An Interface for Input The Object Region Using the Hand Chroma Key

Shuhei Sato, Etsuya Shibayama, and Shin Takahashi

Tokyo Institute of Technology, Dept. of information science,
2-12-1 Ohokayama Meguro-Ku, Tokyo 152-8550, Japan
{satou, etsuya, shin}@is.titech.ac.jp
http://www.is.titech.ac.jp/index-e.html

Abstract. We are developing the mobile system to identify wild flowers and grasses mainly from object images captured by a camera in the outdoor scene. In such systems, it is essential to inform the system an object region that the user interested in by some way. For example, if the captured image contains multiple leaves, the system can not determine which is the target leaf. In this paper, we propose interactive technique to inform the object region by placing hand behind the object. The system detects that situation, and extracts the object region automatically. Using this technique, a user can inform the system the object region in the interactive response time. Furthermore this input way is considered as natural because we often do this action to watch the object closely.

1 Introduction

We are developing the system to identify the wildflower mainly from it's flower and leaf images captured by a camera. The system consists of a handheld computer and a camera. (Fig. 1). The user interactively shoots the wildflower's flower part or leaf, and the system informs the user the wildflower's information, e.g. the flower's name.

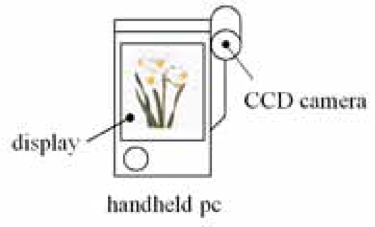

Fig. 1. The mobile system components

In such systems, it is necessary to extract the object region, e.g. flower region, from the shot image to remove the influence of background. For instance, the flower color information must be obtained from the image of the flower region. However, it is difficult to obtain such regions fully automatically. For example it is difficult to segment a leaf from the other similar leaves shown in Fig. 2. The contour of the leaf in the center of the image seems to be difficult to identify even for a human. It is difficult for the system to extract the contour by image processing robustly from the limited information that the object is in the center.

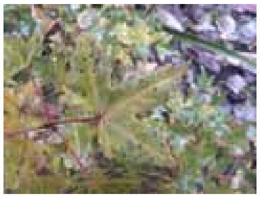

Fig. 2. An obscure leaf

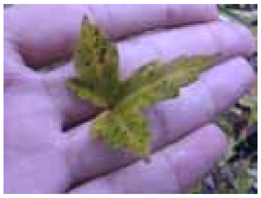

Fig. 3. a leaf on a hand

In this paper, we propose a method to specify the region of interest in the field with a mobile PC by placing a hand behind the object like in Fig. 3. In the field, the color of the image captured by camera is changes a lot even the same scene under various illumination.

Though the leaf in Fig. 3 is the same as the leaf in Fig. 2, the contour of the leaf is more conspicuous than that in Fig. 2. Using this method, the system can extract the object region semi-automatically in the interactive response time and the user can specify the object region expressly.

2 Interface for input object regions using a hand

We propose a method that the user places a hand behind the object to input the region of the object. The region with background of a hand is the object region. The system captures images continuously and if the system detects an object is on a hand, the region of the object is automatically extracted. The user does not need to press the button to input the region. Putting an object on the palm of the hand is a natural way when we watch the object closely. This system needs no input device other than a camera essentially. Consequently the user can concentrate in putting the target in a frame of a camera.

Detecting the hand region is necessary before extracting the object region. The skin color extraction technique is effective because the skin color is distributed in the small portion of the whole color space. However our system is used at outdoor situation under various illumination and the distribution of skin color varies substantially. Consequently skin color model should be created dynamically to adapt to various illumination.

Our system has the calibration interface easy to input skin color model's parameters. If the user shoots a hand to fill the image with it like Fig. 4, the system detects this action automatically, and uses the distribution of the image to calculate parameters of the skin color model.

It is necessary to use automatic exposure control with a camera to adapt various brightness. However, it does not necessarily work well, and in that situation the color of the object is lost because of overexposure or underexposure like Fig. 5. Usually, overexposure occurs when the object is placed against dark background. Using our input method, overexposure or underexposure is reduced comparatively because the skin reflectance is near the 18%. The normal automatic exposure producing an average 18% reflectance image. Fig. 6 shows that overexposure of the leaf is suppressed by placing a hand at the back of the leaf.

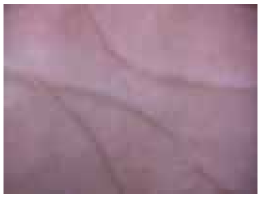

Fig. 4. An image for calibaration

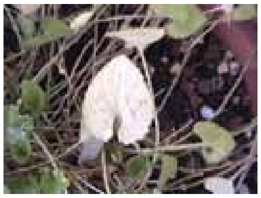

Fig. 5. A leaf is overexposured

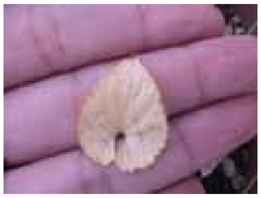

Fig. 6. A leaf is exposured sufficeintly

The color of the object changes according to illumination condition. For example, the object is tinged with red by the setting sun. This change has a negative impact on the object recognition system. The automatic white balance can suppress this change of color and produce normalized colored images under standard white light. However the automatic white balance is not perfect, the object color can not be normalized depending on the input image. Using our method, the object region and the skin region are both detected in the same image. If we know the color of the skin in white light, we can suppress the change of color of the object based on the color of the skin.

2.1 Comparison with other region input interfaces

Pen or Touchpad There are many works that input regions with a pen or a touchpad. For instance, to use the method of F. Zanoguera et al.[8], a user can input the region, only by drawing a line on the target region roughly. It is suitable way in front of a desk. However this action needs user's both hands to use in mobile systems. The user must specify the line with a pen or a finger, and the user must hold the display by the other hand. There is often a case that the user must use his hand to clear the obstructive weeds to shoot the object. For instance, leaves occlude the target flower or the target flower is moving because of the wind. In such a case, the user can not shoot a target picture and specify the region with a pen at the same time. Our interface needs only one hand and the other hand can be used for clearing the obstructive weeds.

using camera with distance information There are systems that can produce the distance from a camera to the pixels' resource point. For instance, it can be computed by using two cameras. Using these systems, the object region can be obtained robustly in the situation that background objects are far from camera compared to the object. But when many leaves are overlapped each other, the system may not detect the object region with distance information. In such situation, our system can extract the object region, by picking the target leaf and put it on the hand. This method is not suitable for the system using distance information, because the hand and the leaf is too close.

3 Implementation

In this section, we describe the implementation of the region input method. It is necessary to detect the hand region at first in order to recognize the object region with background of a hand. We detects a hand region by using color information utilizing the human skin colors cluster in the color space [4].

3.1 Skin color model

We used three dimensional gaussian distribution in HSV color space as a model for a skin color distribution[1]. We uses HSV color space because it is relatively insensitive to the change of the brightness component. It is also known that the skin color extraction in HSV color space is more effective than in RGB color space[3]. A color vector captured by a camera is described in RGB color space, should be transformed to HSV color space at first. This process can be done at high speed with a transformation table. The parameters of this model are a mean vector and a covariance matrix. We describe how to get these parameters of skin distribution in next subsection.

3.2 Acquisition of skin color model parameters

We prepare the interface that the user can input skin color model parameters very easily. The system can detect skin color regions under various illumination. The system calculates a mean vector and a covariance matrix of the color vector of all the pixels in each images captured at every frame. If the values in the covariance matrix are all less than thresholds then it is considered that the image is filled by almost a single colored object like Fig. 4. We use the mean vector and the covariance matrix at that time as parameters for the skin color distribution. Notice that a single colored object does not need to be a hand. It is not a problem because it usually does not happen without the user's intention. It is rather convenient because single colored objects can replace a hand as background if the user wants.

3.3 Skin color extraction

Using gaussian color filter[1], the system calculates a probability of skin image from the captured image in each frame. Gaussian color filter is expressed as the formula (1).

$$c^i = e^{(X_i - \hat{X})^T \sum^{-1} (X_i - \hat{X})} \qquad (1)$$

X_i denotes the color vector of i'th pixel in the image. c^i denotes the dissimilarity from the skin color correspond to i'th pixel. $\hat{X}$ denotes the skin color mean vector and $\sum^{-1}$ denotes the inverse of skin color covariance matrix. The original image is shown at Fig. 7. The result image is shown at Fig. 8. The brightness of the image represents the dissimilarity from the skin color. Consequently the hand region is emerged as a darker region.

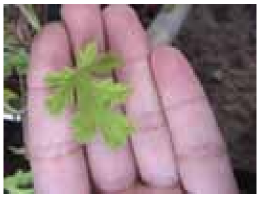

Fig. 7. An original image

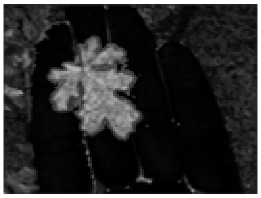

Fig. 8. Dissimilarity image of the skin

Before applying gaussian color filter, we apply moving average filter to original images to degrade an influence of noise.

3.4 Identification of the object region

The object region is surrounded by the region with the color of skin. First, skin colored region is calculated by thresholding the dissimilarity of skin image. Then, dilate filter and successive erode filter is applied to remove noise and narrow regions between adjacent two fingers shown at Fig. 8. Finally, we identify connected components in the image, and the hole of the largest region is regarded as the input image. Small holes under the threshold are ignored as noise. For example, in Fig. 9, suppose white connected pixels as a skin region, B2 and B4 are regarded as the object region, if B2 and B4 are larger than the threshold. Notice that W4 is not included in the input region. The W4 is regarded as the hole of the object.

4 Evaluation

4.1 Processing time

We measure the processing time with two machines, a mobile PC SONY PCG-GT1(TM5600, 600MHz) with a incorporated camera, and a desktop PC (Pentium III 600E, 600MHz) with a camera IO-DATA USB-CCD.

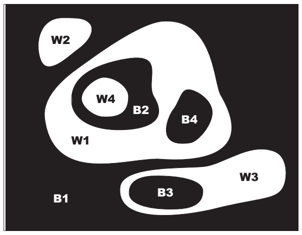

Fig. 9. regions

Table 1 shows the average number of frames processed per second. The number in parentheses is the number of frames to calculate that average. There is a tendency that the amount of the processing task if the system detects the region is larger than that of not detected. For example if the system does not detects the object region the skin colored region may be not detected and an area of the skin colored region is not calculated. On the other hand the object region is detected then an area of the skin colored region is calculated certainly. The first row represents the frame rate while the system detects the input region. The second row represents the frame rate while the system does not detect the input region. The third row shows the average of all frames. For example, with PCG-GT1, the input region is detected in 573 frames per 1440 frames.

From the average frame rate of the all frames, the interactive response time is achieved with both PCs. The difference between the average rate of frames while the system can detect the region and that of not detected is sufficiently small. Consequently, the processing speed is considered as stable.

Table 1. frame rate

	TM5600	P3 600E
frames that the input region detects	6.8 fps (265)	16.2 fps (573)
frames that the input region are not detects	8.4 fps (735)	17.0 fps (867)
all frames	7.9(1000)	16.7(1440)

4.2 Precision and Input time

We performed an experiment that a user inputs the regions of 15 leaves and flowers using "SONY PC-GT1". The experiment was done at about 16:00 in August. It was fine weather. From Fig. 10 to Fig. 15 shows examples of the detected regions. Which are represented as closed white lines. Fig. 10, 11 and 12 shows the examples that succeeded to detect intended regions. Fig. 13, 14 and 15 are examples that failed to input intended regions.

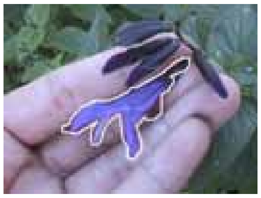

Fig. 10. 1-flower

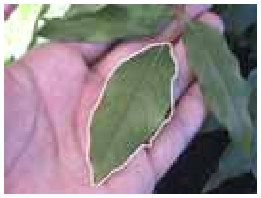

Fig. 11. 6-leaf

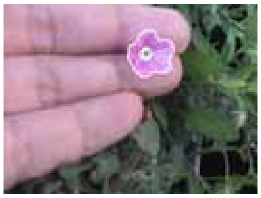

Fig. 12. 9-flower

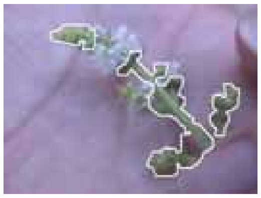

Fig. 13. 3-flower

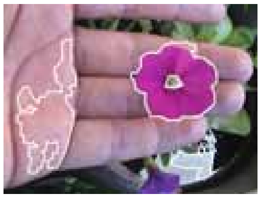

Fig. 14. 8-flower

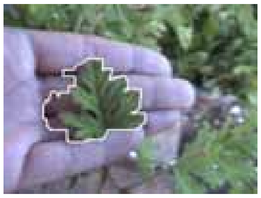

Fig. 15. 10-leaf

The detected regions in Fig. 13 do not match the real shape of the flower. The flower part is so small that color of the flower is weakened by applying moving average filter to suppress noise components. If we use a camera that can produce images with lower noise, this kind of flowers can be extracted. In Fig. 14 unintended region is detected. This region can be input by moving the object close to a camera. The unintended region is detected because the skin color distribution is changed from the time the user input the skin color by the automatic exposure control. To avoid this situation, more applicable skin color model is against the change of brightness of images needed. The region in Fig. 15 is almost correct but the outline of the shape is not accurate. The shadow of the leaf on the hand is not recognized as the skin, because the skin can not to be expressed by the gaussian color model sufficiently. The more expressive model for the skin color is needed to detect this kind of shapes accurately.

Fig. 16 shows the rate of regions that can be detected by the system precisely as time elapses form the beginning. The timing starts when the object region is captured in a frame of a camera completely. The graph shows that 20% of regions can be input in 2 seconds. The experiment is done with 15 kinds of region, 3 kinds of region can be input in 2 second. This result indicate that this interface has capability to input the object region very fast. As time elapses the number of regions that can be detected increases because the user tries to input the correct region in various ways. For example, the user may reinput the skin color parameters described at Section 2, or move a target close to the camera. In the case shown in Fig. 14, the user can understand that the unintended region will be removed if the user move a target close to the camera. About 70% of regions can be input in 14 seconds. This result indicates that most kinds of region can be input in acceptable time.

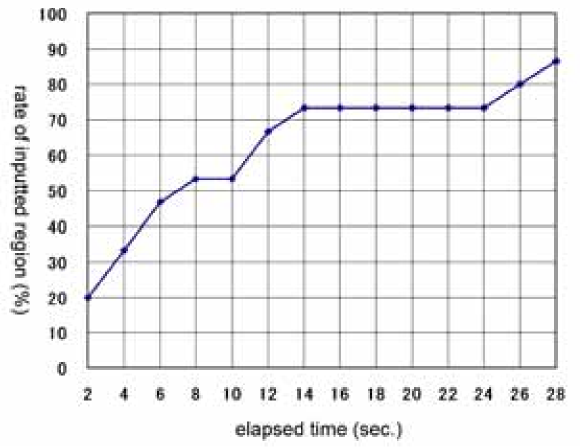

Fig. 16. Rate of the inputted region

5 Related works

S. Walherr et. al developed the system that extracts skin regions to make a robot that understands arm gestures[1]. Our system uses the gaussian color filter described in the paper. However it uses static skin color model which is not suitable for outdoor use, we added the interface to change skin color model's parameters easily.

S.Ahmad[6]'s system tracks hands on the desk to recognizing gestures. The system divides the input image into small rectangles, and match the histogram of each rectangle and the skin color histogram prepared beforehand. The rectangles whose histograms are similar to the skin color histogram are recognized as skin color regions. In this way, more robust skin extraction can be possible than pixel wise extraction of the skin, the resolution of the skin color image will be smaller than that of the input image. In our system, the shape of the hand region is more important than their system, because the hole of the hand region is used as the target object shape.

L.Sigal[3] at. el tracks skin color regions in movies encoded with MPEG-2. The system changes color model to utilize the location of the skin colored region in the image assumed not to change a lot. Using this technique, the hand region can be tracked robustly, but the skin color at the first frame probably have to be ordinary color of the skin illuminated by almost white light. Therefore this technique does not have sufficient ability to detect the skin region under unusual illuminations, like red light in the sunset. Our system equipped the interface to input the skin color parameters easily then any color distributions that can be represented as Gaussian can be input as skin color distributions. Consequently the skin region under unusual illuminations can be detected.

Madirakshi Das at. el developed the system that classifies images of flower patents based on colors of the flowers[7]. The system is not supposed the background color of the flower unlike our system. First, the system removes the region regarded as background iteratively and regard the conspicuous region remained as the flower region. But this technique is valid if the color used for the background is different color from the flower color. The flower patent images satisfies this condition. But this technique is not suitable to input a leaf region clustered with the other leaves. Because a leaf and the other leaf have almost the same color. Consequently, even using this technique, it is necessary to place a hand behind the object in order to make conspicuous the object contour.

6 Future works

Improving precision Descrived at previous section, gaussian distribution is too simple to model skin color distribution of even a specific people sufficiently. More applicable skin color model against the change of brightness of the skin is also needed. We are developing a more robust skin extract technique.

Enhancement of kinds of the region that can be input In our method it is impossible to input the region larger than a hand. It's region cannot be specified by placing a hand behind the target object.

Implementation of the flower recognition system In this paper, we described the region input method for the flower recognition system. Using this method, we can input flower regions or leaf regions. Now we can match the regions with that in the database of flowers. To implement the whole system we have to select the region matching algorithm suitable for and the inference engine that returns the species incorporating the result of the matching a flower and a leaf and the other information, such as the season.

7 Conclusion

We proposed the object region input method that the user places a hand in back of the object. The system detects this automatically so that the user can input the target region without any other input devices. Consequently the user can concentrate in putting the target in a frame of a camera. This action is considered as natural, because this action is shown when one watch the object closely. This interface is suitable for the system that used in the outdoor scene such as the system that identifies the flower species.

References

1. S. WALDHERR, S. THRUN, R. ROMERO, and D. MARGARITIS. Template-Based Recognition of Pose and Motion Gestures On a Mobile Robot, In Proc. of AAAI, pp. 977-982, 1998.

2. Yu-Hua (Irene) Gu, and T.Tjahjadi. Corner-Based Feature Extraction for Object Retrieval, In Proc. of ICIP. pp.119-123, 1999
3. L. Sigal, S. Sclaroff, and V, Athitsos. Estimation and Prediction of Evolving Color Distributions for Skin Segmentation Under Varying Illumination, In Proc. of CVPR, 2000.
4. J. Yang, W. Lu, and A. Waibel. Skin-color Modeling and Adaptation, In Proc. of ACCV'98, Vol. II, 1998, pp.687-694.
5. B. D. Zarit, B. J. Super, and F K. H.Quek. Comparison of Five Color Models in Skin Pixel Classification., ICCV'99 International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems, pp58-63, 1999
6. S. Ahmad. A Usable Real-Time 3D Hand Tracker. Conference Record of the Asilomar Conference on Signals, Systems and Computers. pp. 1257-1261, 1994
7. Madirakshi Das, R. Manmatha, and E. M. Riseman. Indexing Flower Patnet Images Using Domain Knowledge, IEEE Inteligent Systems, pp. 24-33, 1999
8. F. Zanoguera, B. Marcotegui and F.Meyer. A Toolbox for Interactive Segmentation Based on Nested Partitions. ICIP, pp. 21-25, 1999
9. Intel Open Source Computer Vision Library, http://www.intel.com/research/mrl /research/opencv/