Query-Adaptive Shape Topic Mining for Hand-Drawn Sketch Recognition

Zhenbang Sun[1]*, Changhu Wang[2], Liqing Zhang[1], Lei Zhang[2]
[1]Brain-like Computing Lab, Shanghai Jiao Tong University, Shanghai 200240, P.R.China
[2]Microsoft Research Asia, No. 5 Danling Street, Beijing 100080, P.R.China
sunzhenbang@hotmail.com, chw@microsoft.com
zhang-lq@cs.sjtu.edu.cn, leizhang@microsoft.com

ABSTRACT

In this work, we study the problem of hand-drawn sketch recognition. Due to large intra-class variations presented in hand-drawn sketches, most of existing work was limited to a particular domain or limited pre-defined classes. Different from existing work, we target at developing a general sketch recognition system, to recognize any semantically meaningful object that a child can recognize. To increase the recognition coverage, a web-scale clipart image collection is leveraged as the knowledge base of the recognition system. To alleviate the problems of *intra-class shape variation* and *inter-class shape ambiguity* in this unconstrained situation, a query-adaptive shape topic model is proposed to mine object topics and shape topics related to the sketch, in which, multiple layers of information such as sketch, object, shape, image, and semantic labels are modeled in a generative process. Besides sketch recognition, the proposed topic model can also be used for related applications such as sketch tagging, image tagging, and sketch-based image search. Extensive experiments on different applications show the effectiveness of the proposed topic model and the recognition system.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing

General Terms

Algorithms, Experimentation

Keywords

Sketch Recognition, Query-adaptive Shape Topic Model, Sketch-based Image Search

*Zhenbang Sun performed this work while being an intern at Microsoft Research Asia.

MM'12, October 29–November 2, 2012, Nara, Japan.

Figure 1: Examples of hand-drawn sketches. The sketch recognition system targets at recognizing a reasonable hand-drawn sketch.

1. INTRODUCTION

Nowadays, sketching has become one of the most natural ways for human-computer interaction, especially as the increasing popularity of devices with touch screens. Thus, how a computer can recognize a human's hand-drawn sketch, as a basic problem of artificial intelligence, has attracted more and more attentions. An effective sketch recognition system can help a computer know more about human intentions, and thus will be of great value to a variety of applications, such as human-computer interaction, game design, sketch-based search, and children education.

Sketch recognition has been studied since 1990s in computer vision and graphics. Most of existing approaches for sketch recognition [1, 11, 12, 13, 18, 23, 26, 27] mainly focus on recognizing basic shapes in specific domains such as UML diagrams and mechanical engineering. The dependency on domain-specific knowledge makes it difficult to adapt these algorithms to solve problems in other domains, let alone recognize an arbitrary hand-drawn sketch.

Related work such as shape recognition and classification [2, 19, 20, 21] mainly targets at designing effective shape descriptors and matching models to handle global and/or local non-rigid shape deformations. Thus, although kept from the uncertainty bought by hand-drawn sketches, this kind of work is still limited in small-scale datasets, due to the complexity of matching models and the lack of an efficient index solution.

Figure 2: Examples of intra-class variation. The shapes of the same objects are still of large diversity.

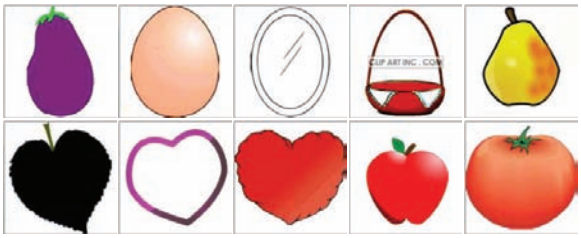

Figure 3: Examples of inter-class ambiguity. Different objects might present quite similar shapes.

In this work, we study the problem of hand-drawn sketch recognition. There is no particular constraint, and any semantically meaningful object a child can recognize might be a drawing target, as shown in Fig. 1. This differentiates the problem we are targeting at from existing work, and thus some new challenges arise, i.e. object coverage[1], intra-class shape variation, inter-class shape ambiguity, and sketch uncertainty.

Object coverage: Since there are potentially unlimited objects with typical shapes in the world, a practical sketch recognition system should have the ability to recognize as many objects as possible.

Intra-class shape variation: As shown in Fig. 2, even for one object, there might be many shapes. Traditional domain-specific shape modeling methods might not work here, since the shape models of potentially unlimited objects will be of large diversity.

Inter-class shape ambiguity: In such an unconstraint situation in terms of potentially unlimited objects and shapes, there might exist many ambiguous shapes which could represent different objects, as shown in Fig. 3. This makes the recognition process more challenging, and motivate us to find other information besides shape features.

Sketch uncertainty: Hand-drawn sketches always exhibit variations and ambiguities, as shown in Fig. 4. The recognition process should not only well reflect the shape of the sketch query, but also be robust enough to an imprecise sketch.

[1]With a slight abuse of terminology, in this work we use "object" to represent a class of objects with a same semantic meaning.

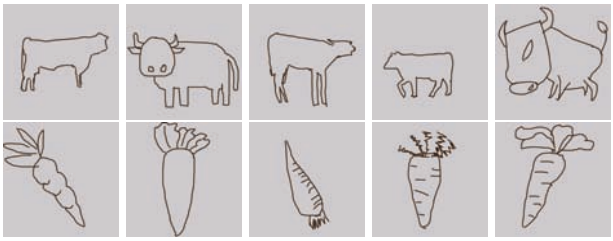

Figure 4: Examples of sketch uncertainty. The hand-drawn sketches from different persons might be quite diverse even when drawing the same object.

To increase the *object and shape coverage*, a large-scale database is highly desired to be the knowledge base of the recognition system. Thus, we collected one million clipart images from the web as the knowledge base, for the contours in clipart images have a similar style to hand-drawn sketches. Most of these images have noisy textual information such as titles and the surrounding text. Since it is impractical to model all objects and shapes in this huge knowledge base, an effective sketch-based image search technology [7] is leveraged to find visually similar images to the sketch, and then a query-adaptive image collection is obtained. Thus, the problem is reduced to how to recognize the sketch based on a collection of images with noisy words.

Due to *sketch uncertainty*, their might be different objects with variant shapes in the collection, which causes the problems of *intra-class shape variation* and *inter-class shape ambiguity*. These problems, together with the noisy descriptions of the images, make it less effective to mine keywords directly from the textual information of the collection. Thus, to alleviate these problems, we leverage both the shape and textual features to recognize the sketch.

Therefore, the task of sketch recognition is to discover *object topics* and *shape topics* from the query-adaptive image collection, both of which are actually coupled together due to the problems of *shape variation* and *ambiguity*. For example, as shown in Fig. 2 and Fig. 3, the shapes of the same object may be very different, while those of different objects may be visually similar. Therefore, for the task of sketch recognition, an effective algorithm is required to simultaneously model object topics and shape topics, with ability of handling both visual and textual features. Here an *object topic* represents a kind of object that occurs in the collection, and a *shape topic* represents a certain shape of objects.

In this work, we propose a probabilistic topic model for hand-drawn sketch recognition, i.e. Query-adaptive Shape Topic (QST) model, to simulate the generative process of an image and its textual information. In QST, two layers of latent topics, i.e. object topics and shape topics, are presented to alleviate the *shape variation* and *ambiguity* problems. More specifically, an image is supposed to contain one object topic, and this object topic generates a typical shape topic and related semantic tags. Moreover, the sketch query is not only used to find the query-adaptive image collection, but also supervises the generative process of the shape features from shape topics in QST.

The QST model is appropriate for solving the sketch recognition problem because of the following aspects. First, the

concurrent generating of textual features and shape features makes it possible to differentiate objects with similar shapes, and thus alleviate the shape ambiguity problem. Second, the representation of two-layer latent topics helps overcome the shape variation problem, since it can group together multiple shapes of one object. Moreover, this hierarchical structure is robust to hand-drawn variances, and related objects and shapes could be discovered for an imprecise sketch, as shown in Fig. 5.

As far as we know, this is the first work to target at recognizing an arbitrary but semantically meaningful hand-drawn sketch. Besides sketch recognition, the proposed system can also be used for related applications, such as sketch tagging, image tagging, and sketch-based image search. Extensive experiments on different applications show the effectiveness of the proposed QST model and sketch recognition system.

2. RELATED WORK

In this section, we present the related work in sketch recognition, shape classification, and topic models.

Sketch recognition. Sketch recognition has been studied for decades in computer vision and graphics. Early research mainly focused on domain-specific sketch recognition, such as recognizing basic shapes in UML diagrams [12], mechanical engineering [27], and webpage design [18]. In order to alleviate the substantial efforts in developing sketch interfaces in new domains, in 2003, a sketch description language LADDER [14] was introduced, based on which a domain-independent sketch recognition system was built. However, to recognize sketches in a new domain, users still need to write a sketch grammar describing the domain-specific information. Based on LADDER, sketch recognition systems such as PaleoSketch [23] were developed, most of which need strong domain knowledge, focusing on recognizing limited basic shapes, such as line, polyline, circle, ellipse, and arc. In 2008, an open-domain sketch recognition system CogSketch [11] was built, which, however, is a sketch managing tool focusing on reasoning over recognition. Note that previous systems only recognized on the order of at most 20 different shapes. In 2010, Hammond et al. [13] tried to recognize hundreds of shapes. However, the target was to recognize course-of-action diagrams instead of arbitrary natural objects. Thus, in spite of continuous efforts, most of existing recognition systems focus on specific domains with limited shapes.

Shape classification. The studies in shape recognition and classification mainly target at designing effective shape descriptors and matching models to handle global and/or local non-rigid shape deformations. Roughly speaking, shape descriptors differ according to whether they are applied to contours or regions. Contour-based shape descriptors include wavelets [22], Fourier descriptors [30], contour descriptors [20], etc. Region-based shape descriptors include [17, 25, 28], etc. Shape matching is usually influenced by the adopted descriptors. Typical algorithms include contour matching [2] to solve the optimal assignment problem to discover the mapping of two set of points, and descriptor matching [21] to measure the similarity of two set of local descriptors. Most existing studies target at limited classes and each shape has explicit labels, which are not easy to adapted for solving our problem.

Topic models. In recent years, probabilistic topic models, such as PLSA[16], LDA[5], have been widely used in semantic data mining owing to their capability of discovering meaningful latent topics. To leverage the class information of data, supervised topic models such as s-LDA[4] were proposed. Topic models were also successfully applied to problems with multi-type features such as image classification/annotation [3], in which both visual features and semantic tags exist. Moreover, some more complex models were further designed to solve specific problems. For example, for the task of web image categorization, [10] proposed a topic model which was derived from PLSA[16], by additionally considering the locations of visual words. In [24], the author-topic model, which was extended from LDA [5], was proposed to discover topics in academic articles. In the domain of multimedia document mining, [15] modeled the process of generating travelogues. However, in spite of the success in many applications, existing work is either too simple to handle the problems we meet in sketch recognition, or leverages too much domain knowledge from specific problems and thus is difficult to adapt to solve our problem. For example, PLSA and LDA were designed for one type of features, whereas our data contains both shapes and words. Corr-LDA and s-LDA cannot be easily adapted to handle so many layers including sketch, object, shape, image, and semantic words.

3. QUERY-ADAPTIVE SHAPE TOPIC MODEL

In this section, we present the proposed Query-adaptive Shape Topic (QST) model for sketch recognition. Besides the generative process of the model and its parameter estimation, we also show how to utilize the model for sketch recognition and tagging, followed by some illustrations to the effectiveness of the latent topics.

3.1 Query-Adaptive Image Collection

We collected one million clipart images from the web as a knowledge base to recognize a hand-drawn sketch. Most of these images are created by humans and might cover most of drawings of familiar objects.

For a hand-drawn sketch, we leverage the technology of Edgel Index [7, 29] and build a sketch-based clipart image search engine to search similar images to the sketch. After filtering out complex images, most clipart images in our database only contain one single object. Thus, in the clipart engine, both of the sketch query and contours of database images are normalized to a uniform location and size before searching and indexing, which makes this engine invariant to sketch/shape translation and scaling. The effectiveness of this engine has been validated in different retrieval tasks [7, 8, 29]. For details please refer to [7, 8, 29].

Based on this engine, for each hand-drawn sketch, the visually similar images to the sketch, their surrounding texts, and their matching scores, will be used as the input for the QST model to recognize the sketch query.

3.2 Problem Formulation

The sketch recognition problem is formulated as how to discover semantic topics possibly representing the sketch from the query-adaptive image collection.

We first analyze and discover the generative process for the observed information and latent topics. Since there are only simple clipart images in the collection, it is natural to assume there is only one object topic in each image. To overcome the challenge of *shape ambiguity*, besides the visual

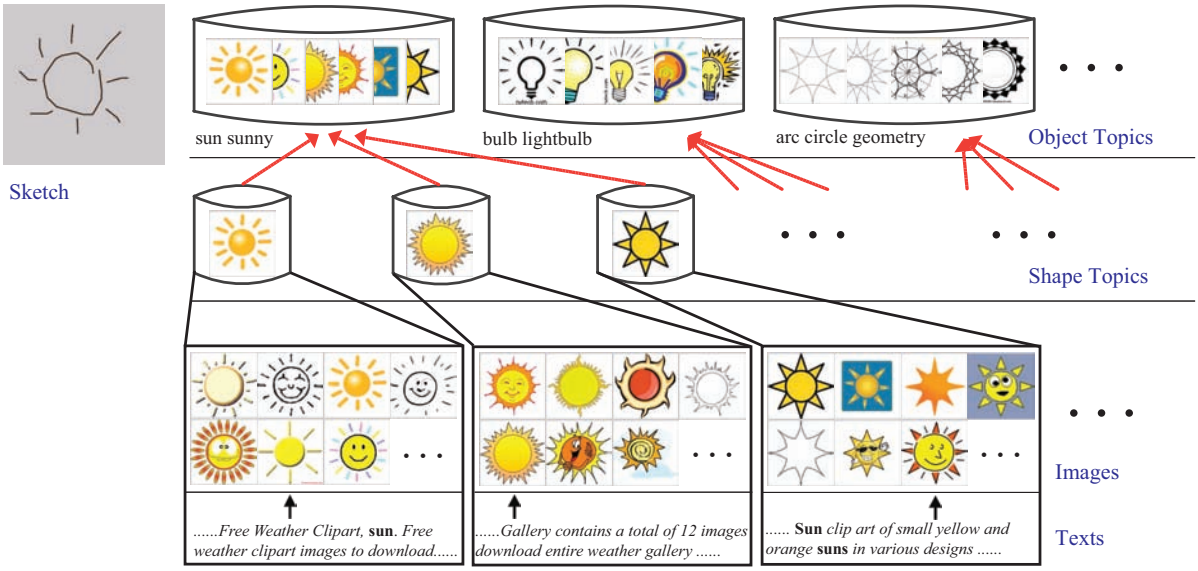

Figure 5: The hierarchical structure of the two-layer latent topics. Assume the image I has shape feature r_I. The shape topic s of image I is determined by the highest probability $p(r_I|s)$, and the object topic z which the shape topic s belongs to is determined by the highest probability $p(s|z)$.

feature, the textual information of images is also leveraged in the model. Therefore, for each image, its object topic further generates both visual information like shapes and semantic information like the surrounding text of the image.

Since one object might correspond to different shapes, it is natural to add another layer of topics, i.e. shape topics, to represent variant types of shapes related to the sketch. This can further alleviate the problems of *shape variation* and *shape ambiguity*. The shape feature is then generated according to the shape topic.

Instead of a purely unsupervised topic mining problem, in this work it is necessary to discover object topics that are more relevant to the sketch query. Different from sLDA [4], in which the supervised information (a response variable) is generated by the latent topics, in QST we use the sketch query to influence the possibility of generating a shape feature from a shape topic. This guarantees that the discovered shape topics are more relevant to the sketch query, and so are the object topics.

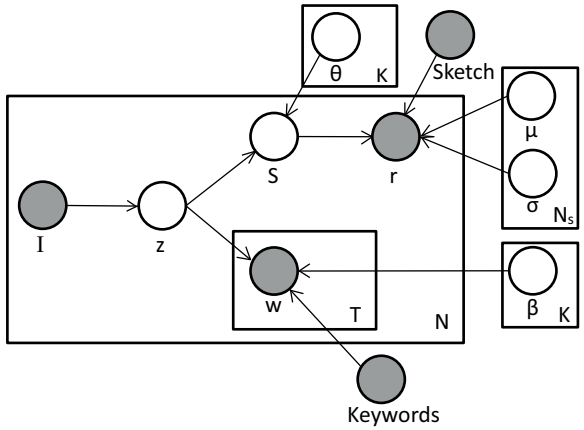

Figure 6: Graphical representation of the proposed QST model.

3.3 Generative Process

The graphical representation of the QST model is shown in Fig. 6. It should be noted that, to make the model more general, we also enable users to optionally add keywords when they draw sketches. Thus, two factors, i.e. the sketch and the keywords, will supervise the generative process, in which the keywords could be an empty set $\emptyset$.

We first introduce some notations and definitions. Assume there are N images $\{I_1, I_2, ..., I_N\}$ in the collection, and the words in the dictionary are $\{w_1, w_2, ..., w_M\}$. The shape feature of I_n is represented by r_n and the noisy labels of image I_n are represented by $\{w_1, w_2, ..., w_T\}$. $\delta(w_m, I_n) = 1$ if $w_m \in \{w_1, w_2, ..., w_T\}$; and 0, otherwise.

Let z denote a latent variable to represent an object topic, with discrete values $z = 1, ..., K$, and s denote a latent variable to represent a shape topic, with discrete values $s = 1, ..., N_s$. We abbreviate "sketch" and "keywords" to "ske" and "key" for long equations. Given N, M, K and N_s, the generative process of QST model is given as follow:

1. For each image I_n, sample the object topic: $z \sim p(z|I_n)$
2. For each object topic z:
(a) sample T words $\{w_1, w_2, ..., w_T\}$, in which for each word w_m we have:
$w_m \sim p(w_m|z, \beta, keywords) = \beta_{z,w_m}^{\delta(w_m, keywords)}$
(b) sample the shape topic:
$s \sim p(s|z, \theta) = \theta_{z,s}$
3. For each shape topic s, sample the shape feature:

$$r_n \sim p(r_n|s, \mu, \sigma, \beta, sketch) = \frac{1}{\sqrt{2\pi\sigma_s^2}} \exp(-\frac{dist(r_n, \mu_s)^2}{2\sigma_s^2} dist(r_n|sketch)),$$

in which $dist(r_n, \mu_s)$ is defined as the distance between r_n and μ_s, and $dist(r_n|sketch)$ is the distance between r_n and the sketch. Both of $dist(r_n, \mu_s)$ and $dist(r_n|sketch)$ are obtained according to [7]. In particular, we define $\delta(w, \emptyset) = 1$.

Given the parameters θ, μ, σ, and β, we can get the fol-

lowing joint probability of a set of N object topics z, a set of N shape class s, and a set of N shape feature $\mathbf{r}$ and words $\mathbf{w}$:

$$p(\mathbf{I}, \mathbf{w}, \mathbf{r}, z, s|\theta, \mu, \sigma, \beta, sketch, keywords)$$
$$= \prod_{n=1}^{N} \{p(I_n)p(z|I_n)p(r_n|s, \mu, \sigma, \beta, sketch)p(s|z, \theta)$$
$$\prod_{m=1}^{M}(p(w_m|z, \beta, keywords))^{\delta(I_n, w_m)}\}.$$

3.4 Parameter Estimation

We estimate the parameters by maximizing the log-likelihood function using the EM algorithm. The log-likelihood L could be written by:

$$L = F + \sum_{n=1}^{N} KL(q(z, s)||p(z, s|I_n, \mathbf{w}, r_n, \theta, \mu, \sigma, \beta, ske, key)),$$

in which the lower bound F of the log-likelihood is:

$$F = \sum_{n=1}^{N} \sum_{z} \sum_{s} q(z, s) \log \frac{p(I_n, \mathbf{w}, r_n, z, s|\theta, \mu, \sigma, \beta, ske, key)}{q(z, s)}.$$

We use the EM algorithm to iteratively maximize the lower bound F and minimize the KL divergence. Thus, the E-step is to calculate:

$$q(z, s|I = I_n) = \frac{p(z, s, I = I_n, \mathbf{w}, r_n|\theta, \mu, \sigma, \beta, ske, key)}{p(I = I_n, \mathbf{w}, r_n|\theta, \mu, \sigma, \beta, ske, key)}.$$

With some deductions which will be omitted due to space limitation, we can get the estimate of latent variables:

$$\begin{aligned} q(z, s|I = I_n) \quad &\propto \quad p(I = I_n)p(z|I = I_n) \\ &\times \quad p(s|z, \theta)p(r_n|s, \mu, \sigma, \beta, sketch) \\ &\prod_{m=1}^{M} \quad (p(w_m|z, \beta, keywords))^{\delta(I_n, w_m)}. \end{aligned}$$

Since we can obtain the joint distribution of s and z in condition of observed variables and parameters, we do not need variational approach to separate the joint probability of s and z.

By maximizing the lower bound F, in the M-step we can get:

$$\begin{aligned} p(z|I_n) \quad &\propto \quad \sum_{s} q(z, s|I_n), \\ \theta_{z,s} = p(s|z, \theta) \quad &\propto \quad \sum_{n=1}^{N} q(z, s|I_n), \\ \beta_{z,m} = p(w_m|z, \beta) \quad &\propto \quad \sum_{n=1}^{N} \sum_{s} \delta(I_n, w_m) q(z, s|I_n). \end{aligned}$$

Since it is not easy to find a vectorized expression of the shape feature [7], the expectation of Gaussian is represented by the centremost sample of the distribution:

$$\mu_s = \arg\max_{\mu} \sum_{n=1}^{N} \sum_{z} q(z, s|I_n) dist(r_n|ske)(-\frac{dist(r_n, \mu)^2}{2\sigma_s^2}).$$

The estimated variance is

$$\sigma_s = \sqrt{\frac{\sum_{n=1}^{N} \sum_{z=1}^{K} q(z, s|I_n) dist(r_n|sketch) dist(r_n, \mu_s)^2}{\sum_{n=1}^{N} \sum_{z=1}^{K} q(z, s|I_n) dist(r_n|sketch)}}.$$

We iteratively conduct E-step and M-step, until the parameters become convergent.

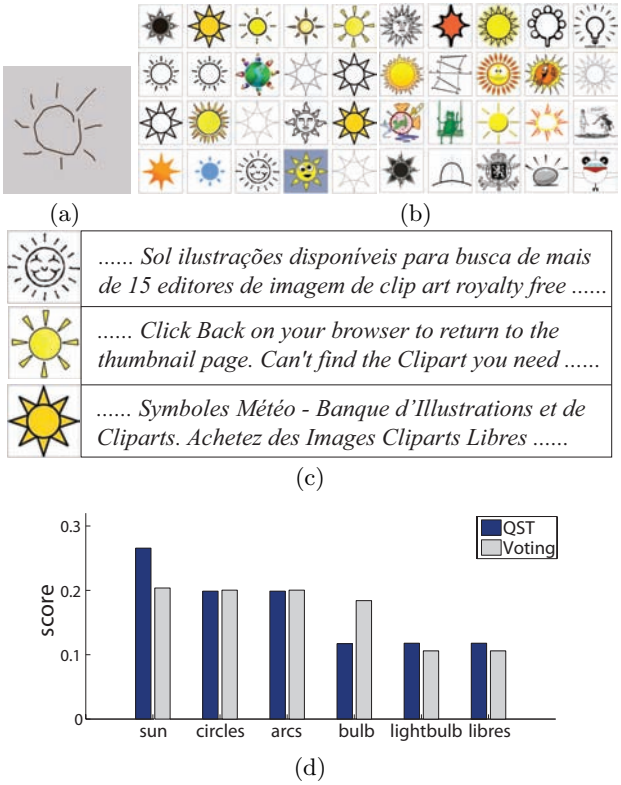

Figure 7: Illustration of the effectiveness of latent topics. (a) A hand-drawn sketch of the 'sun'. (b) Top results of a sketch-based image search engine. (c) Object-irrelevant labels of some images. (d) Comparison of the scores, i.e. $p(w|sketch)$, of the recommended tags by QST and a voting-based approach.

3.5 Utilizing the Model

Based on the probability of latent topics and the word distribution under each object topic, the probability of each word given the sketch can be used for sketch recognition and tagging:

$$p(w = w_m|sketch) = \sum_{n=1}^{N} p(I_n) \sum_{z} p(w = w_m|z, \beta)p(z|I_n).$$

The recommended tags are ranked according to their probabilities, and the top ones will be considered as the recognition/tagging results.

3.6 Effectiveness of Latent Topics

In this section, we illustrate the effectiveness of the QST model in object and shape topic mining.

3.6.1 Object Topic Mining

The QST model has the ability to discover the relationship between visual features and textual features, and thus fully utilizes the images without object-relevant labels.

For example, if we draw a sketch of the sun, in the query-adaptive image collection, there might be some images without any object-relevant labels, as shown in Fig. 7. In this case, a voting-based method (see Section 4.2), which only takes account of the word frequencies in this collection re-

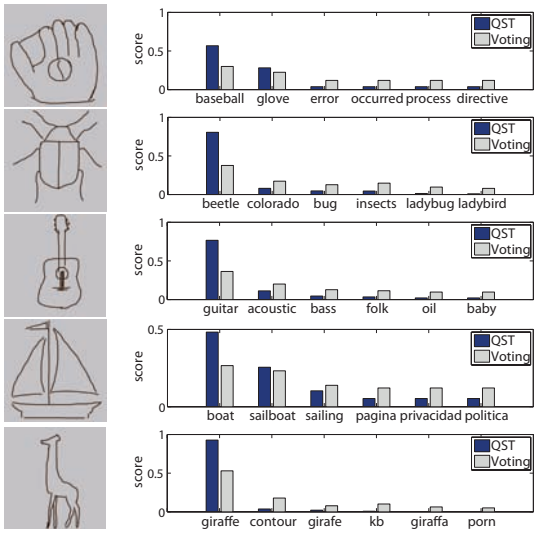

Figure 8: Example recognition results. For each sketch, the top six tags recommended by the two approaches together with their scores are shown.

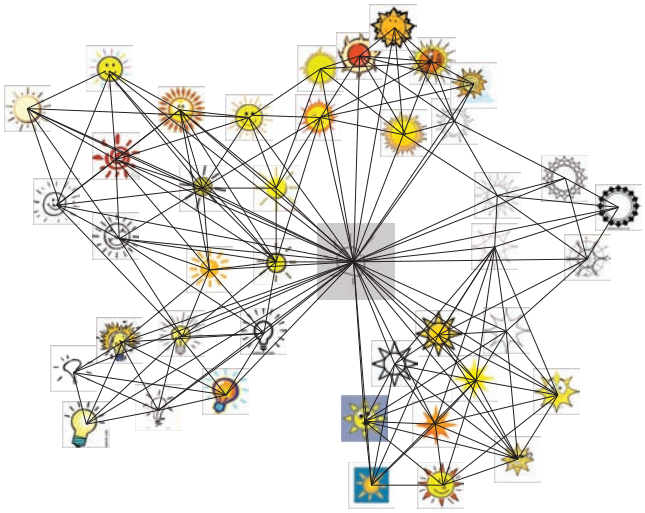

Figure 9: The images projected to a 2D space. The distance between two images represents the dissimilarity of their shape features. If the dissimilarity between two shapes is larger than a certain threshold, the line between these two images is not shown.

gardless of visual features, could not tell from the top three tags, i.e. sun, circle, and arc, as shown in Fig. 7 (d). However, in the QST model, for a 'sun' image without a useful label, the model can still recognize the image as the 'sun', for the probability $p(z = sun|I)$ will be relatively high. Thus, the final probability $p(w = 'sun'|sketch)$ will be large enough to distinguish from other tags.

To further see the advantages of QST, we show more comparisons in Fig. 8. We can see that, in some cases, although these two algorithms might recommend the same tags which will result in the same recognition results, the QST model has better ability to distinguish the right tags from noisy ones.

3.6.2 *Shape Topic Mining*

To illustrate the effectiveness of QST in shape topic mining, we project the images related to a drawing of the sun to a 2D space, preserving the relative shape dissimilarity between images, as shown in Fig. 9. We can see that, the shapes of the 'sun' locate at different clusters, and some of the 'sun' images are even closer to other objects.

We can consider the process of modeling data as a grouping process. If we use only one layer of object topics, the images of one object might not be merged together to one group but several groups, due to the variance of their shape features.

When we use the two-layer latent topics in QST to model the images, as shown in Fig. 5, they will be organized in a hierarchical way, in which different objects will be clustered into different groups, and each object group corresponds to multiple shape topics.

4. EXPERIMENTS

In this section, we evaluate the proposed QST model for sketch recognition.

Two data sets, i.e. the MPEG-7 20-category shape data set and the manually collected 500-category sketch data set (called Sketch-500 in this work), are used as testing sets to test the sketch recognition system.

4.1 Global Parameters of QST Model

In this part, we introduce the preprocess of the topic model to estimate some global parameters, i.e. the number of images in the collection N, the candidate word list $\{w_1, w_2, ..., w_M\}$, the number of object topics K, and the number of shape topics N_s. Both N and M are empirically set to be 30. The most frequent words in the collection are used as the candidate word list. Given the shape similarities between images[2], we use the graph-cut algorithm [9] to group the images, and use N_s to denote the number of groups. The number of object topics K is obtained in a similar way, excepted that the surrounding text in the vector space is used to replace the shape feature.

4.2 Voting-based Approach

As far as we know, this is the first work for general hand-drawn sketch recognition. Thus, we implemented a voting-based approach as the baseline for comparison. After getting the query-adaptive image collection for the sketch query, it is natural to select the most frequent word(s) as the recognition result(s)[3]. Considering both the image similarity to the sketch and the word frequency, the score of a word w, $Score(w|sketch)$, is calculated as follows:

$$Score(w|sketch) = \sum_{n=1}^{N} \#(w, I_n) \times Score(I_n|sketch),$$

where $\#(w, I_n)$ is the occurrence number of w in the surrounding text of image I_n, and $Score(I_n|sketch)$ represents the similarity between I_n and the sketch query as in [7].

[2]We use the two-way chamfer matching [7] to calculate the similarity between two images, which will not be introduced in this work due to space limitation.

[3]The stopwords were filtered out in advance.

Figure 10: Shape examples in MPEG-7. The names of shape classes from left to right are apple, bat, beetle, bell, bone (first line), bird, device8, device4, comma, device2, and brick (second line).

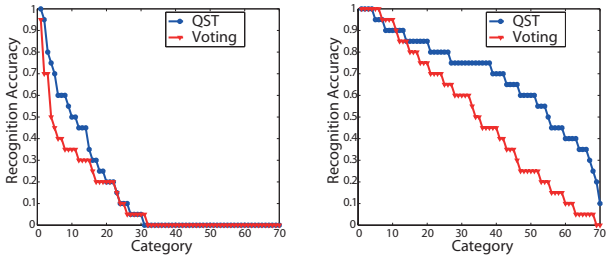

Figure 11: Shape recognition results on MPEG-7. The categories are ranked based on the performance. (a) Recognition results using unlimited tags. (b) Recognition results after filtering out the tags which do not belong to the names of the 70 classes.

Then, the word with the highest score is used as the recognition result for the sketch.

4.3 Shape Recognition on MPEG-7

MPEG-7 data set [6] defines 70 shape classes, where each shape class contains 20 different shape contours in the form of binary images. In total, this data set contains 1400 shape contours. A subset of the shape classes are illustrated in Fig. 10.

Different from existing shape classification work such as [19, 28], which were well trained in the 1,400 shapes, in this experiment, we try to use our system to recognize the shapes in MPEG-7 without any class-specific training. That is, the database, the vocabulary, and the evaluation are not constrained to the 1,400 shapes and 70 labels, which makes the recognition problem much more challenging.

For each category, we measure the proportion of shapes correctly recognized by the recommended tag with highest $Score(w = w_m|sketch)$, in which correct recognition means that the tag is the same as the class label or its synonymy. Recognition results for each category are shown in Fig. 11 (a), in which the categories are ranked based on the performance. We can see that, the performance of QST model is much better than that of the voting-based method.

Moreover, as shown in Fig. 11 (a), for some shape classes, such as heart, cellular phone, and truck, the performances are quite good; while for some other classes, such as glas, device0, and comma, the performances are really bad, most of which are zero. The reasons are three folds. First, the task of shape classification is quite different from sketch recognition. That is, the shapes in one category might represent different objects, and in this case the category name is not

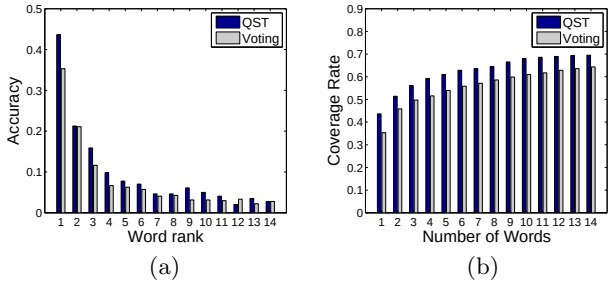

Figure 12: Recognition performance on Sketch-500. (a) Percentages of sketches correctly recognized by the kth recommended word. (b) Percentages of sketches correctly recognized by at least one word among the top k words.

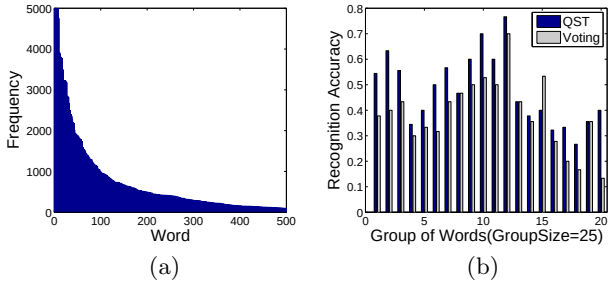

Figure 13: Relationship between the recognition performance and keyword frequency in the database. (a) Frequency of the 500 nouns in Sketch-500. (b) Average recognition results for each 25 sketches ordered by the frequency of the nouns.

meaningful at all. Examples are the categories of device0 to device8. Second, some shapes are not very common in clipart images, such as the shape "comma" as shown in Fig. 10, and thus cannot be recognized by our system. Third, most importantly, the unlimited tags and training data make the problem quite difficult. For some shapes, some noisy tags might be ranked higher than the groundtruth tag, and thus the recognition performance is not good. If we remove all recommended tags which do not belong to the 70 labels, the performance will become much better, as shown in Fig. 11 (b). We can also imagine that there will be another performance increase if we also constrain the database from one million to the 1,400 shapes as in [19, 28], which is however not the target of this system. The performance difference of Fig. 11 (a) and (b) also shows the difference of the sketch recognition task and existing shape classification work to some extent.

4.4 Sketch Recognition on Sketch-500

Since there is not an open data set for the task of recognizing an arbitrary but semantically meaningful hand-drawn sketch, we manually build the Sketch-500 data set to test the performance of the proposed recognition system. The Sketch-500 data set is built based on a list of 1000 non-

abstract nouns[4], which is a free word list collected for elementary students and contains most of commonly used nouns. For each word, we asked a graduate student to manually draw a sketch based on the top results returned from a commercial keyword-based clipart image search engine. We ignored the words which are hard to draw a sketch to represent, such as "back", "earthquake", and "dust", and finally collected 500 sketches together with groundtruth names. Example sketches are shown in Fig. 1.

The performance of sketch recognition is reported in Fig. 12 from two aspects. (a) shows the accuracy, i.e. the percentage of sketches correctly recognized by the kth word recommended by the system, and (b) is the coverage rate, i.e. the percentage of sketches correctly recognized by at least one word among the top k words. We can see that, only based on the top-one tag recommended by the system, the recognition rate is about 45%, which is a quite impressive performance, considering that the system is an open system without any preference to the testing data. We can also see that, compared with the voting-based algorithm, the QST model performs higher recognition accuracy. For example, the accuracy on the top-one tag is almost 10 percent higher than that of the voting-based algorithm.

To discover the relationship between the system's recognition ability to an object and the amount of data of the object in the system, we grouped the sketches into different groups according to the occurrence numbers of their names in the database, and calculated the average recognition performance for each group, as shown in Fig. 13. We can see that, the recognition system is somewhat robust to the frequency of occurrences of an object in the database.

Since there might be large intra-class variations in human drawings, to test this point, we randomly and independently moved all the strokes of a sketch to different directions. The moving distance is a parameter multiplied by the longer side of the sketch, and this parameter is called "variance" in this work. Fig. 14 shows the recognition results under different variances. We can see that, for a relatively large variance, e.g. the fish with variance 20%, the recognition results are still acceptable.

Besides recognizing sketches, our system can also suggest semantically related keywords to a sketch. A graduate student was asked to manually label whether a recommended tag is related to the groundtruth keyword. For example, the keyword "fruit" is a good tag for groundtruth keyword "apple", which however was considered as a wrong recognition result in previous experiments. The performance of the sketch tagging task is shown in Fig. 15. We can see that, there is about a 20% performance increase if we relax the recognition criteria from exactly matching the objet name or its synonymy to matching a semantically related keyword.

4.5 Other Applications

The learnt probabilities between variables of the QST model can be used in various applications, such as sketch-based image search and image tagging. In this section, we will illustrate the effectiveness of QST in the above two applications.

4.5.1 *Enhance Sketch-based Image Search*

[4]http://www.free-teacher-worksheets.com/support-files/list-of-nouns.pdf

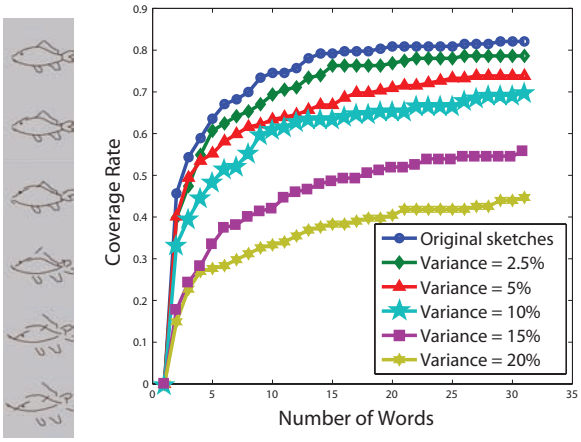

Figure 14: Recognition results under different variances. Left: an example with variances of 0%, 2.5%, 5%, 10%, 15%, 20%. Right: the coverage rates under different variances.

An important application of QST is to enhance the sketch-based image search. Based on the sketch recognition system, there are two ways to refine the search results.

The first way is to use the probabilities of the trained QST model, which is called "QST refined" here. The distance between the sketch and a database image could be measured in the topic space:

$$Dist(I_n, sketch) = KL(p(z|I = I_n)||p(z|sketch)),$$

in which

$$p(z = k|sketch) = \sum_{n=1}^{N} p(z = k|I_n)p(I_n).$$

The probabilities $p(z|I = I_n)$, $p(z = k|I_n)$, and $p(I_n)$ can be obtained from the QST model. Then, the search results are refined by sorting the images in the ascending order of $Dist(I_n, sketch)$.

The second method is to combine the recognition result with the sketch itself to query the clipart engine, which is named as "Sketch+Recognition".

In this experiment, we randomly selected 20 sketches from Sketch-500 data set as the queries, and evaluated the precision of top N search results returned from the clipart image search engine using different algorithms. For all methods in our comparison, the criteria of relevance is not only structurally (in terms of shape) but also semantically (in terms of concept) matched with the sketch query. The comparison results are shown in Fig. 17, from which we can see the effectiveness of the two methods in enhancing sketch-based image search. Some examples are shown in Fig. 18. We can see that some irrelevant results will be filtered out after understanding the sketch.

Besides, the recognition results based on the clipart image collection can also be used to improve the natural image search. As indicated by [7], sketch-based natural image search is more challenging than clipart search, since the cluttered background in natural images makes it not easy to extract the salient objects. Thus, the assumption that there is only one object in an image cannot be adopted, which causes that the search is translation and scaling sensitive

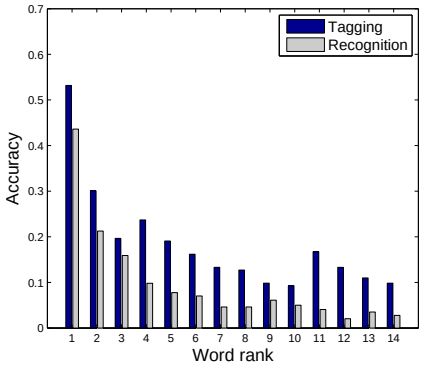

Figure 15: The performance of sketch tagging.

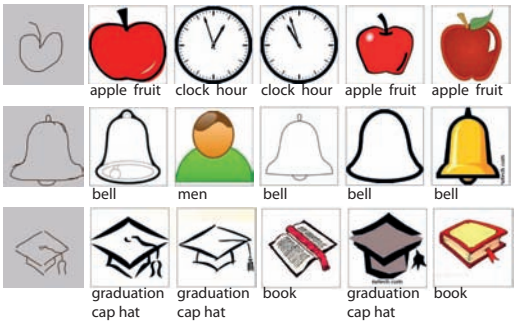

Figure 16: Examples of image tagging.

(to some extent). Hence, in many cases, even if the sketch query has a good shape, the search results might still be unacceptable. Fig. 18 also shows some examples of the search results on a 2-million natural image database. We can see that, by leveraging the recognition results to the sketch, the search results were greatly improved compared with the EIS method [7] (denoted by 'original search'). It should be noted that, the "QST refined" method cannot be used for natural image search, since the QST was learnt based on the clipart collection.

4.5.2 *Image Tagging*

As aforementioned, the collection of images we are using are not well labeled. Although each image has corresponding surrounding text, there may be many irrelevant and incorrect tags, as shown in Fig. 7 (c). Therefore, the QST model can be used to recommend tags to the images for further indexing and searching. The probability of a word given an image is used here:

$$p(w = w_m|I = I_n) = \sum_z p(w = w_m|z, \beta)p(z|I = I_n),$$

in which the probabilities on the right side are obtained by the QST model. The words with highest probabilities which are above a certain threshold will be used to tag the image. Some examples of image tagging are shown in Fig. 16.

5. CONCLUSIONS

In this work, we studied the problem of hand-drawn sketch recognition, and developed a practical system to recognize an arbitrary but semantically meaningful sketch. Based on the analysis to the difficulties in this unconstrained situation, the Query-adaptive Shape Topic (QST) model was proposed to model the recognition process and discover related object and shape topics. The learnt probabilistic relations in QST model could support various applications, such as *sketch recognition and tagging*, *image tagging*, and *sketch-based image search*. Extensive experiments have shown the effectiveness of the proposed model and recognition system.

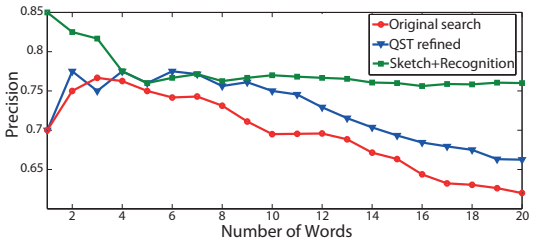

Figure 17: Performance comparison for sketch-based image search.

6. ACKNOWLEDGEMENTS

The work of Zhenbang Sun and Liqing Zhang was partially supported by the National Natural Science Foundation of China (Grant No. 90920014, 91120305) and NSFC-JSPS international exchange program (Grant No. 61111140019).

7. REFERENCES

[1] C. Alvarado and R. Davis. Sketchread: a multi-domain sketch recognition engine. In *SIGGRAPH*, 2007.

[2] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *PAMI*, 2002.

[3] D. Blei and M. Jordan. Modeling annotated data. In *SIGIR*, 2003.

[4] D. Blei and J. McAuliffe. Supervised topic models. *Arxiv preprint arXiv:1003.0783*, 2010.

[5] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *JMLR*, 2003.

[6] M. Bober. Mpeg-7 visual shape descriptors. *CSVT*, 2001.

[7] Y. Cao, C. Wang, L. Zhang, and L. Zhang. Edgel index for large-scale sketch-based image search. In *CVPR*, 2011.

[8] Y. Cao, H. Wang, C. Wang, Z. Li, L. Zhang, and L. Zhang. Mindfinder: Interactive sketch-based image search on millions of images. In *ACM Multimedia*, 2010.

[9] P. Felzenszwalb and D. Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 2004.

[10] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning object categories from google's image search. In *ICCV*, 2005.

[11] K. Forbus, J. Usher, and A. Lovett. Cogsketch: Open-domain sketch understanding for cognitive science research and for education. In *EUROGRAPHICS*, 2008.

[12] T. Hammond. Tahuti: A geometrical sketch recognition system for uml class diagrams. In *SIGGRAPH*, 2006.

[13] T. Hammond. et al. a sketch recognition interface that recognizes hundreds of shapes in course-of-action diagrams. In *CHI*, 2010.

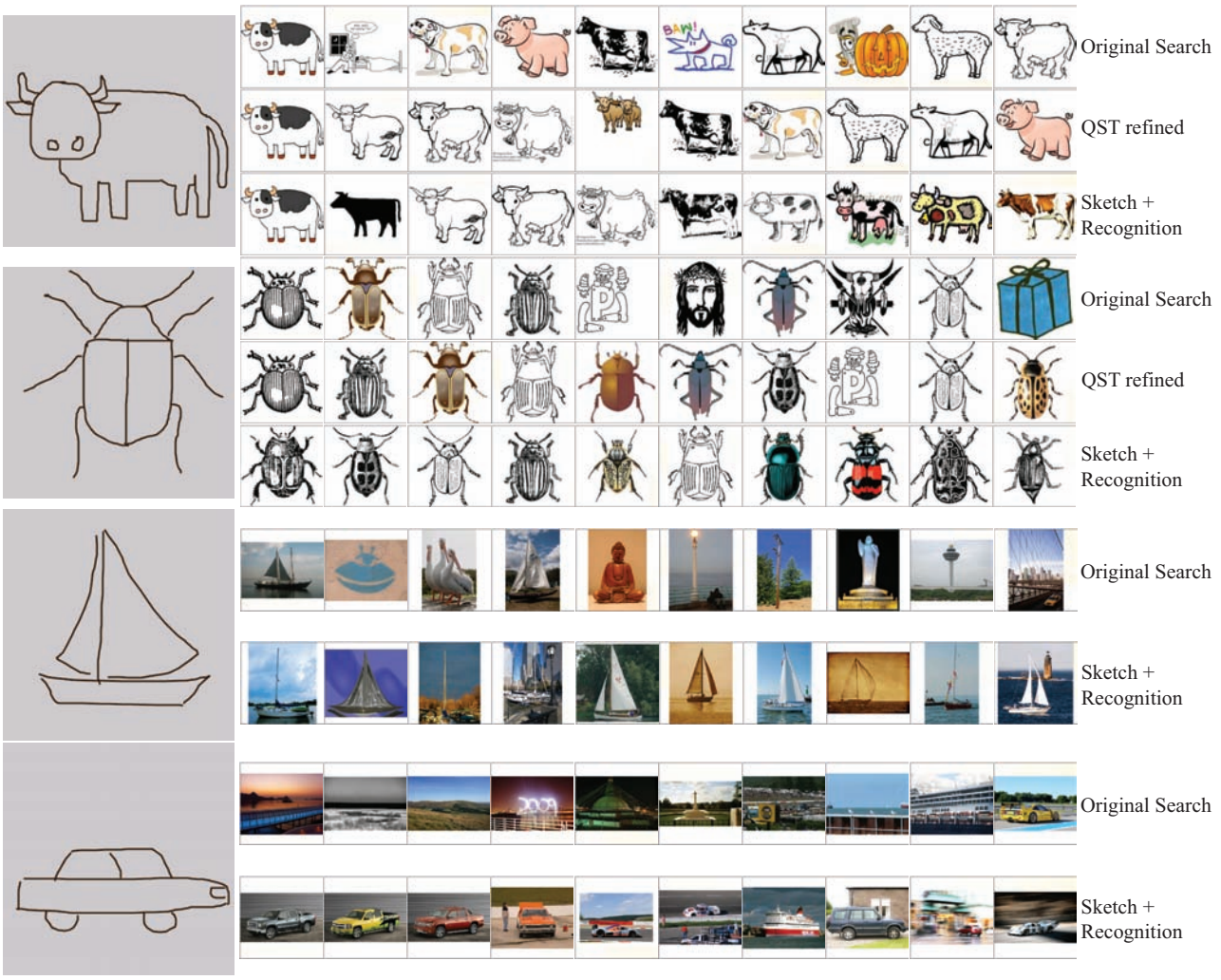

Figure 18: Example of enhanced sketch-based image search. Top two rows: top results of clipart image search. Bottom two rows: top results of natural image search.

[14] T. Hammond and R. Davis. Ladder, a sketching language for user interface developers. *Computers & Graphics*, 2005.

[15] Q. Hao, R. Cai, C. Wang, R. Xiao, J. Yang, Y. Pang, and L. Zhang. Equip tourists with knowledge mined from travelogues. In *WWW*, 2010.

[16] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, 1999.

[17] Y. Lee and K. Grauman. Shape discovery from unlabeled image collections. In *CVPR*, 2009.

[18] J. Lin, M. Newman, J. Hong, and J. Landay. Denim: finding a tighter fit between tools and practice for web site design. In *SIGCHI*, 2000.

[19] H. Ling and D. Jacobs. Shape classification using the inner-distance. *PAMI*, 2007.

[20] K. Mikolajczyk, A. Zisserman, C. Schmid, et al. Shape recognition with edge-based features. 2003.

[21] G. Mori, S. Belongie, and J. Malik. Efficient shape matching using shape contexts. *PAMI*, 2005.

[22] E. Paquet, M. Rioux, A. Murching, T. Naveen, and A. Tabatabai. Description of shape information for 2-d and 3-d objects. *Signal Processing: Image Communication*, 2000.

[23] B. Paulson and T. Hammond. Paleosketch: accurate primitive sketch recognition and beautification. In *IUI*, 2008.

[24] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *UAI*, 2004.

[25] T. Sebastian, P. Klein, and B. Kimia. Recognition of shapes by editing their shock graphs. *PAMI*, 2004.

[26] T. Sezgin and R. Davis. Hmm-based efficient sketch recognition. In *IUI*, 2005.

[27] T. Stahovich. Sketchit: a sketch interpretation tool for conceptual mechanism design. In *MIT AI Lab. TR*, 1996.

[28] A. Temlyakov and B. Munsell. Two perceptually motivated strategies for shape classification. In *CVPR*, 2010.

[29] C. Wang, J. Zhang, B. Yang, and L. Zhang. Sketch2cartoon: composing cartoon images by sketching. In *ACM Multimedia*, 2011.

[30] D. Zhang and G. Lu. A comparative study of fourier descriptors for shape representation and retrieval. In *ACCV*, 2002.