

Detecting Moving Objects with an Omnidirectional Camera Based on Adaptive Background Subtraction

Kazumasa Yamazawa¹ and Naokazu Yokoya¹

¹ Nara Institute of Science and Technology, Graduate School of Information Science,
8916-5 Takayama, Ikoma, Nara, Japan
{yamazawa, Yokoya}@is.aist-nara.ac.jp
<http://yokoya.aist-nara.ac.jp/index.html>

Abstract. This paper describes a new method for detecting moving objects with an omnidirectional camera. The proposed method uses a fixed omnidirectional camera which can acquire images surrounding the camera and a background subtraction technique based on adaptively updating background images. In subtracting the background image from the input image, we consider the oscillation of intensity so that the proposed method can robustly detect moving objects under the condition that there are fluorescent lights and incoming lights from windows. In experiments, moving objects have been successfully detected in indoor environments.

1 Introduction

When we detect moving objects with a common video camera, the field of view is limited. There are some methods for solving this problem. Although most of the methods use a PTZ (panning, tilting and zooming) camera[1] or multiple cameras[2]. An omnidirectional camera[3,4] has a number of advantages. For example, if we use background subtraction for detecting moving objects, with a PTZ camera, we must synchronize the PTZ of the camera and the PTZ of the background image. When we use multiple cameras, the system becomes complex. On the other hand, an omnidirectional camera, which can acquire an omnidirectional view at a time, is suitable for detecting moving objects in the surrounding environment. However, most of conventional methods suffer from oscillations of intensity, such as fluorescent lights. This paper describes a new method for robustly detecting moving objects with an omnidirectional camera and a background subtraction technique based on adaptively updating background images.

2 Detecting Moving Objects

2.1 Outline of the method

The proposed method uses an omnidirectional camera and a background subtraction technique. The fixed omnidirectional camera acquires omnidirectional images sur-

rounding the camera. First, we estimate the background image and amplitude image from the first several frames of the input omnidirectional image sequence. After the background image and amplitude image are estimated, we detect moving objects from the input images, the background image and the amplitude image, updating the background image and the amplitude image. Next, we generate common perspective images of moving objects, calculating fan-shaped bounding-boxes which enclose the detected moving objects in the omnidirectional image.

2.2. Omnidirectional camera: HyperOmni Vision

We use HyperOmni Vision[5] as an omnidirectional camera in the present work. HyperOmni Vision is composed of a hyperboloidal mirror and a video camera as illustrated in Fig.1. The camera acquires an omnidirectional scene reflected by the hyperboloidal mirror. The hyperboloidal mirror is constructed of a hyperboloid of two sheets of revolution, which has two focal points (O_M and O_C). The camera lens center is fixed at the focal point O_C . Given a world coordinate (X, Y, Z) and an image coordinate (x, y) as shown in the Fig.1(b), the shape of hyperboloidal mirror and the two focal points are represented as follows.

$$\begin{aligned}
 \text{Mirror Shape : } & \frac{X^2 + Y^2}{a^2} - \frac{Z^2}{b^2} = -1, \\
 \text{Inner focal point } O_M : & (0, 0, +c), \\
 \text{Outer focal point } O_C \text{ (Camera lens center): } & (0, 0, -c), \\
 \text{where } c = & \sqrt{a^2 + b^2}.
 \end{aligned} \tag{1}$$

A ray going from the point $P(X, Y, Z)$ in 3D toward the inner focal point O_M is reflected by the mirror and passes through the outer focal point O_C intersecting the image plane at the point $p(x, y)$. Therefore, the projection of HyperOmni Vision is not common planar perspective, but satisfies the single viewpoint constraint. The relationship between $P(X, Y, Z)$ and $p(x, y)$ is given by:

$$\begin{aligned}
 x = & \frac{f(b^2 - c^2)X}{(b^2 + c^2)(Z - c) - 2bc\sqrt{X^2 + Y^2 + (Z - c)^2}}, \\
 y = & \frac{f(b^2 - c^2)Y}{(b^2 + c^2)(Z - c) - 2bc\sqrt{X^2 + Y^2 + (Z - c)^2}}.
 \end{aligned} \tag{2}$$

By using the above equation, an omnidirectional input image can be converted to a common perspective image at the viewpoint O_M [6] (see Fig.2). In this study, we generate common perspective images of detected moving objects.

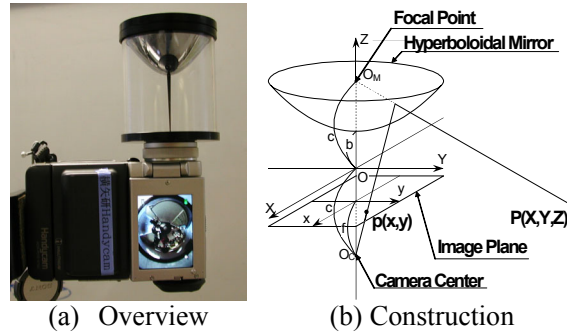


Fig. 1. HyperOmni Vision

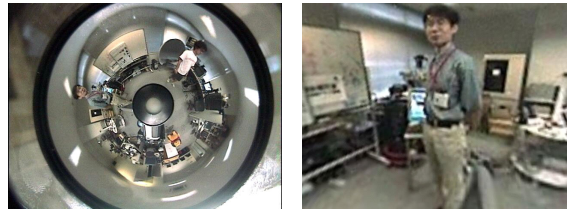


Fig. 2. Input image (*left*) and generated image (*right*)

2.3. Detecting moving object areas from omni-directional input images

The proposed moving object detection method is based on background subtraction. The background subtraction is carried out at each pixel.

We model the intensity of background pixel as follows.

$$I = I_{ave} + \sigma \sin(2\pi \omega t) + k * noise, \quad (3)$$

where

I : intensity of background pixel,

I_{ave} : average of I in time,

σ : amplitude of oscillation of intensity,

ω : frequency of oscillation of intensity,

t : time,

k : -1, 0, or +1,

$noise$: noise which is dependent only on the video camera.

The term $\sigma \sin(2\pi \omega t)$ in Eq.(3) means a flicker of a fluorescent light, CRT, etc. The term $k * noise$ means a noise which is dependent only on the video camera due to gain-up etc. Therefore, the intensity has the range of:

$$I_{ave} - \sigma - noise \leq I \leq I_{ave} + \sigma + noise \quad (4)$$

If the intensity I of the pixel is outside the range, we decide that the pixel is a part of a moving object. First, we predetermine the parameter $noise$ which is dependent only on the video camera setup. Next, we consider the slow change of the background. When a pixel is determined to represent the background, the parameters I_{ave} and σ are updated by the following equation.

$$\begin{aligned} I'_{ave} &= I_{ave} * (n-1)/n + I * 1/n, \\ \sigma' &= \sigma * (n-1)/n + 2 * (I - I_{ave}) * 1/n, \end{aligned} \quad (5)$$

where

n : parameter of the update speed .

(When n increases, updating speed decreases.)

Eq. (5) is computed for all the pixels determined as the background on each frame. We use I'_{ave} and σ' as I_{ave} and σ in the next frame.

For a pixel estimated at a part of moving object, we do not calculate Eq. (5). However, in order to handle cases that an object is put or taken and the rapid change of background, when the pixel is estimated as a part of moving objects, the parameter σ is updated using the following equation.

$$\sigma' = \sigma * (m-1)/m + 2 * (I - I_{ave}) * 1/m, \quad (6)$$

where

m : parameter for the fading out speed of the detected object.

(When m increases, the speed of fading out decreases.)

Normally, we determine m larger than n . Therefore, if the detected moving object stays at a position, σ increases, and then the detected moving object fade out. If we set m infinity, the detected new objects do not fade out.

2.4. Generating common perspective images of detected objects

The method of generating common perspective images of detected moving objects is as follows.

1. Compute a histogram with respect to the longitude direction θ of the detected moving objects.
2. Detect blobs whose values are more than 0 in the histogram as the ranges of the longitude of the moving objects.
3. Compute the maximum and minimum latitude of the detected moving object for each blob.
4. Generate the common perspective images based on the computed ranges of the longitude and the latitude.

The generation of perspective images focusing on detected objects is important in some surveillance applications and can be carried out in real time by using the technique in [6].

3. Experiment

We have prototyped a surveillance system using the proposed method on a Linux PC. The specification of the PC is as follows.

OS: Redhat 7.1

CPU: Pentium4 2GHz

Memory: 512MB

The left of Fig. 3 shows the screen of the system. The screen is divided into four images. The three images show common perspective images of the largest three moving objects. The other shows an input omnidirectional image, histogram, etc.

We set the parameters as follows.

$0 \leq I \leq 255$, $noise = 4$, $n=360$, and $m=1080$.

The parameters are determined empirically. However, the selected parameters have been proven to be suitable for different indoor environments such as lobby and meeting rooms.

In this experiment, there were two walkers and a putted doll under the condition that there were fluorescent lights and incoming lights from windows (see Fig.3 right). Figure 4 shows the stream of screen images. The system can detect moving objects robustly and can update images at intervals of 0.2 sec.

4. Conclusions

We have proposed a new method for robustly detecting moving objects with an omnidirectional video camera. The proposed method uses a background subtraction technique which is based on adaptively updating background images. In subtracting the background image from the input image, we consider the oscillation of intensity so that the proposed method can robustly detect moving objects under the condition that there are fluorescent lights and incoming lights from windows. We have also developed a prototype system for surveillance application. The system has been successfully demonstrated in indoor environments.

The future work includes the improvement of updating the areas of detected moving objects and the tracking of detected moving objects between adjacent frames.

References

1. T.Wada and T.Matsuyama: Appearance sphere: Background model for pan-tilt-zoom camera, Proc. ICPR96, vol.A (1996) 718-722
2. I.Mikic, S.Santini, and R.Jain: Tracking Objects in 3D using Multiple Camera Views, Proc. ACCV2000 (2000) 234-239
3. R.Miki, N.Yokoya, K.Yamazawa, and H.Takemura: A Real-time Surveillance and Monitoring System Using Multiple Omnidirectional Video Cameras, Proc. ACCV2000 (2000) 528-534

4. T.Sogo, H.Ishiguro and M.M.Trivedi: Real-time target localization and tracking by N-ocular stereo, Proc. IEEE Workshop on Omnidirectional Vision (2000) 153-160
5. K.Yamazawa, Y.Yagi, and M.Yachida: Obstacle Detection with Omnidirectional Image Sensor HyperOmni Vision, Proc. Int. Conf. on Robotics and Automation (1995) 1062-1067
6. Y.Onoe, K.Yamazawa, H.Takemura, and N.Yokoya: Telepresence by Real-Time View-Dependent Image Gen-eration from Omnidirectional Video Streams, Computer Vision and Image Understanding, Vol. 71, No.2 (1998) 154-165

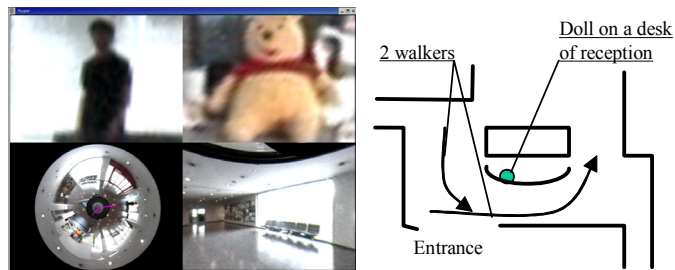


Fig. 3. Screen image of the system (*left*) and Experimental environment (*right*)

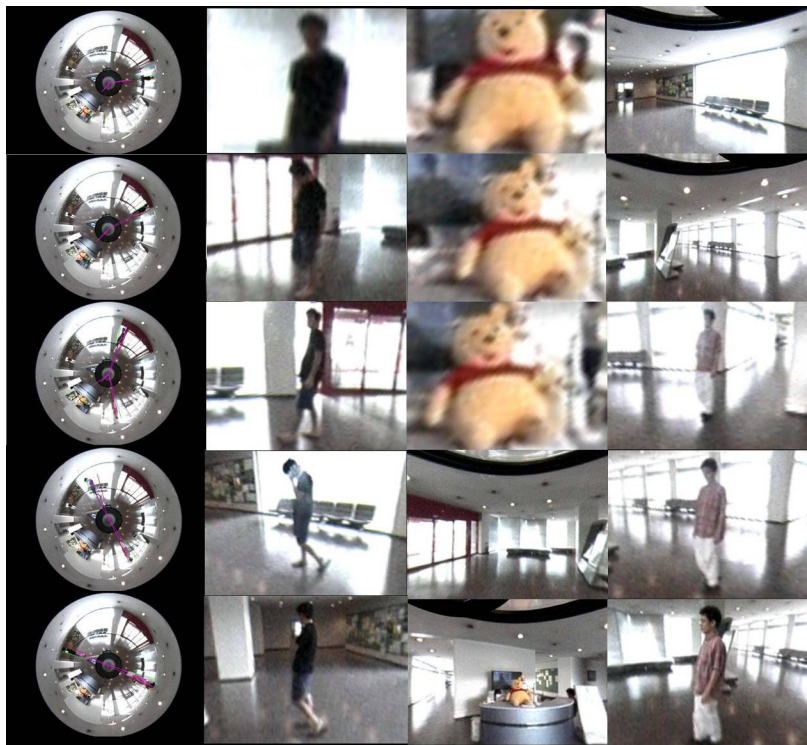


Fig. 4. Stream of input image and detected object views