Outdoor Image Retrieval with Object Sketch Using Automatically Generated Index

MUKUNOKI Masayuki

March 1999

Outdoor Image Retrieval with Object Sketch Using Automatically Generated Index

MUKUNOKI Masayuki 1995

Outdoor Image Retrieval with Object Sketch Using Automatically Generated Index

MUKUNOKI Masayuki

March 1999

Abstract

In this thesis, we describe the outdoor image retrieval method using automatically generated index.

At present, the ability of computers to recognize objects is not as good as that of human beings. But some applications do exist, which can make use of the current object recognition results. One of such applications is to use them as the index of images. In traditional image database, someone must set the keys to the images. It is boring routine work and takes a lot of time, particularly when there are many images in the database. Object recognition techniques will help to solve this problem.

We have conducted research on object recognition of outdoor scenes, and we introduce our object recognition method to image retrieval. The recognition rate of the pixel-based method is higher than that of region-based methods. In addition to that, we need not use signal-level segmentation in our method. As a consequence, we can assign object labels automatically. We applied our pixel-based method to generate the index images for image retrieval. The retrieval ratio of top 5 candidates was 87.9% when the goal image was referred. This result tells that we can use our pixel-based method for indexing images.

Then, we proposed a method to extract training data from images containing same kind of object. In our pixel-based object labeling method, we need a large number of training data which had to be given by hand. The setup of using "a large number of the images containing the same kind of object" is easily realizable by using image libraries each image of which has been indexed with the kind of object within the image as a key word. With our training-data auto-

extraction method, we can save the cost to give such data. We use the constraint that "all images contain same kind of object," into the processing in automatic extraction of training data and outline the procedure to extract the training data automatically. In our experiments, the correctness of the generated training data was 64.9% which was better than the expected result.

When using the automatically extracted training data in the pixel-based object labeling method, the error of training data leads to the errors in the recognition results. We stood the model of the recognition error and examined the characteristic which the recognition result should require for image retrieval. The analysis revealed that the randomness in the recognition error was preferable while the uniformity in the index images was not preferable. According to the analysis, we proposed two methods to cope with the errors in the index images, both of which increased the randomness and decreased the uniformity in the index images. Using our methods, even if some errors exist in the training data, the recognition result can still be used as the index of image in the image library.

At last, we show the method to improve the retrieval ratio using the retrieval examples. The object recognition is not unique method to generate the index of image. When a retrieval user draws an object sketch and gets the goal image, the pair of the object sketch and the goal image is given to the retrieval system. The pair is called the retrieval example. The retrieval example contains the information that "when the object sketch is given, the goal image should be shown." This information is also useful for retrieval. We proposed two method to use the retrieval examples. One was a method to add the obtained retrieval examples into the index image. The other was a method to perform re-training using the training data extracted from the retrieval examples. Our experimental results told that both method contributed to improve the retrieval results.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In recent years, the capacity of computers are dramatically improved and we can treat a large number of digital images on the computers. Once the images are stored on computers, a method to access these images is required. In one case, the access method is the filename of the image file and in another case it is the image retrieval method.

In image retrieval, a retrieval user expresses his demands as a key and gives it to the retrieval system. The retrieval system searches the images which match the key and shows them to the user. When the key is the keyword and the keywords are given to each of the images beforehand, the same technology as document retrieval can be used here. But for image retrieval, the contents of the images may be more important in retrieving. Thus, the technology of "Query by image contents" has been developed [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. In these method, the feature values are used for the keys but in some cases, a user wants to retrieve an image with object names in the image. For example, a user may want to retrieval an image which contains a mountain and a lake. To enable such retrieval, the information whether the image contains a mountain and a lake is given as the index beforehand. To generate the index automatically, we use the image understanding by the computer.

There have been some research on object labeling of outdoor scenes[11, 12, 13, 14, 15, 16, 17]. They basically employ the following strategies: Generating initial regions with signal-level segmentation and assigning the object labels to the regions. There are many researches on segmentation of images [18, 19, 20, 21]. But it is difficult to generate the regions corresponding to objects for outdoor scenes because there are various objects in them and the conditions in which the image was taken cannot be controlled. It prevents the automatic object labeling. This is a problem.

Another problem to use the image understanding by the computer to generate the index is that the ability of computers to recognize objects is not as good as that of human beings at present. But the recognition result has no measure to evaluate the efficiency in itself. In most object recognition researches, the recognition result is evaluated only by the recognition rate, which indicates how much correctly the method can answer the object name in the images. It is sure that higher recognition rate is more preferable than lower recognition rate, but it is not clear that "80% of recognition rate is sufficient" or "99% of recognition rate is much better than 95% of that." It is because they do not assume any application. We use the object recognition result in image retrieval and evaluate our method. Furthermore, we improve our recognition method to fit the image retrieval application.

When we build an object recognition system, we must give some knowledge about the object. In traditional image understanding system, complicated rule base is used to object recognition. In such cases, it is difficult to give the knowledge about the object. The correctness of recognition result is mostly determined by the given knowledge, i.e. the man's skill to give and express the knowledge about the object. We usually do not realize "why this region is SKY," and it is a very hard task to give good knowledge, especially in the case that we use the knowledge implicitly.

Another method to give the knowledge about the object is to give some sample data to the recognition system and the system learn the knowledge about the object from the sample data (training data). In this case, we can easily give the sample data comparing with the rule-base method because we can indicate "this point is included by SKY region" even if we cannot tell why the point is SKY. But in general, we must give a large number of training data to the system to enhance the recognition rate. It takes much cost and a method automatically extracting training data is necessary. We propose such method. The automatically extracted data contains some error. We must cope with such error as well.

The object recognition is not unique method to generate the index of image. When a retrieval user draws an object sketch and gets the goal image, the set of the object sketch and the goal image is given to the retrieval system. The set is called the retrieval example. The retrieval example contains the information that "when the object sketch is given, the goal image should be shown." The retrieval examples can be regarded as the "case" in the case-based reasoning[37]. This information is also useful for retrieval.

1.2 Outline of the Thesis

In Chapter 2, we show our pixel-based object labeling method. This method need not generate signal-level segmentation and is easy to automate the object recognition. Then we apply the results of our labeling method as the index of images and use it in image retrieval system to show the efficiency of our method.

In Chapter 3, we show a method to extract training data automatically. We prepare images which contain same kind of object and extract a characteristic common to all images as the information of the object. We use the information as the training data of object recognition system and show the experimental results.

In Chapter 4, we analyze the characteristic of our object recognition method and our image retrieval method. Then we show the method to cope with the error in the automatically generated index images.

In Chapter 5, we show two methods to use the retrieval examples to improve the retrieval results. One is a method to add the obtained retrieval examples into the index image. The other is a method to perform re-training using the training data extracted from the retrieval examples. We implement these two methods and show the experimental results.

In Chapter 6, we conclude this thesis and describe some open problems and future works.

Chapter 2

Retrieval of Outdoor Scenes Using Pixel-based Object Models

2.1 Introduction

At present, the ability of computers to recognize objects is not as good as that of human beings. But some applications do exist, which can make use of the current object recognition results. One of such applications is to use them as the index of images for image retrieval. We have conducted research on object recognition of outdoor scenes, and in this chapter, we introduce our object recognition method to image retrieval.

In traditional image databases, someone must set the keys to the images. It is boring routine work and takes a lot of time, particularly when there are many images in the database. Object recognition techniques will help to solve this problem. The ultimate goal of object recognition in this case is to describe the objects in images automatically. Our "pixel-based object labeling method" for object recognition is suitable for this purpose.

There have been some research on object labeling of outdoor scenes[11, 12, 13, 14, 15]. They basically employ the strategies as follows: Generating initial regions with signal-level segmentation and then assigning the object labels to the regions. But it is difficult to generate the initial regions for outdoor scenes because there are various objects in them and the conditions for the camera and the light source cannot be fixed. It prevents the automatic object labeling. In our "pixel-based object labeling method," we do not generate initial regions, hence we can avoid this problem.

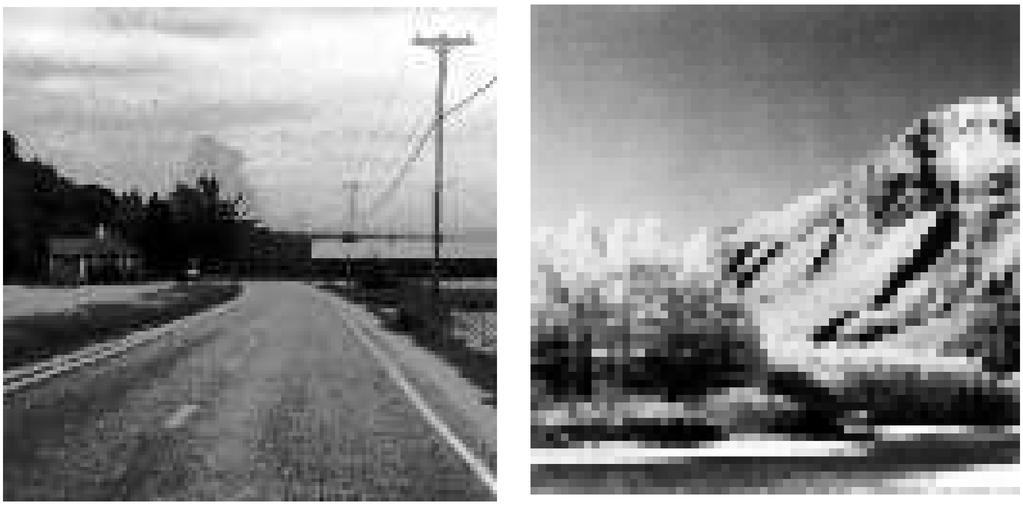
Next, we use the result of object labeling of outdoor scenes as the index of the images for image retrieval. We employ a pictorial query method for retrieval[22, 23, 24]. A users draws an object sketch as the condition for the retrieval. The retrieval system compares this object sketch with the index, and then presents images in the order of similarity. The indexes generated by object recognition do contain errors, but we can realize a fault tolerant retrieval system with the pictorial query method.

In the following sections, we explain our "pixel-based object labeling method" and its application to image retrieval.

2.2 Pixel-based Approach for Object Labeling

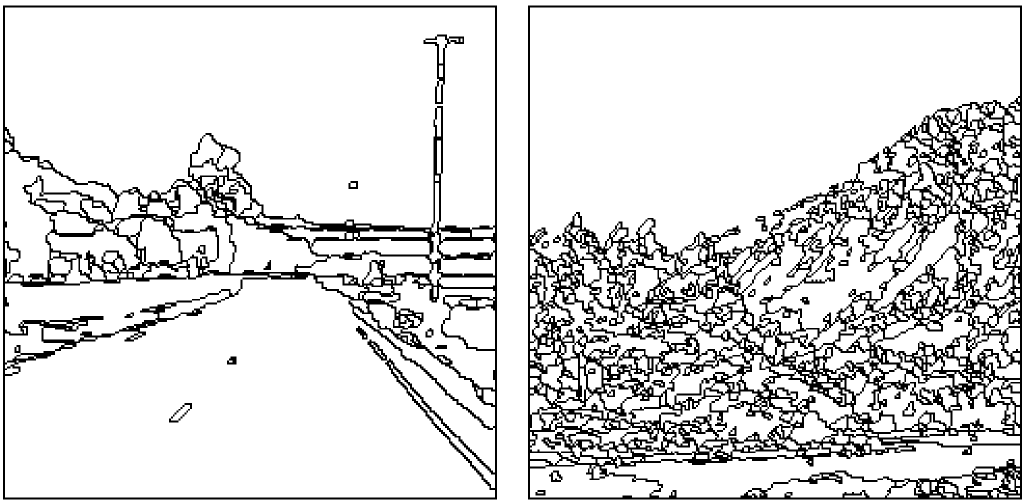
The goal of object labeling of an image is to segment it into regions each of which corresponds to one object and has a unique object label. Most of the studies so far adopted the "region-based" approach. They initially segment an image into regions and label them with object models. The object models are mappings between the features derived from the image and the object labels. However, in this strategy, there is a problem in generating the initial regions.

The initial regions are generated based on signal-level similarity of pixels: They are generated only from the statistical similarity or the spatial closeness of the pixels in an image. The knowledge about the objects is difficult to be introduced. Because of the variety of objects, generating the initial regions based on the signal-level similarity does not guarantee that the generated region actually



(a) image 1 (b) image 2

Figure 2.1: Examples of the outdoor scenes



(a) image 1 (b) image 2

Figure 2.2: Examples of signal-level segmentation

corresponds to an object.

The results of the initial segmentation have a great impact on the object labeling. But the results are largely affected by the parameters chosen for the signal-level processing. For example, when the outdoor scenes in Figure 2.1(a)(b) are processed with the same processing parameters, the results becomes Figure 2.2(a)(b). In Figure 2.2(a), a region almost corresponds to an object, while in Figure 2.2(b), regions are over-segmented and there are too many small regions each of which do not correspond to one object. It is difficult to select adequate processing parameters, particularly for outdoor scenes. In most studies, these parameters are heuristically determined for each image. Consequently, it prevents automatic labeling.

On the other hand, since objects in outdoor scenes are natural objects, they do not have any definite shape. The features of shape and size are not essential for the object labeling, even if we could get a region that corresponds to a whole object. Therefore, it is pointless to make the initial regions for object labeling.

Considering the discussion above, we have proposed a new approach to the problem of object labeling of outdoor scenes. It is called the "pixel-based object labeling method." It proceeds as follows (Figure 2.3):

1. Calculating feature values of every pixel.

2. Evaluating the object label of each pixel using the pixel-based object models, which are mappings between the features, derived from a pixel, and the object labels. The result is the evaluation value for each object label.

3. Assigning an object label to every pixel based on the evaluation values obtained in step 2.

Regions are generated by assembling the pixels that have the same object labels. Though pixels do not have the features of shape and size, they are not essential for the labeling of natural objects. Other features, such as color, texture and location, can be obtained from the pixels.

There are no heuristic thresholding parameters in the processing. The processing does not be interfered by such parameters and, as a consequent, automatic object labeling can be realized.

2.3 Construction of Pixel-based Object Models

2.3.1 Object Models with Neural-Network

We construct the object models with neural network. We use a 3-layered perceptron. There is one input node for each feature and one output node for each object label. Each feature value is normalized into [0.0,1.0] and used as input to the corresponding node in the input layer.

We use the back propagation algorithm for training, which minimizes the mean square error between the outputs of neural network and the teaching signals. The teaching signal is 1 for the correct output node, and 0 for the other output nodes. As the result of this training process, the network learns the mapping between the local features of a pixel and its object label.

In our approach, it is essential to construct object models by training. Although we cannot describe the object models clearly in the pixel-based approach, we can give samples of the correspondence between a pixel and its object label, and we can calculate the local features of the pixel. In this way, we can easily construct the object models. And there are other benefits of constructing the object models with neural network: We can get the result of the evaluation numerically; neural network can simulate any mapping between the features and the object labels and can show high performance after the training.

2.3.2 Objects

Table 2.1 shows the object labels we use. These objects frequently appear in outdoor scenes. Most of them are natural objects and they have neither definite

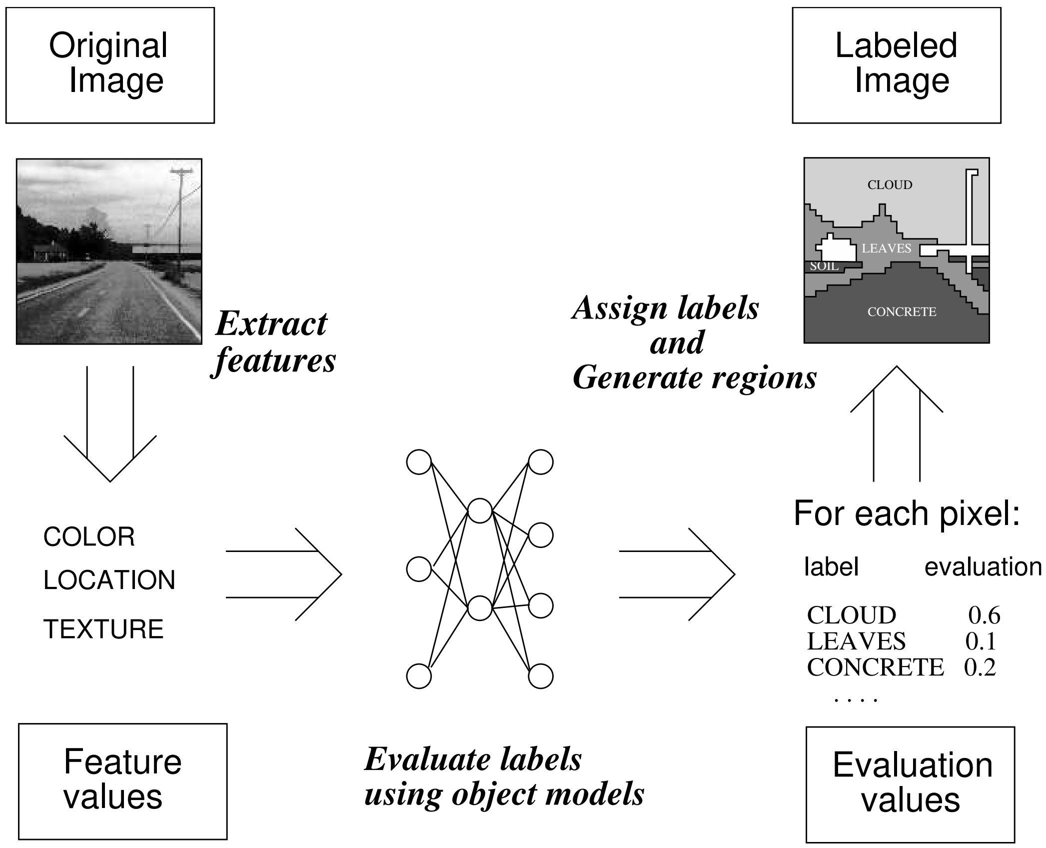


Figure 2.3: Pixel-based object labeling method

shape, size nor structure, but show notable features of color and texture in images. Therefore, these objects are suitable for labeling by our method. Note that other objects, such as cars, ships, buildings, trains etc., appear in outdoor scenes as well. We do not deal with these artificial objects, because the features of color or texture are not the key for the object labeling in this case. We should employ other method to the recognition of these artificial objects.

2.3.3 Features

Table 2.2 shows the features used for labeling the objects. Only COLOR, LOCATION and TEXTURE features are used in the "pixel-based method." SIZE and SHAPE features are used in comparative experiments in Section 2.3.4.

In the "pixel-based method," the COLOR and TEXTURE features are computed within 15$\times$15 neighboring pixels of each pixel. This suppresses noise in the image and enables us to calculate TEXTURE features for each pixel.

R, G and B indicate means of pixel values of red, green and blue values in the neighboring pixels, respectively. Original images are color images and each pixel has red, green and blue values.

Upper-R, Upper-G, Upper-B, Lower-R, Lower-G and Lower-B are R, G and B features at the position of 128 pixels above and below the pixel, respectively. If the position is beyond the size of the image, the features of the pixels at the uppermost or lower-most is used. We use these features because the color features are not stable, that is, easily affected by the camera condition or the light condition. For example, if we take pictures of a same place both in a fine day and a cloudy day, the colors of objects in them are different. The color features of other pixel may supplement this information. The positions (128 pixel above or below) of the pixels are decided experimentally.

X and Y indicate the location of the pixel in the image. The objects near the top of the image are likely SKY, CLOUD, MOUNTAIN etc.. The objects near the bottom of the image are likely CONCRETE, WATER, LEAVES etc.. The features of X and Y can contribute to the categorization of these objects.

Table 2.1: Object labels

label	object
SKY	blue sky
CLOUD	cloudy sky and cloud
CONCRETE	paved road surface
LEAVES	green leaves, grasses
SOIL	soil
SHADOW	dark shadow
MOUNTAIN	blue far mountains
WATER	water surface
DEAD-LEAVES	dead leaves
ROCK	rock

Table 2.2: Features used for labeling objects

	features
COLOR (3 points)	R, G, B Upper-R, Upper-G, Upper-B Lower-R, Lower-G, Lower-B
LOCATION	X, Y
TEXTURE (2-directions for each feature)	Energy, Entropy, Contrast, Correlation
SIZE (region only)	Size, Width, Height
SHAPE (region only)	Ith, Iratio, ISratio

TEXTURE features are calculated from a co-occurrence matrix [25]. The co-occurrence matrix $P(i, j|d, \theta)$ is a set of probabilities that two pixels, each of which has pixel-value i and j respectively, appear at distance d in direction θ within the neighborhood of the pixel. Energy, Entropy, Contrast and Correlation are calculated as follows:

$$\begin{aligned}
\mathrm{Energy}(d,\theta) &= \sum_{i,j} P(i,j|d,\theta)^2. \\
\mathrm{Entropy}(d,\theta) &= \sum_{i,j} P(i,j|d,\theta) \log \frac{1}{P(i,j|d,\theta)}. \\
\mathrm{Contrast}(d,\theta) &= \sum_{i,j} (i-j)^2 P(i,j|d,\theta). \\
\mathrm{Correlation}(d,\theta) &= \frac{\sum_{i,j} ijP(i,j|d,\theta) - \mu_x\mu_y}{\sigma_x\sigma_y}.
\end{aligned}$$

where

$$\begin{aligned}
\mu_x &= \sum_{i,j} iP(i,j|d,\theta), \\
\mu_y &= \sum_{i,j} jP(i,j|d,\theta), \\
\sigma_x^2 &= \sum_{i,j} (i-\mu_x)^2 P(i,j|d,\theta) \quad \text{and}, \\
\sigma_y^2 &= \sum_{i,j} (j-\mu_y)^2 P(i,j|d,\theta).
\end{aligned}$$

Energy, Entropy, Contrast and Correlation stand for the uniformity of the texture, the complexity of the texture, the contrast of the texture, and the slope of the texture, respectively. These features are frequently used to distinguish textures. We calculate these features with a distance $d = 1$ and with two directions $\theta = 0,\ 90°$. As a result, we get 8 feature values for TEXTURE.

SIZE and SHAPE features are used in the experiments in Section 2.3.4. They are the features for regions. SIZE is the number of pixels in a region. Width and Height are the horizontal and vertical length of a region, respectively. SHAPE features are calculated from the moments. Ith, Iratio and ISratio stand for the direction of a region, the slenderness of a region and the roundness of a region, respectively.

2.3.4 Index Generation Experiments

Comparison between pixel-based and Region-based method For experiments, we used 66 outdoor scenes. An example of the outdoor scenes is shown in Figure 2.1. We selected 975 sample points from 33 scenes for training and 970 sample points from the remaining 33 scenes for testing. The training was finished when the mean square error was less than 0.05. The number of nodes in the intermediate layer affects the results[26, 29]. We used 11 nodes, which was determined empirically. The recognition rate of trained neural network is shown in Table 2.3.

For comparison, we constructed two region-based object models. One of them uses COLOR, LOCATION and TEXTURE features, just like the pixel-based object models. The other uses SHAPE and SIZE features, in addition to the COLOR, LOCATION and TEXTURE features. In these cases, the feature values are calculated within the region in which the sample point is included. Regions of the images are generated by signal-level segmentation. Parameters of the segmentation are determined by hand to gain the most suitable segmentation result for each image. The recognition rates of the region-based object models are also shown in Table 2.3.

In Table 2.3, the two region-based object models show almost equal recognition rate. This result shows that the features of SHAPE and SIZE are not essential for object labeling. Pixels do not have these features but we can see that this is not a problem.

The recognition rate of the pixel-based object models is higher than those of the region-based ones, even when the SHAPE and SIZE features are added. This shows that the features derived from a pixel have as much information as features derived from a whole region for object labeling.

Sensitivity analysis To investigate the fact that the SHAPE and SIZE features are less important than other features, we conducted a sensitivity analysis

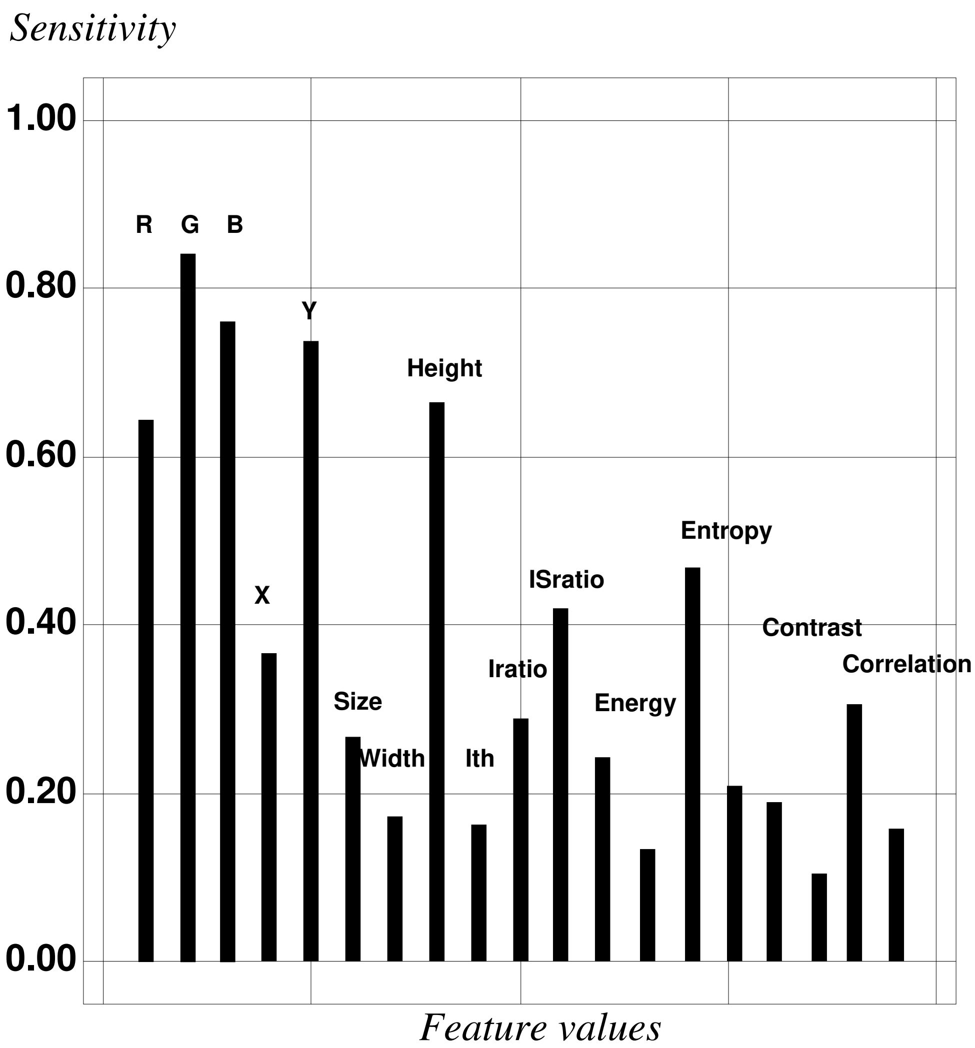


Figure 2.4: Sensitivity of neural network

of neural network. Sensitivity of neural network is defined as follows [27]:

$$\begin{aligned} S_{ik} &= \frac{\partial O_k}{\partial I_i} \\ &= O_k(1-O_k)\sum_j M_j(1-M_j)W_{jk}W_{ij} \end{aligned}$$

where

I_i	:output value of input layer node
M_j	:output value of intermediate layer node
O_k	:output value of output layer node
W_{jk}	:weight between intermediate and output layer
W_{ij}	:weight between input and intermediate layer

S_{ik} denotes the change ratio of k-th output value O_k to the change of i-th input value I_i. If S_{ik} is small for any input value I_i, then the input node does not contribute to the object recognition. In order to ascertain it, we need to give all pattern of input values to the neural network, but the combination of such input patterns is enormous. To its approximation, we give the training data to the neural network and analyze the results.

Figure 2.4 shows the result of sensitivity analysis for the neural network which has trained with the training data after 6000 loops of back propagation algorithm. The horizontal axis denotes each feature and the vertical axis denotes the maximum value of sensitivity S_{ik} for the training data. Note that even if the mean value of sensitivity is used in stead of the maximum value, the tendency is almost similar.

Figure 2.4 shows that the sensitivity of the colors (R, G, B), the vertical position feature (Y) and the vertical size feature (Height) is high. To the contrast, the sensitivity of horizontal size feature (Width), the incline of moment feature (Ith) and the texture energy and contrast features (Energy, Contrast) is low. Since the lower side of the image contains ground and higher side of the image contains sky in the objective images, the vertical feature is useful but the horizontal feature is not useful for object recognition. The reason why the sensitivity of the incline of moment feature is low is that the objects contained in the images have no definite shape and these objects appears in various size and direction in the images.

Feature elimination experiment Next, we conducted a feature elimination experiment. In this experiment, input values of some features in Table 2.2 are fixed to 0.0 and given to the neural network in training and evaluating stage. In the training stage, the back propagation loop was iterated until the mean square error between output of neural network and the training data became less than 0.05 or the loop count became 6000. With changing the initial weight of neural network, the training was conducted to 10 networks. Table 2.4 shows the number of converged (the mean square error became less than 0.05) network and the iteration count when the training was converged. Figure 2.5 and Table 2.5 shows that the best recognition rate within the recognition rates of the 10 trained neural networks. NON denotes the case that the all features in Table 2.2 are used. CL denotes the case that only COLOR and LOCATION features are used. In the COLOR and LOCATION cases, no network in the 10 trained neural networks was converged.

Figure 2.5 shows that even when SHAPE or TEXTURE features are eliminated, the recognition rate does not decline. This result reinforces the previous result(Figure 2.4). On the other hand, although Figure 2.4 suggests that the recognition rate would decline when SIZE features are eliminated, Figure 2.5 shows no remarkable difference. The following fact shows that the effective features for object recognition is COLOR and LOCATION.

- When COLOR and LOCATION features are eliminated, the training of neural network does not converge.
- In CL case, the recognition result is almost same as the NON case.

Since the COLOR and LOCATION features are effective for object recognition of outdoor scenes and they can be calculated from each pixel, we can conclude

Table 2.3: Recognition rate

	testing set (%)	training set (%)
pixel	77.3	94.4
region with SHAPE and SIZE	69.5	92.1
region without SHAPE and SIZE	68.2	91.5

Table 2.4: Eliminated feature and convergence of network

Eliminated feature	NON	COL	LOC	SIZ	SHP	TXT	CL
Number of converged network	6	0	1	6	5	8	0
Iteration	764	(6000)	1895	1450	1055	1508	(6000)

Table 2.5: Eliminated feature and recognition ratio

Eliminated feature	NON	COL	LOC	SIZ	SHP	TXT	CL
Recognition rate	76.4	47.0	63.7	73.5	75.4	80.3	79.4

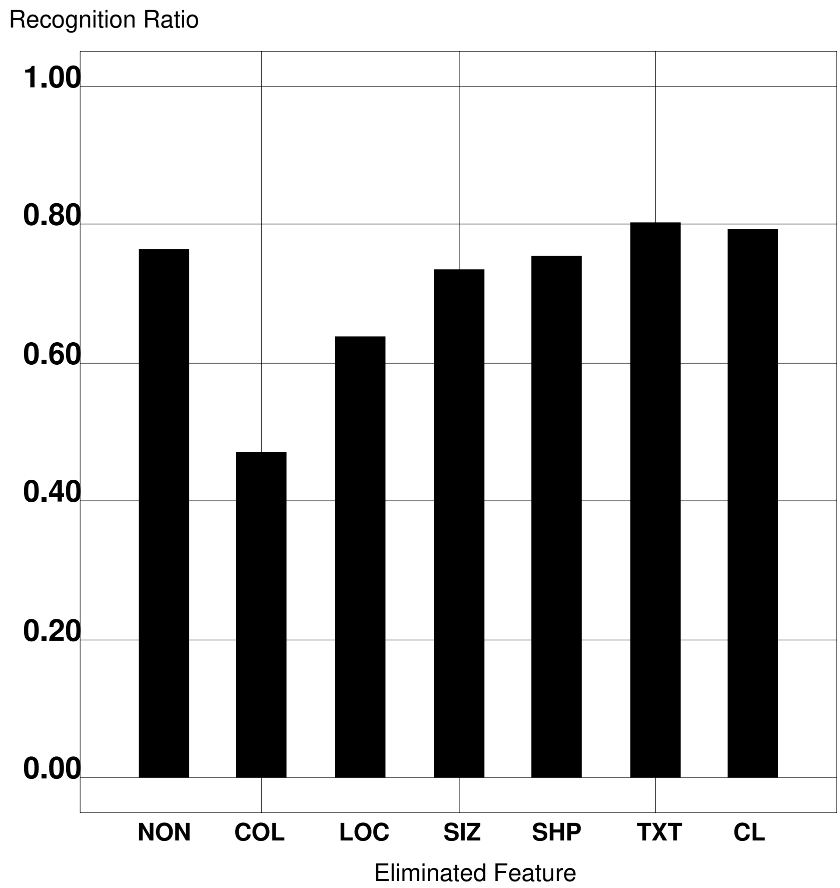


Figure 2.5: Eliminated feature and recognition ratio

that our pixel-based approach has as good facility as region-based approach in object recognition of outdoor scenes.

Recognition result of whole image At last, we apply these pixel-based object models to whole images. Figure 2.6 shows the result of object labeling of Figure 2.1(a). The output node which has the highest output value in the output layer is taken as the object label of the pixel. It is the simplest way to assign the object label to each pixel. There are some pixels which are labeled incorrectly. The object models are simple mappings between the feature values and the labels, and they do not consider the labels of other pixels. To improve the result, other knowledge, for example the spatial relation of objects, could be introduced to object labeling.

2.4 Retrieval with Object Sketch

2.4.1 Application of Our Method to Image Retrieval

We apply our object labeling method to image retrieval of outdoor scenes. General process of image retrieval is as follows: a user gives some keys; the retrieval system compare the keys with the indexes of images; and show some candidate images the indexes of that match the keys.

One useful kind of indexes for image retrieval is the contents of the image, i.e. the objects within the image. For example, if a user wants to retrieve some images which contain "mountain," it is natural to use "the-mountain-is-contained" as the index.

The result of our object labeling is the image assigned an object label to each pixel. It includes the information of the contents of the image and it can be used as the index. But, as shown in the previous section, the recognition rate of our method is not 100% and there are errors in the result. This means that some objects which do not exist in the image may appear in the result, and on the other hand, some objects which exist in the image may not appear in the result. If we

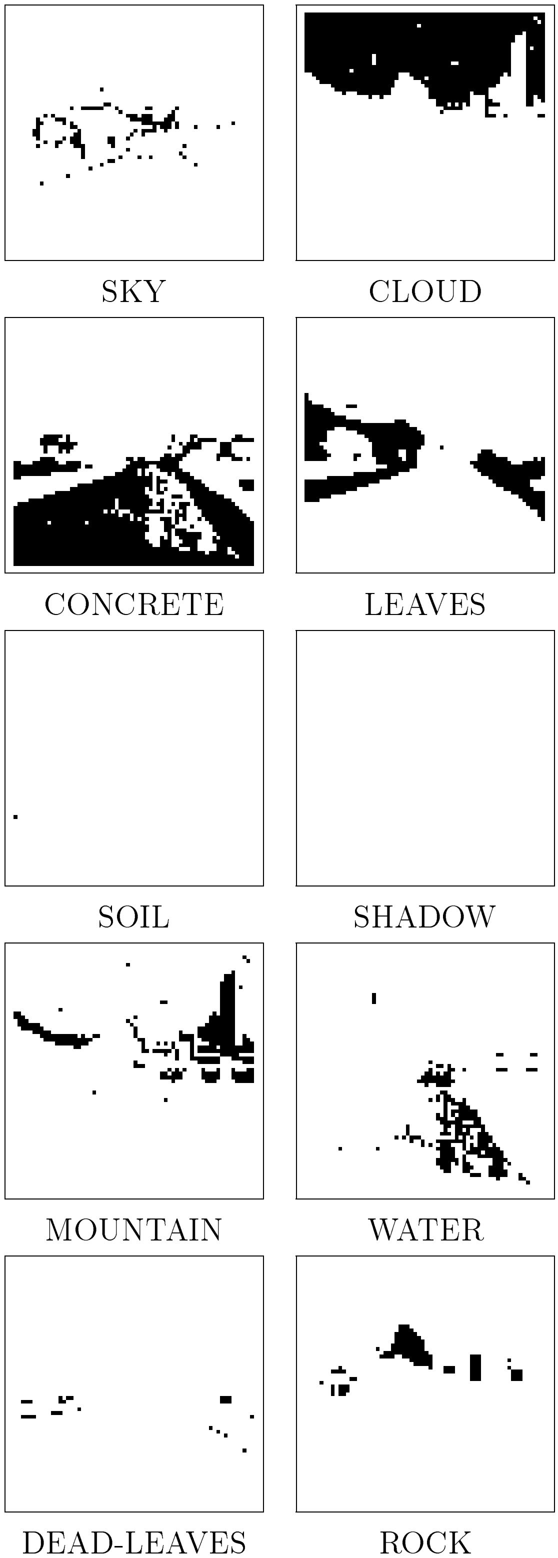


Figure 2.6: Results of object labeling

use the list of object labels appearing in the result as the index of the image, the image retrieval may fail. In order to cope with such errors, we employ a pictorial query method for retrieval.

2.4.2 Pictorial Query Method for Retrieval

A pictorial query method is the method which uses an image to express the condition of retrieval. We use an object sketch of the image for retrieval. The object sketch is an image where each pixel value represents the object label of that pixel in the original image.

Our object labeling method assigns labels to all the pixels in the image. We use the labeled image as the index of the original image and call it "the index image." The size of the original images is 256×256 pixels. In order to reduce the size of the index images, we assign an object label to every 4 pixels. The size of the index images thus becomes 64×64. This is sufficient resolution for the retrieval of outdoor images. The indexing process is automatically performed by computer.

To retrieve images, a user draws an object sketch of a goal image that he wants to retrieve. The user does not have to specify all labels of the pixels in the object sketch. The pixels that users do not specify labels are labeled as UNKNOWN by the retrieval system. The size of the object sketch is 64×64.

To select "candidate images," the system calculates the degree of similarity S between the object sketch and the index images. The degree of similarity is calculated as follows:

$$S = \frac{M}{N}$$

where

M : number of pixels residing at the same position in the object sketch and the index image and having the same object label.

N : number of pixels in the object sketch which have labels specified by the user, i.e.

64×64 - (number of UNKNOWN pixels).

The system then shows the candidate images in the order of similarity.

We use an object sketch. An object sketch is one useful way to express the goal images. Since images have much information, it is difficult to express the sort of objects, the positions of them, the spatial relations between them etc. with linguistic description.The object sketch can express these information simultaneously.

Furthermore, the pictorial query method is robust against errors in the index images. The recognition rate of our object labeling method is about 77%. We cannot know which pixels are correctly labeled and which pixels are incorrectly labeled only from the result of object labeling. But, at the retrieval stage, if a user can give the correct labels, the similarity between the index image of the goal image and the object sketch will be about 77%, which may higher value than that of the other images. Consequently, the goal image will be shown as higher candidate than other images, even if there are errors in the index.

2.4.3 Retrieval Experiments

Retrieval with Referring the Goal Image

We conducted retrieval experiments. The number of the images in the image library is 461 which include the images used in Section 2.3.4. First, we made the index images for all the images in the image library using the pixel-based object models constructed in Section 2.3.4. Second, we selected the 33 images from them which were not used to construct the object models. And third, we retrieved these images among the 461 images. We drew the object sketches referring to the goal images. It takes about 1.7 seconds for a retrieval on our machine, HP9000/735(124 MIPS).

The retrieval ratio of the retrieval is shown in Table 2.6(a). The retrieval ratio is the ratio that the goal image is within the top n-candidates. The Table shows that 60.6% of the images are presented as the first candidate. In this experiment,

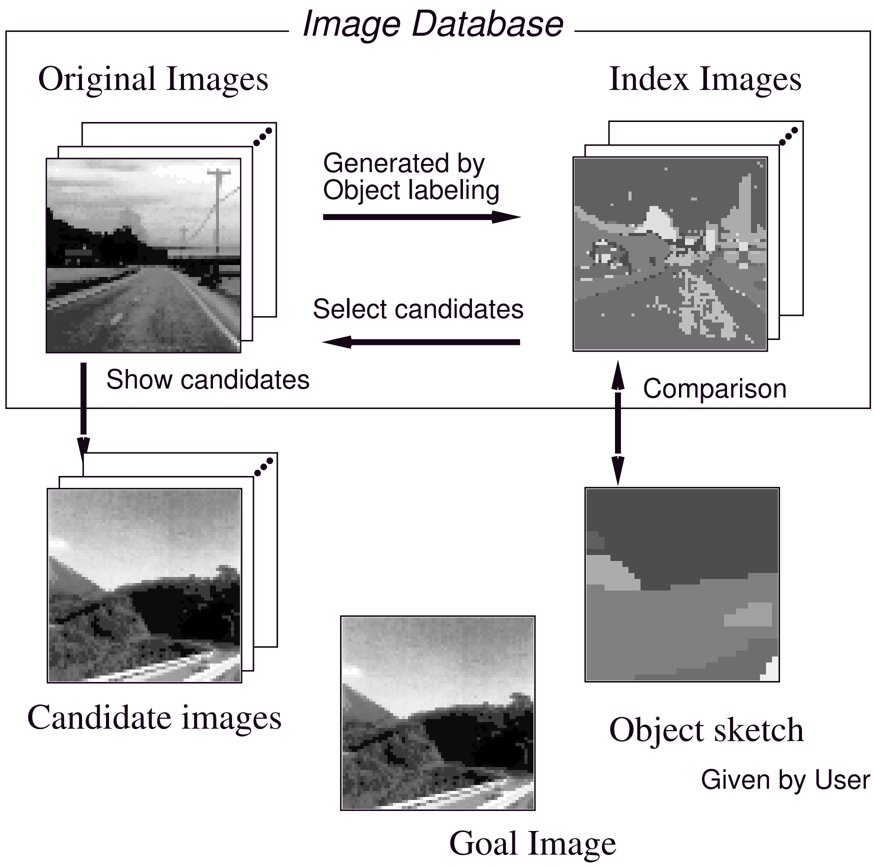


Figure 2.7: Retrieval method

Table 2.6: Retrieval ratio

(a) Referring the goal image

Top 1	Top 5	Top 20
60.6%	87.9%	93.9%

(b) Without referring the goal image

Top 1	Top 5	Top 20
19.0%	47.6%	71.4%

the object sketch had almost correct labels because it was drawn referring to the goal image. The retrieval ratio of the first candidate should be almost 100% in the ideal situation. But there are several images which are similar each other in the image library. They get down the retrieval ratio.

There were four images which were not within the top 5 candidates. In the worst case, the goal image was the 36th candidate. The similarity between the index image of the goal image and the object sketch was within $0.43 \sim 0.52$ for those four images. They are worse values than those of the other images. This shows that there are some images which cannot be labeled correctly. One reason of this problem is the light condition of the images. To improve the retrieval ratio, we should consider the method to get rid of such affection, as well as improve the average recognition rate of object labeling.

Retrieval without Referring the Goal Image

In a real situation, a user retrieves an image without referring to the goal image. We conducted an experiment simulating such a situation.

We showed several images in our image library to a user. The user selected an image from them as the goal image, and drew the object sketch without referring it. We tested for 18 user, with a total of 42 retrieval trials. The results of retrievals are shown in Table 2.6(b). The goal image was retrieved within the top twenty candidates for 71.4% of the trials. In the worst case, the goal image was the 184th candidate.

Examples of the typical cases of retrieval are shown in Figure 2.8-2.11. Figure 2.8 shows the successful case. There are 3 typical cases of retrieval failure. Figure 2.9 shows a case that fails because of a wrong index image. This is the problem of object labeling described in Section 2.4.3.

Figure 2.10 and 2.11 show failure caused by wrong object sketches. In Figure 2.10, the user draws the sky as a cloudy sky, but it is labeled as a blue sky in the index image. There were several cases where the users mixed up a blue sky and a cloudy sky, especially when the sky was not clear blue. People do not remember whether the sky is clear or cloudy. To cope with this problem, we should introduce a "sky" category which is compatible to these two categories. In Figure 2.11, the user draws the objects at wrong positions. The system compares the label of pixels at the same positions in the index image and the object sketch. This method is vulnerable to shift of position. We should also consider a solution to deal with this problem.

2.5 Conclusion

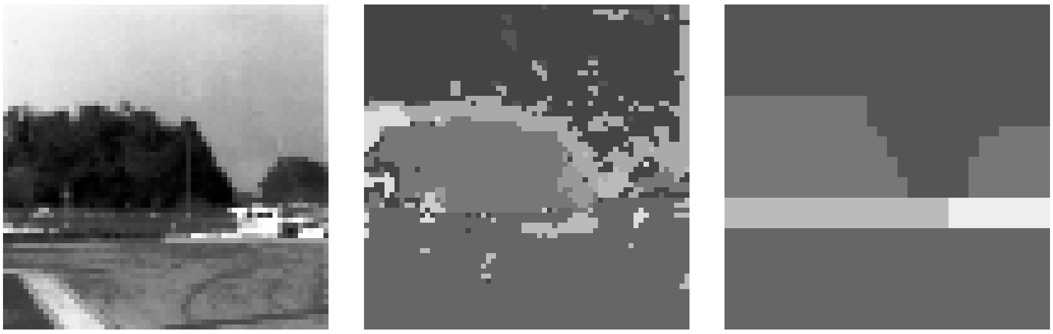
In this chapter, we described the method to generate the index for the image retrieval automatically. We proposed the pixel-based object labeling method for outdoor scene recognition. In traditional methods, initial regions each of which corresponding to an object are generated by the signal-level segmentation and then the object labels are assigned to the regions. In practice, the generated region does not have to correspond to an object because the objects show various statistical characteristics and the signal-level segmentation does not use the characteristics. In our method, we do not generate initial regions. Feature values are calculated from each pixels and the object label of the pixel is assigned using the object models. The object models are the relation between the feature values and the object labels. We constructed the object models with neural networks.

Goal image | Index image | Object sketch

Figure 2.8: Successful case

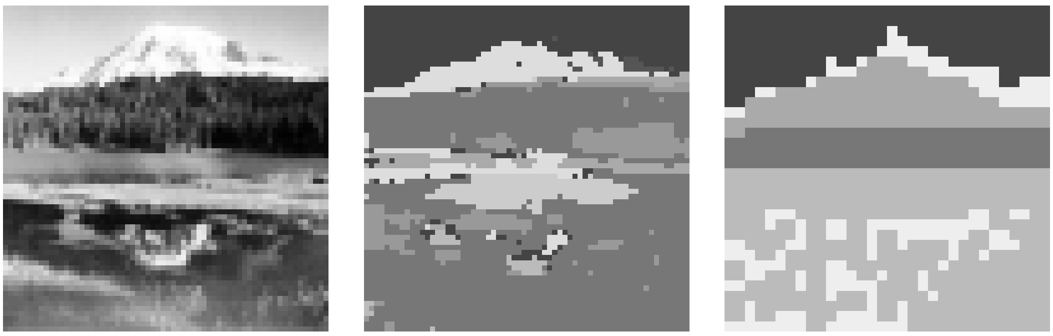
Goal image | Index image | Object sketch

Figure 2.9: Unsuccessful case: (1) Failure of indexing



Goal image Index image Object sketch

Figure 2.10: Unsuccessful case: (2) Drawing different object



Goal image Index image Object sketch

Figure 2.11: Unsuccessful case: (3) Drawing objects at different positions

We conducted the experiments to compare our method with traditional region-based methods. The features of shape and size are not obtained from each pixel. But the recognition rate with our method is higher than that of region-based ones. Moreover, we conducted the sensitivity analysis of the object models. The results showed that the features of shape and size do not affect the result of object labeling in our experiments. The recognition rate analysis also suggested the same conclusion.

Next, we applied the recognition results to use as the index of image retrieval. We employed a pictorial query method for retrieval. In this method, a user draws the object sketch of the goal image. Then the retrieval system compares the object sketch with the index images and shows the candidate images in the order of similarity between them. Although the recognition results include some errors, we expected that the pictorial query method is robust against errors in the index images because it compares a large number of pixels independently and the errors in some portion do not largely affect the retrieval results.

We conducted the retrieval experiments with the index images generated by the pixel-based object labeling method. The retrieval ratio of top 5 candidates was 87.9% when the goal image was referred. This result tells that we can use our pixel-based method for indexing images. When the goal image was not referred, the retrieval ratio of top 5 candidates was 47.6% and that of top 20 candidates was 71.4%. There are 461 images in the image library and there are similar images among them. The result is not bad under this condition. There are some problems in our retrieval method:

- We should prepare a large number of training data for the training of the neural networks.

- There are some images which cannot be labeled correctly and our retrieval method cannot cope with this.

- People sometimes do not remember the object correctly and sometimes draw the objects at different position from the goal image.

Some of them will be discussed in the following chapters.

Chapter 3

Automatic Extraction of Training Data for Index Generation from Images Containing Same Kind of Object

3.1 Introduction

The object recognition in the image by the computer is an important research subject. The results can be applied to many applications, such as automatic index generation for image retrieval system.

As the technique of object recognition, the method of discriminating in the feature space is effective. In this method, a large number of feature values which reflect the characteristics of "objects" in the image is required as training data. These training data were conventionally given by hand, but to recognize the "object" of various kinds, a lot of training data is needed, and there is a problem that the cost to give training data becomes large. Moreover, the criterion what data should be given as training data does not exist, although the recognition results depend on given training data. In order to solve these problems, the method of extracting data from images automatically is needed on the basis of the rough information about the "object," and the recognition system can use them as training data.

To cope with the problem, we propose the method which will carry out automatic extraction of training data required in order to recognize the objects in the images. In [28], they use the encyclopedia texts to extract the index of an image. It was the method for the retrieval with keywords. We use a large number of images containing the same kind of object. In this method, since a man should give only the rough information that an object is contained in the images, man's cost to give training data is smaller than the case where the image feature of the object is given by hand. Moreover, since the selection criterion of training data is included in processing, man becomes unnecessary to choose training data in arbitrary.

Although it looks unnatural at a first glance that the setup of using "a large number of the images containing the same kind of object," there exist image libraries each image of which has been indexed with the kind of object within the image as a key word is marketed, and it is an easily realizable setup.

In this chapter, we show the method to extract training data automatically from images containing the same kind of object. At first, we show the outline of the algorithm to extract training data and discuss how to use the constraint that "all images contain same kind of object." Our algorithm consists of 4 stages, the feature extraction stage, the projection to discriminant space stage, the clustering stage and the cluster selection stage. Then we show the implemented method to use the constraint in latter 3 stages. Below, the method to use this property and the experimental result will be shown.

3.2 Basic Algorithm

3.2.1 Flow of Processing

Figure 3.1 shows the flow of the processing which carries out automatic extraction of training data. As original images, a large number of the images which contain the same kind of object is given.

1. Extracting features from the original images (the feature extraction stage)

   At a fixed interval, the feature values is calculated for every pixel from the original images. The feature values to use is 4 elements. They are R, G and B pixel values and the vertical position of the pixel in the image (Y). These feature values are effective in object labeling as shown in previous chapter. Let the obtained feature values be the 4-dimensional feature vector.

2. Projecting the feature vectors to discriminant space (the projection to discriminant space stage)

   The each element of the feature vector has different characteristics. The vertical position Y has different unit from other RGB pixel value elements. The distributions of these elements in the original feature vector space are different. We cannot compare these elements in the original feature vector space.

   Furthermore, the original feature space is not the best space to get the training data. In order to get the training data which correspond to the object, the feature vectors of the object should be condensed and other feature vectors should not overlap the feature vectors of the object.

   We transform the feature vectors into more preferable vector space. We call this vector space as discriminant space and the projected feature vector into the discriminant space as the discriminant vector here.

3. Clustering in the discriminant space (the clustering stage)

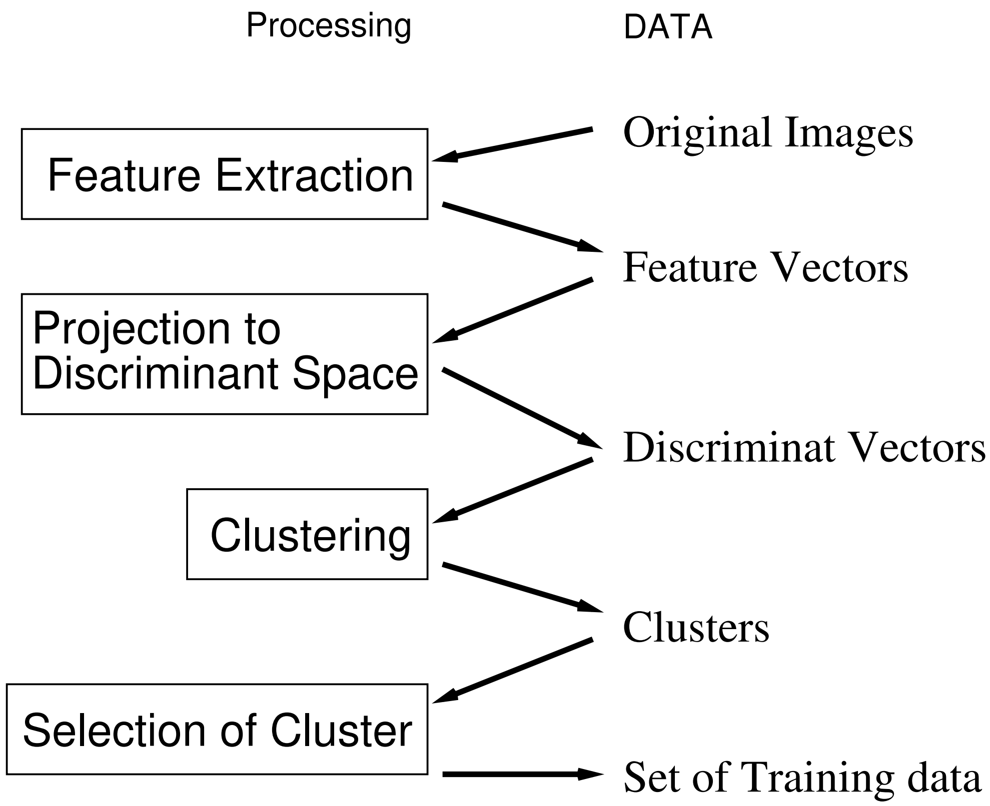


Figure 3.1: Flow of processing

We call the feature vector corresponding to the object as the object feature vector. The purpose of this algorithm is to get a set of feature vectors which mainly contains the object feature vectors. We use a clustering method to get such set, because the object feature vectors has dense distribution in discriminant space and such vectors can be extracted by the clustering.

4. Selecting the cluster corresponding to the object from the generated clusters (the cluster selection stage)

   Since the object feature vector is not given, it cannot generate only one cluster that is the object feature vector. By clustering, two or more clusters are generated. The cluster which mainly contains the object feature vectors should be chosen from them on a certain criteria. The cluster selected in this stage is called the training data cluster here.

As the result of this algorithm, we get the cluster which should contain the object feature vectors mainly. We regard all the feature vectors in the training data cluster as the training data and construct object models using them.

3.2.2 Use of the Constraint

Since an original image contains an object of the same kind, the total area of the object occupied in original images becomes large. The rate of the object feature vectors occupied in the feature vectors also becomes large. Therefore, the rate of the object feature vectors in the cluster obtained becomes large, and becomes easy to carry out automatic generation of training data by selection of cluster, no matter what clustering it may carry out (Figure 3.2). Moreover, since the distribution of the object feature vectors itself becomes dense, the cluster which consists of the object feature vectors tends to generate easily.

The conditions "all original images contain an object of the same kind" can also be used further (Figure 3.3). That is, the cluster corresponding to an object should contain the feature vectors obtained from all original images. This condition can be used in each stage of processing. The method how to use the conditions in each stage is considered as follows:

- The feature extraction stage

  The feature is extracted by calculating the image feature for every pixel. It is hard to use the above-mentioned conditions. However, the object of the same kind may have different features among the images. For example, the color of the object may be different because of the lighting conditions. Since all images contain the object of the same kind, and if the corresponding point in it is known, change of the color between images can be presumed and rectified.

- The projection to discriminant space stage

  We can use the principle component analysis method to get a discriminant space each axis of which has no correlation. This discriminant space best represent the distribution of the feature vectors. But in the preferable discriminant space, the object feature vectors should be condensed and other feature vectors should not overlap the object feature vectors. In order to obtain such discriminant space, we generate a vector space in which the feature vectors are distributed in wide spread. If the object feature vectors in all images have similar values, the object feature vectors still distribute dense in the space. Other feature vectors distribute dispersed. As the result, we can obtain the preferable discriminant space.

- The clustering stage

  In the clustering stage, the desirable cluster is a cluster which consists only of the object feature vectors, or a cluster like it. According to the above-mentioned conditions, this cluster should include the feature vectors from all original images. Therefore, if we use the method in generating the discriminant space that each cluster contains the feature vectors coming from as much original image as possible, the desirable cluster will tend to be generated.

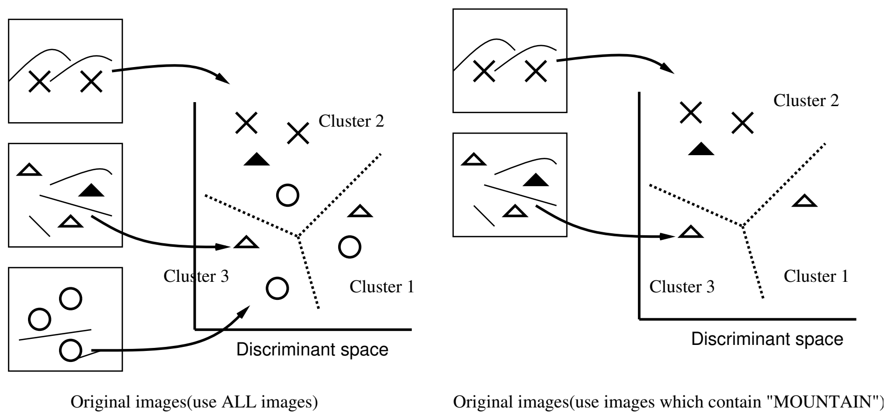


Figure 3.2: Effect of constraint "Containing Same Kind of Object"(1)

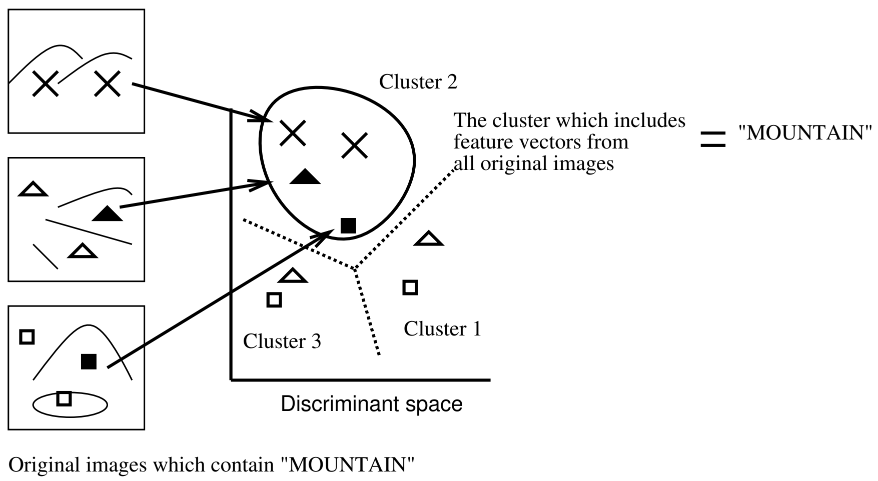


Figure 3.3: Effect of constraint "Containing Same Kind of Object"(2)

- The cluster selection stage

  The training data cluster should contain the object feature vectors mainly. We prepare negative example images, which do not contain the object. Here, the condition that "all images do not contain the object" can be used in the negative example images. It is the reverse condition to the above-mentioned conditions. We can obtain the training data cluster by selecting the cluster which contains more feature vectors from the positive example images and less feature vectors from the negative example images.

At present, we do not use the condition in the feature extraction stage. In the following, the implementation to use the conditions in the projection to discriminant space stage, the clustering stage and the cluster selection stage is shown.

3.3 Construction of the Discriminant Space

The principal component analysis and the discriminant analysis are usually used for construction of discriminant space.

The principal component analysis is the technique for expressing the multi-dimensional vector with a small number of variables (vector of a low dimension). In this technique, the axis is taken sequentially in the order that the distribution along the axis is the maximum. In the resulting vector space, each axis has no correlation and the distribution along the axis is normalized. As the result, we can measure the distance between feature vectors by the Euclid distance.

The discriminant analysis (or the canonical analysis) is the technique of building the discriminant space where the distribution within each group becomes small and the distribution between the groups becomes large when two or more data groups are given. In this space, since it is gathered within the inside of each data group and it is scattered between different data groups, the generated discriminant space becomes effective in discriminating these data groups.

For the discriminant analysis and the principal component analysis, an optimum solution can be analytically found as linear conversion by solving an eigen value problem.

If the object category of each feature vector is given, we apply the discriminant analysis technique over the categories. The result will be the best discriminant space in the respect that the object feature vectors have dense distribution and other feature vectors tend not to overlap them. But in our problem, the object category is not given. In such situation, the principal component analysis is usually applied.

We propose a method to generate the discriminant space that carry out the discriminant analysis for every image (Figure 3.4). The object of the same kind is contained in all images. Supposing that the feature values of the pixel corresponding to the object are similar over the images, no matter what conversion it may be applied to the feature vectors, the object feature vectors will be having condensed as it is.

Therefore, if the feature vectors are divided into data groups each of which contains all the feature vectors from the same image and the discriminant space is built by the discriminant analysis over these data groups, the object feature vectors will remain dense and the other feature vectors will be scattered more. Then in the resulting vector space, the cluster in which the rate of the object feature vectors is high will be easy to generate in the clustering stage.

3.4 Clustering under the Constraint

The method of Kononen's Self Organizing feature Map (SOM) is used for clustering. Fundamentally, this method is similar to the K-mean clustering. In this method, at first, an initial cluster center for each cluster is determined. Then, the distance between the initial cluster center and the feature vector is calculated. Thirdly, the feature vector is classified to the cluster to which the distance is nearest. Fourthly, the new cluster center is re-calculated with the new classification results. The processing will be repeated and at last the desirable cluster is obtained. If the discriminant space is settled, since the distance between the feature vectors will also be settled, the clustering result becomes settled uniquely in most cases.

In order to take into consideration the constraint "containing the object of the same kind" in a clustering stage, we make a certain change in clustering processing.

We propose a method to weight by the ratio of feature vectors in the cluster (Figure 3.5). In order to make clusters contain the feature vectors from as many original images as possible, a weight is applied in comparing the feature vector and the cluster center so that if the cluster contains many feature vectors which comes from the same image as the comparing feature vector, then the distance between the feature vector and the cluster center becomes far, and if not many, the distance becomes near.

Here, we consider the distance between the feature vector A and the center point C of the Cluster1 (see Figure 3.5). The feature vector A comes from ImageA and the Euclid distance between A and C is L. The weighted distance is calculated as follows:

$$\mathrm{Weighted\_Distance} = L \times \left(1 + \sqrt{\frac{D_A}{D_{all} + 1}}\right),$$

where D_{all} is the number of the feature vectors in Cluster1 and D_A is the number of the feature vectors from ImageA in Cluster1.

3.5 Selection of the Training Data Cluster

Once several clusters are generated, we must select the object cluster. We introduce negative example images. The negative example images are images which does not contain the object of which we want to extract the training data. We call the original images which contain the object as positive example images. The procedure goes as follows(Figure 3.6) :

1. Extracting the feature vectors from the negative example images. The feature values are calculated pixel-by-pixel manner in similar way to the

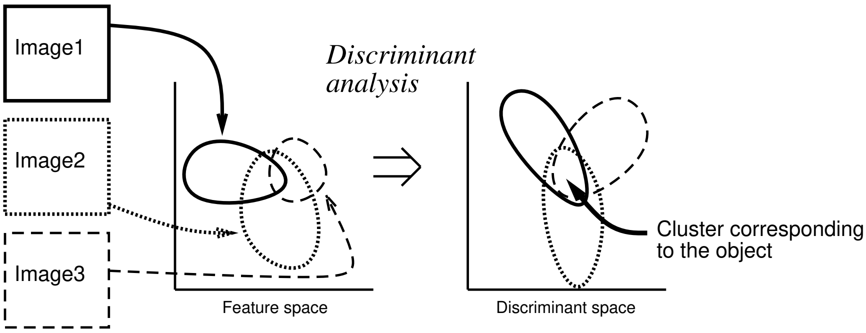


Figure 3.4: Method using discriminant analysis

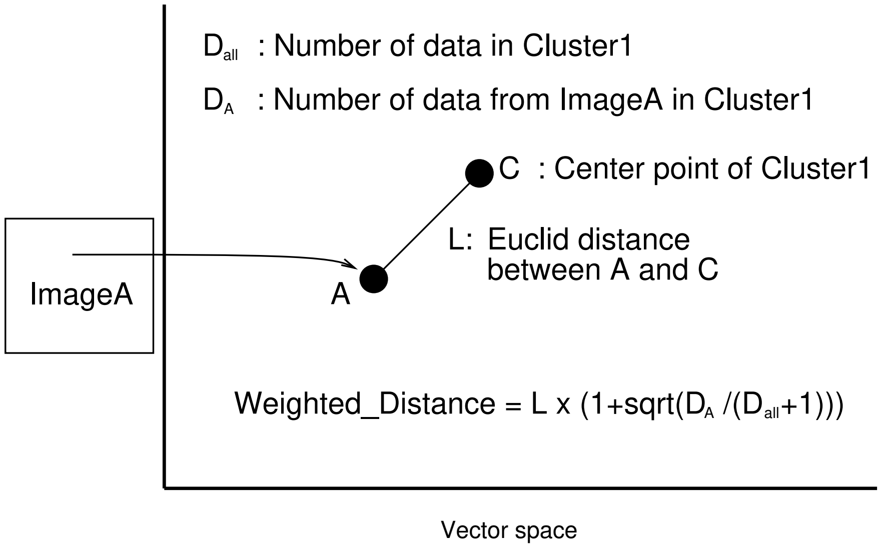


Figure 3.5: Method using weighting by feature vector ratio

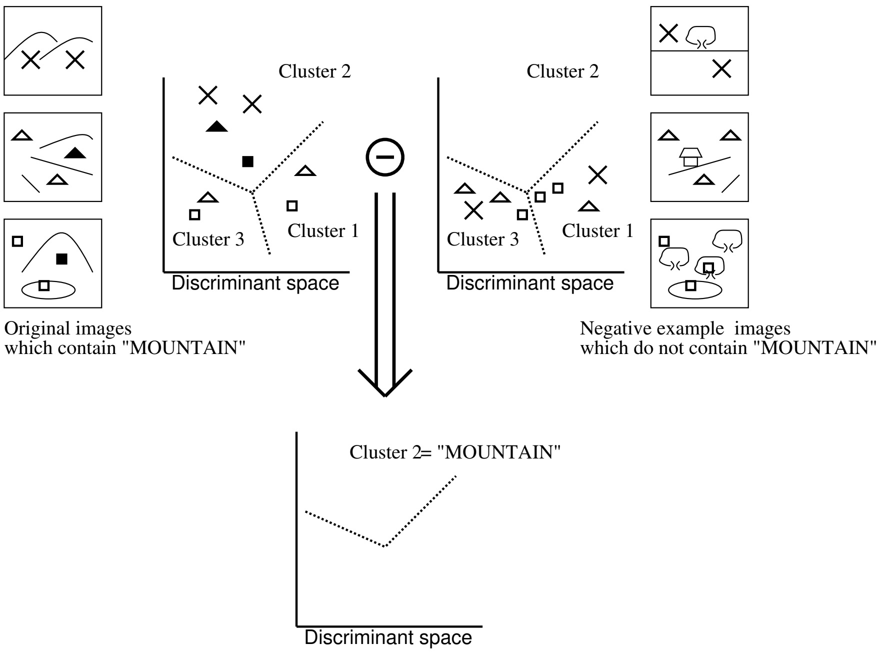


Figure 3.6: Selection of training data cluster using negative examples

positive example images. Each feature vector is projected to discriminant space.

2. Classifying feature vectors from negative example images into the clusters which have been generated from the positive example images.

3. Counting the number of the feature vectors in the clusters.

4. Selecting the clusters which contains more feature vectors from the positive example images and fewer feature vectors from the negative example images.

If a cluster is the object cluster, the cluster contains more feature vectors from the original image, but it contains fewer feature vectors from the negative example images, because the negative example images do not contain the object. If a cluster is not the cluster corresponding to the object, the cluster contains almost same number of the feature vectors. Considering that, we can select the training data cluster by the above criterion.

Formally, we select the clusters which satisfy the following criteria as the training data cluster.

$$\frac{NC}{PC} \times \frac{TP}{TN} < \text{threshold},$$

where PC/NC is the number of feature vectors from positive/negative example images in the cluster, and TP/TN is the total number of feature vectors in the positive/negative example images, respectively. The threshold is set to 0.5 in the following experiment.

3.6 Experiments

3.6.1 Experimental Method

We conducted the experiments to extract training data automatically using the constraint that "containing the object of the same kind."

We prepared 130 outdoor images. We selected the 10 objects in them which are same as those in Chapter 2 (Table 2.1). For each object, we selected several positive example images each of which contained the object more than 10% of the area in it and negative example images each of which does not contain the object in it. The number of the positive and negative example images for each object is shown in Table 3.1.

In our experiments, the following three methods were used to construct the discriminant space:

- Method using discriminant analysis with object category information
- Method using discriminant analysis with original image information
- Method using principal component analysis

The clustering was conducted with the following two method:

- Normal SOM method
- SOM with weighting by feature vector ratio method

3.6.2 Experimental Results

Table 3.2 shows the result of the training data extraction. The column of "Normal SOM" shows the result using normal SOM clustering method and the column of "SOM weighting by ratio" shows the result using SOM with weighting by feature vector ratio method described in Section 3.5. The number of clusters in clustering affects the result. We show the result when the number of clusters was 17.

The column of "BEST" shows the result of the method using discriminant analysis with object category information. We can construct the best discriminant space with this method. When applying our method, the object category information is not given and cannot employ this method. The result is shown only for reference how good result can be obtain when the best discriminant space is given. The column of "DISC" shows the result of the of the method using discriminant analysis with original image information described in Section 3.3. The column of "PCOMP" shows the result of the method using the principal component analysis.

The "correct data" means the simple mean of the ratio of the correct object feature vectors in the training data for each object. It can be shown that the "BEST" shows highest discriminant ability. Although the information about the cluster cannot be known a priori in the real application, the result can be regarded as the higher limit of our method. We set the threshold to 0.5 in selecting the training data cluster stage. It is expected that the rate of correct data in the training data cluster will be 50%. The result shows 67.9% or 69.9% which are better than the expected rate.

In the rest results, the combination of the "DISC" discriminant space construction method and the "SOM weighting by ratio" clustering method shows the best result. Table 3.3 shows the distribution of the number of images in each cluster generated by each clustering method. In all the results, the mean value of images in each cluster is larger and the variance of that is small with "SOM weighting by ratio" method. This result shows that the "SOM weighting by ratio" method has the preferable tendency that every cluster contains as much number of positive example images as possible. The "SOM weighting by ratio" method shows better result than "normal SOM" method in the "BEST" and "DISC" discriminant space construction method. This result shows the effectiveness of our clustering method. The analysis of the reason why the "PCOMP" method shows opposite result is left as future works.

3.7 Conclusion

In this chapter, we proposed an algorithm to extract training data from images which contains same kind of object. In our pixel-based object labeling method, a large number of training data is necessary for the training of the neural networks. The "large number of the images containing the same kind of object" can be easily gotten from image libraries each image of which has been indexed with the kind of object within the image as a key word. This algorithm decrease the cost to prepare the training data.

Table 3.1: Number of example images for each object

Label	Positive	Negative
SKY	31	62
CLOUD	32	52
CONCRETE	16	77
LEAVES	44	17
SOIL	3	120
SHADOW	8	101
MOUNTAIN	10	61
WATER	29	70
DEAD-LEAVES	10	102
ROCK	9	110

Table 3.2: Result of training data extraction

Clustering	normal SOM			SOM weighting by ratio		
method	BEST	DISC	PCOMP	BEST	DISC	PCOMP
correct data (%)	67.3	59.8	63.8	69.9	64.9	59.4

The algorithm consists of 4 stages: the feature extraction stage, the projection to discriminant space stage, the clustering stage and the cluster selection stage. We used the constraint that "all images contain same kind of object," and proposed a discriminant space construction method and a clustering method. Furthermore, when selecting the training data cluster, we prepared the negative example images and use the constraint that "all images do not contain the object in the negative example images."

The experimental results show that by combining the proposed methods, better training data can be obtained comparing with the normal method. We expected that about 50% of the data in the training data cluster is correct training data. In the results, the correct data was 64.9% which is better than the expected result.

At present, we need the negative example images to select the training data cluster. Although this method is effective and reasonable, it takes some extra costs to prepare the negative example images. To investigate the method to select the training data cluster without the negative example images is left as future works.

We will use the generated training data for the training of neural network and generate the index images for image retrieval. The result of the retrieval will be shown in the following chapter.

Table 3.3: Distribution of images in each cluster(object=SKY)

Discriminant space		Normal SOM	SOM weighting by ratio
BEST	mean	23.1	24.1
	variance	34.5	31.6
DISC	mean	22.0	25.1
	variance	45.6	25.2
PCOMP	mean	22.8	28.9
	variance	43.8	25.3

Chapter 4

Index Generation and Retrieval to Cope with Recognition Errors

4.1 Introduction

The pixel-based object labeling method is the technique of having taken the pattern-recognition-technique to recognition of outdoor scenes. The knowledge about an object is acquired by learning. Thereby, it becomes unnecessary that man analyzes the character of the object in detail and gives a computer the characteristics. As the result, anyone can construct recognition processing easily and the recognition method is enabled to be used as a tool.

However, in the method of gaining an object model by learning, training data for learning needs to be given by hand.

In the pixel-based object labeling method, the information that each pixel is contained in what object needs to be given, and it takes too much cost to preparing many training data. Moreover, since the built object model greatly depends on the character of training data, the skill of the man who gives training data will influence.

On the other hand, there are electronic image libraries which include a large number of images. In some image libraries, the kind of object contained in the image is given as index keyword. In the previous chapter, we used such image library to automatic extraction of the training data for learning. It can be considered that the auto-generated training data should be used in the pixel-based object labeling method.

However, it contains not only the information derived from the desirable object but also the information derived from other objects. This may lead the learning process to undesirable direction. If we use the automatic extracted training data, there may be many errors in the recognition results as well. We should cope with such errors to avoid the undesirable situation.

In this chapter, we develop a method to use the result of auto-generated training data in the pixel-based object labeling method. Especially, we concern the method to cope with the error in the recognition results. There are several researches to cope with the error or noise in the training data or recognition results[30, 31, 32, 33]. We assume that the result is used in image retrieval application and analyze the characteristics that our retrieval method has. Then we develop a new technique to use the erroneous recognition results as the index of images and show the experimental results.

4.2 Extraction of Training Data

The outline of the method to extract the training data is as follows:

1. Preparing images which contains the same kind of object and has different backgrounds. The images is called positive example images.

2. Calculating feature values for each pixel of positive example images. As the feature values, RGB colors and vertical position Y in the image is used.

3. Mapping the feature values to the feature space. Each feature value is considered as a multi-dimensional vector in the feature space. We call it as a feature vector.

4. Transforming the feature space into discriminant space using the discriminant analysis.

5. Generating clusters in discriminant space to gather the similar feature vectors in the same cluster.

   For clustering, improved Kohonen's Self Organizing feature Map (SOM) clustering method is used.

6. Selecting the cluster corresponding to the object and using it as the training data.

   To select the cluster, at first, a large number of images which does not contain the object is given as the negative example images. Then, feature vectors from the negative example images are mapped to the generated clusters in the same way as those from the positive example images. The clusters which contain many feature vectors from the positive example images and contains few feature vectors from the negative example images are selected as the object cluster.

Here, assumption "several clusters will correspond to the object if clustering is performed for a large number of positive example images" is placed.

4.3 Using the Automatic Extracted Training Data in Recognition

4.3.1 Method

We conducted the experiment to generate index images by training data obtained with previous chapter. Experimental environment is shown in Table 4.1.

We have used 4 kinds of feature values to extract training data, but they are not sufficient to object recognition. When we obtained training data, we re-calculated the 19 kinds of feature values from each training data.

In the pixel-based object labeling method, the feature values are calculated from the neighboring pixels of the pixel. It is considered that the feature values obtained from the neighboring pixels are the feature values which the pixel of the center has. The feature values represent the color, position and texture features. We used 15x15 neighboring pixels to calculate the features.

The object to be labeled are the same 10 kinds objects as Chapter 2. These objects consist of natural objects. The category of each object is called as "class."

The number of training data for every class and the rate of the correct training data are shown in Table 4.2. Training data contains 2.3% – 76.1% of the error for every class.

An object model was built using obtained training data. The neural network of three-layer perceptron was used for constructing object models. The number of nodes in intermediate layer was 11. The back propagation algorithm is used for training. The recognition rate for non-training data was 52.5% after the training. When training data is given by hand, the recognition rate was 70.7%.

4.3.2 Results

In order to evaluate 52.5% of recognition rate, the object models were used and the index image was generated. The number of original images was 130. For each image, the index image was generated both by hand and by computer using the generated object models. The image which added the object label by hand is called as the "correct label image."

Next, the correct label image was given as an object sketch, and image retrieval was performed by comparing with the index images. The degree of similarity between label images is defined as the rate of the same object labels in the same position. The candidate images were shown in the order of similarity. We evaluate the retrieval results with the rank that the original image corresponding to the correct label image was shown in the candidates.

The retrieval results are shown in Table 4.3. The rate with which the original image was correctly retrieved as the first candidate was 68.5% when the training data was given by hand and 39.2% when the index image was generated by the above-mentioned method. It turns out that the retrieval result of the above-mentioned technique is not so good.

4.4 Analysis of Retrieval Processing

According to the result of previous section, when the training data is given by hand, the recognition ratio is 70.7% and the retrieval rate is 68.5%. On the other hand, when the training data is generated automatically with our method, the recognition ratio is 52.5% but the retrieval rate is 39.2%. The difference between these results should be caused with the error in the training data, but the relation between the recognition rate and the retrieval ratio is not clear. The purpose of this chapter is to clarify the relation among them and propose an improved method for image retrieval which cope with such errors.

4.4.1 Analysis by Random Error Model

We consider the recognition model in which a recognition error occurs at random independently with the feature values (refer to Figure 4.1).

In Figure 4.1, we suppose that the correct label image of an original image1 is given as the object sketch. The index image1 is generated by the computer for the original image1. The index image2 is also generated by the computer for other original image2. The area A, B, C and D correspond to same object (the label of which is L) in the original images. The area A, B and C∪E exists at the same position in the images. All the pixels in the area A have the object label L. Most of the pixels in the area B, C and D also have the object label L but some of them have the other object labels because of the errors in the index generation. Most of the pixels in the area E have other object label than L but some of them have the object label L because of the errors.

Here, we suppose that the recognition rate of the image to be used for generating index images was r, the similarity between the original image1 and the

Table 4.1: Experimental environment

Kinds of objects:	10 (BLUESKY, CLOUDYSKY, ... etc.)
Kinds of features:	19 (Color, Position, Texture)
Clustering method :	Kohonen's Self Organizing Feature map
Learning of object model:	3Layer perceptron, 11 intermediate nodes, Training with back propagation method
Image retrieval:	Given correct label image as the retrieval key Similarity is defined as the ratio of coincidence in object label

Table 4.2: Auto extracted training data

Class	Number of data	Correct ratio(%)
BLUESKY	957	97.7
CLOUDYSKY	884	96.9
CONCRETE	450	73.6
LEAVES	3433	68.2
SOIL	84	67.9
SHADOW	660	42.3
MOUNTAIN	989	44.6
WATER	972	81.4
DLEAVES	814	53.4
ROCK	188	23.9

Table 4.3: Experimental results

	Auto extracted	Given by hand
Recognition ratio	52.5%	70.7%
Retrieval rate		
Top 1	39.2%	68.5%
Top 5	61.5%	90.0%
Top 10	71.5%	93.1%

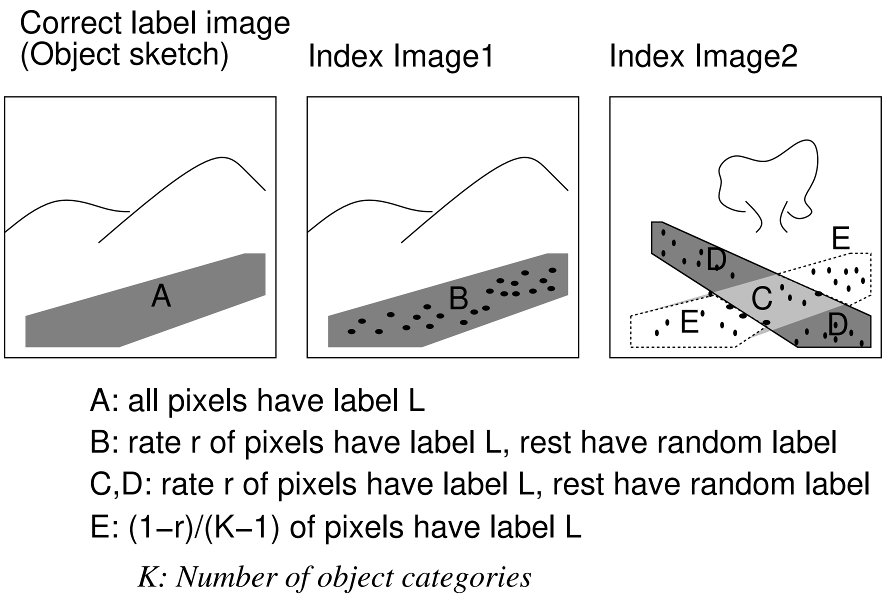


Figure 4.1: Characteristics of similar image retrieval using object sketch

original image2 was S and the number of the kind of objects was K.

The simulation of the recognition error is carried out by the random error. In this case, in the portion of A and B, the similarity becomes r. In the common portion of A and C, the similarity becomes also about r. In the common potion of A and E, the similarity becomes

$$\frac{(1-r)}{(K-1)}$$

This formula can be applied all the portions which contain the other objects than the original image1 in the index image2. As the result, the similarity between the object sketch and the index image2 becomes

$$S \cdot r + (1-S) \cdot \frac{1-r}{K-1}.$$

The similarity between the index image1 and the object sketch is r. Therefore, the necessary and sufficient condition that the original image1 is shown as higher candidate than the original image2 for the object sketch is

$$r > S \cdot r + (1-S) \cdot \frac{1-r}{K-1}.$$

i.e.

$$r > \frac{1}{K}. \tag{4.1}$$

4.4.2 Analysis by Image Similarity Model

In an actual recognition result, a recognition error does not arise regardless of the feature values. Since the object model is obtained by learning in the pixel-based object labeling method, the portion which is common among two or more images has high possibility that the labeling will be carried out similarly. Then, we consider the recognition model that the portion which is common between images is carried out correctly.

Supposing that the recognition rate is r, the kind of object to be labeled is K and the similarity between original image1 and original image2 is S,

- Similarity between the object sketch and the index image1 is r.
- Similarity between the object sketch and the index image2 is

$$S + (1 - S)\frac{1 - r}{K - 1}$$

.

It is assumed that the portion of the object which differs between the image1 and the image2 is mistaken at random.

According to the upper formula, the following partial equation is obtained.

$$r > S + (1 - S)\frac{1 - r}{K - 1}.$$

Moreover, from the upper formula, the following partial equation is the necessary condition.

$$r > S \tag{4.2}$$

4.4.3 Analysis of Retrieval Error

We discuss the reason why the retrieval error is occurred based on the result in the previous sections.

1. There is inclination in the tendency of recognition errors.

   If the recognition errors arise at random regardless of the feature values or the objects, the correct retrieval results are expectable even at the very low recognition rate according to the Formula (4.1). In our retrieval system, since the number of object K is 10, the necessary and sufficient condition to the recognition rate is no more than 10%.

   However, there is inclination in the tendency of recognition errors. If the feature values of two objects are similar, the recognition processing confuses these two objects. On the other hand, If the feature values of two objects are not similar, the recognition processing seldom confuses these two objects.

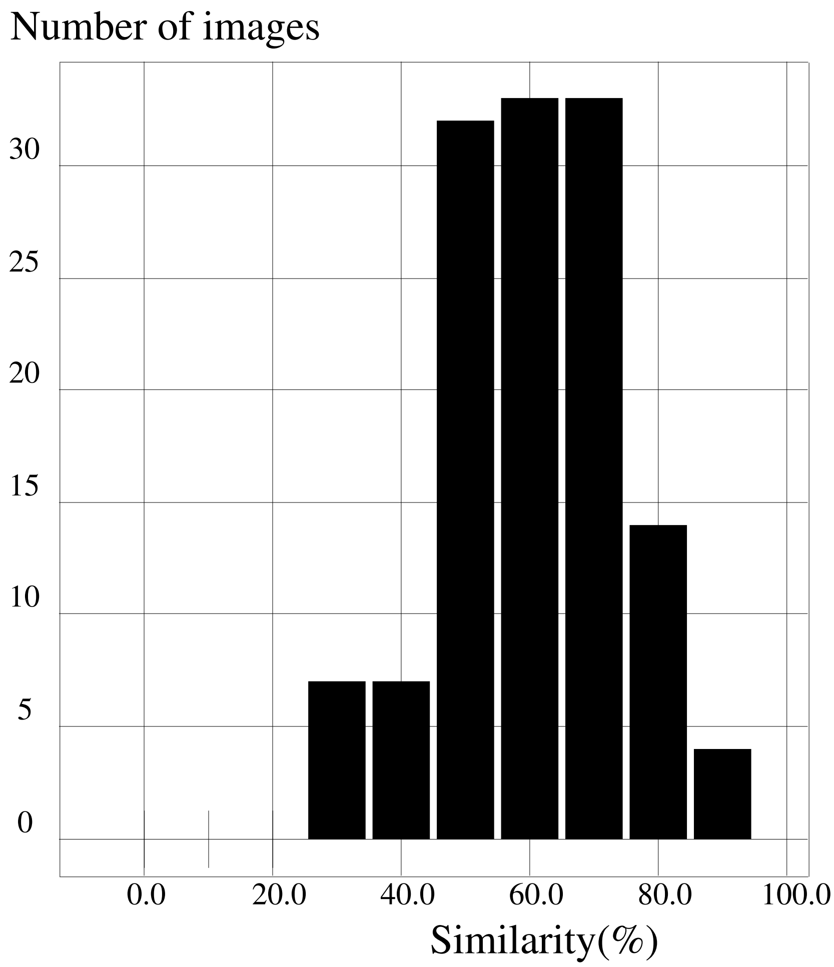


Figure 4.2: Distribution of similarity between original images

Table 4.4 shows the inclination of errors in the object labeling. For example, object SKY tends to mislabeled as CLOUD but does not tends to mislabeled as SOIL or SHADOW. This is equivalent to the number of object K having decreased in the Formula (4.1), and in order to obtain the correct retrieval results in such cases, the higher recognition rate is needed.

2. The similarity between the original images is high.

   The images we use are outdoor scenes which mainly contain natural objects. In the case of such images, the number of kind of object in the image is not so many and the composition of the image is comparatively alike. Therefore, the original images are often alike.

   Figure 4.2 shows the distribution of the similarity between the correct label images. Among the 130 images, there are four images which have the similarity of 90% or more. The average of the similarity among the 130 images is 65.7%. According to the Formula (4.2), 65.7% or more of the recognition rate is required in this case.

4.4.4 Improved Retrieval Method

According to the discussion in the preceding sections, the preferable nature of the index images is that:

- The errors in the index images are at random.
- The index images are not similar each other.

Considering this nature, we propose two methods to modify the index images using the object labeling results.

1. Correcting the labeling results reflecting the tendency of recognition error

   As shown in Section 4.4.1, the nature of random error is preferable in our image retrieval method. But if an object label is simply changed at random in order to increase the random nature of the recognition error of a labeling result, the portion changed at random will only commit as Don't-Care which hardly affects the retrieval result.

If the object label is changed in the probability reflecting the tendency of the recognition errors, a possibility that a required image can be retrieved if labeling of other portions is carried out correctly, since a part of information which the pixel-based object labeling method added is saved.

2. Using the result obtained from the several neural networks.

   In training of a neural network, the result depends on the initial value of the weight of the network, or the process of training[36]. It is thought that the neural network which changed initial value and trained shows a different clustering result. If the result obtained from two or more neural networks is used and at least the one network attach the correct label, a possibility that the required image can be retrieved will grow since the information of the correct label can be reflected in the result.

4.5 Experiments

The method to improve the retrieval results described in the previous section is applied to actual data. The experiments are conducted in the same environment as the experiments performed in Section 4.3.

4.5.1 Experimental Method

- Correcting the labeling result reflecting the tendency of recognition errors

  The object label is determined probability according to the tendency of the errors of object labeling.

  Let $P(j|i)$ be the probability that the labeling result is i when the right label is j. A uniform random number is generated, and when a labeling result is i, final label j is applied by probability $P(j|i)$.

- Using the results obtained from several neural networks

  Four neural network are trained with different initial values. After the training, each network bears the label of one pixel within the 2x2 pixels in an original image. A labeling result is collected and it is used as an index image. For the four neural networks, each network shows about 60% of the recognition rate.

4.5.2 Experimental Results

Table 4.5 shows the retrieval results using the index image generated by each method. The column "By hand" and "Generated" shows the same results as Table 4.3. The column "RAND" shows the result of "Correcting the labeling result reflecting the tendency of recognition errors." The column "MOD4" shows the result of "Using the results obtained from several neural networks." The column "MOD4+RAND" shows the result that the "RAND" method is applied to the result of "MOD4" method. Figure 4.3 shows examples of the index images.

In all method, Top 1 and Top 10 retrieval rate is improved comparing with the original results. The "RAND" method shows best result. The "MOD4" method itself shows better result, but the combined method ("MOD4+RAND") shows almost same result as the "RAND" method. The result shows that the "RAND" method is most effective among these methods.

4.6 Conclusion

In this chapter, the problem of the errors in the index images was discussed. The index images are generated with the pixel-based object labeling method by a computer. The recognition rate of this method is 50-70 % and the results contain some errors. The relation between the recognition rate and the retrieval ratio using the recognition results as the index images is not clear. To analysis the relation, we introduced two recognition error models.

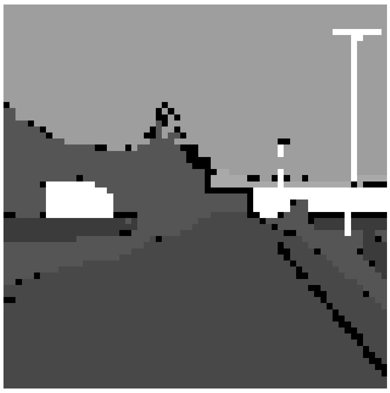
Table 4.4: Inclination of errors in object labeling

Correct label	Result of labeling(%)									
	SKY	CLD	CNC	LVS	SOI	SHD	MNT	WTR	D-L	RCK
SKY	68.4	20.9	0.3	5.8	0.0	0.0	0.0	3.2	1.0	0.4
CLD	29.3	58.8	0.0	6.7	0.0	0.0	0.2	4.4	0.6	0.0
CNC	0.6	0.2	28.4	24.1	0.0	0.0	0.0	28.6	17.3	0.8
LVS	0.5	0.2	1.0	85.0	0.1	0.4	2.3	3.3	6.4	0.8
SOI	0.0	0.0	3.1	75.3	3.1	0.0	0.0	8.2	9.3	1.0
SHD	0.0	0.0	0.0	63.1	1.9	14.6	11.7	8.7	0.0	0.0
MNT	11.7	16.0	0.2	23.5	0.2	0.0	23.5	24.8	0.2	0.0
WTR	1.9	2.7	3.9	38.2	0.2	0.2	0.1	44.8	6.3	1.7
D-L	0.8	0.0	0.0	57.7	0.0	0.2	3.8	0.6	35.2	1.8
RCK	0.9	1.2	7.0	48.4	0.0	0.3	0.0	8.8	15.5	17.9

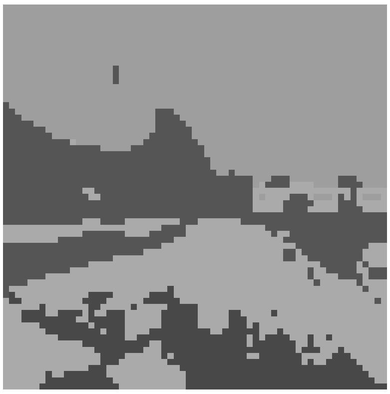
Table 4.5: Retrieval ratio(%)

	By hand	Generated	RAND	MOD4	RAND+MOD4
Top 1	68.5	39.2	42.3	40.5	43.1
Top 5	90.0	61.5	71.5	66.2	71.5
Top 10	93.1	71.5	82.3	79.2	81.5

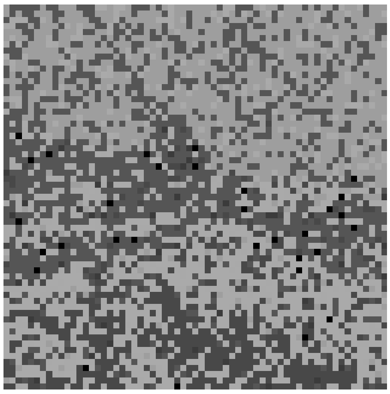
(1) Original image



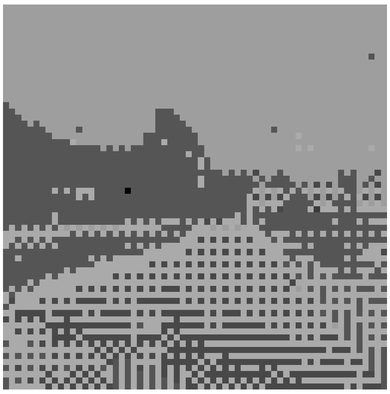
(2) Object sketch



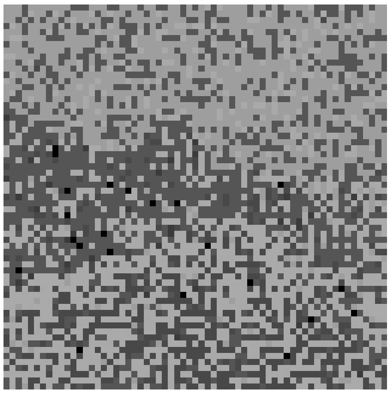
(3) Genarated



(4) RAND



(5) MOD4



(4) MOD4+RAND

Figure 4.3: Examples of index images

At first, we presented the random error model in which the recognition error occurs at random. According to this model, the necessary and sufficient condition for the recognition rate r in order to show the goal image at the first candidate is $r > 1/K$, where K is the number of the kind of the objects. In our image retrieval system, the necessary recognition rate is no more than 10% since the number of the kind of the objects K is 10. The analysis with this model revealed that the randomness in the recognition error is very preferable nature in our image retrieval method.

Then we presented the image similarity model. In this model, the common objects between the original images at the same position are correctly labeled while the other objects are labeled with some random errors. The necessary condition for the recognition rate r is $r > S$ in this case, where S is the similarity between the original images. Since the average similarity is 65.7% in our image library, the necessary recognition rate is also 65.7% in our image retrieval system. The analysis with this model revealed that if the original images are similar each other, it is necessary for object labeling results to be high recognition rate. It also suggests that the uniformity among the index images is not preferable nature in our image retrieval method.

Considering the results of the analysis, we proposed two methods to cope with the errors in object labeling. The one is correcting the labeling results reflecting the tendency of recognition errors. By this method, randomness of the object label in the index image will increase. The other is using the results obtained from several neural networks. By this method, the affection of burst error will decrease.

We applied these two methods and conducted the retrieval experiments. The results showed that the both methods improve the retrieval results. Although these methods improve the results, the results of retrieval are still not as good as the case when the training data was given by hand. It is left as future works.

Chapter 5

Improving Retrieval Results Using Retrieval Examples

5.1 Introduction

In recent years, it is enabled to accumulate a lot of images on computers thanks to improvement of the computer performance and the enlargement of the capacity of storage on computers. Furthermore, the necessity for the image library is increasing by arrival of the time "multimedia" which utilizes combining the various media, such as a sound and an image or in the field of the design using images, such as advertisement or publication industry.

In order to utilize a lot of accumulated images, a required image must be able to be retrieved from them. In image retrieval, the index of the image is given when the image is accumulated. When a user want to retrieve an image, he expresses his demands as a key and the retrieval system shows the image whose index match the key. For example, when a user want "the image in which the mountain of a long distance is contained," it needs to give beforehand whether "the mountain of a long distance" is contained in the image as an index. Such an index is usually given by hand. In this case, the cost to give the index becomes a problem.

To cope with this problem, there are several researches in which the technique of the feature extraction from an image is applied to an index generation and automates the index generation[9][10]. In these researches, the image features simply obtained from an image, such as pixel values and edges, are used for retrieval. Since such image features are obtained from any images, flexibility is high. However, since the gap between the image features and an object is large, a retrieval user is bothered with expressing his retrieval demands with such image features. For example, a retrieval user must specify "the mountain of a long distance" with the value (numerical value) of the feature values.

We have examined automating the index generation using the image recognition processing. Since an image recognition result shows what objects exist in an image, a retrieval user can retrieve images by specifying objects.

One of the problems of using image recognition for the automatic index generation is that the right result is not necessarily obtained because the image recognition ability by the present computer is considerably inferior to human beings. In the preceding chapters, we described the method to apply the erroneous image recognition results to image retrieval with refining the generated index and the retrieval method but it is not enough to practical use.

In this chapter, we propose a method using retrieval examples to improve the retrieval results. The retrieval example is the pair of the goal image a user want to retrieve and the object sketch the user has drawn to retrieve the goal image. It can be obtained at the situation when the user retrieves an image with our retrieval system. Using the retrieval examples, we can improve the retrieval results because they contain the information that the goal image in the retrieval example should be shown when an object sketch similar to the object sketch in the retrieval examples is given. We show two methods to use the retrieval examples and evaluate them through some experiments.

5.2 Index Generation by Image Recognition

The pixel-based object labeling method is used to generate the index images. By the pixel-based object labeling method, from the feature values of the color, the position and the texture obtained from every pixel, it judges in what object each pixel is contained, and an object label is added for each pixel (Figure 5.1). The feed-forward type neural network of three layers is used for conversion from the feature values to an object label. This neural network is trained to learn the relation of the feature values of a pixel and an object label by the back propagation algorithm.

10 kinds of followings objects which appears in an outdoor scene are chosen for labeling.

blue sky	cloudy sky and could	concrete load
green leaves, grasses	soil	dark shadow
far mountain	water surface	dead leaves
rock, rocky mountain		

The result of the pixel-based object labeling can be seen as an image each pixel of which has the object label of the spot. We call this image as a label image. Since it has the information what object is contained in each portion of an original image, the label image can be used as an index in image retrieval.

Although the size of an original image is 256x256 pixels, we do not need such high resolution for the retrieval method used in this thesis. We take sampling points every 4 pixels for the horizontal and vertical axis in an original image, and we get a label image with a size of 64x64 pixels. This image which has been labeled is called the index image.

5.3 Image Retrieval Using Object Sketch

We use the object sketch to specify the key to retrieve an image that a retrieval user want to get. The procedure of the image retrieval is as follows (Figure 5.2).

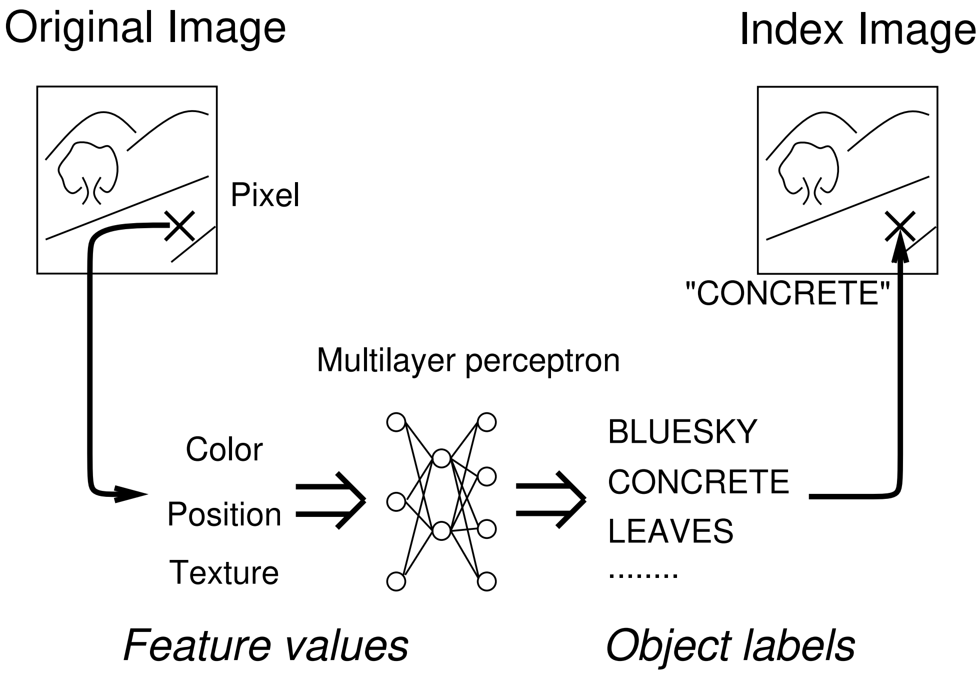


Figure 5.1: Pixel based object labeling method

1. For an original image, the image recognition processing of the previous section generates the corresponding index image.

2. A retrieval user visualizes an image (retrieval goal image) to search, and draws the object sketch of the retrieval goal image.

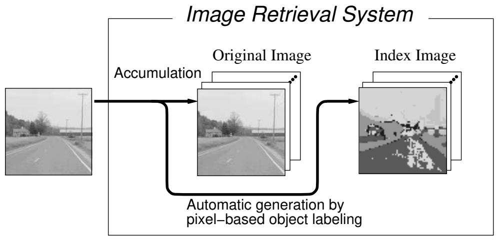
3. The object sketch is compared with the index image, and the similarity between them is calculated.

4. The retrieval system shows sequentially the original images corresponding to the index image in the order of similarity to the user.

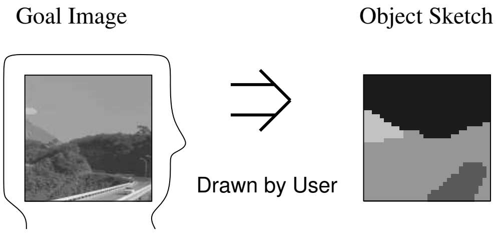
An object sketch is the image which specifies the kind and the position of an object in a retrieval goal image as retrieval conditions. Specification of this retrieval condition is performed by specifying the object label for every pixel in the way like drawing an image. A retrieval user does not have to specify an object label of some pixels. The size of an object sketch is the 64x64 which is same size as the index images.

The similarity between the object sketch and the index image is the rate of the same object labels of the pixel between the index image and the object sketch in the corresponding position. The degree of similarity is calculated by M/S, where S is the number of the pixels whose object labels are specified in the object sketch and M is the number of pixels whose object label are equal between the object sketch and the index image at the same position.

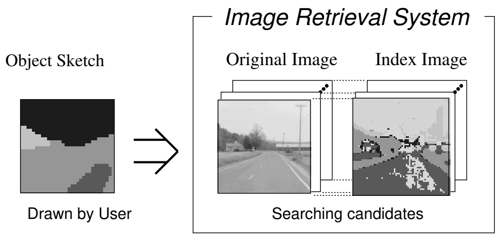
A retrieval result is shown sequentially from the original image in the order of similarity. With this method, since a large number of pixels is compared and the similarity is calculated independently, the partial error of labels in the index image does not significant on the result. Therefore, even if the recognition error is contained in the index image, the goal image can be shown as a candidate of higher rank. Moreover, in an object sketch, since the kind and the position of object within a goal image can be specified simultaneously, a retrieval user also has the advantage of being easy expressing his retrieval demand when the user's demand of "wanting such an image" is clear.



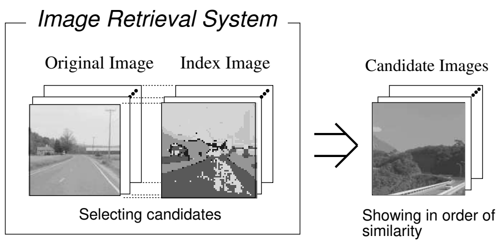
1. Registering image to image library



2. Drawing sketch as retrieval key



3. Searching similar image from the image library



4. Showing the result of retrieval

Figure 5.2: Retrieval method

5.4 Use of Retrieval Example

By this retrieval method, a candidate image is shown according to the similarity of an object sketch and an index image. Supposing a retrieval user draws an object sketch with him looking at a retrieval goal image, it can be considered that the object sketch is "the image by which object labeling was carried out correctly."

If both of the object sketch and the index image are labeled correctly, the retrieval goal image is shown as the first candidate in the retrieval results. But in some cases, the retrieval goal image does not be shown in the higher candidates because of the following reasons:

- Error in the user's subjective or error in the user's memory. Because a user draws an object sketch based on his memory.

- Error in the index image. Because the index image was generated by the object recognition by the computer which includes some errors.

A retrieval example is utilized in order to cope with them.

A retrieval example is the pair of the goal image which a retrieval user want to retrieve and the object sketch which the retrieval user has drawn to retrieve the goal image. A retrieval example is obtained whenever a user uses our retrieval system, if the retrieval system has a means to know that "this image is the retrieval goal image" from a retrieval user after showing candidate images. Since it is normal operation that a retrieval user checks the retrieval results at the time of use the result of image retrieval, a retrieval example is obtained without bothering the retrieval user so much.

The retrieval example expresses "show this (goal) image when such an object sketch like the object sketch of the retrieval example is given." If this information is utilized for subsequent retrieval, it will be enabled to show a retrieval goal image as a candidate of higher rank. We will show two methods to use the retrieval examples.

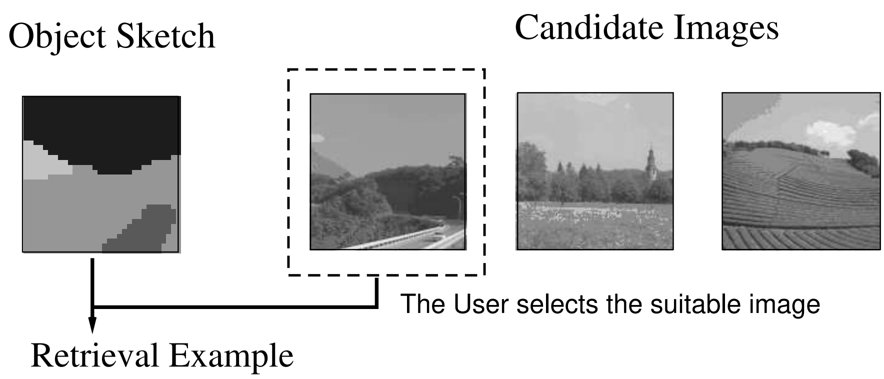


Figure 5.3: Retrieval example

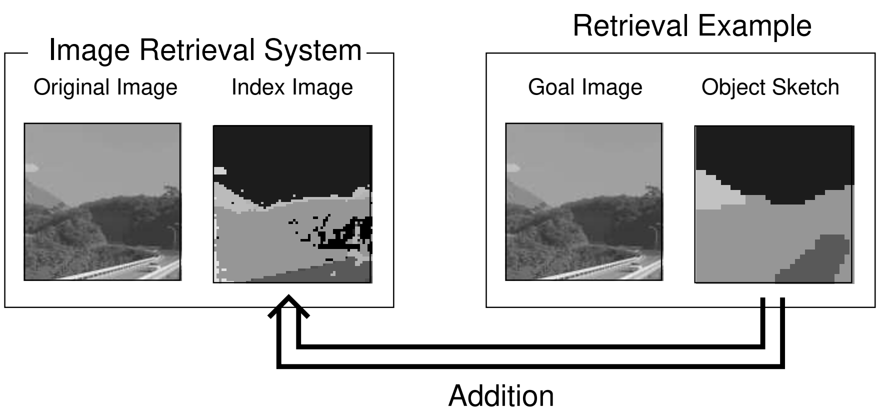


Figure 5.4: Adding index image

5.4.1 Method of Adding Indexes from Retrieval Examples

In our retrieval system, both of the object sketch drawn by a user and the index image generated by the computer are the label images each pixel of which has an object label. Then, a method to add the retrieval example to an index image can be considered (Figure 5.5).

At first, only one index image is generated for each original image in the image library by our image recognition method. A retrieval example is obtained whenever a user retrieves with our retrieval system. The object sketch in a retrieval example is added as an index image of the goal image. Therefore, one original image comes to have two or more index images. The similarity of an original image and an object sketch is defined as the highest similarity between the object sketch and the index images of the original image, and candidates are shown to the user in the order of the similarity of original images.

By adding retrieval examples as an index image, the error in the index image by the recognition error comes to seldom influence the retrieval results. Moreover, if the same tendency of mis-remembrance in different men is shown among the retrieval users, the error of the object sketch by mis-remembrance can also be coped with. Furthermore, original images which are frequently referred will be apt to be shown within higher candidates since they have many index images comparing with other images and the similarity is maximum value of the many index images.

5.4.2 Method of Re-training Using Retrieval Examples

By the method of Section 5.4.1, a possibility of being shown as a candidate of a higher rank increases by the original image to which the index image was added, compared with before adding the index image. However, there is no merit to other images which do not have an additional index image from the retrieval example. Furthermore, their ranking in the candidates may decrease if the added object sketch is accidentally similar to the index image.

In order to cope with this, it is required to the method that makes the information obtained from the retrieval example reflect to the all index images. Therefore, we propose another method to use the retrieval examples in which we perform re-training of the neural network for automatic index generation.

The retrieval example includes the information on the object label and its corresponding position in the original image. We can calculate the feature values of the pixel from the original image. Therefore, training data of a neural network which consists of a set of the feature values of a pixel and its object label can be generated from the retrieval examples. Since the rate of recognition generally improves so that many training data is used, if a neural network is re-trained using this training data, we expect that the recognition rate will be improved. If an index image is re-generated about all images using the re-trained neural network, the retrieval result is thought to be improved.

5.5 Experiments

5.5.1 Experimental Method

First, we showed a retrieval user one image in the image library for a while, and had him memorize it as a retrieval goal image. Next, he drew the object sketch without seeing the goal image. The retrieval system compares the object sketch with the index images and shows the candidate images in the order of similarity. We evaluate our method with the ranking that the goal image is shown in the candidates.

There are 461 images in the image library. The index image was generated to each image using the pixel-based object labeling method. The rate of recognition to non-training data of the neural network was 77.3%.

Retrieval experiments are performed using 2 kinds of data sets, namely set A and set B.

set A 14 users retrieve same 5 goal images. We obtain 14 object sketches for each of the 5 goal images. The retrieval examples total 70. Some samples of original images, index images and object sketches are shown in Figure 5.6.

set B 40 users retrieve one image at a time. All the goal images are different from the goal images in set A. We obtain 1 object sketch for each of the goal image in set B. The retrieval examples total 40.

Below, we evaluate our methods by the retrieval ratio. The retrieval ratio is the ratio among the retrieval trials that the goal image is shown within the top 1, top 5 or top 10 candidates. We retrieve with a sample data in the set A or set B while rest of the sample data are used as retrieval examples.

5.5.2 Evaluation of the Retrieval Method

The retrieval ratio only using the index images which are generated by the pixel-based object labeling method is shown in Table 5.1. In both cases of sets A and B, the goal image is shown within top 10 candidates in about 70% of the retrieval trials. In the trials in which the goal image could not be shown within top 10 candidates, the user specifies different object labels from those of the index image, such as "blue sky" to "cloudy sky" or "green leaves" to "dead leaves." The distinction criteria of such objects may be different among the users. Moreover, since some users did not distinguish these objects and memorized only as "sky" or "leaves." The difference of distinction criteria and the mis-remembrance in such objects also need to be coped with.

5.5.3 Effect of Adding Indexes from Retrieval Examples

Next, we conducted the experiment to evaluate the effect of adding the index images from the retrieval examples. The retrieval ratio is shown in Table 5.2. We retrieved images with the sample data in set A. The column of "Adding set A" shows the result when the rest of sample data in set A are added to the index images. In this case, the sample data used in retrieving is not used as the retrieval

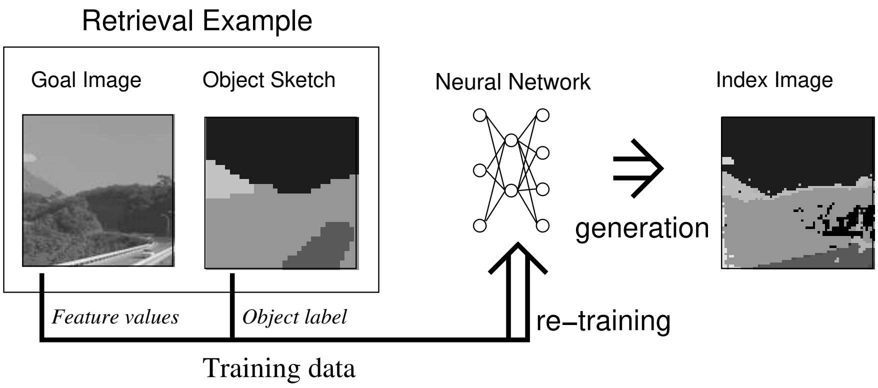


Figure 5.5: Re-training neural network

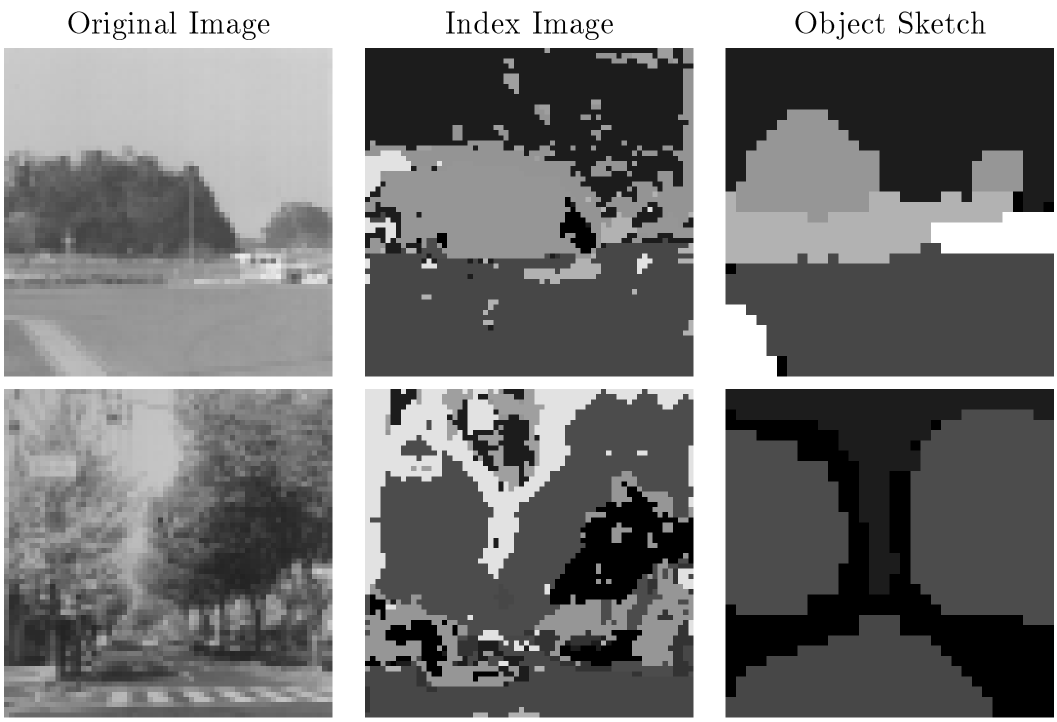


Figure 5.6: Examples of images

Table 5.1: Retrieval result 1

Without using retrieval examples

candidates	Retrieval ratio (%)	
	set A	set B
Top 1	34.3	15.0
Top 5	64.3	57.5
Top 10	72.9	72.5

Table 5.2: Retrieval result 2

Adding indexes from retrieval examples(set A)

candidates	Retrieval ratio(%)	
	Adding Set A	Adding Set B
Top 1	84.3	4.3
Top 5	100.0	51.4
Top 10	100.0	72.9

Table 5.3: Retrieval result 3

Re-training using retrieval examples (set A)

candidates	Retrieval ratio (%)	
	set A	set B
Top 1	40.0	15.0
Top 5	70.0	47.5
Top 10	74.3	65.0

example. The column of "Adding set B" shows the result when all the sample data in set B are added to the index images. Some samples of original images and added index images in set A is shown in Figure 5.7.

In the case of "Adding set A," the 5 kinds of original images in set A have the 14 or 15 index images. In this case, all the goal images in set A are shown within the top 5 candidates. This result shows that this method improves the retrieval results drastically for the images which have added index images.

In this experiment, a retrieval user memorizes a retrieval goal image while the goal image is shown for a short time. As the result, the user may have a vague memory of the goal image and draws an object sketch which contains some errors in object labels. Nevertheless, the retrieval results are improved greatly. It is because retrieval users tend to confuse object labels in the same way and to draw the similar object sketch to the same original image even if it contains some errors. In order to improve the retrieval results, it is not only necessary to generate the correct index image by automatic object labeling but also it is necessary to generate the index image reflecting such users' tendency. This method is effective on this point as well.

In the case of "Adding set B," the retrieval ratio is decreased, especially the retrieval ratio at the top 1 candidate. However, the retrieval ratio is not decreased within the top 10 candidates. It is thought that the influence by an index image being added to other images will not become a problem so much if the candidate of a certain amount of range (for example, the top 10 candidates) takes into consideration.

5.5.4 Effect of Re-training Using Retrieval Examples

Next, we conducted the experiment to evaluate the effect of re-training the neural network with the training data generated from the retrieval examples.

Training data are taken from the object sketches of the set A. We selected the training data each of which has the same object label through all the 14 object sketches in set A. As the result, we got new 108 training data from 3 of 5 kinds

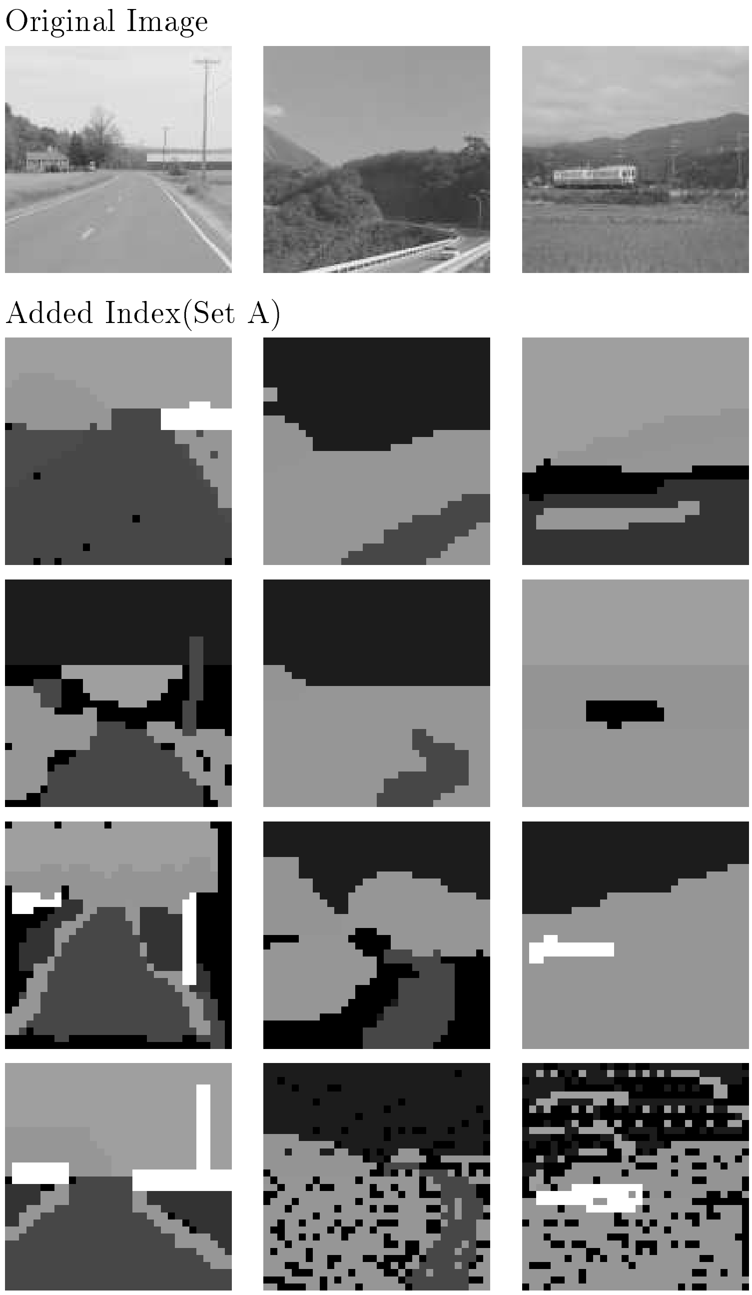


Figure 5.7: Added indexes

of original images. We merged the new training data to the original 970 training data and performed the re-training of the neural network using them with the back propagation algorithm. The recognition rate for non-training data after the re-training becomes 78.9% (the rate of recognition before re-training is 77.3%).

Using the re-trained neural network, we re-generated the index images to all the 461 original images in the image library. After that, we conducted the retrieval experiments using the re-generated index images. The retrieval result is shown in Table 5.3.

Since training data obtained from set A was added and re-training was performed, the retrieval ratio is improved about set A (refer to the column of "set A").

It has been expected that the retrieval ratio will improve also to set B since the recognition rate of the neural network was improved. Indeed, the retrieval ratio of set B at the top 1 candidate is equal to before the re-training in the experimental results (refer to the column of "set B"). Furthermore, the retrieval ratio within the top 5 or 10 candidates is decreased .

Even if the error of the object label in an index image decreases, the retrieval ratio is not necessarily improved because a retrieval user draws an object sketch which contains some errors by a mis-remembrance of the object in the goal image. Moreover, the re-training of neural network causes some object labels to be changed which have correct object labels before the re-training. If those object labels takes an important role in the retrieval, the retrieval ratio may decrease.

Here, the re-training of a neural network is used in order to cope with the error in the index image by the recognition error. In order to cope with the mis-remembrance of the object by the users, we should examine another method for extracting training data to adapt for the tendency of the mis-remembrance in retrieval users.

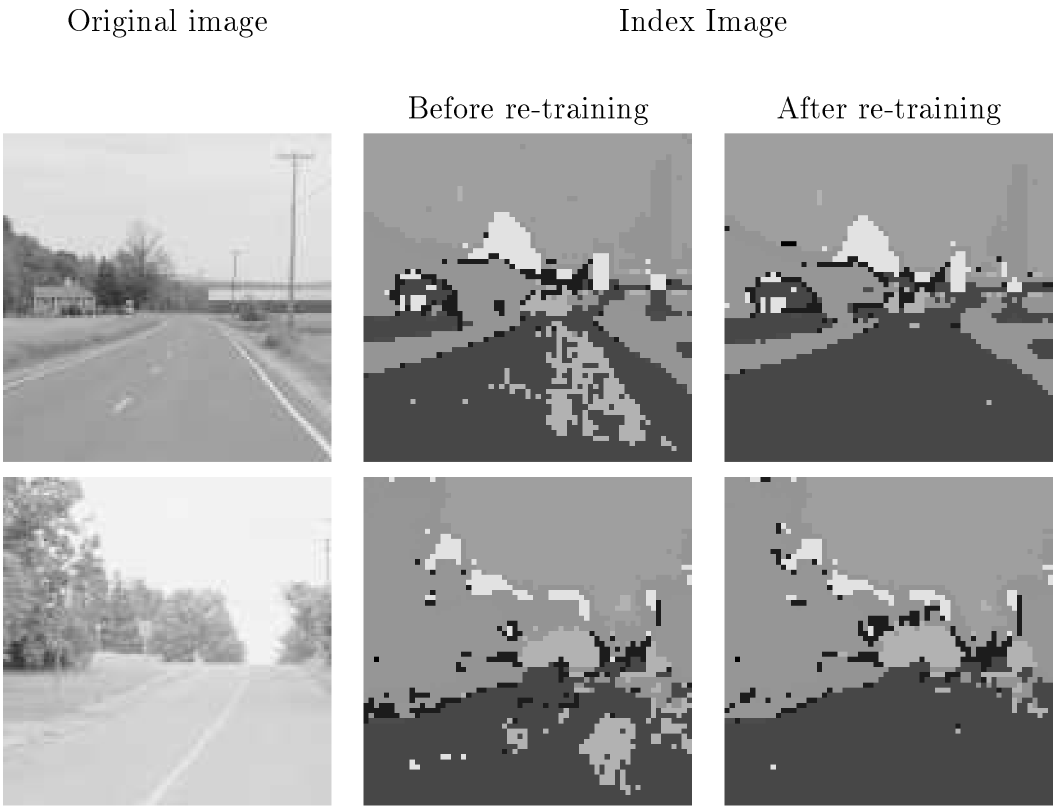


Figure 5.8: Results of re-training

5.6 Conclusion

In this chapter, we proposed two methods to use the retrieval examples to improve the retrieval results. The retrieval example is the pair of the goal image and the object sketch which can easily obtained from the practical retrieval situation. The retrieval examples include the information that the goal image should be shown when the object sketch is given.

The one method to use the retrieval examples was the method in which the object sketches in the retrieval examples are added as the index image of the original image. After adding the index image, the similarity is calculated as the maximum similarity among the plural index images. As the result, for the original images which has been added the index image, the retrieval results would be improved. The other is the method in which the training data is extracted from the retrieval examples, the neural network is re-trained using the extracted training data and the index images are re-generated. We selected the training data which is very likely correct ones from the retrieval examples. Using the extracted training data, the recognition rate will be improved and the retrieval ratio will also be improved.

The experimental results showed that the retrieval ratio can be improved greatly by the former method. In latter method, the retrieval ratio was also improved for some images but it was not improved in some cases although the recognition rate was improved and we can generate more exact index images. From these experiments, we showed that the retrieval examples are effective to improve the retrieval results because it can cope with the tendency of mis-remembrance of the objects by the user.

There are some problems in both methods. In the former method, the speed for retrieval will decrease and the necessary storage size will increase as the accumulated retrieval examples be increased. A method to compress the information in the retrieval examples is necessary. In the latter method, we cannot cope with the tendency of mis-remembrance of objects by the user because we extract the very correct training data. These problems are left as future works.

Chapter 6

Conclusion

In this thesis, we described the outdoor image retrieval method using automatically generated index.

In Chapter 2, we described our pixel-based object labeling method. At first, the feature values of each pixel are calculated. Then the object label is assigned to each pixel based on the object models which are the mappings between the feature values and the object labels. The object models are implemented with the 3-layered neural networks.

In our method, we do not generate initial regions by signal-level segmentation. Because the signal-level segmentation needs to adjust several parameters to fit the various original images, it prevents the automatic generation of the index. We can avoid this problem with our method. Furthermore, the object models are obtained by training. It becomes unnecessary that man analyzes the characteristics of the objects in detail and gives them to the computer. As the result, anyone can easily construct the object models and this recognition method can be used as a basic tool.

We conducted the experiments to compare our pixel-based method with traditional region-based methods. As the result, our method shows higher recognition rate than the region-based methods. The analysis of the relations between the feature values to be used and the recognition rate also shows that our pixel-based method is reasonable method for the recognition of the natural objects in outdoor scenes.

Next, we applied our pixel-based object labeling method to generate the index images for image retrieval. We proposed a pictorial query method for retrieval, in which a user draws the object sketch of the goal image, the retrieval system compares the object sketch with the index images and shows the candidate images in the order of similarity. The pictorial query method is robust against errors in the index images because it compares a large number of pixels independently.

We conducted the retrieval experiments with the index images generated by the pixel-based object labeling method. The retrieval ratio of top 5 candidates was 87.9% when the goal image was referred and 47.6% when the the goal image was not referred. The results shows that the recognition ratio of our pixel-based method is almost enough but we must cope with the variety of the object sketch drawn by the users.

In Chapter 3, we proposed a method to extract training data from images containing same kind of object. In our pixel-based object labeling method, we need a large number of training data which had to be given by hand. With our training-data auto-extraction method, we can save the cost to give such data. The setup of using "a large number of the images containing the same kind of object" is easily realizable by using image libraries each image of which has been indexed with the kind of object within the image as a key word.

The algorithm consists of 4 stages: the feature extraction stage, the projection to discriminant space stage, the clustering stage and the cluster selection stage. We used the constraint that "all images contain same kind of object," and proposed the methods to use the constraints in the processing.

The experimental results showed that, by combining the proposed methods, better training data could be obtained comparing with the normal method. The correctness of the generated training data was 64.9% which was better than the expected result.

In Chapter 4, the problem of the errors in the index images was discussed. When using the automatically extracted training data for pixel-based object labeling method, the errors in the recognition results become a problem. The recognition rate was 52.5% and the results contain many errors. To clarify the relation between the recognition rate and the retrieval ratio, we introduced two recognition error models. The one was the random error model and the other was the image similarity model.

According to the random error model, the necessary and sufficient condition for the recognition rate r was $r > 1/K$, where K was the number of the kind of the objects. It suggests that the random errors are very preferable recognition errors in our image retrieval method.

According to the image similarity model, the necessary condition for the recognition rate r was $r > S$, where S was the similarity between the original images. It suggests that the uniformity among the index images is not preferable nature in our image retrieval method.

Considering the results of the analysis, we proposed two methods to cope with the errors in the index images. The one was correcting the labeling results reflecting the tendency of recognition errors. The other was using the results obtained from several neural networks. Both method increased the randomness and decreased the uniformity in the index images.

We applied these two methods and conducted the retrieval experiments. The results showed that the both methods improve the retrieval results.

In Chapter 5, we described the method to improve the retrieval results using the retrieval examples. The retrieval example is the pair of the goal image and the object sketch and it can obtained whenever a user retrieves an image with our retrieval system. We proposed two methods to use the retrieval examples.

One was a method to add the retrieval examples into the index image. The retrieval examples include the information that the goal image should be shown when the object sketch is given. As the result, the retrieval ratio would be improved.

The other was a method to perform re-training using the training data extracted from the retrieval examples. Using the extracted training data, the recognition rate would be improved and the retrieval ratio would also be improved.

In the experimental results, the retrieval ratio was greatly improved by the former method and that the recognition rate was improved by the latter method. It means that we can use the retrieval examples to improve the retrieval results.

There are still several open problems in our image retrieval system.

- Improving retrieval results furthermore
- Adapting to the users with retrieval examples
- Applying our system to the practical usage
- Evaluating our system with a large scale image library

These problems are left as future works.

Bibliography

[1] M.Flickner, H..Sawhney, W.Niblack et.al.: Query by Image and Video Content: The QBIC system, IEEE Computer, pp.23-32, (1995-9).

[2] Y.Takahashi, N.Shima, F.Kishino: An Image Retrieval Method Using Inquiries on Spatial Relationshps(in Japanese), Trans. IPSJ vol.331, no.11, pp.1636-1643(1990-11).

[3] T.Kurita, T.Kato, I.Fukuda, A.Sakakura: Sense Retrieval on a Image Database of Full Color Paintings, Trans. IPSJ vol.33, no.11, pp.1373-1383(1992-11).

[4] T.Kurita, H.Shimogaki, T.Kato: A Personal Interface for Similarity Retrieval on an Image Database System, Trans. IPSJ vol.31, no.2, pp.227–237(1990-2).

[5] T.Kato, H.Shimogaki, K.Fujimura: TRADEMARK: Multimedia Image Database System with Intelligent Human Interface, Trans. IPSJ vol.J72-D-II, no.4, pp.535–544(1989-4).

[6] C.C.Chang, S.Y.Lee: Retrieval of Similar Pictures on Pictorial Database, Pattern Recognition, vol.24, no.7, pp.675–680(1991).

[7] H.Nishiyama, Y.Matsushita: An Image Retrieval System Considering Image Composition, Trans. IPSJ vol.37, no.1, pp.101–109(1996-1).

[8] M.Minoh, H.Okazaki, T.Sakai: Image retrieval with attribute feature of object(in Japanese), Trans. IPSJ vol.32 no.4, pp.513–522 (1991-4).

[9] H.Mori, M.Minoh, K.Ikeda: Retrieval of multimedia document image with sketch(in Japanese), 45th Annual Conv IPSJ 2S-1.

[10] T. Kato et. al.: A Sketch Retrieval Method for Full Color Image Database(in Japanese), 11th ICPR pp.530–533 (1992).

[11] Yu-ichi Ohta: A Region-Oriented Image-Analysis System by Computer, Doctor thesis of Kyoto Univ.(1980-3).

[12] T.M.Strat, M.A. Fischler: Context-Based Vision: Recognizing Objects Using Information from Both 2-D and 3-D Imagery, IEEE Trans. on PAMI., vol. 13, no. 10 (1991-10).

[13] Shinichi Hirata, Yoshiaki Shirai, Minoru Asada: Scene interpretation using 3-D information extracted from monocular color images (in Japanese), Trans. IEICE vol. J75-D-II, no. 11, pp. 1839-1847 (1992-11).

[14] Hidehiro Ohki, Tsutomu Endo: Scene understanding system using contextual information – Hypothesis generation and conflict resolution – (in Japanese), Technical report of IEICE PRU92-43 (1992-10).

[15] J.M.Tenenbaum, H.G.Barrow: IGS: A Paradigm for Integrating Image Segmentation and Interpretation, Proc. 3rd IJCPR, pp.504-513(1976).

[16] J.W.Modestino, J.Zhang: A Markov Random Field Model-Based Approach to Image Interpretation, Trans. PAMI vol.14, no.6, pp.606-615 (1992-6).

[17] H.Q.Lu, J.K.Aggarwal: Applying Perceptual Organization to the Detection of Man-Made Objects in Non-Urban Scenes, Pattarn Recognition vol.25, no.8, pp.835-853(1992-8).

[18] D.Geiger, A.YUILLE: A Common Framework for Image Segmentation, Int Journal of Computer Vision, vol.6, no. 3, pp. 227-243(1991).

[19] N.R.pal, S.K.Pal: A Review on Image Segmentation Techniques, Pattern Recognition, vol.26, no.9 pp.1277-1294(1993).

[20] T.Yoshizawa, M.Sone, M.Takagi: Automatic Classification for Texture Using Fractal Dimention and Lower Order Statistics, Trans. IPSJ, vol.31, no.7, pp.1027-1037(1990-6).

[21] T.N.Pappas, N.S.Jayant: An Adaptive Clustering Algorithm for Image Segmentation, Proc. 2nd ICCV pp.310-315(1989).

[22] T.Kato, T.Kurita, N.Otsu, K.Hirata: A Sketch Retrieval Method for Full Color Image Database – Query by Visual Example –, 11th ICPR, pp. 530-533 (1992).

[23] M.Kurokawa, J.K.Hong: An Approach to Similarity Retrieval of Images by Using Shape Features (in Japanese), Trans. IPSJ, vol. 32, no. 6, pp. 721-729 (1991-6).

[24] M.Sakauchi: Image retrieval technology(in Japanese), Journal of IEICE, vol. 71, no.9, pp.911–914 (1988-9).

[25] R.M.Haralick: Statistical and structural approaches to texture, Proc. of the 4th IJCPR, pp. 45-69 (1978).

[26] T.Kurita: A Method to Determine the Number of Hidden Units of Three Layered Neural Networks by Information Criteria, Trans. IEICE vol.J73-D-II, no.11, pp.1872-1878(1990-11).

[27] Y.Takei, H.Sawada, K.Taketani, A study on the recginition mechanism of three layered neural networks(in Japanese), Tech. IEICE NC89-57 (1989).

[28] I.Keshi, H.Ikeuchi, R.Tanaka, M.Osaki: Multimedia Information Retrieval Using Knowledge in Encyclopedia Texts, Technical report of Sharp, vol.60 (1994-12).

[29] H.Ogawa, K.Yamazaki: The theory of Over-Lerning, IEICE vol.J76-D-II, no.6, pp.1280–1288(1993-6).

[30] S.Numakura, E.Tanaka, H.Aoki, et. al.: An Information Retrieval System Accessible by Keys with Errors, Proc. IPSJ vol.30, no.11, pp.1468–1478(1989-11).

[31] B.Dubuisson, M.Masson: A Statistical Decision Rule with Incomplete knowledge about Classes, Pattern Recognition vol.26, no.1, pp.155–166(1993).

[32] Kah-Kay Sung: Learning and Example Selection for Object and Pattern Detection, MIT A.I.T.R. no. 1592 (1996-1).

[33] A.Nakajima, A.Hirabayashi, H.Ogawa: Error correcting memorization learning for noisy training data, Tech. IEICE NC97-51 (1997-11).

[34] K.Yamauchi, N.Ishii: An incremental learning method for GRBF with retrieving of interfered patterns, Tech. IEICE NC97-137 (1998-3).

[35] S.Nakagawa, Y.Ono: Estimation of a posteriori probability and probability density function using neural networks and recognition of vowels, Trans. IEICE, vol.j76-D-II, no.6, pp.1081–1089(1993-6).

[36] Q.Jia, N.Toda, S.Usui: A Study on Initial Value Setting of the Back-Propagation Learning Algorithm, Trans. IEICE vol.J73-D-II, no.8, pp.1179–1185(1990-8).

[37] Y.Kobayashi: Case-Based Reasoning, Journal of Artificial Intelligence, vol.8, no.1, pp.26-36(1993).

Acknowledgments

I would like to express sincere gratitude to Professor IKEDA Katsuo of Kyoto University. He gave me the opportunity of this research, and has been giving me continuous guidance, interesting suggestions, intensive criticism and encouragement during this research. I would also like to appreciate Professor MINOH Michihiko of Kyoto University for his invaluable suggestions, accurate criticism and encouragement. I can't write this thesis without him.

I would like to devote my special thanks to Mr. TAKAYA Izuru, he began the study of image retrieval when he belonged to our laboratory. I took over his study and it bears fruit as this thesis. Special thanks to Mr. MAEDA Shigenori and Mr. MATSUMOTO Yasuaki in our laboratory, who work together on my topics in different approach. I consult the results and enhance them in this thesis.

Thanks are also due to all members of Professor IKEDA's Laboratory for their discussions on my work and cooperation to the experiments through this research.

List of Publications by the Author

Major Publications

1. MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: A retrieval method of outdoor scenes using object sketch and an automatic index generation method, IEICE Trans. D-II, vol.J79-D-II, no. 6, pp. 1025–1033, 1996-06(in Japanese).

2. MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: Retrieval of Images using Pixel-based Object Models, Advances in Intelligent Computing – IPMU'94, Lecture Notes in Computer Science 945, pp.557-566, 1995, Springer.

3. MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: Pixel-based object labeling method of out-door scenes, Asian Conference on Computer Vision, pp.470-473, 1993-11.

4. MAEDA Shigenori, MUKUNOKI Masayuki and IKEDA Katsuo: A Classification Method of Image Based on Composition and Its Application to Image Retrieval, ICMCS'99, 1999-06(to be published).

5. MAEDA Shigenori, MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: An Interactive Image Retrieval System with Self-Explanation Function, IEICE Trans. D-II (Submitting).

Technical Reports

1. MATSUMOTO Yasuaki, MUKUNOKI Masayuki and IKEDA Katsuo: An object identification method based on the searching of common subregions, MIRU'98 II-77–II-82, 1998-07(in Japanese).

2. MAEDA Shigenori, MUKUNOKI Masayuki and IKEDA Katsuo: Image representation supporting system communicating with its user by images, Proc. 3rd IIMEDIA symposium, 309-316, 1997-12(in Japanese).

3. SAKIYAMA Takuro, MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: A recording system of lectures based on events in lectures, Proc. 3rd IIMEDIA symposium, 149-156, 1997-12(in Japanese).

4. IMAO Koji, MUKUNOKI Masayuki and IKEDA Katsuo: Improvement of motion perception with motion blur in scroll screen, Technical Report of IEICE, PRMU97-149, 1997-11(in Japanese).

5. Prospero C.NAVAL Jr., Masayuki MUKUNOKI, Michihiko MINOH and Katsuo IKEDA: Estimating Camera Position and Orientation from Geographical Map and Mountain Image, 38th technical report SICE, 1997-01.

6. MAEDA Shigenori, MUKUNOKI Masayuki and IKEDA Katsuo: An interactive image retrieving system that explains itself, Proc. 2nd IIMEDIA symposium, 99-106, 1996-12(in Japanese).

7. MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: Object recognition in sunny and shade environment, 33th technical report SICE, 1995-11(in Japanese).

8. YAMADA Ryo, MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: Human head tracking with contour model, 149th technical report IEEJ, 1995(in Japanese).

9. MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: Outdoor scene retrieval using the result of image recognition as the index, 141th technical report IEEJ, 1993-09(in Japanese).

10. MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: A Study of the relation between Object recognition and Image Segmentation, MIRU'92, pp. 1-17 – 1-24, 1992-07(in Japanese).

Convention Records

1. TANIMOTO Masato, MUKUNOKI Masayuki and IKEDA Katsuo: A Motion Recognition Method for the Cut which Contains Multiple Motions, 1999 IEICE General Conference, D-12-188, 1999-03(in Japanese).

2. MURATAA Yasuhiro, NISHIGUTI Satosi, MUKUNOKI Masayuki and IKEDA Katsuo: Motion Algorithm for Camera-man Robot based on Estimation and Verification, 1999 IEICE General Conference, D-12-116, 1999-03(in Japanese).

3. OHONO Naoki, SAKIYAMA Takurou, MUKUNOKI Masayuki and IKEDA Katsuo: Video Stream Selection according to Lecture Context on Remote Lecture, 1999 IEICE General Conference, D-15-49, 1999-03(in Japanese).

4. SAKIYAMA Takurou, MUKUNOKI Masayuki and IKEDA Katsuo et. al.: Editing Video Stream of Blackboard for Archiving, Proc. 58th Annual Convention IPSJ, 4U-2, 3-181, 1999-03(in Japanese).

5. MUKUNOKI Masayuki, SATOH Tetsuo and IKEDA Katsuo: Automatic lecture recording system using recognition of lecturer and blackboard stae, Proc. 1997 Information and Systems Society Conference, D-12-74, 1997-09(in Japanese).

6. MAEDA Shigenori, MUKUNOKI Masayuki and IKEDA Katsuo: A Study on Image Representation Supporting System, 1997 IEICE Society Conference, A-15-7, 1997-09(in Japanese).

7. IMAO Koji, MUKUNOKI Masayuki and IKEDA Katsuo: A dynamic cloth shape simulation based on motion quantization, 1997 IEICE Society Conference, D-12-65, 1997-09(in Japanese).

8. IMAO Koji, MUKUNOKI Masayuki and IKEDA Katsuo: Estimation of motion direction from motion blur in image sequence, 1997 IEICE General Conference, D-12-207, 1997-03(in Japanese).

9. MUKUNOKI Masayuki and IKEDA Katsuo: Pixel based object labeling of outdoor scene using general purpose DSP, 1997 IEICE General Conference, D-11-134, 1997-03(in Japanese).

10. MAEDA Shigenori, MUKUNOKI Masayuki and IKEDA Katsuo: Image Retrieval by Selecting Composition Category, 1997 IEICE General Conference, D-11-123, 1997-03(in Japanese).

11. MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: Another candidate selection method for outdoor image retrieval system using object sketch, Proc. 53th Annual Convention IPSJ, 2G-3, 3-465, 1996-09(in Japanese).

12. MAEDA Shigenori, MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: An image retrieving system reading user's intention adequately, IEICE National Convention Record of Fundamentals, A-173, 1995-09(in Japanese).

13. MUKUNOKI Masayuki, NAGANO Maki, MINOH Michihiko and IKEDA Katsuo: Analysis of Outdoor Scenes using Image Generation Model, IEICE National Convention Record of Information and Systems, D-344, 1994-09(in Japanese).

14. MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: Outdoor scen retrieval using object sketch, 22th Annual Convention IEEJ, 1994-06(in Japanese).

15. MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: A study of the relations between evaluation value and degree of reliance in image understanding, Proc. 47th Annual Convention IPSJ, 1K-6, 1993-10(in Japanese).

16. SHIMAZU Shigeaki, MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: Extraction of object region with occlusion–an example of constraint satisfication image processing–, 22th Annual Convention IEEJ, 1993-06(in Japanese).

17. MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: Evaluation of effective features to create object models using neural networks, IEICE National Convention Record of Information and Systems, SD-11-5, 1992-09(in Japanese).

18. MUKUNOKI Masayuki, MINOH Michihiko and IKEDA Katsuo: A study of methods to deal with the ambiguity in the image undastanding, Proc. 43th Annual Convention IPSJ, 2-393, 1991-10(in Japanese).