Local Configuration of SIFT-like Features by a Shape Context

Martin KLINKIGT† and Koichi KISE†

† Graduate School of Engineering, Osaka Prefecture University 1-1 Gakuen-cho, Naka, Sakai, Osaka, Japan
E-mail: †klinkigt@m.cs.osakafu-u.ac.jp, ††kise@cs.osakafu-u.ac.jp

Abstract The representation of information plays the key role in searching a document. While for text documents it is simple to extract keywords to find a document, for images this task is more challenging. In this paper we discuss different representations of images, point out their advantages and disadvantages and present our proposed method for stable and reliable information extraction to address the difficulties in object recognition.

Key words computer vision, object recognition, shape model

1. Introduction

In these years everyone is working with documents in digital form and gained expertise to organize and find them if needed. To find a specific document often some short keywords typed into some forms are enough to get the desired result. While for text one has no problems to extract and represent information, for images this task is not so easy. The use of the pixel information becomes meaningless if the resolution of the image changes. So researches consider to represent information in a more abstract way by using color information [1], contours [2] or scale-invariant feature transformation [3]. All these approaches have their advantages and disadvantages.

We increase the performance to find a certain image in a database without any additional hand work done by the user. We employ an approach of stable local features which can be extracted repeatedly even for different resolutions and under changing lighting conditions. For an increased performance we utilize a shape context to create a model. With our system we increase the performance of around 4% in terms of mean average precision even if the database is artificially increased.

2. Related Work

There is a high variety of online image search tools as for example Google Image Search [4] or TinEye [5]. These system have a large index of billions of images and perform still fast in less than 1 second. However, their use as a desktop system to search for the private image collection seems to be minor. The user can not define the images stored in the database, thus these systems are more for entertainment than for image search.

A system which is worth mentioning is provided by Jegou et al. [6]. This system handles up to some millions of images and can be used without any problems even on standard notebook computers. The results of this system are quite impressive. However, this system has one drawback which makes it unusable in some cases. In a simplified explanation, from the visual information of the whole image a compact representation of 20 bytes is calculated. If only a smaller part of the image is visible this compact representation will fail to return the correct result.

Such one fixed representation of the whole image at once is only useful in some specific contexts. By working on local features even partly visible images or objects can be recognized. Leibe et al. [7] and Fei-Fei et al. [8] used such local features. Based on these features they build a model how they are arranged. Although the user would have a benefit from these systems, much hand work must be done. The user has to provide a bounding box around the object of interest or several hundreds of images showing the object.

Our system also describes the visual information by using local features. In contrast to the systems of Leibe et al. and Fei-Fei et al. our system works even with only one image and does not require a bounding box around the object.

3. Image Representation

The representation of information from images is not as straight forwards as for text documents. This is due to the fact that image information has to challenge against obvious changes which do not touch essentially the information of them. The resolution for example can change easily. However, the "same" image showing an object once in lower resolution and once in higher resolution still shows the same object. The same holds if the image is rotated. Here it becomes clear that the raw pixel information can not be used to explain the content of an image in a reasonable way.

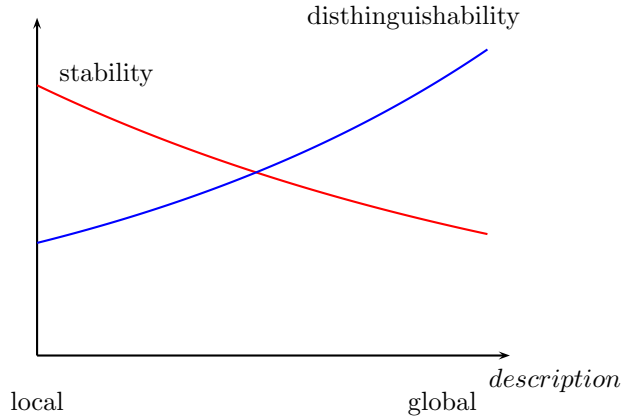


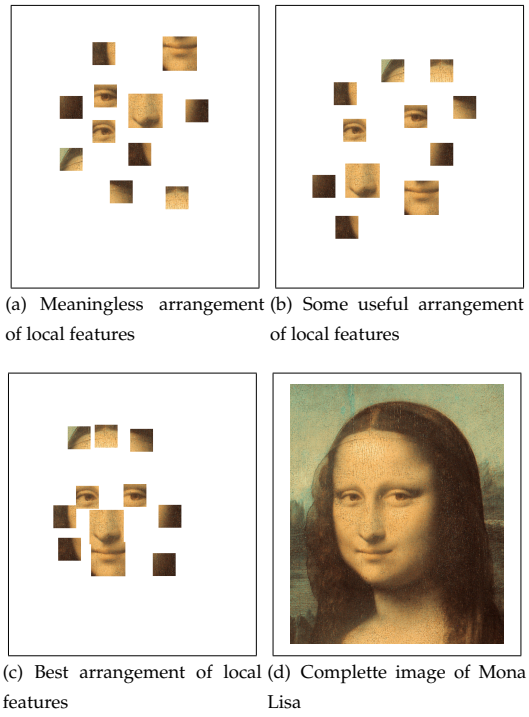
Figure 1 Influence of description to stability and distinguishability.

The question arise which representation should be used instead. One solution is to define the color information in a more abstract way like "In this region of the image is blue, in that black,..." The drawback of such a representation is that objects of similar color information would result in somehow the same description, e.g., a black cat and a black car in the center of the image.

Opelt et al. [2] have proposed a system which works on boundaries. Such an approach has many advantages, e.g., knowledge of similar objects like horse and cow can be combined and the remaining difference can be boosted to distinguish between them. The drawbacks are that up to now no practical representation of such boundaries is provided. Opelt et al. define their descriptor based on the location of pixel from edges. In such a case the object must have always an equal size in pixel in the training images and in the later query images. This is hard to achieve and limits the usefulness. Also such systems can tend to detect many different objects in cluttered regions like trees. Such misleading results are of no benefit for the user.

3.1 Local Features or Global Descriptors

In recent research scientists often use local features. With the Scale-Invariant Feature Transform (SIFT) proposed by Lowe [3] great results were achieved. In general for a small region of interest sophisticated mathematical functions are used to describe the region. Such a region can be seen as peephole by just looking on one small part of the object. One advantage of this local restricted approach is that even for occlusion or partly visible objects for the remaining visible parts still the same description is calculated. The disadvantage of such a peephole view is that its ability to distinguish between images and objects is lower as compared to approaches using more global information. Figure 1 illustrate this effect. If the description remains local the stability is high and the distinguishability is low. If the description becomes more and more global, the distinguishability increases with the side effect that



(a) Meaningless arrangement of local features (b) Some useful arrangement of local features (c) Best arrangement of local features (d) Complette image of Mona Lisa

Figure 2 Problem if approaches relying only in local features.

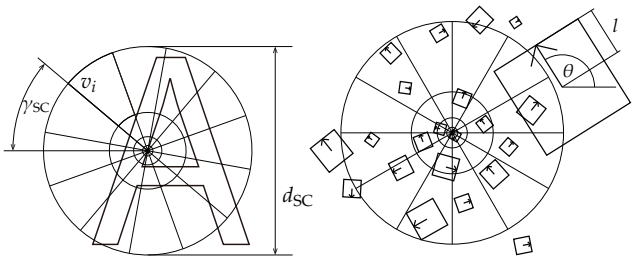
the stability in extraction of the features decreases.

Researches are often not aware of this relationship and design a system to use either local features or very global descriptions and ignore or by-pass the drawbacks.

3.2 Flexible Shape Model

Existing models like the implicit shape model proposed by Leibe et al. [7] define a global description with a help of a common reference point. In their system the authors learn a centroid from a large set of training images. In their system Leibe et al. require a bounding box around the object. This involves a high burden for the user who has to provide all these images together with the bounding box. In the constellation model as proposed by Fei-Fei et al. [8] the system would require only one image. Here a shape model is created from a few significant features of the object. The constellation model is a really strict model by creating a connected graph of the features. During recognition the main problem of this model is graph homomorphism which is NP-complete and, therefore, the number of used features was limited to less than 6. If one of these features can not be detected due to occlusion, the whole approach fails to recognize the object.

From Fig. 2 we can conclude which information is worth using. The idea is to get the stability of local features and the discriminative power of global shapes. Ideally the description scales within these two extrema. The left image (Fig. 2(a)) shows a collection of features, however, with a shuffled configuration. In Fig. 2(b) we have the same local features as before and additionally a better configuration of these features. Finally Fig. 2(c) has the best configuration when we



(a) Shape Context for characters and its properties. (b) Shape context created from SIFT features.

Figure 3 Shape context properties and its use for PCA-SIFT features.

compare with the full image in Fig. 2(d). We build such a model by working in reliable PCA-SIFT features [9] and calculate a shape context from these features. The necessary elements and our proposed method will be explained in the next section.

4. Shape Context for SIFT features

Instead of calculating one model, we keep the training simple and create our model during recognition. By doing so, it can be adapted on the fly to different conditions. It can be corrected to work at any point between local features and global description to achieve the best recognition performance.

First we extract local PCA-SIFT features from the training images and store them together with their position into a database. Beside this no additional calculation is performed.

During recognition we create our model with the help of the shape context. The shape context as proposed by Mori et al. [10] was designed to detect simple shapes like character. For real images one could use an edge image as Mortensen et al. in [11] have proposed. However, such a solution can not address the problem of cluttered regions as we described in Section 3. Figure 3(a) shows such a shape context and its important properties. Indicated are the sectors v_i of the shape context, its diameter d_{SC} and orientation γ_{SC}.

4.1 Utilization for PCA-SIFT features

We use the shape context to describe the local configuration around a PCA-SIFT feature. This is done by analysing in which certain section v_i nearby features are located. In Fig. 3(b) these features are indicated as squares. These features have two geometric properties which are a scale l and a orientation θ

The verification of the shape is done in the following steps which are also illustrated in Fig. 4:

Step 1: *(Feature Matching)* Compare the features stored in the database with features extracted from the query image based on the PCA-SIFT descriptor

Step 2: *(Feature Cleanup)* Remove features without corresponding matches for further processing

Step 3: *(Shape calculation)* Place the shape context over corresponding features of the database c_d and the query image c_q and adjust them concerning the properties θ and l of these features

Step 4: *(Final Scoring)* Ignore features outside the shape context and remove corresponding features located in different sections v_i of the shape context by counting them as fault and finally verify the local configuration of all matching features

Step 3 and 4 need more explanation which we give in Section 4.2 and 4.3.

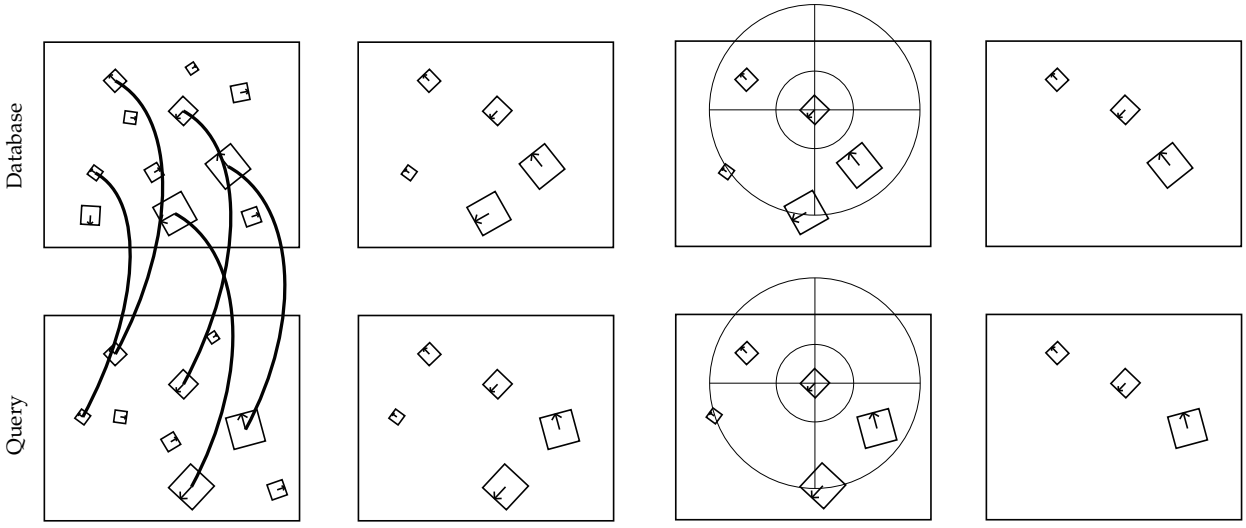
4.2 Verification of Location

As we mention before the shape context has a property d_{SC} which defines its scale and orientation γ_{SC} as shown in Fig.3(a). For a proper verification of the local configuration of the features, these values must be adapted to the current conditions. Namely we have to consider the case that the size and orientation of the object itself in the image are different between the training and query phase. If the shape context would always have a fixed size and orientation, the nearby features would be located in different sections of the shape context and a verification would become meaningless.

Let $(l_d, \theta_d, l_q, \theta_q)$ be a pair of matching features, where l_d is the scale of the feature from the database, θ_d its orientation and respectively l_q and θ_q for the feature from the query image. We create one shape context for the feature in the database and for the feature from the query image. As orientation γ_{SC} of these shape contexts we choose the orientation θ of the database feature and respectively of the query feature. The scale d_{SC} is calculated as $d_{SC} = a \cdot l$, where a is a fixed value and l the scale l_d of the feature c_d or l_q for feature c_q (Fig.3(b)). The value of a is set to, e.g., a size of 200 pixel. This will give us the required conditions for the verification, since our used PCA-SIFT features guaranty us to be adapted to the current size and orientation of the object in the image. In the example of Fig. 4 we have in Step 4 three features remaining with fitting location in the database and the query image.

4.3 Verification of Shape

After in Step 3 the location of the features is verified which corresponds to the problem indicated in Fig. 2, we analyse the configuration of the matching features itself in Step 4. We calculate in a similar manner as Jegou et al. in [12] the weak geometric consistency by taking the differences in the scale $\delta_l = l_d - l_q$ and orientation $\delta_\theta = \theta_d - \theta_q$ of the matching features $(l_d, \theta_d, l_q, \theta_q)$. These differences are quantized and stored as a histogram. The main idea of this approach is that local features are always transformed uniformly and not one part of the object is scaled up while other parts are scaled down. The same holds for its orientation.



Step 1: Feature matching **Step 2**: Feature cleanup **Step 3**: Shape Calculation **Step 4**: Final scoring

Figure 4 Shape context verification steps during recognition.

Let $g(\delta_{l_j})$ be the score in scale difference histogram bin δ_{l_j} and respectively $h(\delta_{\theta_j})$ the score in a bin of orientation difference histogram. The score s resulting from the matches is:

$$s = \min\left(\max_{\delta_{l_j}\in\delta_l} g(\delta_{l_j}), \max_{\delta_{\theta_j}\in\delta_\theta} h(\delta_{\theta_j})\right). \tag{1}$$

The sum of the scores s of all shape context is the score $w_{\mathbf{SC}}$ based on the shape context for a certain object.

4.4 Final Score

The shape context, as we constructed it, is a really strict shape model and may fail to recognize some objects. Therefore, we do not apply it alone and use a combination of a simple voting approach and our proposed shape model. For the simple voting local features are compared by taking only into account the PCA-SIFT descriptor part. A similarity beyond a certain threshold cast a vote for the corresponding object. The object which accumulates the most votes has the highest confidence to be shown in the image. For our example in Fig. 4 the object would have five votes for its five matching features.

Let $w_{\mathbf{Vote}}$ be this score for a certain object which we calculate in parallel at step 1 and 2. We analysed a combination of $0.55 \cdot w_{\mathbf{Vote}} + 0.45 \cdot w_{\mathbf{SC}}$ for the final score for an object to be well performing with $w_{\mathbf{SC}}$ as defined in section 4.3.

5. Evaluation

The intention of our evaluation is to find suited parameters of the shape context. These are the scale d_{SC} and the number of sections v_i. As the dataset for our evaluation we choose the Oxford buildings dataset from [13]. From this dataset we selected the 272 query images and as database images we searched in the Internet for *one* image for each of the 11 objects. Figure 5. is showing *all* these database images. All images were scaled down to a resolution of 640x480 pixels.

In the same manner as [14] we calculated the mean average precision (mAP). This value combines precision and recall by also taking into account the rank of the correct object in a ranked list of results. Let N be the number of retrieved results and N_{rD} the number of relevant results. Then the average precision P_{ave} is:

$$P_{\mathrm{ave}} = \frac{\sum_{r=1}^{N}(P(r)\times \mathrm{rel}(r))}{N_{rD}} \tag{2}$$

where r is the rank, rel() a binary function on the relevance of a given rank, and $P(r)$ precision at a given cut-off rank:

$$\mathrm{rel}(r) = \begin{cases} 1, & \text{if } r \text{ is relevant;} \\ 0, & \text{otherwise.} \end{cases}, \quad P(r) = \frac{\sum_{i=1}^{r}\mathrm{rel}(i)}{r} \tag{3}$$

By finally taking the mean over all queries we get the mean average precision. The better the rank is, the higher is the mAP. Since in our evaluation only one result is correct it means, if the correct object is at the first rank, we have a mAP of 1, at the second $\frac{1}{2}$, then $\frac{1}{3}$, $\frac{1}{4}$ and so on.

We also measured the performance of the different parameter settings for an increased database. Here we load distractor images from Flickr [15] which, if returned, are counted as the incorrect result. The results of the evaluations are shown in Table 1. We compared our method to the simple voting approach as explained in Section 4.4.

For the database only containing images of the objects we want to recognize, the best results were achieved with a shape context of a scale d_{SC} of 600 pixel and only two segments in form of half of a circle. For the experiment on the database which was increased with distractor images (second column) the best performing set of parameters are a scale d_{SC} of 400

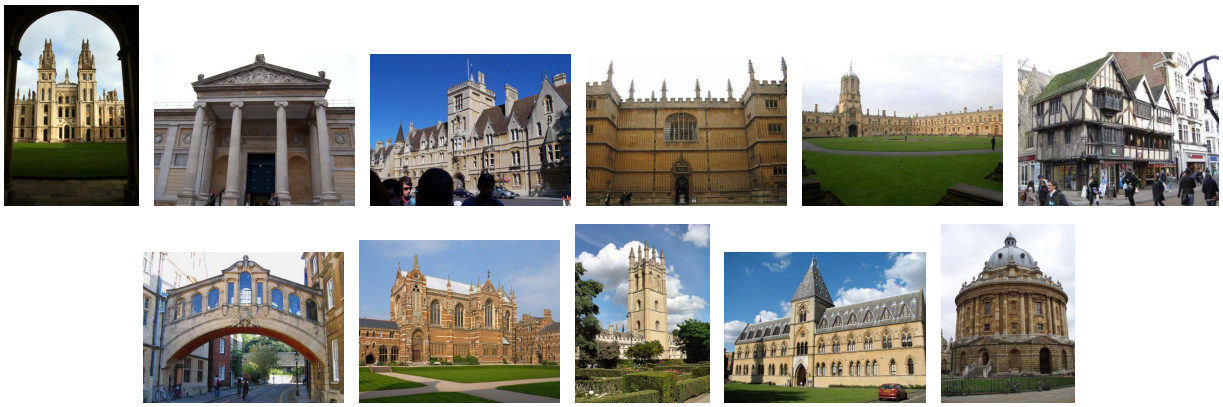


Figure 5 Database images of Oxford buildings.

Table 1 Results for evaluation on Oxford Buildings dataset. First row shows results of simple voting strategy. Below results of shape context approach over different setting of parameters. The parameters are: scale d_{SC} of shape context in pixel, number of segmentation in scale / number of segmentation in angle. First column database containing only Oxford buildings, second with additional distractor images.

	no distractor img.	distractor img.
Voting	47.24%	40.05%
Shape Context		
200px, 2/4	47.79%	42.48%
400px, 1/2	51.41%	42.86%
400px, 1/3	50.86%	42.75%
400px, 2/4	49.63%	43.11%
600px, 1/2	51.78%	42.88%
600px, 1/3	51.59%	42.66%
600px, 2/4	50.00%	43.09%

pixel with 2 sections in scale and 4 in angle.

These results show some tendency. While working under isolated conditions, the use of very global descriptors increases the performance. If the system has to challenge with a larger database this tendency is not so clear anymore. Here the reliability of local features becomes more important and, therefore, smaller shape context achieve good performance.

6. Conclusion

We have proposed a novel use of a shape context to verify the local configuration of SIFT like features. With this flexible shape model we improved the recognition performance in terms of mean average precision by 4% on the public available Oxford buildings database. The advantage of our proposed method is its flexibility to describe the content in the image. It can easy be adapted to the current conditions, working on small or large database by using local or more global descriptors.

Acknowledgment

This research was supported in part by the Grant-in-Aid for Scientific Research (B)(22300062) from Japan Society for the Promotion of Science (JSPS) and A-Step(AS2111193A) from Japan Science and Technology Agency (JST).

References

[1] D. Borth, J. Hees, M. Koch, A. Ulges, C. Schulze, T. Breuel, and R. Paredes, "Tubefiler - an automatic web video categorizer," ACM Mutimedia. ACM International Conference on Multimedia (ACM MM), October 19-24, Beijing, China, ed. by ACMACM, ACM 2009. http://demo.iupr.org/tubefiler

[2] A. Opelt, A. Pinz, and A. Zisserman, "A boundary-fragment-model for object detection," Proc. of ECCV, 2006.

[3] D.G. Lowe, "Object recognition from local scale-invariant features," Proc. of ICCV, p.1150, 1999.

[4] "Google image, former google similar images," `http://similar-images.googlelabs.com`, 2010.

[5] "Tineye," `http://www.tineye.com`, 2010.

[6] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," CVPR, jun 2010. http://lear.inrialpes.fr/pubs/2010/JDSP10

[7] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," Int. J. Comput. Vision, vol.77, no.1-3, pp.259–289, 2008.

[8] L. Fei-Fei, R. Fergus, and P. Perona, "Learning generative visual models from few training examples," Workshop on Generative-Model Based Vision, IEEE Proc. CVPR, 2004.

[9] Y.K. Rahul, Y. Ke, and R. Sukthankar, "Pca-sift: A more distinctive representation for local image descriptors," Proc. of IEEE CVPR, pp.506–513, 2004.

[10] G. Mori, S. Belongie, and J. Malik, "Efficient shape matching using shape contexts," IEEE PAMI, vol.27, no.11, pp.1832–1837, 2005.

[11] E.N. Mortensen, H. Deng, and L. Shapiro, "A sift descriptor with global context," CVPR, 2005.

[12] H. Jégou, M. Douze, and C. Schmid, "Hamming embedding and weak geometric consistency for large scale image search," Proc. of ECCV, pp.304–317, 2008.

[13] J. Philbin and A. Zisserman, "Oxford buildings dataset," `http://www.robots.ox.ac.uk/~vgg/data/oxbuildings/`, 2007.

[14] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," Proc. of CVPR, 2007.

[15] "Flickr," `http://www.flickr.com`, 2010.