Sketch-based Retrieval of Images and 3D Shapes

Yu Xia

A thesis submitted in partial fulfilment of the requirements

of Bournemouth University for the degree of

Doctor of Philosophy

October 26, 2021

Copyright statement

Abstract

With the ubiquitous proliferation of touch screens in consumer electronics such as mobile phones and tablet PCs, the demand of consumers for more convenient product search methods is rising. Since sketching meets this need as a form to better express visual intentions, sketch-based retrieval techniques get increasing attention in the computer vision community. Currently, sketch-based retrieval techniques mainly focus on image and 3D shape retrieval, and some issues such as single- and multi-colour sketch based image retrieval and 3D shape retrieval able to deal with a big domain discrepancy between 2D sketches and 3D shapes have not been well investigated. This thesis will address these issues.

For the image retrieval, a single-colour sketch based image retrieval (SCSBIR) approach using RGB and HSV colour features is investigated. Previous methods only consider black-and-white sketches and ignore colour matching between sketches and images, which induce a low retrieval precision. To address this problem, the SCSBIR approach is proposed to consider both shape matching and colour matching with a novel ranking method. Since existing methods cannot effectively distinguish images of the same type but different colours, SCSBIR is further extended to multi-colour sketch based image retrieval (MCSBIR) using a two-stage network architecture, in which a new feature embedding for explicably describing the shape and colour information is proposed and a triplet loss function based on a new Euclidean distance, which separates the shape and colour features, is developed. For the 3D shape retrieval, a teacher-student guided and sketch-based 3D shape retrieval (TSS3DSR) approach is presented to tackle the big domain discrepancy between 2D sketches and 3D shapes. The pre-learned semantic features of 3D shapes are first extracted from the teacher network and then used to guide the feature learning of 2D sketches in the student network.

A series of experiments have been carried out to demonstrate the effectiveness of the proposed methods in both the image and 3D shape retrieval. A user interface is also developed to facilitate practical applications of the developed colour sketch-based image retrieval and sketch-based 3D shape retrieval in this thesis.

Acknowledgements

I would like to express my deepest gratitude to my supervisors Prof. Lihua You and Prof. Jian J. Zhang for their continued guidance and advice. Their support and help in my research work and life make me feel warm. I am really grateful to meet such supervisors as them, who are always patient to listen to my thoughts and ideas. This work is the result of our countless meetings, especially with Prof. You, who gives me many concerns and encouragements throughout these four years.

I would like to express my sincere thanks to Prof. Xiaosong Yang, who helps me a lot during my research work. I really appreciate his willingness to discuss new ideas with me and his patient advice.

During my four years study in NCCA, I have built some friendships that I cherish very much. Thanks to Nan Xiang, Kun Qian, Tao Jiang, Li Wang, Yanran Li, Yinyu Nie and Ruibin Wang for their help and support in my research. Thanks to Bing Yang, Mengqing Huang, Ouwen Li and other friends for the amazing time.

I am grateful to my family and my husband for accompanying me.

Declaration

This thesis has been created by myself and has not been submitted in any previous application for any degree. The work in this thesis has been undertaken by myself except where otherwise stated.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Retrieval is an important data analytics problem in online shopping and design industries. In the online shopping industry, a report by the Centre for Retail Research indicates that Britain had the largest online retail sector in Western Europe, whose total online retail sales reached £99.308 billion in 2020 [Centre for Retail Research 2020]. It shows that online shopping has already become a new way of shopping in our daily life. To keep and improve the prosperity of online shopping, a good retrieval strategy is very important in helping not only sellers to sell more goods but also customers to find what they want. In the design industry, with the development of the Internet and storage technologies, designers no longer need to start the product design from scratch, and they can directly search the whole or parts of existing products through support retrieval tools to generate new designs, which greatly shortens the design cycle. Therefore, how to allow users to accurately retrieve desired targets plays an important role in online shopping and design industries.

Existing retrieval methods rely on the texts or exemplar images, which have many limitations including low accuracy and efficiency, poor personalized demand and unfriendly to vulnerable groups such as people with low intelligence and poor description skills. With the ubiquitous proliferation of touch screens in consumer electronics such as mobile phones and tablet PCs, the sketch-based retrieval is a solution to the above limitations and provides a more convenient and user-friendly search method based on data-driven analysis. It can also perform accurate retrieval results based on the user's input since the sketch can better express visual intentions and has more detailed information. Figure 1.1 illustrates an example of using the free-hand sketch for image retrieval compared with using the text. When users want to search for their desired female heels online, it is difficult to describe the details and style of the heels by words. The retrieval results are usually category-level, which roughly include all shoes from the category of female heels. In contrast, a sketch of the heels can be easily drawn through touch screens, which can describe the details and style of the desired heels. The retrieval results by the sketch are instance-level, which only include the most similar heels as well as the right one.

Thanks to the great progress of the computer performance and the deep learning approach, the sketch-based retrieval techniques get increasing attention as new computer vision problems. Currently, many researchers have studied several retrieval problems based on the sketch, including sketch-based image retrieval [Eitz et al. 2010b; Hu & Collomosse 2013; Yu et al. 2016; Bui & Collomosse 2015] and sketch-based 3D shape retrieval [Wang et al. 2015; Qi et al. 2018; Daras & Axenopoulos 2010; Eitz et al. 2012b]. However, some issues such as single- and multi-colour sketch based image retrieval, 3D shape retrieval able to deal with

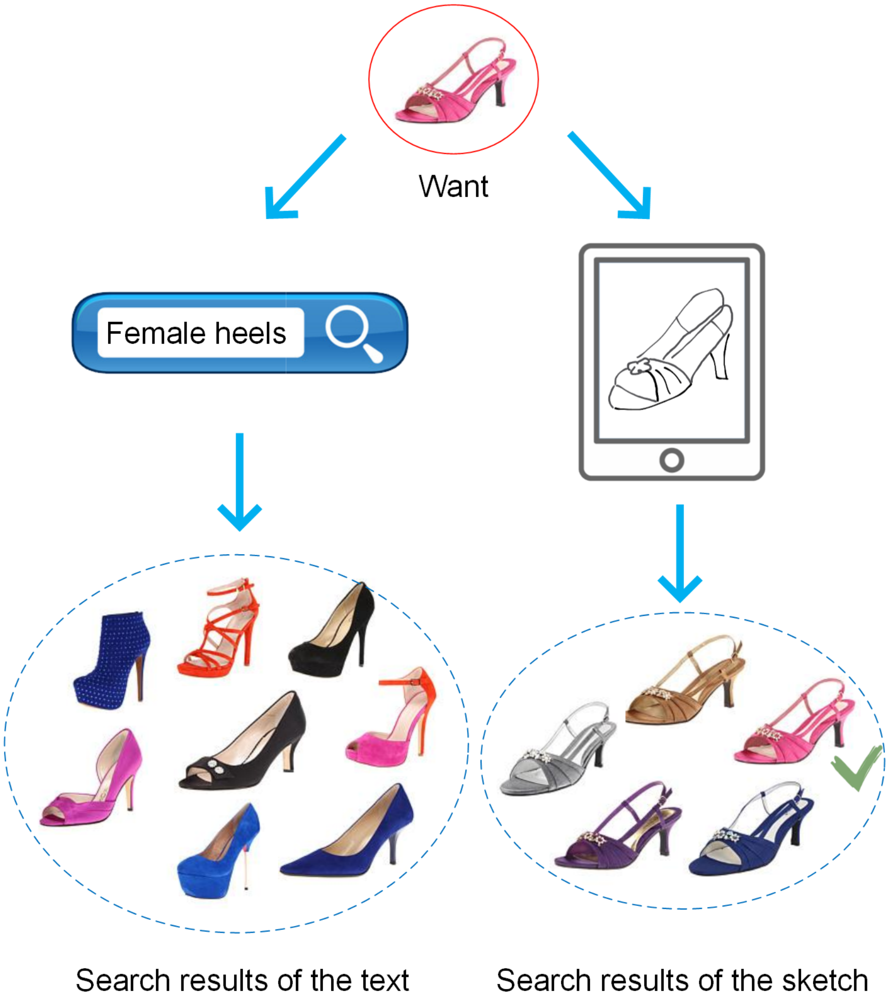


Figure 1.1: *An example of using the free-hand sketch for image retrieval compared with using the text.*

a big domain discrepancy between 2D sketches and 3D shapes, and a user-friendly interface to facilitate sketch-based retrieval have not been well investigated. This thesis will address these issues.

1.1 Colour sketch-based image retrieval

In the field of sketch-based image retrieval, fine-grained matching between sketches and retrieved images attracts an increased attention. Currently, researchers who study fine-grained sketch-based image retrieval use black-and-white sketches as input, but such input is divorced from practical applications because objects in real life are coloured. Inspired by the work of [Bui & Collomosse 2015], one of the aims of this thesis is to solve the problem of fine-grained image retrieval based on colour sketch, and make the retrieval results consider both shape detail matching and accurate colour matching. Solving this problem is particularly important in commercial applications such as searching a specific item on an online shopping platform by colour finger-sketching using a touchscreen device. For example, when users search a red female heeled boot using the black-and-white sketch-based image retrieval method, the black-and-white sketch may find the image of a black female heeled boot because their shapes are matched only during the retrieval process. If there is a colour matching in the process, the red female heeled boot can be found using the red sketch, as shown in Figure 1.2. Therefore, the study of colour sketch-based image retrieval has a more important commercial application value.

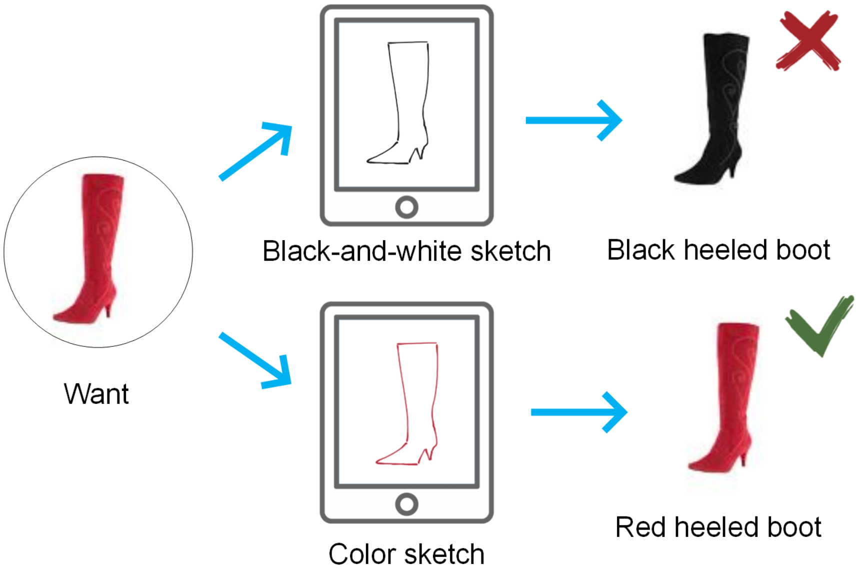


Figure 1.2: *An example of image retrieval using a black-and-white sketch and a colour sketch.*

Achieving colour sketch-based image retrieval has many challenges. First, there is no publicly available dataset of colour sketches. The existing publicly available sketch datasets consist of black-and-white sketches, and cannot meet the needs of colour sketch-based image retrieval. Thus, creating a new colour sketch dataset is necessary. Second, since colour sketches are very different from images in appearance, the colour sketch-based image retrieval is a cross-domain retrieval problem, as shown in Figure 1.3. It is required to find a joint feature embedding space to narrow the large domain gap. Third, human sketches are very abstract and somewhat distorted. Unlike image edge extraction, the sketch does not completely fit the object contour in the image, which increases the difficulty of retrieval. Fourth, it is difficult to obtain retrieval results with correct shapes and colours effectively based on colour sketches. How to determine which result is more matched with the query colour sketch is a big problem. An advanced sorting algorithm should be developed to make the final retrieval results optimal in both shape matching and colour matching. Last but not the least, the retrieval method should distinguish images of the same type but different colours, which can greatly improve the retrieval accuracy.

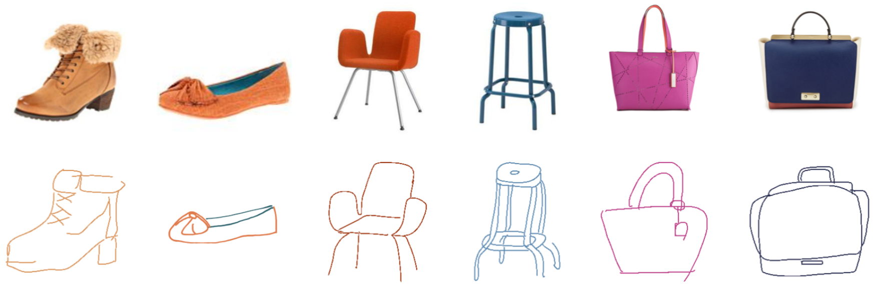


Figure 1.3: *Some examples of colour sketches and corresponding images.*

In this thesis, in order to tackle the problems of colour sketch-based image retrieval, a single-colour sketch based image retrieval (SCSBIR) method is first proposed and then a multi-colour sketch based image retrieval (MCSBIR) method is developed.

SCSBIR A novel SCSBIR method is proposed based on the multi-branch deep convolutional neural network (CNN). The network consists of three identical branches, one of which takes colour sketches as input and the other two take images as input during training. With the network, not only the objects with fine-grained similarity to the sketch are obtained, but also the similarity of colour is considered. For achieving the optimal performance of the neural networks, a lot of training data is needed. Since the deep FG-SBIR model [Song et al. 2017] provides a suitable CNN foundation for black-and-white sketch-based image retrieval, a pre-training model is built based on the deep FG-SBIR model and a dataset of single-colour sketch-image pairs for SCSBIR is created based on the Shoe and Chair Datasets from [Yu et al. 2016] and the Handbag Dataset from [Song et al. 2017].

MCSBIR Since most retrieved objects contain multi-colours and the single-colour cannot completely represent their colour information, a MCSBIR method is further proposed to tackle this problem. First, a MCSBIR dataset is created based on UT Zappos50K [Yu & Grauman 2014] and a new feature embedding is designed to clearly and explicably describe the shape and colour information within a single feature vector. Then, a triplet loss function based on a new Euclidean distance is developed, which separates the shape and colour features, and a two-stage network architecture is designed to learn the proposed feature embedding. In order to demonstrate the effectiveness of the proposed method, two baselines are also designed to compare with the proposed MCSBIR method and the influence of different hyper-parameters and stages on the retrieval performance is further analyzed.

1.2 Sketch-based 3D shape retrieval

The virtual 3D shape plays an increasingly important role in our daily lives due to the rapid development of digitalization techniques, such as visual effects, medical imaging and 3D printing. How to retrieve a desired 3D shape among a great number of 3D shapes is a popular research topic in many years [Chen et al. 2003; Shih et al. 2007; Shao et al. 2011; Li et al. 2014a]. Compared to using texts as queries, sketches can more intuitively describe 3D shapes and are also convenient for humans to use. In design industry, designer can draw sketches to retrieve desired 3D shapes for fast 3D modeling and scene generation, as shown in Figure 1.4, which can speed up their design process and help their creation work. Therefore, sketch-based 3D shape retrieval has attracted considerable attention in the community of computer vision and graphics [Li et al. 2013a, 2014b].

The main challenge for sketch-based 3D shape retrieval is the big domain discrepancies [Qi et al. 2018]. First, sketches are represented in a 2D space while 3D shapes are embodied in a 3D space, so their heterogenous data structures make it extremely difficult to directly retrieve 3D shapes from a query sketch. Second, sketches are abstract free-hand drawings, which usually consist of several simple lines and contain very limited information. Conversely, 3D shapes are surface-represented geometric objects and have many details of their shape characteristics. Third, sketches are presented with only one view of 3D shapes, and it is very hard to find the best or most similar view of 3D shapes according

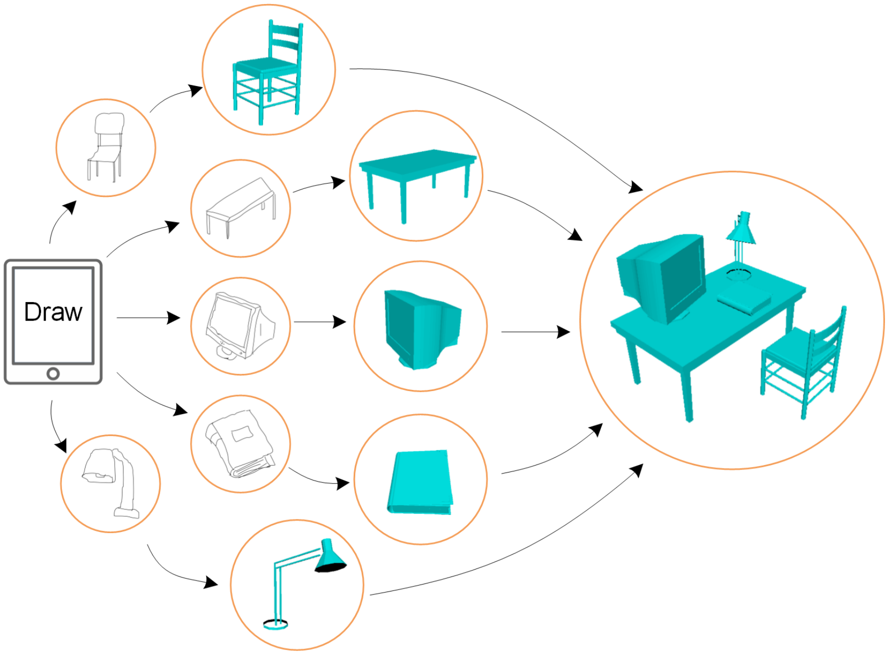


Figure 1.4: *A 3D scene generation with objects retrieved using sketches*

to query sketches. Figure 1.5 gives some examples of sketches and corresponding 3D shapes from the same class, and shows the large domain gap between them. In order to tackle the aforementioned challenge of sketch-based 3D shape retrieval, a variety of research efforts have been dedicated to this task, and their main purpose is to improve the retrieval accuracy. There are mainly two ways to achieve the accuracy improvement: 1) learning robust features representations for both sketches and 3D shapes [Chen & Fang 2018; Xie et al. 2017; Tasse & Dodgson 2016], and 2) developing effective ranking or distance metrics between sketches

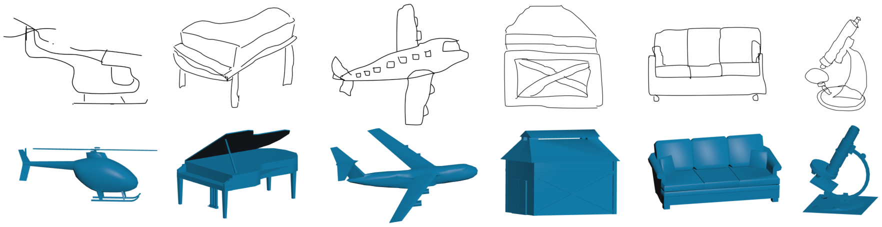


Figure 1.5: *Some examples of sketches and corresponding 3D shapes.*

and 3D shapes [Qi et al. 2018; Wang et al. 2015; Dai et al. 2017]. Due to the great success of deep CNNs applied in the image feature extraction in recent years, all state-of-the-art methods have used deep metric learning for sketch-based 3D shape retrieval and achieved a better retrieval accuracy compared with traditional methods [Chen et al. 2019]. However, these studies have two weaknesses. First, they address the domain discrepancy problem by mapping sketches and 3D shapes into a joint feature embedding space, where the similarity is measured using a shared loss function. It is difficult to effectively reduce the domain discrepancy because sketches and 3D shapes cannot be aligned perfectly within the same embedding space. Second, they have two different network structures to extract features of sketches and 3D shapes, respectively, and the parameters of the two networks are unshared and updated simultaneously during the training process, which leads to a high computational cost.

In this thesis, a novel semantic similarity metric learning method named as teacher-student guided and sketch-based 3D shape retrieval (TSS3DSR) is proposed to overcome the above-mentioned disadvantages of recent studies. Note that the aim of sketch-based 3D shape retrieval is to find 3D shapes belonging to the class labels of query sketches, so their label spaces are shared and can be used as a semantic embedding space. In such a semantic space, sketches and 3D shapes are aligned perfectly [Qi et al. 2018]. Inspired by the knowledge distillation technique, which uses a large teacher network to guide a small student network [Hinton et al. 2015], a teacher-student strategy is adopted to obtain efficient networks for learning semantic similarity between sketches and 3D shapes. It can not only reduce the computational burden but also make the semantic features alignment easier. In the proposed TSS3DSR method, the proposed metric learning network consists of a teacher network and a student network. The teacher network is a pre-trained classification network based on MVCNN [Su et al. 2015] to extract the semantic features of 3D shapes and the student network is a transfer network based on ResNet-50 [He et al. 2016] to learn the semantic features of sketches. The transfer network is trained by the guide of a new similarity loss for optimizing the semantic feature distance between sketches and 3D shapes.

In order to facilitate practical applications of the developed colour sketch-based image retrieval and sketch-based 3D shape retrieval, a user interface, which integrates sketch-based retrieval functions for images and 3D shapes, is developed. Users can freely choose different retrieval modes, use a brush with different colours and sizes for sketching, and search similar images or 3D shapes. The user interface not only acts as a demo to visually present the research on sketch-based retrieval, but also provides an effective tool to release the potential of the developed techniques for future commercial applications.

1.3 Aims and objectives

The research aim of this thesis is to develop a sketch-based retrieval system for images and 3D shapes. For the sketch-based image retrieval, the colour sketch is focused on making the retrieval results consider both shape matching and colour matching as well as improve the retrieval performance. For the sketch-based 3D shape retrieval, the alignment of semantic features is focused on solving the cross-domain retrieval problem. According to the research aims, there are three primary research questions that need to be addressed.

- **How to match the shape and colour information between sketches and images**? The stroke pixels of a black-and-white sketch indicate the shape information, which can be recognized and mapped to hand-designed or deep features. However, the stroke pixels of a colour sketch contain not only shape information but also colour information. It is difficult to compare the similarities between colour sketches and images.

- **How to construct the embedding features of multi-colour sketches**? Since an image usually contains different colours, a multi-colour sketch can better describe the image. However, the major difficulty is to construct the embedding feature of the multi-colour information as well as shape information. In addition, in different retrieval preferences, the focus may be inclined to the shape or the colour. How to reflect this characteristic in the feature embedding is a challenge.

- **How to reduce cross-domain discrepancies between sketches and 3D shapes**? Since sketches and 3D shapes are represented in different dimensional spaces, it is unprocurable to directly compare their similarity. The primary difficulty is to find a feature embedding space for both sketches and 3D shapes.

To answer the above questions, this thesis aims to achieve three main objectives:

- The first objective is to make the retrieval results consider both the shape detail matching and the accurate colour matching. There are two ways to achieve this objective. The first is to separate the matching processes of the shape and colour, which is discussed in Chapter 3, and the second is to match the shapes and colours simultaneously, which is discussed in Chapter 4.

- The second objective is to construct embedding features of multi-colour sketches. This objective focuses on finding a single feature that can describe the shape and colour information together. In particular, the constructed feature should be explicable, i. e., the colour and shape information are unmixed, so that the similarity between sketches and images can be measured. This objective is discussed in Chapter 4.

- The third objective is to find a feature embedding for both sketches and 3D shapes and reduce their cross-domain discrepancies. Since the sketches and 3D shapes share the same semantic information, the key is to find 3D shapes belonging to the same class labels of query sketches, which is discussed in Chapter 5.

1.4 Contributions

The main contributions of the work in colour sketch-based image retrieval are listed as follows:

For SCSBIR:

- A single-colour sketch-image dataset is created, which contains three categories, i. e., 419 sketch-image pairs of shoes, 297 sketch-image pairs of chairs and 568 sketch-image pairs of handbags.

- A dominant colour extraction method is proposed to detect the most attractive colour of an image and help to colour the black-

and-white sketches.

- A deep learning approach is developed to achieve image retrieval based on single-colour sketch, and the generalization of this approach is verified in different categories.
- Two colour similarity comparison methods, in RGB colour space with Hellinger distance and HSV colour space with Bhattacharyya distance, are proposed to rank retrieval images after the shape matching process.

For MCSBIR:

- A multi-colour sketch-image dataset containing 232 sketch-image pairs of shoes is created by using the k-means clustering algorithm to extract a set of quantized colours from images and generate the multi-colour sketches.
- A novel feature embedding for explicably describing the shape and colour information within a single feature vector is proposed.
- A triplet loss function based on a new Euclidean distance, which separates the shape and colour features, is developed.
- A two-stage network architecture is designed, which consists of a classification stage and a retrieval stage.

The main contributions of the work in sketch-based 3D shape retrieval are listed as follows:

- A metric learning network using the teacher-student strategy is proposed to conduct sketch-based 3D shape retrieval in a joint semantic embedding space.

- A similarity loss function is developed to optimize the semantic feature distance between sketches and 3D shapes.

- Several experiments are carried out on a large benchmark dataset of sketch-based 3D shape retrieval and show that the proposed TSS3DSR method outperforms other state-of-the-art methods.

1.5 List of publications

Relevant publications

[1] **Xia, Y.**, Wang, S., You, L. and Zhang, J., 2021, June. Semantic Similarity Metric Learning for Sketch-Based 3D Shape Retrieval. In International Conference on Computational Science (pp. 59-69). Springer, Cham.

[2] **Xia, Y.**, Wang, S., Li, Y., You, L., Yang, X. and Zhang, J.J., 2020. Single Color Sketch-Based Image Retrieval in HSV Color Space. In Transactions on Computational Science XXXVII (pp. 77-90). Springer, Berlin, Heidelberg.

[3] **Xia, Y.**, Wang, S., Li, Y., You, L., Yang, X. and Zhang, J.J., 2019, June. Fine-grained color sketch-based image retrieval. In Computer Graphics International Conference (pp. 424-430). Springer, Cham.

[4] **Xia, Y.**, Wang, S., You, L., Yang, X. and Zhang, J.J., A explicable feature embedding for multi-color sketch-based image retrieval, to be submitted.

Co-authored publications

[5] Wang, S., Xiang, N., **Xia, Y.**, You, L. and Zhang, J., 2021. Real-

time surface manipulation with C^1 continuity through simple and efficient physics-based deformations. The Visual Computer, pp.1-13.

[6] Wang, S., Wang, R., **Xia, Y.**, Sun, Z., You, L. and Zhang, J., 2021. Multi-objective aerodynamic optimization of high-speed train heads based on the PDE parametric modeling. Structural and Multidisciplinary Optimization, pp.1-20.

[7] Wang, S., **Xia, Y.**, You, L. and Zhang, J., 2020. Reconstruction of Curve Networks from Unorganized Spatial Points. Journal of Universal Computer Science (J. UCS), 26(9), pp.1265-1280.

[8] Wang, S., **Xia, Y.**, Wang, R., You, L. and Zhang, J., 2019. Optimal NURBS conversion of PDE surface-represented high-speed train heads. Optimization and Engineering, 20(3), pp.907-928.

Contributions in co-authored publications For paper [5], the author has conducted the experiments and tested the user interface. For paper [6], the author has designed the whole optimization flow of high-speed train heads. For paper [7], the author has contributed to the 3D curve network model design and created a test dataset. For paper [8], the author has contributed to the visualization of PDE surfaces and NURBS surfaces.

1.6 Structure of the following chapters

The following part of this thesis contains six more chapters:

- Chapter 2 reviews related research topics, including sketch recognition, sketch-based image retrieval and sketch-based 3D shape retrieval.

- Chapter 3 presents a SCSBIR method to retrieve images by single-colour sketches with a ranking method combined with the shape similarity matching and colour similarity matching, which improves the retrieval accuracy.
- Chapter 4 presents a MCSBIR method to retrieve images by multi-colour sketches, which is not only able to distinguish the images with the same shape but different colours, but also has higher retrieval accuracy compared with baselines.
- Chapter 5 presents a TSS3DSR method to retrieve 3D shapes by sketches and compares it with state-of-the-art methods using different evaluation metrics.
- Chapter 6 presents a user interface integrating the proposed MCSBIR and TSS3DSR methods to demonstrate the potential application of sketch-based retrieval.
- Chapter 7 concludes the thesis and discusses future work.

Chapter 2

Literature Review

This thesis mainly investigates sketch-based retrieval in two aspects, i. e., colour sketch-based image retrieval and sketch-based 3D shape retrieval. Therefore, this chapter reviews the most related works to these two parts. Since modern sketch-based retrieval is closely related to sketch recognition [Radenovic et al. 2018], an overview of sketch recognition will be presented in Section 2.1, and then the methods of sketch-based image and 3D shape retrieval will be reviewed in Sections 2.2and 2.3, respectively.

2.1 Sketch recognition

Sketch recognition is inseparable from sketch-based retrieval because both of them are looking for a feature embedding which can exactly describe the sketch features. Moreover, their benchmark dataset overlaps. The sketch recognition dataset provides free-hand sketch resources for the establishment of the sketch-based retrieval dataset. Therefore, it is necessary to first review the related works of sketch recognition.

The purpose of sketch recognition is to predict an appropriate classification for the input sketch. A demonstration of sketch recognition using TU-Berlin dataset [Eitz et al. 2012a] is shown in Figure 2.1. It is one of the most fundamental studies in the field of computer vision. According to the different classification algorithms used in sketch recognition, existing studies can be roughly divided into two parts, i. e., sketch recognition using hand-designed features and sketch recognition using deep features.

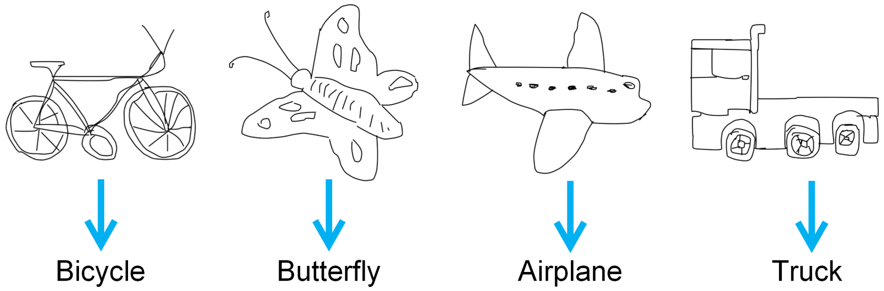


Figure 2.1: *A demonstration of sketch recognition using TU-Berlin dataset [Eitz et al. 2012a].*

2.1.1 Sketch recognition using hand-designed features

Sketch recognition was studied by Herot [1976] in the early stage. An interactive system for graphical input was reported, which involved the user's interactive design in the recognition process. In their experiments, the sketching was proved to be a viable medium for human-computer interaction. Sezgin et al. [2007] built a system, which could recognize simple freehand sketches consisting of straight lines and curves. This system allowed a single stroke to draw arbitrary shapes and detected feature points automatically. It did not need users to switch modes when drawing different geometric object classes, which gave the user a more natural feeling during the sketching process. After that, various algorithms were proposed in order to achieve the recognition function of more complex sketches.

LaViola Jr & Zeleznik [2006] built a system named MathPad^2 to implement rapid visualization of mathematical formulations. A gestural user interface was designed to associate handwritten mathematical expressions with free-hand diagrams, which supported some computational functions, such as graphing, solving, simplifying and factoring, on recognized mathematical expressions. Ouyang & Davis [2011] developed a system named ChemInk to recognize hand-drawn chemical diagrams in real time, which addressed graphics and text to generate a complete molecular structure. The recognition accuracy of this system achieved 97.4% on a chemical diagrams dataset. A learning-based corner detection approach was also presented which achieved over 99% accuracy in the chemical domain. However, these studies have some limitations. They are only applicable to some certain tasks and do not work well for identifying other types of sketches.

Since Eitz et al. [2012a] created the TU-Berlin dataset which is a benchmark for sketch recognition, the sketch recognition community has been developed rapidly in recent years. TU-Berlin dataset is the first large scale dataset of human freehand sketches, which is exhaustive, recognizable and specific. It has a total of 20,000 sketches containing 250 categories with 80 sketches for each category. With the TU-Berlin dataset, a bag-of-features sketch representation [Sivic & Zisserman 2003] and multi-class support vector machines Schölkopf et al. [2002] were employed by Eitz et al. [2012a] to classify sketches. The accuracy of computer recognition was 56%, while the accuracy of human performance was 73.1%. Since then, a lot of research studies focused on trying to beat the human performance on the TU-Berlin dataset.

Li et al. [2013b] presented an ensemble matching method based on star graphs, which can encode the geometrical structures information of sketches, including holistic structures and local features. Extensive comparative experiments were carried out using the TU-Berlin dataset and showed that the star graphs based approach was superior to the method used by Eitz et al. [2012a] in sketch recognition. They further studied the structured representation of sketches and proposed a multi-kernel feature learning framework to fuse several features of sketches and overcome the visual sparse problem [Li et al. 2015]. The performance of this method on the TU-Berlin dataset is 65.81%. In the meanwhile, Schneider & Tuytelaars [2014] introduced a Fisher vectors based sketch recognition approach, which improved the recognition performance to 68.9% on TU-Berlin dataset. Overall, these sketch recognition methods using hand-designed features fail to meet human performance.

2.1.2 Sketch recognition using deep features

With the development of deep learning approaches, many deep models have been proposed in the field of sketch recognition. Yu et al. [2015] proposed Sketch-a-Net, the first deep neural network dedicated to sketch recognition. This network fused multi-scale networks via joint Bayesian fusion [Chen et al. 2012], which was used to obtain the abstraction of sketch and the sequential ordering of strokes. The recognition performance of Sketch-a-Net on the benchmark TU-Berlin dataset is 74.9%, which surpasses human performance for the first time. After that, Yu et al. [2017] extended their previous work and proposed two data argumentation strategies to increase the volume and diversity of sketches in the training set. The improved Sketch-a-Net had a better performance than the original Sketch-a-Net. Sarvadevabhatla & Babu [2015] also explored the application of deep features for freehand sketch recognition and proposed a deep feature based framework by using the ImageNet CNN [Krizhevsky et al. 2012] and LeNet CNN [LeCun et al. 1998; Jia et al. 2014]. In the further study, [Sarvadevabhatla & Kundu 2016] proposed a deep recurrent neural network, which considered the inherently sequential and cumulative nature when people draw sketches in a natural state. Deep features obtained by this network contained long-term sequential and structural regularities in stroke data. Zhang et al. [2016] proposed a novel deep convolutional neural network SketchNet to carry out the sketch recognition task and took a triplet as input, which consists of a sketch, a positive image and a negative image. SketchNet contains three sub-networks, i. e., R-NET for extracting image features, S-NET for extracting sketch features and C-NET for discovering the common structures of sketches and images. In the field of sketch recognition, abundant research studies based on deep features indicate that their recognition performance is far better than that of using traditional hand-designed features.

In summary, sketch recognition can be regarded as a base for sketch-based image and 3D shape retrievals. The sketch recognition using hand-designed features has a low recognition performance and is gradually replaced by deep features since deep networks perform better in recognizing sketches. Therefore, in this thesis, deep features are used to describe the sketches.

2.2 Sketch-based image retrieval

The techniques on sketch-based image retrieval to be proposed in this thesis are related to category-level sketch-based image retrieval (Section 2.2.1), fine-grained sketch-based image retrieval (Section 2.2.2) and colour sketch-based image retrieval (Section 2.2.3). In this section, the most related work in these three fields is briefly reviewed.

2.2.1 Category-level sketch-based image retrieval

Category-level sketch-based image retrieval is to retrieve images from the same category as the query sketch. Some typical examples are illustrated in Figure 2.2. Many existing researches focused on category-level sketch-based image retrieval and employed hand-designed features or deep features to represent sketches and images. In the early stage, hand-designed features are frequently used to solve this problem. Eitz et al. [2010a] presented an interactive system for sketch-based image retrieval and created 43 sketch-image pairs to evaluate and compare the retrieval performance of 27 descriptor variants. They also developed bag-of-features descriptors (BOF) and created a new dataset for category-level sketch-based image retrieval including 31 sketches and 40 images associated with each sketch [Eitz et al. 2010b]. Based on the BOF codebook, Hu & Collomosse [2013]; Hu et al. [2010, 2011] introduced Gradient Field HoG (GF-HOG) as a depiction invariant image descriptor to improve retrieval accuracy. Novel processing schemes for large-scale databases were also proposed in [Cao et al. 2011, 2010] to calculate the similarity between a query sketch and images. The limitation of hand-designed features is that the subtle detail information cannot be well noticed.

Thanks to deep learning technology and the release of large scale free-

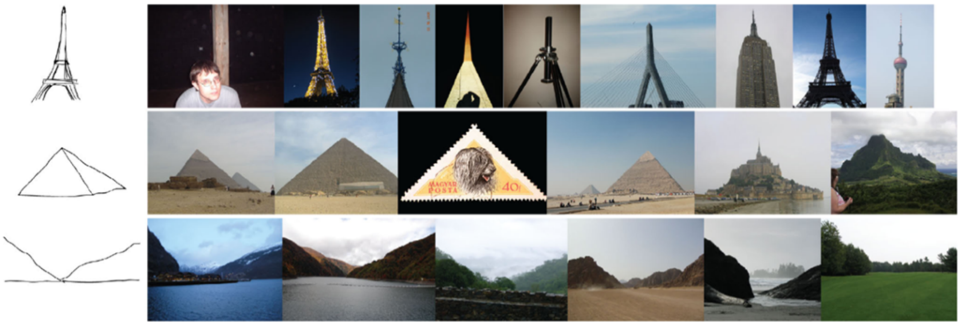


Figure 2.2: *Some examples of category-level sketch-based image retrieval [Eitz et al. 2010b]. ©[2010] IEEE.*

hand sketch datasets, such as TU-Berlin dataset [Eitz et al. 2012a] and Sketchy database [Sangkloy et al. 2016], deep features have been used to solve the problem of category-level sketch-based image retrieval in recent years. Qi et al. [2016] proposed a Siamese convolutional neural network for sketch-based image retrieval, which can reduce the Euclidean distance between the output feature vectors of an input sketch and a similar image and increase the Euclidean distance between an input sketch and an irrelevant image. Subsequently, Bui et al. [2017] presented a triplet convolutional neural network whose weights were half-shared for sketch-based image retrieval. The network consisted of an anchor network which took a sketch as input, a positive network which took an image from the same category as input and a negative network which took an image from any other category as input. With this network, they employed a modified triplet loss function to minimize the distance between the sketch and the same-category image and maximize the distance between the sketch and the different-category image. Seddati et al. [2017] proposed a quadruplet network for sketch-based image retrieval, which enabled the output features to contain more global and local information. The input of the quadruplet network consisted of four parts, i. e., sketches, similar images, dissimilar images from the same category and dissimilar images from a different category. These methods, however, belong to the category-level, which cannot distinguish fine-grained subtle differences well between retrieved images in the same category.

2.2.2 Fine-grained sketch-based image retrieval

Fine-grained sketch-based image retrieval requires instance-level search precision, which means that an image is considered to be the match of the query sketch only when they are similar in shape, pose, viewpoint, iconic pattern, etc. [Yu et al. 2016; Song et al. 2017]. Some examples of fine-grained sketch-based image retrieval are shown in Figure 2.3, which can distinguish the subtle differences of images and further find the correct top 1 match. The concept of fine-grained retrieval was first proposed in [Li et al. 2014c] in 2014 to further leverage the descriptive power of sketches. Some researchers employed deep learning approaches to address the fine-grained sketch-based image retrieval problems. Yu et al. [2016] created a dataset including two categories of shoes and chairs and used a triplet ranking homogeneous network with a triplet ranking loss to train model parameters for shoes and chairs, respectively. Subsequently, Song et al. [2017] improved the method in [Yu et al. 2016] by introducing the shortcut connection architecture of ResNet and the attention modeling, which was mainly applied in the NLP field. They added a new handbag category and proposed a novel loss function named HOLEF, which was an improved version based on the classic triplet ranking loss. Sangkloy et al. [2016] did a similar study on fine-grained sketch-based image retrieval. Their main contribution was a Sketchy database with 125 categories and a triplet ranking heterogeneous network with the classical triplet ranking loss for training model parameters, which can be applied in a multi-category retrieval. Although some previous studies have been done on the fine-grained retrieval, all of them retrieve images based on black-and-white sketches, and there is no precedent considering colour sketches.

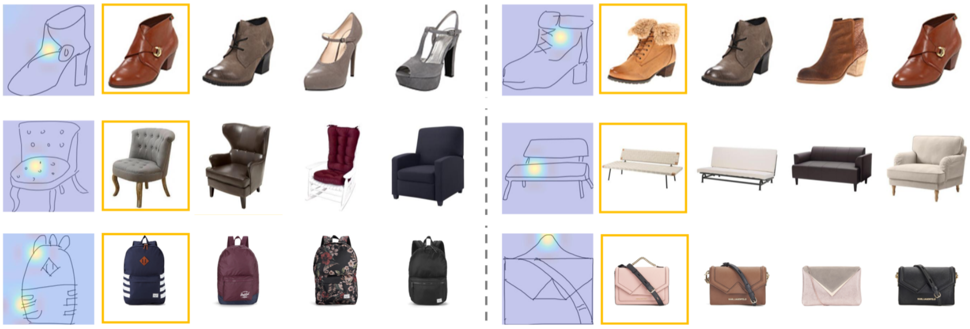


Figure 2.3: *Some examples of fine-grained sketch-based image retrieval [Song et al. 2017]. The correct matches are highlighted in yellow squares. ©[2017] IEEE.*

2.2.3 Colour sketch-based image retrieval

Recent existing colour sketch-based image retrieval methods mainly focus on the extraction and comparison of hand-designed features of colour sketches and images based on gradients [Eitz et al. 2010b; Hu & Collomosse 2013]. These methods have some limitations and cannot preserve the subtle details of sketches and images well. Therefore, their retrieval results cannot meet the requirements of the fine-grained retrieval, such as good matching in the posture, direction and details of objects. Reddy et al. [2014] used the HSV colour space to extract colour features and a gray-level co-occurrence matrix to extract texture features of free-hand colour sketches and images separately. Then, the Euclidean distances of colour and texture features between query sketches and target images were calculated to obtain the similarities. The paper mainly focuses on the category-level search, but ignores the subtle details and directions of the retrieved instances. Moreover, the query sketch is not the abstract hand-drawn colour line sketch, but the photo edge extraction sketch with colour blocks, which is easier to retrieve images compared with hand-drawn sketches. Bui & Collomosse [2015] first used line-art query sketches and presented a gradient field HoG based on mathematical formulas for colour sketch-based image retrieval. They converted colour sketches and images into GF-HoG descriptors separately and then made subsequent comparisons.

Up to now, deep learning methods were rarely applied in colour sketch-based image retrieval. Cheng et al. [2016] used a CNN to address a pedestrian colour naming problem. Their work is to obtain accurate colour descriptions of real world pedestrian images without paying attention to the extraction of subtle detailed features of sketches or images, which is quite different from what to be solved in this thesis. Fuentes & Saavedra [2021] proposed a quadruplet-based convnet architecture to deal with a colour sketch-based image retrieval problem. In order to represent shape and colour information in one model, they used a quadruplet, consisting of a colour sketch, a positive image with the same class and colour as the sketch, a neutral image with the same class but a different colour and a negative image with a different class and colour, to generate a feature space that can discriminate different colours. However, most of the sketch data they used are sketches with colour blocks rather than colour lines, and their feature space cannot clearly and explicably describe the shape and colour information.

In summary, the development of sketch-based image retrieval is from the category-level, which retrieves images of the same category to the instance-level, which retrieves the most similar images. With the application of deep learning approaches, the retrieval performance is greatly improved. However, most studies focus on the image retrieval based on black-and-white sketches and a few studies try to deal with colour sketches. There are three main challenges for achieving colour sketch-based image retrieval, i. e., creation of a colour sketch dataset, an accurate colour matching method between sketches and images, and a feature embedding to represent both shape and colour information. All these three challenges will be addressed in this thesis.

2.3 Sketch-based 3D shape retrieval

The proposed TSS3DSR method is related to sketch-based 3D shape retrieval (Section 2.3.1) and a teacher-student strategy in metric learning (Section 2.3.2). In this section, the most related work in these two fields is briefly reviewed.

2.3.1 Retrieval methods

In the early stage, most sketch-based 3D shape retrieval methods relied on the handcrafted features for describing sketches and 3D shapes [Li et al. 2014a, 2013a]. With the rapid growth of CNNs, learning-based methods have been developed in recent years. Wang et al. [2015] used two projection views to characterize 3D shapes, defined a loss function on the within-domain and the cross-domain similarities, and applied a Siamese network to learn a joint embedding space for sketches and 3D shapes. Some typical sketch-based 3D shape retrieval examples are illustrated in Figure 2.4. In order to reduce cross-domain discrepancies between sketches and 3D shapes, Zhu et al. [2016] developed pyramid cross-domain neural networks by cooperating with a hierarchical structure, and they trained a neural network pair for sketches and 3D shapes, respectively, by allocating identical representations at the target layer for instances of the same class. To address the same problem, Chen & Fang [2018] proposed a cross-modality adaptation model for sketch-based 3D shape retrieval. They employed an importance-aware metric learning to learn modality-specific discriminative features and developed a transformation network, which transferred the sketch features into the feature embedding space of 3D shapes, to remove the cross-modality discrepancy between sketches and 3D shapes. Chen et al. [2019] developed a deep sketch-shape hashing framework for sketch-based 3D shape retrieval with a stochastic sampling strategy for 3D shapes and a binary coding strategy for learning discriminative binary codes.

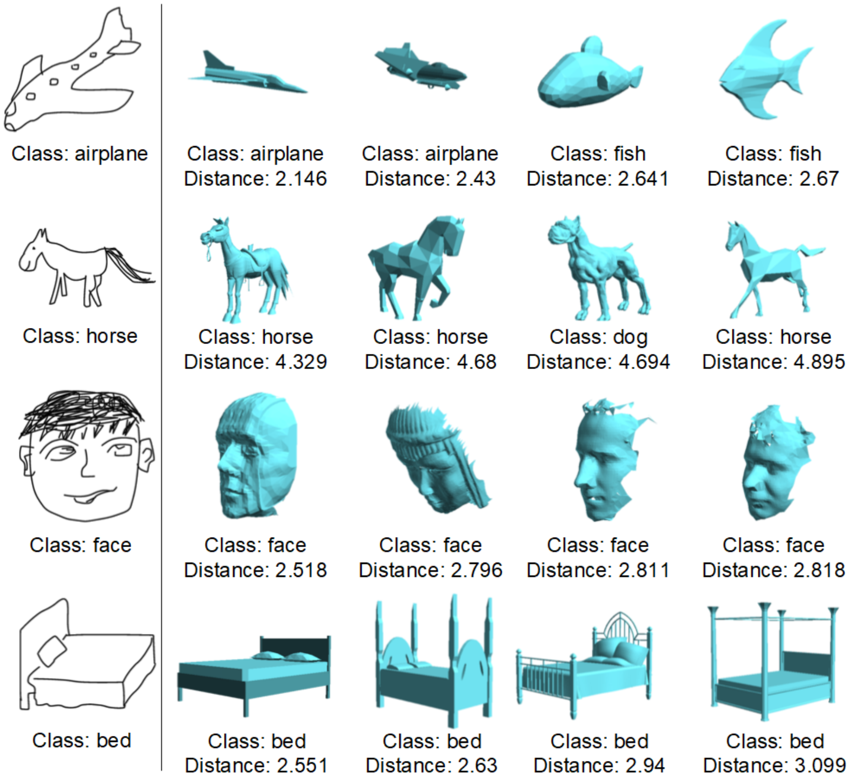


Figure 2.4: *The sketch-based 3D shape retrieval examples [Wang et al. 2015]. The retrieved 3D shape with a smaller distance is more similar to the query sketch.*

Unlike above projection-based methods, Dai et al. [2017] presented a deep correlated metric learning method to mitigate the discrepancy by directly extracting the features of 3D shapes. They extracted the 3D scale-invariant feature transform (3DSIFT) features for 3D shapes and further encoded these features by locality-constrained linear coding (LLC) to get a global shape description. In order to learn a joint semantic embedding space, Qi et al. [2018] developed a deep neural network consisting of heterogeneous branches for the sketch and 3D shape domains, respectively, and they used a PointNet network to extract 3D shape features due to its strong classification performance.

2.3.2 Teacher-student strategy in metric learning

Since Hinton et al. [2015] showed that a complex and powerful teacher model can guide the training of a small student network, which can decrease the inference time and improve its generalization ability, the teacher-student strategy has received attention in the field of metric learning. Chen et al. [2018] proposed cross sample similarities for knowledge transfer in deep metric learning, and modified the classical list-wise rank loss to bridge teacher networks and student networks. Yu et al. [2019] presented a network distillation to compute image embeddings with small networks and developed two loss functions to communicate teacher and student networks. For the sketch-based 3D shape retrieval, Dai & Liang [2020] proposed a cross-modal guidance network by using teacher-student strategy and used pre-learned features of 3D shapes to guide feature learning of 2D sketches.

In summary, compared with image retrieval, sketch-based 3D shape retrieval is more normative because some popular benchmarks such as

SHREC'13 [Li et al. 2013a] and SHREC'14 Li et al. [2014b] are built very early, which give a guide to test different retrieval methods. Therefore, the main purpose of recent studies is to improve a retrieval score in these benchmarks. Similar to image retrieval, the application of deep learning approaches also greatly improves the accuracy of sketch-based 3D shape retrieval. However, most of these retrieval methods have two operative networks, which cause a high computational cost. Besides, since they directly map features into a joint embedding space, it is difficult to effectively reduce the domain discrepancy and minimize between-class similarity as well as maximize within-class similarity. In this thesis, the above limitations will be overcome, and a high-effective method will be developed to retrieve 3D shapes based on sketches.

2.4 Summary

In this chapter, the related works of sketch-based retrieval for images and 3D shapes are reviewed. As discussed in previous sections, existing retrieval methods cannot retrieve images using colour sketches and are difficult to find an applicable feature embedding space of sketches and 3D shapes. In particular, the challenges mentioned in Sections 1.1 and 1.2 are unsolved and the question posed in Section 1.3 cannot be well answered by existing works. In the following chapters, the challenges and questions will be addressed, and the single-colour sketch based image retrieval, multi-colour sketch based image retrieval and sketch-based 3D shape retrieval will be presented in Chapters 3, 4 and 5, respectively.

Chapter 3

Single-colour sketch based image retrieval

Up to now, almost all production retrieval applications of online shopping platforms are based on semantic retrieval. Since semantics cannot accurately describe the detailed shape and colour of a product, users may not easily obtain desired retrieval results by using semantic retrieval. This chapter will aim to solve this problem and propose a retrieval method based on single-colour sketches. The colour sketch has enough appearance information of a retrieval target, which can be used to retrieve products and obtain optimal retrieval results. Since a deep CNN is used to extract features of colour sketches and images in this chapter, a single-colour sketch-image dataset is needed to train the network parameters for better feature representation. In Section 3.1, the method of building the single-colour sketch based image retrieval (SCSBIR) dataset is presented. The proposed deep CNN method and two new colour similarity comparison methods are described in Section 3.2. The experimental results and analysis are presented in Section 3.3, and finally the summary is drawn in Section 3.4.

3.1 SCSBIR dataset

A dominant colour extraction method for creating SCSBIR dataset is developed in Section 3.1.1 and then some experiments and comparisons are carried out to verify the effectiveness of the proposed dominant colour extraction method in Section 3.1.2. The details of SCSBIR dataset are presented in Section 3.1.3.

3.1.1 The extraction method of the dominant colour

In order to create the dataset of single-colour sketches, it is required to first extract the dominant colour from the corresponding image of each sketch, and then add the dominant colour to the black-and-white sketch. Colour extraction from images plays an important role in the field of graphic art and design [Lin & Hanrahan 2013]. Commonly used extraction methods, such as clustering and histogram-based approaches, are usually used to find a set of colours from an image [Ciocca et al. 2019]. In this task, the goal is to extract a dominant colour rather than a set of colours. Although the above methods can be applied to generate a set of colours and choose the colour with the largest proportion in the set as a dominant colour, it is difficult to get accurate results, as shown in Figure 3.3.

Generally, for fashion goods like shoes and bags, a dominant colour should be the most attractive colour of the goods, as shown in Figure 3.1. Thus, in order to obtain a dominant colour, the key is to extract the colour, which is the human's visual focus. In a colour space, the HSV

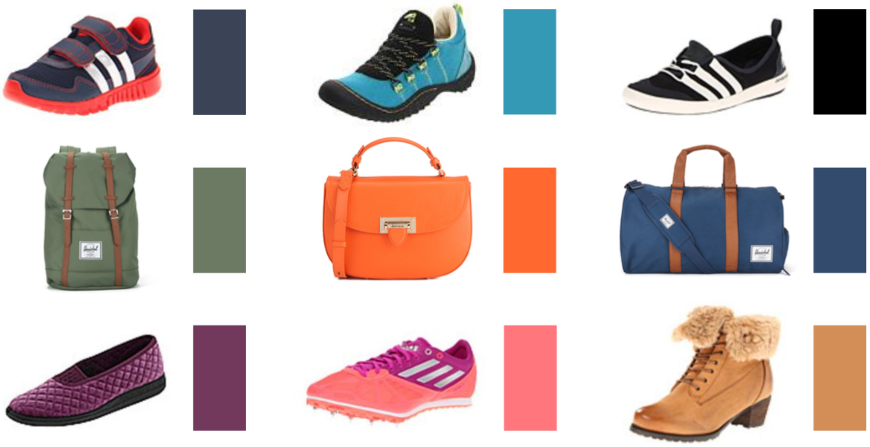


Figure 3.1: *Examples of dominant colours of images.*

model correlates well with human colour sensation and the saturation component S represents colour purity, i. e., how much the real colour is diluted by white [Smith 1978]. Therefore, the saturation can describe the vividness of the colour, which is the visual focus of an image. Here, a method is presented to extract a dominant colour using the saturation.

The saturation S can be converted from RGB colour space and expressed as [Smith 1978]

$$S = \begin{cases} \frac{V-X}{V}, & \text{if } V \neq 0 \\ 0, & \text{otherwise} \end{cases} \tag{3.1}$$

where $V = \max(R, G, B)$ and $X = \min(R, G, B)$.

For each pixel of an image, its score of saturation is calculated. If the score of one pixel is the maximum, the colour of the pixel is regarded as the dominant colour. The score is defined as

$$score = \begin{cases} \max[(S + \alpha)M], & L < 0.9 \\ 0, & L \geq 0.9 \end{cases} \tag{3.2}$$

where

$$L = \frac{0.3R + 0.59G + 0.11B}{255} \tag{3.3}$$

In Eq. (3.2), $S = \{S_1, S_2, ..., S_N\}$ and $M = \{M_1, M_2, ..., M_N\}$. S_i is the saturation of the ith pixel of an image and M_i is the amount of pixels which have the same RGB colour as the ith pixel. N is the total number of pixels. Since the saturation of the grayscale is zero, a threshold value α is added to avoid ignoring the pixel with grayscale. The influence of α on the dominant colour is discussed in the next subsection. In addition, L represents the brightness. In order to avoid the influence of brightness on the image, the *score* of the pixel with a high brightness ($L \geq 0.9$) is defined as zero in Eq. (3.2). The brightness can be acquired from the L component of HSL colour space and converted from RGB colour directly using Eq. (3.3) [Smith 1978].

3.1.2 Experiments

According to (3.2), assuming α is a predetermined constant, the score of a colour in an image depends on its saturation and pixel count. For example, if a colour has a higher saturation and a larger number of related pixels compared with other colours, it will get the highest score. As a result, the colour with the highest score can be regarded as the dominant colour because it expresses more intensity among other colours in the image. However, to judge the dominant colour of the same image, different people may have different answers because of their intuitive preference. In order to avoid artificial disturbances, Eq. (3.2) is adapted to decide the dominant colours of images in this chapter.

In order to investigate the influence of α in Eq. (3.2) on the dominant colours of images, different values of α, i. e., α=0, 0.01, 0.05, 0.1, 0.15 and 0.2, were tested on different shoe images, and six examples are shown in Figure 3.2. The results show that α has no effect on highly saturated images (The first two rows) but impacts the dominant colour of lowly saturated images (The last four rows). The reason is that α can increase the score in Eq. (3.2) when S is small and help to extract the correct dominant colour of lowly saturated images. According to the test results, $\alpha = 0.05$ and $\alpha = 0.1$ perform better in finding the dominant colours of different images. For the lowly saturated images, they can recognize colours with low saturation and have less influence on calculating the score. In this chapter, α is set to 0.1.

The effectiveness of the method proposed above is further illustrated by comparing it with the popular k-means method, as shown in Figure 3.3. The results show that the dominant colour extracted by the method proposed above is more accurate than the k-means method. Although a colour similar to the dominant colour can be found in the colour set of the k-means method, it is difficult to pick it out. In contrast, the method proposed above is more effective.

3.1.3 SCSBIR dataset

With the dominant colour extraction method, a SCSBIR dataset was created based on the Shoe Dataset [Yu et al. 2016] , Chair Dataset [Yu et al. 2016] and Handbag Dataset [Song et al. 2017] specifically to meet the requirements of the proposed SCSBIR method. It contains 419, 297 and 568 single-colour sketch-image pairs of shoes, chairs and handbags, respectively. The edge maps extraction method [Zitnick & Dollár 2014] was used to extract the corresponding single-colour edge maps from im-

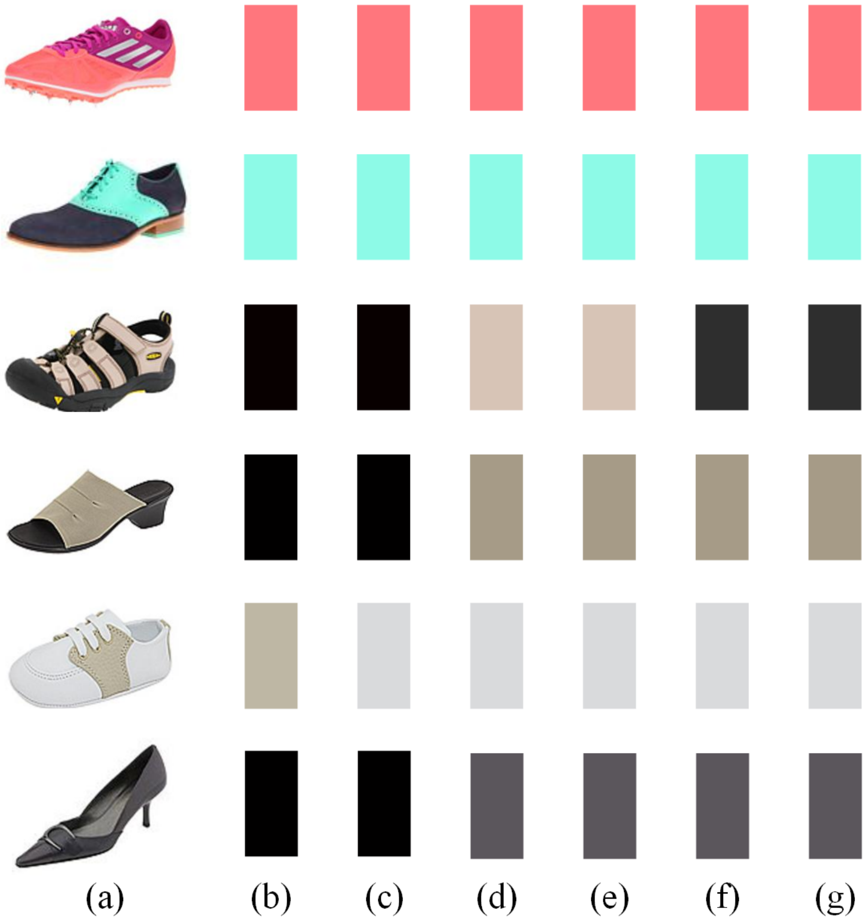


Figure 3.2: *The influence of* α *on the dominant colour. (a) Input images. (b)* $\alpha = 0$*. (c)* $\alpha = 0.01$*. (d)* $\alpha = 0.05$*. (e)* $\alpha = 0.1$*. (f)* $\alpha = 0.15$*. (g)* $\alpha = 0.2$*.*

ages with the dominant colours and they were taken as the input of the image branches during model training.

Similarly, the colour sketch corresponding to each image is obtained by using the dominant colour of the image to colour the original black-and-white sketch. Figure 3.4 shows some examples of single-colour sketch-image pairs in the SCSBIR dataset. Although the created SCSBIR dataset is not very large, it fully meets the needs of fine-tuning and testing of the proposed SCSBIR model (see Section 3.3).

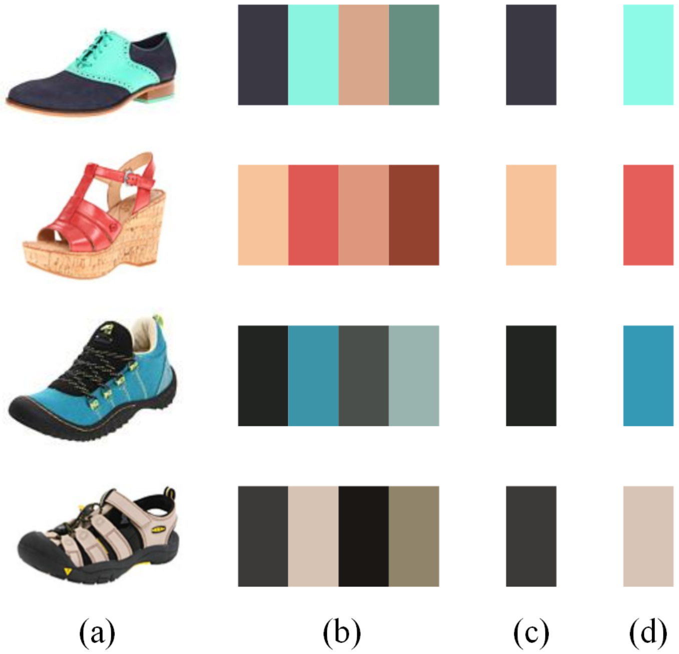


Figure 3.3: *Comparison between k-means and the method proposed in Subsection 3.1.1. (a) Input images. (b) The sets of k-means colours ($k = 4$) of images. (c) The dominant colours with the largest proportion in the sets of k-means colours. (d) The dominant colours of images using the method proposed in Subsection 3.1.1.*

3.2 SCSBIR method

The proposed SCSBIR method includes two stages, i. e., shape matching and colour matching. In the shape matching stage, a homogeneous triplet network is applied, which is an improved version of the Siamese network, to extract shape features of single-colour sketches and images, and a soft attention model and two shortcut connection architectures are adopted to improve retrieval precision of the network. In the colour matching stage, two popular colour models, i. e., RGB and HSV, are adopted to estimate the colour similarity between single-colour sketches and the retrieved images from the shape matching stage. The details of the two stages are introduced in the following subsections.

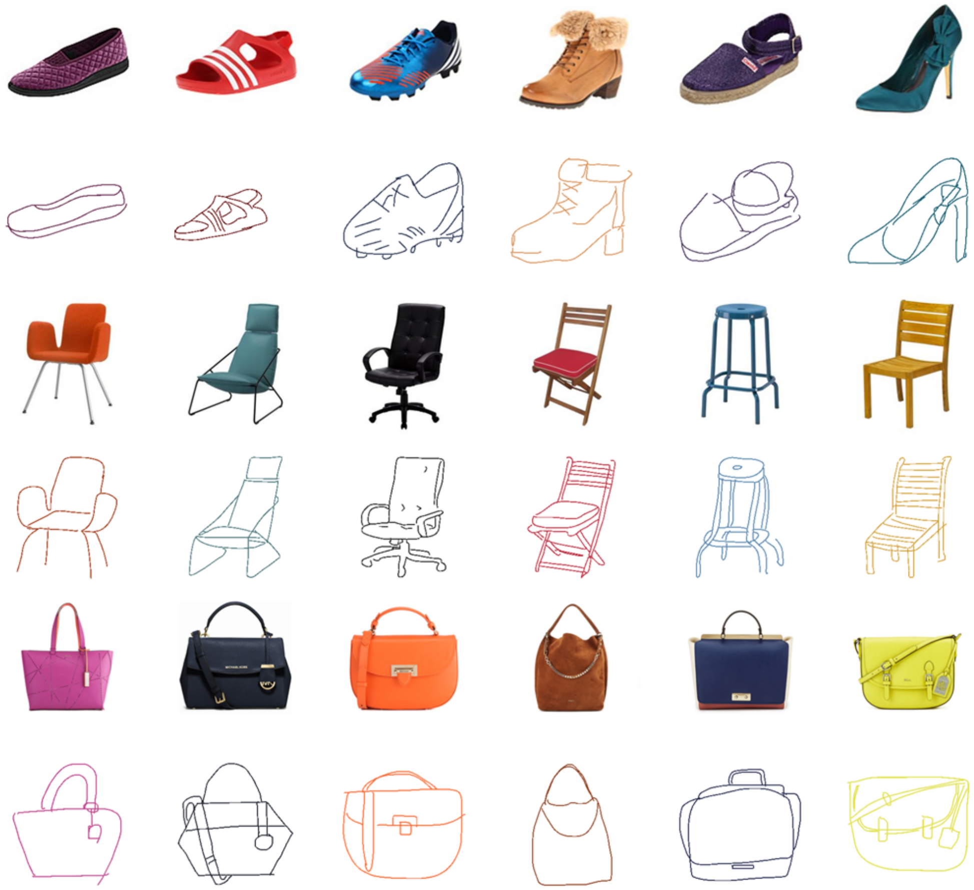


Figure 3.4: *Examples of the SCSBIR dataset.*

3.2.1 Shape matching

In this stage, a deep triplet network is used to learn shape feature imbedding of sketches and images, and a triplet loss is adopted as the optimization objective to train the deep triplet network.

Network architecture

The triplet network is composed of three CNN branches, and it needs a triplet for each training process. Since the inputs of the second and third branches belong to the same domain, these two branches share a set of parameters. If the first branch shares the same set of parameters with the second and third branches, the network is a homogeneous triplet network, as shown in Figure 3.5 where q is a query sketch , p^{+} is a similar image and p^{-} is a dissimilar image. Note that there is only one CNN model and the three CNNs in the figure are the same. If the parameters of the first branch are different from those of the second and third branches and have an independent set of weights, the network is a heterogeneous triplet network. The limitation of the homogeneous triplet network is that the three inputs in a triplet should have small domain differences, otherwise it is difficult to train the network and find the joint feature embedding space for the three inputs. Since the training dataset created by this work is not enough for training needs, in order to avoid the overfitting problem and alleviate the domain discrepancy, the homogeneous network is selected and the dataset is processed by extracting colour edge maps, which are used as inputs of the second and third branches instead of images.

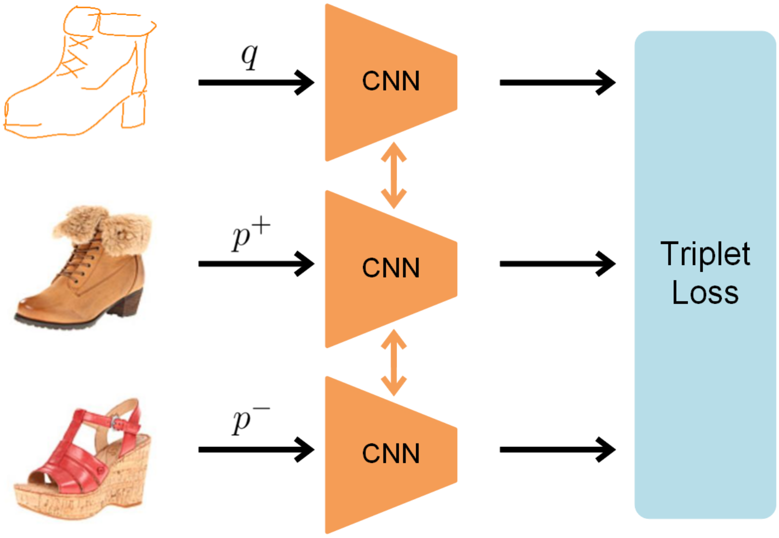


Figure 3.5: *The homogeneous triplet network.*

Since the Sketch-a-Net [Yu et al. 2015] is specifically designed for sketch recognition, it is used as the base net of CNNs in the triplet network. Inspired by the work of [Song et al. 2017], a soft attention model is implemented to improve the retrieval accuracy of the SCSBIR. The vector distribution of the soft attention output is a kind of soft distribution, which means that an attention distribution probability of any region in the input image is given. Bahdanau et al. [2014] first proposed the soft attention model and applied it to the field of machine translation. In this chapter, the soft attention model is adopted in each branch of the triplet homogeneous network. In addition, the shortcut connection architectures [He et al. 2016] are employed to solve the problem of gradient disappearance in deep networks. The final CNN structure in the triplet network is the same as [Song et al. 2017], as shown in Figure 3.6.

Triplet loss

In the shape matching stage, the goal is to find the images, which have similar shapes to a query sketch. The shape similarity between sketches and images can be described by a Euclidean distance [Yu et al. 2015]. A small value of distance means the sketch and image are similar, whereas a large value indicates the dissimilarity. Given a triplet of a query sketch q, a similar image p^+ and a dissimilar image p^-, two Euclidean distances can be obtained simultaneously using the homogeneous triplet network

$$\begin{cases} d(q,p^+) = \left\| f_\theta(q) - f_\theta(p^+) \right\|_2^2 \\ d(q,p^-) = \left\| f_\theta(q) - f_\theta(p^-) \right\|_2^2 \end{cases} \tag{3.4}$$

where $d(\cdot)$ is the Euclidean distance, $f_\theta(\cdot)$ is the feature embedding of the branch, which maps the three inputs to a joint feature embedding space, respectively.

In order to make the triplet network learn to decrease the distance between the query sketch and the similar image, i. e., $d(q,p^+)$, while

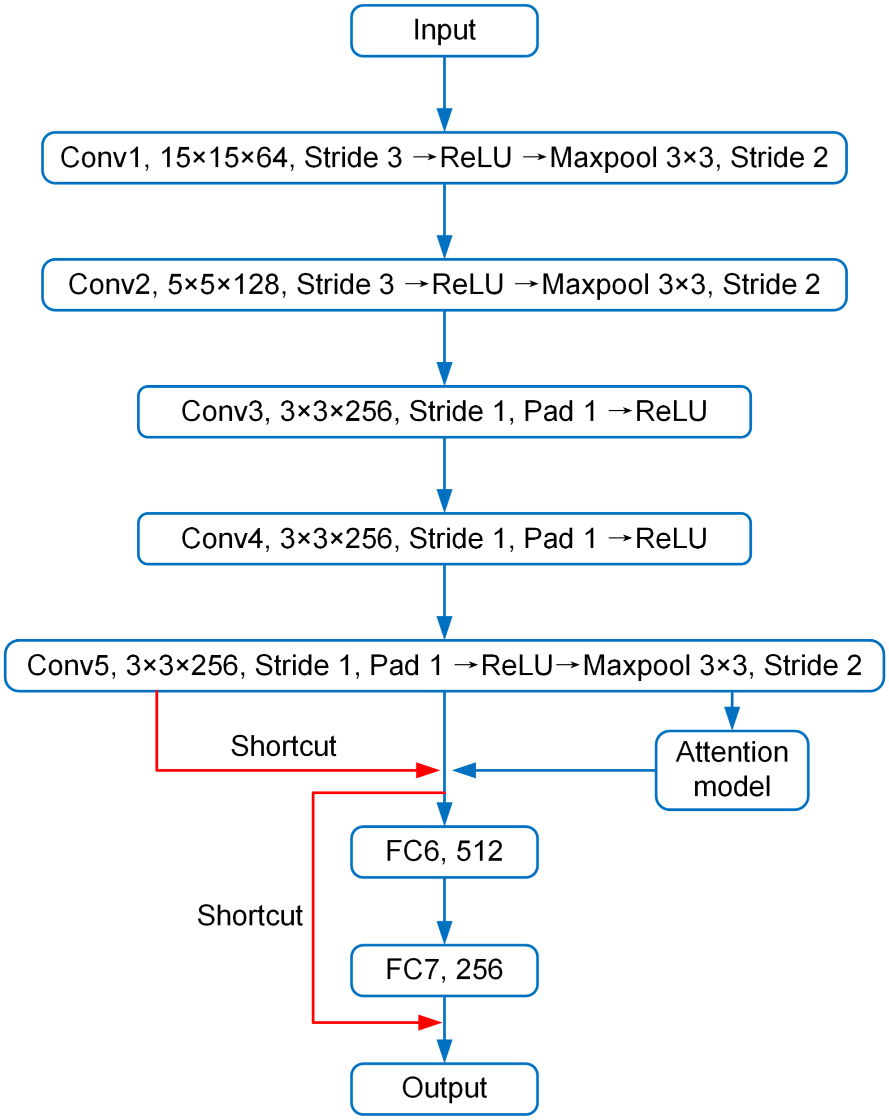

Input
Conv1, 15×15×64, Stride 3 →ReLU →Maxpool 3×3, Stride 2
Conv2, 5×5×128, Stride 3 →ReLU →Maxpool 3×3, Stride 2
Conv3, 3×3×256, Stride 1, Pad 1 →ReLU
Conv4, 3×3×256, Stride 1, Pad 1 →ReLU
Conv5, 3×3×256, Stride 1, Pad 1 →ReLU→Maxpool 3×3, Stride 2
Shortcut
Attention model
FC6, 512
Shortcut
FC7, 256
Output

Figure 3.6: *The CNN structure.*

increase the distance between the query sketch and the dissimilar image, i. e., $d(q, p^-)$, the triplet network needs to satisfy

$$d(q, p^+) - d(q, p^-) + \lambda \leq 0 \tag{3.5}$$

where λ is a margin, which means the distance between $d(q, p^+)$ and $d(q, p^-)$. Eq. (3.5) is to make $d(q, p^-)$ larger than the sum of λ and $d(q, p^+)$.

To achieve this goal, a triplet loss is defined as

$$L(q, p^+, p^-) = \max[d(q, p^+) - d(q, p^-) + \lambda, 0] \tag{3.6}$$

where $L(q, p^+, p^-)$ is the maximum value between $d(q, p^+) - d(q, p^-) + \lambda$ and 0. When $L(q, p^+, p^-)) = 0$, Eq. (3.5) is constantly satisfied.

Considering all triplets in the dataset, the ultimate optimization goal is

$$\min_{\theta} \sum_{i=1}^{N} L(q_i, p_i^+, p_i^-) \tag{3.7}$$

where N is the total number of triplets and θ represents the parameters of the sketch and image input branches.

By minimizing Eq. (3.7), the distance between q and p^+ will be narrowed while the distance between q and p^- will be widened. The triplet network can acquire the feature representations of inputs with colour information if there are sufficient triplet annotations.

To achieve single-colour sketch based image retrieval, the triplet network with the triplet loss is applied to carry out shape matching between a single-colour sketch and images. The final retrieval results are obtained by calculating the colour similarity values between the input single-colour sketch and the top ten retrieval results of shape matching.

3.2.2 Colour matching

In the colour matching stage, two popular colour models, i. e., the RGB model and the HSV model, are separately used to describe the colour features of single-colour sketches and edge maps of images

RGB model

The RGB model is a three-colour model, which is the most commonly used colour model for hardware devices [Chernov et al. 2015]. It is a kind of additive colour scheme which adds different intensities of red, green and blue lights to produce different visible colours. The main disadvantage of the RGB model is that it is not intuitive for humans. It is difficult to know the cognitive attributes of the colour represented by the values of R, G and B, so that the RGB model does not conform to human's colour perception. In addition, the RGB colour space is nonuniform, and the perceptual difference between two colours cannot be represented by the Euclidean distance between two colour points in the colour space.

Histograms are used to describe the three RGB channels of a single-colour sketch and an image, respectively, and then Hellinger distance is applied to calculate the colour similarity between the histograms of the single-colour sketch and the image. Hellinger distance is widely used to study the convergence of likelihood ratios between two distributions [Le Cam & Yang 2012], which is expressed as

$$H_k(D^k, E^k) = \frac{1}{\sqrt{2}} \left[\sum_{i=1}^{n} \left(\sqrt{\frac{D_i^k}{\sum_{j=1}^{n} D_j^k}} - \sqrt{\frac{E_i^k}{\sum_{j=1}^{n} E_j^k}} \right)^2 \right]^{\frac{1}{2}} \tag{3.8}$$

where D^k and E^k are the histogram vectors of k (k is R, G or B) channel of the colour sketch and the image, respectively, and D_i^k and E_i^k are the ith bin in D^k and E^k, respectively. Eq. 3.8 can be simplified into

$$H_k(D^k, E^k) = \left[1 - F_k(D^k, E^k)\right]^{\frac{1}{2}} \tag{3.9}$$

where $F_k(D^k, E^k)$ is the Bhattacharya coefficient of D^k and E^k, which is defined as

$$F_k(D^k, E^k) = \sum_{i=1}^{n} \sqrt{\frac{D_i^k E_i^k}{(\sum_{j=1}^{n} D_j^k)(\sum_{j=1}^{n} E_j^k)}} \tag{3.10}$$

Note that Eq. 3.9 is also known as Bhattacharyya distance. The Hellinger distance of three RGB channels is defined as

$$dist = \frac{1}{3}[H_R(D^R, E^R) + H_G(D^G, E^G) + H_B(D^B, E^B)] \tag{3.11}$$

HSV model

Compared with the RGB model, the HSV model is closer to the human's colour perception. In this model, H (hue) stands for true colours, S (saturation) for colour purity, and V (value) for brightness [Chernov et al. 2015]. The HSV model has correlated and uniform coordinates matching the human perception of colour and its histogram is easy to extract [Ortega et al. 1998]. Since the V coordinate in the HSV colour space is easily affected by the lighting condition, the H-S coordinates are used to form 2D histograms of the single-colour sketch and the image, respectively, and then the same form of Hellinger distance in Eq. 3.9 is applied to calculate the colour similarity between the 2D histograms of the single-colour sketch and the image, which is expressed as

$$dist(S, I) = \left[1 - \sum_{i=1}^{n} \sqrt{\frac{S_i I_i}{(\sum_{j=1}^{n} S_j)(\sum_{j=1}^{n} I_j)}}\right]^{\frac{1}{2}} \tag{3.12}$$

where S and I are the 2D histograms of the single-colour sketch and the image, and S_i and I_i are the ith bin in S and I, respectively.

3.2.3 Pipeline

After obtaining the SCSBIR model trained by the training set of SCSBIR dataset, it is applied to the testing set to verify the retrieval accuracy of the proposed SCSBIR method. Taking shoes as an example, the pipeline of the proposed SCSBIR method is illustrated in Figure 3.7.

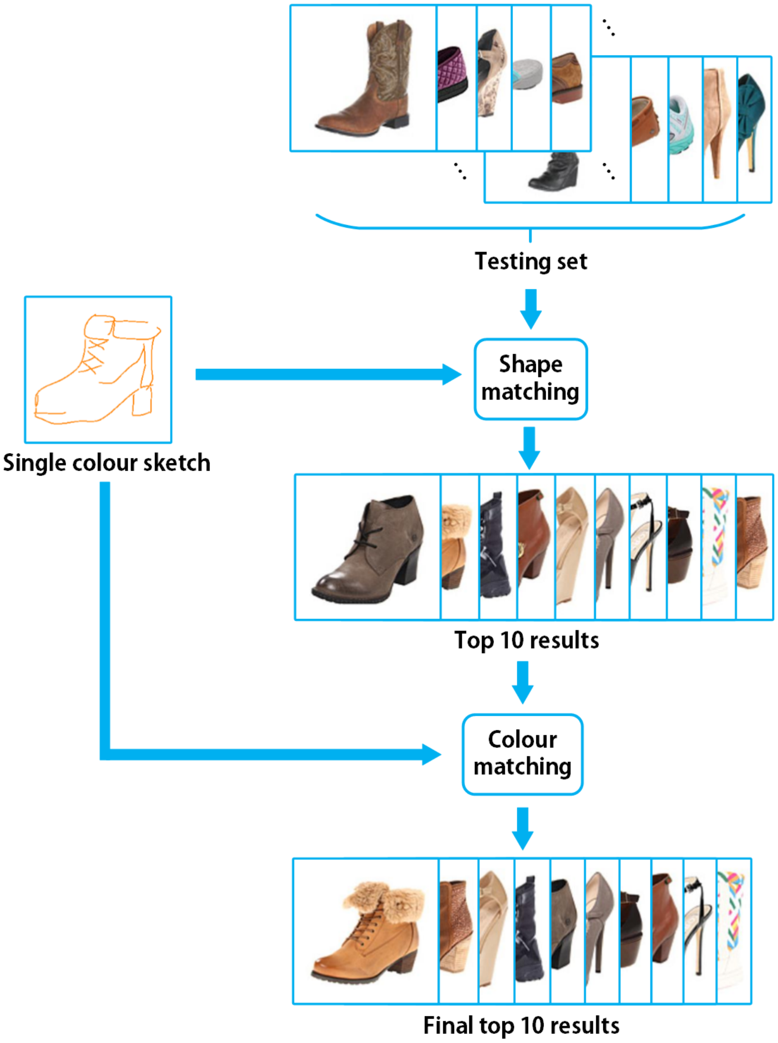


Figure 3.7: *Pipeline of the SCSBIR method.*

In the pipeline of the SCSBIR method, all shoe images in the testing set have obtained their feature vector representations through preprocessing to improve the speed of real-time retrieval. The SCSBIR method includes three steps. First, the user inputs a single-colour sketch of a shoe as a probe into the SCSBIR model and gets its feature vector representation in real time. Second, the shape matching is applied to estimate shape similarity between the sketch feature vector and all the image feature vectors and find the top ten retrieval results, which are most similar to the shoe sketch in the dataset. Third, colour matching is used to estimate the colour similarity between the single-colour sketch and the top ten results of shape matching, and reorder the ten results according to the colour similarity. Note that the proposed colour matching has two methods based on the RGB model and the HSV model, respectively. As shown in Figure 3.7, the first retrieval result in the final ten results is the most similar shoe to the input single-colour sketch in the shape and colour.

3.3 Experiments

3.3.1 Experiment settings

The SCSBIR model employs the Sketch-a-Net [Yu et al. 2015] as the basic model and is pre-trained in three steps [Yu et al. 2016]. First, the basic network is trained to recognize 1,000 categories of the ImageNet dataset [Deng et al. 2009] with the edge maps extracted from images. Then, the model is fine-tuned to recognize the 250 categories of TU-Berlin 20,000 sketch dataset [Eitz et al. 2012a]. At last, fine-grained retrieval ability of the model is obtained by retraining the model with the dataset consisting of 187 sketch-image categories selected from the TU-Berlin dataset and the ImageNet dataset separately.

After pre-training, the pre-trained model is fine-tuned using the created SCSBIR dataset. The SCSBIR dataset contains three categories: shoe, chair, and handbag. Each category is split into two parts. Following the same splits in [Yu et al. 2016] and [Song et al. 2017], 304 pairs, 200 pairs and 400 pairs are used for fine-tuning training, respectively, and 115 pairs, 97 pairs and 168 pairs are used for testing, respectively. The training set of each category is used to fine-tune the model specifically for the target category. In the fine-tuning process, the initial learning rate is set to 0.001 and the mini-batch size is set to 128. The attention module consists of 2 convolutional layers with kernel size 1×1. The proposed SCSBIR method is implemented on TensorFlow with a NVIDIA Titan XP GPU.

3.3.2 Results

The proposed SCSBIR method is compared with other two fine-grained sketch-based image retrieval methods, i.e., DTRM [Yu et al. 2016] and FG-SBIR [Song et al. 2017], which apply deep CNN for feature extraction. The DTRM is the first to use deep CNN for fine-grained sketch-based image retrieval. To improve the retrieval accuracy, the FG-SBIR applies a soft attention model and shortcut connection architectures based on DTRM. The method developed in this chapter, DTRM, and FGSBIR are tested on the SCSBIR testing set and the retrieval accuracies within top K ($K = 1, 2, ..., 10$) retrieval results are calculated. Precision @ K is used to describe the retrieval accuracy, which is the percentage of the amount of times when the true-match image of a single-colour sketch is ranked in the top K retrieval results.

Since two colour matching algorithms, one in the RGB colour space and the other in the HSV colour space, are used, the results of using different colour matching algorithms are presented, respectively. The results of the comparison for $K = 1$ to 10 are shown in Figure 3.8. Compared with the DTRM and FG-SBIR methods, the proposed SCSBIR method has better retrieval accuracy within top K ($K = 1, 2, ..., 10$) on all three categories except for top 10 on the chair category. Moreover, three cases: $K = 1$, $K = 3$ and $K = 5$ are chosen to compare the retrieval accuracy of DTRM, FG-SBIR and the proposed SCSBIR method, which are shown in Table 3.1. The results show that the retrieval accuracy of the proposed SCSBIR method in the HSV colour space averagely increase around 33.90% and 29.83% at $K = 1$, 16.45% and 12.19% at $K = 3$, and 7.54% and 5.62% at $K = 5$ compared with DTRM and FG-SBIR, and the retrieval accuracy of the proposed SCSBIR method in the RGB colour space averagely increase around 35.92% and 31.85% at $K = 1$, 16.55% and 12.39% at $K = 3$, and 7.54% and 5.62% at $K = 5$ compared with DTRM and FG-SBIR. The results indicate that the proposed SCSBIR method in both RGB and HSV colour spaces has a better performance than other two models in fine-grained single-colour sketch based image retrieval.

3.3.3 Visualizing retrieval results

Part of the retrieval results is visualized to show the better retrieval accuracy of the proposed SCSBIR method compared with the DTRM and FG-SBIR. In Figure 3.9, the first row is the retrieval results of the proposed SCSBIR method in the HSV colour space with query colour sketch, the second row is the retrieval results of the proposed SCSBIR method in the RGB colour space with query colour sketch, the third row is the retrieval results of DTRM with black-and-white sketch, which has the same contour lines as the colour sketch, and the fourth row is the

Table 3.1: *Comparison of retrieval accuracy at $K = 1$, $K = 3$ and $K = 5$*

Dataset-Shoe	$K = 1$	$K = 3$	$K = 5$
DTRM	53.04%	73.91%	83.48%
FG-SBIR	57.39%	77.39%	85.22%
SCSBIR method (HSV)	91.30%	93.04%	93.04%
SCSBIR method (RGB)	92.17%	93.04%	93.04%
Dataset-Chair	$K = 1$	$K = 3$	$K = 5$
DTRM	72.16%	85.57%	93.81%
FG-SBIR	75.26%	90.72%	94.85%
SCSBIR method (HSV)	96.91%	97.94%	97.94%
SCSBIR method (RGB)	97.94%	97.94%	97.94%
Dataset-Handbag	$K = 1$	$K = 3$	$K = 5$
DTRM	39.29%	64.29%	73.81%
FG-SBIR	44.05%	68.45%	76.79%
SCSBIR method (HSV)	77.98%	82.14%	82.74%
SCSBIR method (RGB)	82.14%	82.74%	82.74%

retrieval results of FG-SBIR using the same black-and-white sketch.

By comparing the visual retrieval results, the proposed SCSBIR method performs better in appearance matching including shape matching and colour matching. Unlike DTRM and FG-SBIR, the proposed SCSBIR model can move the image with similar colour up to the top of the retrieval results. For example, on the top shoe example in the right column, since the input single-colour sketch is a brown boot sketch, the brown boots are moved up to the top while the boots of other colours are moved behind.

3.4 Summary

In this chapter, a fine-grained SCSBIR method based on multi-branch deep CNNs is proposed, and a triplet homogeneous network is used to solve the fine-grained SCSBIR problem on three categories. In addition, a SCSBIR dataset of single-colour sketch-image pairs is created and a new ranking method combined with the shape similarity matching and colour similarity matching is proposed, which makes the retrieval results get better matching in appearance. Extensive experiments are implemented to demonstrate the effectiveness and verify better retrieval performance of the proposed SCSBIR approach.

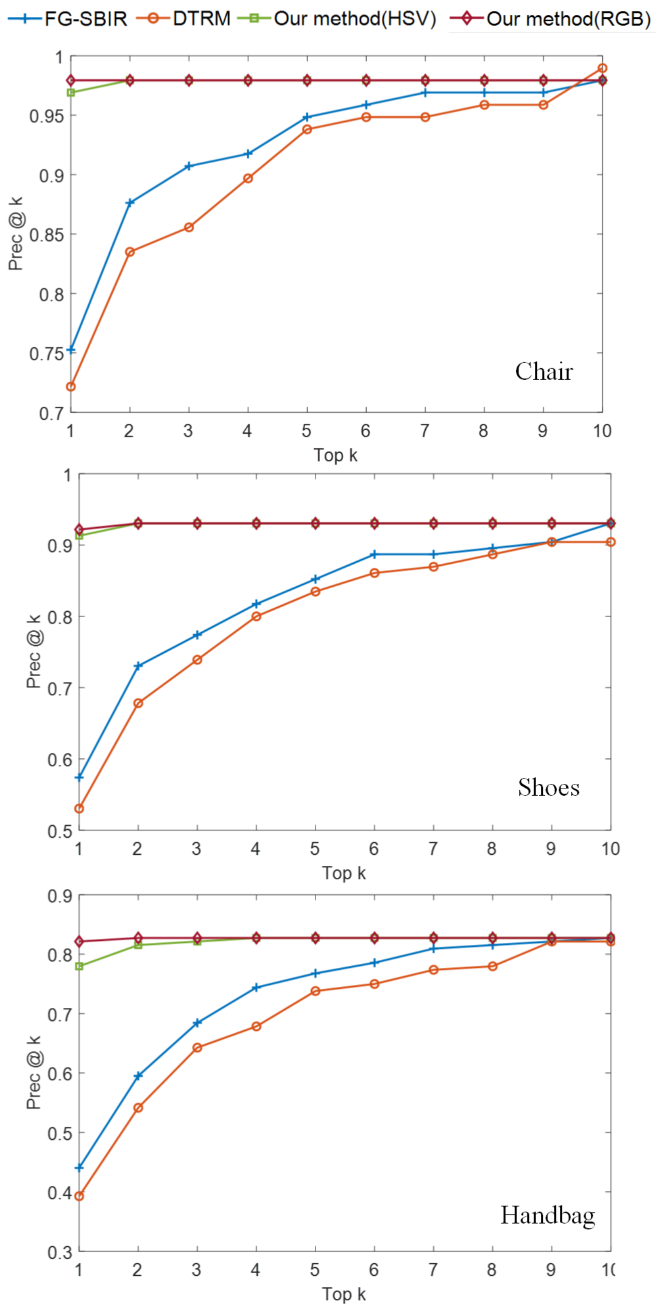


Figure 3.8: *Retrieval precision @ K for K = 1 to 10 of DTRM, FG-SBIR and the proposed SCSBIR method in the chair, shoes and handbag datasets.*

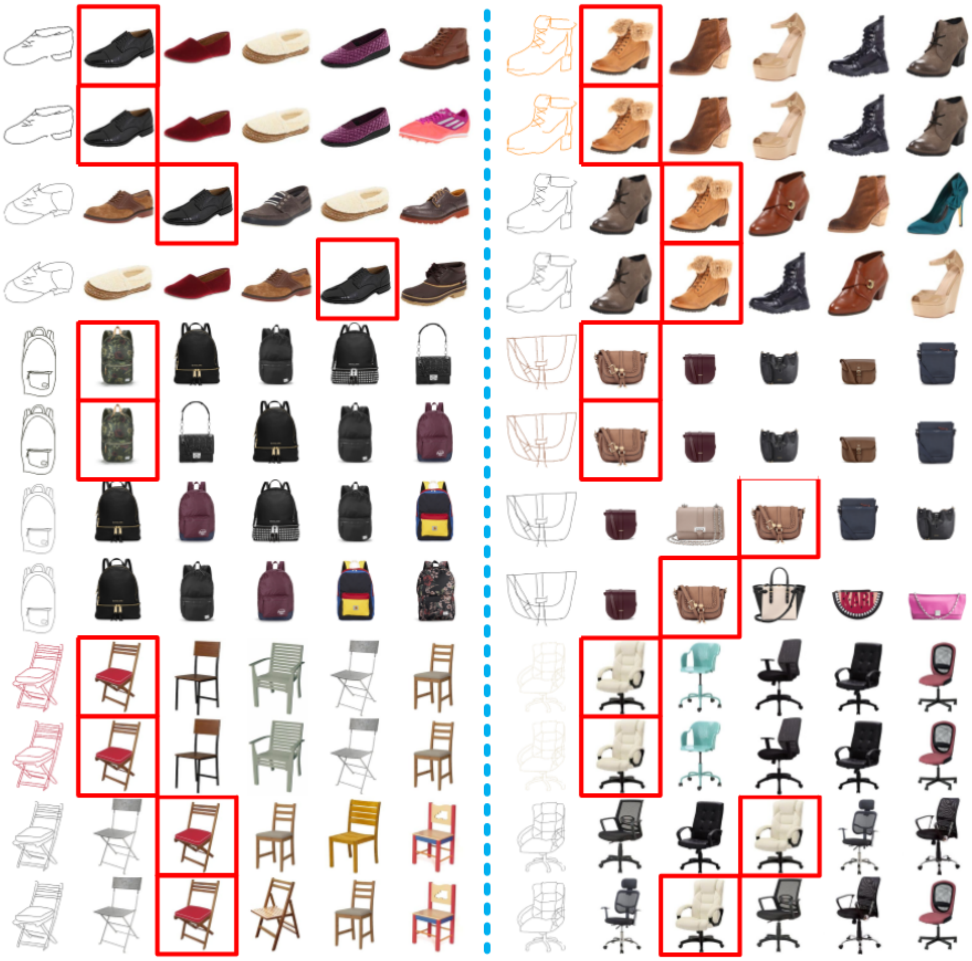


Figure 3.9: *The top five retrieval results by the proposed SCSBIR method in the HSV (the first row) and RBG (the second row) colour spaces, DTRM (the third row) and FG-SBIR (the fourth row). The true matches are highlighted in red..*

Chapter 4

Multi-colour sketch based image retrieval

Generally, sketch-based image retrieval is to search similar colour images based on a query sketch. When the retrieval objective is the goods, such as clothes and shoes, one style of them usually contains several different colours, as shown in Figure 4.1. Therefore, the retrieval methods only considering the shape matching cannot provide sufficient retrieval performance because they cannot distinguish the images which have the same shape but different colours. A few research studies try to retrieve images using colour sketches [Bui & Collomosse 2015; Xia et al. 2019; Fuentes & Saavedra 2021], but there are still some unsolved issues. First, there is no suitable dataset for MCSBIR, which contains not only multi-colour sketches but also images with the same shape but different colours. Second, there is no explicable feature representation method to describe the shape and colour information of an image together. In this chapter, the first multi-colour sketch dataset (Section 4.1) will be created and a novel MCSBIR method (Section 4.2) will be proposed to solve the above issues. At the end, several experiments are presented to demonstrate the effectiveness of the proposed MCSBIR method (Section 4.3).

4.1 Multi-colour sketch-image dataset

In order to deal with the task of MCSBIR, the dataset of multi-colour sketches is indispensable in the proposed MCSBIR method. Although there are many popular sketch datasets, such as TU-Berlin [Eitz et al. 2012a] and QuickDraw [Ha & Eck 2017], which are usually used for sketch recognition and sketch retrieval, all of them are black-and-while without any colour information. In addition, since the goal of this work is to apply colour sketches to retrieve the most similar images of goods in both shapes and colours, the complex image background will have extra information to affect the retrieval accuracy. Although matching the goods with a complex image background is an interesting research issue, the work here only focuses on the goods themselves.

Based on the above discussion, there are two requirements for building a multi-colour sketch-image dataset. First, the sketch and its corresponding image should be multi-colour and the image background should be clean. Second, there should be more than two images, which have the same shape but different colours. In Chapter 3, the SCSBIR method has been introduced, which uses the shoes as one category of its dataset. The shoes are ideal retrieval targets because the shape styles of shoes are various and they always have different colours for the same style, as shown in Figure 4.1. Therefore, shoe images are used as the retrieval targets.

The created dataset consists of two parts, i. e., images and sketches. Since the UT Zappos50K [Yu & Grauman 2014] is a large shoe dataset consisting of over 50 thousand images, 71 different styles of shoe images are selected from the shoe dataset. Each of the selected styles has 2-5 images with different colours and there are 232 images in total. Figure 4.1 shows some image examples of the created dataset.

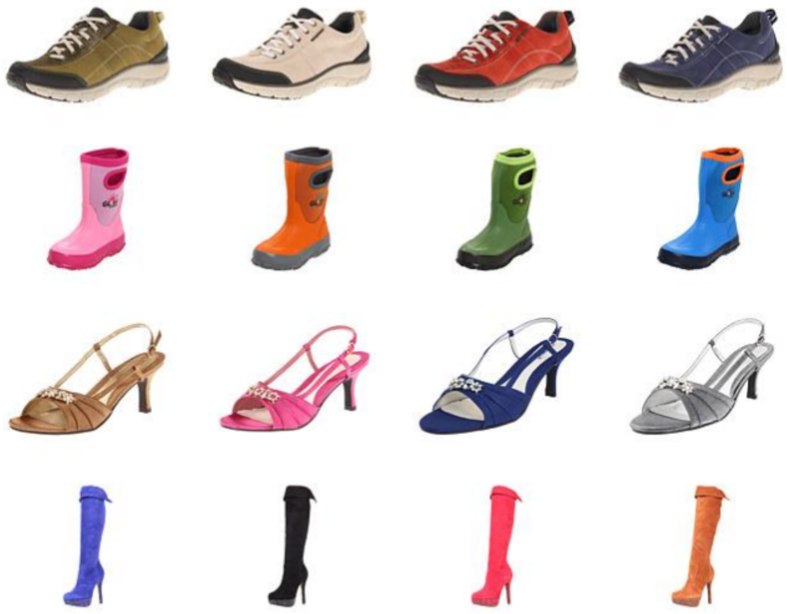


Figure 4.1: *Some image examples of the created dataset.*

In order to create the sketches of each style of shoes, three volunteers are invited to manually draw multi-colour sketches of the 232 shoe images. The image examples shown in Figure 4.1 indicate that the colour information of an image is complex. The image has a big colour range variation and it is very difficult to describe the variation by using sketch lines with single colours. More importantly, it is time-consuming for volunteers to draw sketches with many different colours. Therefore, it is required to reduce the number of colours in the image but also keep the visual appearance of the image intact. This process is also known as colour quantization [Celebi 2011].

One of the most widely used methods for colour quantization is the k-means clustering algorithm. In order to reduce the number of colours and find the main colours of an image, the classical k-means clustering algorithm is applied to cluster the colours. Given an image data set $X = \{x_1, x_2, ..., x_N\}$ where x_i is the RGB colour vector of the ith pixel, the objective of k-means clustering is to separate X into K clusters $C = \{C_1, C_2, ..., C_K\}$ and minimize the sum of the squared distances between each pixel colour and its closest center. The objective function can be described as [Celebi 2011]

$$E = \sum_{j=0}^{K} \sum_{x_i \in C_j} \|x_i - \mu_j\|_2^2 \tag{4.1}$$

where $\|\cdot\|_2$ denotes the Euclidean norm and μ_j is the center of cluster C_j calculated as the mean of all colour vectors in this cluster which is defined as

$$\mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i \tag{4.2}$$

By using the k-means clustering algorithm, quantized images are obtained, which have few numbers of colours once K is set to a small number. In this chapter, K is set to 3 and the corresponding multi-colour sketches are created according to the quantized images. Figure 4.2 shows some examples of multi-colour sketches, quantized images and original images.

Based on the 232 selected shoe images, 232 corresponding multi-colour sketches are created and the MCSBIR dataset is built.

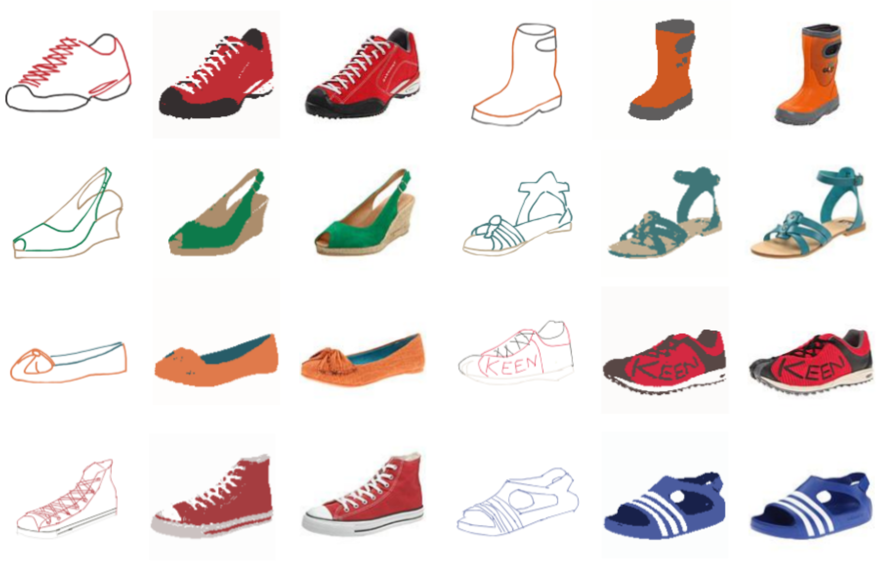


Figure 4.2: *Examples of created multi-colour sketches (the left column), quantized images (the medium column) and original images (the right column).*

4.2 MCSBIR method

For a given query colour sketch q and a set of candidate images P, the retrieval task is to compute the similarity between q and each image $p \in P$ and rank all candidate images based on their similarities in the hope that the true image can be found, which matches the shape and colour of the query sketch and ranks at the top. However, there are two challenges: (1) finding a joint feature embedding space of the multi-colour sketches and images, and (2) describing the features of the shape and colour of both sketches and images. To address the two challenges, a novel feature embedding is first proposed to describe the shape and colour information together and then a triplet loss function is developed to learn the feature embedding with a two-stage network architecture.

4.2.1 Feature embedding

Feature embedding is to use a feature vector to represent the information of input data. In previous related studies, the shape and colour information are usually represented by two different features, respectively [Bui & Collomosse 2015; Xia et al. 2019] or mixed into a single feature like a 'black box' without manual interventions [Fuentes & Saavedra 2021]. In this chapter, a new feature embedding $f_{sc}(\cdot)$ is proposed to clearly and explicably describe the shape and colour information within a single feature vector named Shape-Colour (SC) feature, as shown in Figure 4.3. The SC feature includes a front vector with N_s elements, which represents the shape information and a back vector with N_c elements, which represents the colour information.

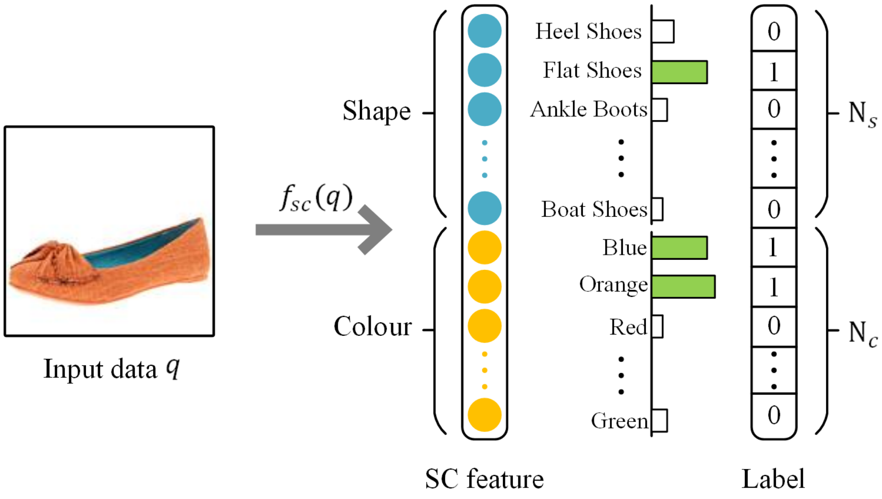


Figure 4.3: *Feature embedding of the MCSBIR.*

Shape information

Generally, the images in different categories have different shapes. Most retrieval studies distinguish different image shapes by utilizing classification tasks in their pre-train stages [Bui & Collomosse 2015; Xia et al.

2019; Fuentes & Saavedra 2021]. Therefore, the same strategy is applied to describe the shape information in the SC feature. According to the UT Zappos50K dataset, the dataset can be roughly classified into 17 ($N_s = 17$) categories, i. e., heel shoes, flat shoes, ankle boots, etc. Since a shoe image can only belong to one category, the number of categories can be adopted as the size of the front vector, i. e., $1 \times 17.$, and each vector element corresponds to one category. In the data labelling process, a one-hot encoding is used to generate labels, i. e., the corresponding element in the front vector is set to be 1 and the rests are set to be 0, as shown in Figure 4.3. In the training process, the feature embedding is optimized to generate the front vector in which the corresponding element of the input has the highest possibility compared with other elements according to the given labels. For example, in Figure 4.3, since the input is a flat shoe, the corresponding element in the front vector has the highest possibility presented in green colour.

Colour information

There are thousands of categories of colour according to different RGB values. It is difficult to describe all colours in the feature embedding as well as label them. To address this problem, colours are visually classified into several groups according to colour ranges. Since the luminance and chrominance properties of the RGB colour are not separated, the colour differences cannot be correctly captured [Chernov et al. 2015]. Therefore, the HSL colour model is used for the colour classification task in this chapter.

The hue (H) of HSL represents a true colour and it is a colour portion, which can be expressed as a number from 0 to 360 degrees: 0° or 360°

is red, 60° is yellow, 120° is green, 180° is cyan, 240° is blue, and 300° is magenta [Chernov et al. 2015], as shown in Figure 4.4.

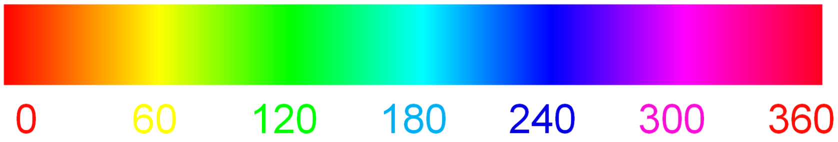


Figure 4.4: *The hue from 0 to 360 degrees.*

Thus, the hue can be applied to classify colours into several groups. The hue can be calculated by [Smith 1978]

$$H = \begin{cases} 5+b, & \text{if } R=V, G=X \\ 1-g, & \text{if } R=V, G\neq X \\ 1+r, & \text{if } G=V, B=X \\ 3-b, & \text{if } G=V, B\neq X \\ 3+g, & \text{if } B=V, R=X \\ 5-r, & \text{if } B=V, R\neq X \end{cases} \tag{4.3}$$

where $H := H \times 60$, $V = \max(R, G, B)$, $X = \min(R, G, B)$, and

$$r = \frac{V-R}{V-X}, g = \frac{V-G}{V-X}, b = \frac{V-B}{V-X} \tag{4.4}$$

Since the hue cannot deal with the grey-scale colour, the saturation (S) and brightness (L) of the HSL colour model are applied to recognize the black, grey and white colours. The saturation and brightness can be obtained from Eqs (3.1) and (3.3), respectively. By using the HSL colour, 10 colours for SC feature embedding, i. e., $N_c = 10$, are defined. For each pixel in the input image, its brightness is first judged, then its saturation is checked, and finally its hue is calculated. Table 4.1 shows the classification standard for the 10 colours.

Table 4.1: *The classification standard for the 10 colours.*

Colour	Classification standard
Red	$0 < H \leq 20$, $310 < H \leq 360$
Orange	$20 < H \leq 50$
Yellow	$50 < H \leq 68$
Green	$68 < H \leq 154$
Cyan	$154 < H \leq 198$
Blue	$198 < H \leq 248$
Magenta	$248 < H \leq 310$
Black	$L < 0.2$
White	$L > 0.8$
Grey	$S < 0.25$

4.2.2 Loss function

Since the feature embedding $f_{sc}(\cdot)$ is defined to extract the SC feature of input data in Section 4.2.1, the similarity between the sketch q and the image p can be measured by using Euclidean distance between $f_{sc}(q)$ and $f_{sc}(p)$, i. e., $D(q,p) = \|f_{sc}(q) - f_{sc}(p)\|_2^2$. However, in the e-commerce application, the shape of the retrieved image is the most important because if the image shape is different to the query sketch, the image just belongs to other goods even though their colours are the same. Therefore, the Euclidean distance $D(q,p)$ should pay more attention to the shape information. Here, a new Euclidean distance is presented to separate the shape and colour information

$$D(q,p) = \beta \left\| f_{sc}(q)_s - f_{sc}(p)_s \right\|_2^2 + (1-\beta) \left\| f_{sc}(q)_c - f_{sc}(p)_c \right\|_2^2 \quad (4.5)$$

where $f_{sc}(\cdot)_s$ and $f_{sc}(\cdot)_c$ represent the front vector with the shape information and the back vector with the colour information of the SC feature, respectively. β is a trade-off parameter between 0 and 1 which indicates the proportion of the shape similarity in Eq. (4.5). $\beta > 0.5$ is chosen to make the shape information more important in the distance function.

In order to learn this feature embedding $f_{sc}(\cdot)$, the annotated triplet T is used as input training data which is defined as

$$T = (q, p^+, p^-) \tag{4.6}$$

where q is a query colour sketch, p^+ is a positive image with the same shape as q, and p^- is a negative image with a different shape from q.

The goal here is to learn the feature embedding $f_{sc}(\cdot)$ that maps sketches and images into a joint feature embedding space, in which the similarity between the query sketch and the image with the same shape and colour is high and the similarity between the query sketch and the image with different shapes and colours is low, which means that the distance between query q and positive p^+ should be smaller than the distance between query q and negative p^-, i. e., $D(q, p^+) < D(q, p^-)$. A schematic illustration of how the joint feature embedding space should behave is shown in Figure 4.5.

Given a mini-batch with size N, there are N triplets T_i $(i = 1, 2, ..., N)$. To achieve the goal of learning the feature embedding, a triplet loss function is formulated as

$$Loss = \frac{1}{N}\sum_{i=1}^{N} \max[D(q_i, p_i^+) - D(q_i, p_i^-) + \lambda, 0] \tag{4.7}$$

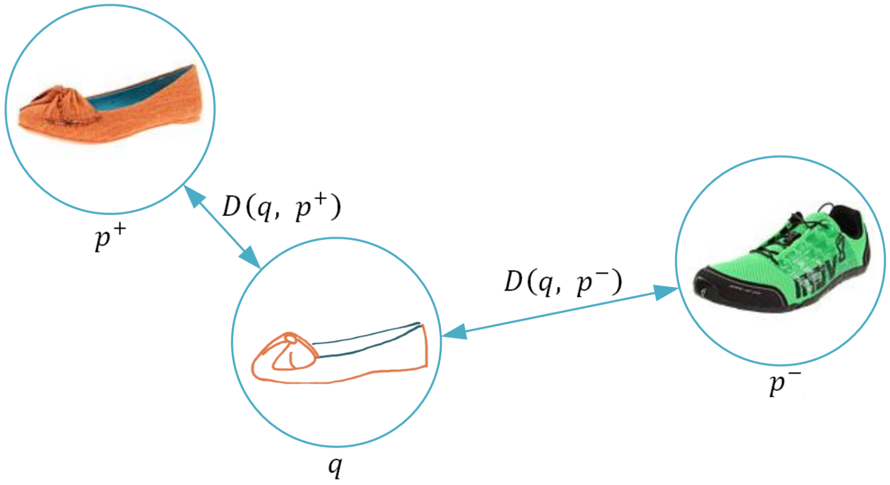


Figure 4.5: *The joint feature embedding space.*

where λ is a margin between $D(q, p^+)$ and $D(q, p^-)$, which makes $D(q, p^+)$ smaller than $D(q, p^-)$ by at least a margin λ.

Substituting Eq. (4.5) into (4.7), the smooth similarity loss function is obtained

$$\begin{aligned} Loss =& \frac{1}{N}\sum_{i=1}^{N} \log\{1+\exp\{\beta[\left\|f_{sc}(q_i)_s - f_{sc}(p_i^+)_s\right\|_2^2 \\ & - \left\|f_{sc}(q_i)_s - f_{sc}(p_i^-)_s\right\|_2^2] + (1-\beta)[\left\|f_{sc}(q_i)_c - f_{sc}(p_i^+)_c\right\|_2^2 \\ & - \left\|f_{sc}(q_i)_c - f_{sc}(p_i^-)_c\right\|_2^2] + \lambda\}\} \end{aligned} \tag{4.8}$$

4.2.3 Network architecture

There is a big domain discrepancy between the query sketches and the retrieved images because the sketches are abstract free-hand drawings, which consist of several simple lines with very limited information. To address this problem, some previous studies apply a heterogeneous network which has two separate CNN branches for sketches and images, respectively [Fuentes & Saavedra 2021; Bui et al. 2018], and other studies first extract the edge maps of images and then use a homogeneous network to train the edge maps and sketches with shared weights [Xia et al. 2019; Yu et al. 2016]. The edge map is similar to the sketch, so they have a small domain discrepancy and their CNN branches can share the same weights.

As discussed in Section 3.2.1, since the training data in the MCSBIR dataset is sparse, the homogeneous triplet network is selected to achieve the retrieval task. The network architecture of the proposed MCSBIR method is described in Figure 4.6, which consists of two stages: one for a classification task and the other for the retrieval task. In both stages, the same CNN is used to learn the feature embedding $f_{sc}(\cdot)$ and the last fc layer of the CNN is set to be the SC feature layer with $(N_s + N_c)$ neurons. The CNN in Stage 1 is first pre-trained, and then the pre-trained CNN is used to the three branches in Stage 2, which share weights to learn a joint feature embedding space. The training procedure is introduced in the next subsection.

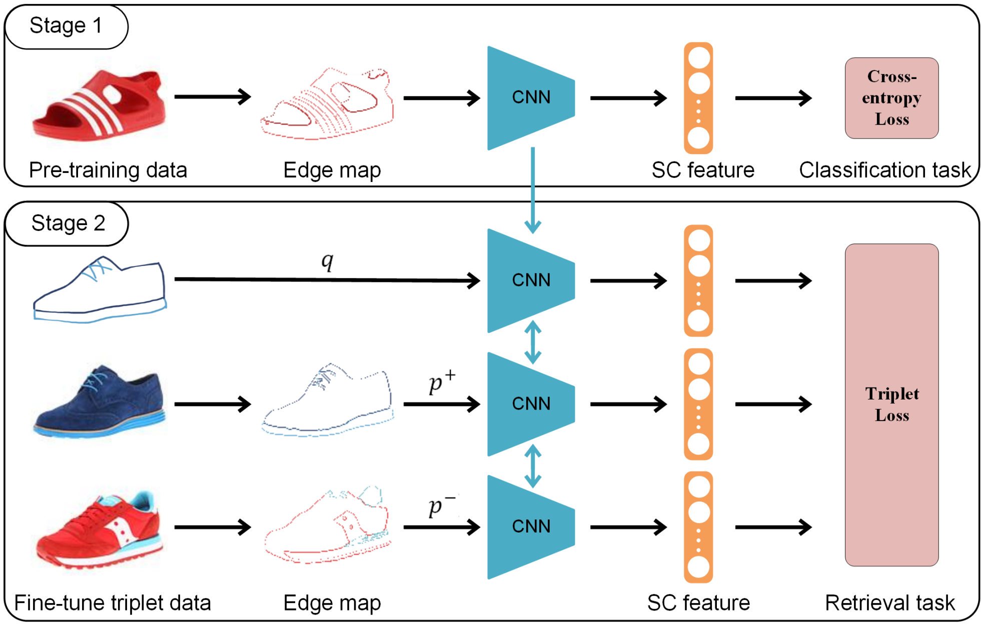


Figure 4.6: *The two-stage network architecture.*

In the run-time process, all test sketches and images are passed through the trained CNN model to generate their SC features. For each test sketch, the distances between the SC feature of the sketch and the SC features of all images are calculated and sorted from the smallest to the largest. The image with the smallest distance is the best retrieval result according to the test sketch.

In this chapter, the CNN structure applies ResNet50 [He et al. 2016], as shown in Figure 4.7. The rectangle block indicates the convolutional layer, in which the notation $(k \times k,\, n)$ denotes a filter of size k and n channels. The number on the top of the rectangle block represents the repetition of each unit, and the annotation on the bottom represents the layer name.

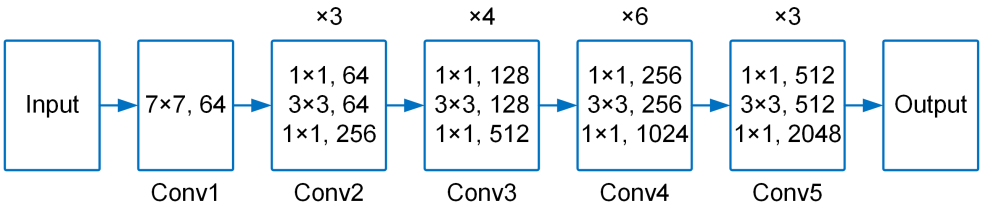


Figure 4.7: *The ResNet50 structure.*

4.2.4 Training procedure

Before inputting some images into the network, it is required to transfer the images into edge maps. Although there are many mature edge detection methods such as canny edge [Canny 1986] and gPb-OWT-UCM [Arbelaez et al. 2010], most of them cannot extract colours. Here, a simple method to extract colour edge maps from images is presented. First, quantized images are generated by Eq. (4.1), and then the canny edge detection method [Canny 1986] is used to extract black-and-white edge maps from the grey-scales of the quantized images. Given the RGB matrix $[Q]$ of the quantized images and the matrix $[E]$ of the black-and-white edge maps, the colour edge maps $[C]$ can be obtained by $[C] = [Q] \odot [E]$ where $\odot$ denotes component-wise multiplication. Some examples of colour edge maps are shown in Figure 4.8.

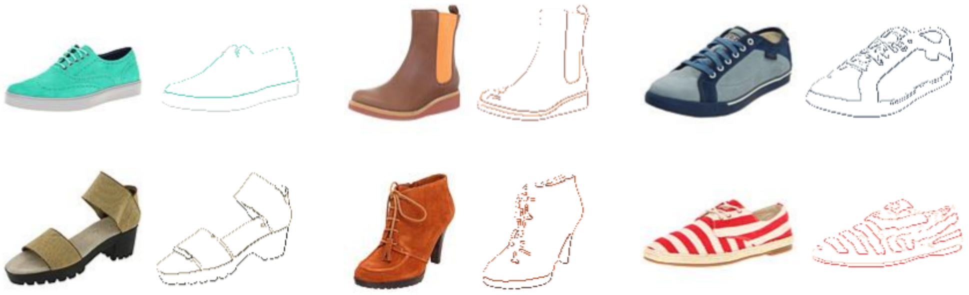


Figure 4.8: *Examples of colour edge maps.*

In this chaper, the coloured edge is used to desribe the colour information of an image. Taking a different approach, previous work usually generates the coloured sketch by colouring different local regions [Cheng et al. 2016; Fuentes & Saavedra 2021]. This kind of sketch is more intuitive than the sketch with coloured edges because the colours of an practical image are originally distributed in different regions rather than edges. However, the sketch with coloured regions cannot effectively find the feature embedding space. There are two reasons. First, it is difficult to judge whether the coloured region indicates the colour information only or both colour and shape information. For the sketch with coloured edges, there is no such ambiguity because the colour only exists on the edge. Second, strictly speaking, the sketch with coloured regions is not a sketch which consists of several strokes, and it is more like an image. The domain discrepancy between this type of sketch and image is very small, and it is not a difficult retreival problem compared with the challenges mentioned in Section 1.1.

The training procedures of the two stages are introduced as follows:

Stage 1 This stage is to pre-train the CNN before it is plugged into Stage 2. The UT Zappos50K [Yu & Grauman 2014] dataset is used as the pre-training data, which contains over 50 thousand shoe images with 4 major categories, i. e., shoes, sandals, slippers, and boots. They are further subdivided into 17 classes with different styles and a one-hot encoding is used to generate labels including the information of the class ($N_s = 17$) and the colours ($N_c = 10$). There are 44,995 training images and 4,988 testing images in total. After generating the edge maps from images, the CNN is trained for a classification task with the classical binary cross entropy loss function, which is formulated as

$$L_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}\{y_i \log \sigma(f_{sc}^i) + (1 - y_i)\log[1 - \sigma(f_{sc}^i)]\} \tag{4.9}$$

where f_{sc}^i and y_i are the ith SC feature and corresponding label in the mini-batch N, respectively, and $\sigma(f_{sc}^i)$ is the Sigmoid probability defined as

$$\sigma(f_{sc}^i) = \frac{1}{1 + \exp(-f_{sc}^i)} \tag{4.10}$$

Stage 2 This stage is to fine-tune the pre-trained CNN from Stage 1 using the created MCSBIR dataset. There are 232 pairs of sketches and images in the dataset. 168 pairs in the dataset are used for training and the rest are used for testing. In order to increase the number of training triplets, two triplet groups, i. e., easy triplets and hard triplets, are constructed. In the easy triplets, the image with the same shape and colour as the query sketch q is selected to be p^+ and the rest images are p^-. In the hard triplets, the image with the same shape and different colours is selected to be p^+ and the images with different shapes are p^-. As a result, 99,574 training triplets are generated. After generating the edge maps from images, the triplets are inputted into Stage 2, and the pre-trained CNN from Stage1 is fine-tuned for the retrieval task with the triplet loss function in Eq. (4.8).

4.3 Experiments

In this section, several experiments are conducted to demonstrate the effectiveness of the proposed MCSBIR method. First, the implementation details are introduced, and then two baselines are designed to compare with the proposed MCSBIR method. Finally, the results of the comparison and further analysis on the proposed MCSBIR method are presented.

4.3.1 Implementation details

The proposed MCSBIR method is implemented on Pytorch with a NVIDIA GeForce GTX 2080 Ti GPU, and the following experimental settings are used.

- ResNet50 [He et al. 2016] is applied as the CNN structure in Stage 1 and Stage 2 and the sketches and images (edge maps) are uniformly resized into a resolution of $136 \times 136 \times 3$. The Adam optimizer is employed for both stages and the weight decay is set to 0.

- In Stage 1, the learning rate is 1×10^{-5}, the batch size is 16, and the number of training epochs is set to 10.

- In Stage 2, the learning rate and batch size are 1×10^{-5} and 64, respectively, and the number of training epochs is 25. Moreover, the margin λ and the trade-off parameter β are set to be 0.1 and 0.7, respectively.

The test dataset has 64 pairs of sketches and images. To evaluate the performance of the proposed MCSBIR method in the test dataset, the following two evaluation metrics are adopted.

- Precision-at-K (Prec @ K): It quantifies the cumulative matching accuracy at various ranks and shows the percentage of query sketches whose true-match images are ranked at the top K [Yu et al. 2016].
- Mean Reciprocal Rank (MRR): It is a relative score that calculates the mean of the inverse rank of the correct retrieved images [Fuentes & Saavedra 2021]. MRR can be represented as [Burges et al. 2006]

  $$MRR = \frac{1}{N_q} \sum_{i=1}^{N_q} \frac{1}{r_i} \tag{4.11}$$

  where N_q is the number of query sketches and r_i is the rank position in the retrieval results of the ith query sketch.

4.3.2 Baselines

Recently, there have been a few studies dealing with colour sketches for the image retrieval. In order to indicate the effectiveness of the proposed MCSBIR method, two baselines are implemented to compare with the proposed MCSBIR method.

Baseline 1 This baseline uses the SCSBIR method [Xia et al. 2019], which retrieves the images by first matching the shapes of black-and-white images (edge maps) and then re-ranking them by calculating their colour similarities. In the shape matching process, the training dataset is UT Zappos50K, which is the same data resource as the created MCSBIR dataset. Therefore, their deep network for the shape feature embedding can be directly used in MCSBIR dataset. In the colour matching process, three different baseline solutions are used: (1) the RGB colour matching method using Eq. (3.11) (Baseline 1-RGB), (2) the HSV colour matching method using Eq. (3.12) (Baseline 1-HSV), and (3) no colour matching (Baseline 1-NoColour).

Baseline 2 This baseline applies the Sketch-Qnet [Fuentes & Saavedra 2021], which represents the shape and colour information in one CNN model. Since the Sketch-Qnet adopts the heterogeneous network, which cannot be directly compared with the proposed MCSBIR method, their core algorithm, i. e., the quadruplet loss, instead of the proposed loss function is used in Stage 2. The quadruplet loss is defined as

$$Loss_{baseline2} = CE_q + CE_{p^+} + CE_{p^{+-}} + CE_{p^-} + \beta(L_{t1} + L_{t2}) \quad (4.12)$$

where $\beta = 2$, which is increased by 0.5 each epoch, CE_i is the cross-entropy loss of input i, q is a query sketch, p^+ is an image with the same shape and colour as q, p^{+-} is an image with the same shape as q but has a different colour, and p^- is an image with a different shape as q. L_{t1} and L_{t2} are two triplet losses [Fuentes & Saavedra 2021]:

$$\begin{aligned} L_{t1} &= \max[0,\ D^+ - D^{+-} + \alpha\lambda] \\ L_{t2} &= \max[0,\ D^{+-} - D^- + (1-\alpha)\lambda] \end{aligned} \quad (4.13)$$

where D^i is the Euclidean distance between q and p^i, and α and λ are a trade-off parameter and a margin, respectively. α is set to 0.1 and λ

is set to 1.5 by trial and error, which can generate a better result than other values.

4.3.3 Results

Comparisons against baselines

First, the performance of the proposed MCSBIR method and the two baselines is illustrated. Figure 4.9 shows the results for cumulative retrieval accuracy at top 1 to 10. The following observations are obtained. (1) The proposed MCSBIR method has better performance than other baselines at top K ($K = 1, 2, \ldots, 10$). (2) The baseline 1-NoColour, RGB and HSV have the same precision at top 10 because they are based on the same top 10 results from the shape matching. (3) The baseline 2 presents a better result at top 1 and 2 compared with baseline 1-NoColour. In addition, the MRR for the proposed MCSBIR method and the two baselines is calculated, as shown in Table 4.2, which indicates the MRR score of the proposed MCSBIR method is better than others. As a result, the two comparisons indicate that the proposed MCSBIR method is effective in retrieving the corresponding images of the query sketches.

Table 4.2: *MRR results of the proposed MCSBIR method and the two baselines.*

Method	MRR
Baseline 1-NoColour	0.4330
Baseline 1-RGB	0.6047
Baseline 1-HSV	0.8324
Baseline 2	0.5459
MCSBIR method	0.8766

Some examples of the retrieval results of the proposed MCSBIR method and the two baselines are shown in Figure 4.10. The top 6 retrieved im-

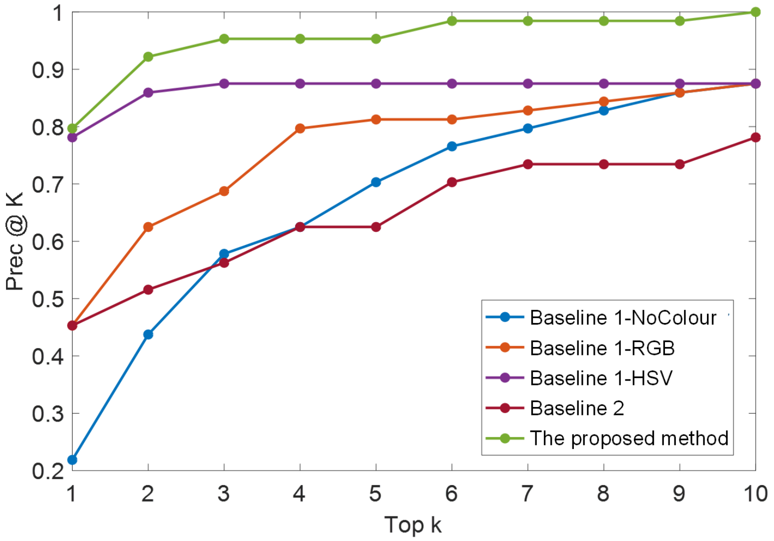


Figure 4.9: *Prec @ K for $K = 1$ to 10 of two baselines and the proposed MCSBIR method.*

ages are ranked from the left to the right. The red square indicates the correct result with the same shape and colour and the blue square indicates the result with the same shape but different colours. It can be seen that the proposed MCSBIR method is more capable of not only retrieving the correct image but also finding other images with the same shape as the correct one.

Further analysis on the proposed MCSBIR method

As described in Section 4.2.2, the loss function of the proposed MCSBIR method contains two hyper-parameters, i. e., the margin λ and the trade-off parameter β. For the margin λ, a larger value can make a stronger distinguishability between $D(q, p^+)$ and $D(q, p^-)$ but a harder training process. For the parameter β, a larger value can train the network to be easier to recognize different shapes but harder to distinguish different colours. In order to investigate the influences of λ and β on the retrieval precision, an experiment by varying different values of the two hyper-

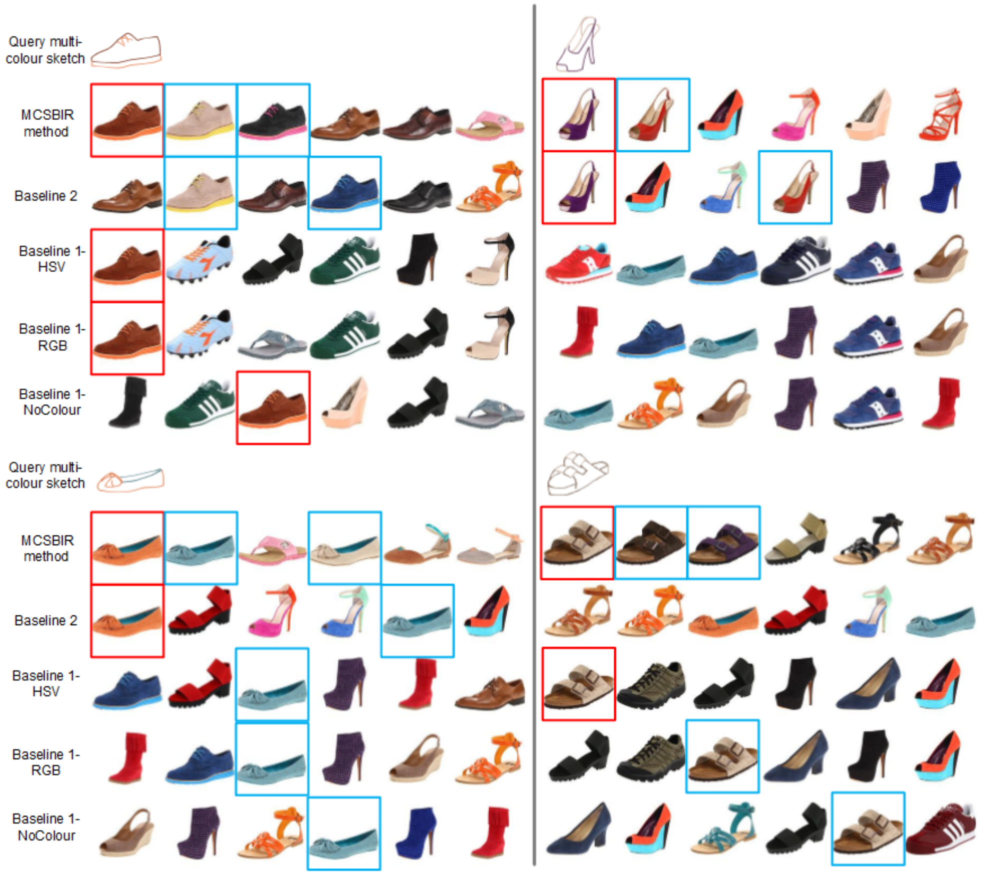


Figure 4.10: *Some examples of the retrieval results of the proposed MCSBIR method and the two baselines.*

parameters is conducted. In Figure 4.11, (a) shows the MRR scores with different λ, i. e., λ=0, 0.1, 0.25, 0.5, 0.75 and 1 when β=0.7, and (b) shows the MRR scores with β=0.5, 0.6, 0.7, 0.8, 0.9 and 1 when λ=0.1. The results indicate that the trend of the MRR score increases initially and follows by a fall when the values of λ and β increase. Therefore, setting λ=0.1 and β=0.7 leads to the best performance.

In addition, the contribution of each stage of the proposed MCSBIR method is investigated. Table 4.3 shows the retrieval precision at top 1, top 5 and top 10 and the MRR score of the proposed MCSBIR method using Stage 1 only, Stage 2 only and Stages 1+2, respectively. The results indicate that Stages 1 and 2 are effective and both of them are necessary

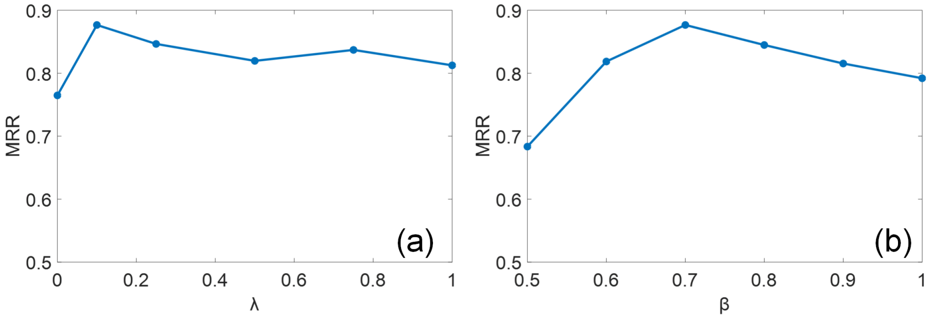


Figure 4.11: *The MRR scores with different values of* λ *(a) and* β *(b).*

in improving the retrieval performance of the proposed MCSBIR method.

Table 4.3: *Contributions of different stages.*

	Top 1	Top 5	Top 10	MRR
Stage 1	0.2656	0.7031	0.9063	0.4514
Stage 2	0.3750	0.8594	0.9531	0.5849
Stage 1+2	0.7969	0.9531	0.9844	0.8766

4.4 Summary

In this chapter, a multi-colour sketch based image retrieval method is proposed. First, a MCSBIR dataset is created, and a new feature embedding is designed to clearly describe the shape and colour information within a single feature vector. Then, a triplet loss function based on a new Euclidean distance is developed, which separates shape and colour features, and a two-stage network architecture is designed to learn the proposed feature embedding. The experiments show that the proposed MCSBIR method has a better retrieval performance compared with two baselines and the further analysis also indicates the effectiveness of the proposed MCSBIR method.

Chapter 5

Sketch-based 3D shape retrieval

Sketch-based 3D shape retrieval is one of the most important research topics in the field of the sketch-based retrieval since the domain discrepancy between sketches and 3D shapes is much larger than that of sketches and images. The big challenge is how to find a joint feature embedding space of sketches and 3D shapes, and improve the retrieval accuracy. In this chapter, this challenge is addressed, and a semantic similarity metric learning is proposed for sketch-based 3D shape retrieval. The proposed teacher-student guided and sketch-based 3D shape retrieval (TSS3DSR) method is described in Section 5.1 and the experimental results and analysis are presented in Section 5.2, and finally the summary is drawn in Section 5.3.

5.1 TSS3DSR method

5.1.1 Network architecture

The network architecture of the proposed TSS3DSR method is described in Figure 5.1, which consists of a teacher network and a student network. Since sketches are abstract simple lines with limited information and 3D shapes are surface-represented geometric objects with more details, 3D shapes are selected as the input of the teacher network and the semantic features are extracted from them to guide the training of the student network that takes sketches as input. By using the similarity loss to measure the cosine distance between sketches and 3D shapes, the features of sketches are optimized and gradually close to the pre-learned semantic features of 3D shapes during the training process of the student network.

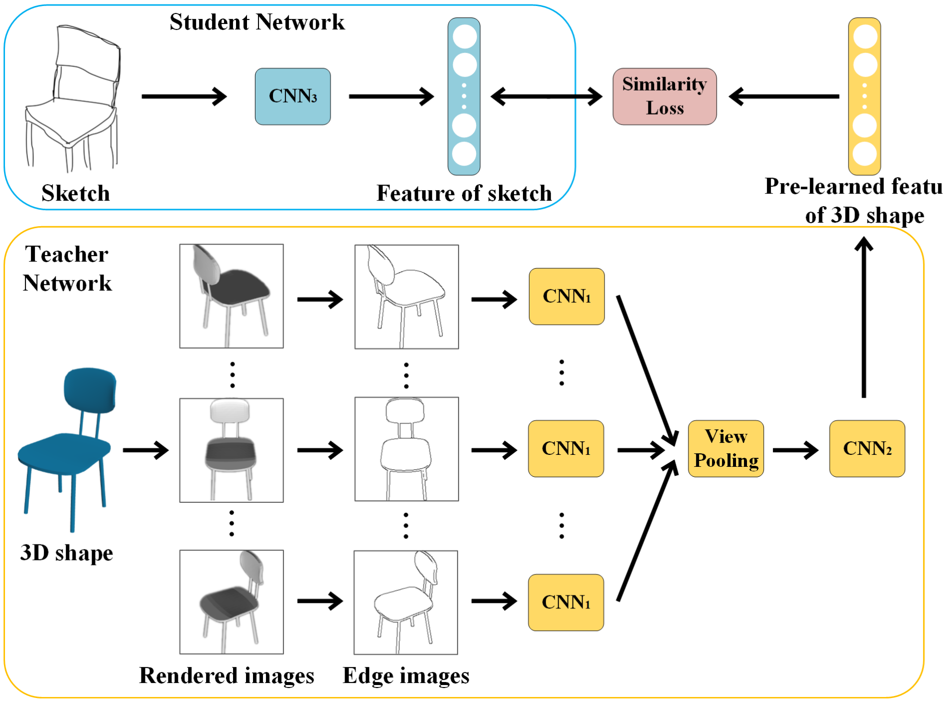


Figure 5.1: *The network architecture of the proposed TSS3DSR method.*

Teacher network

In the teacher network, the MVCNN [Su et al. 2015] architecture is applied, including CNN_1 and CNN_2, which are connected by a view-pooling layer, to represent multi-views of 3D shapes and extract the semantic features. There are three steps to generate the features of 3D shapes in the teacher network.

The first step is to render the 3D shape to 2D images in different views. The rendering strategy in [Su et al. 2015] is adopted. Assuming the input 3D shape is upright oriented along the Z axis in the XYZ coordinate system. 12 virtual cameras are placed around the 3D shape every 30 degrees. These cameras are elevated 30 degrees from the XY plane and the directions of these cameras are aimed at the centroid of the 3D shape. After rendering the 3D shape, 12 2D images with different views are generated, as shown in Figure 5.1.

The second step is to extract edge maps of the rendered images. Since there is still a big domain discrepancy between rendered images and sketches, it is difficult to find a joint feature embedding space. In order to reduce the domain discrepancy, the classic canny edge detector [Canny 1986] is adopted to extract the edges of rendered images, which are similar to the sketch lines.

The third step is to generate features for the 3D shapes. After obtaining the edge images of 3D shapes, these edge images are passed through CNN_1 separately to obtain view based features. Note that all branches of CNN_1 share the same parameters. In order to synthesize the information from all views into a single, element-wise maximum operation is used across the views in the view-pooling layer. Finally, these pooled feature maps are passed through CNN_2 to obtain the shape features.

CNN_1 and CNN_2 apply VGG11 structure [Simonyan & Zisserman 2014], as shown in Figure 5.2 . The VGG11 consists of five convolutional layers and three fully connected layers (The last layer is also the output). Each convolutional layer is followed by a 2×2 max-pooling with stride 2 and the notation $(k \times k,\ n)$ denotes a filter of size k and n channels. Note that CNN_1 and CNN_2 are not a completed VGG11 structure. CNN_1 only contains five convolutional layers while CNN_2 only contains three fully connected layers. The view-pooling layer is adopted between CNN_1 and CNN_2. The view-pooling process is similar to the max-pooling which is to calculate the maximum value for patches of a feature map. The difference is that the view-pooling calculates the maximum value for the elements at the same position of the feature maps from the 12 view branches.

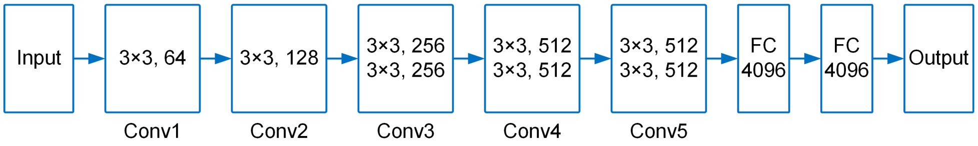


Figure 5.2: *The VGG11 structure.*

After finishing training the teacher network, all data of 3D shapes pass through the teacher network, and the pre-learned semantic features of 3D shapes are obtained.

Student network

In the student network, a transfer network CNN_3 is adopted to learn the semantic features of sketches. The input sketches are directly passed through CNN_3 to obtain the features. Note that the input sketches are black-and-white without any colour. The student network is trained according to the optimization objective function, i. e., the similarity loss, which is guided by the pre-learned semantic features of 3D shapes.

CNN_3 applies ResNet50 structure [He et al. 2016], as shown in Figure 4.7 which is introduced in 4.2.3. Since the input sketches are black-and-white, they can be regarded as gray scale images which have only one input channel for the Conv1 layer. In the run-time process, all test sketches are passed through the trained CNN_3 model to generate features. For each test sketch, the distances between its feature and all pre-learned features of 3D shapes are calculated, respectively, to find the most similar 3D shape according to the sketch. The calculation of the distance is introduced in next section.

5.1.2 Similarity loss

In order to find the desired 3D shape, it is always wanted that the extracted feature of the sketch is more similar to that of the same-class 3D shape and more dissimilar to that of the different-class 3D shape, i. e., maximizing the within-class similarity and minimizing the between-class similarity. However, a query sketch usually has tens or hundreds related 3D shapes with the same class label, and it is difficult to tell which 3D shape is more similar or dissimilar to the query sketch. Note that the aim here is to find 3D shapes belonging to the class labels of query sketches rather than find the most similar 3D shapes. Therefore, the focus is on extracting the class features rather than the individual features of 3D shapes. The class feature is the mean value of the pre-learned features of the 3D shapes in the same class. Cosine similarity is used to measure the distance between a sketch and a 3D shape, which is defined as

$$s = \frac{f_s \cdot f_c}{\|f_s\|_2 \, \|f_c\|_2} \tag{5.1}$$

where f_s is the sketch feature and f_c is the class feature of the 3D shape.

In a mini-batch with size N, we have N sketches and N_c pre-learned class features of 3D shapes. For each sketch i in the mini-batch, we calculate its cosine similarity with the class features of all 3D shapes. We denote the cosine similarity between the sketch and the class feature of the same-class 3D shape by s_p^i, i. e., the positive pair, and the cosine similarity between the sketch and the rest class features of 3D shapes by $s_n^i = \{s_1, s_2, \ldots, s_{N_c - 1}\}$, i. e., the negative pairs. In order to maximize the similarity score of the positive pair and minimize the similarity score of the negative pair, the similarity loss function is defined as:

$$L = \frac{1}{N} \sum_{i=1}^{N} [\max(s_n^i) - s_p^i + \lambda]_+ \tag{5.2}$$

where $[]_+$ is a ramp function and λ is a margin for a better similarity separation between positive and negative pairs.

The reason why to choose the maximum similarity score from the group of s_n^i to represent the negative pair in Eq. (5.2) is that it can ensure the scores of all negative pairs are smaller than the positive pair and also increase the difficulty of learning as the same effect of λ. Since it is difficult to optimize Eq. (5.2), a smooth approximation is adopted by using a modified LogSumExp function to replace $\max(s_n^i)$ and a softplus function to replace $[\cdot]_+$, which can be formulated as

$$\max(s_n^i) = \log \left[\sum_{n=1, n \neq p}^{N_c} \exp(r s_n^i) \right] \tag{5.3}$$

$$[\cdot]_+ = \log[1 + \exp(\cdot)] \tag{5.4}$$

where r is a scale factor.

Substituting Eqs. (5.3) and (5.4) into (5.2), the smooth similarity loss function is obtained as

$$L_{smooth} = \frac{1}{N}\sum_{i=1}^{N} \log\left\{1 + \exp\left[\log\left(\sum\nolimits_{n=1,n\neq p}^{N_c} \exp(rs_n^i)\right) - s_p^i + \lambda\right]\right\} \tag{5.5}$$

By training the student network with L_{smooth}, the sketch feature f_s is gradually close to the pre-learned class feature f_c of the same-class 3D shapes and keeps away from that of the different-class 3D shapes simultaneously.

5.2 Experiments

5.2.1 Datasets

The proposed TSS3DSR method is evaluated on a frequently-used benchmark dataset, i. e., SHREC'13 [Li et al. 2013a], for sketch-based 3D shape retrieval. Some examples of sketches and corresponding 3D shapes in the dataset are shown in Figure 1.5. The dataset is built by collecting large-scale hand-drawn sketches from TU-Berlin sketch dataset [Eitz et al. 2012a] and 3D shapes from Princeton Shape Benchmark [Shilane et al. 2004], which consists of 90 classes including 7,200 sketches and 1,258 shapes. In each class, there are a total of 80 sketches, and 50 of which are for the training and the rest are for the test. The number of 3D shapes varies in different classes. For example, the largest class is 'airplane', which has 184 3D shapes, and there are 12 classes containing only 4 3D shapes.

5.2.2 Implementation details

The proposed TSS3DSR method is implemented on Pytorch with two NVIDIA GeForce GTX 2080 Ti GPUs.

Network structure The structure is illustrated in Figure 5.1. The teacher network adopts the MVCNN [Su et al. 2015] architecture and the CNN_1 and CNN_2 use the VGG-11 network [Simonyan & Zisserman 2014]. In the student network, CNN_3 utilizes the ResNet-50 network [He et al. 2016].

Prepossessing The prepossessing includes the network pre-training and data processing. The teacher network is pre-trained on ImageNet [Deng et al. 2009] with 1k categories, and then fine-tuned on all edge images of the 3D shapes. The student network is first pre-trained for the classification task based on a part of QuickDraw dataset [Ha & Eck 2017] with 3.45 million sketches in 345 categories, and then fine-tuned on the training dataset of sketches according to minimize Eq. (5.5). For the data processing, the sketch images and the edge images of 3D shapes are uniformly resized into a resolution of $224 \times 224 \times 1$.

Parameter settings In the teacher network, the learning rate and batch size are 5×10^{-5} and 8, respectively, and the number of training epochs is set to 20. In the student network, the learning rate and batch size are 1×10^{-4} and 48, respectively, and the number of training epochs is 10. Moreover, the margin λ and the scale factor r are set to be 0.15 and 64, respectively. The Adam is employed as an optimizer for both networks and the weight decay is set to 0.

5.2.3 Experimental results

Some retrieval results on the SHREC'13 dataset are shown in Figure 5.3. The query sketches are listed on the left including the class of chair, bicycle, piano, table, palm tree and sea turtle, and the retrieved top 8 3D shapes are listed on the right according to the ranking of similarity scores. In Figure 5.3, the proposed TSS3DSR method is effective in retrieving the corresponding 3D shapes of the query sketches. The reasons for generating incorrect results are the limited number of 3D shapes (e. g., the classes of bicycle and sea turtle, which only contain 7 and 6 3D shapes in the dataset, respectively) and the high similarity score of similar shapes from different classes (e. g., the couch and bench shapes, which get high similarity scores according to the query sketch of piano).

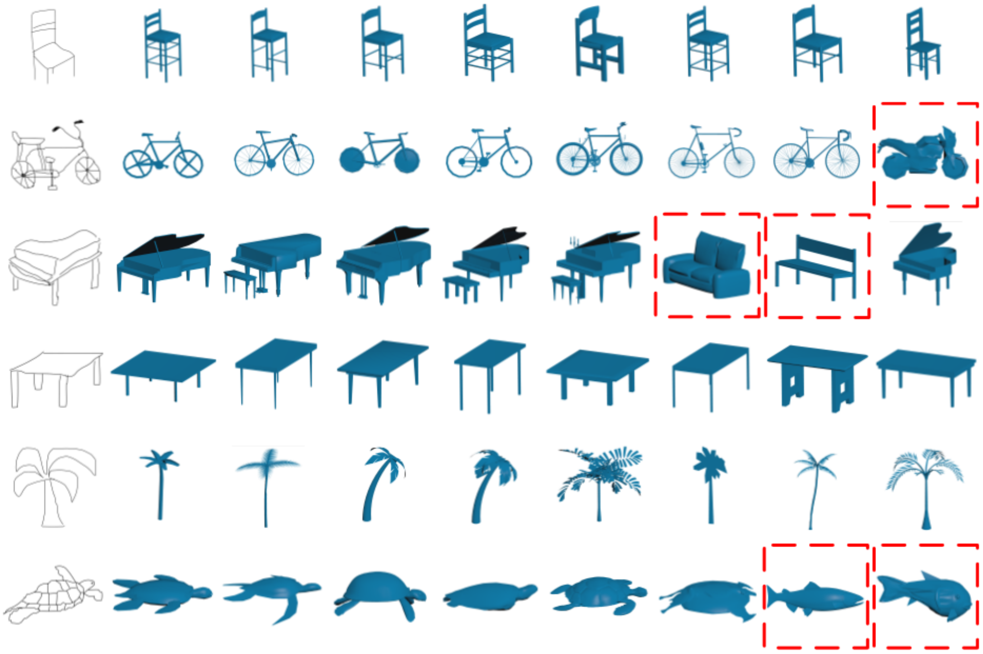


Figure 5.3: *Some examples of retrieval results. The left column is the query sketches and the right columns are the top 8 retrieved 3D shapes, and the wrong results are highlighted by red dashed squares.*

In order to demonstrate the effectiveness of the proposed TSS3DSR method, it is compared with several state-of-the-art methods, including SBR-VC [Li et al. 2013a], Siamese [Wang et al. 2015], Shape2Vec [Tasse & Dodgson 2016], DCML [Dai et al. 2017], LWBR [Xie et al. 2017], DCA [Chen & Fang 2018], SEM [Qi et al. 2018] and DSSH [Chen et al. 2019]. In addition, the widely-used evaluation metrics for the sketch-based 3D shape retrieval, including the nearest neighbor (NN), first tier (FT), second tier (ST), E-measure (E), discounted cumulated gain (DCG) and mean average precision (mAP) are adopted, which are introduced below [Li et al. 2014a]

- NN: It measures the accuracy of the top 1 retrieval list.
- FT: Assume there are C relevant 3D shapes in the dataset, FT is the recall of the top $C-1$ retrieval list. The benchmark of SHREC'13 sets C to be 20.
- ST: Similarly, ST is the recall of the top $2(C-1)$ retrieval list.
- E: It is a composite measure of the precision P and recall R for the first 32 of retrieved results, which is defined as $E = 2/(1/P + 1/R)$.
- DCG: It is defined as the normalized summed weighted value related to the positions of the relevant 3D shapes:

$$DCG = \frac{DCG_n}{1 + \sum_{j=2}^{C} \log_2 j} \tag{5.6}$$

where n and C represent the total number of 3D shapes in the dataset and the relevant class, respectively, and

$$DCG_i = \begin{cases} G_1, & i = 1 \\ DCG_{i-1} + \frac{G_i}{\log_2 i}, & i = 2, 3, ..., n \end{cases} \tag{5.7}$$

where $G_i = 1$ if the ith retrieved 3D shape belongs to the same class as the query sketch, otherwise $G_i = 0$.

- mAP: It is to find the average precision for each query sketch and compute the mean of average precisions over all query sketches.

Table 5.1 shows the quantitative comparison of the proposed TSS3DSR method with the state-of-the-art methods on the SHREC'13 dataset. Except for the DSSH, it is clear to see that the proposed TSS3DSR method achieves the best performance than the state-of-the-art methods for all the evaluation metrics. Compared to the latest method DSSH, the proposed TSS3DSR method performs better or equally in the NN, E and DCG metrics.

Table 5.1: *The comparison of our method and the state-of-the-art methods on the SHREC'13 dataset.*

Method	NN	FT	ST	E	DCG	mAP
SBR-VC	0.164	0.097	0.149	0.085	0.348	0.114
Siamese	0.405	0.403	0.548	0.287	0.607	0.469
Shape2Vec	0.620	0.628	0.684	0.354	0.741	0.650
DCML	0.650	0.634	0.719	0.348	0.766	0.674
LWBR	0.712	0.725	0.785	0.369	0.814	0.752
DCA	0.783	0.796	0.829	0.376	0.856	0.813
SEM	0.823	0.828	0.860	0.403	0.884	0.843
DSSH	0.831	**0.844**	**0.886**	0.411	0.893	**0.858**
TSS3DSR method	**0.836**	0.833	0.883	**0.411**	**0.896**	0.853

The proposed TSS3DSR method is also visually compared with DSSH to show its advantages. As shown in Figure 5.4, for the hand and horse sketch examples, the retrieved 3D shapes with the proposed TSS3DSR method are more accurate than DSSH. First, the retrieved 3D shapes with mismatched details have a low-ranking in the proposed TSS3DSR method. For example, an unextended hand is ranked last in the proposed TSS3DSR method but ranked second in DSSH, and a horse with a lifted leg is ranked fourth in the proposed TSS3DSR method but ranked second in DSSH. Second, the incorrect dog models retrieved by the proposed TSS3DSR method have a similar shape to the correct horse model. However, the incorrect guitar models retrieved by DSSH have an extraordinarily different shape to the correct horse models.

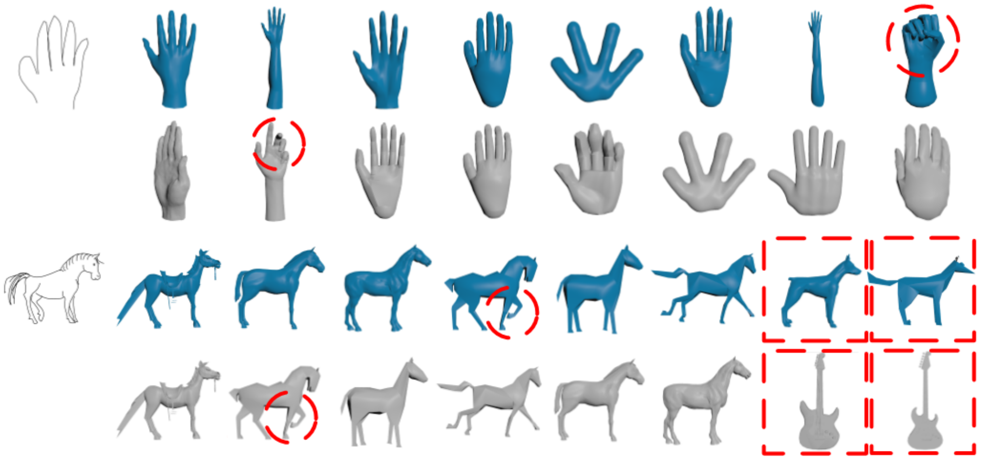


Figure 5.4: *The comparison of the proposed TSS3DSR method and DSSH [Chen et al. 2019] in two retrieval examples. The blue and gray colours denote the retrieval results of the proposed TSS3DSR method and DSSH, respectively, and the wrong results and mismatched details are highlighted by red dashed squares and circles, respectively.*

5.3 Summary

In this chapter, a novel semantic similarity metric learning method is proposed for sketch-based 3D shape retrieval, and a teacher-student strategy is used to obtain efficient networks for learning semantic similarity between sketches and 3D shapes. First, the pre-trained classification network is adopted as the teacher network to extract the semantic features of 3D shapes. Then, the student network is trained by using the pre-learned features of 3D shapes with a similarity loss function. Finally, the semantic features of sketches are learnt. As a result, the proposed TSS3DSR method effectively maximizes the within-class similarity and minimizes the between-class similarity. The experiments show that the proposed TSS3DSR method performs better than the state-of-the-art methods.

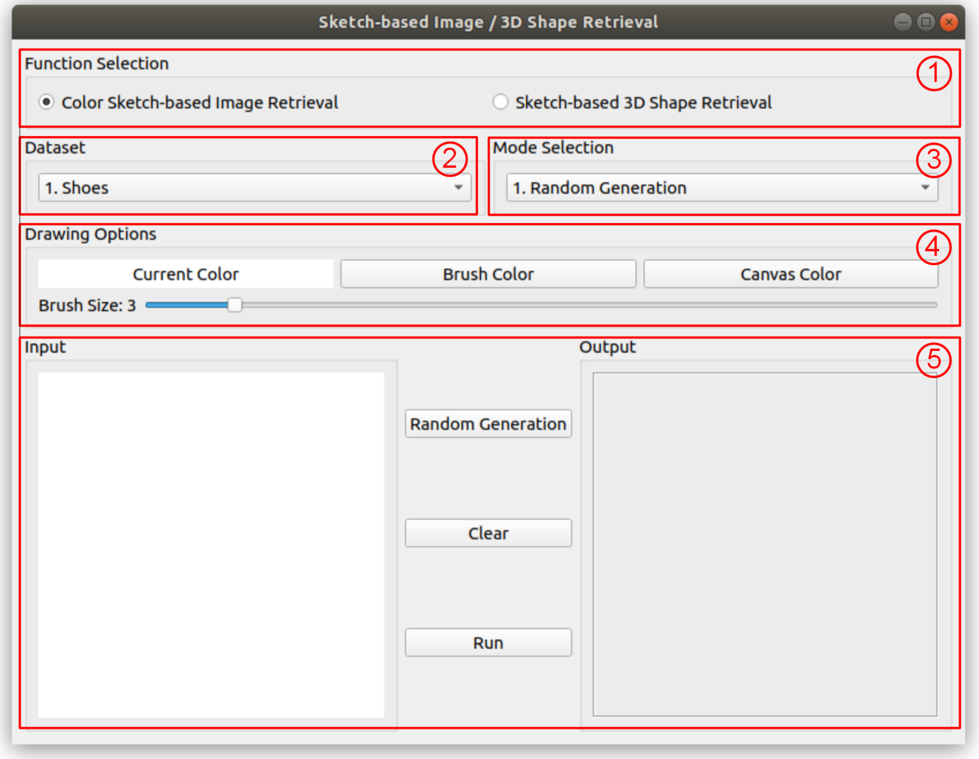
Chapter 6

User interface

In order to visually present the developed colour sketch-based image retrieval and sketch-based 3D shape retrieval techniques and promote the potential commercial applications of them, a user interface is developed in which users can draw sketches and retrieve images and 3D shapes. Since the proposed sketch-based retrieval techniques use the Pytorch library based on the Python programming language, the user interface also needs to be developed using the same programming language. PyQt5 is a popular library, which allows users to write graphical user interface (GUI) applications using Python [Summerfield 2007]. Therefore, PyQt5 is applied to design the user interface. In this chapter, the functions of the user interface are introduced in Section 6.1, and the operation procedure is presented in Section 6.2.

6.1 Functions of the user interface

The user interface consists of five parts, i. e., Function Selection, Dataset, Mode Selection, Drawing Options, and Input and Output zones, as



Function Selection

● Color Sketch-based Image Retrieval ○ Sketch-based 3D Shape Retrieval

Dataset

1. Shoes

Mode Selection

1. Random Generation

Brush Size: 3

Figure 6.1: *The user interface.*

shown in Figure 6.1. The function of each part is introduced below.

Function Selection Two function types are provided for sketch-based retrieval, which are "Colour Sketch-based Image Retrieval" and "Sketch-based 3D Shape Retrieval". Users can choose different function types to retrieve images or 3D shapes. The techniques behind the two functions are introduced in Chapters 4 and 5, respectively.

Dataset The combo box in this part includes two items, i. e., Shoes and SHREC-13, as shown in Figure 6.2(a). The Shoes dataset is the MCSBIR dataset from Chapter 4 and it is prepared for the function "Colour Sketch-based Image Retrieval". The SHREC-13 datasets are created from [Li et al. 2013a], which are prepared for the function "Sketch-based 3D Shape Retrieval"

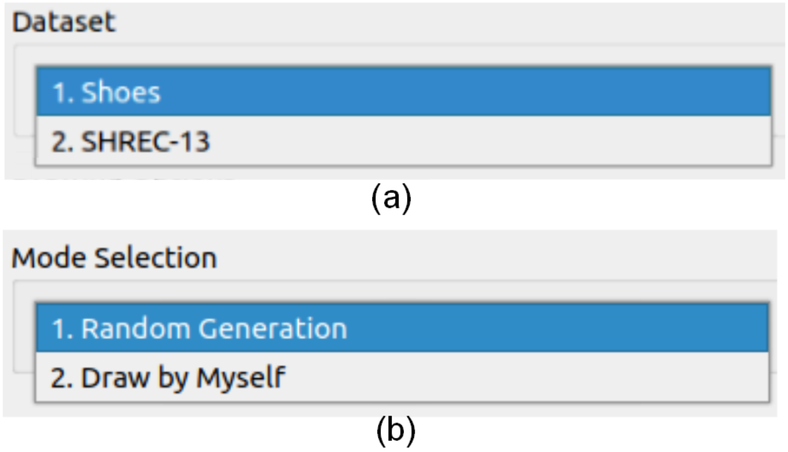


Figure 6.2: *The combo boxes of Dataset and Mode Selection.*

Mode Selection The combo box includes two items, i. e., Random Generation and Draw by Myself, as shown in Figure 6.2(b). Random Generation is to randomly select a sketch from the chosen dataset, and Draw by Myself is to draw the sketch by users themselves.

Drawing Options For users to draw different colour sketches by themselves, four sub-functions, i. e., Brush Colour, Canvas Colour, Brush Size and Current Colour are designed. The Brush Colour is to set the colour of the brush, which is black by default, the Canvas Colour is to set the colour of the canvas in the input area, which is white by default, and the Brush Size decides the thickness of the drawing line, which is set to 3 by default. The range of the size is from 1 to 20. By clicking the button Brush Colour or Canvas Colour, a colour palette will be displayed and any desired colour can be selected as the brush colour or canvas colour, as shown in Figure 6.3. The Current Colour is a place to predictively display the selected colours before drawing, in which the background colour represents the canvas colour and the font colour represents the brush colour. Some examples with different brush colours, canvas colours and brush sizes are shown in Figure 6.4.

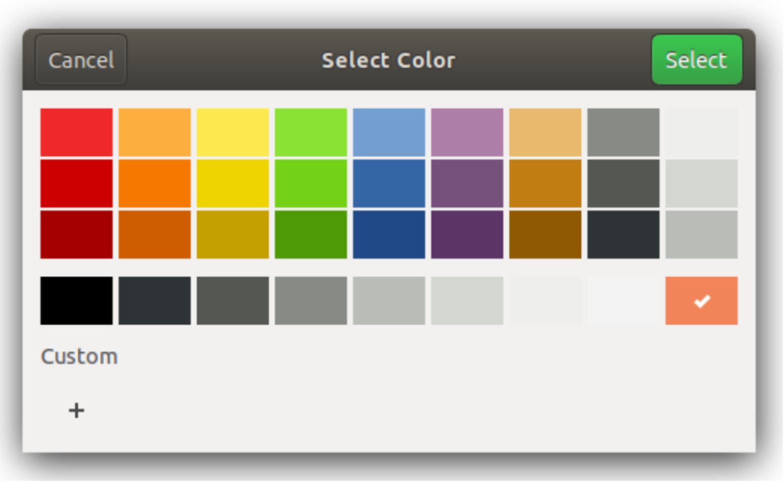
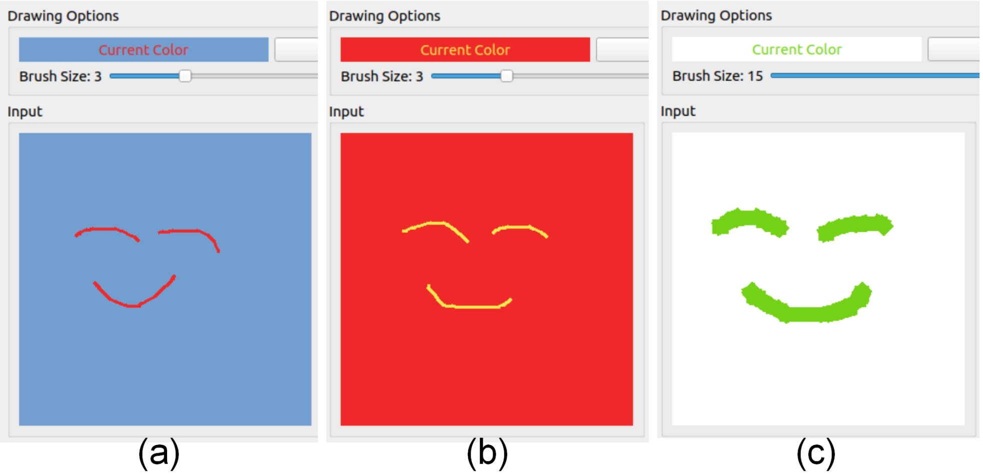


Figure 6.3: *The colour palette.*



Brush Size: 3

Brush Size: 3

Brush Size: 15

Figure 6.4: *Some examples with different brush colours, canvas colours and brush sizes. (a) The brush is red, the canvas is blue, and the brush size is 3. (b) The brush is yellow, the canvas is red, and the brush size is 3. (c) The brush is green, the canvas is white, and the brush size is 15.*

Draw and Generation Zone This zone includes the input display area, the output display area and three function buttons, i. e., Random Generation, Clear and Run. The input display area is a square place to draw and display the sketch. The output display area is a place to show the top six retrieved results as well as their names. The function of

Random Generation is to randomly generate a sketch from the selected dataset and display it in the input area, and the function of Run is to generate the top six results and display them in the output area. The button Clear can clean the content in the input and output display areas simultaneously.

6.2 Operation procedure

The procedure of using the user interface is mainly divided into six steps:

- Step 1: Select the function of the sketch-based retrieval. If the user would like to retrieve images, they can choose the Colour Sketch-based Image Retrieval. Sketch-based 3D Shape Retrieval is the choice for retrieving 3D shapes.

- Step 2: Select the retrieval dataset. Once the dataset is selected, the random sketches and retrieval results will be generated from this dataset. If "Colour Sketch-based Image Retrieval" is chosen in Step 1, users can only select the Shoes dataset. If "Sketch-based 3D Shape Retrieval" is chosen, users can only select the SHREC-13 dataset. Although there are only two datasets for now, more datasets will be involved in the user interface in the future.

- Step 3: Select the drawing mode in Mode Selection. If users would like to draw their desired sketches, they can choose Draw by Myself. If Random Generation is chosen, Step 4 can be skipped.

- Step 4: Select the desired colours for the drawing brush and canvas, and select the desired size of the brush.

- Step 5: If Random Generation is chosen in Step 3, users can click the function button Random Generation, and the system will automatically generate a sketch from the selected dataset and display it in the input area. Otherwise, users need to draw by themselves on the canvas in the input area. The way of drawing is to press the left mouse button and move the mouse on the canvas. During the drawing process, users can change the brush colour at any time to create multi-colour sketches, and they can clean the input display area by clicking the function button Clear and redraw the sketch.

- Step 6: After finishing the sketch in the input area, users can click the button Run to retrieve similar images or 3D shapes according to the selected function type. The top six results with their names from the selected dataset in Step 2 will be displayed in the output display area. Figures 6.5 and 6.6 show the retrieved top six results of "Colour Sketch-based Image Retrieval" and "Sketch-based 3D Shape Retrieval" functions according to the input sketches, respectively.

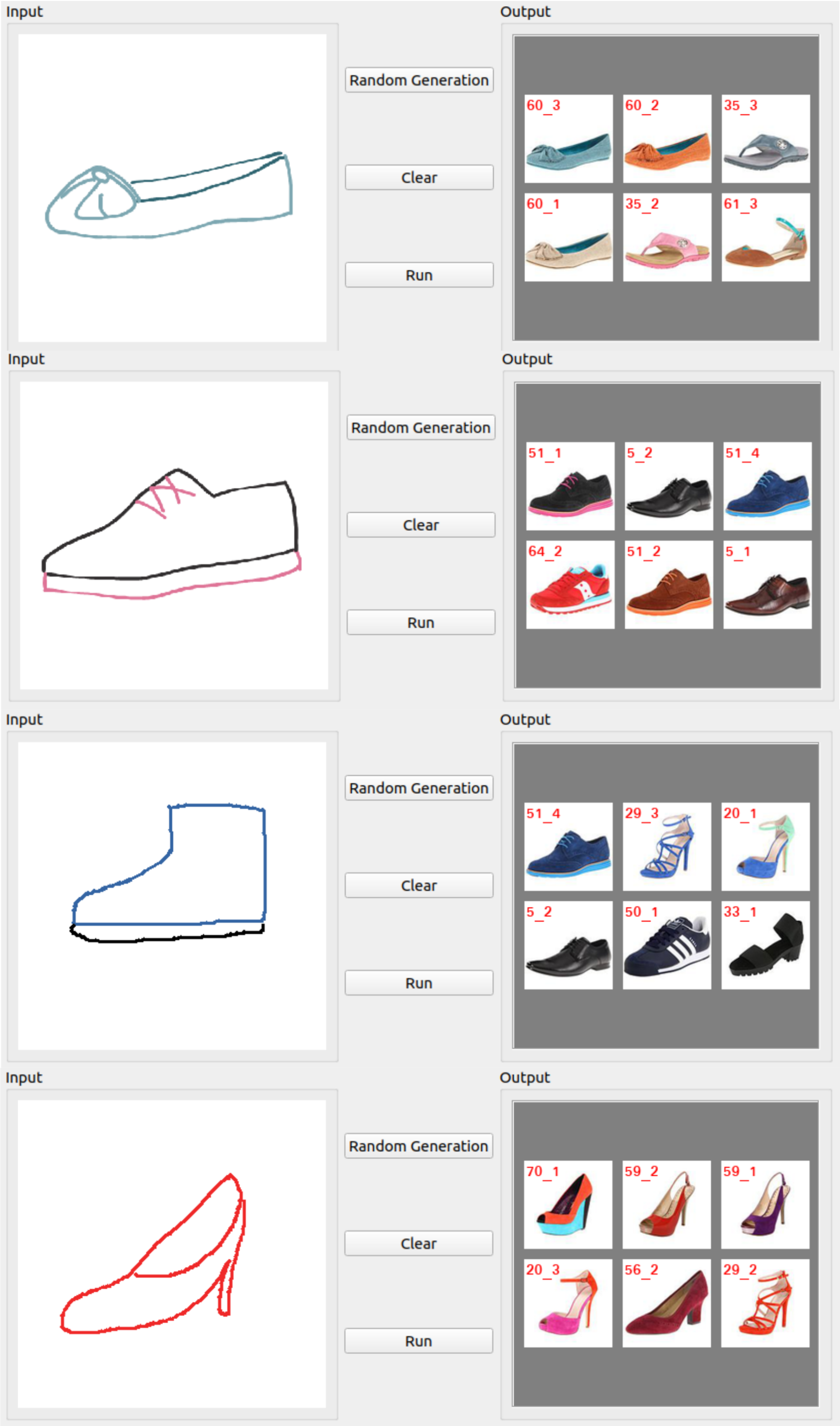


Figure 6.5: *Retrieved top six results of the "Colour Sketch-based Image Retrieval" function. In the first two rows, the two query sketches are randomly generated from the Shoes dataset. In the last two rows, the two query sketches are drawn in real time.*

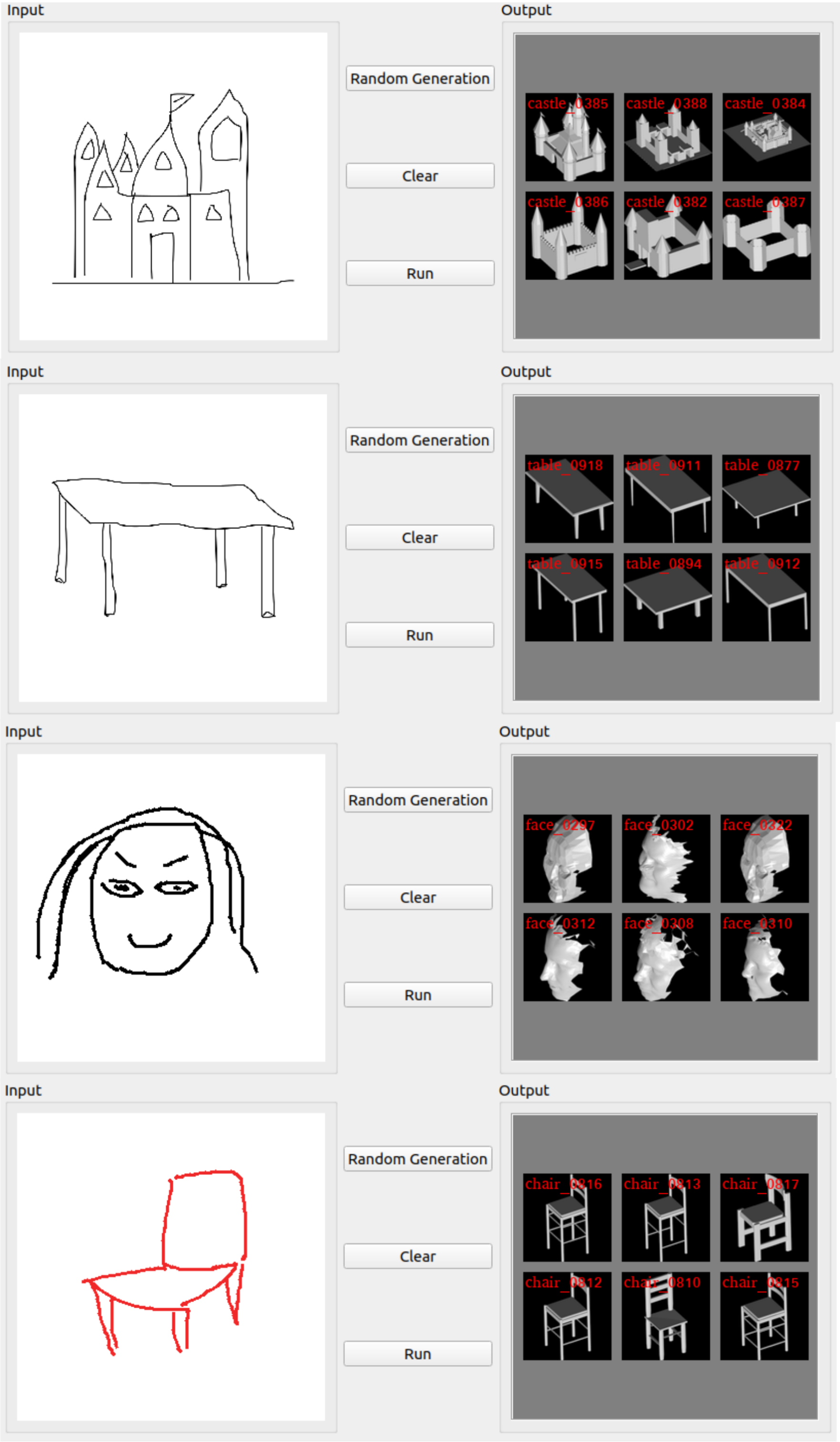


Figure 6.6: *Retrieved top six results of the "Sketch-based 3D shape Retrieval" function. In the first two rows, the two query sketches are randomly generated from the SHREC-13 dataset. In the last two rows, the two query sketches are drawn in real time.*

Chapter 7

Conclusion and future work

7.1 Conclusion

The sketch can better describe what humans see and provide more information for the retrieval task compared with the text. Due to the practical availability of large-scale sketch datasets and the development of computer performance and deep learning approaches, sketch-based retrieval techniques have flourished in recent years. In this thesis, the challenges of colour sketch-based image retrieval and sketch-based 3D shape retrieval have been tackled.

In order to consider the colour information in the retrieval task, the single-colour sketch based image retrieval method has been first proposed in Chapter 3. A single-colour sketch-image dataset has been created based on the proposed dominant colour extraction method, and a ranking method combined with the shape similarity matching and colour similarity matching has been developed, which makes the retrieval results get better matching in appearance. Compared with previous retrieval methods based on black-and-white sketches, the proposed method has better retrieval accuracy. However, practical images are usually multi-coloured and the single-colour sketch cannot completely and accurately represent the colour information of an image. Therefore, the multi-colour sketch based image retrieval method has been further developed to tackle the description of multiple colours in the sketch in Chapter 4. The first multi-colour sketch-image dataset has been built and the two-stage network architecture has been designed. A novel feature embedding for explicably describing the shape and colour information and a triplet loss function with separated shape and colour features have been proposed to generate an end-to-end solution for the MCSBIR task. In the experiments, two baselines have been designed to compare with the proposed method, and the results have shown that the proposed method performs better in retrieving images.

Since the domain discrepancy between sketches and 3D shapes is much bigger than that of sketches and images, Chapter 5 has focused on the solution for reducing the domain discrepancy. The sketch-based 3D shape retrieval has been solved by a novel metric learning network using the teacher-student strategy, which is capable of reducing computational cost and improving efficiency. A similarity loss function has been developed to optimize the semantic feature distance between sketches and 3D shapes. Experiment results based on a large benchmark dataset have demonstrated the effectiveness of the proposed TSS3DSR method. To visually present the developed colour sketch-based image retrieval and sketch-based 3D shape retrieval, a user interface has been developed in Chapter 6, which can be used to promote the potential commercial applications of the techniques developed in this thesis.

In this thesis, the challenges mentioned in 1.1 have been addressed and the research objectives have been achieved, but there are still some limitations. First, although the sketch is convenient and user-friendly for users to draw on touch screens, drawing a clear and well-expressed sketch still needs drawing skills to some extent. If the sketch is disorganized to describe users' desired object, it is difficult to retrieve the correct images or 3D shapes no matter how accurate the retrieval method is. This limitation inherently exists in the drawing process of sketch, which cannot be solved in the retrieval method. In order to help users to draw sketches well, some supplementary means are necessary such as the shake correction of drawing and the provision of exemplar sketches. Another limitation is that the developed sketch-based retrieval techniques are constrained by the datasets, which means the proposed retrieval methods are not generative for different retrieval tasks. For example, the proposed MCSBIR method is designed for retrieving shoe images, which cannot retrieve bags, clothes or other kinds of images. In order to develop a generative model, a comprehensive and large-scale dataset needs to be created.

7.2 Future work

This thesis has investigated sketch-based retrieval, and developed a few novel methods to achieve colour sketch-based image retrieval and sketch-based 3D shape retrieval. Some further possible work can be considered in the future.

- The benchmark of the colour sketch-based image retrieval: Although the proposed SCSBIR and MCSBIR methods as well as a few studies [Bui & Collomosse 2015; Fuentes & Saavedra 2021] tried to deal with image retrieval based on colour sketches, there is no general evaluation metric to judge the similarity of shapes and colours. For example, in the situation where one retrieved image has a similar shape to and same colour as the query sketch and another image has the same shape as and similar colour to the query sketch, an evaluation metric to evaluate which one has a higher score of the similarity has not been proposed. Therefore, how to figure out the benchmark of colour sketch-based image retrieval is an unsolved problem. One solution under consideration is to adopt a large-scale user survey to analyze the human's sensitivity for slightly different shapes and colours, and design a reasonable tolerance range for shape and colour differences in sketch-based retrieval.

- Retrieving the images with complex content: In this thesis, the proposed SCSBIR and MCSBIR methods as well as other studies [Yu et al. 2016; Song et al. 2017; Fuentes & Saavedra 2021] only retrieve the images with one object in the pure white background. In practice, many images have complex content such as the worn shoe images or several objects in a single image. In order to make the proposed MCSBIR retrieval method more suitable for practical applications, the following work is to develop an improved MCSBIR method based on a pixel-level segmentation. The feature embedding will include not only the shape and colour information, but also the information of the position, category and size of all objects in the image.

- Descriptor of 3D shapes: Most of sketch-based 3D shape retrieval methods [Wang et al. 2015; Zhu et al. 2016; Chen & Fang 2018;

Chen et al. 2019] including the proposed TSS3DSR method describe the feature of a 3D shape by using the projected 2D images with different degrees. However, this projection will lose some information of the 3D shape and cause errors in the retrieval process. To improve the retrieval accuracy, a direct descriptor for 3D shapes should be investigated. Since the 3D shapes in the popular datasets like SHREC'13 [Li et al. 2013a] are triangle meshes, the latest work SubdivNet [Hu et al. 2021], which develops a CNN framework for 3D triangle meshes, provides a good idea to extract features from 3D shapes without projecting them into 2D images. This requires further investigation.

Bibliography

P. Arbelaez, et al. (2010). 'Contour detection and hierarchical image segmentation'. *IEEE transactions on pattern analysis and machine intelligence* **33**(5):898–916.

D. Bahdanau, et al. (2014). 'Neural machine translation by jointly learning to align and translate'. *arXiv preprint arXiv:1409.0473* .

T. Bui & J. Collomosse (2015). 'Scalable sketch-based image retrieval using color gradient features'. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 1–8.

T. Bui, et al. (2017). 'Compact descriptors for sketch-based image retrieval using a triplet loss convolutional neural network'. *Computer Vision and Image Understanding* **164**:27–37.

T. Bui, et al. (2018). 'Sketching out the details: Sketch-based image retrieval using convolutional neural networks with multi-stage regression'. *Computers & Graphics* **71**:77–87.

C. Burges, et al. (2006). 'Learning to rank with nonsmooth cost functions'. *Advances in neural information processing systems* **19**:193–200.

J. Canny (1986). 'A computational approach to edge detection'. *IEEE Transactions on pattern analysis and machine intelligence* (6):679–698.

Y. Cao, et al. (2011). 'Edgel index for large-scale sketch-based image search'. In *CVPR 2011*, pp. 761–768. IEEE.

Y. Cao, et al. (2010). 'Mindfinder: interactive sketch-based image search on millions of images'. In *Proceedings of the 18th ACM international conference on Multimedia*, pp. 1605–1608.

M. E. Celebi (2011). 'Improving the performance of k-means for color quantization'. *Image and Vision Computing* **29**(4):260–271.

Centre for Retail Research (2020). 'Online: UK, Europe & N. America 2020 estimatesn'. `https://www.retailresearch.org/online-retail.html`. Accessed: 2021-07-03.

D. Chen, et al. (2012). 'Bayesian face revisited: A joint formulation'. In *European conference on computer vision*, pp. 566–579. Springer.

D.-Y. Chen, et al. (2003). 'On visual similarity based 3D model retrieval'. In *Computer graphics forum*, vol. 22, pp. 223–232. Wiley Online Library.

J. Chen & Y. Fang (2018). 'Deep cross-modality adaptation via semantics preserving adversarial learning for sketch-based 3d shape retrieval'. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 605–620.

J. Chen, et al. (2019). 'Deep sketch-shape hashing with segmented 3d stochastic viewing'. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 791–800.

Y. Chen, et al. (2018). 'Darkrank: Accelerating deep metric learning via cross sample similarities transfer'. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32.

Z. Cheng, et al. (2016). ‘Pedestrian color naming via convolutional neural network’. In *Asian Conference on Computer Vision*, pp. 35–51. Springer.

V. Chernov, et al. (2015). ‘Integer-based accurate conversion between RGB and HSV color spaces’. *Computers & Electrical Engineering* **46**:328–337.

G. Ciocca, et al. (2019). ‘Evaluation of automatic image color theme extraction methods’. In *International Workshop on Computational Color Imaging*, pp. 165–179. Springer.

G. Dai, et al. (2017). ‘Deep correlated metric learning for sketch-based 3d shape retrieval’. In *Thirty-First AAAI Conference on Artificial Intelligence*.

W. Dai & S. Liang (2020). ‘Cross-modal guidance network for sketch-based 3D shape retrieval’. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE.

P. Daras & A. Axenopoulos (2010). ‘A 3D shape retrieval framework supporting multimodal queries’. *International Journal of Computer Vision* **89**(2-3):229–247.

J. Deng, et al. (2009). ‘Imagenet: A large-scale hierarchical image database’. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee.

M. Eitz, et al. (2012a). ‘How do humans sketch objects?’. *ACM Transactions on graphics (TOG)* **31**(4):1–10.

M. Eitz, et al. (2010a). ‘An evaluation of descriptors for large-scale image retrieval from sketched feature lines'. *Computers & Graphics* **34**(5):482–498.

M. Eitz, et al. (2010b). 'Sketch-based image retrieval: Benchmark and bag-of-features descriptors'. *IEEE transactions on visualization and computer graphics* **17**(11):1624–1636.

M. Eitz, et al. (2012b). 'Sketch-based shape retrieval'. *ACM Transactions on graphics (TOG)* **31**(4):1–10.

A. Fuentes & J. M. Saavedra (2021). 'Sketch-QNet: A Quadruplet ConvNet for Color Sketch-based Image Retrieval'. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2134–2141.

D. Ha & D. Eck (2017). 'A neural representation of sketch drawings'. *arXiv preprint arXiv:1704.03477* .

K. He, et al. (2016). 'Deep residual learning for image recognition'. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

C. F. Herot (1976). 'Graphical input through machine recognition of sketches'. In *Proceedings of the 3rd annual conference on Computer graphics and interactive techniques*, pp. 97–102.

G. Hinton, et al. (2015). 'Distilling the knowledge in a neural network'. *arXiv preprint arXiv:1503.02531* .

R. Hu, et al. (2010). 'Gradient field descriptor for sketch based retrieval and localization'. In *2010 IEEE International conference on image processing*, pp. 1025–1028. IEEE.

R. Hu & J. Collomosse (2013). 'A performance evaluation of gradient field hog descriptor for sketch based image retrieval'. *Computer Vision and Image Understanding* **117**(7):790–806.

R. Hu, et al. (2011). 'A bag-of-regions approach to sketch-based image retrieval'. In *2011 18th IEEE International Conference on Image Processing*, pp. 3661–3664. IEEE.

S.-M. Hu, et al. (2021). 'Subdivision-Based Mesh Convolution Networks'. *arXiv preprint arXiv:2106.02285* .

Y. Jia, et al. (2014). 'Caffe: Convolutional architecture for fast feature embedding'. In *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 675–678.

A. Krizhevsky, et al. (2012). 'Imagenet classification with deep convolutional neural networks'. *Advances in neural information processing systems* **25**:1097–1105.

J. J. LaViola Jr & R. C. Zeleznik (2006). 'Mathpad2: a system for the creation and exploration of mathematical sketches'. In *ACM SIGGRAPH 2006 Courses*, pp. 33–es.

L. Le Cam & G. L. Yang (2012). *Asymptotics in statistics: some basic concepts*. Springer Science & Business Media.

Y. LeCun, et al. (1998). 'Gradient-based learning applied to document recognition'. *Proceedings of the IEEE* **86**(11):2278–2324.

B. Li, et al. (2013a). *SHREC'13 track: large scale sketch-based 3D shape retrieval*, vol. 2013.

B. Li, et al. (2014a). 'A comparison of methods for sketch-based 3D shape retrieval'. *Computer Vision and Image Understanding* **119**:57–80.

B. Li, et al. (2014b). 'SHREC'14 track: Extended large scale sketch-based 3D shape retrieval'. In *Eurographics workshop on 3D object retrieval*, vol. 2014, pp. 121–130.

Y. Li, et al. (2014c). 'Fine-grained sketch-based image retrieval by matching deformable part models' .

Y. Li, et al. (2015). 'Free-hand sketch recognition by multi-kernel feature learning'. *Computer Vision and Image Understanding* **137**:1–11.

Y. Li, et al. (2013b). 'Sketch Recognition by Ensemble Matching of Structured Features.'. In *BMVC*, vol. 1, p. 2.

S. Lin & P. Hanrahan (2013). 'Modeling how people extract color themes from images'. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 3101–3110.

M. Ortega, et al. (1998). 'Supporting ranked boolean similarity queries in MARS'. *IEEE Transactions on Knowledge and Data Engineering* **10**(6):905–925.

T. Y. Ouyang & R. Davis (2011). 'Chemink: a natural real-time recognition system for chemical drawings'. In *Proceedings of the 16th international conference on Intelligent user interfaces*, pp. 267–276.

A. Qi, et al. (2018). 'Semantic Embedding for Sketch-Based 3D Shape Retrieval.'. In *BMVC*, vol. 3, pp. 11–12.

Y. Qi, et al. (2016). 'Sketch-based image retrieval via siamese convolutional neural network'. In *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 2460–2464. IEEE.

F. Radenovic, et al. (2018). 'Deep shape matching'. In *Proceedings of the european conference on computer vision (eccv)*, pp. 751–767.

N. R. R. Reddy, et al. (2014). 'Color sketch based image retrieval'. *Int. J. Adv. Res. Electrical Electron. Instrum. Eng* **3**:12179–12185.

P. Sangkloy, et al. (2016). 'The sketchy database: learning to retrieve badly drawn bunnies'. *ACM Transactions on Graphics (TOG)* **35**(4):1–12.

R. K. Sarvadevabhatla & R. V. Babu (2015). 'Freehand sketch recognition using deep features'. *arXiv preprint arXiv:1502.00254* .

R. K. Sarvadevabhatla & J. Kundu (2016). 'Enabling my robot to play pictionary: Recurrent neural networks for sketch recognition'. In *Proceedings of the 24th ACM international conference on Multimedia*, pp. 247–251.

R. G. Schneider & T. Tuytelaars (2014). 'Sketch classification and classification-driven analysis using fisher vectors'. *ACM Transactions on Graphics (TOG)* **33**(6):1–9.

B. Schölkopf, et al. (2002). *Learning with kernels: support vector machines, regularization, optimization, and beyond.* MIT press.

O. Seddati, et al. (2017). 'Quadruplet networks for sketch-based image retrieval'. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, pp. 184–191.

T. M. Sezgin, et al. (2007). 'Sketch based interfaces: Early processing for sketch understanding'. In *ACM SIGGRAPH 2007 courses*, pp. 37–es.

T. Shao, et al. (2011). 'Discriminative sketch-based 3d model retrieval via robust shape matching'. In *Computer Graphics Forum*, vol. 30, pp. 2011–2020. Wiley Online Library.

J.-L. Shih, et al. (2007). ‘A new 3D model retrieval approach based on the elevation descriptor’. *Pattern Recognition* **40**(1):283–295.

P. Shilane, et al. (2004). ‘The princeton shape benchmark’. In *Proceedings Shape Modeling Applications, 2004.*, pp. 167–178. IEEE.

K. Simonyan & A. Zisserman (2014). ‘Very deep convolutional networks for large-scale image recognition’. *arXiv preprint arXiv:1409.1556* .

J. Sivic & A. Zisserman (2003). ‘Video Google: A text retrieval approach to object matching in videos’. In *Computer Vision, IEEE International Conference on*, vol. 3, pp. 1470–1477. IEEE Computer Society.

A. R. Smith (1978). ‘Color gamut transform pairs’. *ACM Siggraph Computer Graphics* **12**(3):12–19.

J. Song, et al. (2017). ‘Deep spatial-semantic attention for fine-grained sketch-based image retrieval’. In *Proceedings of the IEEE international conference on computer vision*, pp. 5551–5560.

H. Su, et al. (2015). ‘Multi-view convolutional neural networks for 3d shape recognition’. In *Proceedings of the IEEE international conference on computer vision*, pp. 945–953.

M. Summerfield (2007). *Rapid GUI Programming with Python and Qt: The Definitive Guide to PyQt Programming (paperback).* Pearson Education.

F. P. Tasse & N. Dodgson (2016). ‘Shape2vec: semantic-based descriptors for 3d shapes, sketches and images’. *ACM Transactions on Graphics (TOG)* **35**(6):1–12.

F. Wang, et al. (2015). ‘Sketch-based 3d shape retrieval using convolutional neural networks'. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1875–1883.

Y. Xia, et al. (2019). 'Fine-grained color sketch-based image retrieval'. In *Computer Graphics International Conference*, pp. 424–430. Springer.

J. Xie, et al. (2017). 'Learning barycentric representations of 3d shapes for sketch-based 3d shape retrieval'. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5068–5076.

A. Yu & K. Grauman (2014). 'Fine-grained visual comparisons with local learning'. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 192–199.

L. Yu, et al. (2019). 'Learning metrics from teachers: Compact networks for image embedding'. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2907–2916.

Q. Yu, et al. (2016). 'Sketch me that shoe'. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 799–807.

Q. Yu, et al. (2017). 'Sketch-a-net: A deep neural network that beats humans'. *International journal of computer vision* **122**(3):411–425.

Q. Yu, et al. (2015). 'Sketch-a-net that beats humans'. *arXiv preprint arXiv:1501.07873* .

H. Zhang, et al. (2016). 'Sketchnet: Sketch classification with web images'. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1105–1113.

F. Zhu, et al. (2016). 'Learning cross-domain neural networks for sketch-based 3d shape retrieval'. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 30.

C. L. Zitnick & P. Dollár (2014). ‘Edge boxes: Locating object proposals from edges’. In *European conference on computer vision*, pp. 391–405. Springer.