Face Detector using Network-based Services for a Remote Robot Application

Yong-Ho Seo

Department of Intelligent Robot Engineering, Mokwon University
Mokwon Gil 21, Seo-gu, Daejeon, Republic of Korea
yhseo@mokwon.ac.kr

Abstract

This paper proposed a face detector using network-based services for a remote robot application. To perform an efficient face detecting in a real robot, a face detector needs to light computation cost and low false detection. The proposed face detector has been implemented as a network-based service so that it can be deployed either in any computer nodes in network environments or in the robot itself. The human-following service has been also developed to perform a coordination service for a robot that can follow a user. The proposed method reduces the overall computation time and reduces the number of false positives. And the proposed method has been verified by successfully demonstrating a human-following of a robot in a real indoor environment.

Keywords: *Network-based Services, Human-Following, Face Detector, Vision Based Tracking*

1. Introduction

An image processing technique that performs face detection is an interesting issue in research areas including human computer interaction and human robot interaction. In general, face detection is a time-consuming process due to the large search space involved: to find the location and size of a facial sub-window, it is necessary to adapt a face/non-face classifier to various sizes and locations of given images.

A vision based tracking that a robot can follow a user for a robot equipped with a camera is also an interesting issue in robotics research nowadays [1]. A network-based programming skill has been also widely used for a robot to do an autonomous task remotely by developers and researchers in robotics area. The key characteristics and patterns of a service-oriented network programming have been widely adapted in the various remote robot applications [2].

In this paper, an efficient face detector using a network-based service programming for a robot to do a reliable robot tasks such as human-following has been proposed. In order to accelerate face detector, an efficient sub-window scanning scheme and a face/non-face classifier based on facial color density has been adapted. The proposed method contributes to reducing the overall computation time of face detection and to eliminating false alarms.

2. Developing Network-based Services

To implement a network-based face detector for remote robot application, MSRDS (Microsoft Robotics Developer Studio) which is a development platform for the robotics community, supporting a wide variety of users, hardware, and application scenarios was used in this research. This software toolkit is based on Microsoft’s .NET framework [3].

This tool supports a network based service programming for robotics through a lightweight REST-style, service-oriented runtime so that a programmer can create network applications easily to monitor or control a remote robot. Figure 1 shows the structure of MSRDS application under its windows host system and runtime.

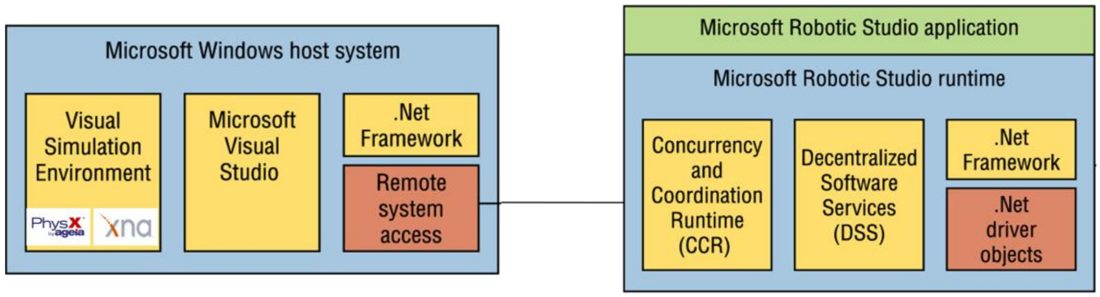

Figure 1. Structure of MSRDS Application

Major features of Microsoft Robotics Developer Studio are CCR and DSS for programming network services for developing robotic applications. CCR means Concurrency and Coordination Runtime to deal with various asynchronous inputs from multiple robotic sensors and outputs to actuators [4]. DSS is a Decentralized Software Services that provides a lightweight, state-oriented service model that combines the notion of representational state transfer with a system-level approach for building high-performance, scalable applications.

In DSS services are exposed as resources which are accessible both programmatically and for UI manipulation. By integrating service isolation, structured state manipulation, event notification, and formal service composition, DSS addresses the need for writing high-performance, observable, loosely coupled applications running on a single node or across the network. DSS controls the creation of network-based services and manages communication amongst them. The DSS runtime provides a hosting environment with built-in support for service composition, publish/subscribe, lifetime-management, security, monitoring, logging, and much more both within a single node and across the network. Figure 2 shows the internal structure of DSS service implementation and an example of deploying network-based services into network environment using DSS.

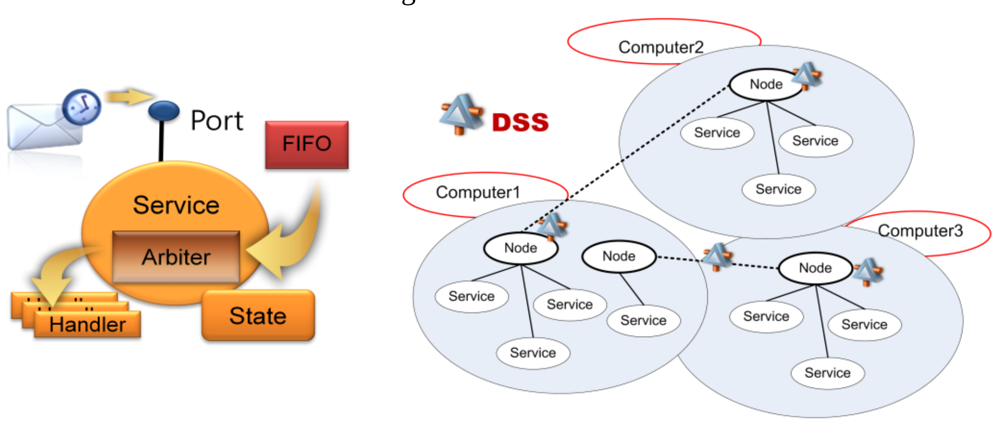

Figure 2. Internal Structure of DSS Service (left) and Network-based Services using DSS (right)

3. Face Detector Service

In this research, face detector was developed as a service named Simple Face service that implements image processing functions using a conventional Webcam USB camera. In order to adopt face detector as network-based service, it is important to make light-weight face detector. In this research, we propose a means to use facial color to accelerate a conventional adaboost face detector [5]. The proposed method adapts a sub-window scanning skim and a face/non-face classifier based on facial color density.

3.1. Facial Color Filtering

In a statistical model of facial color can be obtained by a Bayesian rule as follows.

$$p(face|color) = \frac{p(face \cap color)}{p(color)} \quad (1)$$

In equation (1), *p(face|color)* means the facial color likelihood, *p(color)* is the color probability distribution of all the images, and *p(face ∩ color)* refer to the color probability distribution of all facial images. In addition, a facial color membership function *M(color)* that has a high membership value to include rare facial colors is proposed as follows.

$$M(color) = \frac{\max_{I_i \in face}(p_i(color))}{p(color)} \quad (2)$$

In equation (2), $p_i\,(color)$ denotes the color probability distribution of an image I_i. By merging the color probability distributions of facial images through max operation, rare facial colors in sample space can have high likelihoods. To obtain M(color), all facial or non-facial images are represented in the HSV color coordinate and a histogram over hue, and saturation is calculated to obtain p(color) and pi(color). M(color) is stored in a look-up table indexed by hue and saturation, and is convolved with a 2D Gaussian function for generalization. Using M(color), the facial color membership value of each pixel can be obtained.

By thresholding the facial color membership value with θ, a facial color filter image $I_f(x,y)$ can obtained as follows.

$$I_f(x,y) = \begin{cases} 1 & if\ M\big(h(x,y), s(x,y)\big) \geq \theta \\ 0 & otherwise \end{cases} \quad (3)$$

In equation (3), *h(x,y)* and *s(x,y)* denotes the hue and saturation of pixel *(x, y)* of a given image, respectively. $I_f(x,y)$ is a binary image whose pixel value is 1 when its color belongs to a facial color. To calculate facial color information efficiently, we use integral image $I_{if}(x,y)$ of $I_f(x,y)$.

$$I_{if}(x,y) = \sum_{x' \leq x, y' \leq y} I_f(x', y') \quad (4)$$

Using $I_{if}(x,y)$, the density of a sub-window is calculated with a relatively light computation load. When the range of a sub-window is (x_{tl}, y_{tl})~ (x_{tr}, y_{tr}), the density is calculated as follows.

$$d(win) = \frac{I_{if}(x_{br}, y_{br}) - I_{if}(x_{br}, y_{tl}-1) - I_{if}(x_{tl}-1, y_{br}) + I_{if}(x_{br}, y_{tl}-1)}{(x_{br}-x_{tl}+1)(y_{br}-y_{tl}+1)} \quad (5)$$

In equation (5), *win* denotes the range of the sub-window $\{x_{tl}, y_{tl}, x_{br}, y_{br}\}$. Using facial color density *d*, two methods to enhance the conventional face detector are proposed.

3.2. Face Detection using Facial Color

In the conventional face detector, the detector scans the target image linearly. However, this scanning skim is a time-consuming process due to the large search space involved. In this research, we propose face detection skim which scan an image sparsely based on facial color density. The proposed method determines the horizontal scan interval based on *d(win)* as follows.

$$si_h = \begin{cases} \delta & d(win) = 1 \\ \omega(1 - d(win)/\emptyset) & d(win) < 1 \end{cases} \quad (6)$$

In equation (6), si_h is the horizontal scan interval, δ is the minimum scan interval, ω is the width of the sub-window, and $\emptyset$ is the minimum density, which is determined to be 0.55 by experiment. By changing the scan interval, the detector skips sub-windows that have no possibility to be faces. A similar method can be applied to the vertical direction as follows.

$$d_{min} = min_{win \in A}(1 - d(win)/\emptyset) \quad (7)$$

$$si_v = \begin{cases} \delta & d_{min} = 1 \\ h \cdot d_{min} & d_{min} < 1 \end{cases} \quad (8)$$

In equation (7), A denotes a set of sub-windows that are in the same row and dmin denotes minimum density among sub-windows in the same row. In equation (8) siv is the vertical scan interval and h is the height of the sub-window.

The proposed model adopts a face/non-face classifier using facial color for the initial stage of the adaboost face detector. From the facial color integral image, the facial color classifier uses the facial color density. If the density is below the minimum density $\emptyset$, then the classifier rejects the sub-window.

4. Experimental Results

To evaluate the proposed model, we performed a face detection test using our own face image dataset. This dataset consists of 80 color images. Each image contains one or more upright faces. We conducted the experiment on a Pentium 4 2.4Ghz PC. We applied the proposed skim to a face detector supported by OpenCV [6]. To detect both frontal and profile faces, we combined 1 frontal classifier and 2 profile classifiers sequentially. We compared the proposed method with the conventional adaboost face detector.

Overall results are shown in Table 1. Compared the proposed detector with the conventional detector, the number of false alarms is considerably reduced and the computational time is reduced to 54% for the dataset. While the detection rate of the proposed face detector is the same as that of the conventional detector.

In the experiments, the proposed face detector service can detect user's face under varying lighting conditions and complex backgrounds from the live frame images captured from a webcam on a mobile robot at a high detection rate of over 90%. This result is similar with the conventional face detection method using adaboost. However the computation time of this face detection service is much faster than other existing methods.

Table 1. Overall Results of Face Detection Test

	Conventional adaboost	Proposed face detector
Detection Rate	95.24%	95.24%
False Alarm	170	87
Average Processing Time	733.33ms	396.40ms

In this research, a human-following service has also been developed to perform an orchestration service which means a coordination service so that a robot can follow a user using the proposed face detector service.

The follower service controls a mobile robot which has a two-wheel differential drive, front and rear contact sensors. The service traces a human using the location information provided by the face detector service. The follower service orchestrates several partner services such as drive which is used for robot movements, contact sensor service as an implementation of robot bumper to avoid obstacles and Simple Face service which performs face detecting.

The developed face detector and following services have been successfully demonstrated using a mobile robot called "X-Bot" which is modified from a conventional cleaning robot, "iClebo" from Yujin Robot to place a laptop computer on top of the mobile robot [7]. The developed services successfully perform following a human as shown in the following figure 3.

Figure 3. Experiment of a Human-Following Services using a Mobile Robot, X-Bot

5. Conclusion

This paper proposed a new approach using a network service programming for a robot to do a vision based tracking in a real environment. A face detector as a network service for a real robot can be run in a remote network node and uses light and efficient face detection skim based on facial color filtering.

In order to make efficient face detector service, we have proposed a color filtering-based face detection service with efficient sub-window scanning. Proposed service scans the image space sparsely based on facial color density. As compared with adaboost face detector, we have shown that proposed detection results in lower overall computational costs and fewer false alarms, while detection ratio is same with conventional one.

The proposed face detector service in a network environment successfully performs face detections so that a real robot can smoothly follow a user. Finally the feasibility and effectiveness of the proposed detector has been successfully demonstrated.

Acknowledgements

This study was financially supported by research fund of Mokwon University in 2010.

References

[1] C. Schlegel, J. Illmann, H. Jaberg and M. Schuster, "Vision based person tracking with a mobile robot", In Ninth British Machine Vision Conference, BMVC '98, Southhampton, pp. 418-427, **(1998)**.

[2] S. L. Remy and M. B. Blake, "Distributed Service-Oriented Robotics", *Internet Computing, IEEE*, vol.15, no.2, pp. 70-74, **(2011)** March-April.

[3] Microsoft Robotics Developer Center, http://msdn.microsoft.com/robotics, and .NET Framework Developer Center, http://msdn.microsoft.com/netframework.

[4] O. Almeida, J. Helander, H. Nielsen and N. Khantal, "Connecting Sensors and Robots through the Internet by Integration Microsoft Robotics Studio and Embedded Wen Services", Proceeding of IADIS International Conference, **(2007)**.

[5] Viola, Paul, Jones and J. Michael, "Robust Real-Time Face Detection, International Journal of Computer Vision", vol. 54(2), pp. 137-154, Springer Netherlands, **(2004)**.

[6] OpenCV, http://opencv.willowgarage.com/wiki/.

[7] Yujinrobot iClebo, http://iclebo.com/english.

Author

Yong-Ho Seo received his BS and MS degrees from the Department of Electrical Engineering and Computer Science, KAIST, in 1999 and 2001, respectively. He also received a PhD degree at the Artificial Intelligence and Media Laboratory, KAIST, in 2007. He was an Intern Researcher at the Robotics Group, Microsoft Research, Redmond, WA in 2007. He was a consultant at Qualcomm CDMA Technologies, San Diego, CA in 2008. He is currently a Professor of the Department of Intelligent Robot Engineering, Mokwon University. His research interests include humanoid robot, human-robot interaction, robot vision and wearable computing.