Pattern Recognition Letters 100 (2017) 110–116

Contents lists available at ScienceDirect

Pattern Recognition Letters

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

Adaptive bootstrapping management by keypoint clustering for background initialization

Danilo Avola[a], Marco Bernardi[b,*], Luigi Cinque[b], Gian Luca Foresti[a], Cristiano Massaroni[b]

[a] Department of Mathematics, Computer Science and Physics, University of Udine, Via delle Scienze 206, 33100 Udine, Italy
[b] Department of Computer Science, Sapienza University, Via Salaria 113, 00198 Rome, Italy

ARTICLE INFO

Article history:
Received 26 May 2017
Available online 20 October 2017

MSC:
41A05
41A10
65D05
65D17

Keywords:
Background initialization
Background modeling
Keypoint clustering
Foreground detection
Bootstrapping

ABSTRACT

The availability of a background model that describes the scene is a prerequisite for many computer vision applications. In several situations, the model cannot be easily generated when the background contains some foreground objects (i.e., bootstrapping problem). In this letter, an Adaptive Bootstrapping Management (ABM) method, based on keypoint clustering, is proposed to model the background on video sequences acquired by mobile and static cameras. First, keypoints are detected on each frame by the A-KAZE feature extractor, then Density-Based Spatial Clustering of Application with Noise (DBSCAN) is used to find keypoint clusters. These clusters represent the candidate regions of foreground elements inside the scene. The ABM method manages the scene changes generated by foreground elements, both in the background model initialization, managing the bootstrapping problem, and in the background model updating. Moreover, it achieves good results with both mobile and static cameras and it requires a small number of frames to initialize the background model.

1. Introduction

Background modeling is a way to represent the background of a scene through construction of a model. Once that background representation is chosen, typically, there are two main steps [4]: *Background Initialization* and *Background Updating*. The first regards the initialization of the model based on the analysed scene. Instead, the second adapts the model based on the changes of the scene. In the last years, most works have focused on the background updating whereas limited attention is given to the background initialization, although it is a complex task. A significant problem in the background initialization is represented by *bootstrapping* [26]. Bootstrapping management is the process of creating the background model of a scene when foreground elements are contained in the video sequence. Most existing methods [4] work only with static cameras, while few works deal with the bootstrapping management with mobile cameras. In this letter, we extend the work presented in [2] (focused on foreground detection) to propose a new solution for the background modeling with mobile cameras, mainly focused on the background initialization and bootstrapping management. The ABM method is based on keypoint clustering which uses the A-KAZE [1] feature extractor to detect interesting keypoints and DBSCAN [9] to determine keypoint clusters useful to distinguish foreground elements from background.

The proposed ABM method, differently from the solution presented in [2], does not employ the image stitching algorithm for the model updating, but it uses a simple and more effective pixel upgrade process with a reduced time complexity (in the order of quadratic complexity). Moreover, the ABM method, in addition to the steps proposed in [2], introduces a tracking procedure of the foreground keypoints along the frame sequence able to reduce the number of false positives in the background keypoint estimation. This fact requires that, in the analysed scene, the foreground elements cannot appear or disappear without performing movements, as normally occur in the real situation (e.g., a car leaves a parking lot or people come in and out of the scene). The keypoints extracted from the background are used to manage distortions that may occur in video sequences acquired by a mobile camera. The A-KAZE feature extractor is used for its high speed and accuracy [1]. DBSCAN algorithm has been selected because it does not require a priori knowledge of the cluster number as other standard clustering algorithms (e.g., K-Means or model based clustering algorithms). The main contributions of the proposed ABM method are:

1. Adaptivity in managing bootstrapping problem and the background model updating when a background object becomes foreground (e.g., a car leaves the parking lot).

* Corresponding author.
E-mail address: bernardi@di.uniroma1.it (M. Bernardi).

https://doi.org/10.1016/j.patrec.2017.10.029
0167-8655/

2. Need for a small number of frames to initialize and update the background model. In this letter, we present an analysis about the frame number required by the ABM method based on the movement of background pixels.
3. Unlike the majority of state-of-the-art methods that can work only with static cameras, it is able to work well also with video sequences acquired by mobile cameras and it does not require complex training procedures to create and update the background model.

The rest of the paper is organized as follows. In Section 2, a brief state-of-the-art is presented. The proposed ABM method is described in Section 3. Experimental results are presented in Section 4. Section 5 contains the conclusions.

2. Related work

In recent years, lots of methods were proposed to manage the background initialization process, which can be divided into two categories: methods using static camera [7,12,14,15,20,24,27] and mobile camera [3,10,17,29].

Methods based on static camera are applied in scenarios that need to be constantly monitored (e.g., parking, airport, etc.). According to [4], they can be classified on the basis of the applied methodology. The most popular class of methods is based on temporal statistics. The temporal median filter [8] computes each pixel of the background as the mean of the pixels at the same location across all the frames. This method does not have a systematic strategy to determine the degree of completion of background model. Consequently, it sometimes generates unreliable background models. Subsequences of Stable Intensity (SSI) methods rely on the assumption that a background value always has the longest stable value. For each image region, a non-overlapping subsequence with similar intensity values is chosen. These regions are used to generate iteratively the background model based on suitable spatial consistency criteria. The approach presented in Ortego et al. [24] is based on a temporal-spatial block-level procedure for background estimation to cope with moving and stationary objects. Instead, Laugraud et al. [18] have proposed a two step method. First, it estimates the motion in order to recognize background candidate regions. Then, the background model is obtained by applying the median filter to the subset of these regions. However, these methods suffer in presence of stationary entities, such as standing person or sleeping person. Another interesting approach is based on the use of Neural Networks (NNs) for automatically learn the background model in an unsupervised or supervised way. Recently, De Gregorio and Giordano [12] have proposed a pixel-based background initialization method based on Weightless Neural Networks (WNNs). Typically, pixel-based methods provide excellent performance but have some difficulties in background maintenance due to well known problems, such as local or global illumination changes. The background initialization can be also formulated as an optimal labeling problem. Support Vector Machines (SVMs) are used to determine which image blocks are part of the background [20]. These methods perform well but a lot of data to train the classifier is required. Missing data reconstruction methods consider the background initialization process as a reconstruction problem from missing data. Matrix completion algorithms [27] have shown to be suitable for background initialization. These approaches have a nice potential in background initialization process but present some color divergence artifacts because the reconstruction process is done for each individual color channel. Another two-phase approach is the Iterative Model Completion (IMC) method that first selects regions as reference and then iteratively completes the background model on the base of suitable spatial consistency criteria. In [7], a patch-based method exploits

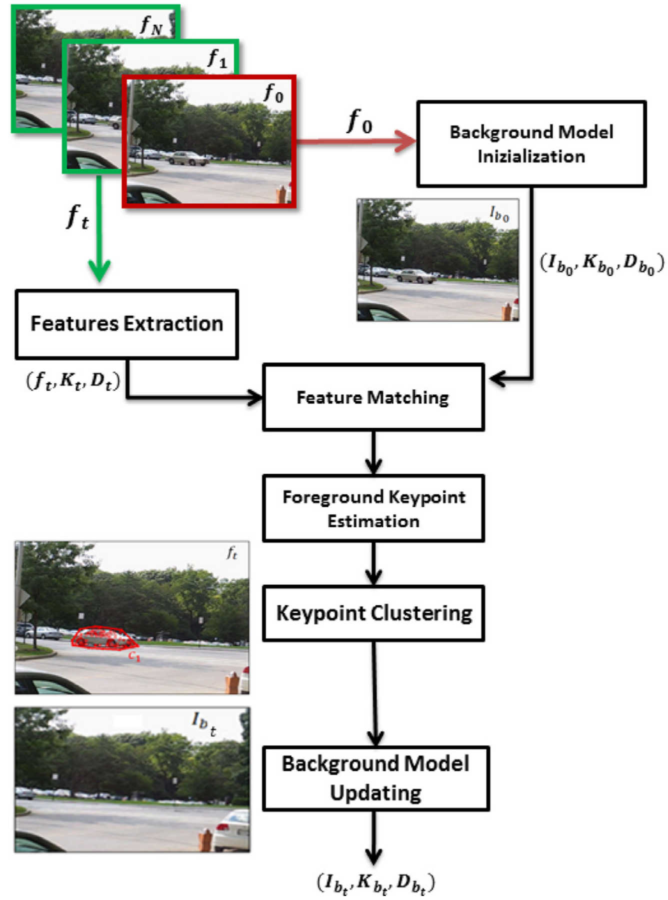

Fig. 1. Logical architecture of the proposed ABM method. The output background model can be used to perform several scopes, such as foreground detection or tracking.

spatio-temporal consistency of the static background. A disadvantage of this class of methods is the high computational complexity.

The use of mobile sensors, such as Pan-Tilt-Zoom (PTZ) cameras, allows the supervision of larger areas, but it introduces the need to manage the problem of the scene motion. Background initialization in presence of mobile cameras is an interesting challenge. Machine learning approaches are among of most promising methods in this context. Ferone et al. [10] describe a method where the background model automatically adapts in a self-organizing way to changes in the scene background. In [25], a background model is generated by using Bag of Features (BoFs) and a SVM. However, these methods do not directly address the bootstrapping management problem. The proposed work belongs to the class of methods using mobile camera. Unlike existing approaches, it works well with both static and mobile camera and it is suitable for bootstrapping management. Moreover, it requires a reduced number of frames to initialize and update the background model.

3. Adaptive bootstrapping management method

The ABM method receives in input a video sequence and produces in output a background model of the scene updated over time. Fig. 1 shows the logical architecture of the proposed ABM method. The keypoints and the associated descriptors are extracted from each frame of the input video sequence by using the A-KAZE algorithm. The feature matching module applies the K-Nearest Neighbors (K-NN) method [19] to perform a comparison among the keypoints and the descriptors extracted from each input frame and the background model. Keypoints provide