Hindawi
Mathematical Problems in Engineering
Volume 2018, Article ID 6323942, 13 pages
https://doi.org/10.1155/2018/6323942

Research Article

Learning Multimodal Deep Representations for Crowd Anomaly Event Detection

Shaonian Huang,[1,2] **Dongjun Huang**,[1] **and Xinmin Zhou**[2]

[1]*School of Information Science and Engineering, Central South University, Changsha 410083, China*
[2]*Key Laboratory of Hunan Province for New Retail Virtual Reality Technology, School of Computer and Information Engineering, Hunan University of Commerce, Changsha 410205, China*

Correspondence should be addressed to Shaonian Huang; hsn@hunnu.edu.cn

Received 22 August 2017; Revised 6 December 2017; Accepted 8 January 2018; Published 31 January 2018

Academic Editor: Tae Choi

Anomaly event detection in crowd scenes is extremely important; however, the majority of existing studies merely use hand-crafted features to detect anomalies. In this study, a novel unsupervised deep learning framework is proposed to detect anomaly events in crowded scenes. Specifically, low-level visual features, energy features, and motion map features are simultaneously extracted based on spatiotemporal energy measurements. Three convolutional restricted Boltzmann machines are trained to model the mid-level feature representation of normal patterns. Then a multimodal fusion scheme is utilized to learn the deep representation of crowd patterns. Based on the learned deep representation, a one-class support vector machine model is used to detect anomaly events. The proposed method is evaluated using two available public datasets and compared with state-of-the-art methods. The experimental results show its competitive performance for anomaly event detection in video surveillance.

1. Introduction

With the rapidly increasing demand for public safety and security, surveillance videos are extremely important in security monitoring of public places. Among the various applications of surveillance videos, anomaly event detection is becoming one of the fundamental challenges and has attracted considerable attention from both academia and industry in recent years [1–4]. However, it is still relatively difficult to design a general framework for anomaly event detection and localization owing to the typical difficulties of anomaly detection.

One fundamental difficulty is the definition of an anomaly event, which varies significantly for different types of video scenes [2]. In general, an anomaly refers to an irregular event that occurs rarely in long time videos. However, the anomaly detection task is extremely difficult because enumerating all possible anomalies in a given surveillance video is infeasible. Therefore, such task needs to identify anomalies based on given the normal training events. One common solution to this one-class learning problem is to learn normal event patterns from training videos and then detect anomaly events based on the distance between the test video and the normal event patterns.

Another difficulty in anomaly event detection is how to extract discriminative features to model video events [2]. With the rapid development in computer vision techniques, many studies have modeled video events from various perspectives. For example, low-level features such as the histogram of gradient and the histogram of optical flow are calculated to describe the space–time distribution of motion patterns [5]; however, these low-level feature descriptions are usually hand-crafted. In recent years, many studies have modeled video events using object trajectories [6, 7]. Trajectory-based features usually capture the high-level semantic relations between motion objects; however, these methods need to implicitly segment and track each object in crowd scenes, which is still a challenging issue due to occlusions and camera motion.

Currently, deep learning architectures have shown impressive performance for various tasks in computer vision, such as image segmentation [8], image classification [9],

object detection [10], and activity recognition [11]. These works mainly focus on supervised learning applications with convolutional neural networks; however, labeled anomaly events in surveillance videos are often difficult to obtain in the field of anomaly event detection. Fortunately, unsupervised deep learning methods have also been studied to address important tasks such as object tracking [12] and face alignment [13]. Nevertheless, studies on unsupervised deep learning architecture are rare in the field of anomaly event detection. Xu et al. [14] first introduced an unsupervised deep learning framework to learn a deep representation of appearance and motion based on stacked denoising autoencoders [15]. Anomaly events are then detected by fusing three one-class support vector machines (SVMs). Feng et al. [16] developed a deep Gaussian mixture model (GMM), which stacks multiple GMM layers on top of each other to learn normal event models, where high-level feature descriptors are extracted by training a principal component analysis network (PCANet) [17] from 3D gradients.

In this study, we propose a novel multimodal deep representations framework for anomaly event detection in complex crowded surveillance videos. In contrast to previous studies [14, 21], instead of just using appearance or motion features or other low-level features to model the event patterns, we propose to learn the discriminative deep feature representation by fusing multimodal features including mid-level visual features, spatiotemporal energy features, and multiscale motion map features. A novel framework based on convolutional deep belief networks (CDBNs) [22] is introduced to achieve this goal. Foreground video patches are first extracted based on the spatiotemporal energies of the video sequence, and then low-level visual features, low-level energy features, and low-level motion map features are utilized as inputs of the three separate CDBNs to first learn mid-level feature representations. To further learn semantic deep feature representation, these three mid-level features are input to a multimodal fusion scheme to learn high-level deep features. Finally, a one-class SVM classifier is introduced to predict the anomaly events.

Compared with existing algorithms, contributions of this study are listed as follows:

(i) We propose a new unsupervised deep learning framework to learn mid-level feature representations by incorporating low-level visual features, low-level energy features, and low-level motion map features.

(ii) We introduce a multimodal fusion framework fusing mid-level deep features to learn semantic feature representations for modeling normal crowd events.

(iii) We obtain a comparative performance of our multimodal deep feature representation with state-of-the-art methods for anomaly detection on challenging anomaly datasets.

2. Related Works

Recent advances in anomaly event detection model the discrimination between normal and abnormal patterns. Normal patterns are first modeled from training data, and then abnormal patterns are detected as test samples having minimum error decision values. Generally, existing works for anomaly event detection can be roughly divided into trajectory-based methods and spatiotemporal patch-based methods.

For trajectory-based methods, trajectory extraction is usually the first step; then trajectory cluster or trajectory-based feature representation is performed to model the normal event patterns. For example, Piciarelli et al. [23] clustered groups of trajectories sharing similar features based on a single-class SVM; anomaly detection then became as only a matter of comparing the testing trajectory with the cluster model. Cui et al. [24] modeled the interaction energy potential of tracking points based on the positions and velocities of the neighboring points. In their method, the interaction energy potential function reflects the current state of a person and the relationship between current states, and then the corresponding reactions are explored to model the normal/abnormal patterns. In order to address occlusion and segmentation problems, Bera et al. [25] conducted a multiple-person tracking by a reciprocal velocity obstacle algorithm and represented the state of agent based on the ensemble Kalman filter to provide the position, velocity, and intermediate goal position of the current agent. Then the local feature representation and global feature representation of the pedestrian behavior are extracted based on the state of the agent. In addition, some works focused on tracking particle dynamics to model normal crowd flows. For example, Wu et al. [6] used Lagrangian particle trajectories to model the dynamics of crowd flow. Then the chaotic invariant feature from the representative trajectory can be extracted based on the largest Lyapunov exponent and correlation dimension.

As for spatiotemporal patch-based methods, event motion patterns are learned based on the 2D image patch or 3D video volume. Adam et al. [26] modeled the histograms of optical flow in a local region based on the exponential distribution. Mehran et al. [27] introduced a social force model to analyze crowd behavior, where interaction forces between pedestrian particles are calculated based on optical flow. Li et al. [18] developed a hierarchical mixture of dynamic textures (H-MDT) to model video representations, where temporal normalcy and spatial normalcy are modeled to detect the spatiotemporal behavior of crowd videos. In this method, a conditional random field (CRF) is used to measure the spatial and temporal abnormalities. Kim and Grauman [19] introduced a mixture of probabilistic principal component analyzers (MPPCA) to model local optical flow patterns. The space–time Markov random field (MRF) is then constructed to model the normal activities in the video. Recently, some works focusing on sparse representation have achieved promising results in the field of anomaly detection and localization. Cong et al. [28] proposed a sparse cost (SRC) over the normal dictionary to measure the normalness of a testing sample, where the training dictionary is designed with sparsity consistency constraint. However, this method is time consuming. Lu [20] et al. proposed a high-speed sparse combination learning framework to model normal events based on the 3D gradient features of a spatiotemporal video cube. This method has successfully detected anomaly event with a speed of 150 frames per second.

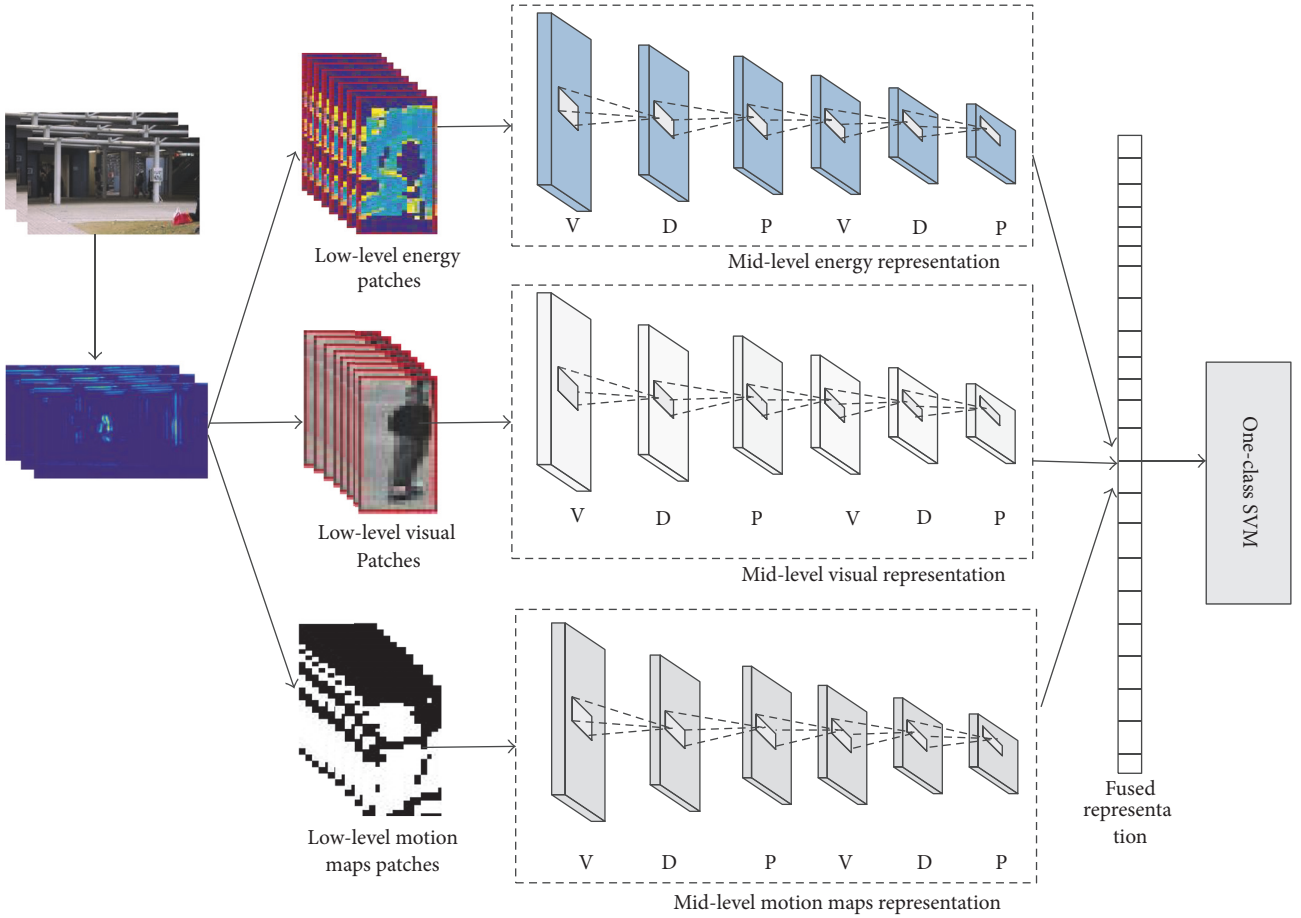


Figure 1: Overview of the proposed deep representation for crowd anomaly detection.

3. Learning Multimodal Deep Representations

In general, a typical approach to detect anomaly event is to learn model representation which describes the normal activities in the video scene and then discovers abnormal event by comparing feature patterns with the normal model representation. Among the proposed approaches in the literature, low-level visual appearance and motion feature based event model representation has gained much popularity [1, 6, 7, 14, 24, 27, 29]. Commonly used low-level features include color histograms from different color space, 3D spatiotemporal gradient, and histogram of optical flow. However, motion feature representation based on optical flow or spatiotemporal gradient fails to capture the motion variation in a long-range time. Meanwhile, many proposed approaches ignore the effect of motion pattern changes on the event model representation [1, 5, 7, 14, 24]. Based on the above observation, we develop a multimodal deep representation framework as shown in Figure 1 to generate more robust and complex representation of crowd event. In this framework, three low-level features are incorporated: low-level energy feature is used to capture the principal motion feature of the foreground motion objects, the low-level motion maps feature is designed to describe the motion patterns of the motion objects, and low-level visual feature is adopt to capture the appearance patterns of crowd event.

The proposed multimodal deep representation for anomaly event detection is based on three main stages. First, three CDBNs are introduced to learn the mid-level feature representation of crowd data. Particularly, to ensure the reliability of the model crowd event, different low-level features are incorporated including visual features, motion interactions, and distribution of the interaction features. In the second phase, these learned mid-level feature representations are further combined through a multimodal fusion scheme, resulting in a fused high-level deep representation that captures the correlation between the learned three mid-level feature representations. Finally, a one-class support vector machine (SVM) model is learned from all the learned multimodal deep representations. Then, whether any test crowd data is an anomaly can be judged based on the learning one-class SVM. In the following subsections, we will introduce the proposed approach in detail.

3.1. Convolutional Restricted Boltzmann Machines. The convolutional restricted Boltzmann machine (CRBM) is a generative model, which is trained to learn deep representations from image data [22]. The basic CRBM consists of an input visible layer and a hidden layer. The input layer consists of an $N_v \times N_v$ array of binary-valued or real-valued visible units. The hidden layer consists of a detection layer and a pooling

layer. Both detection layer and pooling layer have k groups of units. Each group in the detection layer includes an $N_h \times N_h$ array of binary units $\mathbf{h}$ and each group in the pooling layer includes an $N_p \times N_p$ array of binary units $\mathbf{p}$. The visible and detection layers are related by an $N_W \times N_W$ weight matrix $\mathbf{W}$, which is shared among the detected units in the current group across all the spatial locations to capture useful spatial structures in one part of an image.

Then, the detection layer is partitioned into $C \times C$ blocks of locally neighboring detected units (B_a), which are connected to one pooling unit in the pooling layer (p_a). Then the energy function of CRBM with probabilistic max-pooling can be defined as

$$\begin{aligned} &E(\mathbf{v}, \mathbf{h}) \\ &= -\sum_k \sum_{i,j} \left(h_{i,j}^k \left(\widetilde{\mathbf{W}}^k * v \right)_{i,j} + b_k h_{i,j}^k \right) \\ &\quad - c \sum_{i,j} v_{i,j} \\ &\text{subject to} \quad \sum_{(i,j) \in B_a} h_{i,j}^k \leq 1, \quad \forall k, a, \end{aligned} \tag{1}$$

where b_k is a shared bias for each group in the detection layer and c is a shared bias for the visible units, $*$ denotes convolution operation, and $\widetilde{\mathbf{W}}$ denotes flipping the weight matrix $\mathbf{W}$ horizontally and vertically. Similarly, if the visible units are real-valued, the energy function of CRBM with probabilistic max-pooling can be defined as follows:

$$\begin{aligned} &E(\mathbf{v}, \mathbf{h}) \\ &= \frac{1}{2} \sum_{i,j} v_{i,j}^2 \\ &\quad - \sum_k \sum_{i,j} \left(h_{i,j}^k \left(\widetilde{\mathbf{W}}^k * v \right)_{i,j} + b_k h_{i,j}^k \right) \\ &\quad - c \sum_{i,j} v_{i,j} \\ &\text{subject to} \quad \sum_{(i,j) \in B_a} h_{i,j}^k \leq 1, \quad \forall k, a. \end{aligned} \tag{2}$$

The joint probability distribution and conditional probability distribution are given by

$$P(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} \exp\left(-E(\mathbf{v}, \mathbf{h})\right), \tag{3}$$

where Z is a normalization constant.

$$\begin{aligned} P\left(h_{i,j}^k = 1 \mid \mathbf{v}\right) &= \frac{\exp\left(I\left(h_{i,j}^k\right)\right)}{1 + \sum_{(i',j') \in B_a} \exp\left(I\left(h_{i',j'}^k\right)\right)}, \\ P\left(p_a^k = 0 \mid \mathbf{v}\right) &= \frac{1}{1 + \sum_{(i',j') \in B_a} \exp\left(I\left(h_{i',j'}^k\right)\right)}, \end{aligned} \tag{4}$$

where $I(h_{i,j}^k) \triangleq b_k + (\widetilde{\mathbf{W}}^k * v)_{i,j}$.

$$P\left(v_{ij} = 1 \mid \mathbf{h}\right) = \sigma\left(\left(\sum_k \mathbf{W}^k * h^k\right)_{ij} + c\right) \quad \text{(for the binary visible units)}, \tag{5}$$

where $\sigma(\cdot)$ is the sigmoid function.

$$P\left(v_{ij} = 1 \mid \mathbf{h}\right) = G\left(\left(\sum_k \mathbf{W}^k * h^k\right)_{ij} + c\right) \quad \text{(for the real visible units)}, \tag{6}$$

where $G(\cdot)$ is a Gaussian distribution.

For training the CRBM, a sparsity penalty term is added to the log-likelihood objective to avoid trivial solutions [30]. The update rule is defined as follows:

$$\Delta b_k^{\text{sparsity}} \propto p - \frac{1}{N_H^2} \sum_{i,j} p\left(h_{ij}^k = 1 \mid \mathbf{v}\right). \tag{7}$$

For more details of the training algorithm, see [22, 30]. By stacking the CRBMs, a CDBN is formed to capture the deep feature representations.

3.2. Model Mid-Level Energy Representations with CDBN. The majority of the existing works use optical flow to model the feature representation of videos [1, 3, 18, 31, 32]; however, optical flow suffers from the problem of occlusions and camera motion. In this study, we propose principal spatiotemporal-oriented energy to represent the low-level motion feature. The local spatiotemporal orientation energy at each pixel $\mathbf{x} = (x, y, t)$ can be estimated using the third derivative of 3D Gaussian filters as follows [33]:

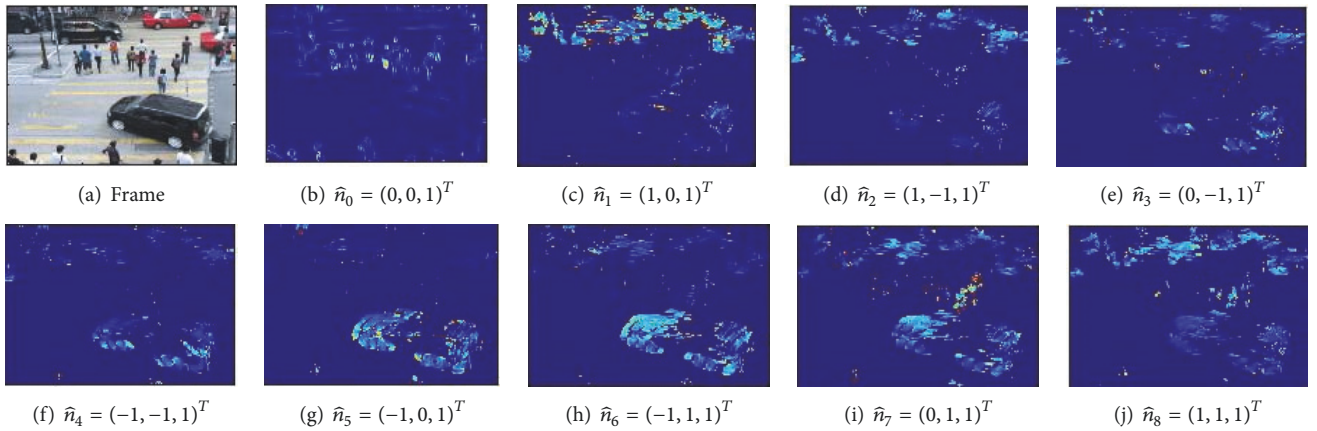
$$\text{SOE}_{\widehat{\theta}}(\mathbf{x}) = \sum_{\mathbf{x} \in \Omega} \left(G_{\widehat{\theta}}^3 * V \right), \tag{8}$$

where $G_{\widehat{\theta}}^3$ is the 3D Gaussian filter with the unit vector $\widehat{\theta}$ capturing the 3D direction of the filter symmetry axis, V denotes the input video, $*$ denotes the convolution operation, and Ω is a subregion around $\mathbf{x}$. However, owing to the separable characteristic of Gaussian steerable filters, the spatiotemporal response is usually processed by a "marginalization" step [34]; then the spatiotemporal orientation energy along a frequency domain plane with normal $\widehat{\mathbf{n}}$ is defined as follows:

$$\text{SOE}_{\widehat{\mathbf{n}}}(\mathbf{x}) = \sum_{i=0}^{d} \text{SOE}_{\widehat{\theta}_i(\widehat{\mathbf{n}})}(\mathbf{x}), \tag{9}$$

where d is the order of the Gaussian derivation and $\widehat{\theta}_i(\widehat{\mathbf{n}})$ is the space–time orientation (see [34] for the computation of $\widehat{\theta}_i(\widehat{\mathbf{n}})$).

In our implementation of energy measurement, we extract nine spatiotemporal orientation energies with different orientations as shown in Table 1. For illustrative purpose, Figure 2 displays the energies that are captured from a



(a) Frame (b) $\widehat{n}_0 = (0,0,1)^T$ (c) $\widehat{n}_1 = (1,0,1)^T$ (d) $\widehat{n}_2 = (1,-1,1)^T$ (e) $\widehat{n}_3 = (0,-1,1)^T$

(f) $\widehat{n}_4 = (-1,-1,1)^T$ (g) $\widehat{n}_5 = (-1,0,1)^T$ (h) $\widehat{n}_6 = (-1,1,1)^T$ (i) $\widehat{n}_7 = (0,1,1)^T$ (j) $\widehat{n}_8 = (1,1,1)^T$

Figure 2: Examples of crowd sequence spatiotemporal-oriented energies from the dataset: (a) a frame from the dataset; (b)–(j) spatiotemporal energies for the following directions of $\widehat{n}$: (b) static, (c) rightward, (d) upper right, (e) upward, (f) upper left, (g) leftward, (h) lower left, (i) downward, and (j) down right.

Table 1: Nine energies along different orientations.

	Energy measurement	Selected orientation
(1)	$\widehat{E}_s$	$\widehat{n}_0 = (0,0,1)^T$
(2)	$\widehat{E}_r$	$\widehat{n}_1 = (1,0,1)^T$
(3)	$\widehat{E}_{ru}$	$\widehat{n}_2 = (1,-1,1)^T$
(4)	$\widehat{E}_u$	$\widehat{n}_3 = (0,-1,1)^T$
(5)	$\widehat{E}_{lu}$	$\widehat{n}_4 = (-1,-1,1)^T$
(6)	$\widehat{E}_l$	$\widehat{n}_5 = (-1,0,1)^T$
(7)	$\widehat{E}_{ld}$	$\widehat{n}_6 = (-1,1,1)^T$
(8)	$\widehat{E}_d$	$\widehat{n}_7 = (0,1,1)^T$
(9)	$\widehat{E}_{rd}$	$\widehat{n}_8 = (1,1,1)^T$

single frame of a crowd sequence. It can be observed that the extracted energies can capture the local spatiotemporal structure of different orientations; for example, the rightward energy in Figure 2(c) has a strong response to the cars driving east, the down left energy in Figure 2(h) captures the movement of the black car turning left, and the static energy in Figure 2(b) has an obvious response to the pedestrians waiting on the zebra crossing.

The nine extracted energy measurements can capture the local motion patterns along different orientations; however, we need to extract the crowd motion dynamics in a short period. In this study, we describe the global motion pattern by combining the spatiotemporal-oriented energies in the principal energy measurement as follows:

$$\mathrm{SOE}_p(\mathbf{x}) = \max_{1 \le i \le 8} \mathrm{SOE}_{\widehat{\mathbf{n}}_i}(\mathbf{x}), \tag{10}$$

where the subscript p denotes the principal energy measurement.

For illustrative purpose, Figure 3 displays the principal energy measurements that are captured from a crowd running sequence. It can be observed that the extracted principal energy measurements can efficiently differentiate between the background and foreground objects in the crowd scene. Based on this observation, we extract the foreground principal energy patches and then input these patches to a CDBN to capture the mid-level energy representation of the crowd scene. We divide the crowd video into volumes of size $w_v \times d_v \times l_v$. For each volume vol, we use the mean of the principal spatiotemporal energies in the volume as a measure of the foreground patch as follows:

$$s(\mathrm{vol}) = \frac{1}{w_v \times d_v \times l_v} \sum_{\mathbf{x} \in \mathrm{vol}} \mathrm{SOE}_p(\mathbf{x}). \tag{11}$$

Then if the value of s(vol) is larger than a given threshold, that volume is considered as a foreground patch. In our experiments, the value of the given threshold is set to 1.5 times the average principal spatiotemporal energies of the current crowd frame.

We use the extracted 3D foreground spatiotemporal energy patches to train the energy CDBN, which can learn mid-level energy representations from the original crowd video sequence. To capture the rich spatiotemporal energy attribution, we adopt a multiscale sliding window to extract dense crowd patches, which are then warped into $w_e \times h_e \times c_e$, where w_e and h_e are the width and height of each patch, respectively, and c_e is the number of channels. This energy CDBN has two detection layers. The first detection layer filters the inputs with 40 kernels, each of size $5 \times 5 \times c_e$. The first detection layer is followed by a probabilistic max-pooling layer with its pooling factor set as 2. The second detection layer, with 60 kernels of $3 \times 3 \times c_e$, is then applied and followed by a probabilistic max-pooling layer with its pooling factor set also as 2.

3.3. Model Mid-Level Visual Representations with CDBN. We use a CDBN to model the visual features of crowd anomaly events. In a previous research [22], CDBNs have demonstrated an excellent capability of capturing visual patterns in the application of object recognition and handwritten digital classification. Here, we are interested in studying how well they perform in the task of crowd anomaly event detection.

The low-level visual features of the foreground crowd patches extracted as described in Section 3.1 are utilized

Frame | The principal spatiotemporal energy

Figure 3: Illustration of principal spatiotemporal energies.

for training the visual CDBN to learn mid-level visual representations of crowd videos. The visual features of the all foreground crowd patches are linearly normalized into a range [0, 1]. This visual CDBN has two detection layers as well, and the other settings of the visual CDBN are the same as those of the energy CDBN explained in Section 3.1.

3.4. Model Mid-Level Multiscale Motion Maps with CDBN. The principal spatiotemporal energy of the foreground patches captures the pixelwise local motion energy; however, we observe that different foreground patches with different motion patterns may exhibit a similar principal energy distribution. Consider the three patches in Figure 4(a): the red patch containing a walking man with normal speed, the blue patch containing a running man with high speed, and the green patch containing a man leaning against a post. Figures 4(b), 4(c), and 4(d) represent the principal spatiotemporal energy distribution corresponding to these three patches. While these patches represent completely different motion patterns, one cannot easily distinguish them by only inspecting their principal energy distribution.

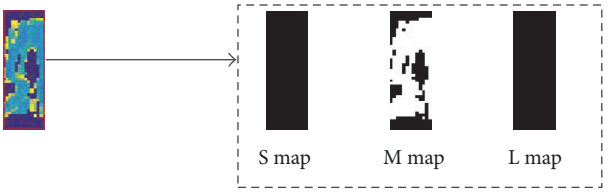
We propose multiscale energy maps to discriminate different energy patterns. An energy map is a binary image with elements covered by foreground pixels in a crowd patch set to 1. We use the multiscale energy maps to capture the multiple-scale level of principal spatiotemporal energy distribution, where different maps represent different energy scales. Specifically, we use three-scale energy map channels to represent detected principal energy patches, where each channel is a binary map with the same size as the principal energy patch. Two scale thresholds τ_1 and τ_2 ($\tau_1 < \tau_2$) are used to choose the different channels. Given a foreground crowd patch, we normalize the sum of nine energies with different spatiotemporal directions and then compute the mean of energy within the patch (M_{patch}). If $M_{\text{patch}} < \tau_1$, we project the patch into the first channel, setting all the foreground pixels in this patch to 1. Otherwise, the current patch is projected to the second or the third channel depending on whether $M_{\text{patch}} < \tau_2$ holds. From the right column in Figure 4, we can observe that the three patches with different motion patterns are projected to different multiscale energy maps, where the different patches can be easily distinguished by the different scale levels.

Similar to the motion speed, motion direction is another important factor to recognize different motion patterns in a video sequence [2]. In this study, we propose multiscale direction maps to capture the motion direction distribution of the extracted patches. For each patch, the direction of the principal spatiotemporal energy exists in the eight space–time orientations given in Table 1 (from $\widehat{n}_1$~$\widehat{n}_8$; $\widehat{n}_0$ in Table 1 is the static energy). Similar to the multiscale energy maps, we use multiscale direction maps composed of eight channels to represent the direction distribution of the extracted patch. In particular, for each pixel in the patch, if the space–time direction of the principal spatiotemporal energy is $\widehat{n}_1$, we project it to the first channel, setting the current pixel of the first channel to 1 and the other channels to 0. From Figure 5, we can observe that the two patches with different motion directions can be distinguished by the eight-channel direction maps.

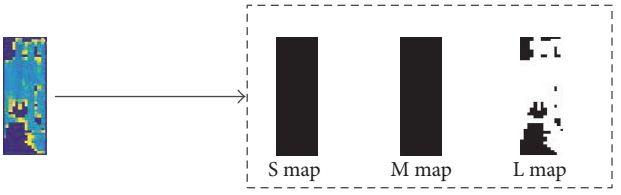
By stacking the multiscale energy maps and the direction maps for the extracted patches, we obtain an integrated map with 11 channels, each with the same size as the extracted patch. We construct a CDBN to derive the mid-level motion map representation. Similar to the energy CDBN explained in Section 3.1 and the visual CDBN presented in Section 3.2, these motion map CDBNs have two detection layers as well. The first detection layer contains 40 filters, each of size $5 \times 5 \times 11$. The second layer contains 60 filters, each with size of $3 \times 3 \times 11$. The other settings of the motion map CDBN are the same as those of the energy CDBN and visual CDBN. Particularly, the input units of these motion map CDBNs are binary-valued visible units, while the input units of the energy CDBN and visual CDBN are real-valued visible units.

3.5. Multimodal Deep Representations of Three Mid-Level Features. Our multimodal deep representation framework includes two steps: (1) training the visual CDBN, energy CDBN, and motion map CDBN to obtain three mid-level feature representations of the training crowd videos; (2) training the multimodal restricted Boltzmann machines to learn the correlations between the three mid-level features

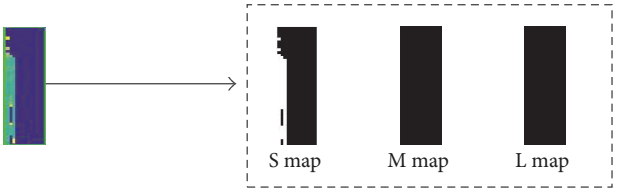
(a) Crowd frame



(b) Principal spatiotemporal energy of red patch

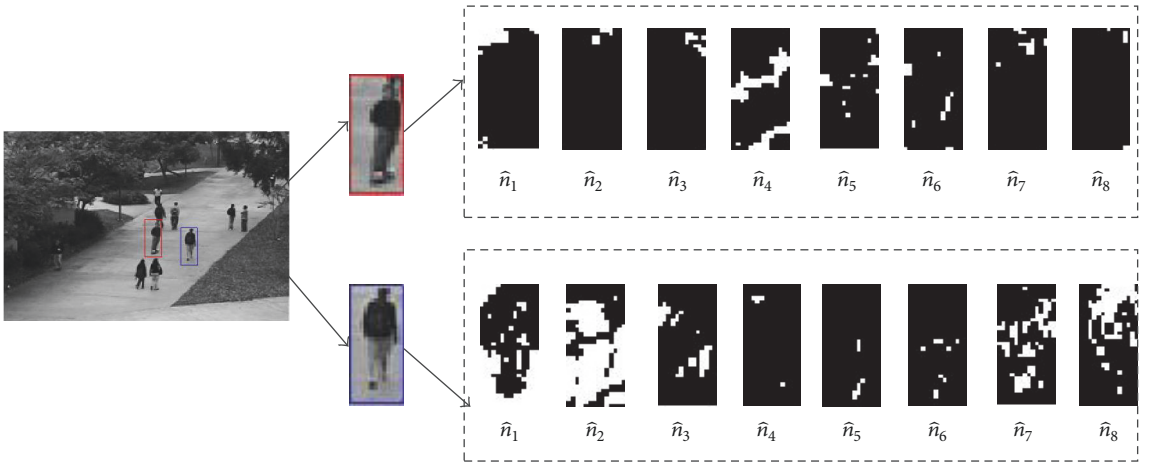


(c) Principal spatiotemporal energy of blue patch



(d) Principal spatiotemporal energy of green patch

Figure 4: Illustration of multiscale energy maps.



(a) Crowd frame (b) Two patches (c) Direction maps of the two patches

Figure 5: Illustration of multiscale direction maps.

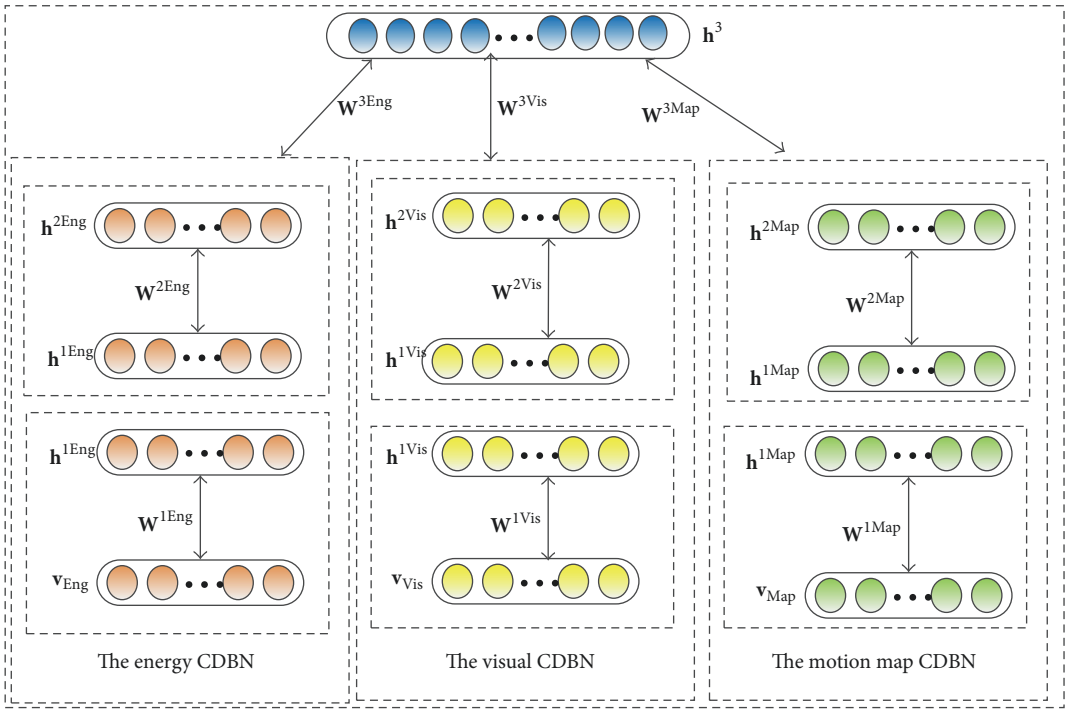


Figure 6: Multimodal deep representations framework.

and form multimodal deep feature representations. Figure 6 illustrates our multimodal deep representation framework.

To train the mid-level feature representations, we adopt a greedy, layer-by-layer unsupervised learning algorithm that can learn CDBN's one layer at a time [22]. Here we describe the training process of the energy CDBN; the training of the visual CDBN and the motion map CDBN is the same. The input visible units $\mathbf{v}_{\text{Eng}}$ in the energy CDBN are the spatiotemporal energy patches extracted in Section 3.2; once the first layer of the energy CDBN is trained, the parameters $\mathbf{W}^{1\text{Eng}}$, $\mathbf{b}^{1\text{Eng}}$, and $\mathbf{b}^{0\text{Eng}}$ are frozen and the hidden unit values $\mathbf{h}^{1\text{Eng}}$ of the first hidden layer are inferred. The inferred values $\mathbf{h}^{1\text{Eng}}$ serve as the input data used to train the second hidden layer in the energy CDBN and then the hidden unit values $\mathbf{h}^{2\text{Eng}}$ can be inferred. Similarly, the values of $\mathbf{h}^{2\text{Vis}}$ and $\mathbf{h}^{2\text{Map}}$ can be inferred.

To train a multimodal deep representation, we construct a multimodal restricted Boltzmann machine over the three sets of pretraining mid-level feature representations $\mathbf{h}^{2\text{Eng}}$, $\mathbf{h}^{2\text{Vis}}$, and $\mathbf{h}^{2\text{Map}}$. The proposed multimodal restricted Boltzmann machine is an undirected graphical model with stochastic binary units. It contains three sets of visible units $\mathbf{h}^{2\text{Eng}} \in \{0,1\}^{F_{2\text{Eng}}}$, $\mathbf{h}^{2\text{Vis}} \in \{0,1\}^{F_{2\text{Vis}}}$, and $\mathbf{h}^{2\text{Map}} \in \{0,1\}^{F_{2\text{Map}}}$ and stochastic hidden units $\mathbf{h}^3 \in \{0,1\}^{F_3}$. The energy of the joint configuration $\{\mathbf{h}^{2\text{Eng}}, \mathbf{h}^{2\text{Vis}}, \mathbf{h}^{2\text{Map}}, \mathbf{h}^3\}$ is defined as

$$
\begin{aligned}
&E\left(\mathbf{h}^{2\text{Eng}}, \mathbf{h}^{2\text{Vis}}, \mathbf{h}^{2\text{Map}}, \mathbf{h}^3; \theta\right) \\
&= \sum_{i=1}^{F_{2\text{Eng}}} b_i^{2\text{Eng}} h_i^{2\text{Eng}} + \sum_{i=1}^{F_{2\text{Vis}}} b_i^{2\text{Vis}} h_i^{2\text{Vis}} + \sum_{i=1}^{F_{2\text{Map}}} b_i^{2\text{Map}} h_i^{2\text{Map}} \\
&\quad + \sum_{j=1}^{F_3} b_j^3 h_j^3 + \sum_{i=1}^{F_{2\text{Eng}}} \sum_{j=1}^{F_3} h_j^3 W_{ji}^{3\text{Eng}} h_i^{2\text{Eng}} \\
&\quad + \sum_{i=1}^{F_{2\text{Vis}}} \sum_{j=1}^{F_3} h_j^3 W_{ji}^{3\text{Vis}} h_i^{2\text{Vis}} + \sum_{i=1}^{F_{2\text{Map}}} \sum_{j=1}^{F_3} h_j^3 W_{ji}^{3\text{Map}} h_i^{2\text{Map}},
\end{aligned}
\tag{12}
$$

where $\theta = \{\mathbf{b}^{2\text{Eng}}, \mathbf{b}^{2\text{Vis}}, \mathbf{b}^{2\text{Map}}, \mathbf{b}^3, \mathbf{W}^{3\text{Eng}}, \mathbf{W}^{3\text{Vis}}, \mathbf{W}^{3\text{Map}}\}$ are the model parameters ($\mathbf{b}^3$ is the bias term of the hidden layer $\mathbf{h}^3$ and $\mathbf{b}^{2\text{Eng}}$ is the bias term of the visible unit $\mathbf{h}^{2\text{Eng}}$) and $W_{ji}^{3\text{Eng}}$ represents the interaction term between hidden unit j and the visible unit $h_i^{2\text{Eng}}$. The conditional distributions over the three sets of visible units and the hidden units are given by

$$
p\left(h_k^{2\text{Eng}} = 1 \mid \mathbf{h}^3\right) = \sigma\left(b_k^{2\text{Eng}} + \sum_{i=1}^{F_3} W_{ki}^{3\text{Eng}} h_i^3\right),
$$

$$
p\left(h_k^{2\text{Vis}} = 1 \mid \mathbf{h}^3\right) = \sigma\left(b_k^{2\text{Vis}} + \sum_{i=1}^{F_3} W_{ki}^{3\text{Vis}} h_i^3\right),
$$

$$
p\left(h_k^{2\text{Map}} = 1 \mid \mathbf{h}^3\right) = \sigma\left(b_k^{2\text{Map}} + \sum_{i=1}^{F_3} W_{ki}^{3\text{Map}} h_i^3\right),
$$

$$
p\left(h_k^3 = 1 \mid \mathbf{h}^{2\text{Eng}}, \mathbf{h}^{2\text{Vis}}, \mathbf{h}^{2\text{Map}}\right) = \sigma\left(b_k^3\right.
$$

$$+\sum_{i=1}^{F_{2\mathrm{Eng}}} W_{ki}^{3\mathrm{Eng}} h_i^{2\mathrm{Eng}} + \sum_{i=1}^{F_{2\mathrm{Vis}}} W_{ki}^{3\mathrm{Vis}} h_i^{2\mathrm{Vis}} + \sum_{i=1}^{F_{2\mathrm{Map}}} W_{ki}^{3\mathrm{Map}} h_i^{2\mathrm{Map}}\Bigg), \tag{13}$$

where σ is the sigmoid function. Then the derivative of the log-likelihood with respect to the model parameters takes the form

$$\frac{\partial \log P\left(\mathbf{h}^{2\mathrm{Eng}}, \mathbf{h}^{2\mathrm{Vis}}, \mathbf{h}^{2\mathrm{Map}}; \theta\right)}{\partial \mathbf{W}^{3\mathrm{Eng}}} = E_{P_{\mathrm{data}}}\left[\mathbf{h}^{2\mathrm{Eng}}\left(\mathbf{h}^{3}\right)^{T}\right] - E_{P_{\mathrm{model}}}\left[\mathbf{h}^{2\mathrm{Eng}}\left(\mathbf{h}^{3}\right)^{T}\right], \tag{14}$$

where $E_{P_{\mathrm{data}}}[\cdot]$ denotes an expectation with respect to the complete data distribution and $E_{P_{\mathrm{model}}}[\cdot]$ denotes an expectation with respect to the distribution defined by the model. We use mean-field inference to estimate the value of $E_{P_{\mathrm{data}}}[\cdot]$ and an MCMC based stochastic approximate procedure to approximate the value of $E_{P_{\mathrm{model}}}[\cdot]$ [35]. Similarly, the derivative of the log-likelihood with respect to $\mathbf{W}^{3\mathrm{Vis}}$ and $\mathbf{W}^{3\mathrm{Map}}$ can be computed.

After learning the multimodal restricted Boltzmann machines, the posterior probability density of the hidden variables given the three sets of pretraining mid-level feature representations can be viewed as the multimodal deep representation of the training data. For any training crowd video patch $\mathbf{x}$, we denote its multimodal deep representation as $\mathbf{x}^m$.

4. Anomaly Event Detection and Localization with Deep Representations

The anomaly event detection problem in a crowd video is formulated as a binary classification problem. We adopt a one-class SVM model [36] to detect crowd anomaly events. The one-class SVM attempts to learn a hypersphere in the feature space, which can separate most of the training data from the original with a maximum margin, and then the small fraction of training data lying outside the hypersphere are considered as anomaly. Formally, given a set of training crowd patches $P = \{\mathbf{x}_i^m\}_{i=1}^N$, $\mathbf{x}_i^m \in \mathbb{R}^d$, a one-class SVM problem can be formulated as the following quadratic program:

$$\begin{aligned} \min_{\mathbf{w},\xi,\rho} \quad & \frac{\|\mathbf{w}\|^2}{2} - \rho + \frac{1}{\eta N}\sum_{i=1}^{N}\xi_i \\ \text{subject to} \quad & \mathbf{w}^T\phi\left(\mathbf{x}_i^m\right) \geq \rho - \xi_i, \\ & \xi_i \geq 0 \quad \forall i = 1, \ldots, N, \end{aligned} \tag{15}$$

where $\mathbf{w}$ is the learned weight vector defining the hypersphere, $\boldsymbol{\phi}(\cdot)$ is a feature map transforming the feature vector $\mathbf{x}_i^m$ into an inner product space by the kernel function, $\boldsymbol{\xi}_i$ is the slack variable for the training patch i, N is the size of the training dataset, ρ is the offset, and η is the regularization parameter, which is defined by the user to regulate the outlier fraction lying outside the hypersphere.

In practice, the quadratic program is transformed into the following dual problem:

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j k\left(\mathbf{x}_i^m, \mathbf{x}_j^m\right) \\ \text{subject to} \quad & 0 \leq \alpha_i \leq \frac{1}{\eta N}, \quad \sum_i \alpha_i = 1, \end{aligned} \tag{16}$$

where $k(\cdot)$ is the kernel function and $\boldsymbol{\alpha}$ is the Lagrange multiplier.

In our experiment, we use a radial basis function kernel, $k(\mathbf{x}_i^m, \mathbf{x}_j^m) = e^{-\|\mathbf{x}_i^m - \mathbf{x}_j^m\|^2/2\sigma^2}$. Given the optimal $\boldsymbol{\alpha}$ by solving (16), the optical weighting vector is given by $\mathbf{w} = \sum_i \alpha_i \phi(\mathbf{x}_i^m)$; then the decision function $f(\mathbf{x}) = \mathrm{sgn}(\mathbf{w}^T\phi(\mathbf{x}^m) - \rho)$ is used to determine whether the test crowd data is an anomaly.

5. Experiments

In this section, we systematically apply our proposed algorithm on two public datasets to verify its effectiveness. Three different measurements are applied to evaluate the anomaly detection accuracy. Then qualitative and quantitative comparisons are performed with state-of-the-art algorithms.

5.1. Evaluation Criterion. In order to evaluate the performance, three commonly used levels of measurements, namely, frame-level, pixel-level, and patch-level, which were introduced in [18, 20] and exploited in the majority of previous studies, are applied. The measurements are defined as follows.

(i) Frame-level: in this measurement, a frame is considered to be anomalous if at least one pixel in the frame is detected as anomalous. These frame-level detection results are compared to the frame-level ground truth of each frame; then the detection results are adopted to determine the number of true positive numbers and false positive numbers. However, this frame-level measurement cannot ensure that the detected anomalous pixel coincides with the actual anomalous position. This is because some portion of false pixel detection may cause a true positive frame.

(ii) Pixel-level: in this measurement, detection results are compared to the pixel-level ground truth of each frame. If 40% (or more) of anomalous ground truth pixels are identified as true positives, then the frame is considered to be an anomaly. On the other hand, a normal frame will be identified as false positive if any normal pixel is predicated as anomalous. Compared with the frame-level measurement, the pixel-level measurement is much stricter and focuses more on the correct localization of an anomaly event.

(iii) Patch-level: a high true positive rate (TPR) in pixel-level measurements always accompanies a high false positive rate (FPR). This is because more normal pixels may be considered as anomalous when more anomalous pixels are detected as true positives. The patch-level frame measurement focuses

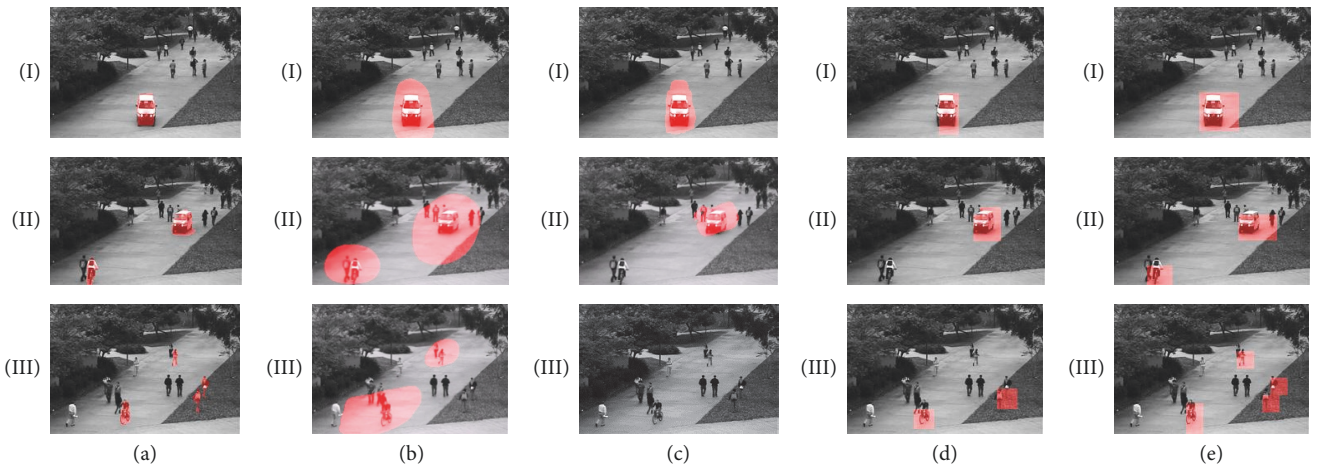


Figure 7: Examples of abnormal detection on UCSD Ped1 dataset: (a) the ground truth, (b) MDT [18], (c) MPPCA [19], (d) SCL [20], and (e) the proposed method.

more on the truly detected anomaly patch, and the true positive is defined as follows:

$$\frac{\text{Detected anomaly} \cap \text{True anomaly}}{\text{Detected anomaly} \cup \text{True anomaly}} \geq \lambda, \tag{17}$$

where λ is a predefined threshold.

The receiver operating characteristic (ROC) curve is employed to measure the detected accuracy based on the frame-level and pixel-level measurements. The ROC curve consists of TPR and FPR, which are defined as follows:

$$\begin{aligned} \text{TPR} &= \frac{\text{True positive}}{\text{True positive} + \text{False negative}}, \\ \text{FPR} &= \frac{\text{False positive}}{\text{False positive} + \text{True negative}}. \end{aligned} \tag{18}$$

Based on the ROC curve, the performance is summarized as the following evaluation criteria:

(i) Area under curve (AUC): the area under the ROC curve

(ii) Equal error rate (EER): the ratio of misclassified frames at which FPR = 1 − TPR for the frame-level measurements

(iii) Equal detected rate (EDR): the detection rate at which EDR = 1 − EER for the pixel-level measurements.

5.2. UCSD Dataset. The UCSD (http://www.svcl.ucsd.edu/projects/anomaly/dataset.htm) dataset is recorded with a static camera mounted at an elevation, overlooking pedestrian walkways on the UCSD campus. The crowd density in this dataset varies from sparse to very crowded. The dataset is organized into two subsets called Ped1 and Ped2. An anomaly event is indicated by the emergence of a car, skateboarder, wheelchair, or bicycle moving with the normal walking patterns of pedestrians. Ped1 contains 34 and 36 image sequences for training and testing, respectively, at a spatial resolution of 158 × 238. Ped2 has 16 training videos and 12 test videos with a resolution of 360 × 240. All training video frames in the dataset are normal, that is, containing only normal pedestrian walking patterns, and the test videos in Ped1 and Ped2 both contain image sequences with approximately 5500 normal frames and 3400 abnormal frames.

In the first series of experiments, we evaluate the performance of the proposed method using the UCSD dataset. For the mid-level energy feature learning, patches are extracted using a sliding window approach at 20 × 20 pixels. We train the energy CDBN with two hidden layers from the energy patches. The first hidden layer consists of 40 groups of filters, while the second hidden layer consists of 60. We use a fixed learning rate $\lambda = 0.05$ and a batch size $N_b = 256$. The pooling ratio C for each layer is set as 2, the target sparsity of the first hidden layer as 0.03, and the target sparsity of the second hidden layer as 0.02. For training of the visual CDBN and the motion map CDBN, the parameters are the same as those of the energy CDBN. For the semantic fusion of the multimodal representation, the mid-level visual features, mid-level energy features, and mid-level motion map features are the inputs of the multimodal RBM, which contains 120 hidden units. For one-class SVMs, the parameter η is tuned with the cross validation.

We compare our anomaly event detection framework with state-of-the-art approaches: mixture of dynamic textures (MDT) [18], social force model (SF) [27], mixture of optical flow models (MPPCA) [19], Adam et al.'s method [26], sparse reconstruction cost (SRC) method [37], sparse combination learning (SCL) method [20], and the appearance and motion DeepNet (AMDN) method [14]. Some testing results are shown in Figure 7, where the first column contains the ground truth, the second column contains the MDT [18] model results, and the third and fourth columns contain the results of MPPCA [19] and SCL [20] methods, respectively. The last column provides the results of the proposed method. For the MDT algorithm, its results contain a large quantity of background pixels, and the algorithm misses the two people walking across the grass (row III, column (b)). For MPPCA, it completely misses the biker and the people on the grass

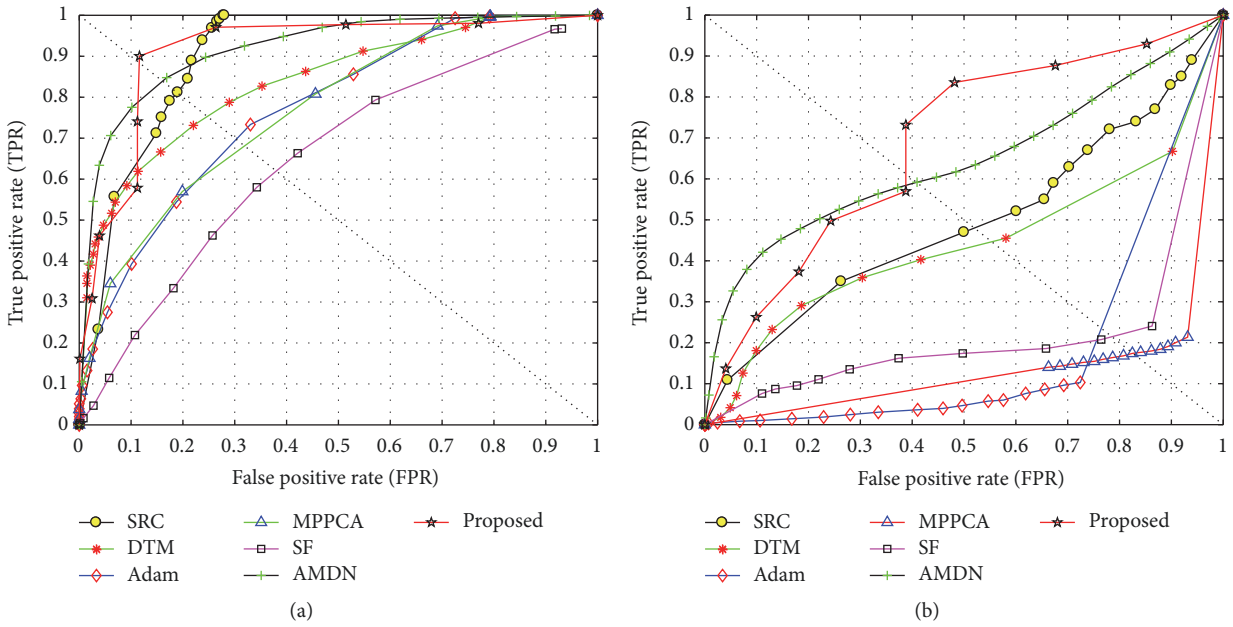


Figure 8: Results of UCSD Ped1 dataset: (a) frame-level ROC and (b) pixel-level ROC.

Table 2: Comparison of frame-level results on UCSD Ped1 dataset in terms of EER and AUC.

Algorithm	EER	AUC
SF [27]	31%	67.5%
MPPCA [19]	40%	76.96%
MDT [18]	25%	81.8%
Adam et al. [26]	38%	77.05%
SRC [37]	19%	86%
AMDN [14]	16%	92.1%
Proposed	11.2%	92.6%

Table 3: Comparison of pixel-level results on UCSD Ped1 dataset in terms of EDR and AUC.

Algorithm	EDR	AUC
SF [27]	21%	19.7%
MPPCA [19]	18%	20.5%
MDT [18]	45%	44.1%
Adam et al. [26]	24%	46.1%
SRC [37]	46%	48.7%
AMDN [14]	59.9%	67.2%
Proposed	61.3%	69.71%

(row III, column (c)). As for SCL, it misses the biker (row II, column (d)) and the runner (row III, column (d)). By contrast, the proposed method obtains satisfactory results and the overall performance is better than those of the others.

Figure 8 shows the frame-level and pixel-level detection results on Ped1 comparing our method with the state-of-the-art methods. Figure 8(a) shows the frame-level performance and Figure 8(b) shows the pixel-level performance. It can be observed that the ROC curve for our approach lies above the other six curves, which indicates that our approach is not only comparable to other methods, but also superior to them. Based on the ROC curves, Tables 2 and 3 show the quantitative comparisons in terms of AUC, EER, and EDR of our method against several state-of-the-art approaches. From these values, it is evident that our method outperforms the majority of previous methods for both frame-level and pixel-level measurements.

5.3. Avenue Dataset. The avenue dataset (http://www.cse.cuhk.edu.hk/leojia/projects/detectabnormal/dataset.html) contains 16 training and 21 testing video clips. The videos are captured in CUHK campus avenue with 30,652 (15,328 training and 15,324 testing) frames in total [20]. The main challenges of this dataset are due to slight camera shaking, a few outliers in the training data, and the absence of some normal motion patterns in the training data. The image resolution is 360×640 pixels.

Figure 9 illustrates some anomaly event detection results on the avenue dataset. In this figure, the first row is the ground truth, the second row the SCL algorithm [20] result, and the third row the proposed algorithm result. From Figure 9, we can observe that SCL misses the running boy (column (a)) and marks the normal pattern as abnormal (column (d)); however, the proposed algorithm detects the anomaly events correctly. Table 4 also presents the patch-level measurements by using different overlapping thresholds λ in (14). Compared with the SCL method, our method improves the average accuracy by 2.65%, which proves the effectiveness of multimodal deep representation.

6. Conclusions

This study presents a novel anomaly event detection method based on a multimodal deep learning framework. Specifically, effective video features based on spatiotemporal energy

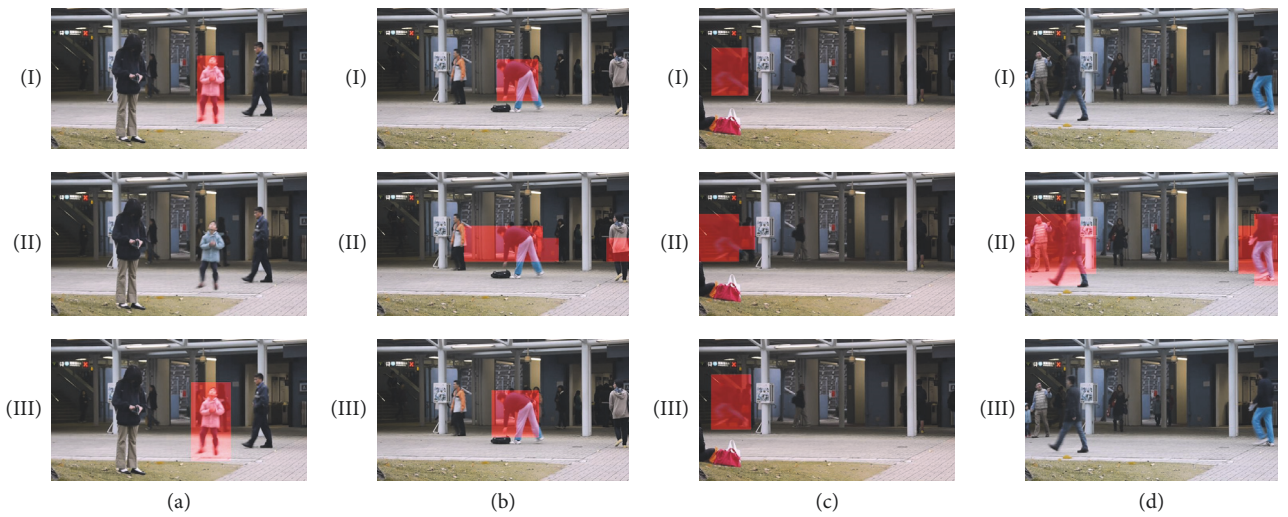


Figure 9: Examples of anomaly event detection on avenue dataset: (I) ground truth, (II) SCL method, and (III) the proposed algorithm.

Table 4: Comparison of patch-level results on avenue dataset.

λ	SCL	Proposed
0.2	72.3%	76.8%
0.3	71.1%	74.9%
0.4	68.8%	71.8%
0.5	65.8%	67.9%
0.6	64.1%	65.3%
0.7	62.9%	63.6%
0.8	61.9%	62.8%

measurements are automatically extracted to represent the low-level visual features, energy features, and motion map features. In order to learn the normal event patterns, three CDBNs are utilized to learn mid-level representations of normal event patterns. Then a multimodal RBM is trained to learn deep representations fused on the learned mid-level features. The extensive experiments on two challenging datasets demonstrate the effectiveness of the proposed method compared with state-of-the-art algorithms and prove the advantages of deep representation. In future studies, more efforts will be made to build other deep network architectures and construct alternative multimodal fusion schemes for anomaly detection in complex video scenes.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

With regard to the research, authorship, and/or publication, this work was financially supported by the Key Laboratory of Hunan Province for New Retail Virtual Reality Technology (2017TP1026), the Key Project of Hunan Provincial Education Department (Grant no. 17A113), and Hunan Provincial Philosophy and Social Science Fund (Grant no. 16YBA228).

References

[1] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos, "Anomaly detection in crowded scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 1975–1981, San Francisco, Calif, USA, June 2010.

[2] A. A. Sodemann, M. P. Ross, and B. J. Borghetti, "A review of anomaly detection in automated surveillance," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 42, no. 6, pp. 1257–1272, 2012.

[3] M. Sabokrou, M. Fathy, M. Hoseini, and R. Klette, "Real-time anomaly detection and localization in crowded scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW '15)*, pp. 56–62, USA, June 2015.

[4] A. Del Giorno, J. A. Bagnell, and M. Hebert, "A discriminative framework for anomaly detection in large videos," in *Proceedings of the European Conference on Computer Vision (ECCV '16)*, vol. 9909 of *Lecture Notes in Computer Science*, pp. 334–349, Springer, Cham, 2016.

[5] B. Zhao, L. Fei-Fei, and E. P. Xing, "Online detection of unusual events in videos via dynamic sparse coding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 3313–3320, IEEE, June 2011.

[6] S. Wu, B. E. Moore, and M. Shah, "Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '10)*, pp. 2054–2060, San Francisco, Calif, USA, June 2010.

[7] F. Jiang, J. Yuan, S. A. Tsaftaris, and A. K. Katsaggelos, "Anomalous video event detection using spatiotemporal context," *Computer Vision and Image Understanding*, vol. 115, no. 3, pp. 323–333, 2011.

[8] C. Farabet, C. Couprie, L. Najman, and Y. Lecun, "Learning hierarchical features for scene labeling," *IEEE Transactions on*

Pattern Analysis and Machine Intelligence, vol. 35, no. 8, pp. 1915–1929, 2013.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the 27th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14)*, pp. 580–587, Columbus, Ohio, USA, June 2014.

[11] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '16)*, pp. 1933–1941, USA, July 2016.

[12] N. Wang and D.-Y. Yeung, "Learning a deep compact image representation for visual tracking," in *Advances in Neural Information Processing Systems*, pp. 809–817, 2013.

[13] J. Zhang, S. Shan, M. Kan, and X. Chen, "Coarse-to-Fine Auto-encoder Networks (CFAN) for real-time face alignment," in *Proceedings of the European Conference on Computer Vision (ECCV '14)*, vol. 8690 of *Lecture Notes in Computer Science*, pp. 1–16, Springer, Cham, 2014.

[14] D. Xu, Y. Yan, J. Song, and N. Sebe, "Learning deep representations of appearance and motion for anomalous event detection," in *Proceedings of the British Machine Vision Conference (BMVC '15)*, Swansea, UK, 2015.

[15] P. Vincent, H. Larochelle, I. Lajoie, and P. Manzagol, "Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion," *Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.

[16] Y. Feng, Y. Yuan, and X. Lu, "Learning deep event models for crowd anomaly detection," *Neurocomputing*, vol. 219, pp. 548–556, 2017.

[17] T.-H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma, "PCANet: a simple deep learning baseline for image classification?" *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5017–5032, 2015.

[18] W. Li, V. Mahadevan, and N. Vasconcelos, "Anomaly detection and localization in crowded scenes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 1, pp. 18–32, 2014.

[19] J. Kim and K. Grauman, "Observe locally, infer globally: a space-time MRF for detecting abnormal activities with incremental updates," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR '09)*, pp. 2921–2928, Miami, Fla, USA, June 2009.

[20] C. Lu, J. Shi, and J. Jia, "Abnormal event detection at 150 FPS in MATLAB," in *Proceedings of the 2013 14th IEEE International Conference on Computer Vision (ICCV '13)*, pp. 2720–2727, Australia, December 2013.

[21] V. Reddy, C. Sanderson, and B. C. Lovell, "Improved anomaly detection in crowded scenes via cell-based analysis of foreground speed, size and texture," in *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW '11)*, pp. 55–61, IEEE, June 2011.

[22] H. Lee, R. Grosse, R. Ranganath, and A. Y. Ng, "Unsupervised learning of hierarchical representations with convolutional deep belief networks," *Communications of the ACM*, vol. 54, no. 10, pp. 95–103, 2011.

[23] C. Piciarelli, C. Micheloni, and G. L. Foresti, "Trajectory-based anomalous event detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1544–1554, 2008.

[24] X. Cui, Q. Liu, M. Gao, and D. N. Metaxas, "Abnormal detection using interaction energy potentials," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 3161–3167, June 2011.

[25] A. Bera, S. Kim, and D. Manocha, "Realtime Anomaly Detection Using Trajectory-Level Crowd Behavior Learning," in *Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW '16)*, pp. 50–57, USA, July 2016.

[26] A. Adam, E. Rivlin, I. Shimshoni, and D. Reinitz, "Robust real-time unusual event detection using multiple fixed-location monitors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 3, pp. 555–560, 2008.

[27] R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '09)*, pp. 935–942, Miami, Fla, USA, June 2009.

[28] Y. Cong, J. Yuan, and J. Liu, "Sparse reconstruction cost for abnormal event detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 3449–3456, Providence, RI, USA, June 2011.

[29] Y. Xiong, K. Zhu, D. Lin, and X. Tang, "Recognize complex events from static images by fusing deep channels," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '15)*, pp. 1600–1609, USA, June 2015.

[30] H. Lee, C. Ekanadham, and A. Y. Ng, "Sparse deep belief net model for visual area V2," in *Advances in Neural Information Processing Systems*, pp. 873–880, 2008.

[31] D. Du, H. Qi, Q. Huang, W. Zeng, and C. Zhang, "Abnormal event detection in crowded scenes based on Structural Multi-scale Motion Interrelated Patterns," in *Proceedings of the 2013 IEEE International Conference on Multimedia and Expo (ICME '13)*, pp. 1–6, USA, July 2013.

[32] J. Shao, C. C. Loy, K. Kang, and X. Wang, "Slicing convolutional neural network for crowd video understanding," in *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '16)*, pp. 5620–5628, USA, July 2016.

[33] E. H. Adelson and J. R. Bergen, "Spa-tiotemporal energy models for the perception of motion," *Journal of the Optical Society of America A: Optics and Image Science, and Vision*, vol. 2, no. 2, pp. 284–299, 1985.

[34] K. G. Derpanis, M. Sizintsev, K. J. Cannons, and R. P. Wildes, "Action spotting and recognition based on a spatiotemporal orientation analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 3, pp. 527–540, 2013.

[35] R. Salakhutdinov and G. Hinton, "Deep boltzmann machines," in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, vol. 24, pp. 448–455, 2009a.

[36] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[37] Y. Cong, J. Yuan, and J. Liu, "Abnormal event detection in crowded scenes using sparse representation," *Pattern Recognition*, vol. 46, no. 7, pp. 1851–1864, 2013.