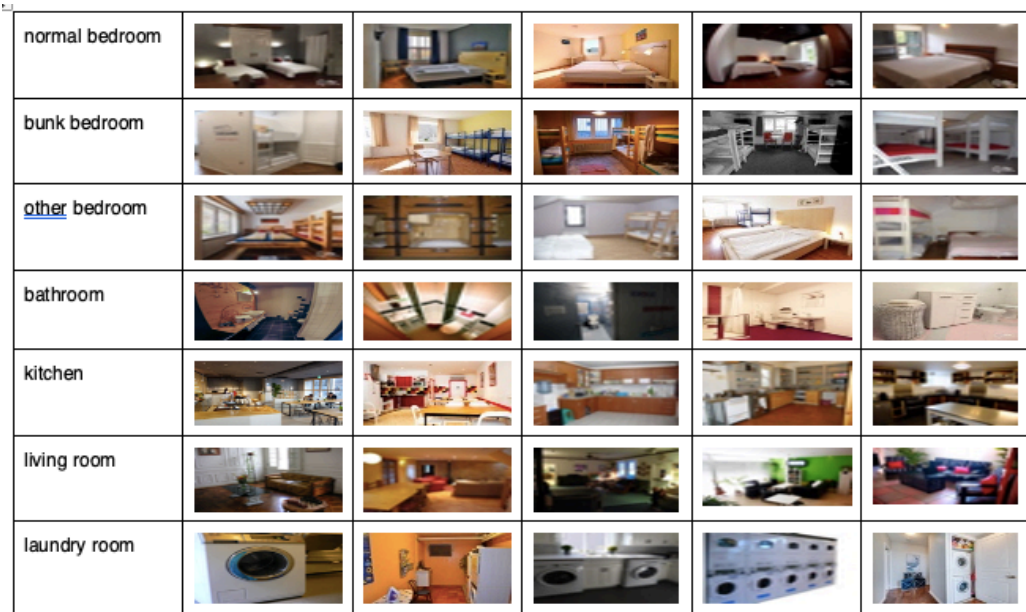


Figure 6.3: Hostel image examples of each class

### IV. Experiments

To test the potential of 11 pretrained CNNs with transfer learning for hostel image classification, this study is conducted in the Google Colaboratory environment, a free cloud service with free GPU, using Pytorch on Intel(R) Core (TM) i7-6700 CPU (Central Processing Unit) @3.40GHz (Gigahertz) machine with 16GB RAM (Random Access Memory) and NVIDIA Tesla K80 GPU with 12GB (Gigabyte) memory. The implemented Pytorch code is an adapted version of Rath's algorithm [240]. The approach is to directly employ 11 pretrained CNNs. Then, transfer learning is implemented by freezing the learned weights and parameters in the early deep layers and fine-tuning some parameters on the latter layers to match the hostel image database collected as shown in figure 6.4. Furthermore, in these new latter layers, all 7,000 images in the training/validation set are randomly rotated between -30 and 30 degrees, centre cropped and resized to 224x224 format as an input format required by the selected pretrained CNNs, and horizontally flipped for the data augmentation process. On top of this, batch normalization which is a supervised learning technique that standardizes the inputs in a neural network layer, weight sharing which is a technique that uses the same numbers within each

## CHAPTER 6: HOSTEL IMAGE CLASSIFICATION

neuron/filter in a particular layer, and dropout which is a technique that ignores randomly selected neurons during training are chosen as the regularization techniques to facilitate these CNN models.

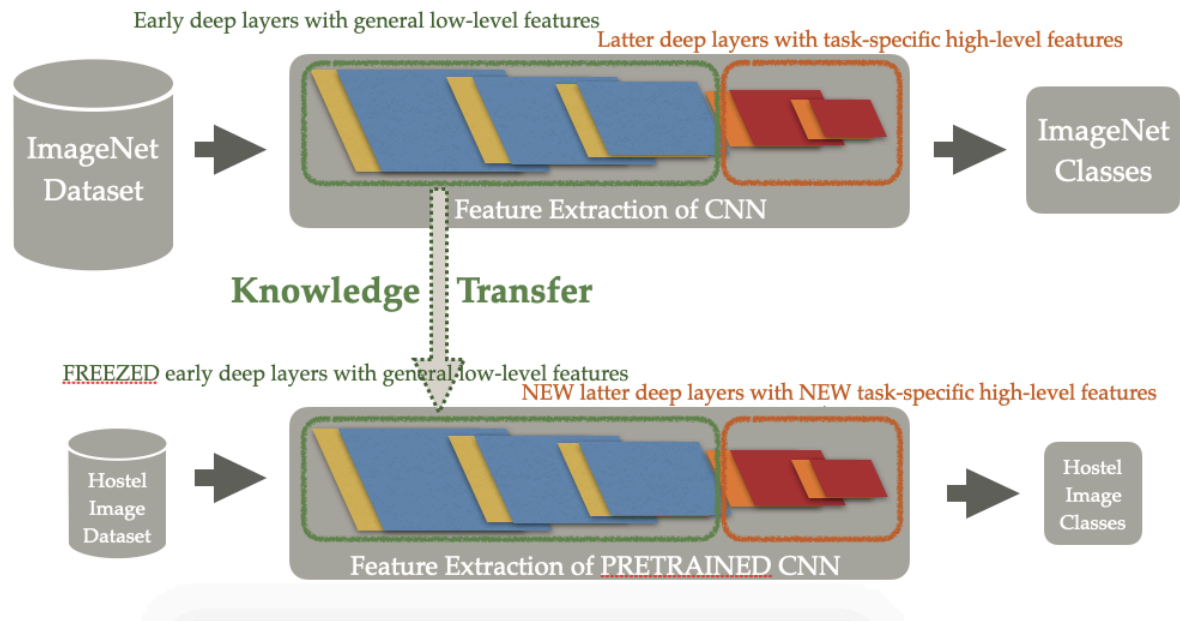


Figure 6.4: The implementation of transfer learning for image classification task

SqueezeNet1\_0 model is selected as a CNN baseline in this study which is divided into seven experiments as follow.

**1st experiment** is to test the baseline performance for five epochs on various proportions of data split in training/validation set (90:10, 80:20, 70:30, 60:40, and 50:50) and its effect on the test set performance.

**2nd experiment** is to test the baseline performance for five epochs on various dropout rates in the training/validation set (0.1, 0.2, 0.3, 0.4, and 0.5) and its effect on the test set performance.

**3rd experiment** is to test 11 pretrained CNNs performance for various epochs at the best-result data split and dropout rate.

**4th experiment** is to test 11 pretrained CNNs performances for 25 epochs at the best-performed data split and dropout rate and its effect on training time.

**5th experiment** is to test 11 pretrained CNNs performance for 25 epochs at the best-result data split and dropout rate and its effect on the test set performance.

**6th experiment** is to test 11 pretrained CNNs performance for 10 epochs at the best-result data split and dropout rate and its effect on the test set performance on the database with original image sizes and the database with resized images (300x300).

**7th experiment** is to analyze the accuracy of the best-performed CNN on the training set and validation set.

### V. Performance measurement

To evaluate the transfer learning performance of pretrained CNN models on hostel image classification task, two measures are implemented as follow.

A. Accuracy is a ratio of the number of correctly classified images to the total number of all images.

$$Accuracy = \frac{\textit{The number of correctly classified images}}{\textit{The total number of all images}} \quad (6.1)$$

B. Training time (minutes)

## 6.2 Results of Proposed Fast Embedding

### I. 1st experiment

Data split (Training set: Validation set)	Accuracy (%)	
	Validation set	Test set
90:10	<b>74.23</b>	70.86
80:20	72.63	69.71
70:30	72.96	70.29
60:40	69.47	67.14
50:50	66.92	<b>71.71</b>

Table VI.II: An accuracy comparison of SqueezeNet1\_0 on validation set & test set at different data split. Bold values indicate the highest accuracy and the optimal option is highlighted.

Looking at an accuracy of SqueezeNet1\_0 (baseline CNN) on the validation set, it can be seen that the data split between the training set and the validation set at 90:10 provides the highest accuracy, 74.23%, whereas the accuracy on the test set demonstrates the highest at 50:50 proportion, 71.71%. Having said this, as one proportion can only be selected to apply in further experiments, the accuracies on the validation set and test set need to be considered together. Despite the data split at 90:10 achieves the best results, the 70:30 proportion could also be considered as it provides the second-best results. Therefore, due to the benefit of the validation set which is to provide an unbiased performance of the training model, in the 2<sup>nd</sup> experiment data split at 90:10 and 70:30 are tested to finalize the most appropriate data proportion for the rest of the experiments. Additionally, it is worth noting that even though a low-cost model like SqueezeNet1\_0 is implemented, the classification performances are reasonable for mobile applications.

## CHAPTER 6: HOSTEL IMAGE CLASSIFICATION

### II. 2nd experiment

Dropout	Accuracy (%) at 90:10 data split		Accuracy (%) at 70:30 data split	
	Validation set	Test set	Validation set	Test set
0.1	71.769	69.376	71.632	70.000
0.2	67.395	61.958	68.966	62.571
0.3	<b>72.140</b>	69.965	<b>72.960</b>	70.286
0.4	69.264	70.011	69.748	70.571
0.5	64.752	<b>71.378</b>	63.378	<b>72.571</b>

Table VI.III: An accuracy comparison of SqueezeNet1\_0 on validation set & test set at different dropouts at 90:10 and 70:30 data split. Bold values indicate the highest accuracy and the optimal option is highlighted.

Considering an accuracy of the baseline CNN on the validation set, at 0.3 dropout rate results in the highest accuracy, 72.140% at 90:10 data split and 72.96% at 70:30 data split, whereas on the test set at 0.5 dropout achieves the highest result, 71.378% at 90:10 data split and 72.57% at 70:30 data split. One of the possible reasons why the accuracies of the test set at 0.5 and 0.4 dropout rates are higher than the accuracies of the validation set is that these dropouts are too heavy meaning there are too many disabled neurons which causes the greater loss of information for the training model resulting in model underfitting. Another possible reason is that SqueezeNet1\_0, the baseline CNN, is a small-size network. With heavier dropouts, the capacity of the network could be reduced more drastically. Furthermore, similar to the 1st experiment on different data splits, it is essential to consider the accuracies on the validation set and test set all together at one dropout rate. Consequently, as shown in Table VI.III, the dropout rate at 0.3 contributes to the highest accuracies of SqueezeNet1\_0 after transfer learning and therefore, the remaining experiments also apply the 0.3 dropout rate. As for the most appropriate data proportion, despite 90:10 data split provides the highest accuracy in the 1st experiment, when considering different dropout 70:30 data split performs better in many cases. Therefore, for the rest of the experiments, 70:30 proportion is applied.



## III. 3rd experiment

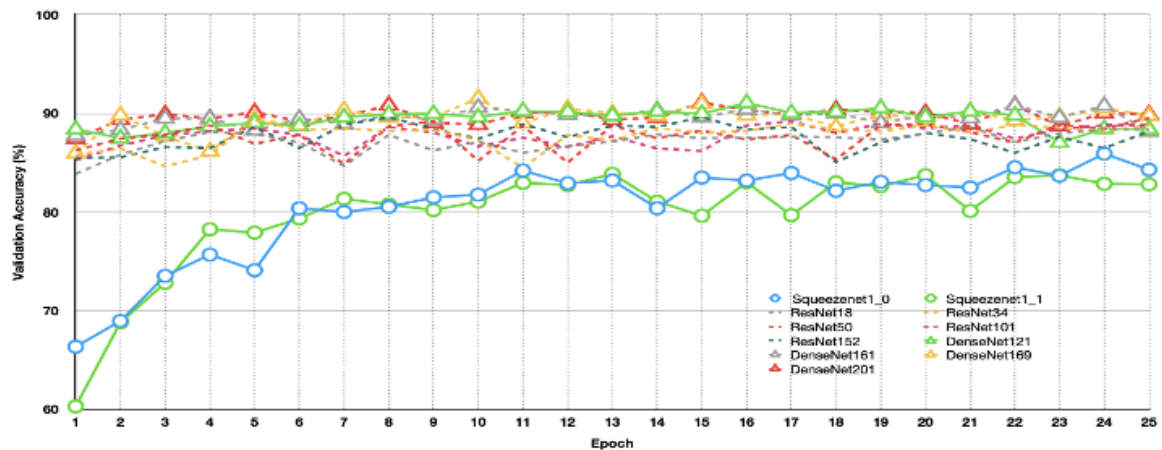


Figure 6.5: An accuracy comparison of 11 pretrained CNNs after transfer learning at different epoch

When testing pretrained CNNs at different epoch, figure 6.5 shows that DenseNet models and ResNet models provide similar and stable performance across each of these models, whereas SqueezeNet models start to perform better at 6 epochs. After this, their accuracy improves slightly and become more stable. Therefore, it can be seen that the CNN models could achieve high accuracies with only a few iterations. Having said this, in order to continue investigating SqueezeNet models at their best performances, including SqueezeNet1\_0 as a baseline CNN, the rest of the experiments are conducted at 25 epochs.

# CHAPTER 6: HOSTEL IMAGE CLASSIFICATION

## IV. 4th experiment

CNN model	Accuracy (%) on validation set	Training time (mins)
<i>SqueezeNet1_0</i>	80.50	18.02
SqueezeNet1_1	79.83	<b>17.00</b>
ResNet18	87.16	17.67
ResNet34	87.73	20.59
ResNet50	87.64	28.18
ResNet101	87.79	39.40
ResNet152	87.58	50.89
DenseNet121	89.39	35.95
DenseNet161	89.56	53.74
DenseNet169	89.44	44.91
DenseNet201	<b>89.72</b>	51.69

Table VI.IV: An accuracy and training time comparison of 11 pretrained CNNs after transfer learning. Bold values indicate the best performance and the optimal options are highlighted.

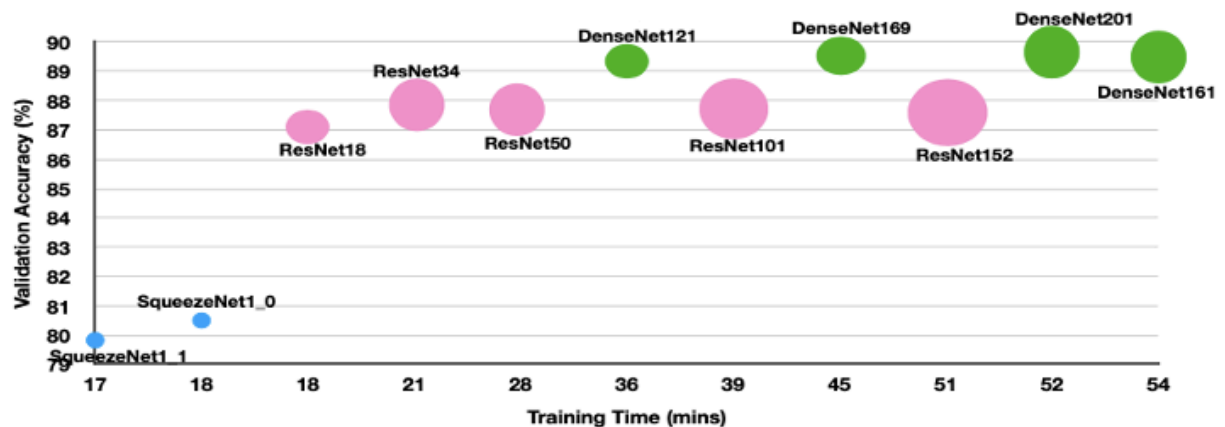


Figure 6.6: An accuracy and training time comparison of 11 pretrained CNNs after transfer learning

As demonstrated in Table VI.IV, it can be seen that DenseNet models, especially DenseNet201, outperform the other CNN models after applying the transfer learning technique. Nonetheless, if considering only training time, SqueezeNet models spend less computational time due to their smaller model sizes which are illustrated in smaller circles. Having said this, when taking both accuracy and training time into

## CHAPTER 6: HOSTEL IMAGE CLASSIFICATION

account, it seems that DenseNet121 is one of the most potential models for hostel image dataset. Therefore, in further experiments, DenseNet121 and DenseNet201 are in focus.

## CHAPTER 6: HOSTEL IMAGE CLASSIFICATION

### V. 5th experiment

CNN model	Accuracy (%)	
	Validation set	Test set
<i>SqueezeNet1_0</i>	<i>80.50</i>	<i>81.43</i>
SqueezeNet1_1	79.83	82.00
ResNet18	87.16	<b>88.57</b>
ResNet34	87.73	88.29
ResNet50	87.54	83.43
ResNet101	87.72	86.86
ResNet152	87.58	83.14
DenseNet121	89.39	83.71
DenseNet161	89.56	83.43
DenseNet169	89.43	85.71
DenseNet201	<b>89.72</b>	86.86

Table VI.V: An accuracy comparison of 11 pretrained CNNs after transfer learning on validation set & test set. Bold values indicate the highest accuracy and the optimal options are highlighted.

It can be seen in Table VI.V that in most CNNs simulations the accuracies on the validation set are higher than the accuracies on the test set. However, four models, SqueezeNet1\_0, SqueezeNet1\_1, ResNet18, and ResNet34, have the opposite results. It could be assumed that the lesser deep layers, the more it becomes overfitting. Nonetheless, the accuracies of DenseNet121 and DenseNet201 on both validation set and test set are still considerably high, 89.39% and 83.71% for DenseNet121 and 89.72% and 86.86% for DenseNet201, which reassure their potential for hostel image classification with transfer learning.

# CHAPTER 6: HOSTEL IMAGE CLASSIFICATION

## VI. 6th experiment

CNN model	Original size dataset			Resized dataset		
	Accuracy (%)		Training time (mins)	Accuracy (%)		Training time (mins)
	Validation set	Test set		Validation set	Test set	
<i>SqueezeNet1_0</i>	<i>76.26</i>	<i>81.43</i>	<i>7.21</i>	<i>71.20</i>	<i>77.14</i>	<i>4.00</i>
SqueezeNet1_1	76.06	82.00	<b>6.80</b>	75.35	77.14	<b>3.73</b>
ResNet18	86.56	<b>88.57</b>	7.07	87.17	86.57	4.01
ResNet34	87.24	88.29	8.24	87.73	87.14	4.86
ResNet50	87.28	83.43	11.27	87.34	86.86	8.21
ResNet101	87.31	86.86	15.76	87.07	81.14	12.12
ResNet152	87.34	83.14	20.36	87.48	84.86	16.14
DenseNet121	88.93	83.71	14.38	88.56	<b>88.57</b>	9.95
DenseNet161	89.15	83.43	21.50	<b>90.05</b>	87.43	17.14
DenseNet169	88.88	85.71	17.97	89.42	85.71	12.51
DenseNet201	<b>89.38</b>	86.86	20.68	89.30	86.29	15.18

Table VI.VI: An accuracy comparison of 11 pretrained CNNs on original size database & resized database at 10 epochs. Bold values indicate the best performance and the optimal options are highlighted.

Due to different image data sources, gathered hostel images come in various image sizes. This 6th experiment tested another hostel image dataset, which is the same dataset in the previous experiments but reduced its image sizes to one image size (300x300) prior and in addition to data augmentation in the data pre-processing stage. Table VI.VI shows that accuracies on the validation set are not affected much by resizing the dataset and remain similar in most CNN cases even with fewer iterations at 10 epochs. Having said this, accuracies on the test set improve slightly in some CNN models, including DenseNet121 which is increased from 83.71% to 88.57%. Furthermore, the computational time is significantly decreased on the resized dataset for all CNN models.

## VII. 7th experiment

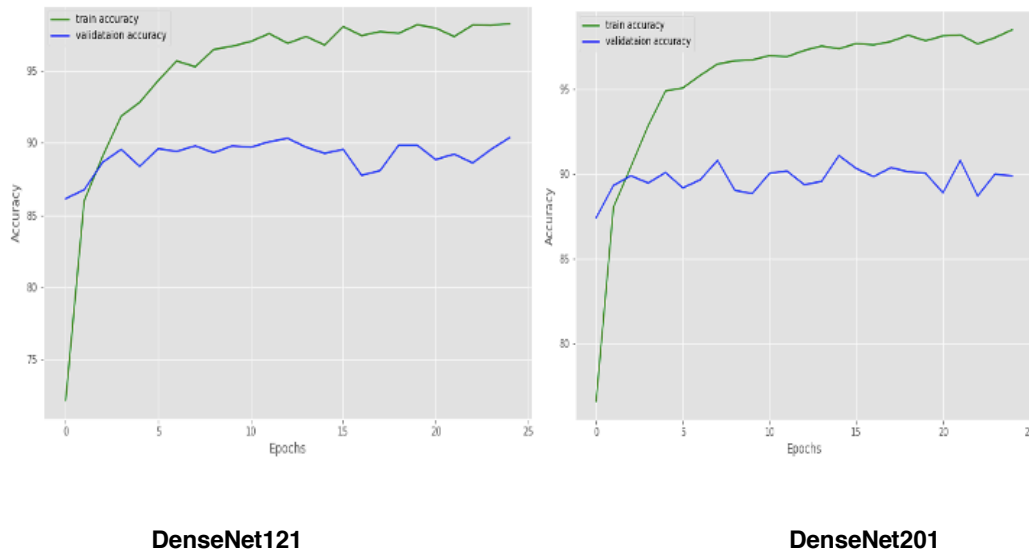


Figure 6.7: An accuracy comparison of DenseNet121 & DenseNet201 on training set & validation set

Figure 6.7 illustrates the plots of the accuracy of DenseNet121 and DenseNet201 on both the training set, green colour, and validation set, blue colour. It is clear that validation accuracies remain stable at around 90% for both models. That means the models perform well and are not overfitting the current dataset size. Therefore, the dataset size could be increased in order to potentially increase the accuracy of hostel image classification.

### VIII. Conclusion

The results from seven experiments highlight that, with the optimization of parameter fine-tuning, various pretrained CNNs with transfer learning can provide good performance. Particularly, DenseNet121 and DenseNet201 outperform the other CNNs, in terms of accuracy rate and training time, and achieve impressive outcomes for hostel image classification. Different elements have been explored in order to find the optimal combination and achieve the highest classification performance. SqueezeNet1\_0 is selected as a baseline CNN due to its most compact model size, the smallest number of deep layers, and a smaller number of parameters. In terms of the data split during model training, it shows that the training set and the validation set should be split into 70% and 30%, respectively. The dropout rate at 0.3 contributes to the highest accuracy during training and relatively high during testing. When all 11 pretrained CNNs are considered, the classification accuracy can reach its peak with only a few iterations and become stable for DenseNet models and ResNet models, whereas SqueezeNet models reach 80% accuracy at six epochs and continue to improve slightly. DenseNet201 best performs during training and is relatively similar during testing, while DenseNet121 demonstrates competitive results. Furthermore, when the size of the hostel image dataset is reduced, in addition to the data augmentation, the computational time is significantly decreased, even though the classification accuracies only slightly increase. Therefore, by applying this state-of-the-art transfer learning technique in pretrained CNNs for image classification with the optimal settings addressed in this study, hostel image managers/e-intermediaries could automatically organize their growing hostel image collections, fasten the image indexing task with high accuracy, less labour cost, and less human errors on this accommodation type which has a clutter of content on an image, and become better user-friendly to the potential customers offering improved services/products by utilizing the correctly classified images in a database and providing more targeted solutions to their current or potential customers which could lead to a better position to recover from the current global economic downturn.

### 6.3 Summary

A subset of the collected hostel image dataset is tested on the image classification task. The 7,350 images of interior hostel images are labelled and grouped into seven classes which are normal bedroom, bunk bedroom, other bedroom, bathroom, kitchen, living room, and laundry room. In this simulation, seven experiments are conducted aiming to find the optimal setting of parameter fine-tuning using various pretrained CNN networks. The results reveal that DenseNet201 model outperforms the other CNNs, in terms of accuracy rate and training time, and achieve impressive outcomes with 70% of images allocated to the training set and the remaining 30% to the validation set for the model training process, as well as 0.3 dropout rate. Furthermore, only a few iterations are needed in order to reach the peak in classification accuracy and become stable. Nonetheless, DenseNet121 demonstrates competitive results. On top of these, when the size of the hostel image dataset is reduced by centre-cropping each original image to 300x300 pixels prior to the beginning of the simulation, in addition to the data augmentation, the computational time is significantly decreased, even though the classification accuracies only slightly increase. Therefore, by applying the state-of-the-art transfer learning technique in pretrained CNNs for image classification with the optimal settings addressed in this study, hostel image collections could be automatically organized and the image indexing task could be fastened with high accuracy, less labour cost, and less human errors.



## Chapter 7: Hostel Image Retrieval

### 7.1 Experiment settings

**(This manuscript, Instance-level hostel image retrieval with unsupervised learning, is submitted to the Journal of Hospitality and Tourism Technology and is currently in a peer-reviewed process)**

#### I. Artificial Intelligence-based image features for CBIR

Traditionally, researchers consider classical features such as colours, shapes, textures, spatial positions or Scale-Invariant Feature Transform (SIFT) of an image as its representation. Then, these features of an image will be compared to the features of another image in order to compare their similarity in Content-Based Image Retrieval (CBIR) or classification task. Nonetheless, in recent decades artificial intelligence-based image features have been in the spotlight as Artificial Neural Network (ANN), especially Convolutional Neural Network (CNN) could simulate human cognition process on a deeper level with less labour cost, less time consuming and less human errors. A CNN model is a combination of convolutional layers which traverse input matrices with convolutional kernels to learn parameters during the low-level feature extraction process, pooling layers that subsample the output from convolutional layers in order to learn parameters in the high-level feature extraction process, and fully connected layers which generate numerical output for similarity comparison between images in the next stage. Different CNN model tends to contain a different number of these deep layers and layer type. However, in an image retrieval task, the CNN architectures can typically be placed in a CBIR process as illustrated in figure 7.1 below.

## CHAPTER 7: HOSTEL IMAGE RETRIEVAL

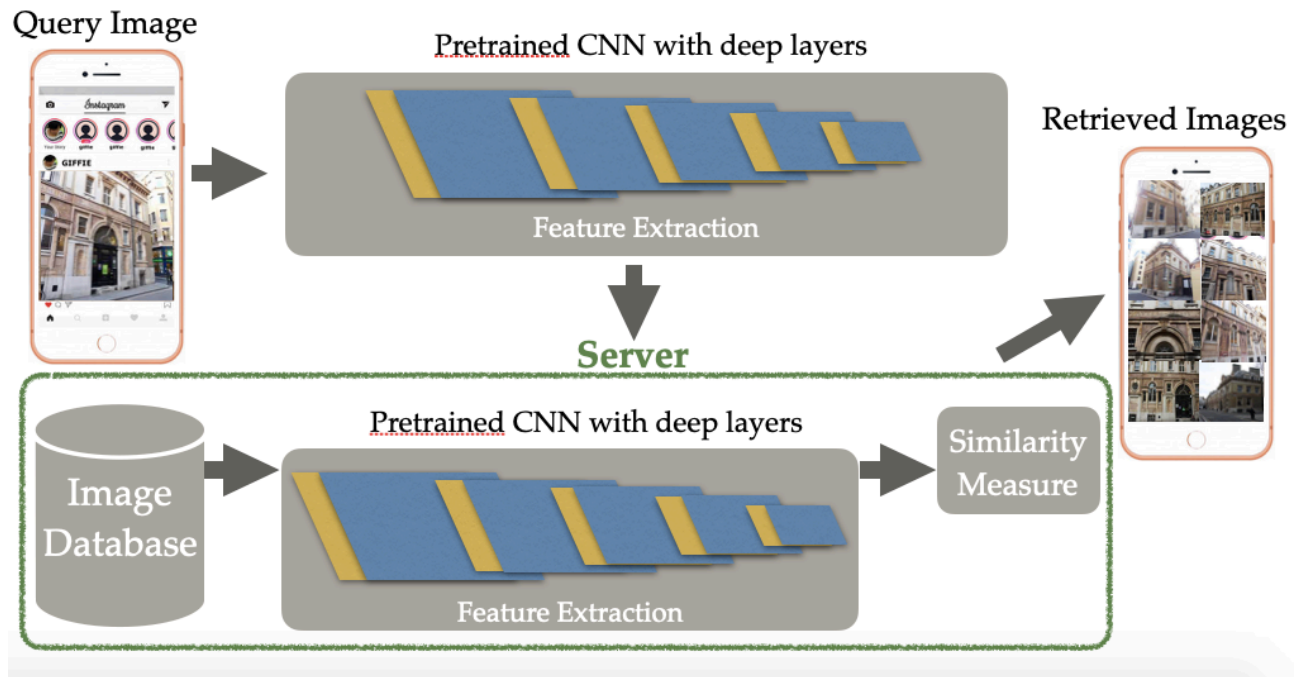


Figure 7.1: CNN model architectures in a CBIR process

However, these deep learning features of digital images contain high-dimensional data which require huge storage and high computational cost with security constraints. Additionally, a large-scale image database for CNN model training is one of the critical factors that could lead to a high accuracy retrieval result as this deep network could create a more detailed image representation. To overcome these hindrances in the task using a much smaller scale image dataset, transfer learning technique is introduced by applying available CNNs which are already pretrained in image classification task with ImageNet dataset, containing over 14 million images, transferring the learned low-level features which are general features in any image, fine-tuning the high-level features to match the new dataset, and training this adapted CNN for the new task. On top of this, amending/changing the latter layer(s) of pretrained CNN and applying data augmentation such as horizontal/vertical flipping, rotation, and resizing images to provide more information to the adapted CNN via this artificial increase of training images could result in higher accuracy in the new retrieval task. Consequently, with these artificial intelligence-based image features from pretrained CNNs, experiments on small scale image datasets such as hostel image collections could be more feasible with less computational resources required when

compared to training a CNN model from scratch and the high accuracy could also be more achievable without overfitting.

### II. Chosen pretrained CNN models

Though a variety of pretrained CNNs are available, the robustness of ResNet models in several experiments makes them appealing as these models provide optimized solutions to the degradation problem, which tends to be caused by an increase of deep layers, by letting the stacked layers fit a residual mapping instead of the underlying mapping. Moreover, the considerably high number of their deep layers and parameters which could provide satisfying accuracies also contribute to ResNet models being chosen for this study. In addition, their model sizes are considerably small which demonstrate the potential for mobile application or small-scale search engine and they require less computational resources while providing the feasible relative prediction time using GPU for this work when compared to DenseNet models which require heavier GPU memory consumption and longer training time [255][256]. Even though DenseNet models slightly outperform ResNet models in the previous chapter, their high consumption of computational resources leads to ResNet models being selected over DenseNet models for this hostel image retrieval experiment.

Therefore, ResNet50, ResNet101, and ResNet152 are applied in the following experiments. On top of this, to compare with a larger model with a higher number of parameters but fewer deep layers than ResNet models, the VGG16 model is also included in the experiments. Furthermore, ResNet and VGG models are popular for object recognition tasks as they discard the fully connected layers which provide initialization when fine-tuning. The information about the chosen CNN models is shown in Table VII.1 as follow.

## CHAPTER 7: HOSTEL IMAGE RETRIEVAL

CNN model	Model size	No. of deep layer	No. of parameter
ResNet50 [203]	96MB	50	25.6 million
ResNet101 [203]	167MB	101	44.6 million
ResNet152 [203]	230MB	152	58.2 million
VGG16 [47]	515MB	16	138 million

Table VII.I: Chosen CNN models information

### III. Instance-level image retrieval with unsupervised learning

To begin with the concept of Instance-level image retrieval, it is a search task in a large database for images that represent the same object instance as in a query image. An object instance differs from an object class as the instance is considered an element of a class. For example, the hostel is an object class but YHA London St. Pancras is an instance of that hostel. Therefore, the instance-level recognition task tends to be more challenging than the class-level recognition one. The process of this type of image retrieval consists of two steps, identifying similar regions of images in the database and the query image and refining the matching results. The first step compares the global image descriptors representing each image with a single vector and this compact image representation contains up to ten thousand dimensions which could be more feasible in large-scale search. Then, the domain-specific refinement is performed by patch-level matching the candidate images using local descriptors for higher precision. The classic geometric verification [241][242] and aggregated selective match kernels (ASMK) techniques [243][244] are widely used in the step, as well as the nearest neighbour.

As for the unsupervised learning concept, it is the use of a machine-learning algorithm to train a model without any human intervention using an unlabeled dataset in order to analyze and discover patterns within the dataset, whereas the supervised learning model relies on a labelled dataset to identify the data patterns.

### IV. Data pre-processing

In this simulation both London hostel building datasets, Hostels-900 and Hostels-2K, are tested with the application of data augmentation using rotation and resizing the centre-cropped images to 224 x 244 pixels as required by the models for an appropriate image input size. Furthermore, regularization techniques such as weight sharing, batch normalization, dropout, early stopping, noise injection, and parameter norm penalties could help to facilitate CNN models. Consequently, weight sharing which is a technique that uses the same numbers within each neuron/filter in a particular layer, batch normalization which helps to standardize the inputs in a neural network layer, dropout which is a technique to randomly discard a portion of neurons to avoid a model from overfitting, and noise injection using descriptor whitening to reduce the high correlation of feature values are applied in this simulation accordingly.

### V. Data representation

In machine learning, feature vectors are n-dimensional vectors that represent the numerical characteristics/features of objects in an image. Therefore, the collected hostel image data are also mapped into feature vectors. Additionally, it can be seen that the relevant literature aforementioned only focus on floating-coefficient feature vectors to represent their numeric data. This means the coefficients are in floating-point numbers which have a decimal place in order to formulaically represent the real numbers in a more precise manner, as well as fasten the computing process. Having said that, the coefficients of feature vectors could be in binary, particularly for retrieval task which considers relevant and irrelevant items. Binary coefficients reduce the multiplier cost as the number of nonzero bits in the coefficients is minimized. As a result, the computational cost of binary-coefficient feature vectors is less than the floating-coefficient feature vectors as each feature vector has a coefficient of 1 bit instead of 32 bits. Consequently, this study investigates both floating-coefficient feature vectors in their default value and binary-coefficient feature vectors for hostel data representation which could lead to an interesting discovery.

### VI. Experiments

Due to the resource limitation during this study, the experiments are conducted in two environments. The Google Colaboratory environment, a free cloud service with free graphics processing unit (GPU) using Pytorch on Intel(R) Core (TM) i7–6700 CPU @3.40GHz machine with 16GB RAM and NVIDIA Tesla K80 GPU with 12GB memory, is used in the feature extraction process. The MATLAB Online environment, an online platform to access the latest MATLAB version with online sharing/publishing features and Matlab drive synchronization for a free 5GB cloud storage using MathWorks hosted computing resources and storage, is used in the unsupervised learning image retrieval process. The implemented code is an adapted version of Radenovic, Tolias, and Chum's algorithm [174] using whitened and re-normalized Generalized-Mean (GeM) descriptor and Structure-from-Motion (SfM) information for training data, as well as the Siamese learning. The approach is to employ four off-the-shelf pretrained CNNs. Next, the transfer learning technique is adopted by freezing the learned weights and parameters in the early deep layers, as well as freezing the GeM pooling layer with SfM and Siamese learning of Radenovic, Tolias, and Chum's code. Then, fine-tuning some parameters on the latter layers to match the collected London hostel building datasets as illustrated in figure 7.2 below. Furthermore, at test time the input image is represented on three different scales, original scale (1), up-sampled scale ( $2^{1/2}$ ), and down-sampled scale ( $1/2^{1/2}$ ) for multi-scale evaluation.

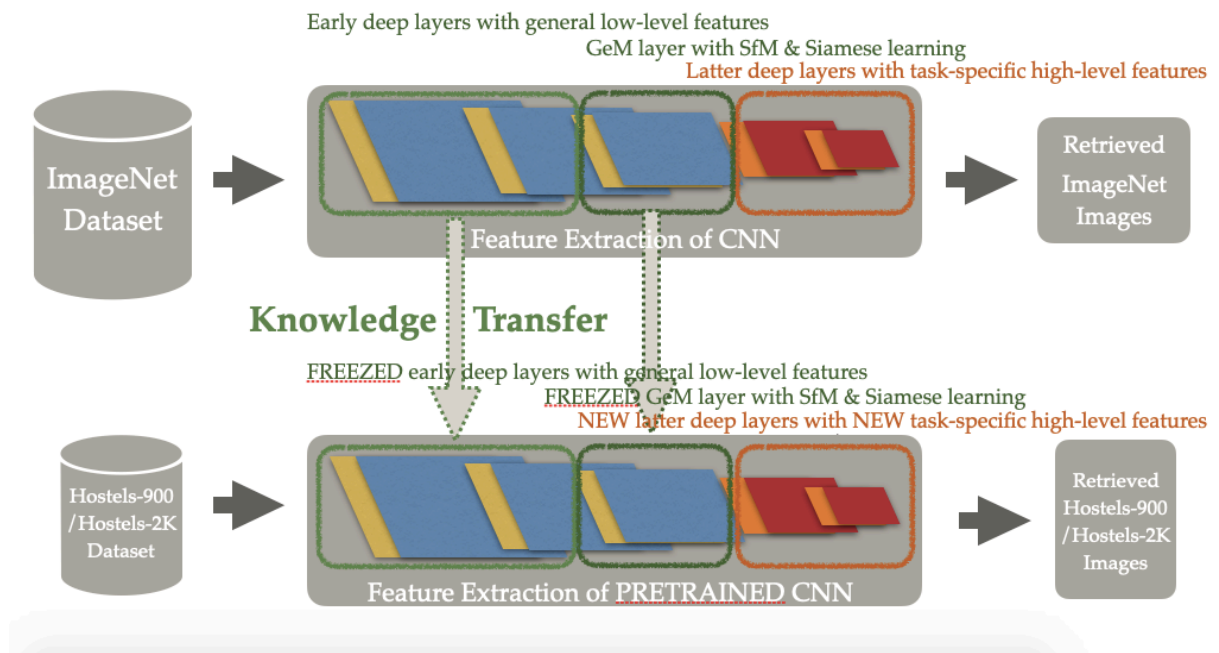


Figure 7.2: The implementation of transfer learning for image retrieval task

Three experiments are conducted as follow.

**1<sup>st</sup> experiment** is to test the robustness of this work's retrieval systems on four datasets with different data sizes, Hostels-900 (972 images), Hostels-2K (2,380 images), and Hostels-900 with extra distracting 100K images (100,972 images) in the test dataset and Hostels-2K with extra distracting 100K images (102,380 images) in test dataset for more extensive evaluation.

**2<sup>nd</sup> experiment** is to further test the robustness of the four pretrained CNN-based retrieval systems on the same four datasets when using the default value (float coefficients with 32 bits) and the binary value (binary coefficients with 1 bit) of feature vectors.

**3<sup>rd</sup> experiment** is to test the best-performing system by retrieving the most dissimilar images and to identify the query images which give the worst and the 2<sup>nd</sup> worst results when using the default value (float coefficients with 32 bits) and the binary value (binary coefficients with 1 bit) of feature vectors.

## VII. Performance measurement

To evaluate the performance of retrieval systems on London hostel building datasets, two measures are implemented as follow.

### A. mean Average Precision (mAP)

Mean Average Precision is a widely used metric in measuring the accuracy of a retrieval system on high dimensional data with many classes or categories. The mean Average Precision can be calculated from Precision with the following formula.

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (7.1)$$

where  $Precision = \frac{True\ Positive}{Total\ Positive\ results}$  and  $AP_i$  is Average Precision.

### B. Precision@k

Another commonly seen measure is Precision@k. It is used in binary problems for recommendation systems when considering relevant and irrelevant items. Firstly,  $k$  is chosen as a number of recommended or retrieved items. Then, the Precision@k which is the proportion of relevant item(s) in the retrieved item(s) is calculated as shown in the formula below.

$$Precision@k = \frac{\text{number of retrieved items@k that are relevant}}{\text{number of retrieved items@k}} \quad (7.2)$$



# CHAPTER 7: HOSTEL IMAGE RETRIEVAL

## 7.2 Results of Proposed Fast Embedding

### I. 1st experiment

Networks	Measures	Hostels-900	Hostels-900 + 100K	Hostels-2K	Hostels-2K + 100K	Oxford5K	Oxford105K	Paris6K	Paris106K
ResNet50	mAP	58	54	61	58	-	-	-	-
	Precision@1	99	98	<b>100</b>	99	-	-	-	-
	Precision@5	95	94	97	97	-	-	-	-
	Precision@10	88	86	96	95	-	-	-	-
ResNet101	mAP (proposed)	56	55	59	59	-	-	-	-
	mAP [174]	-	-	-	-	87.8	84.6	92.7	86.9
	Precision@1	97	97	98	98	-	-	-	-
	Precision@5	93	93	97	97	-	-	-	-
	Precision@10	86	86	94	94	-	-	-	-
ResNet152	mAP	<b>63</b>	<b>59</b>	<b>66</b>	<b>64</b>	-	-	-	-
	Precision@1	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>	-	-	-	-
	Precision@5	<b>97</b>	<b>96</b>	<b>99</b>	<b>99</b>	-	-	-	-
	Precision@10	<b>91</b>	<b>90</b>	<b>97</b>	<b>97</b>	-	-	-	-
VGG16	mAP (proposed)	53	46	56	51	-	-	-	-
	mAP [174]	-	-	-	-	87.9	83.3	87.7	81.3
	Precision@1	99	99	98	98	-	-	-	-
	Precision@5	92	90	97	96	-	-	-	-
	Precision@10	84	82	93	92	-	-	-	-

Table VII.II: An accuracy comparison of four pretrained CNN-based retrieval systems on four datasets. Bold values indicate the highest accuracies, and the best-performing system is highlighted.

Additional comparison to Radenovic, Tolias, and Chum's results are provided.

Looking at the accuracies of four pretrained CNN-based retrieval systems, it can be seen that ResNet152 stunningly outperforms the other three networks in all cases. ResNet152 is able to achieve the highest accuracies on various-size datasets. When using mAP measure, ResNet152 reaches 63% accuracy on the smallest dataset, Hostels-900 with only 972 images, and slightly downgrades to 59% even with 100K images added to the system for extensive evaluation. Similarly, testing on the bigger-size dataset, Hostels-2K with 2,380 images, ResNet152 improves its accuracy to 66%

and 64% with extra distracting 100K images in the test dataset. Precision@k is further applied to measure the systems. Three numbers of retrieval items are chosen,  $k=1$ ,  $k=5$ , and  $k=10$ . In the case of Precision@1, 100% accuracy is achieved by ResNet152 on all datasets, whereas 96%-99% accuracy for Precision@5 and 90%-97% accuracy for Precision@10. Moreover, ResNet50 is also able to reach 100% accuracy on Hostels-2K when measured with Precision@1. Additionally, despite ResNet50, ResNet101, and VGG16 provide lower accuracies, they can still give a good range of accuracies between 46%-61%, 97%-100%, 90%-97%, and 82%-96% for mAP, Precision@1, Precision@5, and Precision@10, respectively, which demonstrate the robustness of pretrained CNN-based retrieval systems with unsupervised learning on small scale hostel datasets. Furthermore, when comparing to the original code on ResNet101 and VGG16 models, it can be seen that despite our mAP results on hostel datasets are lower than Radenovic, Tolias, and Chum's results on Oxford and Paris datasets at 55%-59% to 84.6%-92.7% and 46%-56% to 81.3%-87.9% on ResNet101 and VGG16, respectively, the level of our accuracy is acceptable considering hostel datasets are smaller and have different image features.

# CHAPTER 7: HOSTEL IMAGE RETRIEVAL

## II. 2nd experiment

Networks	Value of Feature vectors	Measures	Hostels-900	Hostels-900 + 100K	Hostels-2K	Hostels-2K + 100K
ResNet50	Default	mAP	58	54	61	58
		Precision@1	99	98	<b>100</b>	99
		Precision@5	95	94	97	97
		Precision@10	88	86	96	95
	Binary	mAP	57	52	60	57
		Precision@1	98	98	<b>100</b>	99
		Precision@5	94	93	98	97
		Precision@10	87	85	96	95
ResNet101	Default	mAP	56	55	59	59
		Precision@1	97	97	98	98
		Precision@5	93	93	97	97
		Precision@10	86	86	94	94
	Binary	mAP	54	51	56	55
		Precision@1	97	97	98	98
		Precision@5	93	93	97	97
		Precision@10	85	84	93	93
ResNet152	Default	mAP	<b>63</b>	<b>59</b>	<b>66</b>	<b>64</b>
		Precision@1	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>
		Precision@5	<b>97</b>	<b>96</b>	<b>99</b>	<b>99</b>
		Precision@10	<b>91</b>	<b>90</b>	<b>97</b>	<b>97</b>
	Binary	mAP	<b>62</b>	<b>57</b>	<b>65</b>	<b>62</b>
		Precision@1	<b>100</b>	<b>100</b>	<b>100</b>	<b>100</b>
		Precision@5	<b>96</b>	<b>95</b>	<b>99</b>	<b>98</b>
		Precision@10	<b>90</b>	<b>89</b>	<b>97</b>	<b>96</b>
VGG16	Default	mAP	53	46	56	51
		Precision@1	99	99	98	98
		Precision@5	92	90	97	96
		Precision@10	84	82	93	92
	Binary	mAP	48	39	50	44
		Precision@1	97	97	97	97
		Precision@5	89	86	95	94
		Precision@10	80	76	90	88

Table VII.III: An accuracy comparison of four pretrained CNN-based retrieval systems when using the default value and the binary value of feature vectors.

Bold values indicate the highest accuracies, and the best-performing system is highlighted.

Considering the potential of pretrained CNN-based retrieval systems on a mobile application or small-scale search engine, the high dimensional hostel image data is embedded into the binary value, having 1 bit-binary coefficients instead of 32 bit-float coefficients, in order to reduce data transmission rate, meaning the volume of transmitted data within a unit of time, memory, and computational cost required. The 1<sup>st</sup> experiment is extended and the results of feature vectors in the default value and the binary value are compared. Table VII.III highlights that ResNet152 remains the best network among all networks when using the binary value of feature vectors. Furthermore, these binary feature vectors can achieve similar results with little degradation to those in default values on all-size hostel datasets, while requiring much less computational cost. When using mAP measure, with binary value 62 %, 57%, 65%, and 62% accuracy are achieved, whereas 63%, 59%, 66%, and 64% accuracy with default value, on Hostels-900, Hostels-900 +100K, Hostels-2K, and Hostels-2K +100K datasets. As for Precision@1, binary feature vectors provide 100% accuracy like the default feature vectors on all datasets. Similar performance shows 95%-99% accuracy and 89%-97% accuracy achieved for Precision@5 and Precision@10. In addition, like in default value, ResNet50 still achieve 100% accuracy for Precision@1 on Hostels-2K. On top of this, ResNet50, ResNet101, and VGG16 are close in robust performance between the binary and default values on all hostel datasets. This is an interesting discovery of hostel image representation using binary-coefficient feature vectors.

# CHAPTER 7: HOSTEL IMAGE RETRIEVAL

## III. 3rd experiment

Cases	Measures	Hostels-900	Hostels-2K
Best results	mAP	63	66
	Precision@1	100	100
	Precision@5	97	99
	Precision@10	91	97
Worst results	mAP	63	66
	Precision@1	100	100
	Precision@5	97	99
	Precision@10	91	97

Table VII.IV: An accuracy comparison of ResNet152 when retrieving the best results and the worst results on Hostels-900 and Hostels-2K datasets.

Cases	Value of Feature vectors	Measure / Query image	Hostels-900	Hostels-2K
Worst results	Default	Precision@15	13.33	13.33
		Query image	St. Christopher's Village_278	St. Christopher's Village_278
	Binary	Precision@15	20	20
		Query image	The Walrus Hostel_3978362	St. Christopher's Village_278
2nd worst results	Default	Precision@15	40	73.33
		Query image	The Walrus Hostel_24	The Walrus Hostel_24
	Binary	Precision@15	20	73.33
		Query image	St. Christopher's Village_278	The Walrus Hostel_24

Table VII.V: An accuracy comparison of ResNet152 when retrieving the worst results and the 2<sup>nd</sup> worst results on Hostels-900 and Hostels-2K datasets using default value and binary value of feature vectors.

# CHAPTER 7: HOSTEL IMAGE RETRIEVAL

Query image	Retrieved images					
						
St. Christopher's Village_278						
						
The Walrus Hostel_24						
						
The Walrus Hostel_3978362						

Figure 7.3: Image examples of the worst results retrieved

Since ResNet152 outperforms other networks on the 1<sup>st</sup> and the 2<sup>nd</sup> experiments, in this experiment the system is further tested by retrieving the most dissimilar images instead and identifying the query images which give the worst and the 2<sup>nd</sup> worst results when using the default value (32 bit-float coefficients) and the binary value (1 bit-binary coefficients) of feature vectors. According to Table VI.IV, it can be seen that the overall accuracy remains the same for both measures when retrieving the best and the worst results on both datasets. This means the system is reliable. Next, the query image which gives the lowest accuracy is sought. Table VII.V highlights that when using default feature vectors St. Christopher's Village\_278 image provides the lowest accuracy of 13.33% for Precision@15 on both datasets, whereas when using binary feature vectors, the worst results with 20% accuracy are found, The Walrus Hostel\_3978362 image for Hostels-900 and St. Christopher's Village\_278 image for Hostels-2K. Looking at the 2<sup>nd</sup> worst results, the accuracies are increased in most cases. The Walrus Hostel\_24 image gives the 2<sup>nd</sup> lowest accuracy of 40% for Hostels-900 and 73.33% for Hostels-2K when considering default value, whereas 20% accuracy reached by St. Christopher's Village\_278 image on Hostels-900 and 73.33%

accuracy achieved by Walrus Hostel\_24 image when using binary value. It is worth noting that the increase of dataset size contributes to the increase of performance accuracy. Furthermore, the same query images are identified for giving the lowest accuracies in most cases, meaning the retrieval system is robust and reliable.

#### IV. Conclusion

The results from the 1<sup>st</sup> experiment which is to test four retrieval systems constructed with different renowned pretrained CNN models, ResNet50, ResNet101, ResNet152, and VGG16 which then fine-tuning and unsupervised learning are adopted, on four hostel datasets with different dataset sizes demonstrate that the ResNet152-based retrieval system impressively outperforms the other three networks in all cases. When using mAP measure, ResNet152 reaches 63% accuracy on the smallest dataset, Hostels-900, and slightly downgrades to 59% when 100K images are added to the system. ResNet152 improves its accuracy to 66% and 64% on Hostels-2K and Hostels-2K + 100K, respectively. Furthermore, 100% accuracy is achieved on all datasets by using Precision@1 measure, whereas 96%-99% accuracy for Precision@5 and 90%-97% accuracy for Precision@10. Additionally, when comparing to the results of the original code, even though our mAP results on hostel datasets are lower than Radenovic, Tolias, and Chum's results on Oxford and Paris datasets at 55%-59% to 84.6%-92.7% and 46%-56% to 81.3%-87.9% on ResNet101 and VGG16, respectively, our systems provide acceptable accuracies as the trained datasets are smaller and have different image features. As for the 2<sup>nd</sup> experiment which is to further test the robustness of the four pretrained CNN-based retrieval systems on the same four datasets when using the default value (32 bit-float coefficients) and the binary value (1 bit-binary coefficients) of feature vectors, its results emphasize that ResNet152 is the best-performing system even when using binary feature vectors and they can provide similar results with little degradation to those in default values on all-size hostel datasets, while requiring much less computational cost, with binary value 57%-65% accuracy achieved when compared to 59%-66% accuracy of default feature vectors. Additionally, when using Precision@1 measure, these binary feature vectors are able to provide 100% accuracy like the

default feature vectors on all datasets. This means the ResNet152-based retrieval system is robust and has potential for a mobile application or small-scale search engine which has greater memory, bandwidth, and power constraints. On top of this, despite ResNet50, ResNet101, and VGG16 provide lower accuracies, they can still give a good range of accuracies in both experiments. Lastly, the 3<sup>rd</sup> experiment which is to test the best-performing system, ResNet152, by retrieving the most dissimilar images instead, as well as identifying the query images which give the worst and the 2<sup>nd</sup> worst results when using default value and the binary value feature vectors. The results illustrate that the overall accuracy remains the same when retrieving the best and the worst results. As for the query image which gives the lowest accuracy for default and binary feature vectors, the same image is identified in many cases, St. Christopher's Village\_278 image with an accuracy of 13.33%-20% for Precision@15. When searching for the 2<sup>nd</sup> worst results, the accuracies are increased in most cases, between 20%-73.33% for Precision@15, and The Walrus Hostel\_24 image is frequently identified as the query image which gives the 2<sup>nd</sup> lowest accuracy. To summarize, the results from these experiments highlight the robustness and reliability of the ResNet152-based instance-level retrieval system on small scale hostel datasets with unsupervised learning.



### 7.3 Summary

The Hostels-900 and Hostels-2K datasets are tested on instance-level image retrieval task. This simulation consists of three experiments for the purpose of testing four retrieval systems constructed with renowned pretrained CNN models, ResNet50, ResNet101, ResNet152, and VGG16 which then fine-tuning and unsupervised learning are adopted. The results highlight that the ResNet152-based retrieval system impressively outperforms the other three networks in all cases even with extra 100K images added to the system for an extensive evaluation. Moreover, when comparing the retrieval results between binary feature vectors with 1 bit-binary coefficients and default feature vectors with 32 bit-float coefficients, ResNet152 remains the best-performing system and the binary feature vectors are able to provide similar results with little degradation to those in default values on all-size hostel datasets, while requiring much less computational cost. In addition, the ResNet152 system is further tested by retrieving the most dissimilar images instead, as well as identifying the query images which give the worst and the 2<sup>nd</sup> worst results when using default value and the binary value feature vectors. The results emphasize the robustness and reliability of the ResNet152-based instance-level retrieval system on small scale hostel datasets with unsupervised learning.

## **Chapter 8: Conclusions and Future Work**

In this final chapter of the thesis, its content is divided into three parts. The first part briefly illustrates the thesis summary. The second part highlights the main findings of this study and its conclusion, as well as the limitations occurred with implemented solutions. Lastly, potential research areas are suggested for future work in order to further overcome the remaining limitations and contribute additional knowledge.

### **8.1 Thesis Summary**

This thesis firstly introduces the concepts of content-based image classification and content-based image retrieval. It further provides an overview of the existing CBIR systems with their promising aspects, as well as challenges. Additionally, a lack of real-life content-based image classification and retrieval application on hostel search is addressed. Consequently, to improve the performance of content-based image classification and retrieval on low-computational devices such as mobile devices and IoT, the focus of this research is to develop fast computable and memory-efficient embeddings for deep learning image features. The attempt to apply proposed approaches to the hostel industry is demonstrated using the first hostel database created for this study. In Chapter 2, a systematic and comprehensive literature review is presented. Starting with the existing CBIR and classification approaches in the computer science and engineering discipline and the tourism discipline. The representation of image content is described in the forms of classical or handcrafted features such as colour, texture, shape, spatial position, and SIFT features and AI-based features. Hostel image examples are also used to picturize these features. Furthermore, the benefits of dimensionality reduction and quantization of image features are highlighted with some approaches found in the relevant literature. Available similarity and performance measures are shown including those are chosen for this research. For example, Hamming distance, Euclidean distance, mean Average Precision, and Precision@k. In Chapter 3, the quantization effect on general image

retrieval and classification is demonstrated including the application of uniform scalar quantizer and the dithering-based scalar uniform quantizer. Next, Chapter 4 investigates the state-of-the-art techniques of fast dimensionality reduction for image classification and retrieval on publicly available datasets which consist of general images. Random projection, PCA, and circular binary embedding are in focus, along with the Hadamard projection and the discrete cosine transform which are implemented in this project development. Chapter 5 presents the first hostel image database which is newly created for this interdisciplinary research. The database includes three datasets including a hostel image dataset used in the hostel image classification task and two London hostel building datasets, Hostel-900 and Hostel-2K, used in instance-level image retrieval task. The image collection process and the information of the collected hostel datasets are provided, as well as image examples. Then, simulations of hostel image classification and retrieval are demonstrated with the application of deep image features and the fast embeddings in Chapter 6 and Chapter 7, accordingly.

## 8.2 Main Findings and Conclusions

As several experiments have been conducted in this study to achieve the aim, developing fast computable and memory-efficient embeddings for content-based image classification and retrieval on low-computational devices such as mobile phones and the Internet of things (IoT) and pioneeringly apply these novel techniques to the hostel industry, this section highlights the main findings as following.

### I. Binary embedding using Golay-Hadamard matrices

In this work, two fast dimensionality reduction operators are proposed based on Golay-Hadamard matrices (GHMs), GHM-Rand and GHM-fix operators, in order to embed an  $N$ -dimensional CNN feature vectors into  $M$  bits. GHM-Rand requires  $N$  random binary bits and  $\mathcal{O}(M \log N)$  additions along with  $N$  sign flipping operations while GHM-fix is completely deterministic with  $\mathcal{O}(N \log N)$  additions and  $2N$  sign flipping operations. To demonstrate the effectiveness of the proposed operators, simulations using various CNN-architectures and different image datasets on unsupervised image retrieval and image classification tasks are carried out. As for the image retrieval which the CNN architecture is adapted from the Resnet101-AP-GeM model and tests on Oxford 5k, ROxford 5k, Paris 6k and RParis 6k datasets, the results highlight that despite their low complexity, the proposed operators offer the best performances in nearly all cases for a given  $M$  except for a couple of cases, in which GHM-Rand and GHM-Fix are slightly worse than those of i.i.d. Gauss matrix. Additionally, when  $M = 2048$  bits, GHM-Rand becomes a fixed operator and its performances are still very close to those of the original one with 65536 bits (2048 floats) whereas GHM-fix offers very similar performance to those of full i.i.d. Gauss matrix and GCMs, though the GHM-Fix is a data oblivious and deterministic operator. On top of these, when the proposed systems are compared to different state-of-the-art supervised deep hashing methods such as supervised semantics-preserving deep hashing (SSDH), hierarchical deep hashing (HSH) and unsupervised ones such as Deepbit, pixels to binary (P2B) codes and embedding and aggregation on selective convolution features (EASC), GHM-Rand operator

offers the best performance in most cases and the performance of GHM-Fix is also very close to that of GHM-Rand while much simple multiplication is required and only followed by a sign operation. In terms of the multi-class image classification experiment, the off-the-shelf GoogLeNet model is tested on Caltech101 and Caltech256 datasets. The results emphasize that GHM-Rand and GHM-Fix operators achieve similar performance and outperform the full i.i.d. Gauss matrix and GCMs operators for all bit length  $M$  on both datasets. In addition, with only 1024 bits, the proposed operators are able to reach higher accuracies than the original length feature vectors (16384 bits) on the Caltech256 dataset. To sum up, despite their low complexity in both computation and storage, GHM-Rand and GHM-Fix offer competitive or even better performance than the state-of-the-art operators. This indicates their promising applications in practical mobile or IoT which tend to be equipped with low-powered and low-buffered elements.

### II. Transfer learning with CNNs for hostel image classification

This simulation of hostel image classification consists of seven experiments aiming to find the most optimal setting when using pretrained CNN models with transfer learning and fine-tuning. SqueezeNet1\_0 is chosen as a baseline CNN model and tested in the first two experiments. The results show that the data split between the training set and the validation set during model training should be 70% and 30%, respectively, and the dropout rate should be 0.3 as they give the highest accuracy during training and relatively high during testing. Further experiments apply additional ten pretrained CNNs, SqueezeNet1\_1, ResNet18, ResNet34, ResNet50, ResNet101, ResNet152, DenseNet121, DenseNet161, DenseNet169, and DenseNet201. DenseNet121 and DenseNet201 outperform other models when considering the accuracy and training time, the accuracy of DenseNet121 is slightly lower but require less training time whereas DenseNet201 reaches higher accuracy but takes longer during training. DenseNet models and ResNet models are able to reach their peak in accuracy and become stable with only a few iterations while SqueezeNet models need around six epochs to reach the peak. When the size of the hostel dataset is reduced, the computational time is significantly decreased across all 11 models with the

classification accuracies only slightly increasing or decreasing. To conclude, with the optimization of parameter fine-tuning and the settings suggested in this simulation, DenseNet201 is the best-performed CNN model on hostel image classification task providing fast and highly accurate image indexing with less labour cost required and fewer human errors. However, DenseNet121 demonstrates competitive results and various pretrained CNNs could also provide good performance.

### III. Instance-level hostel image retrieval with unsupervised learning

This simulation consists of three experiments, the 1<sup>st</sup> experiment is to test four retrieval systems constructed with different renowned pretrained CNN models (ResNet50, ResNet101, ResNet152, and VGG16) which then fine-tuning and unsupervised learning are adopted on four hostel datasets with different dataset sizes, the 2<sup>nd</sup> experiment is to further test the robustness of the four pretrained CNN-based retrieval systems on the same four datasets when using the default value (32 bit-float coefficients) and the binary value (1 bit-binary coefficients) of feature vectors, and the 3<sup>rd</sup> experiment is to test the best-performing system by retrieving the most dissimilar images instead, as well as identifying the query images which give the worst and the 2<sup>nd</sup> worst results when using default value and the binary value feature vectors. The results show the ResNet152-based retrieval system impressively outperforms the other three networks on all four hostel datasets, Hostels-900, Hostels-900 + 100K, Hostels-2K, and Hostels-2K +100K, and achieves 59%-66% accuracy and 90%-100% accuracy when measured with mAP and Precision@1-10, respectively. Similar results of ResNet 152 emphasize its robustness on the 2<sup>nd</sup> experiment as an mAP of 57%-65% accuracy achieved when using binary feature vectors which is a slight degradation compared to default feature vectors and a Precision@1 of 100% accuracy still achieved. Furthermore, when retrieving the most dissimilar images, the overall accuracy remains the same and the same hostel images are identified as the query images which give the worst and the 2<sup>nd</sup> worst results when using default and binary feature vectors. To summarize, the results from these experiments demonstrate the robustness and reliability of the ResNet152-based instance-level retrieval system on small scale hostel datasets with unsupervised learning and has potential for a mobile

application or small-scale search engine which has greater memory, bandwidth, and power constraints.

#### IV. Literature review

On top of the main findings from experiments aforementioned, by conducting the literature review of CBIR and classification study in the computer science and engineering discipline, it is understandable that as a nature of computer science and engineering discipline is to mainly focus on understanding, designing, and developing programmes in relation to computer-oriented subjects, the available literature is dominantly on advancing the existing feature extraction techniques, proposing novel methods for feature extraction, advancing the existing image retrieval techniques, and proposing novel methods for image retrieval in order to create more efficient tools rather than apply the techniques to particular industries. Additionally, several novel CNN architectures are developed for the ImageNet Large Scale Visual Recognition Challenge (ILSVRC), and outside the competition, which contribute to the development of impressive pretrained CNNs from object detection and image classification task using the enormous 14 million ImageNet dataset and many of these CNN models still gain popularity to these days. Furthermore, the fact that this project is interdisciplinary research. The literature review of CBIR and classification studies in the tourism discipline is also included. It can be seen that the trend of the CBIR study is to improve existing feature extraction techniques to better represent image information and eventually accelerate the image retrieval performance with high efficiency and effectiveness. Moreover, some studies attempted to contribute novel techniques in the feature extraction process by fine-tuning fusion features and some attempted to improve the input of the CBIR system, image query. Nevertheless, the applications of these studies are mainly for tourist attractions particularly in art and cultural heritage domains such as landmark images, monument images, historical building images, and painting/object images in the museum. Furthermore, despite there is no evidence of CBIR or classification for the hostel sector, some studies in hotel image retrieval and classification are found in order to improve the existing classification/retrieval techniques and propose new approaches.

### 8.3 Future Work

Despite the promising results from this research, it is undeniable that there is plenty of room to further investigation including overcoming the remaining limitations of this study, following up the current findings, and initiating new approaches in the field. The future works that could undertake immediately in order to solve this work's limitations are suggested below.

#### I. PlacesCNN models comparison

Even though available pretrained CNN models were originally trained on the ImageNet dataset, which contains object-centric image data of over 1.4M images with 1k object classes, Zhou *et al.* [238] gathered Places365 dataset, which contains a large-scale scene-centric image data of over 1.8 M images with 365 scene categories, and trained various CNNs from scratch with this database without changing network architectures. Therefore, comparing the results from pretrained Places365-CNNs with the original pretrained CNNs could provide further comprehensive findings to this work.

#### II. Increasing the size of collected hostel image datasets

Although small-scale image datasets can be used and be able to achieve competitive results in the experiments of this research, upscaling the current size of hostel image datasets, 13,908 images with 28 categories of hostel image dataset and 2,379 images of 20 London hostel building dataset, Hostel-2K, could benefit high accuracy in classification and retrieval tasks as the more data fed to deep learning models, the higher accuracy could be achieved.

Additionally, the follow-up works of this research are outlined below in order to extend the knowledge from the current findings.



### III. Simulations on other large-scale datasets from scratch

As mentioned earlier that the CNNs used in this research are previously trained on the ImageNet dataset, when the computational capacity is increased, other existing large-scale image datasets could be considered for the process of model training. For example, the Open Images dataset [245] is consisted of over 9 million images with 600 diverse object classes. The Street View House Numbers dataset [246] is a real-world image and object detection dataset of over 600k images with 10 classes. COCO [247] is large-scale object detection, segmentation, and captioning dataset which has 330k images with 80 object categories.

### IV. Moving image classification and retrieval

Apart from a vast collection of digital images on online platforms, it is undeniable that digital moving images such as videos, films, and commercials are other data types that are enormously created in today's society. Tankovska [248] reports that, as of May 2019, more than 500 hours of video were uploaded to YouTube in every minute and this number of video content hours grew by around 40% between 2014 and 2019. Therefore, moving image classification and retrieval on low-computational devices could be further researched to accommodate this demand.

Lastly, new approaches could be considered in order to offer an alternative to this field.

### V. Building small-size CNN models

Since smartphone users have surpassed 3B worldwide and are expected to grow hundreds of millions more in the coming years [249], small size CNN models with good performance like SqueezeNet (5.2 MB with 1.24M parameters), MobileNet-v2 (13 MB with 3.5M parameters), EfficientNet-b0 (20 MB with 5.3M parameters) or GoogLeNet model (27 MB with 7M parameters), could be created in order to facilitate mobile applications on image classification/retrieval task and other computer vision tasks in the future.

### VI. Building small-size non-linear CNN models

Most of the available pretrained CNN models consist of a linear sequence of modules which can be clearly understood/explained, easier for the model update, and regularized to avoid overfitting. Nevertheless, non-linear models could improve performance when variables have non-linear relationships and/or there is more than one parameter per variable. NASNet-Mobile (Neural Search Architecture Network-Mobile) (20 MB with 5.3M parameters) and NASNet-Large (332 MB with 88.9 parameters) are examples of the non-linear models. By building the non-linear alternative which is small in size, more CNN options could be explored for tasks on low-computational devices.

### VII. Compressing and accelerating best-performed CNN models

Even though ResNet152, DenseNet121, and DenseNet201 are not the biggest models tested in this study, their model sizes are larger than several pretrained CNNs available. Therefore, model compression and acceleration could be considered while maintaining the model performance without having a significant decrease. Cheng *et al.* [250] categorize the techniques into four approaches, parameter pruning and quantization, low-rank factorization, transferred/compact convolutional filters, and knowledge distillation. In particular, the parameter pruning and quantization, and low-rank factorization approaches support pretrained CNN models. Consequently, these two approaches could be applied in further experiments.

## References

- [1] S. Santini and R. Jain, "The Graphical Specification of Similarity Queries," *Journal of Visual Languages and Computing*, vol. 7, no. 4, pp. 403-421, 1996.
- [2] WTTC, "World Travel & Tourism Council," 2020. [Online]. Available: <https://wttc.org/Research/Economic-Impact>. [Accessed 1 May 2021].
- [3] R. C. Mill and A. M. Morrison, *The Tourism System: An Introductory Text*, Englewood Cliffs: Prentice Hall, 1992.
- [4] P. Murphy, M. P. Pritchard and B. Smith, "The destination product and its impact on traveller perceptions," *Tourism Management*, vol. 21, no. 1, pp. 43-52, 2000.
- [5] S. L. Smith, "The tourism product," *Annals of Tourism Research*, vol. 21, no. 3, pp. 582-595, 1994.
- [6] B. Underwood, "Forbes," 14 July 2016. [Online]. Available: <https://www.forbes.com/sites/under30network/2016/07/14/why-millennials-and-investors-are-flocking-to-u-s-hostels/?sh=3cfa4ec6534f>. [Accessed 1 May 2021].
- [7] T. Mohn, "Forbes," 16 May 2016. [Online]. Available: <https://www.forbes.com/sites/tanyamohn/2016/05/16/a-hostel-revolution-fueled-by-young-travelers/?sh=61bc68af6bb4>. [Accessed 1 May 2021].
- [8] B. Gerrard, "The Telegraph," 27 November 2017. [Online]. Available: <https://www.telegraph.co.uk/business/2017/11/27/hostels-20-place-sling-backpack/>. [Accessed 1 May 2021].
- [9] J. T. Fox, "Hotel Management," 4 January 2017. [Online]. Available: <https://www.hotelmanagement.net/transactions/targeting-millennials-investors-see-value-hostel-market>. [Accessed 1 May 2021].
- [10] P. Whyte, "Skift," 12 November 2018. [Online]. Available: <https://skift.com/2018/11/12/the-big-money-reinvention-of-the-humble-hostel-a-skift-deep-dive/>. [Accessed 1 May 2021].
- [11] P. Whyte, "Skift," 11 April 2019. [Online]. Available: <https://skift.com/2019/04/11/meet-the-ryanair-of-the-hostel-industry/>. [Accessed 1 May 2021].
- [12] R. B. Bunda, "University of Connecticut," 2 May 2014. [Online]. Available: <https://opencommons>.

## REFERENCES

- uconn.edu/cgi/viewcontent.cgi?article=1358&context=srhonors\_theses. [Accessed 1 May 2021].
- [13] S. Aslam, "Omnicores," 6 January 2021. [Online]. Available: <https://www.omnicoreagency.com/facebook-statistics/>. [Accessed 1 May 2021].
- [14] S. Aslam, "Omnicores," 6 January 2021. [Online]. Available: <https://www.omnicoreagency.com/instagram-statistics/>. [Accessed 1 May 2021].
- [15] A. Sabharwal, "Google," 17 May 2017. [Online]. Available: <https://blog.google/products/photos/google-photos-500-million-new-sharing/>. [Accessed 1 May 2021].
- [16] A. Stadlen, "Flickr," 8 March 2019. [Online]. Available: <https://blog.flickr.net/2019/03/08/update-on-creative-commons-licenses-and-in-memoriam-accounts/>. [Accessed 1 May 2021].
- [17] E. Tasli, "Booking," 10 August 2017. [Online]. Available: <https://booking.ai/automated-image-tagging-at-booking-com-7704f27dcc8b>. [Accessed 1 May 2021].
- [18] J. Eakins and M. Graham, "Leeds," 23 November 1999. [Online]. Available: <http://www.leeds.ac.uk/educol/documents/00001240.htm>. [Accessed 1 May 2021].
- [19] F. Strother-Vien, "Mugshot recognition meets witness composite sketches in LA," *Advance Imaging*, vol. 13, no. 1, p. 22, 1998.
- [20] S. Jose, "Almaden," 2000. [Online]. Available: <http://www.almaden.ibm.com/almaden/hermitage.html>. [Accessed 1 May 2021].
- [21] Samsung, "Samsung," 2017. [Online]. Available: <https://www.samsung.com/global/galaxy/apps/bixby/vision/>. [Accessed 1 May 2021].
- [22] Huawei, "Huawei," 2018. [Online]. Available: <https://consumer.huawei.com/uk/support/faq/what-is-hivision/>. [Accessed 1 May 2021].
- [23] Google, "Google," 2017. [Online]. Available: <https://lens.google.com/>. [Accessed 1 May 2021].
- [24] M. Alkhawlani, M. Elmogy and H. E. Bakry, "Text-based, content-based, and semantic-based image retrievals: a survey," *International Journal of Computer and Information Technology*, vol. 4, no. 1, pp. 598-606, 2015.
- [25] M. Tzelepi and A. Tefas, "Deep convolutional learning for Content Based Image Retrieval," *Neurocomputing*, vol. 275, pp. 2467-2478, 2018.
- [26] G.-H. Liu and J.-Y. Yang, "Content-based image retrieval using color difference histogram," *Pattern Recognition*, vol. 46, no. 1, pp. 188-198, 2013.
- [27] S. Murala, R. P. Maheshwari and R. Balasubramanian, "Local Tetra Patterns: A New Feature Descriptor for Content-Based Image Retrieval.," *IEEE Transactions on Image Processing*, vol. 21, no. 5, pp. 2874-2886, 2012.

## REFERENCES

- [28] G.-H. Liu, J.-Y. Yang and Z. Li, "Content-based image retrieval using computational visual attention model," *Pattern Recognition*, vol. 48, no. 8, pp. 2554-2566, 2015.
- [29] B. Nan, Z. Mu and L. Chen, "Content-based Image Retrieval Using Local Texture-Based Color Histogram," in *EEE 2nd International Conference on Cybernetics (CYBCONF)*, 2015.
- [30] J. Yue, Z. Li, L. Liu and Z. Fu, "Content-based image retrieval using color and texture fused features," *Mathematical and Computer Modelling*, vol. 54, no. 3, pp. 1131-1127, 2011.
- [31] R. Brunelli and O. Mich, "Image Retrieval by Examples.," *IEEE Transactions on Multimedia*, vol. 2, no. 3, pp. 164-171, 2000.
- [32] P. Muneesawang and L. Guan, "Automatic Machine Interactions for Content-Based Image Retrieval Using a Self-Organising Tree Map.," *IEEE Transactions on neural networks*, vol. 13, no. 4, pp. 821-834, 2002.
- [33] X. Jin and J. C. French, "Improving Image Retrieval Effectiveness via Multiple Queries," *Multimedia Tools and Applications*, vol. 26, no. 1, pp. 221-245, 2005.
- [34] M. Antonelli, S. G. Dellepiane and M. Goccia, "Design and Implementation of Web-Based Systems for Image Segmentation and CBIR," *IEEE transactions on instrumentation and measure*, vol. 55, no. 6, pp. 1869-1877, 2006.
- [35] R. Rahmani, S. A. Goldman, H. Zhang, J. Krettek and J. E. Fritts, "Localized Content-Based Image Retrieval.," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 11, pp. 1902-1912, 2008.
- [36] R. d. S. Torres, A. X. Falcao, M. A. Goncalves, J. P. Papa, B. Zhang, W. Fan and E. A. Fox, "A genetic programming framework for content-based image retrieval," *Pattern Recognition*, vol. 42, no. 2, pp. 283-292, 2009.
- [37] M. E. ElAlami, "A new matching strategy for content based image retrieval system," *Applied Soft Computing*, vol. 14, pp. 407-418, 2014.
- [38] P. Vijayakumar, R. Abhishek and K. Sandeep, "Hybrid Classifier based Content based Image Retrieval," *Indian Journal of Science and Technology*, vol. 9, no. 46, pp. 1-7, 2016.
- [39] Z. Zhou and L. Zhang, "Content-Based Image Retrieval Using Iterative Search," *Neural Procees Lett*, vol. 47, pp. 907-919, 2017.
- [40] Z.-J. Zha, L. Yang, T. Mei, M. Wang, Z. Wang, T.-S. Chua and X.-S. Hua, "Visual Query Suggestion: Towards Capturing User Intent in Internet Image Search.," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 6, no. 3, pp. 1-19, 2010.
- [41] Y. Wang, Q. Li, T. Lan and J. Chen, "A Comparison of Content Based Image Retrieval Systems,"

## REFERENCES

- IEEE International Conference on Computational Science and Engineering*, pp. 699-673, 2014.
- [42] T. Atre and K. Metre, "MIRS: Text Based and Content Based Image Retrieval.," *International Journal of Engineering Science and Innovative Technology*, vol. 3, no. 4, pp. 579-584, 2014.
- [43] S. Tambe and B. Borkar, "A Review on Image Retrieval System.," *International Journal of Advanced Research in Computer Engineering & Technology*, vol. 3, no. 12, pp. 4268-4271, 2014.
- [44] A. Krizhevsky, S. Llya and H. Geoffrey E., "ImageNet Classification with Deep Convolutional," *Advances in neural information processing systems*, vol. 25, pp. 1097-1105, 2012.
- [45] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich, "Going deeper with convolutions," in *IEEE conference on computer vision and pattern recognition*, 2015.
- [46] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [47] K. Simonyan and A. Zisserman, "VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION," San Diego, 2015.
- [48] F. Landola, S. Han, M. Moskewicz, K. Ashraf, W. Dally and K. Keutzer, 4 November 2016. [Online]. Available: <https://arxiv.org/pdf/1602.07360.pdf> (accessed 3 June 2021). [Accessed 3 June 2021].
- [49] G. Huang, Z. Liu and L. Maaten, 28 January 2018. [Online]. Available: <https://arxiv.org/pdf/1608.06993.pdf>. [Accessed 3 June 2021].
- [50] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva and A. Torralba, "Places: a 10 million image database for scene recognition," Massachusetts, 2017.
- [51] M. Shaha and M. Pawar, "Transfer learning for image classification," in *the 2nd international conference on electronics, communication and aerospace technology*, 656-660.
- [52] B. Zhou, Y. Sun, D. Bau and A. Torralba, "ARXIV," 7 June 2018. [Online]. Available: <https://arxiv.org/abs/1806.02891>. [Accessed 1 October 2021].
- [53] K. E. Van de Sande, T. Gevers and C. G. Snoek, "Evaluating colour descriptors for object and scene recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1582-1597, 2010.
- [54] R. Meltser, S. Banerji and A. Sinha, "What's that Style? A CNN-based approach for classification and retrieval of building images," in *2017 Ninth international conference on advances in pattern recognition*, 2017.

## REFERENCES

- [55] E. J. Pauwels, P. M. de Zeeuw and D. M. Bounantony, "Leatherbacks Matching by Automated Image Recognition," in *ICDM*, 2008.
- [56] C. Town, A. Marshall and N. Sethasathien, "Manta Matcher: automated photographic identification of manta rays using keypoint features," *Ecology and Evolution*, vol. 3, no. 7, pp. 1902-1914, 2013.
- [57] V. P. Singh, S. Srivastava and R. Srivastava, "Automated and effective content-based image retrieval for digital mammography," *Journal of X-Ray Science and Technology*, vol. 26, pp. 29-49, 2017.
- [58] Z. Raisi, F. Mohanna and M. Rezaei, "Applying Content-Based Image Retrieval Techniques to Provide New Services for Tourism Industry," *International Journal Advanced Networking and Applications*, vol. 6, no. 2, pp. 2222-2232, 2014.
- [59] M. Tzelepi and A. Tefas, "Deep convolutional learning for content-based image retrieval," 2017.
- [60] T. Kato and T. Kurita, "Visual interaction with the electronic art gallery.," *Database and Expert Systems Applications: Proceedings on an International Conference*, pp. 234-240, 1990.
- [61] B. Holt and L. Hartwick, "Retrieving art images by image content: the UC Davis QBIC project," *Aslib Proceedings*, vol. 46, no. 10, pp. 243-248, 1994.
- [62] A. K. Jain, S. Goel, S. Agarwal, V. Mittal, H. Sharma and R. Mahindru, "Multimedia systems for art and culture: a case study of Brihadisvara Temple.," in *Storage and Retrieval for image and video databases*, San Jose, 1997.
- [63] W. Premchaiswadi, A. Tungksathan and N. Premchaiswadi, "Mobile image search for tourist information using ACCC algorithm," in *21st Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Istanbul, 2010.
- [64] C. Wengert, M. Douze and H. Jegou, "Bag-of-colors for improved image search," in *The 19th ACM International Conference on Multimedia*, 2011.
- [65] Z. Raisi, F. Mohanna and M. Rezaei, "Content-Based Image Retrieval for Tourism Application," in *2011 7th Iranian Conference on Machine Vision and Image Processing*, Tehran, 2011.
- [66] A. Abdullahzadeh and F. Mohanna, "Content-based image retrieval based on affine noisy invariant colour region," *International Research Journal of Applied and Basic Sciences*, vol. 6, no. 5, pp. 598-606, 2013.
- [67] L. Zheng, S. Wang and Q. Tian, "Coupled Binary Embedding for Large-Scale Image Retrieval," *IEEE Transactions on image processing*, vol. 23, no. 8, pp. 3368-3380, 2014.
- [68] L. Zhu, J. Shen, H. Jin, R. Zheng and L. Xie, "Content-Based Visual Landmark Search via

## REFERENCES

- Multimodal Hypergraph Learning," *IEEE Transactions on cybernetics*, vol. 45, no. 12, pp. 2756-2769, 2015.
- [69] G. Amato, F. Falchi and C. Gennaro, "Fast Image Classification for Monument Recognition," *Journal on Computing and Cultural Heritage*, vol. 8, no. 4, 2015.
- [70] Y. Wang, X. Lin, L. Wu and W. Zhang, "Effective Multi-Query Expansions: Robust Landmark Retrieval," in *The 23rd ACM International conference on Multimedia*, 2015.
- [71] K. Makantasis, A. Doulamis, N. Doulamis and M. Ioannides, "In the wild image retrieval and clustering for 3D cultural heritage landmarks reconstruction," *Multimed Tools Application*, vol. 75, pp. 3593-3629, 2016.
- [72] H. Lacheheb and S. Aouat, "SIMIR: New mean SIFT color multi-clustering image retrieval," *Multimed Tools Application*, vol. 76, pp. 6333-6354, 2017.
- [73] Z. Elleuch and K. Marzouki, "Multi-index structure based on SIFT and color features for large scale image retrieval," *Multimedia Tools and Applications*, vol. 76, no. 13929-13951, 2017.
- [74] V. Lonarkar and B. A. Rao, "Content-based image retrieval by segmentation and clustering," in *2017 International Conference on Inventive Computing and Informations*, Coimbatore, 2017.
- [75] Y. Wang, X. Lin and W. Zhang, "Effective Multi-Query Expansions: Collaborative Deep Networks for Robust Landmark Retrieval," *IEEE Transactions on Image Processing*, vol. 26, no. 3, pp. 1393-1404, 2017.
- [76] C.-C. Hung, "A study on a content-based image retrieval technique for Chinese paintings," *The Electronic Library*, vol. 36, no. 1, pp. 172-188, 2018.
- [77] K. S. Arun, V. K. Govindan and S. M. Kumar, "Enhanced bag of visual words representations for content based image retrieval: a comparative study," *Artificial Intelligence Review*, vol. 53, pp. 1615-1653, 2019.
- [78] P. Basak, P. Dhankar, S. Gupta and P. Verma, "Image Based Information Retrieval," *International Research Journal of Engineering and Technology*, vol. 6, no. 5, pp. 6011-6015, 2019.
- [79] INRIALPES, "INRIALPES," 2008. [Online]. Available: <https://lear.inrialpes.fr/~jegou/data.php>. [Accessed 1 September 2020].
- [80] A. Stylianou, R. Souvenir and R. Pless, "ARXIV," 8 October 2019. [Online]. Available: <https://arxiv.org/pdf/1910.03455.pdf>. [Accessed August 4 2021].
- [81] A. Stylianou, H. Xuan, M. Shende, J. Brandt, R. Souvenir and R. Pless, "Hotels-50K: A Global Hotel Recognition Dataset," in *The AAAI conference on Artificial Intelligence*, 2019.



## REFERENCES

- [82] H. Xuan, A. Stylianou and R. Pless, "TheCVF," 2020. [Online]. Available: [https://openaccess.thecvf.com/content\\_WACV\\_2020/papers/Xuan\\_Improved\\_Embeddings\\_with\\_Easy\\_Positive\\_Triplet\\_Mining\\_WACV\\_2020\\_paper.pdf](https://openaccess.thecvf.com/content_WACV_2020/papers/Xuan_Improved_Embeddings_with_Easy_Positive_Triplet_Mining_WACV_2020_paper.pdf). [Accessed 4 August 2021].
- [83] R. Kamath, G. Rolwes, S. Black and A. Stylianou, "ARXIV," 14 June 2021. [Online]. Available: <https://arxiv.org/pdf/2106.05746.pdf>. [Accessed 4 August 2021].
- [84] B. Tseytlin and I. Makarov, "ARXIV," 15 June 2021. [Online]. Available: <https://arxiv.org/pdf/2106.08042.pdf>. [Accessed 4 August 2021].
- [85] T. Kanchinadam, "Github," 2016. [Online]. Available: [https://tkanchin.github.io/images/ml\\_report.pdf](https://tkanchin.github.io/images/ml_report.pdf). [Accessed 1 May 2021].
- [86] Kaggle, "Kaggle," 2016. [Online]. Available: <https://www.kaggle.com/c/hotel-image-classification/overview/>. [Accessed 1 May 2021].
- [87] M. Ren, H. Q. Vu, G. Li and R. Law, "Large-scale comparative analyses of hotel photo content posted by managers and customers to review platforms based on deep learning: implications for hospitality marketers," *Journal of Hospitality Marketing & Management*, vol. 30, no. 1, pp. 96-119, 2021.
- [88] S. R. Rath, "Deguggercafe," 9 March 2020. [Online]. Available: <https://debuggercafe.com/getting-95-accuracy-on-the-caltech101-dataset-using-deep-learning/>. [Accessed 1 May 2021].
- [89] S. Mei, W. Min, H. Duan and S. Jiang, "Instance-level object retrieval via deep region CNN," *Multimedia Tools and Applications*, vol. 78, pp. 13247-13261, 2019.
- [90] A. Alzu'bi, A. Amira and N. Ramzan, "Content-based image retrieval with compact deep convolutional features," *Neurocomputing*, vol. 249, pp. 95-105, 2017.
- [91] M. Tzelepi and A. Tefas, "Exploiting supervised learning for finetuning deep CNNs in Content Based Image Retrieval," in *23rd International Conference on Pattern Recognition (ICPR)*, Cancun, 2016.
- [92] P.-X. Sun, H.-T. Lin and T. Luo, "Learning discriminative CNN features and similarity metrics for image retrieval," Hong Kong, 2016.
- [93] A. Gordo, J. Almazan, J. Revaud and D. Larlus, "End-to-end learning of deep visual representations for image retrieval," *Int. J. Comput. Vis.*, vol. 124, no. 2, pp. 237-254, 2017.
- [94] F. Radenovic, G. Tolias and O. Chum, "Fine-Tuning CNN Image Retrieval with No Human Annotation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 7, pp. 1655-1668, 2019.
- [95] H. Muller, N. Michoux, D. Bandon and A. Geissbuhler, "A review of content-based image

## REFERENCES

- retrieval systems in medical applications—clinical benefits and future directions," *International Journal of Medical Informatics*, vol. 73, no. 1, pp. 1-23, 2004.
- [96] A. W. Smeulders, M. Worrying, S. Santini, A. Gupta and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349-1380, 2000.
- [97] H. Muller, N. Michoux, D. Bandon and A. Geissbuhler, "A review of content-based image retrieval systems in medical applications—clinical benefits and future directions," *International Journal of Medical Informatics*, vol. 73, no. 1, pp. 1386-5056, 2004.
- [98] C. B. Akgul, D. L. Rubin, S. Napel, C. F. Beaulieu, H. Greenspan and B. Acar, "Content-Based Image Retrieval in Radiology: Current Status and Future Directions," *Journal of Digital Imaging*, vol. 24, no. 2, pp. 208-222, 2011.
- [99] M. Swain and D. H. Ballard, "Color Indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11-32, 1991.
- [100] M. A. Stricker and A. Dimai, "Color indexing with weak spatial constraints.," *Storage and Retrieval for Image and Video Databases*, vol. 2670, no. 1, pp. 29-40, 1996.
- [101] C. Carson, S. Belogie, H. Greenspan and J. Malik, "Region-based image querying.," in *IEEE Workshop on Content-Based Access of Image and Video Libraries*, 1997.
- [102] V. Chitkara, "Color-based image retrieval using compact binary signatures," Tech., Alberta, 2001.
- [103] S. Sural, G. Qian and S. Pramanik, "A histogram with perceptually smooth color transition for image retrieval.," in *the 6th Joint Conf. on Information Science*, North Carolina, 2002.
- [104] N. S. Vassilieva, "Content-based Image Retrieval Methods," *Programming and Computer Software*, vol. 35, no. 3, pp. 158-180, 2009.
- [105] J. M. Cauvin, C. Le Guillou, B. Solaiman, M. Robaszekiewicz, P. Le Beux and C. Roux, "Computer-assisted diagnosis system in digestive endoscopy.," *IEEE Transaction Information Technology Biomed.*, vol. 7, no. 4, pp. 256-262, 2003.
- [106] K. Lubber, "A probabilistic approach to medical image retrieval.," *Multilingual Information Access for Text, Speech and Images*, pp. 761-772, 2005.
- [107] MathWorks, "MathWorks," 2021. [Online]. Available: [https://uk.mathworks.com/help/matlab/creating\\_plots/color-analysis-with-bivariate-histogram.html](https://uk.mathworks.com/help/matlab/creating_plots/color-analysis-with-bivariate-histogram.html). [Accessed 1 May 2021].
- [108] R. M. Haralick, K. Shanmugum and I. Dinstein, "Textural features for image classification.," *IEEE Trans. Systems*, vol. 3, no. 6, pp. 610-621, 1973.
- [109] S. Askoy and R. Haralic, "Graph-theoretic clustering for image grouping and retrieval,"

## REFERENCES

- Computer Vision and Pattern Recognition*, pp. 63-68, 1999.
- [110] H. Tamura, S. Mori and T. Yamawaki, "Textural features corresponding to visual perception.," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 8, no. 6, pp. 460-472, 1978.
- [111] MathWorks, "MathWorks," 2021. [Online]. Available: <https://uk.mathworks.com/help/images/create-a-gray-level-co-occurrence-matrix.html>. [Accessed 1 May 2021].
- [112] C. W. Niblack, R. Barber, W. Equitz, M. D. Flickner, E. H. Glasman, D. Petkovic, P. Yanker, C. Faloutsos and G. Taubin, "The QBIC project: querying images by color, texture and shape.," IBM, 1993.
- [113] R. Mehrotra and J. E. Gary, "Similar-shape retrieval in shape data management.," *IEEE Computer*, vol. 28, no. 9, pp. 57-62, 1995.
- [114] Y. Rui, A. C. She and T. S. Huang, "Modified fourier descriptors for shape representation-a practical approach.," 1996.
- [115] C. L. Huang and D. H. Huang, "A content-based image retrieval system.," *Image Vision Computing*, vol. 16, pp. 149-163, 1998.
- [116] D. Zhang and G. Lu, "A comparative study on shape retrieval using fourier descriptors with different shape signatures.," in *Int. Conf. on Multimedia*, 2001.
- [117] D.-J. Lee, S. Antani and L. R. Long, "Similarity measurement using polygon curve representation and fourier descriptors for shape based vertebral image retrieval.," *Image Processing*, pp. 1283-1291, 2003.
- [118] M. K. Hu, "Visual pattern recognition by moment invariants.," *IEEE Trans. Information Theory*, vol. 8, no. 2, pp. 179-187, 1962.
- [119] S. O. Belkasim, M. Shridhar and M. Ahmadi, "Pattern recognition with moment invariants: A comparative study and new results," *Pattern Recognition*, vol. 24, no. 12, pp. 1117-1138, 1991.
- [120] Y. Luren and A. Fritz, "Fast computation of invariant geometric moments: a new method giving correct results.," in *IEEE Int. Conf. on Image Processing*, 1994.
- [121] M. Mercimek, K. Gulez and T. V. Mumcu, "Real object recognition using moment invariants.," *Adhana-Academy Proc. in Engineering Sciences*, vol. 30, no. 6, pp. 765-775, 2005.
- [122] D. Yang, J. H. Garrett, D. S. Shaw and L. A. Rendell, "An intelligent symbol usage assistant for CAD systems.," *IEEE Expert*, vol. 10, no. 3, pp. 32-41, 2008.
- [123] MathWorks, "MathWorks," 2021. [Online]. Available: <https://uk.mathworks.com/help/matlab/math/fourier-transforms.htm>. [Accessed 1 May 2021].
- [124] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, Massachusetts: The MIT Press, 2016.

## REFERENCES

- [125] R. Parloff, "Fortune," 28 September 2016. [Online]. Available: <http://fortune.com/ai-artificial-intelligence-deep-machine-learning/> . [Accessed 1 May 2021].
- [126] T. Kato, " Database architecture for content-based image retrieval.," *Image Storage and Retrieval Systems*, vol. 1662, no. 1, pp. 112-123, 1992.
- [127] D. Adit, "Github," 2016. [Online]. Available: <https://adeshpande3.github.io/The-9-Deep-Learning-Papers-You-Need-To-Know-About.html>. [Accessed 1 May 2021].
- [128] K. R. Kruthika, R. Rajeswarl and H. D. Maheshappa, "CBIR system using capsule networks and 3D CNN for alzheimer's disease diagnosis," *Informatic in Medicine Unlocked*, vol. 14, no. 1, pp. 59-68, 2019.
- [129] M. McCombe, 24 May 2019. [Online]. Available: <https://towardsdatascience.com/intro-to-feature-selection-methods-for-data-science-4cae2178a00a>. [Accessed 1 May 2021].
- [130] P. P. Ippolito, 10 October 2019. [Online]. Available: <https://towardsdatascience.com/feature-extraction-techniques-d619b56e31be>. [Accessed 1 May 2021].
- [131] R. Agarwal, 27 July 2019. [Online]. Available: <https://towardsdatascience.com/the-5-feature-selection-algorithms-every-data-scientist-need-to-know-3a6b566efd2>.
- [132] Z. M. Hira and D. F. Gillies, "A Review of Feature Selection and Feature Extraction Methods Applied on Microarray Data," *Advances in Bioinformatics*, pp. 1-13, 2015.
- [133] Y. Hatem and S. Rady, "Exploring feature dimensionality reduction methods for enhancing automatic sport image annotation," *Multimedia Tools and Applications*, vol. 77, pp. 9171-9188, 2018.
- [134] M. McCombe, "Towardsdatascience," 24 May 2019. [Online]. Available: <https://towardsdatascience.com/intro-to-feature-selection-methods-for-data-science-4cae2178a00a>. [Accessed 1 May 2021].
- [135] C. H. Park, H. Park and P. Pardalos, "A comparative study of linear and nonlinear feature extraction methods," in *ICDM 2004*, 2004.
- [136] W. Chen, P. C. Yuen, B. Fang and P. S. Wang, *Linear and Nonlinear Feature Extraction Approaches for Face Recognition*, Berlin: Springer, 2011.
- [137] K. F. Pearson, "On lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559-572, 1901.
- [138] P. Comon, "Independent Component Analysis," Elsevier, London, 1991.
- [139] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*,

## REFERENCES

- vol. 7, pp. 179-188, 1936.
- [140] J. Wang, *Locally Linear Embedding*, Berlin: Springer, 2012.
- [141] L. v. d. Maaten and G. Hinton, "Visualizing Data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579-2605, 2008.
- [142] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning internal representations by error propagation," *Parallel Distributed Processing.*, vol. 1, 1986.
- [143] S. Petscharnig, M. Lux and S. Chatzichristofis, "Dimensionality Reduction for Image Features using Deep Learning and Autoencoders," in *the 15th International Workshop on Content-Based Multimedia Indexing*, 2017.
- [144] T. Cover and J. Thomas, *Elements of Information Theory*, John Wiley & Sons, 2006.
- [145] H. Tanioka, "A Fast Content-Based Image Retrieval Method Using Deep Visual Features," in *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, Sydney, 2019.
- [146] A. N. Sushen and A. S. Vibhute, "An Effective Content Based Image Retrieval Using Dot Diffusion Block Truncation Coding," in *2019 International Conference on Communication and Electronics Systems (ICCES)*, Coimbatore, 2019.
- [147] W. Hong, X. Tang, J. Meng and J. Yuan, "Asymmetric Mapping Quantization for Nearest Neighbor Search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 7, pp. 1783-1790, 2020.
- [148] J.-P. Heo, Z. Lin and S.-E. Yoon, "Distance Encoded Product Quantization for Approximate K-Nearest Neighbor Search in High-Dimensional Space," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 9, pp. 2084-2097, 2019.
- [149] F. Gao, H. Zeng and Z. Huang, "Vector Quantization for Large Scale CBIR," in *2019 International Conference on Intelligent Computing, Automation and Systems (ICICAS)*, Chongqing, 2019.
- [150] E. Doutsis, L. Fillatre, M. Antonini and P. Tsakalides, 6 November 2019. [Online]. Available: [https://hal.archives-ouvertes.fr/hal-02351180/file/Doutsis-Dynamic\\_Image\\_Quantization\\_using\\_LIF\\_neurons\\_TIP2020-Revision.pdf](https://hal.archives-ouvertes.fr/hal-02351180/file/Doutsis-Dynamic_Image_Quantization_using_LIF_neurons_TIP2020-Revision.pdf). [Accessed 1 May 2021].
- [151] M. Oger, S. Ragot and M. Antonini, "Model-based deadzone optimization for stack-run audio coding with uniform scalar quantization," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, Las Vegas, 2008.
- [152] A. Cohen, N. Shlezinger, S. Salamatian, Y. C. Eldar and M. Medard, "Distributed Quantization

## REFERENCES

- for Sparse Time Sequences," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Barcelona, 2020.
- [153] P. Wang, X. He, Y. Zhang, W. Wen and M. Li, "A robust and secure image sharing scheme with personal identity information embedded," *Computers & Security*, vol. 85, pp. 107-121, 2019.
- [154] V. Sheinin, A. Jagmohan and D. He, "Uniform Scalar Quantization Based Wyner-Ziv Coding of Laplace-Markov Source," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*, Honolulu, 2007.
- [155] V. Sheinin, A. Jogmohan and D. He, "On the Operational Rate-Distortion Performance of Uniform Scalar Quantization-Based Wyner-Ziv Coding of Laplace-Markov Sources," *IEEE Transactions on Multimedia*, vol. 10, no. 7, pp. 1225-1236, 2008.
- [156] Y. Lu, J. Zheng, Y. Jiang, M. Yang, B. Fu and W. Hou, "Progressive Image Transmission for Medical Applications based on," in *the 28th IEEE EMBS Annual International conference*, New York, 2006.
- [157] Z. Peric, B. Denic and V. Despotovic, "Gaussian source coding based on variance-mismatched three-level scalar quantisation using Q-function approximations," *IET communications*, vol. 14, no. 4, pp. 594-602, 2020.
- [158] M. Marinov, I. Valova and Y. Kalmukov, "Comparative Analysis of Existing Similarity Measures used for Content-based Image Retrieval," in *2019 X National Conference with International Participation (ELECTRONICA)*, Sofia, 2019.
- [159] T. Ahmad, C. Ceyhun, H. Ahmad and C. Dmitry, "Detailed investigation of deep features with sparse representation and dimensionality reduction in CBIR: A comparative study," *Intelligent Data Analysis*, vol. 24, no. 1, pp. 47-68, 2020.
- [160] Y. Gong, S. Kumar, H. A. Rowley and S. Lazebnik, "Learning binary codes for high-dimensional data using bilinear projections," Portland, 2013.
- [161] T. T. Do, T. Hoang, D.-K. L. TAn, H. Le, T. V. Nguyen and N. M. Cheung, "From selective deep convolutional features to compact binary representations for image retrieval," *ACM Trans. Multim. Comput. Commun. Appl.*, vol. 15, no. 2, pp. 1-22, 2019.
- [162] S. Dirksen and A. Stollenwerk, 2020. [Online]. Available: <https://arxiv.org/abs/2009.08320>. [Accessed 1 May 2021].
- [163] V. Tyagi, Content based image retrieval ideas influences and current trends, Singapore: Springer Nature Singapore Pte Ltd., 2017.
- [164] Y. Liu, D. Zhang, G. Lu and W.-Y. Ma, "A survey of content-based image retrieval with

## REFERENCES

- high-level semantics," *Pattern Recognition*, vol. 40, pp. 262-282, 2007.
- [165] A. Alzubi, A. Amira and N. Ramzan, "Semantic content-based image retrieval: A comprehensive study," *J. Vis. Commun. Image R.*, vol. 32, pp. 20-54, 2015.
- [166] F. Yu, S. Kumar, Y. Gong and S.-F. Chang, "Circulant Binary Embedding," in *the 31st International Conference on Machine Learning*, 2014.
- [167] F. Yu, A. Bhaskara, S. Kumar, Y. Gong and S. Chang, "On binary embedding using circulant matrices," *J. Mach. Learn. Res.*, vol. 18, pp. 1-30, 2017.
- [168] T. T. Do, T. Hoang, K. L. Tan, T. Pham, H. Le, N. M. Cheung and I. D. Reid, "Binary constrained deep hashing network for image retrieval without manual annotation," in *IEEE Winter Conference on Applications of Computer Vision WACV Waikoloa Village*, 2019.
- [169] Y. Gong, S. Lazebnik, A. Gordo and F. Perronnin, "Iterative Quantization: A Procrustean Approach to Learning Binary Codes for Large-Scale Image Retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence* (, vol. 35, no. 12, pp. 2916-2929, 2013.
- [170] P. T. Boufounos, H. Mansour, S. Rane and A. Vetro, "Dimensionality reduction of visual features for efficient retrieval and classification," *APSIPA Transactions on Signal and Information Processing*, vol. 5, 2016.
- [171] S. Chavda and M. M. Goyani, "Researchgate," 2019. [Online]. Available: [https://www.researchgate.net/profile/Goyani-Mahesh/publication/338208131\\_Content-Based\\_Image\\_Retrieval\\_The\\_State\\_of\\_the\\_Art/links/5e06ca1292851c83649fd4b2/Content-Based-Image-Retrieval-The-State-of-the-Art.pdf](https://www.researchgate.net/profile/Goyani-Mahesh/publication/338208131_Content-Based_Image_Retrieval_The_State_of_the_Art/links/5e06ca1292851c83649fd4b2/Content-Based-Image-Retrieval-The-State-of-the-Art.pdf). [Accessed 1 May 2021].
- [172] A. Latif, A. Rasheed, U. Sajid, J. Ahmed, N. Ali , N. I. Ratyal, B. Zafar, S. H. Dar, M. Sajid and T. Khalil, "Content-Based Image Retrieval and Feature Extraction: A," *Mathematical Problems in Engineering*, vol. 2019, pp. 1-21, 2019.
- [173] J. Philbin, O. Chum, M. Isard, J. Sivic and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008.
- [174] F. Radenovic, G. Tolias and O. Chum, "Fine-Tuning CNN Image Retrieval with no human annotation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 7, pp. 1655-1668, 2019.
- [175] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov and A. Rabinovich, "Going deeper with convolutions," in *the IEEE conference on computer vision and pattern recognition*, 2015.

## REFERENCES

- [176] L. Jacques and V. Cambareri, "Time for dithering: fast and quantized random embeddings," *A journal of the IMA*, vol. 6, pp. 441-476, 2017.
- [177] C. Xu and L. Jacques, "Quantized compressive sensing with RIP matrices: the benefit of dithering," *A journal of the IMA*, pp. 1-44, 2019.
- [178] Q. Zhao, H. Feng and M. Effros, "Multiresolution source coding using entropy constrained dithered scalar quantization," in *Data compression conference*, Snowbird, 2004.
- [179] D. R. Bull, "Discrete-Time Analysis for Images and Video," in *Communicating Pictures*, Academic Press, 2014, pp. 63-98.
- [180] F. Sheng, L. Xu-Jian and Z. Li-Wei, "A Lloyd-Max-based Non-Uniform Quantization Scheme for Distributed Video Coding," in *Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD 2007)*, Qingdao, 2007.
- [181] W. Ling, 2007. [Online]. Available: <https://www.sciencedirect.com/topics/engineering/quantisation>. [Accessed 1 May 2021].
- [182] K. R. Murthy and A. Ghosh, "Moments discriminant analysis for supervised dimensionality reduction," *Neurocomputing*, vol. 237, pp. 114-132, 2017.
- [183] J. Lacotte, S. Liu, E. Dobriban and M. Pilanci, "Limiting Spectrum of Randomized Hadamard Transform and," in *Conference on Neural Information Processing Systems*, 2020.
- [184] Z. Lei and L. Lan, "Improved Subsampled Randomized Hadamard Transform for Linear SVM," in *the AAAI Conference on Artificial Intelligence*, 2020.
- [185] W. Johnson and L. Lindenstrauss, "Extensions of Lipschitz mappings into Hilbert space," *Contemporary mathematics*, vol. 26, pp. 189-206, 1984.
- [186] R. Upadhyay, P. Panse, A. Soni and U. R. Bhatt, "Principal Component Analysis as a Dimensionality Reduction and Data Preprocessing Technique," in *Recent Advances in Interdisciplinary Trends in Engineering & Applications (RAITEA) 2019*, 2019.
- [187] Mathworks, "Mathworks," 2021. [Online]. Available: <https://uk.mathworks.com/help/stats/principal-component-analysis-pca.html>. [Accessed 1 May 2021].
- [188] M. Angulakshmi and G. L. Priya, "Walsh Hadamard Transform for Simple Linear Iterative Clustering (SLIC) Superpixel Based Spectral Clustering of Multimodal MRI Brain Tumor Segmentation," *IRBM*, vol. 40, no. 5, pp. 253-262, 2019.
- [189] N. Meenakshisundaram, "Design and analysis of dual periodic optical superlattices using Walsh-Hadamard transform matrix," *Opt Quant Electron*, vol. 48, no. 176, pp. 1-9, 2016.



## REFERENCES

- [190] L. Gong, K. Qiu, C. Deng and N. Zhou, "An optical image compression and encryption scheme based on compressive sensing and RSA algorithm," *Optics and Lasers in Engineering*, vol. 121, pp. 169-180, 2019.
- [191] E. E. Abdallah, A. F. Otoom, A. E. Abdallah, M. Bsoul and S. Awwad, "A Hybrid Secure Watermarking Scheme Using," *Journal of Applied Security Research*, vol. 15, no. 2, pp. 185-198, 2020.
- [192] MathWorks, 2021. [Online]. Available: <https://uk.mathworks.com/help/signal/ug/walshhadamard-transform.html>. [Accessed 1 May 2021].
- [193] MathWorks, 2021. [Online]. Available: <https://uk.mathworks.com/help/signal/ref/fwht.html>. [Accessed 1 May 2021].
- [194] A. Kumar, A. Kansal and K. Singh, "Anti-forensic approach for JPEG compressed images with enhanced image quality and forensic undetectability," *Multimedia Tools and Applications*, vol. 79, pp. 8061-8084, 2020.
- [195] S. S. Sawant and P. Manoharan, "Unsupervised band selection based on weighted information entropy and 3D discrete cosine transform for hyperspectral image classification," *International journal of remote sensing*, vol. 41, no. 10, pp. 3948-3969, 2020.
- [196] D. Moghadas, "Probabilistic Inversion of Multiconfiguration Electromagnetic Induction Data Using Dimensionality Reduction Technique: A Numerical Study," *Vadose Zone journal*, 2019.
- [197] B. Yang, "Multiobjective Synthesis of Linear Arrays by Using an Improved Genetic Algorithm," *International Journal of Antennas and Propagation*, vol. 2019, 2019.
- [198] A. Shawahna, E. Haque and A. Amin, "ARXIV," Cornell University, 1 November 2019. [Online]. Available: <https://arxiv.org/abs/1912.10789>. [Accessed 1 May 2021].
- [199] Z. Moghaddasi, H. A. Jalab and R. Noor, "Image splicing forgery detection based on low-dimensional singular value decomposition of discrete cosine transform coefficients," *Image splicing forgery detection based on low-dimensional singular value decomposition of discrete cosine transform coefficients*, vol. 31, pp. 7867-7877, 2019.
- [200] M. Zheng, J. Zheng, Z. Chen, L. Wu, X. Yang and N. Ling, "A Reconfigurable Architecture for Discrete," *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY*, vol. 30, no. 3, pp. 810-811, 2020.
- [201] MathWorks, 2021. [Online]. Available: <https://uk.mathworks.com/help/images/discrete-cosine-transform.html>. [Accessed 1 May 2021].
- [202] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L.-C. Chen, "MobileNetV2: Inverted Residuals

## REFERENCES

- and Linear Bottlenecks," in *IEEE conference on computer vision and pattern recognition*, 2018.
- [203] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," Arxiv, 10 December 2015. [Online]. Available: <https://arxiv.org/pdf/1512.03385.pdf>. [Accessed October 2020].
- [204] V. Nguyen and M. N. Do, "Deep learning based supervised hashing for efficient image retrieval," in *2016 IEEE International Conference on Multimedia and Expo (ICME)*, 2016.
- [205] J. Bai, B. Ni, M. Wang, Z. Li, S. Cheng, X. Yang, C. Hu and W. Gao, "Deep progressive hashing for image retrieval," *IEEE Transactions on Multimedia*, vol. 21, no. 12, pp. 3178-3193, 2019.
- [206] H. Zhang, Y. Gu, Y. Yao, Z. Zhang, L. Liu, J. Zhang and L. Shao, "Deep unsupervised self-evolutionary hashing for image retrieval," *IEEE Transactions on Multimedia*, p. 1, 2020.
- [207] G. Song and X. Tan, "Hierarchical deep hashing for image retrieval," *Frontiers Comput. Sci.*, vol. 11, no. 2, pp. 253-265, 2017.
- [208] K. Lin, J. Lu, C. S. Chen, J. Zhou and M. T. Sun, "Unsupervised deep learning of compact binary descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 6, pp. 1501-1514, 2019.
- [209] H. F. Yang, K. Lin and C. S. Chen, "Supervised learning of semantics-preserving hash via deep convolutional neural networks," *IEEE Trans. Pattern Anal. Mach.*, vol. 40, no. 2, pp. 437-451, 2018.
- [210] Y. Plan and R. Vershynin, "Dimension reduction by random hyperplane tessellations," *Discret. Comput. Geom.*, vol. 51, no. 2, pp. 438-461, 2014.
- [211] X. Yi, C. Caramanis and E. Price, "Binary embedding: Fundamental limits and fast algorithm," in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015*, Lille, 2015.
- [212] A. Choromanska, K. Choromanski, M. Bojarski, T. Jebara, S. Kumar and Y. LeCun, "Binary embeddings with structured hashed projections," in *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, New York, 2016.
- [213] A. Andoni, P. Indyk, T. Laarhoven, I.-P. Razenshteyn and L. Schmidt, "Practical and optimal LSH for angular distance," in *Annual Conference on Neural Information Processing Systems 2015*, Montreal, 2015.
- [214] L. Gan, L. Liu and Y. Shen, "Golay sequence for partial Fourier and Hadamard compressive imaging," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013.
- [215] M. Golay, "Complementary series," *IEEE Trans. Inf. Theory*, vol. 7, no. 2, pp. 82-87, 1961.

## REFERENCES

- [216] N. Ailon and E. Liberty, "Fast dimension reduction using Rademacher series on dual BCH codes," *Discret. Comput*, vol. 42, p. 615, 2009.
- [217] J. Philbin, O. Chum, M. Isard, J. Sivic and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007)*, 2007.
- [218] F. Radenovic, A. Iscen, G. Toliás, Y. Avrithis and O. Chum, "Revisiting Oxford and Paris: Large-scale image retrieval benchmarking," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, 2018.
- [219] J. Revaud, J. Almazan, R. S. Rezende and C. R. de Souza, "Learning with average precision: Training image retrieval with a listwise loss," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019*, Seoul, 2019.
- [220] A. Babenko, A. Slesarev, A. Chigorin and V. S. Lempitsky, "Neural codes for image retrieval," in *Computer Vision - ECCV 2014 - 13th European Conference*, Zurich, 2014.
- [221] J. Almazan and Y. Cabon, 2019. [Online]. Available: <https://github.com/naver/deep-image-retrieval>. [Accessed 1 May 2021].
- [222] F.-F. Li, R. Fergus and P. Perona, "Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories, washington, usa," in *2004 Conference on Computer Vision and Pattern Recognition Workshop*, 2004.
- [223] G. Gregory, H. Alex and P. Pietro, 2007. [Online]. Available: [http://www.vision.caltech.edu/Image\\_Datasets/Caltech256/](http://www.vision.caltech.edu/Image_Datasets/Caltech256/). [Accessed 1 May 2021].
- [224] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Ravinovich, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [225] J. Wang, S. Kumar and S. Chang, "Semi-supervised hashing for large-scale image retrieval," in *Computer vision and pattern recognition*, 2010.
- [226] F.-F. Li and M. Andreetto, "Vision," 2003. [Online]. Available: [http://www.vision.caltech.edu/Image\\_Datasets/Caltech101/](http://www.vision.caltech.edu/Image_Datasets/Caltech101/). [Accessed 1 May 2021].
- [227] G. Griffin, A. Holub and P. Perona, "Caltech," 2007. [Online]. Available: <https://authors.library.caltech.edu/7694/>. [Accessed 1 May 2021].
- [228] B. Zhou, A. Lapedriza, A. Khosla, A. Torralba and A. Oliva, "MIT," 2017. [Online]. Available: <http://places2.csail.mit.edu/download.html>. [Accessed 1 May 2021].
- [229] C. Ammatmanee and L. Gan, "Google Drive," 2021. [Online]. Available:

## REFERENCES

- [https://drive.google.com/drive/u/1/folders/1xt\\_nCzIAzoyARPOhLWqTq09ehJwTeCrp](https://drive.google.com/drive/u/1/folders/1xt_nCzIAzoyARPOhLWqTq09ehJwTeCrp).  
[Accessed 3 June 2021].
- [230] C. Ammatmanee and L. Gan, 2021. [Online]. Available: [https://drive.google.com/drive/u/1/folders/1Nu3jkzdsxMifq2W\\_z8y8-pDvc8rwCWKc](https://drive.google.com/drive/u/1/folders/1Nu3jkzdsxMifq2W_z8y8-pDvc8rwCWKc).
- [231] C. Ammatmanee and L. Gan, 24 August 2021. [Online]. Available: [https://drive.google.com/drive/u/1/folders/1cyCtgQHgmODBhfY2MfBtC\\_iak-eE0ZWI](https://drive.google.com/drive/u/1/folders/1cyCtgQHgmODBhfY2MfBtC_iak-eE0ZWI). [Accessed 2021].
- [232] A. Dawud, K. Yurtkan and H. Oztoprak, "Application of deep learning in neuroradiology: brain haemorrhage classification using transfer learning," *Computational intelligence and neuroscience*, vol. 2019, pp. 1-12, 2019.
- [233] F. Li, S. Kant, S. Araki, S. Bangera and S. S. Shukla, 2020. [Online]. Available: <https://arxiv.org/pdf/2005.08170.pdf>. [Accessed 1 May 2021].
- [234] M. Mateen, J. Wen, N. Nasrullah, S. Sun and S. Hayat, "Exudate detection for diabetic retinopathy using pretrained convolutional neural networks," *Complexity*, vol. 2020, pp. 1-11, 2020.
- [235] T. Xiao, L. Liu, K. Li, W. Qin, S. Yu and Z. Li, "Comparaison of transferred deep neural networks in ultrasonic breast masses discrimination," *Biomed research international*, vol. 2018, pp. 1-9, 2018.
- [236] MathWorks, "MathWorks," 2021. [Online]. Available: <https://uk.mathworks.com/help/deeplearning/ug/pretrained-convolutional-neural-networks.html>. [Accessed 1 May 2021].
- [237] F. Landola, 2017. [Online]. Available: [https://github.com/forresti/SqueezeNet/tree/master/SqueezeNet\\_v1..](https://github.com/forresti/SqueezeNet/tree/master/SqueezeNet_v1..) [Accessed 29 June 2021].
- [238] B. Zhao, B. Huang and Y. Zhong, "Transfer learning with fully pretrained deep convolution networks for land-use classification," *IEEE geoscience and remote sensing letters*, vol. 14, no. 9, pp. 1436-1440, 2017.
- [239] C. Ammatmanee, "Google," 2020. [Online]. Available: [https://drive.google.com/drive/u/1/folders/1CGw29zKvhW8eeXdo-JsCqN-rc2E56O\\_D](https://drive.google.com/drive/u/1/folders/1CGw29zKvhW8eeXdo-JsCqN-rc2E56O_D). [Accessed 1 February 2020].
- [240] T. Rath, "Medium," 2018. [Online]. Available: <https://medium.com/kayak-tech/hotel-image-categorization-with-deep-learning-ffa8429e55b5>. [Accessed 1 May 2021].
- [241] J. Philbin, O. Chum, J. Sivic and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, 2007.
- [242] B. Cao, A. Araujo and J. Sim, Unifying deep local and global features for image search,

## REFERENCES

- Glasgow: Springer, 2020.
- [243] G. Tolas, Y. Avrithis and H. Jegou, "Image search with selective match kernels: Aggregation across single and multiple images," *International Journal of Computer Vision*, vol. 116, pp. 247-261, 2016.
- [244] M. Teichmann, A. Araujo, M. Zhu and J. Sim, "Detect-to-retrieve: Efficient regional aggregation for image search," Long Beach, 2019.
- [245] Googleapis, "Googleapis," 2021. [Online]. Available: <https://storage.googleapis.com/openimages/web/index.html>. [Accessed 1 May 2021].
- [246] Stanford, "Stanford," 2021. [Online]. Available: <http://ufldl.stanford.edu/housenumbers/>. [Accessed 1 May 2021].
- [247] Cocodataset, "Cocodataset," 2021. [Online]. Available: <https://cocodataset.org/#home>. [Accessed 1 May 2021].
- [248] H. Tankovska, "Statista," 26 January 2021. [Online]. Available: <https://www.statista.com/statistics/259477/hours-of-video-uploaded-to-youtube-every-minute/>. [Accessed 1 May 2021].
- [249] S. O'Dea, 2020. [Online]. Available: <https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/>. [Accessed 1 May 2021].
- [250] Y. Cheng, D. Wang, P. Zhou and T. Zhang, "ARXIV," 14 June 2020. [Online]. Available: <https://arxiv.org/pdf/1710.09282.pdf> . [Accessed 1 September 2021].
- [251] J. Mitro, "github," 2016. [Online]. Available: <https://github.com/kirk86/ImageRetrieval>. [Accessed 1 October 2019].
- [252] M. Haahr, "random," 2022. [Online]. Available: <https://www.random.org/randomness/>. [Accessed 11 May 2022].
- [253] Brunel University London, "brunel," 2018. [Online]. Available: <https://blackboard.brunel.ac.uk/>. [Accessed 1 May 2018].
- [254] C. Simon, "github," 2022. [Online]. Available: <https://simonensemble.github.io/2018-10/orthogonal-procrustes/>. [Accessed 26 May 2022].
- [255] C. Zhang, P. Benz, D. Mureja Argaw, S. Lee, J. Kim, F. Rameau, J. Bazin and I. So Kweon, "ARXIV," 2020. [Online]. Available: <https://arxiv.org/abs/2010.12496>. [Accessed 31 May 2022].
- [256] G. Pleiss, D. Chen, G. Huang, T. Li, L. van der Maaten and K. Q. Weinberger, "ARXIV," 2017. [Online]. Available: <https://arxiv.org/pdf/1707.06990.pdf%EB%A5%BC>. [Accessed 31 May 2022].

## Prior Dissemination

- **Peer-reviewed journal publications**

1. C. Ammatmanee and L. Gan, "A ten-year literature review of content-based image retrieval (CBIR) studies in the tourism industry," *The Electronic Library*, vol. 39, no. 2, pp.225-238, 2021.
2. C. Ammatmanee and L. Gan, "Transfer learning for hostel image classification," *Data Technologies and Applications Journal*, vol. 56, no. 1, pp.44-59, 2022.

- **Peer-reviewed conference proceeding**

1. C. Ammatmanee, L. Gan and L. Hongqing, "Fast binary embedding of deep learning image features using Golay-Hadamard matrices," in *The IEEE International Conference on Multimedia and Expo 2021(Virtual)*, Shenzhen, 2021.

- **Poster conferences**

1. C. Ammatmanee, "Hostel Lookbook: Content-Based Image Retrieval for Hostel Market," in *Brunel Research Student Conference*, London, 2019.
2. C. Ammatmanee, "Hostel Lookbook: Content-Based Image Retrieval for Hostel Market," in *Brunel ECE PhD Symposium*, London, 2019.

- **Oral presentations**

1. C. Ammatmanee, "Hostel Lookbook: Content-Based Image Retrieval for Hostel Market," in *Brunel Three-minute thesis competition (College heat)*, London, 2019.
2. C. Ammatmanee, "Hostel Lookbook: Content-Based Image Retrieval for Hostel Market," in *Brunel Three-minute thesis competition(University final)*, London, 2019.