Pictures at an Exhibition

Hyukjoon Kwon, and Jongduk Baek
Stanford University, Stanford, CA 94305
Email: {hjkwon, bjd1219}@stanford.edu

***Abstract*— This project is to recognize an input image through the database which are learned from a set of training images. The first thing we have done is to extract only the main image. This step requires an adaptive threshold to decide whether the selected region is a painting we want or not. The morphological operation follows the filtering process. The edges of the painting extracted in the source image are dilated and eroded. As a result, the four corner points of centered images are searched by using a radon transform. Inverse projective spatial transform leads us to obtain eigenimages, which enable to form a database. The test input is calculated the correlation values with eigenimage in database. The result shows that learning process with three images recognize the test input with one hundred probability.**

I. Introduction

Imagine you are in the Canon Art Center, while seeing one of European classic paintings. If you are not an expert in the area of European paintings, you might feel you need some help to explain a painting you are seeing now. The basic idea of an electronic museum guide is derived from the need of people who wants to appreciate paintings deeply. This is one example of typical applications called "augmented reality" applications which has been researched in computer vision.

The purpose of this project is to develop a core part of the technique used in this computer vision application, which is the recognition of paintings taken by a camera installed in the device. We are given a set of training images that we should recognize. An example image set is figure 1. Each image is not guaranteed to be taken properly. Therefore, some images can have an extra part of other images or a decoration of museum and other images can be taken with an askew angle. In the procedure of recognition, those impediments are factors to make it difficult to recognize perfectly.

Fig. 1. The Example of Training Images

The algorithm we are using in this paper is to extract a centered image, which we want to recognize, by an image mask using an adaptive threshold for each image and to distinguish one image from others by comparing correlation values calculated from an eigenimage method. The reason why the adaptive threshold is need is that one simple criterion does not satisfy all of images' conditions. Some of them are exposed into a strong intensity of light. Others are not. In addition, some paintings have distinct colors against walls or frames, but others do not. Hence, the pre-process step with an adaptive threshold is required in order to extract a proper eigenimage.

The outline of this paper is as follows. Section II provides an algorithm used for segmentation based on color distribution. In section III. we perform the morphological processing to obtain the corner points which become input points for spatial transform used in section IV. Section V shows the eigenimages extracted from a set of training images and compares the correlation among training images. Finally, conclusions are given in Section VI.

II. Color Based Segmentation

In order to recognize the centered painting, what we should do first is to distinguish a painting from a non-painting. This process can be approached with two ways. The first is to extract the frame color based on the fact that all the images are enclosed with an wooden frame. They are almost in an range of green color. The second is to select the color of wall as a criteria to distinguish an image from a non-image.

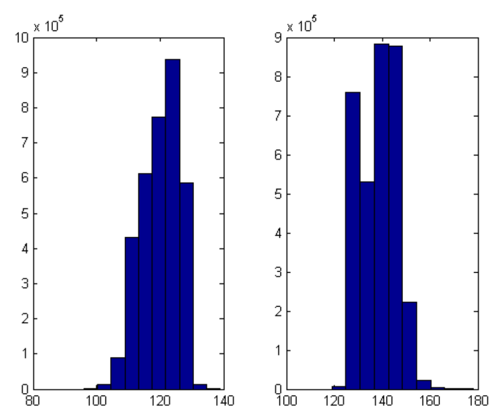

Fig. 2. The CbCr of Frame Color

In pursuing the goal, we have to decide which color format is used. Typically, the input color image is in the RGB format. However, RGB components are heavily dependent on the light intensity conditions. Therefore, the same painting can be recognized as a different image depending on the light location and light intensity. So, this project decides to use YCbCr component. In the YCbCr format, the luminance component Y can be seperated with other chrominance components, Cb and Cr. Therefore, the reference colors of paintings are chosen in a format of Cb and Cr.

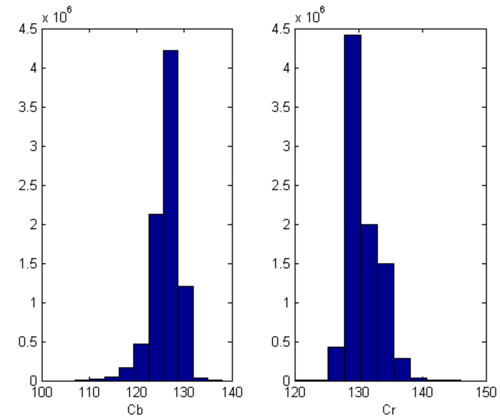

Fig. 3. The CbCr of Wall Color

As investigating the color distribution of frames (figure 2) and walls (figure 3), we conclude that the second method mentioned above is more proper to distinguish the paintings with non-paintings.

The wall histogram shows that the means of Cb, Cr are 125.7468, 131.2772 and the standard deviations of Cb, Cr are 1.5714, 1.9351. On the other hand, the frame has a distribution with larger standard deviations of Cb, Cr, which are 4.0748 and 5.4118. Therefore, it is reasonable to select a criteria having less variation.

Fig. 4. The CbCr of Wall Color

This color segmentation technique has been applied to a set of training images. As a result, the mask binary images can be obtained in figure 4 The leftmost columns show the mask in the first step of the procedure. The mask region is chosen with Cb, Cr values within a mean $\pm$ default-range-factor $\times$ standard deviation.

However, every image does not guarantee the proper mask in the first step. Therefore we consider the adaptive algorithm which decides the range of mask region based on the selected region in the previous step. The rightmost column is the mask in the final step of an adaptive selection.

Table 1. Adaptive Region Selection Algorithm

Initialization:

choose default range factor based on its average
min-cb = mean-cb − std-cb ∗ range-factor
max-cb = mean-cb + std-cb ∗ range-factor
min-cb < Cb < max-cb
min-cr = mean-cr − std-cr ∗ range-factor
max-cr = mean-cr + std-cr ∗ range-factor
min-cr < Cr < max-cr

Recursion:

1. the region around the centroid of frame is filled less than a threshold
2. the convex hull of selected region to a bounding box of the region has a smaller ratio than a threshold
3. the selected pixel count is less than a threshold

Repeat:

change the range-factor with delta and repeat the procedure

The bounding box area of the selected mask is applied to a set of training images. In the middle of the procedure, we can confirm that the centered image is extracted excluding the side painting or the obstruction.

Fig. 5. The Extracted Centered Painting

III. Morphological Processing

A. Edge Detection

The next thing we have to do is to transform an extracted centered image to a fixed, normalized rectangular image, which enables us to calculate an eigenimage correlation. In order to do a spatial transform, we first search four corner points of an centered image. The edge detection is the first step to find corner points, which is seen in figure 6. The edge detection is widely researched in the image processing area and the *Canny Crose Edge Detection Algorithm* is used to compute a spatial gradient crossing points on an image.

Fig. 6. The Detected Edge Image

B. Close Operation and Corner Detection

It is easily shown that the corner points are on the outmost edge. Therefore, it can be easy to detect the corner points through radon transform if only outmost lines remains in the edge image. The morphological technique can be used here to remove ripples in the edge image, which can accelerate the detection speed. Hence, the close (dilation → erode) operation is performed using a 20×20 window of all 1s. Before taking the effect on this close operation, one thing we should be careful is the relation between the image size and mask size. If the mask size is so large that it is close to the image size, then the close operation effect may cause an unexpected effect. Therefore, we put an padding region around the edge image to do a safe close operation.

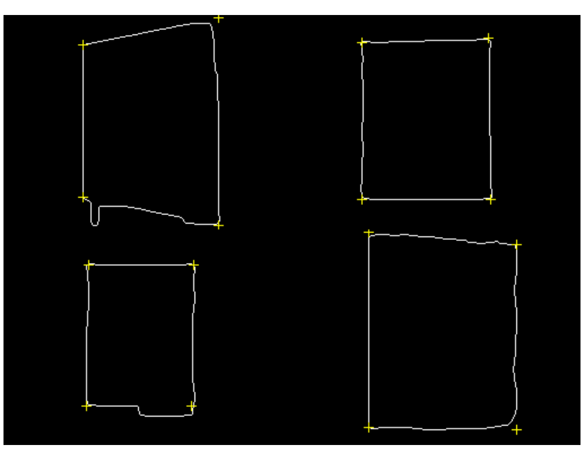

Fig. 7. The Detected Corner Points

Once we performed the close operation, it is easy to extract the corner points. A radon transform provides four peak values of angles and distances in the image. Naturally, they are the outmost lines of the corner points. As an example, the detection algorithm we used can be confirmed with the figure 7

Fig. 8. Inversely Projective Transformed Images

Fig. 9. The EigenImages of Transformed Images

IV. SPATIAL TRANSFORM

Now, we are ready to pile the database up with a set of training images. As we think of the fact that the original training images are taken by one focal point of a camera, we can conclude that the given images are the projective image of the real form in the museum. Therefore, we should take the inverse operation to turn the image back to the original shape. This is a projective spatial transform operation with the degree of 8. The inversely projective transformed images can be seen in figure 8. At the first and the last image, we can check that the slanted line is transformed to the parallel line. The second and third images are also transformed well with proper border lines.

Table 2. Searching Time for Each Training Image

No.	Result	Time	No.	Result	Time	No.	Result	Time
01.a	ok	5.063860	01.b	ok	2.168617	01.c	ok	2.099825
02.a	ok	2.510682	02.b	ok	2.458324	02.c	ok	2.377888
03.a	ok	2.697697	03.b	ok	2.461127	03.c	ok	2.622064
04.a	ok	2.333901	04.b	ok	2.660730	04.c	ok	2.098137
05.a	ok	2.360443	05.b	ok	2.242206	05.c	ok	2.237298
06.a	ok	2.445121	06.b	ok	2.172963	06.c	ok	2.708973
07.a	ok	2.134353	07.b	ok	2.222783	07.c	ok	2.084228
08.a	ok	2.591884	08.b	ok	2.508370	08.c	ok	2.494805
09.a	ok	2.350186	09.b	ok	2.294177	09.c	ok	2.383433
10.a	ok	2.457973	10.b	ok	2.591049	10.c	ok	2.719295
11.a	ok	2.499537	11.b	ok	2.456798	11.c	ok	2.501543
12.a	ok	2.115131	12.b	ok	2.104534	12.c	ok	2.056880
13.a	ok	2.407671	13.b	ok	2.505462	13.c	ok	2.442025
14.a	ok	2.423381	14.b	ok	2.473123	14.c	ok	2.681484
15.a	ok	2.131983	15.b	ok	2.122407	15.c	ok	2.143328
16.a	ok	2.121538	16.b	ok	2.302960	16.c	ok	2.130828
17.a	ok	2.127195	17.b	ok	2.102743	17.c	ok	2.098812
18.a	ok	2.257966	18.b	ok	2.334964	18.c	ok	2.433371
19.a	ok	2.117747	19.b	ok	2.087026	19.c	ok	2.109440
20.a	ok	2.146096	20.b	ok	2.163506	20.c	ok	2.131405
21.a	ok	2.332659	21.b	ok	2.220290	21.c	ok	2.109949
22.a	ok	2.278450	22.b	ok	2.182788	22.c	ok	2.150352
23.a	ok	2.372906	23.b	ok	2.408093	23.c	ok	2.409888
24.a	ok	2.276475	24.b	ok	2.411306	24.c	ok	2.322968
25.a	ok	2.691353	25.b	ok	2.727329	25.c	ok	2.668287
26.a	ok	2.079863	26.b	ok	2.274861	26.c	ok	2.337464
27.a	ok	2.128169	27.b	ok	2.420184	27.c	ok	2.263649
28.a	ok	2.686970	28.b	ok	2.659437	28.c	ok	2.605331
29.a	ok	2.077612	29.b	ok	2.075374	29.c	ok	2.380468
30.a	ok	2.536416	30.b	ok	2.556417	30.c	ok	2.505491
31.a	ok	2.720020	31.b	ok	2.720192	31.c	ok	2.755112
32.a	ok	2.510108	32.b	ok	2.502194	32.c	ok	2.489938
33.a	ok	2.316599	33.b	ok	2.263352	33.c	ok	2.268059

V. Image Matching

A. EigenImage

In order to increase the matching probability, we performed the denoise filter and deblurring filter. Wiener filtering scheme is used to remove a noise in the image, which helps classify the low values. In addition, deconvlucy filter has an effect on making distinctly each line in the image and boosting a high frequency region. It could increase the distance between the columnized image vectors. Then, using Sirovich and Kriby method, a set of eigenimages is generated for each theme image. Since the total number of training images are 33×3, we can obtain 99 eigenimages, among which 4×3 images are seen in figure 9. The information that an eigenimage has depends on the eigenvalue which means the energy of images.

B. Correlation and Result

A test input image is one of the given training images. By the matching method with training images, we can simulate the affectivity of our recognition algorithm. Table 2. show the result of the recognition ratio and the processing time. In order to reduce the processing time, we use eigenimage database as GLOBAL and PERSISTENT variables, which are piled in memory stack and remove the unnecessary loading time. To exploit all of three eigenimage, the distance metric is decided as a sum of squared correlation with each eigenimage. Then, we search the image in the closest distance, which will be the image we want to recognize.

VI. Conclusion

In color-based segmentation, we applied several ways to obtain a proper mask to extract a centered image. However, some method can be applied to one image and cannot be to the other image. The proposed adaptive masking method provides us a proper mask in all cases of the training images and produces a binary mask to detect the corner points. If we only focus the speed of the recognition within a finite small set, then we may be able to consider more specific method.

However, we are trying to find an algorithm which can be used as generally as possible.

In table 2, the first image takes more time to detect, because the loading time for database is necessary to start the recognition program. In case that three training images can be used in learning process for each image, the result in table 2 shows the perfect recognition can be possible. We also assumed that the number of training images can be reduces. If only two images can be learned for each image, then out algorithm shows 97% (96/99) recognition success ratio. In addition, for the case that only one image is used for recognition process, it also shows 97% (192/198) recognition success ratio. As an improvement way to increase the recognition success ratio, we can apply a fisher linear discriminant analysis (LDA), instead of eigenimage method.

ACKNOWLEDGMENT

We would like to thank Dr. Bernd Girod and Aditya Mavlankar for excellent instruction throughout the quarter. We both enjoyed the material considerably and are sure this course could be helpful for understanding the image property. Thank you very much.

REFERENCES

[1] R. Gonzalez and R. Woods, *Digital Image Processing - Second Edition*, Prentice Hall, 2002.
[2] B. Girod, *Lecture Notes for EE368*, Spring, 2006.