REAL-TIME ADAPTIVE LEARNING SYSTEM USING OBJECT COLOR PROBABILITY FOR VIRTUAL REALITY APPLICATIONS

Chutisant Kerdvibulvech
Department of Information and Communication Technology, Rangsit University
52/347 Muang-Ake, Paholyothin Rd. Lak-Hok, 12000, Pathum Thani, Thailand

Keywords: Real-time System, Adaptive Learning, Visualizing System, Color Probability, Virtual Reality.

Abstract: Segmentation is not a trivial task, especially in challenging situations such as outdoor area. In this paper, we develop an adaptive learning system to segment an object robustly. By using the on-line adaptation of color probabilities, the proposed method presents several specific features: it is able to cope with illumination changes even in the outdoor area, and also it can be done in real-time. Bayes' rule and Bayesian classifier is employed to calculate the probability of an object color. Representative experimental results are also presented and discussed. The system presented can be further used to develop the real-time game of augmented reality in virtual spaces.

1 BACKGROUND

Computer vision is the technology of machines that see and analyze scientifically. Research about color segmentation and a model of color is a very popular topic, due to the popularity of virtual reality and augmented reality recently. Basically, a model of color relates to the selection of the color space is usually mentioned and used previously. Many color spaces have been proposed including RGB (Jebara and Pentland, 1997), normalized RGB (Kim et al, 1998), YCrCb (Chai and Ngan, 1998), etc. Color spaces efficiently separating the chrominance from the luminance components of color are typically considered preferable. Previous method is to find the proper threshold values of each color model, e.g. HSV (Gonzalez and Woods, 2002). Another recent method used (Cabrol et al, 2005) color region segmentation followed by a color classication and region. After they segmented, they use it for RoboCup application, i.e. four-legged league or an industrial conveyor wheeled robot. However, by using these previous methods, it is not able to cope with considerable luminance changes effectively.

This paper proposes a real-time system using a Bayesian classifier that is bootstrapped with a small set of training data and refined through an off-line iterative training procedure (Argyros and Lourakis, 2004). We calculate the color probabilities being object color. The learning process is composed of two phases.

In the first phase, the color probability is obtained from a small number of training images during an off-line pre-process. The color representation used in this process is YUV 4:2:2 (Jack, 2004). This color space contains a luminance component (Y) and two color components (UV). The main advantage for using the YUV space is that the luminance and the color information are independent. Thus, it is easy to separate the chrominance from the luminance components of color. As object tones differ mostly in chrominance and less in intensity, by employing only chrominance-dependent components of color, one can achieve some degree of robustness to changes in luminance.

In the second phase, we gradually update the probability automatically and adaptively from the additional training data images. However, the Y-component of this representation is not employed for two reasons. Firstly, the Y-component corresponds to the luminance of an image pixel. By omitting this component, the developed classifier becomes less sensitive to luminance changes. Secondly, compared to a 3D color representation (YUV), a 2D color representation (UV) is lower in dimensions and,

Kerdvibulvech C..
REAL-TIME ADAPTIVE LEARNING SYSTEM USING OBJECT COLOR PROBABILITY FOR VIRTUAL REALITY APPLICATIONS.
DOI: 10.5220/0003622802000204
In *Proceedings of 1st International Conference on Simulation and Modeling Methodologies, Technologies and Applications* (SIMULTECH-2011), pages 200-204
ISBN: 978-989-8425-78-2

therefore, less demanding in terms of memory storage and processing costs. This disregard of the luminance value has also been shown to be useful in detection and tracking of faces (Hua at al, 2002) and color night vision (Shi at al, 2007).

In our proposed method, we set that the adapting process can be disabled as soon as the achieved training is deemed sufficient. Therefore, when we start to learn the on-line object color adaptation, we assume that there is enough object area in the image. As soon as the on-line adapting process is enough as we prefer (i.e. the object color probability converges to a proper value), we manually stop the adapting process. In this way, after finishing the on-line learning process, although the object area disappears from the scene, it does not affect the object color probability.

Therefore, this method allows us to get accurate color probability of the object from only a small set of manually prepared training images. This is because the additional object region does not need to be segmented manually. Also, because of the adaptive learning, it can be used robustly with changing luminance during the on-line operation.

2 METHOD

This section will explain the method we used for segmenting the object region. We calculate the color probabilities being object color adaptively. The learning process is composed of two phases.

2.1 Off-line Learning

During an off-line phase, a small set of training input images is selected, on which a human operator manually delineates object-colored regions, as shown in Figure 1.

Figure 1: Off-line learning process.

Following this, assuming that image pixels with coordinates *(x,y)* have color values $c = c(x,y)$, training data are used to calculate:

(i) The prior probability *P(o)* of having object color *o* in an image. This is the ratio of the object-colored pixels in the training set to the total number of pixels of whole training images.

(ii) The prior probability *P(c)* of the occurrence of each color in an image. This is computed as the ratio of the number of occurrences of each color *c* to the total number of image points in the training set.

(iii) The conditional probability *P(c|o)* of an object being color *c*. This is defined as the ratio of the number of occurrences of a color *c* within the object-colored areas to the number of object-colored image points in the training set.

By employing Bayes' rule, the probability *P(o|c)* of a color *c* being an object color can be computed by using

$$P(o \mid c) = \frac{P(c \mid o)P(o)}{P(c)} \tag{1}$$

This equation determines the probability of a certain image pixel being object-colored using a lookup table indexed with the pixel's color. The resultant probability map thresholds are then set to be threshold $T_{\max}$ and threshold $T_{\min}$, where all pixels with probability $P(o|c) > T_{\max}$ are considered as being object-colored—these pixels constitute seeds of potential object-colored blobs—and image pixels with probabilities $P(o|c) > T_{\min}$ where $T_{\min} < T_{\max}$ are the neighbors of object-colored image pixels being recursively added to each color blob. The rationale behind this region growing operation is that an image pixel with relatively low probability of being object-colored should be considered as a neighbor of an image pixel with high probability of being object-colored. Smaller and larger threshold values cause the false object detection. For example, if we choose the threshold value $T_{\max}$ that is too big, we cannot detect any pixels that constitute the seeds of potential object blobs. Therefore, the values for $T_{\max}$ and $T_{\min}$ should be determined by test experiments (we use 0.5 and 0.15, respectively, in this experiment). A standard connected component labelling algorithm (i.e. depth-first search) is then responsible for assigning different labels to the image pixels of different blobs.

Size filtering on the derived connected components is also performed to eliminate small isolated blobs that are attributed to noise and do not correspond to interesting object-colored regions. Hence, connected components that consist of less than the threshold size are assumed to be noise and then rejected from further consideration. Each of the remaining connected components corresponds to an object-colored blob. In this step, we choose the biggest region as an object-colored blob.

2.2 Adaptive Learning System

Training is an off-line procedure that does not affect the on-line performance of the tracker. Nevertheless, the compilation of a sufficiently representative training set is a time-consuming and labor-intensive process. To cope with this problem, an adaptive training procedure has been developed. Training is performed on a small set of seed images for which a human provides ground truth by defining object-colored regions. Following this, detection together with hysteresis thresholding is used to continuously update the prior probabilities *P(o)*, *P(c)* and *P(c|o)* based on a larger image data set. The updated prior probabilities are used to classify pixels of these images into object-colored and non-object-colored ones. The final training of the classifier is then performed based on the training set resulting from user editing. This process for adapting the prior probabilities *P(o)*, *P(c)* and *P(c|o)* can either be disabled as soon as the achieved training is deemed sufficient for the purposes of the tracker, or continue as more input images are fed to the system.

The success of the color detection depends crucially on whether or not the luminance conditions during the on-line operation of the detector are similar to those during the acquisition of the training data set. Despite the fact that using the UV color representation model has certain luminance independent characteristics, the object color detector may produce poor results if the luminance conditions during on-line operation are considerably different to those used in the training set. Thus, a means of adapting the representation of object-colored image pixels according to the recent history of detected colored pixels is required. To solve this problem, object color detection maintains two sets of prior probabilities (Zabulis et al, 2009). The first set consists of *P(o)*, *P(c)*, *P(c|o)* that have been computed off-line from the training set. The second is made up of $P_W(o)$, $P_W(c)$, $P_W(c\,|\,o)$, corresponding to the *P(o)*, *P(c)*, *P(c|o)* that the system gathers during the *W* most recent frames respectively. Obviously, the second set better reflects the "recent" appearance of object-colored objects and is therefore better adapted to the current luminance conditions. Object color detection is then performed based on the following moving average formula:

$$P_A(o\,|\,c) = \gamma P(o\,|\,c) + ((1-\gamma)P_W(o\,|\,c)), \qquad (2)$$

where $P_A(o\,|\,c)$ represents the adapted probability of a color *c* being an object color. *P(o|c)* and $P_W(o\,|\,c)$ are both given by Equation (1) but involve prior probabilities that have been computed from the whole training set [for *P(o|c)*] and from the detection results in the last *W* frames [for $P_W(o\,|\,c)$]. γ is a sensitivity parameter that controls the influence of the training set in the detection process $(0 \le \gamma \le 1)$. If $\gamma = 1$, then the object color detection takes into account only the training set (35 images in the off-line training set), and no adaptation takes place; if γ is close to zero, then the object color detection becomes very reactive, relying strongly on the recent past for deriving a model of the immediate future. *W* is the number of history frames. If *W* value is too high, the length of history frames will be too long; if *W* value is set too low, the history for adaptation will be too short. In our implementation, we set γ = 0.8 and *W* = 5 which gave good results in the tests that have been carried out.

Thus, the object color probability can be determined adaptively. By using on-line adaptation of object color probabilities, the classifier is easily able to cope with considerable luminance changes, and also it is able to segment the object even in the case of a dynamic background.

3 RESULTS

In this section, representative results from our experiment are shown. Figure 2 provides a few representative snapshots of the experiment. The reported experiment is based on a sequence that has been acquired with USB camera at a resolution of 320x240 pixels. This process is done in real time and on-line. The experimental room is outdoor area (balcony). Note that the training set (35 images in the off-line training set) was collected from the indoor area. So the luminance change makes it much more challenging.

The left window depicts the input images. The middle window shows the output images. The right window represents the object probability map in the U and V axis in color model, as depicted in Figure 3.

In the initial stage (frame 15), when the experiment starts, the object color probability does not converge to a proper value. In other words, the color probability is scattering. So the segmented output cannot be achieved well because it uses only from the off-line data set which the lighting is

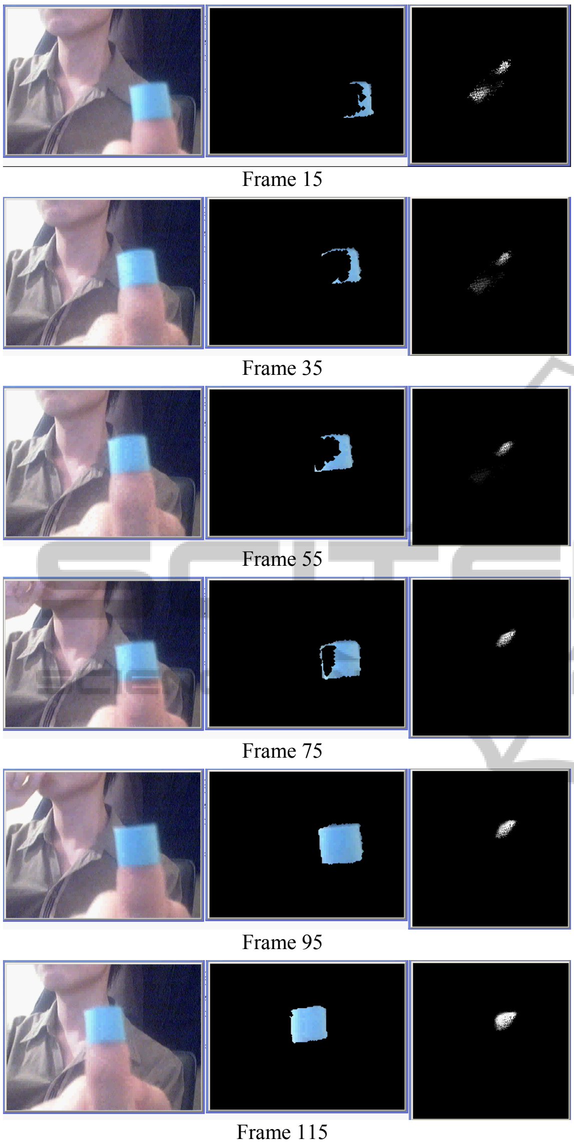

Figure 2: Object segmentation based on the color probability: from off-line learning to adaptive learning.

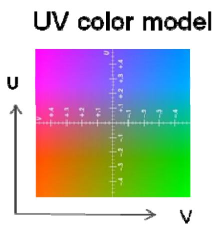

Figure 3: U and V axis in color model.

extremely different. At frame 35, frame 55 and frame 75, after performing the adaptive learning process, the object color probability gradually converges to a proper value. Thus, the result becomes better. Later on at frame 95 and frame 115, the result can be achieved robustly. The lighting used to test between off-line (at indoor area) and on-line (at outdoor area) is obviously different. However, it can be observed that the 2D segmented result of object can be still determined without effects of different light sources in each representative frame. This is because a Bayesian classifier and an on-line adaptation of color probabilities are utilized to deal with this.

4 CONCLUSIONS

In this paper, we have developed an adaptive learning system which is performed by using a Bayesian classifier and the online adaptation of object color probabilities. This method is to effectively deal with any illumination changes both indoor and outdoor areas. Furthermore, the computation time of the method is real-time. The result of the proposed method is the segmenting of objects to be able a human-computer interaction system for virtual reality and augmented reality environments. Our future work is to apply this process to a real-life augmented reality application. In the future, we intend to further refine this problem.

ACKNOWLEDGEMENTS

The work presented in this paper is supported in part by a Grant-in-Aid from the Research Institute of Rangsit university.

REFERENCES

Argyros, A. A., Lourakis, M. I. A., 2004. Real time Tracking of Multiple Skin-Colored Objects with a Possibly Moving Camera. *European Conference on Computer Vision*.

Cabrol, A. D., Bonnin, P., Costis, T., Hugel, V., Blazevic, P. (2005). A New Video Rate Region Color Segmentation and Classication for Sony Legged RoboCup Application. *Lecture Notes in Computer Science. RoboCup*.

Chai, D., Ngan, K. N., 1998. Locating facial region of a head-and-shoulders color image. *3rd IEEE International Conference on Automatic Face and Gesture Recognition (FG'98)*.

Gonzalez, R., Woods, R. E., 2002. *Digital Image*

Processing. 2 ed, Prentice Hall Press., ISBN 0-201-18075-8.
Hua, R. C. K., Silva, L. C. D., Vadakkepat, P., 2002. Detection and Tracking of Faces in Real-Time Environments. *International Conference on Imaging Science, Systems, and Technology*.
Jack, K., 2004. *Video demystified*, Elsevier science, 4th edition.
Jebara, T. S., Pentland, A., 1997. Parameterized structure from motion for 3D adaptive feedback tracking of faces," *IEEE Conference on Computer Vision and Pattern Recognition*.
Kim, S. H., Kim, N. K., Ahn, S. C. and Kim, H. G., 1998. Object oriented face detection using range and color information. *3rd IEEE International Conference on Automatic Face and Gesture Recognition*.
Shi, S., Wang, L., Jin, W., Zhao, Y., 2007. Color night vision based on color transfer in YUV color space. *International Symposium on Photoelectronic Detection and Imaging*.
Zabulis, X., Baltzakis, H., Argyros, A. A., 2009. Vision-based Hand Gesture Recognition for Human-Computer Interaction, *In "The Universal Access Handbook", Lawrence Erlbaum Associates, Inc. (LEA), Series on "Human Factors and Ergonomics"*, ISBN: 978-0-8058-6280-5.