Background Modeling by Weightless Neural Networks

Massimo De Gregorio[1] and Maurizio Giordano[2(✉)]

[1] Istituto di Scienze Applicate e Sistemi Intelligenti – CNR,
Via Campi Flegrei 34, 80078 Pozzuoli, Naples, Italy
[2] Istituto di Calcolo e Reti ad Alte Prestazioni – CNR,
Via P. Castellino 111, 80131 Naples, Italy
{massimo.degregorio,maurizio.giordano}@cnr.it

Abstract. Background initialization is the task of computing a background model by processing a set of preliminary frames in a video scene. The initial background estimation serves as bootstrap model for video segmentation of foreground objects, although the background estimation could be refined and updated in steady state operation of video processing systems. In this paper we approach the background modeling problem with a weightless neural network called WiSARD^{rp}. The proposed approach is straightforward, since the computation is pixel–based and it exploits a dedicated neural network to model the pixel background by using the same training rule.

1 Introduction

Background modeling and estimation in video sequences is a challenging problem in computer vision since it is a required task in several research and commercial applications domains, such as video surveillance, segmentation, understanding, and compression, just to mention a few. Background estimation is the task of distinguishing foreground objects from background areas in video frames. Evaluation and comparison surveys of existing techniques can be found in literature [4,12,13].

Background modeling approaches can be classified into the following main categories: pixel–based [12], region–based [15] and object–based [8]. The first one mainly based on individually pixel changes. The second one based on the analysis of the single pixel with its neighborhood. The latter based on splitting the image into regions that are likely to belong to the same object. Another classification has been proposed by Bouwmans in his survey [4].

Self–organizing neural networks [10], general regression neural networks [5], self–organizing maps [14,16], and adaptive resonance theory neural networks [9] are examples of neural network based approaches to background modeling. Even if the background modeling problem has been approached with different neural architectures, the totality of them is based on a weighted neuron model. In this paper we approach the background modeling task with a weightless neural network (WNN) called WiSARD [1]. The approach is straightforward, since

V. Murino et al. (Eds.): ICIAP 2015 Workshops, LNCS 9281, pp. 493–501, 2015.
DOI: 10.1007/978-3-319-23222-5_60

the WNN system takes into account single pixel information to accomplish the background modeling, although, at the same time, it features highly adaptive and noise–tolerance behavior, due to a never–ending and single–policy learning phase of the adopted WNN model and its capability to absorb small variations of the model at runtime.

We apply the WNN–based background modeling method to address an important although specific topic of background estimation: how to compute an initial background model based on a set of preliminary frames of a video scene. The initial background model is a prerequisite for high quality moving object detection, either if foreground areas are computed at runtime as difference between the estimated background model and the current frame, or if it is used as bootstrapping model for successive updates at steady state system operation when foreground object detection is carried out.

With this target in mind, we did experiments and measured the performance of our method on the SBI dataset [11]. It is worth noticing that the viability and performance of WNNs in background detection has already been proved in [6], although by addressing a slightly but related topic, like change and moving object detection problem [7].

The paper is so organized: in Section 2 the adopted WNN model is introduced; in Section 3 the WNN–based method for background modeling is presented; Section 4 reports and discusses the experimental results of the method applied to videos of the SBI dataset; finally, Section 5 summarizes some concluding remarks.

2 The WiSARD^{rp} Weightless Neural Model

Weightless neural networks are based on networks of Random Access Memory (RAM) nodes [2]. The WNNs have a basis for their biological plausibility because of the straightforward analogy between the address decoding in RAMs and the integration of excitatory and inhibitory signaling performed by the neuron dendritic tree. WiSARD systems are a particular type of WNN. While the use of n–tuple RAM nodes in pattern recognition problems is old, dating about 60

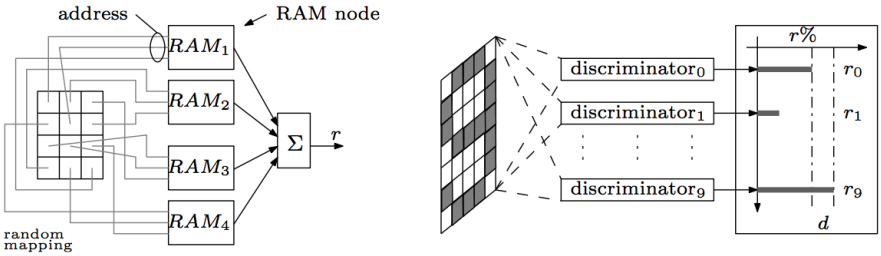

Fig. 1. A WiSARD discriminator (left) and a WiSARD multi–discriminator classifier (right)

years, with the availability of integrated circuit memories in the late 70s, the WiSARD (**W**ilkes, **S**tonham and **A**leksander **R**ecognition **D**evice) was the first artificial neural network machine to be patented and produced commercially [1].

The WiSARD model of computation is described in Figure 1. In this neural model a *discriminator* consists of a set of RAM–neurons, which store the information of occurrence of binary patterns of fixed size during the learning phase. Any sequence of bits fitting the s–sized pattern, the so–called *retina*, can be used to train a discriminator consisting of m RAMs with n–bit addressing, that is each RAM formed by 2^n memory cells, such that $s = m \times n$. In the example of Figure 1, the binary pattern (the retina) is a 3×4 image with pixels whose color can be 1 (for black) and 0 (for white); each RAM has a 3–bit addressing mechanism for cells, thus the number of required RAMs to cover the retina is $12/3 = 4$. Nevertheless, since any kind of information can be coded in binary patterns, by means of *ad hoc* data transformations a WiSARD discriminator can be enabled to learn then recognize data in both symbolic and numeric domains.

Since each RAM cell is uniquely addressed by an n–tuple of bits, the s–sized input pattern (the retina) can be partitioned into a set of n–tuples of bits, where bits forming each n–tuple have no correlation since they are pseudo–randomly extracted from the retina and associated to one RAM. For example, in Figure 1, the RAM_1, is uniquely associated to a triplet (3–tuple) of pixels. Thus, in the general case, the random mapping statically extract for the binary input pattern (stored in the retina) a number m of n–tuples of bits, and each n–tuple represents a binary address with n digits. This address is used to stimulate a RAM–neuron (by accessing one of its cells) either in writing mode (learning phase) or reading mode (classification phase).

A WiSARD *discriminator*, composed by m RAM–neurons, is trained with representative data of a specific class/category. In order to use the neuron network as a discriminator, one has to set to 0 the content of all cells of RAMs (initialization), and choose a training set formed by binary patterns of $m \times n$ bits. For each training pattern, a 1 is stored in the memory cells addressed by this input pattern. Once the training of patterns is completed, RAM contents will be set to a certain number of 0s and 1s. The information stored in RAM nodes during the training phase is used to deal with previously unseen patterns. When one of these is given as input, RAM memory contents addressed by the input pattern are read and summed by the summing device Σ which computes the number m^* of RAMs that output 1. Therefore $r = m^*/m$, which is called the *discriminator response*, provides the percentage of RAMs that outputs 1. It is easy to see that $r = 1$ if the input pattern belongs to the training set, $r = 0$ if none of its constituent n–tuples occurred in the training set. The closer is r to 1 the more "similar" is the input pattern to those patterns in the training set. The summing device enables this network of RAM nodes to exhibit – just like other ANN models based on synaptic weights – generalization and noise tolerance [3].

In this work we adopted a modified version of the WiSARD, that we call WiSARD^{rp}, which is depicted in Figure 2(a) and whose main modifications are the followings:

1. **Training phase** – RAM contents corresponding to n–tuples of binary inputs on the retina instead of being set to 1 are incremented by a positive number ρ (*reward*) at each access, up to a maximum value called *uppermark*, namely β. Thus, during training, RAM contents can store sub patterns frequencies up to a given saturation value β. At each memory cell access, while its content is increased by ρ, all other cells are decreased by ψ (*punishment*), thus lowering the frequencies of not occurring sub patterns. Punishments are applied to cells although they can never change their contents to negative values.
2. **Classification phase** – While in WiSARD, upon a read stimulus, a RAM always outputs its contents, that can be 0 or 1, in WiSARD^{rp} the output is 1 if the accessed RAM cell value is greater than a threshold, namely ω (*bleaching*), otherwise it is 0. With this modification we put a firing condition to RAMs, thus making them to contribute to the classification response only when the stored frequency of the sub patter under test overcomes a certain threshold ω.

It is easy to prove that WiSARD^{rp} with $\rho = 1$, $\psi = 0$, $\beta = +\infty$, and $\omega = 0$, behaves exactly as the original WiSARD system in the training and classification phases. On the other side, different settings of ρ, ψ, β and ω parameters implies very different behaviors of the training and classification phases. In other words, the *reward & punishment* strategy (*rp*) allows to size the time a sub pattern is completely unlearned once it does not occur any longer, as well as to size the time the more frequent sub patterns remain in the learned knowledge of the network. By changing these two latency times it is possible to tune WiSARD^{rp} capabilities both in classification and class model construction over a changeable and long–term training phase.

3 The BGWiS Approach to Background Modeling

The proposed method for background modeling in video sequences is called BGWiS and it exploits the WiSARD^{rp} neural network model in its core logic. The pseudocode of BGWiS is sketched in Algorithm 1 and it is based on the following assumptions:

1. **Color encoding** – The pixel color (in any color space among RGB, HSV, and Lab) is represented by three non–negative numbers, namely *color channels*, in the range 0–255. In order to use the WiSARD^{rp} as the core of our background modeling system a binarisation of pixel colors is required. As illustrated in Figure 2(a), the three channels are scaled and discretized in the range $0, 1, \ldots, nt - 1$, thus representing a color with a binary pattern of size $3 \times nt$. This black and white image is fed as input to the WiSARD^{rp} system for training and classification.

Algorithm 1. BGWiS method pseudocode

```
1 foreach frame in video sequence do
2     transform frame in the chosen color space;
3     foreach pixel in frame do
4         get response on input color from pixel discriminator (with firing threshold ω) ;
5         classify pixel as bg (if response > σ), otherwise as fg;
6         use color to train pixel discriminator (reward & punishment rule {ρ, ψ, β}) ;
7         use pixel discriminator contents to update background model;
```

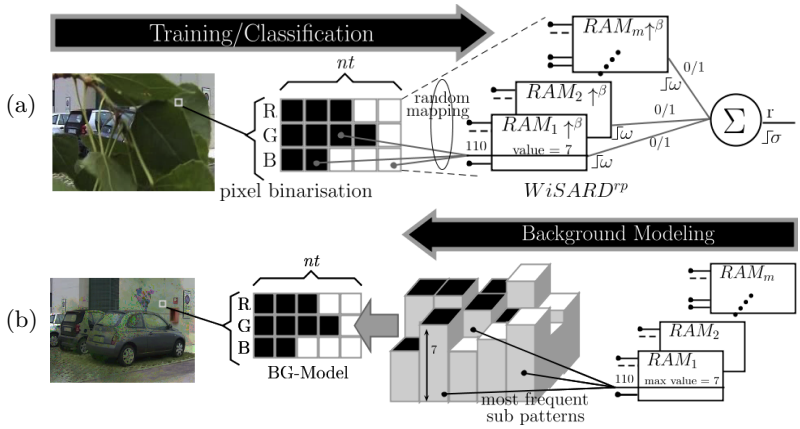

Fig. 2. Training phase (a) and background modeling phase (b) of BGWiS

2. **Background model** – In WiSARD^{rp} a pattern class is represented by the snapshot of RAM contents during time. As already mentioned, each RAM records the number of occurrences of sub patterns inside the binary input. In other words, a RAM content can be seen as the histogram of occurrences of all binary sub patterns occurred during training. In our case study, we assume that the background model of a pixel is formed by combining the most frequent sub patterns contained in RAMs. As illustrated in Figure 2(b), the background model of a pixel is computed by considering in each RAM the sub pattern addressing the cell with the greatest value.
3. **Processing loop** – In video processing learning and classification are overlapped and execute continuously. In the algorithm 1 the role of WiSARD^{rp} is twofold:
   (a) **Foreground detection** – On the basis of the learned knowledge of pixel color history which is stored in RAM cells and managed by means of the reward & punishment mechanism, the system is able to provide a classification response for the current pixel, that is a measure of its similarity to the color knowledge acquired by the WiSARD^{rp} by means of a continuous training. This response is then compared to the threshold σ to state whether the pixel in the current frame is detected as belonging either to the background or to the foreground. Afterwards, the WiSARD^{rp} is

trained on the current pixel color regardless of the classification result (see line 4–6 of Algorithm 1).

(b) **Background modeling** – The learned knowledge of pixel color history contained in the discriminator neurons is used (see line 7 of Algorithm 1) to update the pixel background model, according to the logic already described (see point 2.). BGWiS system outputs a colored image representing the computed background model during the timeline.

The change–detection method proposed in the previous work [6] is based on WiSARD while the current method exploits WiSARD^{rp}. In particular, the change–detection method has no punishment ($\psi = 0$) and a fixed reward ($\rho = 1$) with both no firing threshold ($\omega = 0$) and saturation constraint ($\beta = \infty$). Another difference is the logic of the background modeling algorithm. The CDnet dataset target was foreground object detection rather than background modeling which is the target of SBI dataset. In fact, in the former challenge a set of video frames was allowed to be used only for training, while the remaining video frames were used for classification (foreground detection). In the change–detection method only pixel colors detected as background are used to further update (enrich) the background model for those pixels (i.e. in Algorithm 1 lines 6–7 are executed under the condition of line 5), unless a history buffer storing more recent foreground pixels is full and it is time to use it to reinitialize the background model. This policy, which was adopted to deal with moving/stopping objects in the scene, jointly with the aforementioned (and static) setting for ρ, ψ, β and ω, was experimentally proved to be a high performance (in the average) method in the CDnet competition. SBI videos are very different with respect to the percentage of background occlusion occurring both in space (frame area) and time (video duration). For this reason, BGWiS was designed to be flexible and reconfigurable in the parameters driving background modeling, as well as no comparison between the change–detection method of [6] and BGWiS could be carried out within the scope of SBI experiments.

4 BGWiS Experiment Settings and Results

We did experiments of BGWiS running on the SBI dataset of videos. Table 1 reports the system performance by evaluating eight metrics which, as defined on the SBI dataset website, compare for each sequence the ground truth background

Table 1. BGWiS results on SBI dataset

Sequence	*AGE*	*EPs*	*pEPs*	*CEPs*	*pCEPS*	*MSSSIM*	*PSNR*	*CQM*	ρ	ψ	β	n
HallAndMonitor	2.5177	442	0.0052	161	0.0019	0.9773	31.7928	42.2277	1	1	65	4
HighwayI	1.6885	106	0.0014	8	0.0001	0.9916	39.3795	59.4393	2	1	∞	16
HighwayII	2.2060	357	0.0046	1	0.0000	0.9948	33.2780	39.8760	2	1	∞	16
CaVignal	9.1964	25	0.0009	0	0.0000	0.9933	27.5468	39.7962	1	1	95	8
Foliage	14.7191	3060	0.1062	145	0.0050	0.9465	22.5656	33.7781	2	1	20	16
PeopleAndFoliage	32.2664	25240	0.3286	20514	0.2671	0.6849	13.9668	23.9475	1	3	∞	16
Snellen	38.4451	11324	0.5461	9928	0.4788	0.7700	14.5757	38.5964	1	3	∞	16

Fig. 3. BGWiS output on SBI dataset: GT = Ground Truth; CB = Computed Background; |GT-CB| = absolute difference of GT and CB, EPs = number of CB pixels different from GT over a given threshold; CEPs = number of EPs in CB with all 4–connected neighbors pixels in EPs

image (GT) with the computed background image (CB) produced by our system at the end of the video.

In all measures we chose to transform input frames into Lab color space as a preprocessing step. As reported in Table 1, we used different parameter settings for videos, although in most cases we used 16–bit addressing for RAMs. The higher is the bit address the more precise is the color selection for the background model in bustling video scenes. In *HighwayI*, *HighwayII* and *Foliage*

videos the pixel color patterns are used to access–then–reward neuron cells twice ($\rho = 2$, $\psi = 1$). Indeed, this setting allows the system to forget patterns more slowly and it is useful in those videos with object moving (or changing in shape) in a regular manner. In other cases, like *PeopleAndFoliage* and *Snellen*, the pixel color patterns are used to access–then–reward neuron cells a third than the punishment required for unseen color patterns ($\rho = 1$, $\psi = 3$).

It should be noted that in some video the saturation level of RAMs has been fixed in order to limit that foreground patterns could outnumber (in frequency) the background patterns by shortening the foreground patterns decay time. In the remaining videos the saturation level was disabled ($\beta = \infty$). It should be noted that the neuron threshold σ and bleaching ω only apply to classification phase without interfering with the background modeling computation. That is why their settings are not in Table 1. In order to generated the best CB, nt was set to 256 in all experiments (no loss of information).

Snapshots of BGWiS outputs are shown in Figure 3. By relating the performance measures of Table 1 to the snapshots of Figure 3, we can notice how the system behaves well in the first four videos. The remaining videos (*Foliage*, *PeopleAndFoliage* and *Snellen*) are challenging: in both *Foliage* and *Snellen* a waving plant frequently (more than 50% of timeline) occludes almost completely the target background; in the *PeopleAndFoliage* case a waving plant plus stationary–then–moving persons occlude more than half of the target background for almost the whole timeline. Although these three cases of study are hard to get a clean and effective background model, it is worth noticing that our system at least shows quite good results in the *Foliage* video (both in terms of the eight metrics and by evaluating at sight the difference from the ground truth). In the *Snellen* video, the computed background snapshot is similar to the ground truth, although the AGE, EPs and CEPs metrics are poor.

5 Conclusions

In this work we presented a background modeling approach for videos based on a weightless neural system, namely WiSARD^{rp}, with the aim of exploiting its features of being highly adaptive and noise–tolerance at runtime. Indeed, the adopted neural model is able to operate in a never–ending and single–policy learning phase with a *reward & punishment* mechanism, which allows it to absorb small variations of the learned model in the steady state of operation. The approach is quite simple, and by tuning a set of parameters that rule the reward & punishment mechanism of neural training, we have proved how it is possible to build on it a background modeling system showing very good performance in common case studies (like camera views of highways and traffic crossings), as well as good or promising results in more challenging video sequences (like static views with heavy and waving occlusions in space and time).

References

1. Aleksander, I., Thomas, W.V., Bowden, P.A.: WiSARD a radical step forward in image recognition. Sensor Review **4**, 120–124 (1984)
2. Aleksander, I., De Gregorio, M., França, F.M.G., Lima, P.M.V., Morton, H.: A brief introduction to weightless neural systems. In: ESANN 2009, pp. 299–305 (2009)
3. Aleksander, I., Morton, H.: An introduction to neural computing. Chapman & Hall (1990)
4. Bouwmans, T.: Recent advanced statistical background modeling for foreground detection: A systematic survey. Recent Patents on Computer Science **4**(3), 147–176 (2011)
5. Culibrk, D., et al.: A neural network approach to bayesian background modeling for video object segmentation. In: Proc. of VISAPP 2006, pp. 474–479 (2006)
6. De Gregorio, M., Giordano, M.: Change detection with weightless neural networks. In: Proc. of 2014 IEEE CVPRW, pp. 409–413 (2014)
7. Goyette, N., et al.: Changedetection.net: a new change detection benchmark dataset. In: Proc. of 2014 IEEE CVPRW, pp. 1–8 (2012)
8. Hall, D., et al.: Comparison of target detection algorithms using adaptive background models. In: Proc. 2nd Joint IEEE Int. Workshop VS-PETS, 2005, pp. 113–120 (2005)
9. Luque, R.M., Domínguez, E., Palomo, E.J., Muñoz, J.: An art-type network approach for video object detection. In: ESANN 2010, pp. 423–428 (2010)
10. Maddalena, L., Petrosino, A.: A self-organizing approach to background subtraction for visual surveillance applications. IEEE Trans. on Image Processing **17**(7), 1168–1177 (2008)
11. Maddalena, L., Petrosino, A.: Towards benchmarking scene background initialization (2015). http://arxiv.org/abs/1506.04051 (posted June 12, 2015)
12. Panahi, S., Sheikhi, S., Hadadan, S., Gheissari, N.: Evaluation of background subtraction methods. In: Proc. of DICTA 2008, pp. 357–364 (2008)
13. Piccardi, M.: Background subtraction techniques: a review. In: IEEE International Conference on Systems, Man and Cybernetics (October 2004)
14. Ramirez-Quintana, J., Chacon-Murguia, M.: Self-organizing retinotopic maps applied to background modeling for dynamic object segmentation in video sequences. In: Proc. of IJCNN 2013, pp. 1–8, August 2013
15. Shuai, Y.M., Xu, X., Sun, H., Xu, G.: Change detection based on region likelihood ratio in multitemporal sar images. In: 2006 8th Int. Conf. on Signal Processing, vol. 2 (2006)
16. Zhao, Z., Zhang, X., Fang, Y.: Stacked multilayer self-organizing map for background modeling. IEEE Transactions on Image Processing **24**(9), 2841–2850 (2015)