SEMANTIC TREE-BASED 3D MODEL RETRIEVAL USING 2D SKETCH QUERIES

by

Bo Li, Ph.D.

A thesis submitted to the Graduate Council of
Texas State University in partial fulfillment
of the requirements for the degree of
Master of Science
with a Major in Applied Mathematics
August 2015

Committee Members:

Jian Shen, Chair

Yijuan Lu

Xingde Jia

FAIR USE AND AUTHOR'S PERMISSION STATEMENT

Fair Use

Duplication Permission

ACKNOWLEDGEMENTS

Many thanks go to Dr. Jian Shen, without whose guidance and inspiration I would not complete the research. I would like to thank Dr. Yijuan Lu and Dr. Xingde Jia for their constant support and encouragement during the thesis process. Finally, I wish to thank my family for their patience and understanding during my many late nights at the office in the past year.

TABLE OF CONTENTS

LIST OF TABLES

Table **Page**

LIST OF FIGURES

ABSTRACT

Effectively and efficiently retrieving relevant 3D models (digital representation of objects in computer) for a 2D sketch query is important for various related applications. Due to the big semantic gap existing between rough sketch representation and accurate 3D model coordinates, sketch-based 3D model retrieval (SBR) is one of the most challenging research topics in the field of 3D model retrieval. To bridge the semantic gap, a semantic tree-based SBR algorithm has been proposed in the thesis. Given a 2D sketch query and a dataset of 3D models, we first build a 3D shape network (3D ShapeNet) based on the semantic tree ontology in WordNet, which is a lexical database of concepts/synsets, by classifying the 3D models into certain class nodes of the tree, according to their semantic classification/label information (i.e. semantic concepts or names). Then, we identify the semantic attributes (i.e. semantic components) that the 2D sketch query contains by 2D sketch segmentation and labeling. Finally, by measuring the semantic relatedness between the concept set of the 2D sketch components and each class node, we compute the similarity between the 2D sketch and each class node to shortlist closest class nodes as well as the relevant 3D models for the 2D sketch query. Experimental results on ten classes of sketches and models have demonstrated promising performance in bridging the semantic gap.

CHAPTER 1

INTRODUCTION

1.1 Background

3D model of a real object is a computer file to save its geometric information, i.e. 3D coordinates of a list of vertices and face information for a 3D triangular mesh, for which we show two examples in Fig. 1.1. 3D models are useful and important for various applications. Sketch-based 3D model retrieval is to retrieve 3D models based on a query sketch. It is important for many popular applications such as sketch-based rapid prototyping, recognition, mobile 3D search, 3D printing, and 3D animation production. However, because of the big semantic gap due to the difference between rough sketch representation and accurate 3D model coordinates, it is one of the most challenging research topics in the field of 3D model retrieval.

1.2 Motivations

Existing sketch-based 3D model retrieval systems are mainly based on a direct content-based comparison between a 2D sketch query and all the target 3D models.

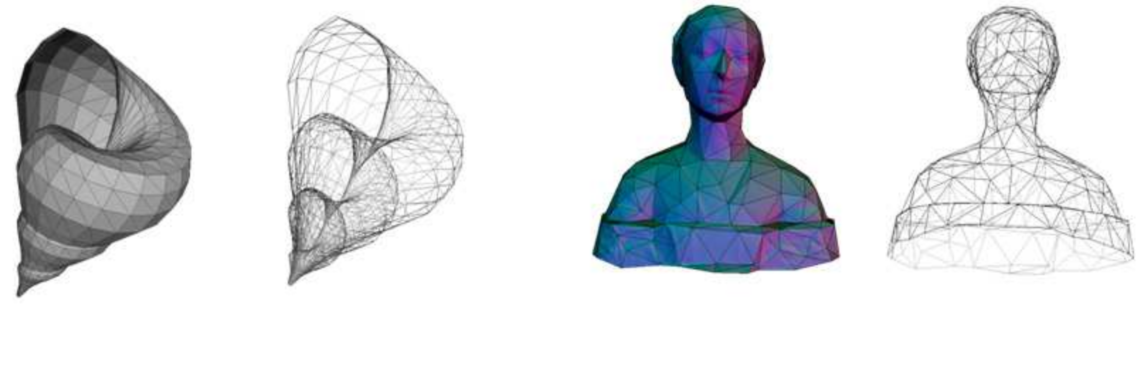


Figure 1.1: Example 3D triangular meshes.

However, there is a semantic gap between the iconic representation of 2D sketches and the accurate 3D coordinate representation of 3D models. This makes the task of retrieval using sketch queries much more challenging than those using 3D model queries, which has been demonstrated by their inferior performance (Li et al. [2014a]) on several latest benchmarks including SHREC'13 Sketch Track Benchmark (SHREC13STB) (Li et al. [2013, 2014a]) and SHREC'14 Sketch Track Benchmark (SHREC14STB) (Li et al. [2015, 2014b]).

Motivated by the above obstacles, an interesting question has been raised: "why not employing semantic information?" Semantic approach may facilitate us to achieve better retrieval performance than content-based search does. However, how to extract semantic information, and how to organize 2D sketches and 3D models and semantically compare them become new research problems. In the thesis, we make some initial study on these two problems and develop a novel semantic tree-based 3D model retrieval algorithm.

1.3 Overview of Our Research

Given a 2D sketch query and a dataset of 3D models, we first build a 3D shape network (3D ShapeNet) based on the semantic ontology in WordNet (Miller [1995]), which is a lexical database of concepts/synsets, represented by a set of synonyms. Each word has one or more senses; each sense has its synset; and a set of words related through following three relationships: hypernyms/hyponyms (IS_A relation), holonyms (MEMBER_OF relation) and meronyms (PART_OF relation).

3D ShapeNet is built by classifying the 3D models into certain class nodes of the semantic tree, according to their semantic classification/label information (i.e. semantic concepts or names). Then, we identify the semantic attributes (i.e. semantic components) that the 2D sketch query contains by 2D sketch segmentation and labeling. Finally, by measuring the semantic relatedness between the concept set of the 2D sketch components and each class node, we compute the similarity between the 2D sketch and each class node to shortlist closest class nodes as well as the relevant 3D models for the 2D sketch query. During this process, we also perform word sense disambiguation for the components' names.

Here, we give two definitions of semantic "attribute": (1) attribute is an inherent characteristic (Merriam-Webster [2015]); (2) attribute is a quality or characteristic inherent in or ascribed to someone or something (TheFreeDictionary [2015]).

Take "Human being" as an example. Human being: any living or extinct member of the family Hominidae characterized by superior intelligence, articulate speech, and erect carriage (TheFreeDictionary [2015]). Its definition is composed of two sets of words: adjective words set (e.g. living, extinct, superior, articulate, erect) and noun words set (i.e., family, Hominidae, intelligence, speech, carriage). Though, maybe not all of them are "inherent" attributes, like living, they represent the most important elements that differentiate it from all of other biological objects. However, some attributes, such as most adjective words including living, extinct, superior, articulate and most noun words like family, Hominidae, intelligence, speech, carriage, may not be possibly reflected in its shape, while we are mainly only exposed in the context/scenario of content-based multimedia retrieval. Therefore, we need to find an approach to bridge these two types of representations: semantics and shape content.

Before building the semantic 3D shape network, we have several questions to consider for our design. (1) Which attributes do we need to choose in order to unambiguously classify an object into a distinct category? (2) Can the features extracted based on the visual contents represent those attributes? (3) Which topology will the 3D shape network have and how to perform retrieval based on the network? (4) How to incorporate the noun attributes into the network and possibly how to represent the relationships among adjectives and nouns as well?

1.4 Thesis Organization

The thesis is organized as follows.

- Chapter 2 reviews the related work in sketch-based 3D model retrieval, attributes-based and WordNet-based semantic multimedia retrieval, and local image features.

- Chapter 3 presents the semantic tree-based retrieval algorithm, including its inputs, pre-processing, and online retrieval stages.

- Chapter 4 demonstrates the evaluation and comparative experimental results with respect to retrieval accuracy.

- Chapter 5 contains the conclusions and future work. We first draw a conclusion on the thesis work and then propose three new research directions for the research topic of semantic sketch-based 3D model retrieval as the future work.

CHAPTER 2

RELATED WORK

In this chapter, we will review different aspects of prior research that are related to our research topic: semantic tree-based 3D model retrieval by a 2D sketch query. They include sketch-based 3D model retrieval, semantic attributes-based multimedia retrieval, WordNet-based semantic multimedia retrieval, as well as local image features.

2.1 Sketch-Based 3D Model Retrieval

Recently, substantial research work has been performed in sketch-based 3D model retrieval. For instance, Histogram of Gradient (HOG) feature has been used in Yoon et al. [2010], Eitz et al. [2010], and Eitz et al. [2011a], followed by an Overlapped Pyramid of HOG (OPHOG) feature by Tatsuma and Aono (Li et al. [2014a]). Later, Eitz et al. [2012b] proposed a Gabor local line-based feature (GALIF), Furuya and Ohbuchi [2013] employed Cross-Domain Manifold Ranking (CDMR) technique, while Li et al. (Li et al. [2015]) developed a parallel shape context-based matching algorithm for the retrieval.

Eitz et al. [2012a] did the first large scale exploration of 2D human-drawn sketches. They gathered 20,000 sketches of 250 different objects created by hundreds of people all over the world. Three Shape Retrieval Contest (SHREC) tracks on the topic of sketch-based 3D model retrieval have been held in conjunction with the 2012, 2013 and 2014 Eurographics Workshops on 3D Object Retrieval (3DOR). In each track, different methods have been evaluated on the corresponding benchmarks, for example the SHREC'13 Sketch Track Benchmark (SHREC13STB) (Li et al. [2013]) which contains 7200 2D sketches and 1258 3D models of 90 classes, and the SHREC'14 Sketch Track Benchmark (SHREC14STB) (Li et al. [2014b]) which contains 13680 2D sketches and 8987 3D models of 171classes.

2.2 Semantic Attributes-Based Multimedia Retrieval

2.2.1 *Pioneer Work*

As early as 2007, attributes have been received attention in the community of image retrieval or recognition (Ferrari and Zisserman [2007]). Farhadi et al. [2009] proposed to describe objects rather than naming them followed by identification by learning object attributes. They proposed a feature selection method to learn attributes based on a set of low-level features and recognized objects based on an attribute classifier.

2.2.2 *For 2D Retrieval*

Douze et al. [2011] combined the Classemes attributes (Torresani et al. [2010]) and Fisher vectors for image retrieval. Siddiquie et al. [2011] developed a multi-attribute query-based image retrieval algorithm by incorporating the correlations between the attributes. Yu et al. [2012] proposed to utilize a large number of weak attributes including classifier (e.g. Classmes (Torresani et al. [2010])) scores and other easily reachable representations, like those in Ferrari and Zisserman [2007].

Cai et al. [2013] proposed to divide a training dataset into several sets, each of which represents different attribute and then based on them learn multiple sub-vocabularies, which are so called attribute-aware dictionaries. Chen et al. [2013] proposed to improve face image retrieval by constructing attribute-enhanced sparse codewords; while Wang et al. [2013] performed clothes retrieval by leveraging low-level color features and high-level attributes, which are more robust to variations and deformations of clothes.

Lin [2012] proposed to utilize 3D models to help attribute-based object (e.g. vehicle) retrieval in terms of obtaining semantic parts information, like Lin et al. [2013]. Zhang et al. [2013a] proposed an attribute-augmented semantic hierarchy for image-based retrieval. The hierarchy organizes semantic concepts/classes, together with their attributes, such that distances among images can be measured via a proposed hierarchical semantic similarity function.

2.2.3 *For 3D Retrieval*

Gong et al. [2013] proposed to use attribute signature (AS) and reference set signature (RSS) to perform semantic 3D model retrieval. They selected 11 attributes including symmetry, flexibility, rectilinearity, circularity, dominant-plane, long, thin, swim, fly, stand with leg(s), and natural. Rather than computing these features, they adopted an indirect way by learning those attributes by extracting three 3D features to represent each model and utilizing LIBSVM to train and learn a set of two-class SVM classifiers. But the experimental results show that RSS often outperforms AS, which is mostly due to the insufficiency of the set of 11 attributes to capture the semantic meanings of diverse classes. They also found that their high-level semantic approaches (AS and RSS) can complement low-level features and they non-trivially improve the retrieval performance when used in a combination. They also claimed that one advantage of their semantic features is the compactness (thus efficient for large-scale retrieval scenarios).

Kim et al. [2004] extracted attributed relational graph (ARG) to represent a 3D model by encoding the topological information of the structure of the decomposed 3D model.

2.2.4 ***Attributes***

2.2.4.1 <u>Semantic Attributes</u>

Gong et al.'s attributes (Gong et al. [2013]): symmetry, flexibility, rectilinearity, circularity, dominant-plane, long, thin, swim, fly, stand with leg(s), and natural;

Russakovsky and Li [2010]: (1) Tool: obtain the ground truth data we use workers on Amazon Mechanical Turk (AMT); (2) Attributes: For color attributes (black, blue, brown, gray, green, orange, pink, red, violet, white and yellow), we ask whether a significant part of the object (at least 25%) is that color. For all other attributes (furry, long, metallic, rectangular, rough, round, shiny, smooth, spotted, square, striped, vegetation, wet, wooden), we ask if they would describe the object as a whole using that attribute.

Farhadi et al. [2009]: select 64 attributes (2D Boxy, 3D Boxy, Round, Vert

Cyl, Horiz Cyl, Occluded, Tail, Beak, Head, Ear, Snout, Nose, Mouth, Hair, Face, Eye, Torso, Hand, Arm, Leg, Foot/Shoe, Wing, Propeller, Jet engine, Window, Row Wind, Wheel, Door, Headlight, Taillight, Side mirror, Exhaust, Pedal, Handlebars, Engine, Sail, Mast, Text, Label, Furn. Leg, Furn. Back, Furn. Seat, Furn. Arm, Horn, Rein, Saddle, Leaf, Flower, Stem/Trunk, Pot, Screen, Skin, Metal, Plastic, Wood, Cloth, Furry, Glass, Feather, Wool, Clear, Shiny, Vegetation, Leather).

Applications: semantic annotation medical analysis (Catalano et al. [2012]), semantic image organization (Li et al. [2010]), re-ranking (Cai et al. [2012]) and facial attributes and canvas layout (Lei et al. [2012]).

2.2.4.2 Discriminative Attributes

Transfer learning-based discriminative attributes. Besides semantic attributes, Farhadi et al. [2009] also considered the discriminative attributes based on the transfer learning approach proposed in Farhadi et al. [2007]. The discriminative attributes are learned based on the idea of projecting image features to a discriminative space based on common comparisons.

Between-class attribute transfer. Lampert et al. [2009] proposed to recognize new classes by transferring the attributes of known classes to unknown classes.

Comparative Attributes. Recently, to avoid semantic drift, Shrivastava et al. [2012] proposed to impose several constraints on the semantic attributes-based learning process based on mutual exclusion, binary attributes and comparative attributes. They defined comparative attributes as a triplet to indicate the constraint between two images.

2.2.4.3 Relative Attributes

Parikh and Grauman [2011] proposed relative attributes to indicate the strength of the presence of an attribute in an image with respect to other images. After that, several following works dealing with relative attributes have been proposed. For example, Sandeep et al. [2014] defined the idea of relative parts, which are distinctive parts in an image, to learn relative attributes; Liang and Grauman [2014] explored active learning strategies to train a relative attribute ranking function for a reliable relative attribute predictions.

2.2.4.4 Attributes-Related Techniques

Usually, some typical techniques related to semantic attributes will be used in semantic attribute-based multimedia retrieval. Among them, are **Feature Selection** (Farhadi et al. [2009]), **Transfer Learning** (Farhadi et al. [2007]), and **Fisher Vector** (Douze et al. [2011], Perronnin et al. [2010]).

2.2.5 *Attributed 2D & 3D Datasets*

a-Pascal and a-Yahoo datasets (Farhadi et al. [2009]). (1) a-Pascal dataset, which comprises 20 classes: aeroplane, bicycle, bird, boat, bottle, bus, car, cat, chair, cow, diningtable, dog, horse, motorbike, person, pottedplant, sheep, sofa, train, tvmonitor. It has 150∼1000 images per class, while 5000 images for people. (2) a-Yahoo, which is composed of 12 classes: donkey, monkey, goat, wolf, jetski, zebra, centaur, mug, statue, building, bag, carriage.

Cross-Object REcognition (CORE) dataset (Farhadi et al. [2010]). It is made up of around 3000 annotated objects contained in 2800 images based on the labeling work in Amazon's Mechanical Turk. Totally, there are 28 annotated classes. The dataset is specially designed for cross-category recognition and it has been used in the following applications: familiar/unfamiliar objects description; parts localization for pose, viewpoint, and object parsing.

ImageNet Attributed dataset (Russakovsky and Li [2010]). It has 20 attributes for 384 popular synsets. The 20 attributes are as follows: (1) Color attributes: black, brown, gray, green, orange, red, white, yellow; (2) Shape attributes: long, round, rectangular; (3) Pattern attributes: spotted, striped; (4) Texture: furry, smooth, rough, shiny, metallic, wooden, wet.

Attributed animal datasets. (1) In Osherson et al. [1991], Osherso et al. defined 48 animal classes and 85 attributes, but there were no real images; (2) In Kemp et al. [2006], Kemp et al. provided an improved version by adding two more classes; (3) In Lampert et al. [2009], Lampert et al. build an image dataset version by finding 30475 images for the 50 classes via Google, Microsoft, Yahoo and Flickr for computer vision research. At least 92 images have been assigned for each class.

Attributed Sketch Datasets. Huang et al. [2014] proposed a data-driven 2D sketch segmentation and labeling algorithm, which provided labeled component information for 300 sketches of 10 classes (each with 30 sketches): chair, table, airplane, bicycle, fourleg, lamp, vase, human, candelabrum, and rifle. For example, it segments a bicycle into two wheel strokes, and a stroke for a frontframe, backframe,

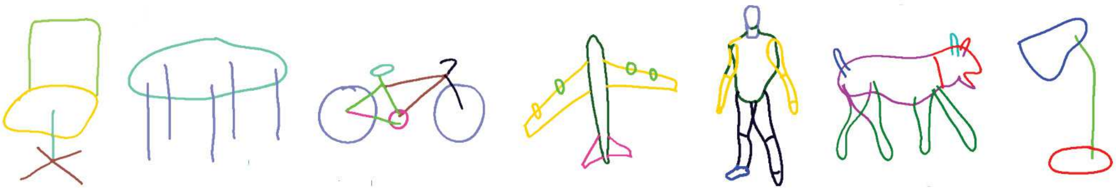


Figure 2.1: 2D sketch segmentation and labeling examples in Huang et al. [2014].

fork, chain, handle, and saddle, respectively. Fig. 2.1 shows several segmentation and labeling results.

Attributed 3D Model Datasets. Similarly, partitioning a 3D model into several semantic parts is important for 3D model recognition. Typical approaches include graph cut (Karger and Stein [1996]) and random cut (Golovinskiy and Funkhouser [2008]), fuzzy or spectral clustering (Katz and Tal [2003], Liu and Zhang [2004]), primitive fitting (Attene et al. [2006]), shape diameter (Shapira et al. [2008]) or machine learning method (Kalogerakis et al. [2010]).

Chen et al. [2009] built a 3D model segmentation benchmark which contains 11 human-generated segmentations for each of a set of 380 models of 19 categories. Recently, Kalogerakis et al. [2010] proposed a data-driven approach to learn the segmentation of a 3D model and they provided automatic segmentation results for the same set of 380 models in Chen et al. [2009].

2.3 WordNet-Based Semantic Multimedia Retrieval

As a lexical dictionary of semantic concepts, WordNet has been vastly applied in semantic multimedia retrieval of either text or image objects.

2.3.1 *For Text Retrieval*

WordNet has been used in text retrieval for disambiguation resolution (Voorhees [1993], Liu et al. [2005]), image caption retrieval (Smeaton and Quigley [1996]), as well as performance improvement (Gonzalo et al. [1998], Liu et al. [2004], Dragoni et al. [2012]).

2.3.2 *For Image Retrieval*

Aslandogan et al. [1997] utilized WordNet for query and database expansion in image retrieval. Database expansion refers to expanding the meta-data in the database. They considered synonyms of nouns and verbs, different number of (first or all) senses of a word, and other three relationships (IS_A, MEMBER_OF, and PART_OF) mentioned before. They found that for query expansion the optimal setting is using synonyms of all senses, or considering the synonyms and the IS_A and MEMBER_OF relations of the first sense of a word. For database expansion, a technique of category verification was developed to reduce the negative effect of spurious matches, which can further improve the retrieval performance.

Marszalek and Schmid (Marszalek and Schmid [2007]) proposed to utilze WordNet to build a semantic and hierarchical graph for the objects involved. Based on labeled training data, they learned a binary classifier for each node in the graph.

Wang et al. [2008] proposed to build an ontology based on WordNet for a 3D model benchmark, infer 3D semantic properties by rule engine based on Semantic Web Rule Language (SWRL), and perform semantic retrieval using the ontology.

The algorithm supports both text and 3D queries, and relevance feedback is used to optimize the retrieval results. However, only the results of 20-model test is reported and there is a lack of database-level performance comparison.

A survey on three typical semantics processing (relevance feedback, machine learning, and ontology) has been performed in Gao et al. [2009], while Tousch et al. [2012] presented a survey on cxsemantics-based image annotation.

2.3.3 *WordNet-Based Semantic Distance Metrics*

Several semantic similarity and relatedness metrics have been proposed. For example, Pedersen et al. [2004] implemented three similarity measures that are based on path lengths between concepts: *lch* (Leacock and Chodorow [1998]), *wup* (Wu and Palmer [1994]), and *path*; and three semantic relatedness measures: *hso* (Hirst and St-Onge [1998]), *lesk* (Banerjee and Pedersen [2003]), and *vector* (Patwardhan [2003]).

Other semantic relatedness and similarities have been proposed in Patwardhan et al. [2003] and Pedersen et al. [2007], as well.

2.3.4 *WordNet-Based Sense Disambiguation*

When we look up a word from WordNet, it usually lists several senses of the word to indicate the different meanings that the word may have in different text contexts. Therefore, deciding which sense should be adopted for a situation is important for its correct interpretation. Different approaches have been proposed for word sense disambiguation. For example, for such purpose, Patwardhan et al. [2003] proposed using an adapted Lesk algorithm (Banerjee and Pedersen [2002]). The main idea of the original Lesk algorithm (Lesk [1986]) is based on the following two hypotheses: a word in a sentence can be disambiguated by its neighboring words by assigning the most closely related sense to it; and overlapping words in the glosses of neighboring words are helpful to identify their related senses. Therefore, they conducted sense disambiguation by comparing the number of overlapping words between the gloss of a word and the glosses of its neighboring words.

Pedersen et al. also developed several software for sense disambiguation, such as WordNet::SenseRelate::TargetWord (Pedersen et al. [2006]), WordNet::SenseRelate:: WordToSet (Pedersen and Michelizzi [2006]) and and WordNet::SenseRelate::AllWords (Pedersen and Kolhatkar [2009]), which are specially designed for the sense disambiguation problems of one word per context, one word per a set of related words, and each word in a text, respectively.

2.4 Local Image Features

In this section, we review several latest promising local image features used to describe semantic attributes of sketches.

Histograms of Oriented Gradients (HOG) (Dalal and Triggs [2005]) feature first divides an image into grids and then for each grid it computes a local and combinational distribution of the gradients, which include both orientation and magnitude. Finally, it concatenates all the local descriptors sequentially to form the HOG feature of the image. While for sketch, the only meaningful part is orientation. Thus, Eitz et al. proposed a simplified HOG (sHOG) (Eitz et al. [2012a, 2011b]) which has only concern on the orientation component. Variations of original HOG also include pyramid HOG (PHOG) (Bosch et al. [2007]) and multiscale HOG (Newell and Griffin [2011]). In addition, rather than combining all the local descriptors, Bag-of-Words model is often used in sketch recognition or sketch-based 3D model retrieval to accelerate the retrieval speed.

Oriented FAST and Rotated BRIEF (Calonder et al. [2012]) (ORB) (Rublee et al. [2011]) is a comparable but more efficient substitute to SIFT to meet the requirements of related applications where there are less powerful computational resources (e.g. GPU) and higher standards for computational efficiency. Two of such examples are realtime embedded system and mobile search. ORB is two orders of magnitude faster than SIFT while it also outperforms another established alternative feature SURF (Bay et al. [2008]).

Edge-SIFT (Zhang et al. [2013b]) is a binary descriptor specially designed for mobile image search. It is an extension of SIFT applied on the edge features of an image and adopts a descriptor compression method to make it more compact. It outperforms ORB in terms of accuracy while their efficiencies are comparable.

Binary Robust Invariant Scalable Keypoints (BRISK) (Leutenegger et al. [2011]) is another new and efficient method designed to perform keypoint detection, description and matching thanks to the utilization of a scale-space FAST (Rosten and Drummond [2006])-based keypoint detection method and the binary nature of the BRISK descriptor. It achieves comparable accuracy as SURF at an order of magnitude less time.

Fast Retina Keypoint (FREAK) (Alahi et al. [2012]) extracts a cascade of binary descriptors over a retina sample pattern, motivated by the human visual system (especially retina) and the efficient binary descriptors extracted in BRIEF, ORB and BRISK. FREAK is more efficient and robust than SIFT, SURF and BRISK.

Scale-Invariant Feature Detector with Error Resilience (SIFER) (Mainali et al. [2013]) is a new feature extracted based on Cosine Modulated Gaussian filter. It reliably and efficiently detects corners and blobs, except that it has a reduced planar rotational invariance.

Most of the above features, such as HOG, ORB, BRISK and FREAK, have been implemented and integrated into the latest OpenCV (OpenCV [2013]) library.

Selected feature sets instances: (1) Lampert et al. [2009]: RGB color histograms, SIFT, rgSIFT, PHOG, SURF and local self-similarity histograms; (2) Farhadi et al. [2009]: 1) material feature: color and texture (texture descriptors), 2) part feature: visual words (PHOG), 3) shape feature: edges (edge histogram). Totally, there are seven histograms, and 9751 dimensions.

CHAPTER 3

SEMANTIC TREE-BASED RETRIEVAL ALGORITHM

Given a 2D query sketch and a set of 3D models arranged on a semantic tree, we need to retrieve relevant 3D models similar to the 2D sketch. As illustrated in Fig. 3.1, the basic idea of our semantic tree-based SBR algorithm is to measure the 2D-3D similarity by fusing the semantic relatedness values between the labeled names of the semantic components of the query sketch and the categorical names of the 3D models, based on the semantic hierarchy in WordNet.

3.1 Inputs

(1) **2D sketch dataset**: We have a number of hand-drawn 2D sketches, each one is black and white image. They represent 3D objects from different classes, such as airplane, bicycle, and chair. During retrieval, users draw or provide a 2D sketch as the input of the retrieval algorithm in order to find relevant 3D models for that query. The retrieved models will be listed according to the similarity values.

(2) **3D ShapeNet**: It is a hierarchy of classes (*nouns*) based on the semantic hierarchy in WordNet. Each class has several attributes (i.e. is-a, has-part, is-made-of, or is-an-attribute-of relation) according to its gloss defined in WordNet. Each leaf node of the 3D ShapeNet also has a number of 3D models belonging to the leaf node class. In detail, the 3D ShapeNet forms a network of classes, attributes and models.

a) **Classes**: The 3D models in the target 3D model dataset are classified into a number of classes, which correspond to the leaf nodes, denoted as $\mathbf{N}$=$\{N_i\}$, in the

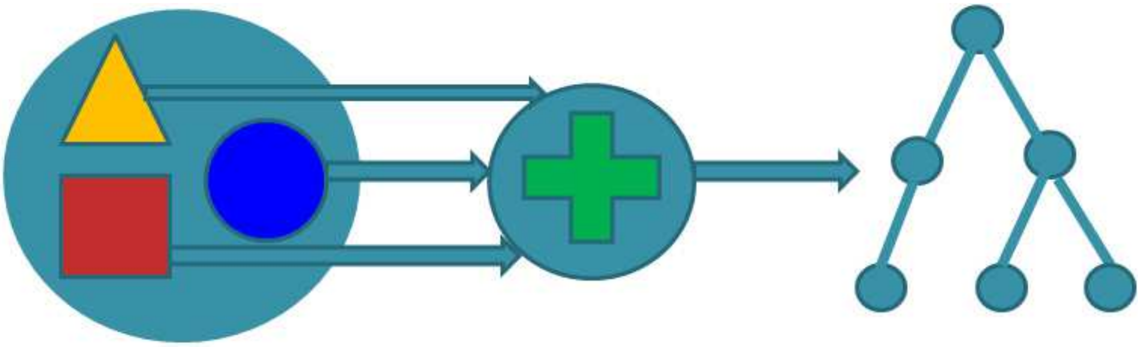


Figure 3.1: Illustration of the basic idea of our semantic tree-based SBR algorithm. The triangular, square and circle denote the components; the cross means fusing relatedness; the tree denotes the semantic hierarchy of the WordNet.

WordNet-based hierarchical tree;

b) **Attributes**: Each leaf node N_i possibly has several attributes according to its definition (gloss), denoted as $\{A_{ij}\}$. Each leaf node (class) N_i has at least one parent node (class) and several ancestor nodes (classes). We name this set of parent nodes (classes) as $\{N_i^{pk}\}$. Similarly, a parent node (class) N_i^{pk} may also have some attributes, denoted as $\{A_{ij}^{pk}\}$;

c) **3D models**: We assume that we were presented a classified 3D model dataset. That is, all the target 3D models are pre-classified into a set of leaf nodes (classes) according to the benchmark information of one or more 3D shape benchmarks. New models can be dynamically and automatically classified and inserted into the 3D ShapeNet.

3.2 Pre-processing

3D models in the target dataset have already been classified, thus it is easy to arrange them onto the semantic tree according to their class names. For example, Fig. 3.2 shows a portion of the semantic tree for the semantic organization of the 3D models.

3.3 Online Retrieval

During online retrieval, we first perform 2D model segmentation and labeling: segmenting a 2D sketch into components and then labeling those components. After this step, each component has a semantic label. For example, a human sketch will be segmented and labeled into following parts: foot, hand, leg, arm, body, and head. Some part labels may appear more than one time, such as the foot, leg and arm labels in the above example.

After that, we compute the semantic distances between the query sketch and the 3D models based on the semantic relatedness between the component set and the model class name. The detailed steps are as follow:

(1) **2D sketch segmentation.** This step partitions a query 2D sketch q into a set of consistent semantic components $\{C_i\}$. Each component is composed of one or more strokes, thus we can compute the total number of pixels in the component, that is, the length of the component.

(2) **Labeling 2D sketch component.** For each component C_i, we assign a semantic name, which is regarded as an attribute A_i of the query sketch q according to the PART_OF relationship: the class that the query q belongs to has a component

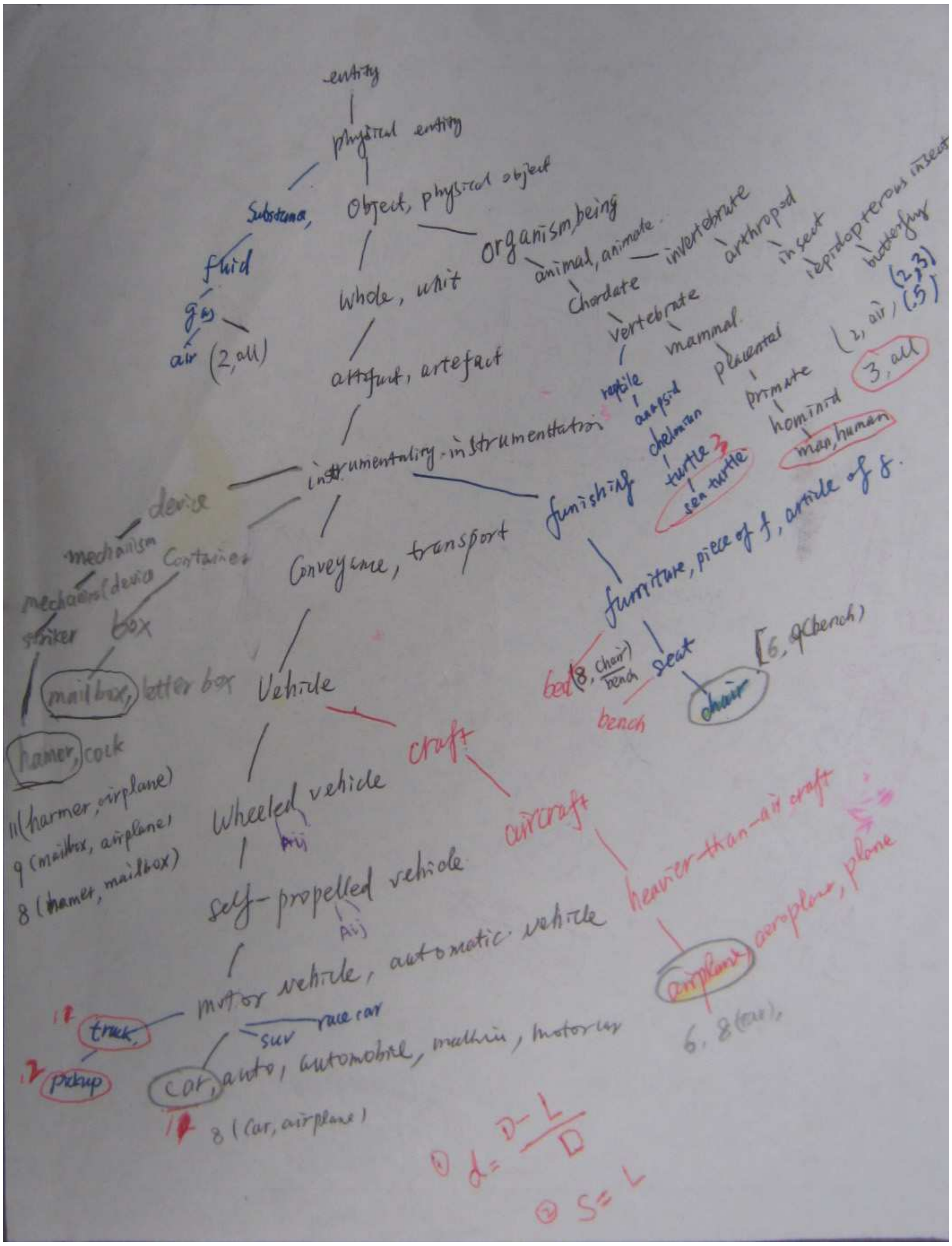


Figure 3.2: Semantic tree example.

C_i which falls into the class of A_i.

(3) **Word sense disambiguation of component labels.** To compute the semantic relatedness value between a labeled semantic component of a 2D sketch and the name of a 3D model category, we need to decide which sense (the meaning of the word) that the labeled name should take. Motivated by the idea and the two hypotheses of the original Lesk algorithm presented in Chapter 2 (Section 2.3.4), for each component label of the query sketch, we regard the labels of other components as its context for its sense disambiguation by comparing the number of overlapping words between the gloss of the component's label and the glosses of other components' labels. In practice, we adopt the WordNet::SenseRelate::WordToSet software available on Pedersen and Michelizzi [2008] to choose the best sense.

As mentioned in Chapter 2 (Section 2.3.4), WordNet::SenseRelate::WordToSet finds the WordNet sense of a single word that is most related to a given set of words. Eleven different similarity and relatedness metrics can be chosen. Considering the fact that usually any two components of a sketch are often semantically different, thus using similarity metrics are inappropriate, we choose the Lesk relatedness metric to measure related senses.

(4) **Semantic distance computation.** This step is to compute the total relatedness distance $d_r(q,\ N_i)$ between the set of component attributes $\{A_i\}$ of the query sketch q and a semantic class N_i. Similarly, we only consider relatedness distance because instead of similarity, the components are only "related" to a class. That is, among the three semantic relatedness relationships: hypernyms/hyponyms (IS_A relation), holonyms (MEMBER_OF relation) and meronyms (PART_OF

relation), PART_OF is the only appropriate one since all the similarity metrics should be excluded from our considerations. In experiments, we mainly consider the following three semantic relatedness measures: *hso* (Hirst and St-Onge [1998]), *lesk* (Banerjee and Pedersen [2003]), and *vector* (Patwardhan [2003]). They have been implemented in the software WordNet::Similarity (Pedersen et al. [2005]). According to our experiments, *hso* performs the best.

(5) **Ranking and output.** Sort $d_r(q, N_i)$ in an ascending order and then list all the models accordingly.

CHAPTER 4

EXPERIMENTS AND DISCUSSIONS

4.1 Datasets

4.1.1 *2D Sketch Dataset*

As queries for our retrieval algorithm, we selected sketches from the 300 sketches dataset collected in Huang et al. [2014]. They are classified into 10 classes, while each class has 30 sketches. When computing retrieval performance, we choose the following 10 query sketches, as shown in Fig 4.1.

4.1.2 *3D Model Dataset*

Similar as the 3D model collection method in Huang et al. [2014], we collected 407 models in total for the same 10 classes as 2D sketches: airplane (70 models), bicycle (38 models), candelabrum (28 models), chair (70 models), quadruped (20 models), human (20 models), lamp (20 models), rifle (19 models), table (61 models), and vase (61 models). Fig. 4.2 shows one example for each class.

4.2 Evaluation Metrics

To conduct a comprehensive evaluation of our semantic 3D model retrieval algorithm using a sketch query on the above datasets, we adopt seven commonly used performance metrics (Shilane et al. [2004]) in information retrieval area: Precision-Recall (PR) diagram, Nearest Neighbor (NN), First Tier (FT), Second Tier (ST), E-Measures (E), Discounted Cumulated Gain (DCG) (Shilane et al. [2004]), and

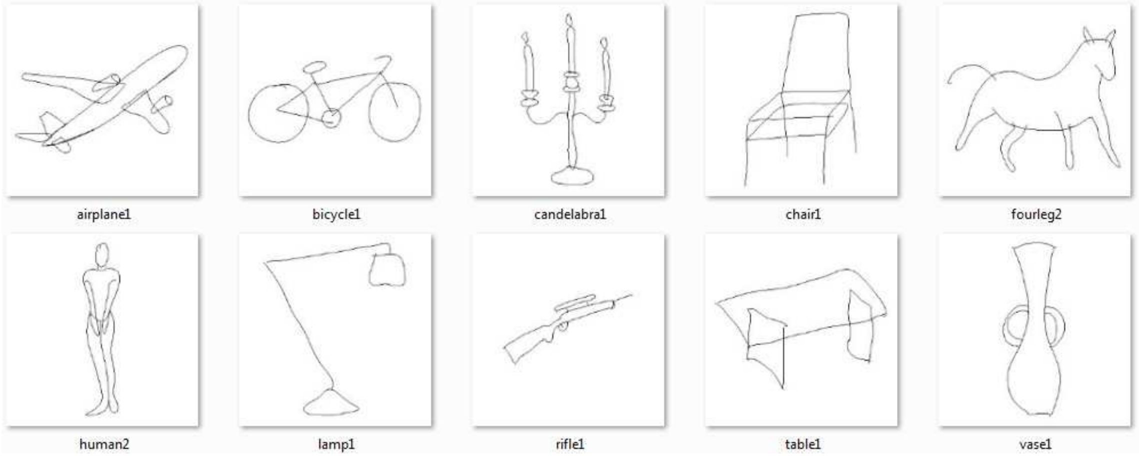


Figure 4.1: Ten 2D sketch queries.

Figure 4.2: Example 3D models.

Average Precision (AP) (Li and Johan [2013]). Their meaning and definitions are listed below.

- **Precision-Recall plot (PR):** Let us assume that the total number of 3D models in the target 3D model dataset is n, and there are m relevant models in the dataset that share the same categorical class as the query, while we have successfully retrieved p models in top k ($1\leq k \leq n$) ranking list. Then, precision P is the accuracy of top k ranking list, that is,

$$P = \frac{p}{k}. \tag{4.1}$$

  Recall R is to compute how much percentage of the relevant models has been retrieved among the top k results, that is,

$$R = \frac{p}{m}. \tag{4.2}$$

- **Nearest Neighbor (NN):** NN is the precision of the topmost model.

- **First Tier (FT):** FT is the recall of the top k result list.

- **Second Tier (ST):** ST is the recall of the top $2k$ results.

- **E-Measure (E):** It is common that people have more interest in the search results on the first page, which can fit the top 32 results. E-Measure is just defined to calculate the overall performance of the results in the first page,

$$E = \frac{2}{\frac{1}{P} + \frac{1}{R}}. \tag{4.3}$$

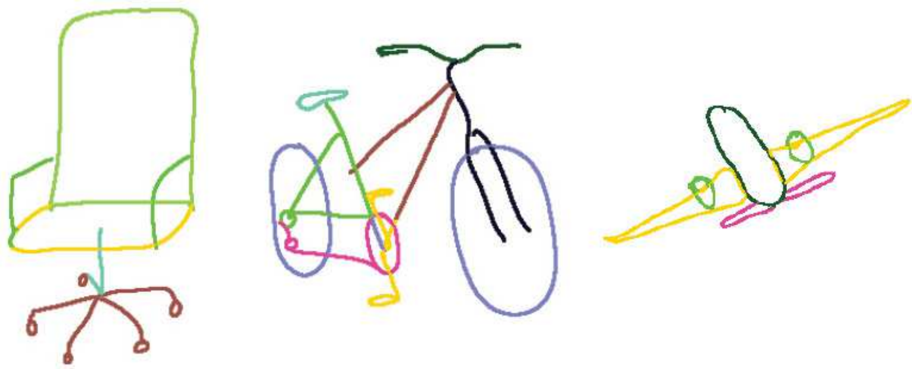


Figure 4.3: 2D sketch segmentation and labeling examples in Huang et al. [2014].

- **Average Precision (AP):** AP is to measure the overall performance by counting the total area under the Precision-Recall curve. It combines both precision and recall performance.

We need to mention that a higher value indicates better performance for each of the above six metrics. When computing the overall performance for the whole set of queries, we average each computed performance value over all the queries and utilize interpolation to generate PR curves for each query.

4.3 Experimental Results

4.3.1 *Step 1 & 2: 2D Sketch Segmentation and Labeling*

We employ the 2D sketch segmentation and labeling method in Huang et al. [2014]. For example, Fig 4.3 shows several example results. Different colored strokes correspond to different components of a sketch: for example, the chair sketch has been segmented and labeled into eight parts: back, gas lift, base, stretcher, seat, rail, stile and arm.

Sketch	Component Senses
Airplane	engine 1 engine 1 stabilizer 2 stabilizer 2 stabilizer 2 wing 2 wing 2 body 5
Bicycle	frame 8 fork 3 chain 3 handle 1 wheel 1 wheel 1 frame 1 saddle 5
Candelabra	arm 2 arm 2 shaft 3 flame 2 flame 2 flame 2 candle 1 candle 1 candle 1 base 2
Chair	armrest 1 seat 4 stretcher 2 leg 3 leg 3 leg 3 leg 3 stile 1 stile 1 back 8 stretcher 1 stretcher 1 stretcher 1
Quadruped	tail 1 leg 2 leg 2 leg 2 leg 2 head 1 ear 1 ear 1 body 5
Human	foot 1 hand 1 hand 1 leg 1 leg 1 leg 1 leg 1 body 1 head 1 arm 1 arm 1 leg 1 leg 1 foot 1
Lamp	shade 3 base 2 tube 1 tube 1
Rifle	sights 1 butt 1 trigger 1 handgrip 1 body 5 barrel 1
Table	leg 3 leg 3 top 1
Vase	body 5 handle 1 handle 1 lip 4

Figure 4.4: Manual word sense disambiguation for the component labels of the ten query sketches.

Sketch	Component Senses
Airplane	engine 3 engine 3 stabilizer 3 stabilizer 3 stabilizer 3 wing 1 wing 1 body 2
Bicycle	frame 8 fork 5 chain 8 handle 1 wheel 1 wheel 1 frame 8 saddle 1
Candelabra	arm 1 arm 1 shaft 3 fire 2 fire 2 fire 2 candle 2 candle 2 candle 2 base 9
Chair	armrest 1 seat 4 stretcher 4 leg 1 leg 1 leg 1 leg 1 stile 1 stile 1 back 5 stretcher 4 stretcher 4 stretcher 4
Quadruped	tail 1 leg 1 leg 1 leg 1 leg 1 head 1 ear 1 ear 1 body 1
Human	foot 1 hand 1 hand 1 leg 1 leg 1 leg 1 leg 1 body 1 head 1 arm 1 arm 1 leg 1 leg 1 foot 1
Lamp	shade 2 base 9 tube 1 tube 1
Rifle	sights 3 butt 9 trigger 1 handgrip 1 body 2 barrel 2
Table	leg 1 leg 1 top 9
Vase	body 2 handle 1 handle 1 lip 1

Figure 4.5: Automatic word sense disambiguation for the component labels of the ten query sketches based on the Lesk relatedness metric.

4.3.2 *Step 3: Word Sense Disambiguation*

In this section, we present our manual (as a baseline) and automatic word sense disambiguation results for all the ten queries, which are shown in Fig. 4.4 and Fig. 4.5. Sense numbers in the latest WordNet v3.1 is used. Based on the results, the step still has much room for further improvement.

4.3.3 *Step 4: Semantic Distance Computation*

After we know the word sense of each component of the query, we want to measure the semantic distance between those component names, together with their senses, and the names (and their senses) of all the target 3D models' classes. Fig. 4.6∼4.8 show the sketch segmentation and labeling results as well as the WordNet gloss and hierarchy information. Three columns of each table show the example sketches, components' labels, as well as WodNet gloss and hierarchy inquiry results, respectively. Based on them, we can observe the relatedness between the segments' names and the gloss of the models' class names in the context of the WordNet hierarchy.

To fuse the component-wise relatedness values, two approaches have been developed: assign each component an equal weight toward the final semantic distance; assign a weight to each component according to its component length. In the following, to evaluate the optimal performance of this step, we list the performance based on the manually word sense disambiguation approach. We also found that if based on the above automatic sense disambiguation method, the performance metrics values are 50%∼75% of the optimal ones presented in the following. For example, the best performance will be: NN: 0.2, FT: 0.317, ST: 0.5, E-Measure: 0.238, DCG: 0.5912, AP: 0.3698. The *hso* (Hirst and St-Onge [1998]) relatedness is adopted according to its best performance among the 11 similarity metrics available in WordNet::Similarity (Pedersen et al. [2005]).

4.3.3.1 Equal Weight Component Relatedness

To compute the final semantic distance, we may assign the same weight for each component within a sketch. However, we can also assign a weight toward the sketch-model semantic distance according to the sketch's component complexity degree, which is measured based on the number of components in the sketch.

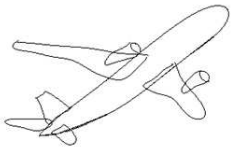
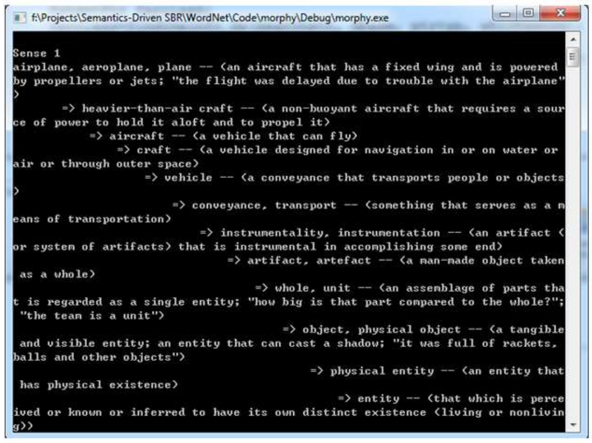
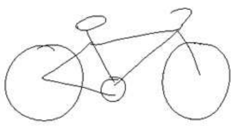
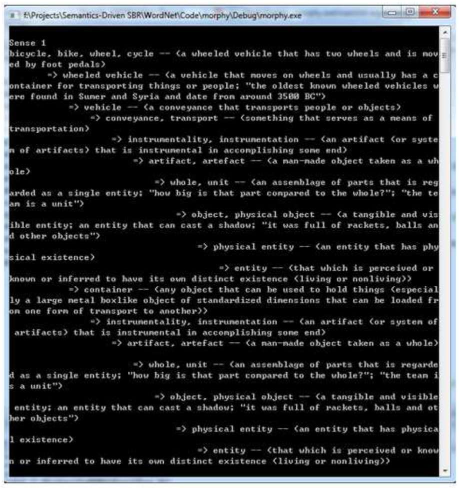
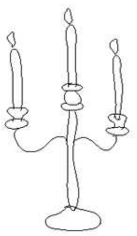
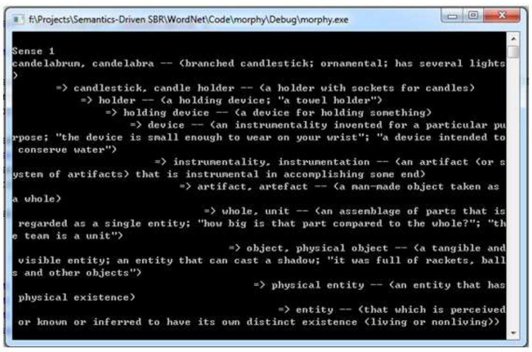
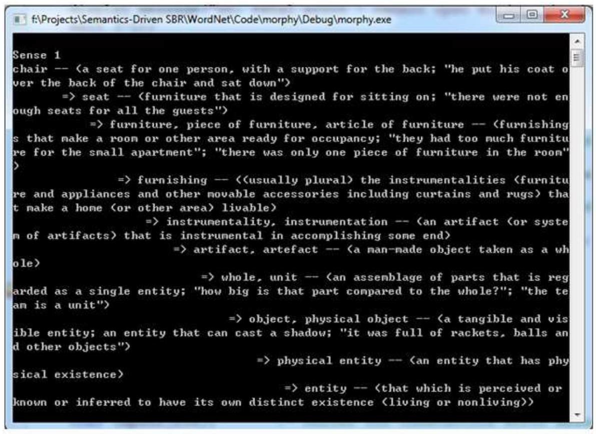
<table>
<tr><th>2D Sketch Example</th><th>Components</th><th>WordNet Gloss and Hierarchy</th></tr>
<tr><td>Airplane:
</td><td>Engine
Engine
Vertical stabilizer
Horizontal stabilizer
Horistab
Wing
Wing
Body</td><td></td></tr>
<tr><td>Bicycle:
</td><td>Back frame
Fork
Chain
Handle
Wheel
Wheel
Front frame
Saddle</td><td></td></tr>
<tr><td>Candelabra:
</td><td>Arm
Arm
Shaft
Fire
Fire
Fire
Candle
Candle
Candle
Base</td><td></td></tr>
</table>

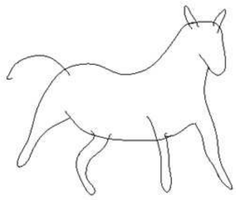
Figure 4.6: Example sketches' components and WordNet gloss and hierarchy information: Airplane, Bicycle and Candelabra.

Chair:

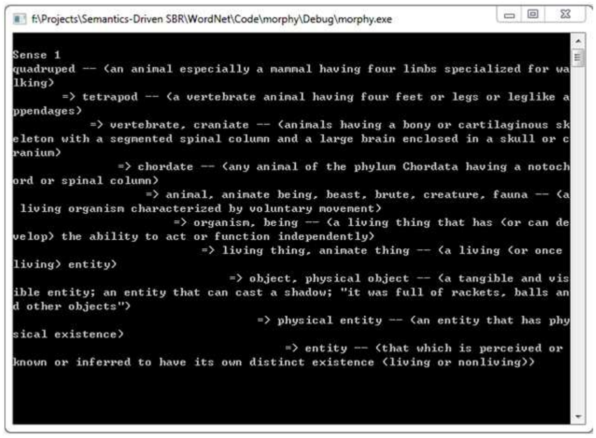
Rail
Seat
Stretcher
Leg
Leg
Leg
Leg
Stile
Stile
Back
Stretcher
Stretcher
Stretcher



Quadruped:



Tail
Leg
Leg
Leg
Leg
Head
Ear
Ear
Body



Human:

Foot
Hand
Hand
Leg
Leg
Leg
Leg
Body
Head
Arm
Arm
Leg
Leg
Foot

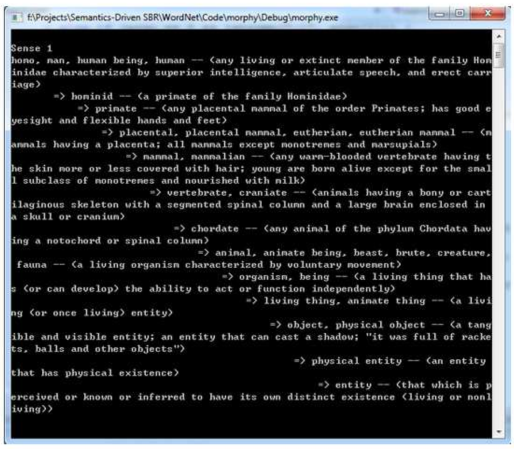


Figure 4.7: Example sketches' components and WordNet gloss and hierarchy information: Chair, Quadruped and Human.

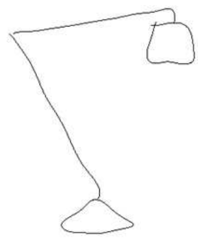
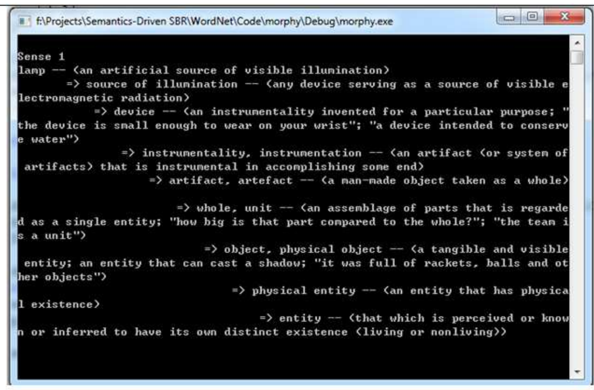
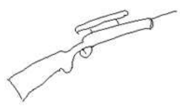
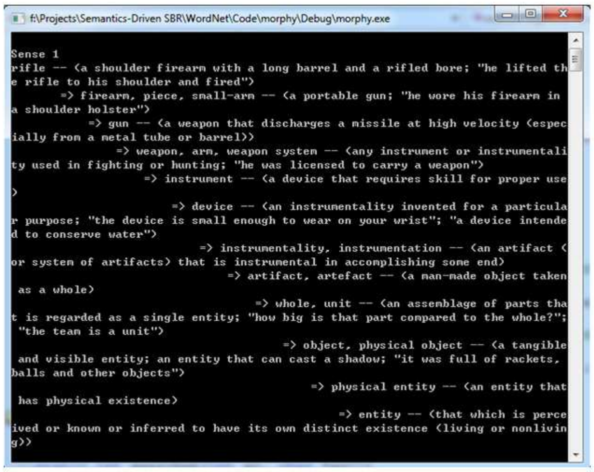
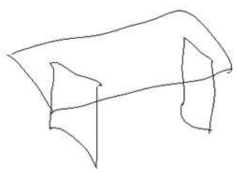
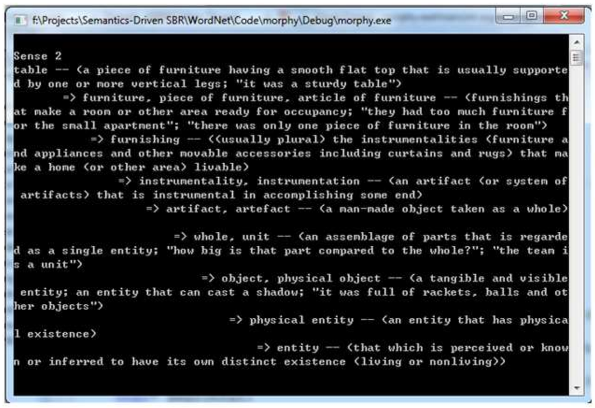
<table>
<tr>
<td>Lamp:
</td>
<td>Shade
Base
Tube
Tube</td>
<td>
</td>
</tr>
<tr>
<td>Rifle:
</td>
<td>Sight
Butt
Trigger
Handgrip
Body
Barrel</td>
<td>
</td>
</tr>
<tr>
<td>Table:
</td>
<td>Leg
Leg
Top</td>
<td>
</td>
</tr>
</table>

Figure 4.8: Example sketches' components and WordNet gloss and hierarchy information: Lamp, Rifle and Table.

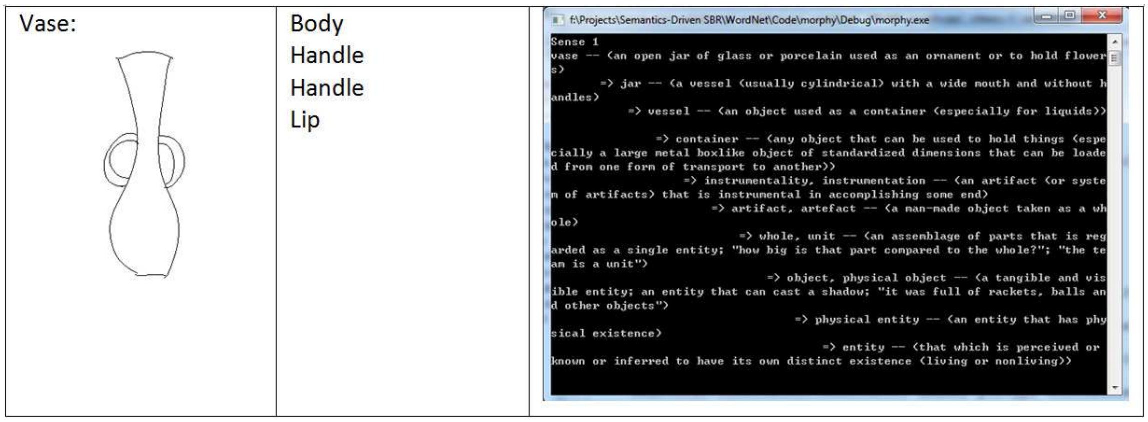


Figure 4.9: Example sketches' components and WordNet gloss and hierarchy information: Vase.

Therefore, three relatedness addition methods have been developed: Average, Sum and Product. The Average method divides the total sum of component-wise relatedness values by the total number of components in the sketch query. The Sum method directly adds all component-wise relatedness values together in order to integrate the sketch's component complexity degree into the semantic distance. The Product approach further multiplies the value computed in the Sum method by the number of components in order to assign bigger weights in this sketch complexity aspect. We have found that more product operations will not result changes in the final ranking results.

Fig. 4.10 and Table 4.1 compare their performance in terms of the seven evaluation metrics. As can be seen, the Product method performs the best, consecutively followed by the Sum and Average methods.

We also show the semantic query-class relatedness matrix for the 10 queries in Fig. 4.11. As can be seen, usually there are non-trivial differences in the *hso* metric

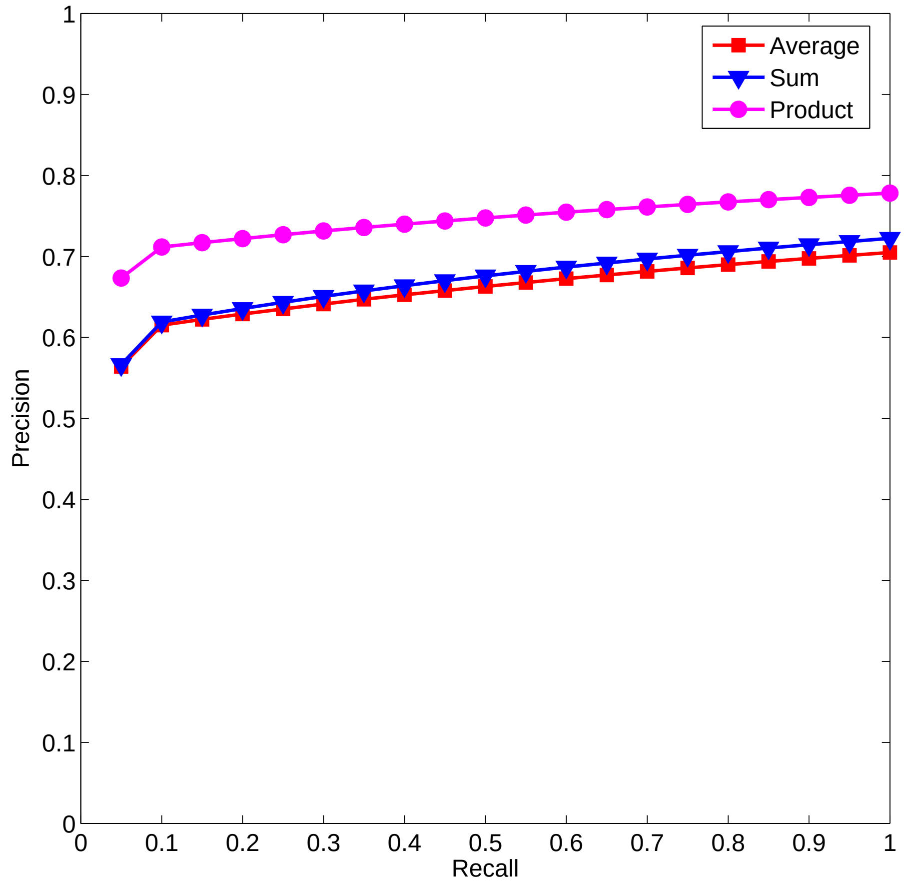


Figure 4.10: Comparison of Precision-Recall plots of the three equal component-wise weighting methods.

values for different classes, which helps us to differentiate different classes.

4.3.3.2 Length-Weighted Component Relatedness

Another method to combine the component-wise relatedness values is using the number of pixels in each component, that is component length. We compute each component length, then divide it by the total length of all the components to compute the weight of each component towards its contribution to the final semantic relatedness value.

31	8	0	12	4	5	10	0	12	0
15	**26**	0	4	3	4	0	6	3	0
14	5	0	0	0	0	0	2	0	0
29	25	0	**33**	0	0	12	14	31	0
0	0	0	0	**40**	**40**	0	0	0	0
0	0	0	0	40	**72**	0	0	0	0
8	**11**	0	2	0	0	6	8	0	0
3	15	0	0	4	5	0	**20**	0	0
6	6	0	**8**	0	0	4	4	**8**	0
2	12	0	0	4	5	0	8	0	**20**

Figure 4.11: Semantic query-class relatedness matrix for the 10 queries based on the *hso* metric. Each row is for a query, while each column indicates a class.

Table 4.1: Comparison of six performance metrics of the three equal component-wise weighting methods.

Benchmark	**NN**	**FT**	**ST**	**E**	**DCG**	**AP**
Average	0.6000	0.6000	0.6852	0.4453	0.7873	0.6609
Sum	0.6000	0.6000	0.7590	0.4453	0.7941	0.6726
Product	0.7000	0.7000	0.7852	0.5141	0.8361	0.7459

Similarly, we also consider the same three sketch-level weighting methods (Average, Sum, and Product) as above. Fig. 4.12 and Table 4.2 compare their performance, from which we can draw a similar conclusion. While, overall the three equal weighting approaches in Section 4.3.3.1 achieve better performance.

Table 4.2: Comparison of six performance metrics of the three length-based weighting methods.

Benchmark	**NN**	**FT**	**ST**	**E**	**DCG**	**AP**
Average	0.5000	0.5000	0.6852	0.4145	0.7397	0.5928
Sum	0.5000	0.5000	0.7262	0.4145	0.7444	0.6005
Product	0.6000	0.6000	0.7852	0.4833	0.7884	0.6778

4.3.4 ***Step 5: Ranking and Output.***

After obtaining the semantic relatedness values between each query and other 3D models, we rank the classes and the corresponding 3D models accordingly. Based on the rank list for each query, we can calculate the seven evaluation metrics. Fig. 4.13

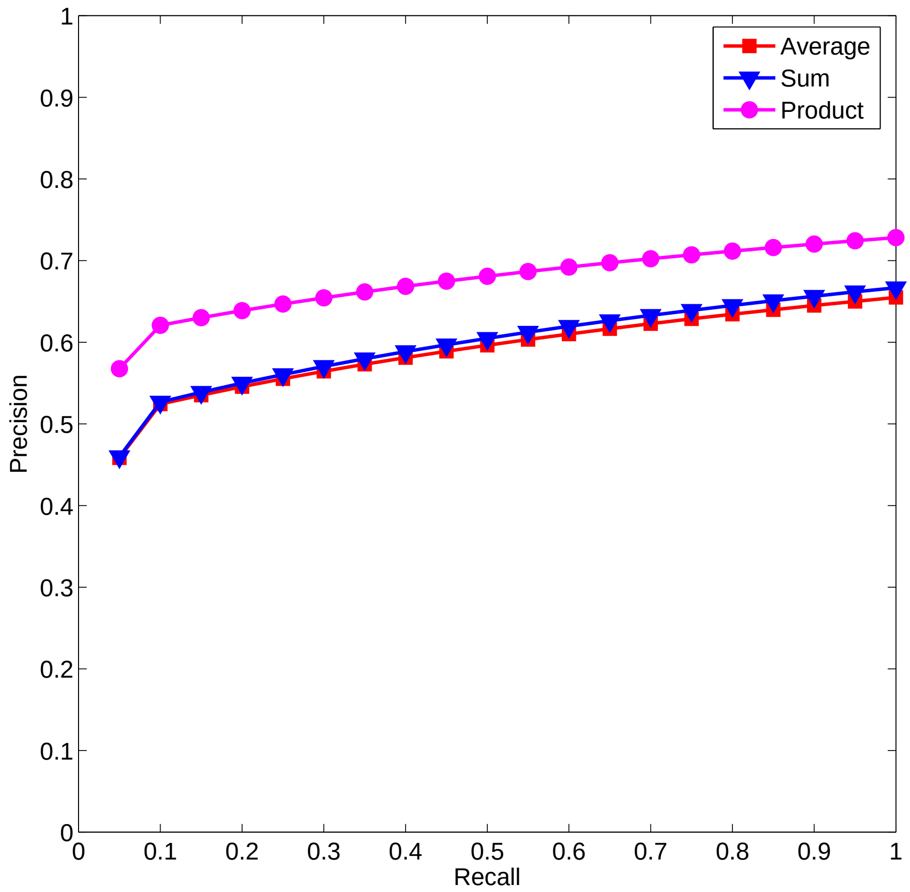


Figure 4.12: Comparison of Precision-Recall plots of the three length-based weighting methods.

lists the ranked classes (one example per class) according to the semantic relatedness for the component-wise product weighting method. As can be seen, classes are logically ranked based on the semantic approach, which meets our expectations.

In a word, as can be seen from the above two experiments' results, our semantic search approach has achieved promising results, though currently only on a small benchmark. It can semantically differentiate different classes and separate a bunch of similar classes from other different classes, which is important to bridge the semantic gap existing in content-based 3D model retrieval approaches.

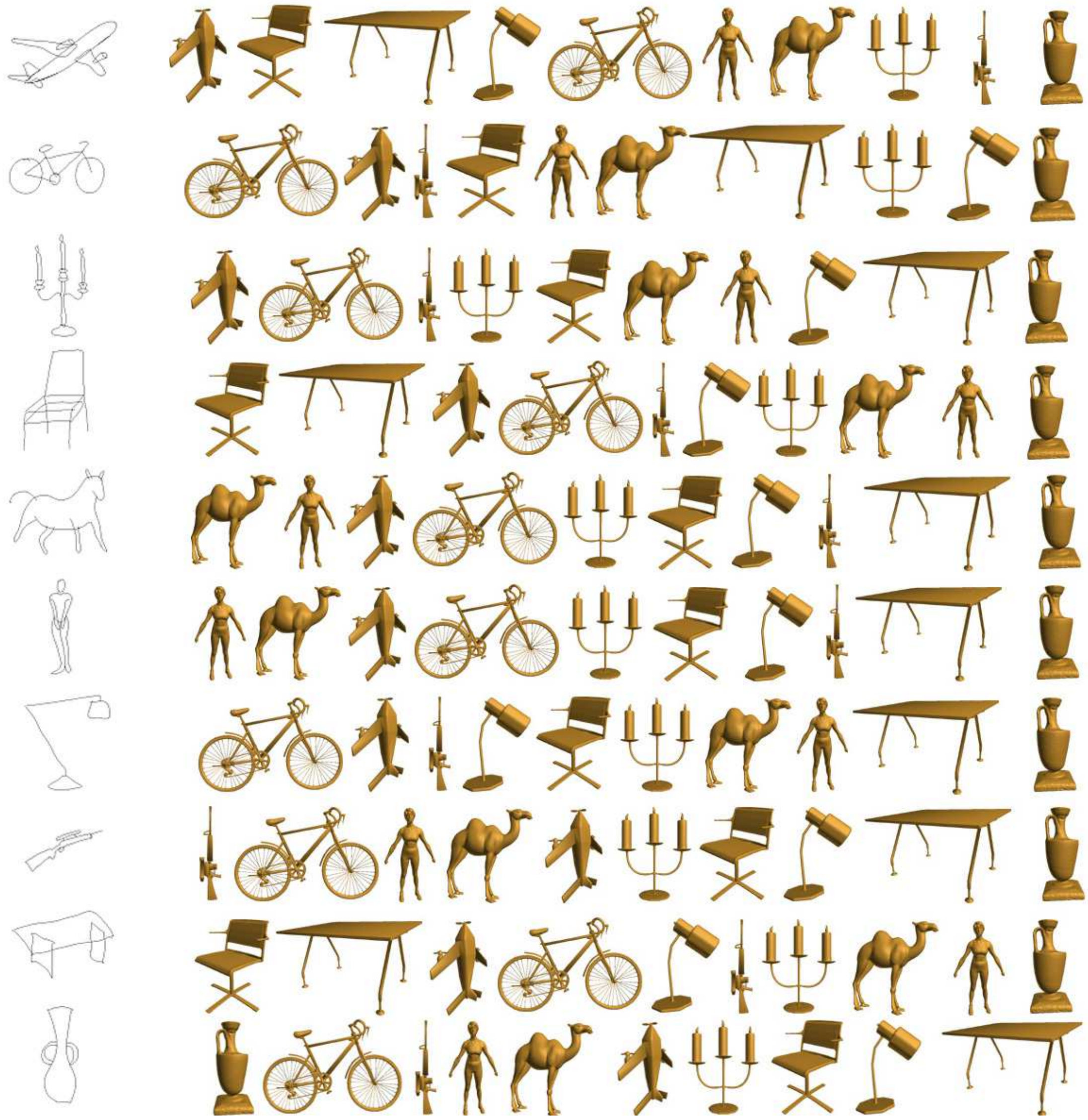


Figure 4.13: Ranking classes for the 10 queries. One example of each of the 10 classes is displayed according to their ranking orders based on their semantic relatedness values.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

In this thesis, we propose a semantic approach to retrieve 3D models for a 2D sketch query. To measure the semantic distance between a query sketch and a 3D model, we utilize WordNet to compare the semantic distance between the labels of sketch components and 3D model class. During the comparison, we also perform word sense disambiguation to assign correct meaning for components' labels. Experiments based on 10 classes of 2D sketches and 3D models have demonstrated its promising performance in bridging the semantic gap existing in traditional content-based 3D model retrieval approaches which rely on direct comparison of 2D and 3D shape information.

5.2 Future Work

5.2.1 Hybrid Approach: Integrating Both Semantic and Content-based Distances

Semantic approaches are helpful in deciding several closest classes for a query sketch, while content-based algorithms outperform in differentiating a 2D sketch and 3D models belonging to certain classes that are close to the class of the query sketch. Therefore, it is good if we can first shortlist several candidate classes for a query sketch based on a semantic approach, and then compare the 2D sketch and all the models in those shortlisted candidate classes in order to improve the retrieval accuracy, especially FT and ST.

5.2.2 Word Sense Disambiguation

Typical word sense disambiguation is in the context of a sentence, text, or even a book. However, for our situation, we only have distinct labels for a set of components that are present in the sketch. We may not be able to logically form a sentence using those component labels only. Therefore, it is a more challenging task if only based on component labels, which has been demonstrated in our word sense disambiguation experiments. A new and better sense disambiguation approach deserves our further research in the scenario of this sketch-based 3D model retrieval application.

5.2.3 Automatic 2D Sketch Segmentation and Labeling

During the research, we have found a lack of good and general algorithms in automatic 2D sketch segmentation and labeling. Therefore, it is important to develop good quality algorithms in this area. An initial idea is during the sketch labeling process, we consider all the component labels simultaneously by considering their semantic relationships in the context of WordNet hierarchy.

REFERENCES

Alahi, A., Ortiz, R., and Vandergheynst, P. (2012). FREAK: Fast retina keypoint. In *CVPR*, pages 510–517.

Aslandogan, Y. A., Thier, C., Yu, C. T., Zou, J., and Rishe, N. (1997). Using semantic contents and WordNet in image retrieval. In *SIGIR '97: Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, July 27-31, 1997, Philadelphia, PA, USA*, pages 286–295. ACM.

Attene, M., Falcidieno, B., and Spagnuolo, M. (2006). Hierarchical mesh segmentation based on fitting primitives. *The Visual Computer*, 22(3):181–193.

Banerjee, S. and Pedersen, T. (2002). An adapted lesk algorithm for word sense disambiguation using WordNet. In Gelbukh, A. F., editor, *Computational Linguistics and Intelligent Text Processing, Third International Conference, CICLing 2002, Mexico City, Mexico, February 17-23, 2002, Proceedings*, volume 2276 of *Lecture Notes in Computer Science*, pages 136–145. Springer.

Banerjee, S. and Pedersen, T. (2003). Extended gloss overlaps as a measure of semantic relatedness. In Gottlob, G. and Walsh, T., editors, *IJCAI*, pages 805–810. Morgan Kaufmann.

Bay, H., Ess, A., Tuytelaars, T., and Gool, L. J. V. (2008). Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3):346–359.

Bosch, A., Zisserman, A., and Muñoz, X. (2007). Representing shape with a spatial pyramid kernel. In *CIVR*, pages 401–408.

Cai, J., Zha, Z.-J., Luan, H.-B., Zhang, S., and Tian, Q. (2013). Learning attribute-aware dictionary for image classification and search. In *ICMR*, pages 33–40.

Cai, J., Zha, Z.-J., Zhou, W., and Tian, Q. (2012). Attribute-assisted reranking for web image retrieval. In *ACM Multimedia*, pages 873–876.

Calonder, M., Lepetit, V., Özuysal, M., Trzcinski, T., Strecha, C., and Fua, P. (2012). Brief: Computing a local binary descriptor very fast. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(7):1281–1298.

Catalano, C. E., Robbiano, F., Parascandolo, P., Cesario, L., Vosilla, L., Barbieri, F., Spagnuolo, M., Viano, G., and Cimmino, M. A. (2012). Exploiting 3D part-based analysis, description and indexing to support medical applications. In *MCBR-CDS*, pages 21–32.

Chen, B.-C., Chen, Y.-Y., Kuo, Y.-H., and Hsu, W. H. (2013). Scalable face image retrieval using attribute-enhanced sparse codewords. *IEEE Transactions on Multimedia*, 15(5):1163–1173.

Chen, X., Golovinskiy, A., and Funkhouser, T. A. (2009). A benchmark for 3D mesh segmentation. *ACM Trans. Graph.*, 28(3).

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR (1)*, pages 886–893.

Douze, M., Ramisa, A., and Schmid, C. (2011). Combining attributes and Fisher vectors for efficient image retrieval. In *CVPR*, pages 745–752.

Dragoni, M., da Costa Pereira, C., and Tettamanzi, A. G. (2012). A conceptual representation of documents and queries for information retrieval systems by using light ontologies. *Expert Systems with Applications*, 39(12):10376 – 10388.

Eitz, M., Hays, J., and Alexa, M. (2012a). How do humans sketch objects? *ACM Trans. Graph.*, 31(4):44.

Eitz, M., Hildebrand, K., Boubekeur, T., and Alexa, M. (2010). An evaluation of descriptors for large-scale image retrieval from sketched feature lines. *Computers & Graphics*, 34(5):482–498.

Eitz, M., Hildebrand, K., Boubekeur, T., and Alexa, M. (2011a). Sketch-based image retrieval: Benchmark and bag-of-features descriptors. *IEEE Transactions on Visualization and Computer Graphics*, 17(11):1624–1636.

Eitz, M., Hildebrand, K., Boubekeur, T., and Alexa, M. (2011b). Sketch-based image retrieval: Benchmark and bag-of-features descriptors. *IEEE Trans. Vis. Comput. Graph.*, 17(11):1624–1636.

Eitz, M., Richter, R., Boubekeur, T., Hildebrand, K., and Alexa, M. (2012b). Sketch-based shape retrieval. *ACM Trans. Graph.*, 31(4):31:1–31:10.

Farhadi, A., Endres, I., and Hoiem, D. (2010). Attribute-centric recognition for cross-category generalization. In *CVPR*, pages 2352–2359.

Farhadi, A., Endres, I., Hoiem, D., and Forsyth, D. A. (2009). Describing objects by their attributes. In *CVPR*, pages 1778–1785.

Farhadi, A., Forsyth, D. A., and White, R. (2007). Transfer learning in sign language. In *CVPR*.

Ferrari, V. and Zisserman, A. (2007). Learning visual attributes. In *NIPS*.

Furuya, T. and Ohbuchi, R. (2013). Ranking on cross-domain manifold for sketch-based 3D model retrieval. In *Cyberworlds 2013*, pages 274–281.

Gao, B., Zheng, H., and Zhang, S. (2009). An overview of semantics processing in content-based 3D model retrieval. In *Artificial Intelligence and Computational Intelligence, 2009. AICI '09. International Conference on*, volume 2, pages 54–59.

Golovinskiy, A. and Funkhouser, T. (2008). Randomized cuts for 3D mesh analysis. *ACM Transactions on Graphics (Proc. SIGGRAPH ASIA)*, 27(5).

Gong, B., Liu, J., Wang, X., and Tang, X. (2013). Learning semantic signatures for 3D object retrieval. *IEEE Transactions on Multimedia*, 15(2):369–377.

Gonzalo, J., Verdejo, F., Chugur, I., and Cigarrán, J. M. (1998). Indexing with WordNet synsets can improve text retrieval. In *Proceedings of the COLING/ACL '98 Workshop on Usage of WordNet for NLP, Montreal, Quebec, Canada*.

Hirst, G. and St-Onge, D. (1998). Lexical chains as representations of context for the detection and correction of malapropisms. In *WordNet: An Electronic Lexical Database*, pages 305–332.

Huang, Z., Fu, H., and Lau, R. W. H. (2014). Data-driven segmentation and labeling of freehand sketches. *ACM Trans. Graph.*, 33(6):175:1–175:10.

Kalogerakis, E., Hertzmann, A., and Singh, K. (2010). Learning 3D mesh segmentation and labeling. *ACM Trans. Graph.*, 29(4).

Karger, D. R. and Stein, C. (1996). A new approach to the minimum cut problem. *J. ACM*, 43(4):601–640.

Katz, S. and Tal, A. (2003). Hierarchical mesh decomposition using fuzzy clustering and cuts. *ACM Trans. Graph.*, 22(3):954–961.

Kemp, C., Tenenbaum, J. B., Griffiths, T. L., Yamada, T., and Ueda, N. (2006). Learning systems of concepts with an infinite relational model. In *AAAI*, pages 381–388.

Kim, D. H., Yun, I. D., and Lee, S. U. (2004). Interactive 3-D shape retrieval system using the attributed realtional graph. In *MDDE*.

Lampert, C. H., Nickisch, H., and Harmeling, S. (2009). Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, pages 951–958.

Leacock, C. and Chodorow, M. (1998). Combining Local Context and WordNet Similarity for Word Sense Identification. In Fellbaum, C., editor, *WordNet: An Electronic Lexical Database*, pages 265–283. MIT Press.

Lei, Y.-H., Chen, Y.-Y., Chen, B.-C., Iida, L., and Hsu, W. H. (2012). Where is who: large-scale photo retrieval by facial attributes and canvas layout. In *SIGIR*, pages 701–710.

Lesk, M. (1986). Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the 5th Annual International Conference on Systems Documentation*, SIGDOC '86, pages 24–26, New York, NY, USA. ACM.

Leutenegger, S., Chli, M., and Siegwart, R. (2011). BRISK: Binary robust invariant scalable keypoints. In *ICCV*, pages 2548–2555.

Li, B. and Johan, H. (2013). 3D model retrieval using hybrid features and class information. *Multimedia Tools Appl.*, 62(3):821–846.

Li, B., Lu, Y., Godil, A., Schreck, T., Aono, M., Johan, H., Saavedra, J. M., and Tashiro, S. (2013). SHREC'13 track: Large scale sketch-based 3D shape retrieval. In *3DOR*, pages 89–96.

Li, B., Lu, Y., Godil, A., Schreck, T., Bustos, B., Ferreira, A., Furuya, T., Fonseca, M. J., Johan, H., Matsuda, T., Ohbuchi, R., Pascoal, P. B., and Saavedra, J. M. (2014a). A comparison of methods for sketch-based 3D shape retrieval. *Computer Vision and Image Understanding*, 119:57–80.

Li, B., Lu, Y., Li, C., Godil, A., Schreck, T., Aono, M., Burtscher, M., Chen, Q., Chowdhury, N. K., Fang, B., Fu, H., Furuya, T., Li, H., Liu, J., Johan, H., Kosaka, R., Koyanagi, H., Ohbuchi, R., Tatsuma, A., Wan, Y., Zhang, C., and Zou, C. (2015). A comparison of 3D shape retrieval methods based on a large-scale benchmark supporting multimodal queries. *Computer Vision and Image Understanding*, 131:1–27.

Li, B., Lu, Y., Li, C., Godil, A., Schreck, T., Aono, M., Burtscher, M., Fu, H., Furuya, T., Johan, H., Liu, J., Ohbuchi, R., Tatsuma, A., and Zou, C. (2014b). SHREC'14 Track: Extended large scale sketch-based 3D shape retrieval. In *Eurographics Workshop on 3D Object Retrieval, Strasbourg, France, 2014. Proceedings*, pages 121–130.

Li, L.-J., Wang, C., Lim, Y., Blei, D. M., and Li, F.-F. (2010). Building and using a semantivisual image hierarchy. In *CVPR*, pages 3336–3343.

Liang, L. and Grauman, K. (2014). Beyond comparing image pairs: Setwise active learning for relative attributes. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 208–215.

Lin, Y.-L. (2012). Investigating 3D model and part information for improving content-based and attribute-based object retrieval. In *ACM Multimedia*, pages 1409–1412.

Lin, Y.-L., Tsai, M.-K., Hsu, W. H., and Chen, C.-W. (2013). Investigating 3-D model and part information for improving content-based vehicle retrieval. *IEEE Trans. Circuits Syst. Video Techn.*, 23(3):401–413.

Liu, R. and Zhang, H. (2004). Segmentation of 3D meshes through spectral clustering. In *Proc. of Pacific Graphics*, pages 298–305.

Liu, S., Liu, F., Yu, C. T., and Meng, W. (2004). An effective approach to document retrieval via utilizing WordNet and recognizing phrases. In *SIGIR*, pages 266–272.

Liu, S., Yu, C. T., and Meng, W. (2005). Word sense disambiguation in queries. In *CIKM*, pages 525–532.

Mainali, P., Lafruit, G., Yang, Q., Geelen, B., Gool, L. J. V., and Lauwereins, R. (2013). SIFER: Scale-invariant feature detector with error resilience. *International Journal of Computer Vision*, 104(2):172–197.

Marszalek, M. and Schmid, C. (2007). Semantic hierarchies for visual object recognition. In *CVPR*.

Merriam-Webster (2015). `http://www.merriam-webster.com/`.

Miller, G. A. (1995). WordNet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Newell, A. J. and Griffin, L. D. (2011). Multiscale histogram of oriented gradient descriptors for robust character recognition. In *ICDAR*, pages 1085–1089.

OpenCV (2013). `http://docs.opencv.org/modules/features2d/doc/feature_detection_and_description.html`.

Osherson, D. N., Stern, J., Wilkie, O., Stob, M., and Smith, E. E. (1991). Default probability. *Cognitive Science*, 15(2):251–269.

Parikh, D. and Grauman, K. (2011). Relative attributes. In Metaxas, D. N., Quan, L., Sanfeliu, A., and Gool, L. J. V., editors, *ICCV*, pages 503–510. IEEE.

Patwardhan, S. (2003). *Incorporating dictionary and corpus information into a context vector measure of semantic relatedness*. Master's thesis, Univ. of Minnesota, Duluth.

Patwardhan, S., Banerjee, S., and Pedersen, T. (2003). Using measures of semantic relatedness for word sense disambiguation. In Gelbukh, A. F., editor, *CICLing*, volume 2588 of *Lecture Notes in Computer Science*, pages 241–257. Springer.

Pedersen, T. and Kolhatkar, V. (2009). WordNet: : SenseRelate: : AllWords - A broad coverage word sense tagger that maximizes semantic relatedness. In Popowich, F. and Johnston, M., editors, *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, May 31 - June 5, 2009, Boulder, Colorado, USA, Demos*, pages 17–20. The Association for Computational Linguistics.

Pedersen, T. and Michelizzi, J. (2006). `https://metacpan.org/pod/WordNet::SenseRelate::WordToSet/`.

Pedersen, T. and Michelizzi, J. (2008). `http://search.cpan.org/~tpederse/WordNet-SenseRelate-WordToSet/`.

Pedersen, T., Pakhomov, S. V. S., Patwardhan, S., and Chute, C. G. (2007). Measures of semantic similarity and relatedness in the biomedical domain. *Journal of Biomedical Informatics*, 40(3):288–299.

Pedersen, T., Patwardhan, S., and Banerjee, S. (2006). `https://metacpan.org/pod/WordNet::SenseRelate::TargetWord/`.

Pedersen, T., Patwardhan, S., and Michelizzi, J. (2004). WordNet: : Similarity - measuring the relatedness of concepts. In McGuinness, D. L. and Ferguson, G., editors, *AAAI*, pages 1024–1025. AAAI Press / The MIT Press.

Pedersen, T., Patwardhan, S., Michelizzi, J., and Banerjee, S. (2005). `https://metacpan.org/pod/WordNet::Similarity/`.

Perronnin, F., Liu, Y., Sánchez, J., and Poirier, H. (2010). Large-scale image retrieval with compressed fisher vectors. In *CVPR*, pages 3384–3391.

Rosten, E. and Drummond, T. (2006). Machine learning for high-speed corner detection. In *ECCV (1)*, pages 430–443.

Rublee, E., Rabaud, V., Konolige, K., and Bradski, G. R. (2011). ORB: An efficient alternative to SIFT or SURF. In *ICCV*, pages 2564–2571.

Russakovsky, O. and Li, F.-F. (2010). Attribute learning in large-scale datasets. In *ECCV Workshops (1)*, pages 1–14.

Sandeep, R. N., Verma, Y., and Jawahar, C. V. (2014). Relative parts: Distinctive parts for learning relative attributes. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 3614–3621.

Shapira, L., Shamir, A., and Cohen-Or, D. (2008). Consistent mesh partitioning and skeletonisation using the shape diameter function. *The Visual Computer*, 24(4):249–259.

Shilane, P., Min, P., Kazhdan, M. M., and Funkhouser, T. A. (2004). The Princeton shape benchmark. In *SMI*, pages 167–178.

Shrivastava, A., Singh, S., and Gupta, A. (2012). Constrained semi-supervised learning using attributes and comparative attributes. In *ECCV (3)*, pages 369–383.

Siddiquie, B., Feris, R. S., and Davis, L. S. (2011). Image ranking and retrieval based on multi-attribute queries. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, pages 801–808, Washington, DC, USA. IEEE Computer Society.

Smeaton, A. F. and Quigley, I. (1996). Experiments on using semantic distances between words in image caption retrieval. In Frei, H.-P., Harman, D., Schäuble, P., and Wilkinson, R., editors, *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'96, August 18-22, 1996, Zurich, Switzerland (Special Issue of the SIGIR Forum)*, pages 174–180. ACM.

TheFreeDictionary (2015). `http://www.thefreedictionary.com/`.

Torresani, L., Szummer, M., and Fitzgibbon, A. W. (2010). Efficient object category recognition using classemes. In *ECCV (1)*, pages 776–789.

Tousch, A.-M., Herbin, S., and Audibert, J.-Y. (2012). Semantic hierarchies for image annotation: A survey. *Pattern Recognition*, 45(1):333–345.

Voorhees, E. M. (1993). Using WordNet to disambiguate word senses for text retrieval. In Korfhage, R., Rasmussen, E. M., and Willett, P., editors, *Proceedings of the 16th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Pittsburgh, PA, USA, June 27 - July 1, 1993*, pages 171–180. ACM.

Wang, X., Lv, T., Wang, S., and Wang, Z. (2008). An ontology and swrl based 3D model retrieval system. In Li, H., Liu, T., Ma, W.-Y., Sakai, T., Wong, K.-F., and Zhou, G., editors, *AIRS*, volume 4993 of *Lecture Notes in Computer Science*, pages 335–344. Springer.

Wang, X., Zhang, T., Tretter, D., and Lin, Q. (2013). Personal clothing retrieval on photo collections by color and attributes. *Multimedia, IEEE Transactions on*, PP(99):1–1.

Wu, Z. and Palmer, M. S. (1994). Verb semantics and lexical selection. In Pustejovsky, J., editor, *32nd Annual Meeting of the Association for Computational Linguistics, 27-30 June 1994, New Mexico State University, Las Cruces, New Mexico, USA, Proceedings.*, pages 133–138. Morgan Kaufmann Publishers / ACL.

Yoon, S. M., Scherer, M., Schreck, T., and Kuijper, A. (2010). Sketch-based 3D model retrieval using diffusion tensor fields of suggestive contours. In *Proc. of the international conference on Multimedia*, pages 193–200.

Yu, F. X., Ji, R., Tsai, M.-H., Ye, G., and Chang, S.-F. (2012). Weak attributes for large-scale image retrieval. In *CVPR*, pages 2949–2956.

Zhang, H., Zha, Z.-J., Yang, Y., Yan, S., Gao, Y., and Chua, T.-S. (2013a). Attribute-augmented semantic hierarchy: towards bridging semantic gap and intention gap in image retrieval. In *ACM Multimedia*, pages 33–42.

Zhang, S., Tian, Q., Lu, K., Huang, Q., and Gao, W. (2013b). Edge-SIFT: Discriminative binary descriptor for scalable partial-duplicate mobile search. *IEEE Transactions on Image Processing*, 22(7):2889–2902.