Neurocomputing 183 (2016) 90–105

Contents lists available at ScienceDirect

Neurocomputing

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

Visual tracking with VG-RAM Weightless Neural Networks

Mariella Berger [a,*], Alberto F. De Souza [a], Jorcy de Oliveira Neto [a], Edilson de Aguiar [b], Thiago Oliveira-Santos [a]

[a] Departamento de Informática, Universidade Federal do Espírito Santo, Av. Fernando Ferrari 541, 29075-910 Vitória, ES, Brazil
[b] Departamento de Computação e Eletrônica, Universidade Federal do Espírito Santo, Rodovia BR 101 Norte km. 60, 29932-540 São Mateus, ES, Brazil

ARTICLE INFO

Article history:
Received 7 April 2014
Received in revised form
27 February 2015
Accepted 20 April 2015
Available online 1 January 2016

Keywords:
Weightless Neural Networks
VG-RAM
Visual Tracking
Saccade
Superior Colliculus

ABSTRACT

We present a biologically inspired long-term object tracking system based on Virtual Generalizing Random Access Memory (VG-RAM) Weightless Neural Networks (WNN). VG-RAM WNN is an effective machine learning technique that offers simple implementation and fast training. Our system models the biological saccadic eye movement, the transformation suffered by the images captured by the eyes from the retina to the Superior Colliculus (SC), and the response of SC neurons to previously seen patterns. We evaluated the performance of our system using a well-known visual tracking database. Our experimental results show that our approach is capable of reliably and efficiently track an object of interest in a video with accuracy equivalent or superior to related work.

1. Introduction

Humans are capable of visually track a large variety of objects efficiently even in the presence of challenging situations such as abrupt object motion, occlusions, changes in view point, changes in the background, and changes in the appearance of the object of interest, including non-rigid transformations. In spite of recent research advances [1–3], performing the same task with automatic systems is still challenging because specific algorithms have to be created to handle each one of these scenarios [1].

The visual tracking problem can be formulated as follows. Given a bounding box defining the object of interest in the first frame of a video, automatically determine the object's bounding box or indicate that the object is not visible in every frame that follows. The key challenges are that the object and background may change appearance after the initial frame, making it harder to detect the object later on. This problem is emphasized in long-term object tracking, since the chances of having large changes in appearance increase with the time. In addition, the object may be occluded in some parts of the video and reappear later in the sequence. Since objects might reappear in different locations, algorithms tailored to continuous tracking cannot be used and detection of the object is necessary. Object tracking has many practical applications including, but not limited to, robotic vision, human-computer interaction, automatic annotation of video, automated surveillance, traffic monitoring, and vehicle navigation. For reviews of the visual tracking literature, please refer to [1–3].

Algorithms for object tracking follow roughly two main approaches [2]: recursive tracking and tracking-by-detection. Recursive tracking methods estimate the current state of an object of interest by applying a transformation on the previous state based on measurements made on previous and current images. The recursive estimation depends on the state of the object in previous frames and is susceptible to error accumulation [2]. For instance, Lucas and Kanade [4] proposed a method for estimating optic flow within a window around pixels of the object of interest and Comaniciu et al. [5] proposed a real-time tracker based on *mean shift*. Tracking-by-detection methods estimate the object state considering measurements made on the current image only. This avoids error accumulation, but requires training an object detector beforehand. One example is the method proposed by Mustafa et al. [6] that generates synthetic views of an object by applying affine warping techniques to a single template and train an object detector on the warped images. Adaptive tracking-by-detection methods try to take advantage of the two approaches by updating the object detector online. We briefly describe some tracking-by-detection methods in the following.

The method presented by Avidan [7] integrates a support vector machine classifier into an optic-flow-based tracker. The technique proposed by Collins and Liu [8] treats tracking as a binary classification problem having either object of interest and background. Javed et al. [9] employ combination of discriminative and generative models in order to label incoming data and finally

* Corresponding author.
E-mail addresses: mberger@inf.ufes.br (M. Berger), alberto@lcad.inf.ufes.br (A.F. De Souza), jorcy@lcad.inf.ufes.br (J.d.O. Neto), edilson@lcad.inf.ufes.br (E. de Aguiar), thiago@lcad.inf.ufes.br (T. Oliveira-Santos).

http://dx.doi.org/10.1016/j.neucom.2015.04.127
0925-2312/

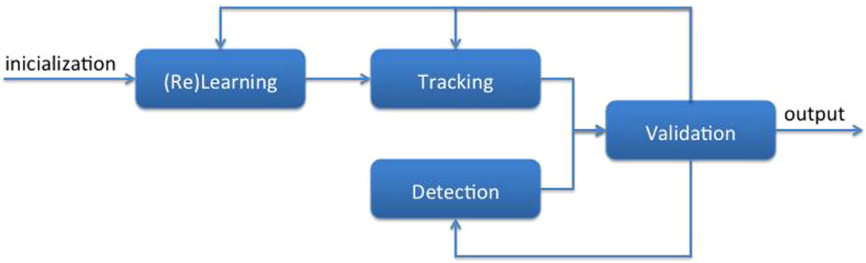

Fig. 1. Block diagram of the visual tracking neural system.

use it to improve an object detector. Ross et al. [10] incrementally learn a low-dimensional subspace representation and adapt it to changes in the appearance of the target. Adam et al. [11] propose *FragTrack,* a method that uses a static part-based appearance model based on integral histograms. Avidan [12] uses self-learning for boosting in order to update an ensemble classifier. Grabner et al. [13] employ a semi-supervised approach and enforces a prior on the first patch. Babenko et al. [14] apply Multiple Instance Learning (MIL) to object tracking. Stalder et al. [15] split the tasks of detection, recognition and tracking into three separate classifiers. Santner et al. [16] propose *PROST,* a cascade of a non-adaptive template model, an optical-flow-based tracker and an online random forest. Hare et al. [17] propose Struck, which generalizes from the binary classification problem to structured output prediction. Kalal et al. [18, 19] uses patches found on the trajectory of an optic-flow-based tracker in order to train an object detector; they named their technique Tracking-Learning-Detection (TLD). Updates are performed only if the discovered patch is similar to the initial patch. In contrast to adaptive tracking-by-detection methods, the output of the object detector is used only to reinitialize the optic-flow-based tracker in case of failure. This enables TLD to achieve superior results and higher frame rates. Recently, a variety of methods have been developed extending and improving the original TLD framework, for instance [20–22]. Although this class of methods has brought a new standard to tracking algorithms, they are still far from the performance achieved by humans during tracking problems. Therefore, there is still space for further improvements.

In this work, we take an alternative strategy for long-term object tracking and use a biologically inspired approach, based on Virtual Generalizing Random Access Memory (VG-RAM) Weightless Neural Networks (WNN) [23,24], for implementing an adaptive tracking-by-detection system. VG-RAM WNN (or VG-RAM for short) is a type of neural network that does not store weights in the synapses. Differently from standard neural networks, knowledge is kept within the neurons. It has been shown that this type of network has high performance for a variety of machine learning applications including face recognition [25,26], multi-label text categorization [27], stock return prediction [28], traffic sign detection and recognition [29–31], and tracking [32–34].

Our VG-RAM visual tracking system mimics the biological saccadic eye movement system. It models the transformations suffered by the images captured by the eyes in its way to the Superior Colliculus (SC) of mammalian brains, and models the response of SC neurons to previously seen patterns. Such biological model is incorporated into a Tracking-Learning-Detection algorithm to address all challenges presented by long-term tracking problems. We evaluate the performance of our system using the TLD database (available at http://personal.ee.surrey.ac.uk/Personal/Z.Kalal/tld.html), and show that it is able to achieve, in average, equivalent or superior performance than TLD.

This paper is organized as follows. After this introduction, we describe the biologically inspired VG-RAM WNN architecture for visual tracking (Section 2). In Section 3, we describe our experimental methodology and, in Section 4, we present and discuss our experimental results. Our conclusions and directions for future work follow in Section 5.

2. Visual tracking system based on VG-RAM WNN

In this section, we present our biologically inspired visual tracking system based on VG-RAM WNN. The system aims at long-term tracking of an unknown object in a video considering a single sample of the object. Basically, a bounding box around the object of interest is defined in the first frame, and the tracking system determines the object's location in subsequent frames. In addition, the system indicates the presence or not of the object in the frames. The location of the object is given by a bounding box surrounding it.

Our system is made of several components whose existence is motivated by findings of research in the area of biological vision systems [35,36]. In particular, we tried to replicate the tracking performance achieved by the human visual system. In this context, three important aspects of the human and other primates' visual system need to be highlighted: (i) the saccadic eye movement, (ii) the transformation suffered by the images captured by the eyes in the way from the retina to the Superior Colliculus, and (iii) the response of the neurons of the Superior Colliculus to patterns of interest in the visual scene. Our system tries and mimics these three aspects of biological visual systems using VG-RAM WNN neurons.

Our visual tracking neural system comprises four modules: (Re) Learning, Tracking, Detection and Validation (see Fig. 1). The (Re) Learning module is responsible for training the system with the annotated object in the first frame, and for retraining the system with a detected object in order to reinforce the current learned description of the object in the system. The Tracking module is responsible for following the object from frame to frame considering the last location of the object. The Detection module is responsible for finding the object once it returns to the scene, after occlusion for example. The Validation module is responsible for (i) deciding if it is necessary to retrain the system, (ii) if the process of tracking should continue, or (iii) if the detection process should start. Based on the decision of Validation module, the system reports either the coordinates of the object's bounding box or the information that the object is not visible in the scene.

In the following subsections, we briefly explain how we modeled the biological visual system that is responsible for the saccadic eye movements using VG-RAM WNN. Thereafter, the details of each module – (Re)Learning, Tracking, Detection, and Validation – are presented, as well as how they interact with our VG-RAM saccadic system.

2.1. Modeling the biological saccadic system with VG-RAM WNN

The saccadic eye movement is present in the visual system of primates (and most mammalians) and it is responsible for pointing