Pattern Recognition 48 (2015) 3527–3541

Contents lists available at ScienceDirect

Pattern Recognition

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

A grid-based tracker for erratic targets ☆

Qian Sang [a], Zongli Lin [b,*], Scott T. Acton [b]

[a] Algorithm Research, Aol Platforms R&D, 395 Page Mill Road, Palo Alto, CA 94306, USA
[b] Charles L. Brown Department of Electrical and Computer Engineering, University of Virginia, P.O. Box 400743, Charlottesville, VA 22904-4743, USA

ARTICLE INFO

Article history:
Received 24 December 2013
Received in revised form
30 March 2015
Accepted 14 May 2015
Available online 23 May 2015

Keywords:
Target tracking
Visual tracking
Grid-based Bayesian approach
Erratic target motion
Predication model adaptation
Iterative position estimation

ABSTRACT

The grid-based Bayesian tracker employs a novel sample generation and weighting mechanism that achieves significantly improved visual tracking performance (in terms of accuracy, robustness, and computational burden) over existing active contour trackers and Monte Carlo trackers. This paper presents a method to enhance its capability in accommodating the tracking of targets in video with erratic motion, by introducing adaptation in the motion model and iterative position estimation. Tracking performance of the resulting algorithm is compared with the grid-based Bayesian tracker in the context of leukocyte tracking, UAV-based vehicle tracking, and *Drosophila* larva tracking to demonstrate its effectiveness in dealing with erratic target movement.

1. Introduction

Automated target tracking in real time is crucial to a wide range of practical video and image applications such as aerial video surveillance [13] and visual servo control [10]. With efficient automatic target tracking, a formidable amount of video data (online or offline) can be processed with high accuracy without human intervention.

Existing visual tracking algorithms in the literature can be mainly classified as either deterministic or stochastic [20]. A deterministic tracker views target tracking as an optimization problem, with the objective of seeking the minimum of a properly chosen cost functional. A typical example is the active contour model [12,17], also called a *snake*, for which an energy functional is defined that takes into account the contour's tension and rigidity (internal energy), as well as the target features of interest (external energy). An external force generated by the target features in the spatial domain of an image pushes the contour towards target features, which correspond to a minimum of the energy functional. Thus the contour evolves to lock onto the desired target. In this case, tracking is achieved by guiding the contour from its location in the current frame to the target in the next frame employing external forces generated by the target features. Various external forces have been proposed to enhance the capture range of the traditional active contour [12], e.g., gradient vector flow [19] and vector field convolution [14].

In the stochastic approach to visual tracking, the problem is cast into an estimation procedure. The starting point is a state space model consisting of a state transition equation and a measurement equation, corrupted by the state and measurement noise processes, respectively. Due to the randomness of the noise processes, the tracking problem fits into a Bayesian framework with a recursive estimation procedure. In each frame, the state transition equation provides a predicted probability density function (pdf) of the target state (position, velocity, and acceleration) in the next frame from an estimate of its current pdf, while the measurement equation determines how to update this prior pdf with a new state measurement to obtain the posterior pdf. Depending on assumptions imposed on the state space model, the estimation problem has different solutions. In the simplest case when the state space model is linear and the noise processes are Gaussian, the Kalman filter [7,9] provides an optimal solution to the tracking problem. If the model is allowed to be nonlinear, the extended Kalman filter (EKF) [18] gives a Gaussian approximation of the target state distribution by linearizing the nonlinear model at local means. In the general case when the model is nonlinear and the noise processes are non-Gaussian, particle filters are employed to approximately represent the density function [1,8].

A particle filter is a sequential Monte Carlo method [3,15,5,6], which implements a Bayesian filter recursively using Monte Carlo simulations. The idea is to generate a set of samples (particles) with associated weights in a random way for the construction of a probability density function to approximate the posterior density

☆The work was supported in part by Army Research Office under Contract no. W911NF1010367.

* Corresponding author. Tel.: +1 434 924 6342; fax: +1 434 924 8818.
E-mail addresses: qs2v@virginia.edu (Q. Sang), zl5y@virginia.edu (Z. Lin), sa4e@virginia.edu (S.T. Acton).

http://dx.doi.org/10.1016/j.patcog.2015.05.014
0031-3203/

of the target state, which is then estimated as a weighted sum of this set of samples. In particular, two major tasks need to be accomplished for successful target tracking: target position prediction and target localization. Position prediction involves computing the target position in the next frame (the "predicted position") based on estimated positions in current and previous frames. The localization procedure designates a search region around the predicted location and estimates the target location (the "estimated position") in the next frame using a certain target feature detection algorithm. For stable and robust tracking performance, a large number of samples are usually required, leading to a substantial computational burden. Such an expense may hinder its implementation in real-time tracking applications.

In contrast to a particle filter, a Grid-based Bayesian Approach (GBA) for robust visual tracking has recently been developed in [16], which proposes a novel method for *deterministic* sample generation and sample weighting. The GBA tracker has been shown to drastically outperform the active contour tracker [17] or a Monte Carlo tracker [4] in the tracking of leukocytes in vivo and vehicles observed from unmanned aerial vehicle (UAV) videos. In particular, its tracking performance is robust to image jitter, background movement, and image noise and clutter, in terms of the total number of frames tracked and the number of sequences with all frames tracked. Tracking accuracy is improved with significantly reduced average root mean square error (RMSE). Furthermore, the GBA tracker can be executed at a speed 100 times faster than the other two trackers. Another distinguishing feature of the GBA tracker is that the target motion model used for position prediction is capable of dealing with target occlusion.

However, a presumption of the GBA tracker is the smoothness of the motion trajectory of the target to be tracked. For effective tracking, the movement of the target cannot be too erratic, i.e., the change in its velocity cannot be abrupt. This is partly due to the smoothing effect in the prediction model, which, when the target is moving erratically, may not generate a prediction that provides sufficient information for successful tracking. The predicted position in the previous frames does not reflect the recent target motion behavior in this case. In addition, an erratically moving target may escape from the coverage of the sample grid, thus getting lost.

As an attempt to cope with erratic target movement, in this paper we present an Erratic Target Grid-based Tracker (ETGT) to enhance the capability of the GBA tracker by introducing adaptation in the target motion model and iterative position estimation, when erratic target movement is detected. We will demonstrate the improved tracking performance over the GBA tracker [16] in tracking a single leukocyte in vivo, ground vehicle target observed from UAV videos, and *Drosophila* larvae, all undergoing abrupt changes in motion.

The reminder of the paper is organized as follows. In Section 2, some preliminaries on the Bayesian tracking approach and the GBA tracking algorithm in particular are presented. Section 3 describes the ETGT tracking algorithm. Experimental results are provided in Section 4. Section 5 concludes the paper.

2. Preliminaries

For a Bayesian tracking approach in general, consider the state transition model (1) and the measurement model (2):

$$\boldsymbol{x}_k = a(\boldsymbol{x}_{k-1}, u_{k-1}) \tag{1}$$

$$\boldsymbol{y}_k = b(\boldsymbol{x}_k, v_k) \tag{2}$$

where the state variable $\boldsymbol{x}_k$ is the real target position, u_{k-1} is the state noise process, $\boldsymbol{y}_k$ is the measurement of $\boldsymbol{x}_k$, corrupted by the measurement noise process v_k, with k denoting the frame number. The objective is to compute $\boldsymbol{x}_{k+1}$, the target position in the next frame, based on information from the model and all available measurements up to the $(k+1)$ th frame, i.e., $\boldsymbol{y}_{1:k+1} = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_{k+1}\}$.

The target position $\boldsymbol{x}_{k+1}$ is a random variable, and an estimate is its expected value computed with the so-called posterior probability density function (pdf) $p(\boldsymbol{x}_{k+1}|\boldsymbol{y}_{1:k+1})$. In the Bayesian tracking procedure, $p(\boldsymbol{x}_{k+1}|\boldsymbol{y}_{1:k+1})$ is obtained recursively in two main steps as briefly outlined below. Let $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1})$ and $p(\boldsymbol{x}_k|\boldsymbol{y}_k)$ denote the state transition pdf and the observation pdf, respectively. The first step involves computing the prior pdf $p(\boldsymbol{x}_{k+1}|\boldsymbol{y}_{1:k})$ based on $p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})$ (computed recursively for the kth frame) and the transition pdf $p(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k)$, through the Chapman–Kolmogorov equation

$$p(\boldsymbol{x}_{k+1}|\boldsymbol{y}_{1:k}) = \int p(\boldsymbol{x}_{k+1}|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\boldsymbol{y}_{1:k})\, d\boldsymbol{x}_k$$

This density gives target position prediction. Then with the measurement of the target position in the $(k+1)$ th frame, $\boldsymbol{y}_{k+1}$, the prior pdf is updated using Bayes' law to compute the posterior pdf as

$$p(\boldsymbol{x}_{k+1}|\boldsymbol{y}_{1:k+1}) = \frac{p(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1}) p(\boldsymbol{x}_{k+1}|\boldsymbol{y}_{1:k})}{p(\boldsymbol{y}_{k+1}|\boldsymbol{y}_{1:k})}$$

where

$$p(\boldsymbol{y}_{k+1}|\boldsymbol{y}_{1:k}) = \int p(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1}) p(\boldsymbol{x}_{k+1}|\boldsymbol{y}_{1:k})\, d\boldsymbol{x}_{k+1}$$

In particle filter methods, the continuous pdf function $p(\boldsymbol{x}_{k+1}|\boldsymbol{y}_{1:k+1})$ is approximated by a discrete, finite set of samples with associated weights: $\{s_{k+1}^{(m)}, w_{k+1}^{(m)}\}_{m=1}^{M}$, where $s_{k+1}^{(m)}$ is the mth sample with its associated weight

$$w_{k+1}^{(m)} = \frac{p\left(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1} = s_{k+1}^{(m)}\right)}{\sum_{i=1}^{M} p\left(\boldsymbol{y}_{k+1}|\boldsymbol{x}_{k+1} = s_{k+1}^{(i)}\right)}$$

and M is the sample size. A weighted average of the samples is taken as the estimated position of the target, i.e.,

$$\boldsymbol{x}_{k+1} = E_{p(\boldsymbol{x}_{k+1}|\boldsymbol{y}_{1:k+1})}(\boldsymbol{x}) \approx \sum_{i=1}^{M} w_{k+1}^{(i)} s_{k+1}^{(i)}$$

The Grid-based Bayesian Approach (GBA) to visual tracking [16] is motivated by the idea behind the weighted sampling of the particle filters. In the GBA tracking framework, a state transition model is so constructed as to cope with target occlusion and to obtain a position prediction more robust to image jitter. Around the predicted position, samples are generated in a deterministic way by gridding within an ellipsoid. They are weighted according to their distance to the predicted position and the number of detected target features (e.g., boundary) after applying some feature detection algorithm. The sample set with associated weights is considered as an approximation to the distribution of the target state variable (i.e., position), and its weighted sum is computed to be the estimated position. A review of the GBA tracker is given next.

2.1. Position prediction in the GBA tracker

A state transition model of the target is constructed first, which predicts its position in the next frame based on the estimated positions in previous frames. In particular, the following state transition model is applied

$$\begin{aligned} \overline{x}_{c,k+1} &= \hat{x}_{c,k} + (\hat{x}_{c,k} - \hat{x}_{c,k-n})/n \\ \overline{y}_{c,k+1} &= \hat{y}_{c,k} + (\hat{y}_{c,k} - \hat{y}_{c,k-n})/n \end{aligned} \tag{3}$$