Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

Temporally Adaptive Restricted Boltzmann Machine for Background Modeling

Linli Xu, Yitan Li, Yubo Wang and **Enhong Chen**
School of Computer Science and Technology
University of Science and Technology of China
linlixu@ustc.edu.cn, {etali, wybang}@mail.ustc.edu.cn, cheneh@ustc.edu.cn

Abstract

We examine the fundamental problem of background modeling which is to model the background scenes in video sequences and segment the moving objects from the background. A novel approach is proposed based on the Restricted Boltzmann Machine (RBM) while exploiting the temporal nature of the problem. In particular, we augment the standard RBM to take a window of sequential video frames as input and generate the background model while enforcing the background smoothly adapting to the temporal changes. As a result, the augmented temporally adaptive model can generate stable background given noisy inputs and adapt quickly to the changes in background while keeping all the advantages of RBMs including exact inference and effective learning procedure. Experimental results demonstrate the effectiveness of the proposed method in modeling the temporal nature in background.

1 Introduction

Background modeling is one of the fundamental problems in automatic video content analysis. The major task involves modeling the relatively stationary background scene based on sequential video frames. This can be followed by subtracting each frame from the background which entails background subtraction and foreground detection. Background modeling is an important preprocessing step for many vision tasks including background subtraction, foreground detection, object tracking, activity recognition, and especially video surveillance.

Various methods have been proposed to tackle the background modeling or subtraction problems over the past decade (Piccardi 2004). Most of these techniques follow a general pixel-level recipe that assumes pixels are statistically independent and build models to estimate the background distributions of pixels.

There are roughly two categories of background modeling techniques: parametric and non-parametric. Among the parametric approaches, a simple yet effective example is the mixture of Gaussians (MoG) model (Stauffer and Grimson 1999) which models the pixel distributions in a multimodal background and dynamic environment by describing each observable background or foreground object with a Gaussian distribution. Based on the MoG scheme, various attempts have been made to address different issues and augment the performance (KaewTraKulPong and Bowden 2002; Zivkovic 2004; Chen et al. 2012). Recently, a parametric approach (He, Balzano, and Szlam 2012) based on low-rank and subspace learning is also proposed. On the other hand, non-parametric techniques build background models from observed pixel values. Among those, various kernel density estimation approaches are applied to model the background distribution for each pixel (Elgammal, Harwood, and Davis 2000). In (Kim et al. 2005; Pal, Schaefer, and Celebi 2010), sample background pixel values are summarized into codebooks which correspond to a compressed background model for a video sequence. Another non-parametric example is the VIBE model which uses random policy to select samples from the local neighborhood of a pixel and builds the background model for that pixel accordingly (Barnich and Van Droogenbroeck 2011).

Most of the traditional approaches discussed above, including MoG, Codebook, VIBE, etc., share the same problem as they ignore the temporal nature of background modeling in video sequences, which may cause problems in situations with dynamic backgrounds and illumination changes. In this paper we take a temporal perspective and address the background modeling problem based on the following intuition: the background of a sequence of video frames should be temporally smooth and stable to noisy fluctuations; more importantly, if the background changes, the model should be able to adapt quickly to that. In the meantime, instead of modeling each pixel independently, we consider the spatial nature of the problem and exploit possible hidden correlations among pixels. To achieve that, we resort to Restricted Boltzmann Machines (RBMs).

An RBM is a generative two-layer neural network that can learn a probability distribution from data. Recent advances in various problem domains demonstrate the superior performance of RBMs in unsupervised tasks including feature learning (Coates, Ng, and Lee 2011) and dimensionality reduction (Hinton and Salakhutdinov 2006), as well as supervised tasks including classification (Larochelle and Bengio 2008) and collaborative filtering (Salakhutdinov, Mnih, and Hinton 2007). Moreover, an RBM enjoys efficient training and exact inference due to the "restriction" that the network

is in a form of bipartite graph and there are no connections within a layer. In terms of the background modeling problem, taking pixel values as visible units, an RBM is a natural way to model the hidden background distribution of the pixels. In this paper, to exploit the temporal nature of the problem, we augment the standard RBM to take a window of adjacent frames and enforce the background smoothly adapting to temporal changes. To the best of our knowledge, it is the first time to apply RBMs to the task of background modeling.

The remainder of this paper is organized as follows. After reviewing necessary background in Section 2, we present the augmented temporally adaptive RBM to tackle the problem of background modeling in video sequences in Section 3. We show that after some reformulation the training procedure of the proposed model has the advantageous properties of exact inference and efficient training as standard RBMs. We present experimental results on background modeling tasks in Section 4 and then conclude.

2 Restricted Boltzmann Machines

A Restricted Boltzmann Machine (RBM) is an energy-based model which constitutes of two layers (Hinton 2002) including a layer of visible units $\mathbf{v}$ and a layer of hidden units $\mathbf{h}$. An RBM is restricted in the sense that there are only connections between layers, and none within a layer, which results in a structure of bipartite graph as shown in Figure 1(a). Typically in an RBM the visible and hidden units take binary values, and the energy given the configuration $(\mathbf{v}, \mathbf{h})$ is defined as:

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{v}^\top \mathbf{W}\mathbf{h} - \mathbf{c}^\top \mathbf{v} - \mathbf{b}^\top \mathbf{h}, \tag{1}$$

where $\mathbf{W} = \{W_{ij}\}$ is the connection weight matrix and W_{ij} is associated with the hidden unit h_i and the visible unit v_j, $\mathbf{b}$ and $\mathbf{c}$ are bias vectors for the hidden layer and visible layer respectively. The probability distribution over pairs of $(\mathbf{v}, \mathbf{h})$ is then defined given the energy function as

$$P(\mathbf{v}, \mathbf{h}) = \exp\{-E(\mathbf{v}, \mathbf{h})\}/Z, \tag{2}$$

where $Z = \sum_{\tilde{\mathbf{v}}, \tilde{\mathbf{h}}} \exp\{-E(\tilde{\mathbf{v}}, \tilde{\mathbf{h}})\}$ is the normalization factor.

However, for the background modeling problem, the input data is composed of continuous pixel values, therefore we employ a slightly modified RBM in which the visible units are linear, continuous variables with Gaussian noise (Welling, Rosen-Zvi, and Hinton 2004), and correspondingly the joint probability distribution is

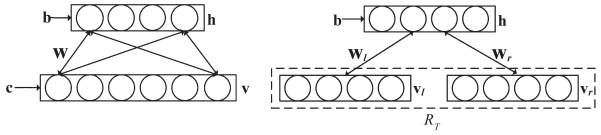
$$P(\mathbf{v}, \mathbf{h}) = \exp\Big\{\sum_{ij} \frac{v_i}{\sigma_i} W_{ij} h_j + \sum_j b_j h_j - \sum_i \frac{(v_i - c_i)^2}{2\sigma_i^2}\Big\}/Z, \tag{3}$$

where σ_i denotes the standard deviation of the visible unit v_i.

In practice, if we rescale the data to unit variance and fix σ_i at 1, the conditional probability distributions of individual units can be computed easily due to the fact that the hidden units are conditionally independent given the visible units and vice versa based on the bipartite graph:

$$\begin{aligned} P(v_i|\mathbf{h}) &= \mathcal{N}(c_i + \sum_j W_{ij} h_j, 1) \\ P(h_j = 1|\mathbf{v}) &= s(b_j + \sum_i W_{ij} v_i), \end{aligned} \tag{4}$$

where s is the logistic function, $\mathcal{N}$ is a multivariate Gaussian, v_i is the ith component of the vector $\mathbf{v}$ and h_j is the jth component of the vector $\mathbf{h}$ (we will use i to index visible units and j to index hidden units by default in the rest of the paper). The factorized conditional distributions shown in (4) ensure an exact inference procedure for the hidden variables.



(a) RBM (b) TARBM

Figure 1: Architectures of RBM and TARBM. Note that although the two architectures look similar, the TARBM has an extra term R_T in the corresponding objective function.

An RBM can be trained to maximize the log likelihood $\log P(\mathbf{v})$ using gradient ascent. However, the joint distribution $P(\mathbf{v}, \mathbf{h})$ is computationally expensive to calculate. To address that, an approximate routine called Contrastive Divergence (CD_k) (Hinton 2002) is adopted during the learning procedure, and we use CD_1 in this paper. The update rules for training include:

$$\Delta W_{ij} \propto \langle v_i h_j \rangle_{\text{data}} - \langle v_i h_j \rangle_{\text{model}} \tag{5}$$

$$\Delta b_j \propto \langle h_j \rangle_{\text{data}} - \langle h_j \rangle_{\text{model}} \tag{6}$$

$$\Delta c_i \propto \langle v_i \rangle_{\text{data}} - \langle v_i \rangle_{\text{model}}, \tag{7}$$

where $\langle \cdot \rangle_{\text{data}}$ denotes the expectation under the distribution $P(\mathbf{h}|\mathbf{v})$, and $\langle \cdot \rangle_{\text{model}}$ is the expectation with respect to the distribution $P(\mathbf{v}, \mathbf{h})$. The first expectation is easy to compute, while one can obtain the reconstruction of the data and compute the second expectation by alternating Gibbs sampling. More technical details can be found in (Hinton 2002; 2010).

3 Temporally Adaptive RBMs for Background Modeling

Background Modeling with RBMs

As discussed above, RBMs are capable of capturing the probability distribution of data with the advantages of fast exact inference and effective approximate learning and therefore naturally suitable to model the background in video sequences. One can intuitively treat the pixels of video frames as the visible units and train an RBM to model the relatively stationary background scene in the video frames. In the trained model, the weight $\mathbf{W}$ works as a sifter to filter the foreground in frames, leaving the background expressed

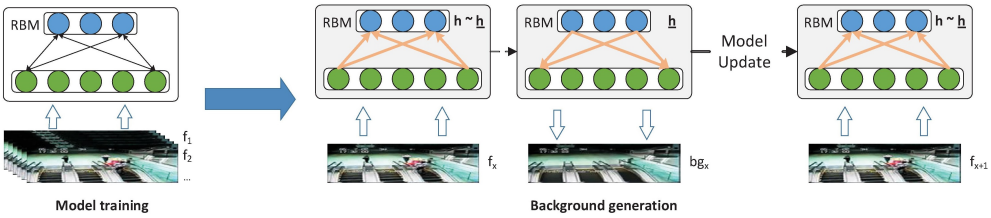


Figure 2: Framework of background modeling. The model is initially trained on the first several adjacent video frames, after that the model generates the background for the subsequent frames and is fine tuned to adapt to new background changes. $\underline{\mathbf{h}}$ denotes the unbiased sample of the hidden variable $\mathbf{h}$.

with the hidden layer. For a new frame, the corresponding background can be generated by sampling from the hidden representation of the filtered input frame followed by reconstructing with $\mathbf{W}$. The framework of the model training and background modeling procedures is shown in Figure 2.

Unfortunately, standard RBMs are not designed to model sequential data. In general, RBMs are trained in batch mode, which means the order of the instances in the batch makes no difference in the training procedure. To cope with the temporal nature of the background modeling problem, we propose the Temporally Adaptive Restricted Boltzmann Machine (TARBM) which takes the temporal information contained in the sequential frames into consideration.

Temporally Adaptive RBMs

To model the temporal information, there exist methods that modify the standard model by adding directed connections from the previous timesteps to the current ones (Sutskever and Hinton 2007; Sutskever, Hinton, and Taylor 2009; Taylor, Hinton, and Roweis 2006), which may result in more parameters to train and imply linear correlation between timesteps. In this paper, we modify the model and force it to adapt to the temporal changes in an implicit way. Specifically, Figure 1(b) shows the architecture of a TARBM: the visible layer consists of a pair of components, each with the same number of units, corresponding to a window of two adjacent frames; one single hidden layer generates the sequential components, where $\mathbf{b}$ is the corresponding bias vector and $\mathbf{W}_l, \mathbf{W}_r$ are the connection weight matrices respectively; a regularization term R_T provides temporal constraints for reconstructed data which will be discussed later. We can also generalize the model to incorporate higher order temporal correlations by increasing the window size to k sequential frames.

The temporally adaptive RBM defines a joint distribution as follows:

$$\begin{aligned} &P(\mathbf{v}_l, \mathbf{v}_r, \mathbf{h}) \\ =&\frac{1}{Z}\exp\Big\{\mathbf{b}^\top\mathbf{h} + \sum_{i,j}\big(\frac{v_{li}W_{lij}h_j}{\sigma_{li}} + \frac{v_{ri}W_{rij}h_j}{\sigma_{ri}}\big) \\ &- \sum_i \big\{\frac{(v_{li}-c_{li})^2}{2{\sigma_{li}}^2} + \frac{(v_{ri}-c_{ri})^2}{2{\sigma_{ri}}^2}\big\}\Big\}, \end{aligned} \tag{8}$$

whereas the conditional distributions $P(\mathbf{h}|\mathbf{v}_l, \mathbf{v}_r)$, $P(\mathbf{v}_l|\mathbf{h})$ and $P(\mathbf{v}_r|\mathbf{h})$ can be similarly derived from the above joint distribution:

$$P(h_j = 1|\mathbf{v}_l, \mathbf{v}_r) = s(b_j + \sum_i \frac{v_{li}W_{lij}}{\sigma_{li}} + \frac{v_{ri}W_{rij}}{\sigma_{ri}}) \tag{9}$$

$$P(v_{li}|\mathbf{h}) = \mathcal{N}(c_{li} + \sum_j W_{lij}h_j, {\sigma_{li}}^2) \tag{10}$$

$$P(v_{ri}|\mathbf{h}) = \mathcal{N}(c_{ri} + \sum_j W_{rij}h_j, {\sigma_{ri}}^2). \tag{11}$$

Notice that TARBM is an extension of general RBMs, and inherits all the advantages of RBMs including exact inference and effective learning. It not only minimizes the negative log likelihood during training, but also adopts a temporal regularization term to enforce smoothness in sequential data. In a sequential video sequence, the background varies with time; meanwhile, the variation in the background between adjacent frames tends to be smooth in general. Therefore we can introduce a temporal cost which measures the divergence of the reconstructed adjacent frames:

$$R_T(\mathbf{v}_l, \mathbf{v}_r) = \big\|\mathbf{E}_{\tilde{\mathbf{h}}|\mathbf{v}_l,\mathbf{v}_r}\big[\mathbf{E}_{\tilde{\mathbf{v}}_l,\tilde{\mathbf{v}}_r|\tilde{h}}[\tilde{\mathbf{v}_l} - \tilde{\mathbf{v}_r}]\big]\big\|_2 \tag{12}$$

where $\mathbf{E}_{\tilde{\mathbf{h}}|\mathbf{v}_l,\mathbf{v}_r}[*]$ and $\mathbf{E}_{\mathbf{v}_l,\mathbf{v}_r|\tilde{\mathbf{h}}}[*]$ denote the expectations under distributions of $P(\tilde{\mathbf{h}}|\mathbf{v}_l, \mathbf{v}_r)$ and $P(\mathbf{v}_l, \mathbf{v}_r|\tilde{\mathbf{h}})$ respectively, and thus $\mathbf{E}_{\tilde{\mathbf{h}}|\mathbf{v}_l,\mathbf{v}_r}\big[\mathbf{E}_{\tilde{\mathbf{v}}_l,\tilde{\mathbf{v}}_r|\tilde{h}}[*]\big]$ implies an expectation reconstruction procedure given observed frames $\mathbf{v}_l, \mathbf{v}_r$. The temporal regularization R_T forces the reconstructed adjacent frames $\tilde{\mathbf{v}}_l$ and $\tilde{\mathbf{v}}_r$ to be close with l_2 norm during the reconstruction procedure.

The final optimization problem can be formulated as minimizing the negative log likelihood with the temporal regularization R_T over adjacent frames in a video sequence:

$$\min_{\mathbf{W}_l,\mathbf{W}_r,\mathbf{b},\mathbf{c}_l,\mathbf{c}_r} \sum_{\mathbf{v}_l,\mathbf{v}_r} -\log P(\mathbf{v}_l, \mathbf{v}_r) + \lambda R_T(\mathbf{v}_l, \mathbf{v}_r), \tag{13}$$

where $\lambda > 0$ is a parameter balancing the two parts in the objective which correspond to fitting the data and enforcing the temporal smoothness respectively.

The effect of the temporal regularization is two-fold: when the background in a video sequence is relatively stationary with moving objects in the foreground, the model forces the reconstruction to be stable and robust to noisy fluctuations; on the other hand, if the background changes in a video sequence, the model will adapt to it quickly.

To solve the optimization problem (13) one needs to evaluate the regularization term. The temporal regularizer comes

with explicit interpretations, and we can approximate it with a tractable form. More specifically, let $\mathbf{h}_s$ denote the hidden state sampled from $P(\mathbf{h}|\mathbf{v}_l, \mathbf{v}_r)$, the regularization term can be derived as follows:

$$\begin{aligned}
&\left\|\mathbf{E}_{\tilde{\mathbf{h}}|\mathbf{v}_l,\mathbf{v}_r}\left[\mathbf{E}_{\tilde{\mathbf{v}}_l,\tilde{\mathbf{v}}_r|\tilde{h}}[\tilde{\mathbf{v}}_l - \tilde{\mathbf{v}}_r]\right]\right\|_2 \\
=&\left\|\mathbf{E}_{\tilde{\mathbf{h}}|\mathbf{v}_l,\mathbf{v}_r}[\mathbf{W}_l\tilde{\mathbf{h}} + \mathbf{c}_l - (\mathbf{W}_r\tilde{\mathbf{h}} + \mathbf{c}_r)]\right\|_2 \qquad (14)\\
=&\left\|(\mathbf{W}_l - \mathbf{W}_r)\mathbf{E}_{\tilde{\mathbf{h}}|\mathbf{v}_l,\mathbf{v}_r}[\tilde{\mathbf{h}}] + (\mathbf{c}_l - \mathbf{c}_r)\right\|_2 \qquad (15)\\
\approx&\left\|(\mathbf{W}_l - \mathbf{W}_r)\mathbf{h}_s + (\mathbf{c}_l - \mathbf{c}_r)\right\|_2 = \tilde{R}_T. \qquad (16)
\end{aligned}$$

We get (14) with Eq. (10) and Eq. (11). By using $\mathbf{h}_s$ to approximate $\mathbf{E}_{\tilde{\mathbf{h}}|\mathbf{v}_l,\mathbf{v}_r}[\tilde{\mathbf{h}}]$, we derive (16) from (15), where $\tilde{R}_T$ denotes the approximate temporal regularizer. The gradients of $\tilde{R}_T$ can be derived as follows:

$$\begin{aligned}
\frac{\partial \tilde{R}_T}{\partial W_{lij}} = -\frac{\partial \tilde{R}_T}{\partial W_{rij}} =& \frac{h_j}{\tilde{R}_T}\big((\mathbf{W}_l\mathbf{h}_s + \mathbf{c}_l) - (\mathbf{W}_r\mathbf{h}_s + \mathbf{c}_r)\big)_i \\
=& \frac{h_j}{\tilde{R}_T}\langle \tilde{v}_{li} - \tilde{v}_{ri}\rangle_{\text{rec}} \qquad (17)\\
\frac{\partial \tilde{R}_T}{\partial c_{li}} = -\frac{\partial \tilde{R}_T}{\partial c_{ri}} =& \frac{1}{\tilde{R}_T}\langle \tilde{v}_{li} - \tilde{v}_{ri}\rangle_{\text{rec}}, \qquad (18)
\end{aligned}$$

where $\langle . \rangle_{\text{rec}}$ denotes the expectation reconstruction procedure $\mathbf{E}_{\tilde{\mathbf{h}}|\mathbf{v}_l,\mathbf{v}_r}\left[\mathbf{E}_{\tilde{\mathbf{v}}_l,\tilde{\mathbf{v}}_r|\tilde{h}}\langle . \rangle\right]$ which can be embedded in the CD_k routine.

Combined with the CD_k algorithm for training RBM in the literature, the optimization problem (13) can be approximately solved by the following update rules:

$$\begin{aligned}
\Delta W_{lij} &\propto \langle \frac{v_{li}h_j}{\sigma_{li}}\rangle_{\text{data}} - \langle \frac{v_{li}h_j}{\sigma_{li}}\rangle_{\text{model}} - \lambda\frac{h_j}{\tilde{R}_T}\langle \tilde{v}_{li} - \tilde{v}_{ri}\rangle_{\text{rec}}\\
\Delta W_{rij} &\propto \langle \frac{v_{ri}h_j}{\sigma_{ri}}\rangle_{\text{data}} - \langle \frac{v_{ri}h_j}{\sigma_{ri}}\rangle_{\text{model}} + \lambda\frac{h_j}{\tilde{R}_T}\langle \tilde{v}_{li} - \tilde{v}_{ri}\rangle_{\text{rec}}\\
\Delta c_{li} &\propto \langle \frac{v_{li} - c_{li}}{\sigma_{li}^2}\rangle_{\text{data}} - \langle \frac{v_{li} - c_{li}}{\sigma_{li}^2}\rangle_{\text{model}} - \frac{\lambda}{\tilde{R}_T}\langle \tilde{v}_{li} - \tilde{v}_{ri}\rangle_{\text{rec}}\\
\Delta c_{ri} &\propto \langle \frac{v_{ri} - c_{ri}}{{\sigma_{ri}}^2}\rangle_{\text{data}} - \langle \frac{v_{ri} - c_{ri}}{{\sigma_{ri}}^2}\rangle_{\text{model}} + \frac{\lambda}{\tilde{R}_T}\langle \tilde{v}_{li} - \tilde{v}_{ri}\rangle_{\text{rec}}\\
\Delta b_j &\propto \langle h_j\rangle_{\text{data}} - \langle h_j\rangle_{\text{model}}.
\end{aligned} \qquad (19)$$

In practice, we rescale the data to unit variance and fix σ_{li} and σ_{ri} at 1 for convenience. For simplification, CD_1 is used when training the TARBM. The variables v_{li} and v_{ri} in $\langle * \rangle_{\text{data}}$ are given by input ${\mathbf{v}_l}^0$ and ${\mathbf{v}_r}^0$, and h_j is determined by sampling from the distribution $P(h_j|{\mathbf{v}_l}^0, {\mathbf{v}_r}^0)$, denoted by $\mathbf{h}^0$. On the other hand, v_{li} and v_{ri} in $\langle * \rangle_{\text{model}}$ are sampled from $P(v_{li}|\mathbf{h}^0)$ and $P(v_{ri}|\mathbf{h}^0)$ respectively, denoted by ${\mathbf{v}_l}^1$ and ${\mathbf{v}_r}^1$, then h_j is drawn from $P(h_j|{\mathbf{v}_l}^1, {\mathbf{v}_r}^1)$, and $\tilde{v}_{li}$ and $\tilde{v}_{ri}$ in $\langle * \rangle_{\text{rec}}$ are sampled from $P(v_{li}|\mathbf{h}^0)$ and $P(v_{ri}|\mathbf{h}^0)$ as well.

Background Modeling

To model the background in video sequences, the TARBM takes two adjacent frames (f_{t-1}, f_t) as input. During training, the mini-batch scheme is adopted. We split the sequence of frames $(f_0, f_1, f_2, ..., f_{m\cdot d})$ into m pairs of chunks with size d: $\{(f_{(i-1)\cdot d}, f_{(i-1)\cdot d+1}, ..., f_{i\cdot d-1}), (f_{(i-1)\cdot d+1}, f_{(i-1)\cdot d+2}, ..., f_{i\cdot d})\}_{i=1}^{m}$, and in each chunk, adjacent frames are paired together sequentially to form a mini-batch as the training data. Once the training procedure of the model with update rules in (19) is complete, the hidden layer can be regarded as a representation of the background in some latent space. Moreover, in order to get the background, pairs of frames $\{f_{t-1}, f_t\}$ are treated as input by starting from the first test frame which is also the next of the last training frame f_{endtr} , and the corresponding background BG_t is generated with the following steps:

1. Initialize (f_{t-1}, f_t) with $(f_{endtr}, f_{endtr+1})$;
2. Sample $\tilde{\mathbf{h}}$ from $P(\mathbf{h}|\mathbf{v}_l = f_{t-1}, \mathbf{v}_r = f_t)$ respectively;
3. Sample $\tilde{\mathbf{v}}_l$ from $P(\mathbf{v}_l|\tilde{\mathbf{h}})$ as the background and update the model in the meantime;
4. Slide to the next pair of frames, $(f_{t-1}, f_t) \leftarrow (f_t, f_{t+1})$.

Note that the proposed model can be naturally updated online with similar update rules given new frames, making the background model adaptive and robust to possible changes in the future. The procedure of background modeling by TARBM is shown in Algorithm 1. The model is fine tuned along with background generation of each frame.

Algorithm 1: Framework of background modeling with TARBM

```
Split training frames into pairs of chunks;
Train TARBM model with learning rules in (19);
while f_t is not the last frame do
    For pair of frames (f_{t-1}, f_t), generate the
    background BG_t of f_t;
    Update the model parameters with the same rules;
    t ← t + 1;
```

4 Experiments

Datasets: We conduct experiments on two datasets: WallFlower Dataset (Toyama et al. 1999) and the dataset used in (Li et al. 2003), which is denoted as I2R here. The WallFlower dataset[1] consists of 7 video sequences covering the main challenges in background modeling or foreground detection, including moved objects, light switch and waving tress etc. The frames in each sequence are 120×160 color images. We train and test the frames as described in the script files associated with the dataset. One thing to mention about the WallFlower dataset is that it provides single static ground truth frame of the background for each sequence in spite of the temporal changes in the video, which implies that the evaluation may be unfair for our temporally adaptive model. On the other hand, I2R[2] consists of 9 video sequences, with 20 frames of ground truth masks provided for

[1] http://research.microsoft.com/en-us/um/people/jckrumm/wallflower/testimages.htm

[2] http://perception.i2r.a-star.edu.sg/bk_model/bk_index.html

each sequence. The frames are all color images, and can be of different sizes in different sequences.

Comparison: We compare our model with three representative methods for background subtraction including both parametric and non-parametric, which are EGMM (Zivkovic and van der Heijden 2006), Codebook model (Kim et al. 2005) and ViBe (Barnich and Van Droogenbroeck 2011) respectively. EGMM is an augmented algorithm based on the mixture of Gaussians model presented in (Stauffer and Grimson 1999). It can automatically adapt to the number of components needed to model one pixel, and is one of the state-of-the-art representatives of parametric methods. CodeBook model is a sample based algorithm, which samples values over time without any parametric assumptions to model the background. It is compact and capable of handling illumination changes (Kim et al. 2005). Similarly, Vibe is also a non-parametric method for background modeling, where a random selection policy is employed to ensure a smooth exponentially decaying lifespan for the sample values (Barnich and Van Droogenbroeck 2011). In addition, we also compare with standard RBMs to demonstrate the necessity of capturing temporal information in video sequences.

Parameters: In order to model the sharp illumination changes in most sequences in WallFlower, the size of the hidden layer both in RBM and TARBM is set to 400, while in I2R the parameter is set to 50 considering the relatively smooth variations in the sequences. The size of the visible layer is equal to the number of pixels, and λ in TARBM is set to 1. When training TARBM, the parameters $\mathbf{W}_l$, $\mathbf{W}_r$ and $\mathbf{b}$ are initialized randomly, while $\mathbf{c}_l$ and $\mathbf{c}_r$ are initialized with the mean of the training frames. The learning rate ϵ is fixed at $1e-3$, and the max epoches is 150. We also follow the tricks of momentum and weight-decay for increasing the speed of learning as advised in (Hinton 2010), which are set to 0.9 and $2e-4$ respectively. When testing new frames, the update rate of the parameters is set to $1e-2$. The parameters of the competing algorithms are adopted by default. Once the background is generated, the foreground mask of the test frame is obtained by thresholding the difference of the original frame and the background followed by a morphological closing operation to remove the isolated pixels (Dougherty 1992).

Evaluation Metrics: To evaluate the performance, we employ the traditional pixel-level measurement F_1-$measure$. Let $T.f$ denote the number of pixels classified as foreground correctly, $P.f$ denote the number of pixels classified as foreground, and $GT.f$ denote the number foreground pixels in the ground truth mask. The definition of the measurement follows:

$$recall = \frac{T.f}{GT.f} \quad precision = \frac{T.f}{P.f}$$
$$F_1 = 2\frac{precision \cdot recall}{precision + recall}.$$

As discussed above, the percentage of frames with ground truth masks is small, and the numerical criteria introduced above are not sufficient to evaluate the performance in dynamic background. Therefore, we will also include visual comparison to demonstrate the effectiveness of the proposed TARBM.

Experimental Result

Table 4 reports the results of foreground detection on the sequences in the WallFlower dataset except for *MovedObject*, which will be discussed separately. As shown in Table 4, our model TARBM could generate a comparable foreground in most of the sequences according to the numerical evaluation. On the sequences of *Bootstrap* and *LightSwitch*, we achieve the highest F_1 value compared with other methods. Sequence *Bootstrap* is a video where people are taking food, in which people move in every frame. The superior performance produced by TARBM on this sequence shows that our model could learn a relatively "clean" background even though the video is "polluted". The table also shows that general RBM is not suitable to capture the temporal information hidden in the video sequences especially in the sequences of *LightSwitch* and *TimeOfDay* where the illumination of background changes a lot over time.

The results on I2R are shown in Table 4. The background in these sequences is relatively stable, which implies the temporal dynamics among frames is rather small. As a consequence RBM and TARBM produce similar results while the latter is generally superior. Compared to traditional algorithms, methods based on RBMs achieve best results on four out of eight sequences as described.

Figure 3 shows a visual comparison of the foreground detected by various algorithms on the sequences in I2R. The top row is the ground truth, while the second row to the bottom correspond to the foreground detected by TARBM, EGMM, Vibe and Codebook respectively. One could observe that TARBM produces results significantly closer to the ground truth in terms of the shape and cleanness of the foreground in visual effects. This indicates that although the numerical measures are close, TARBM is more capable of detecting complex foreground.

Notice that the main purpose of our model is background modeling. To the best of our knowledge, current methods are mostly focused on foreground detection, which is subtly different to our problem. Figure 4 and Figure 5 demonstrate the ability of TARBM to model dynamic background in sequential frames. The first row is the frames taken from the video, while the second row is the corresponding background generated by TARBM, and the third row is the result of standard RBMs. As one can observe from Figure 4, standard RBM is sensitive to illumination changes, while TARBM is robust in the sense that the generated background changes along light-on or light-off. Frame sequences in Figure 5 depict the movements of telephone and chair after the man coming in. It is obvious that the background modeled by a standard RBM is interfered with the movement of the man, while TARBM generates background that is smooth in the static parts while adapting quickly to the movements.

5 Conclusion

In this paper we present a novel approach for background modeling in video sequences. To address the temporal na-

Table 1: Results on WallFlower dataset. Due to the lack of space, we use abbreviated names of the sequences, and the full names could be found at the provided website. The values are calculated based on the single ground truth mask.

Criterion	Method	Data sequences					
		Boot.	Camo.	Fore.	Ligh.	Time.	Wavi.
F_1	EGMM	0.5981	0.9478	0.3996	0.3036	**0.7596**	**0.9312**
	Codebook	0.5508	**0.9617**	**0.8963**	0.2917	0.1510	0.8392
	ViBe	0.4942	0.9061	0.6080	0.1890	0.4449	0.9228
	TARBM	**0.6513**	0.9515	0.5825	**0.5135**	0.3087	0.8814
	RBM	0.6177	0.9268	0.5464	0.1719	0.0243	0.8741

Table 2: Results on I2R. The dataset contains 9 sequences, one of them called *Bootstrap* is the same as the sequence in WallFlower and is removed.

Criterion	Method	Data sequences							
		Camp.	Curt.	Esca.	Foun.	Lobby	Shop.	Water.	Hall
F_1	EGMM	0.3468	0.3216	0.4765	0.5134	0.4535	0.6804	0.3411	0.4299
	Codebook	0.1399	0.2466	0.1816	0.5561	**0.6246**	0.2942	**0.9326**	0.1895
	ViBe	0.3682	0.8219	**0.5703**	0.5515	0.2669	0.6861	0.8587	**0.6201**
	TARBM	**0.4047**	**0.8256**	0.4196	0.6871	0.2033	**0.6943**	0.8979	0.5810
	RBM	0.3579	0.8174	0.4103	**0.7228**	0.1250	0.6799	0.8230	0.5330

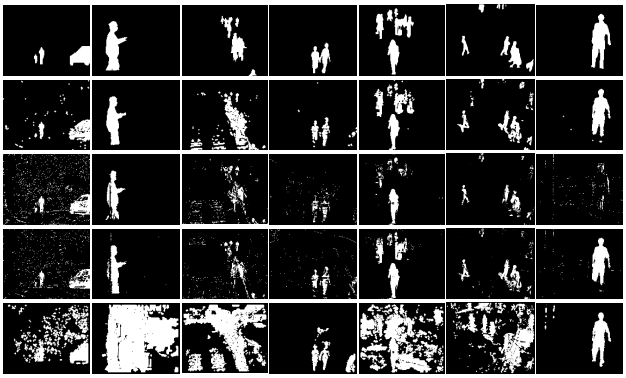


Figure 3: Foreground detected on sequences of *Campus*, *Curtain*, *Escalator*, *Fountain*, *Hall*, *ShoppingMall* and *WaterSurface*. Top to bottom rows: ground truth, TARBM, EGMM, ViBe and Codebook.

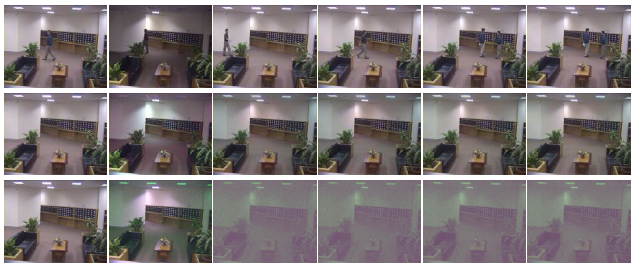


Figure 4: Visual comparison of background generated by TARBM and RBM on *Lobby* in dataset WallFlower. Top to bottom rows: frames taken from the video, TARBM and RBM.

ture of the problem, we incorporate the temporal correlation between adjacent video frames in the framework of

Figure 5: Visual comparison of background generated by TARBM and RBM on *MovedObject* in dataset WallFlower. Top to bottom rows: frames taken from the video, TARBM and RBM.

Restricted Boltzmann Machines and propose a temporally adaptive RBM. As a result, the reconstructed background generated by the model is stable and robust to noisy inputs and quickly adapt to changes in video background. Furthermore, the training procedure of the TARBM keeps all the advantages of standard RBMs including exact inference and effective learning, and the trained model can be updated online.

In future work, we would like to exploit the parallel nature of RBM training to achieve real-time background generation. It is also important to explore the possibilities of applying the temporally adaptive model to more general problems including dimensionality reduction and feature learning in sequential data.

Acknowledgments

Research supported by the National Natural Science Foundation of China (No. 61375060), the Fundamental Research Funds for the Central Universities (WK0110000036), and the National Science Foundation for Distinguished Young Scholars of China (No. 61325010).

References

Barnich, O., and Van Droogenbroeck, M. 2011. Vibe: A universal background subtraction algorithm for video sequences. *Image Processing, IEEE Transactions on* 20(6):1709–1724.

Chen, S.; Zhang, J.; Li, Y.; and Zhang, J. 2012. A hierarchical model incorporating segmented regions and pixel descriptors for video background subtraction. *Industrial Informatics, IEEE Transactions on* 8(1):118–127.

Coates, A.; Ng, A. Y.; and Lee, H. 2011. An analysis of single-layer networks in unsupervised feature learning. In *International Conference on Artificial Intelligence and Statistics*, 215–223.

Dougherty, E. R. 1992. An introduction to morphological image processing. *Tutorial Texts in Optical Engineering*.

Elgammal, A.; Harwood, D.; and Davis, L. 2000. Nonparametric model for background subtraction. In *Computer Vision–ECCV 2000*, 751–767. Springer.

He, J.; Balzano, L.; and Szlam, A. 2012. Incremental gradient on the grassmannian for online foreground and background separation in subsampled video. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 1568–1575. IEEE.

Hinton, G. E., and Salakhutdinov, R. R. 2006. Reducing the dimensionality of data with neural networks. *Science* 313(5786):504–507.

Hinton, G. E. 2002. Training products of experts by minimizing contrastive divergence. *Neural Computation* 14(8):1771–1800.

Hinton, G. 2010. A practical guide to training restricted boltzmann machines. *Momentum* 9(1).

KaewTraKulPong, P., and Bowden, R. 2002. An improved adaptive background mixture model for real-time tracking with shadow detection. In *Video-Based Surveillance Systems*, 135–144. Springer.

Kim, K.; Chalidabhongse, T. H.; Harwood, D.; and Davis, L. 2005. Real-time foreground-background segmentation using codebook model. *Real-time Imaging* 11(3):172–185.

Larochelle, H., and Bengio, Y. 2008. Classification using discriminative restricted boltzmann machines. In *Proceedings of the 25th International Conference on Machine Learning*, 536–543. ACM.

Li, L.; Huang, W.; Gu, I. Y.; and Tian, Q. 2003. Foreground object detection from videos containing complex background. In *Proceedings of the Eleventh ACM International Conference on Multimedia*, 2–10. ACM.

Pal, A.; Schaefer, G.; and Celebi, M. E. 2010. Robust codebook-based video background subtraction. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, 1146–1149. IEEE.

Piccardi, M. 2004. Background subtraction techniques: a review. In *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, volume 4, 3099–3104. IEEE.

Salakhutdinov, R.; Mnih, A.; and Hinton, G. 2007. Restricted boltzmann machines for collaborative filtering. In *Proceedings of the 24th International Conference on Machine Learning*, 791–798. ACM.

Stauffer, C., and Grimson, W. E. L. 1999. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2. IEEE.

Sutskever, I., and Hinton, G. E. 2007. Learning multilevel distributed representations for high-dimensional sequences. In *International Conference on Artificial Intelligence and Statistics*, 548–555.

Sutskever, I.; Hinton, G. E.; and Taylor, G. W. 2009. The recurrent temporal restricted boltzmann machine. In *Advances in Neural Information Processing Systems*, 1601–1608.

Taylor, G. W.; Hinton, G. E.; and Roweis, S. T. 2006. Modeling human motion using binary latent variables. In *Advances in Neural Information Processing Systems*, 1345–1352.

Toyama, K.; Krumm, J.; Brumitt, B.; and Meyers, B. 1999. Wallflower: Principles and practice of background maintenance. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 1, 255–261. IEEE.

Welling, M.; Rosen-Zvi, M.; and Hinton, G. E. 2004. Exponential family harmoniums with an application to information retrieval. In *Advances in Neural Information Processing Systems*, 1481–1488.

Zivkovic, Z., and van der Heijden, F. 2006. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters* 27(7):773–780.

Zivkovic, Z. 2004. Improved adaptive gaussian mixture model for background subtraction. In *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on*, volume 2, 28–31. IEEE.