Image and Vision Computing 38 (2015) 52–64

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

A Self-adaptive CodeBook (SACB) model for real-time background subtraction ☆

Munir Shah *, Jeremiah D. Deng, Brendon J. Woodford

Department of Information Science, University of Otago, New Zealand

ARTICLE INFO

Article history:
Received 4 October 2013
Received in revised form 7 November 2014
Accepted 7 February 2015
Available online 16 April 2015

Keywords:
Background subtraction
Video processing
Parameter learning
Background modeling
CodeBook model

ABSTRACT

Effective and efficient background subtraction is important to a number of computer vision tasks. In this paper, we introduce a new background model that integrates several new techniques to address key challenges for background modeling for moving object detection in videos. The novel features of our proposed Self-adaptive CodeBook (SACB) background model are: a more effective color model using YCbCr color space, a statistical parameter estimation method, and a new algorithm for adding new background codewords into the permanent model and deleting noisy codewords from the models. Also, a new block-based approach is introduced to exploit the local spatial information. The proposed model is rigorously tested and has shown significant performance improvements over several previous models.

1. Introduction

Foreground objects in videos carry important information and are the main focus in most of the high-level computer vision applications. Usually, static or quasi-periodic regions of the visual scene are regarded as background, while moving objects in the scene are considered as foreground objects [1–5]. The precise localization of foreground objects from a video is a typical first step in many computer vision applications [6].

Background subtraction is a common approach used for separating foreground objects from the background. The background subtraction framework comprises four steps: preprocessing, background modeling, foreground detection, and post-processing [1–4,7–8] as shown in Fig. 1.

The preprocessing step includes removal of imaging noises by applying smoothing filters and sub-sampling to achieve fast processing. Background modeling is the core of any background subtraction algorithms and in this step an accurate model of the background of a visual scene is built. The background model can be a single reference image or an advanced statistical model, depending on the complexity of the scene. A good background model should be robust against video sensor noises and environmental changes in the background, but at the same time it should be sensitive enough to detect all objects of interest.

In the foreground detection phase, each video frame is compared against the background model, and those pixels significantly deviating from the background model are classified as the foreground. Finally, blob analysis and image denoising methods are applied as the post-processing step to refine the foreground map. The final foreground map contains a mask for foreground objects only.

Real-world video sequences contain several difficult situations, which makes background modeling and foreground detection a difficult problem [9–13]. The major challenges for background modeling techniques are as follows:

- Camera shaking. If the video is taken with a hand-held camera, there are high chances that it will have an effect of camera shaking. Even with a slight change of the camera angle, pixel-level dynamics of video sequence can change quite drastically. Background modeling techniques not taking camera movement into account may give a large number of false positive results, i.e., labeling background pixels as foreground.
- Dynamic backgrounds. The video background can be dynamic, for example, water flowing in the river or trees swaying in the background. This motion is usually quasi-periodic, and thus gives multiple layers of background, which demands multi-model background modeling techniques. There are many environmental factors in outdoor videos that control the behavior of background layers, and at times it changes from quasi-periodic to irregular motion. This irregular background motion is difficult to distinguish from foreground motions.
- Illumination changes. Illumination changes can be local (e.g., moving objects cast shadows or highlights due to the reflection from the

☆ This paper has been recommended for acceptance by Massimo Piccardi.

* Corresponding author.
E-mail addresses: munirsha@gmail.com (M. Shah), jeremiah.deng@otago.ac.nz (J.D. Deng), brendon.woodford@otago.ac.nz (B.J. Woodford).

http://dx.doi.org/10.1016/j.imavis.2015.02.001
0262-8856/

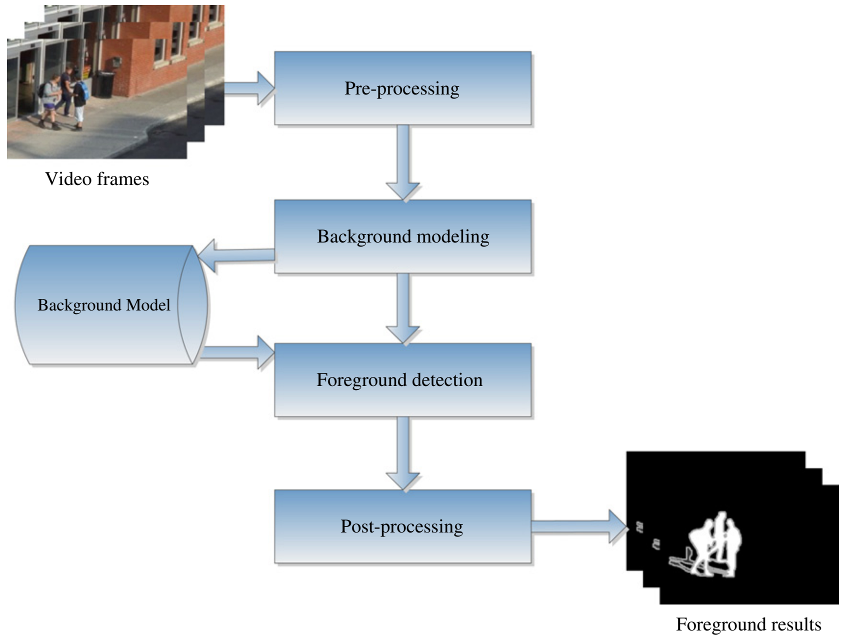

Fig. 1. Background subtraction process to detect foreground objects in videos.

surface of moving objects), or global (e.g., a person entering a room and switching on a light will change the global illumination conditions of the scene). This type of illumination changes (both local and global) is usually rapid and traditional adaptive background modeling techniques cannot deal with them effectively.
- Real-time and online processing. Most computer vision applications demand real-time and online processing, and clean labeled training data are not usually available beforehand. This suggests that background modeling techniques should be based on online learning and have the capabilities to deal with noisy training data. In addition, this imposes the constraint for real-time processing.

In this paper, a Self-adaptive CodeBook (SACB) background model is presented for moving object segmentation in a video. The proposed SACB model is designed to address the limitations of the CodeBook model and to exploit the close proximity within the local neighborhood using a new block based approach. The major components of the paper are a more effective color model, a statistical parameter estimation method, a new algorithm for keeping the background model compact, and a block based approach to use local spatial information.

The rest of the paper is organized as follows. Section 2 describes the CodeBook model and highlights its limitations. Our proposed model is presented in Section 3. Section 4 describes the evaluation methodology, while experimental results and discussions are presented in Section 5. Finally, Section 6 highlights the main conclusion of this study and points to future directions.

2. The CodeBook model

The CodeBook is one of the popular real-time background models for moving object detection [14]. Several of its enhancements have been proposed [15,16]. The basic CodeBook model has two main phases: an initial training phase where the codebook is constructed and a foreground detection phase.

In the initial training phase, a new sample is compared with existing codewords in codebook to determine the matching codeword. If matching codeword is found, it is updated. Otherwise, new codewords are created and added into the codebook. In this model, color distortion and brightness bound are used in the criterion to determine the best matching codeword [14].

The codebook obtained in the previous step may become quite large because it may include moving objects in the training sequence. Thus, at the end of the training process, every codeword is analyzed and the codewords having large λ the longest interval during which it has not re-occurred are considered as foreground codewords, thus filtered-out from the codebook.

The basic Codebook model is effective enough to model a scene where geometry of the scene does not change over time. However, in reality scenes may change after the training, for instance, a parked car starts moving, or the sun comes out. To cope with these types of situations, a layered CodeBook model [14] was proposed by introducing an additional layer called the cache Codebook (H).

In the layered model, during the foreground detection phase, the incoming pixel's value is compared against the permanent CodeBook model. If a matching codeword is found, the pixel is classified as background and is updated. Otherwise, the pixel is classified as foreground and checked against the cache CodeBook model. If a matching codeword is found in the cache CodeBook model, the model is updated. Otherwise, a new codeword is created, and added into the cache CodeBook H. Then, codewords staying long enough in the cache CodeBook are moved to the permanent CodeBook model. After that, noisy codewords are removed from both cache and permanent CodeBook models [14].

The CodeBook model [14], and its enhancements [15–17] perform better than a number of state-of-the-art methods both in terms of segmentation accuracy and processing time. The cylindrical color model, unconstrained training and novel matching criteria are some of its distinguished features. However, it is a parametric model with the following key parameters:

- The thresholds used to filter out noisy codewords (T_{del}).
- A threshold used for moving codewords from cache to the main codebook (T_{add}).
- Color distortion thresholds used in the training (ε_1) and segmentation (ε_2) phases.
- A learning rate (α).

It is not always adequate to use predefined values for these parameters. Thus, manual parameter tuning is required to achieve good results for a typical scene, which is often a cumbersome and tricky