Integrated Multi-scale Event Verification in an Augmented Foreground Motion Space

Qin Gu[1,2](✉), Jianyu Yang[1], Wei Qi Yan[2], and Reinhard Klette[2]

[1] University of Electronic Science and Technology of China,
Chengdu 611731, Sichuan, People's Republic of China
guqin.uestc@outlook.com

[2] Auckland University of Technology, Auckland 1010, New Zealand

Abstract. Moving event verification plays an important role in intelligent traffic supervision systems. We propose a novel event-verification framework using a deep convolutional neural network (CNN) in a proposed augmented foreground-motion space. First, we use a Gaussian mixture model for extracting foreground targets and generate multi-scaled regions to speed-up object or behaviour detection in high-resolution input video frames. Second, we use an augmented foreground motion space to reduce (in a group of adjacent frames) the given video data, motion, and scale information. A CNN-based deep neural network is organised for joint object detection and behaviour verification. The contribution of this paper is to propose a solution for multi-scale event verification. We verify the performance of multi-scale event verification for three typical events via real complex road-intersection surveillance videos.

Keywords: Deep learning · Event verification
Convolutional neural network · Gaussian mixture model

1 Introduction

Vision-based intelligent surveillance is an active field due to its high credibility and relatively low costs [8]. Moving event verification plays an important role in intelligent traffic supervision systems, especially for traffic-violation monitoring and recording. The majority of related algorithms are divided into two main categories.

In the first category, the whole frame, possibly also including time-adjacent frames, is used to obtain a conclusive verification result of the current scene. These coarse scene-understanding frameworks have been widely used for the analysis of abnormal events, such as traffic-accident detection.

In the second category, there is a wide diversity of research focusing on object-centric event descriptions. Here, two steps are implemented in traditional

M. Paul et al. (Eds.): PSIVT 2017, LNCS 10749, pp. 488–500, 2018.
https://doi.org/10.1007/978-3-319-75786-5_39

event-verification frameworks, being object detection (recognition), and object behaviour analysis (recognition) in an interval of time.

In this paper, we focus on the second category and combine moving object detection as well as event verification with a convolutional neural network in a novel compressed feature space. Motion information is compressed into a 3-dimensional (3D) augmented foreground motion space for event representation. Then, a deep regional convolutional neural network is used for object-oriented event verification. We list the contributions of this paper:

1. an augmented foreground motion space for event feature representation,
2. a fast Gaussian mixture model (GMM) based region proposal method for automatically generating a group of regions of interest for real-time traffic event recognition, and
3. a particular deep convolutional neural network (CNN), trained for integrated object detection and behaviour recognition in video data.

The remainder of this paper is organized as follows. Section 2 presents related work. Section 3 details the specification of an augmented foreground motion space for feature representation and compression. A joint CNN framework for object detection and event verification is given in Sect. 4. Section 5 shows experimental results for verifying the proposed method. Section 6 concludes.

2 Related Work

There is already a considerable diversity of existing work in traffic-event detection and verification.

Scene-oriented approaches are used for abnormal event detection [16,19] in videos; this has been tackled directly without locating any moving objects. A conclusive classification (i.e., normal or abnormal) is video-frame-based.

To obtain a precise description for an event, we use two steps which include moving object detection and behaviour understanding, to verify an object-based traffic event.

First, it is beneficial to detect objects of interest and extract the spatial location for further object-based event verification. The GMM is used in [24] for vehicle detection in complex urban traffic scenes. We show that GMM is also of benefit for time-efficient multi-object detection and tracking while this algorithm alone can only be used to extract foreground regions and generate coarse hypotheses.

Template-related algorithms have also been proposed for object detection and event hypothesis generation. A robust object-detection framework can be based on Haar-like features and the use of a cascade AdaBoost classifier [21]. This is verified to be an effective and fast method for rigid and one-class object detection. Deformable part models (DPMs) [5] are proposed for vehicle verification by using a support vector machine (SVM) and histogram-of-gradient (HOG) features. The active basis model [22] has also been widely employed for vehicle detection [10,15] in traffic surveillance. With the assistance of a shared skeleton method, it can be easily trained with a considerable detection performance.

Recently, deep learning [13] achieves remarkable advances to solve these problems. It dramatically develops the performance of frame-wise object detection and recognition [6,12,18]. However, it is also time consuming to detect, track, and understand the objects, frame by frame, using a deep neural network.

Second, object tracking methods [7,17,23] reconstruct the moving path of the detected object for further moving pattern matching. These tracking-based behaviour understanding algorithms are able to represent various moving patterns, but they highly rely on continuous and accurate detection results. Subsequently, hidden Markov models (HMM) [1], Bayesian approaches [3], or 3D models [9] can also be used to understand the trajectory of moving targets.

On the other hand, we can directly use a recurrent neural network (RNN) and long-short-term memory (LSTM) mechanisms [2] to construct a system for spatial-temporal event recognition and verification. However, we need a large number of labeled samples for each possible event category to train such a network.

Different to existing work, our contribution in this paper is an integrated framework for real-time and multi-class event recognition for road intersections. Motion detection and event recognition are conducted with a deep convolutional neural network for a proposed augmented foreground motion space.

3 Feature Representation

This section presents our event feature representation method using an *augmented foreground motion space*. See Fig. 1 for an outline. It is subject to the following considerations:

Simplified Data Dimension. In our application, it is time-consuming to detect multi-class events in series of high resolution and colour frames. Hence, an event representation with simplified data dimensions is greatly beneficial for speed-up. We verified that it is more significant to include motion information of objects in multiple frames than colour information.

High Information Density. Usually, traditional multi-frame image processing methods (e.g. *background subtraction*, or *optical flow*) result in some information loss compared to the given images. We expect to have a simplified event representation method which is still close to ensuring completeness of information for subsequent object detection, objet tracking, and behaviour recognition, after multi-frame compression.

Effectiveness. Considering the real-world applications, the feature representation method should be effective for fast regional proposals, accurate event recognition for different kinds of objects, and an adaption for multi-scale objects.

A common RGB colour image I has pixel values $u(x,y) = (I_R, I_G, I_B)$, where $0 \leq I_R, I_G, I_B \leq G_{\max}$ [11]; coordinates x and y define the pixel locations in an

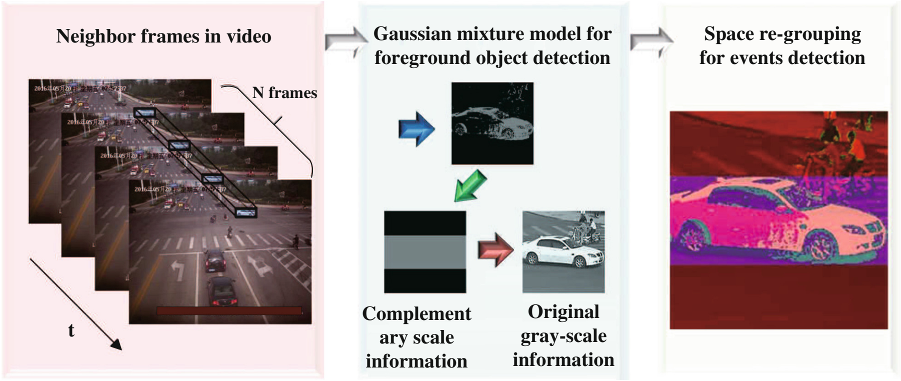
Neighbor frames in video
N frames
t
Gaussian mixture model for foreground object detection
Complement ary scale information
Original gray-scale information
Space re-grouping for events detection

Fig. 1. Framework of the proposed method. *Left:* Generation of a group of region proposals for moving objects. *Middle:* The procedure for feature representation merges original gray-scale images, foreground object detection results, and scale information for compressing the motion information of multiple frames via space re-grouping. *Right:* Further processing of an extracted region for joint event detection and recognition.

image. Value $G_{\max}$ usually equals $2^8 - 1$, or $2^{16} - 1$. For video analysis, we extend this to a 4D video V which has pixel values $v(x, y) = (I_R, I_G, I_B, t)$, where t corresponds to the time slot of a frame in a video.

In this paper, we represent pixel descriptors also in another 3D value space M; we set the value of each pixel as $M(x, y) = (M_U, M_V, M_S)$ where M_U, M_V, and M_S are the proposed 3D features at a pixel location (x, y) with respect to a short video sequence. M_U belongs to the *original image space* for the considered time slot t, M_V is the *complementary foreground space* information of this pixel location during this short video sequence, and M_S refers to the *complementary relative scale space* feature; for details see below.

The Original Image Space. The first value M_U at position (x, y) is the original gray level $\alpha I_R + \beta I_G + \gamma I_B$ of the current frame. It is used to preserve the local skeleton and texture features for object classification. In this paper, for convenience, we use $\alpha = \beta = \gamma = \frac{1}{3}$.

Complementary Foreground Region. To record motion information for a short sequence of adjacent frames, we use the second value M_V for specifying the foreground space. Using GMM, the pixel value at position (x, y) is described by $\{X_1,...,X_t\} = \{I(x, y, i) : 1 \leq i \leq t\}$. Here, $I(x, y, t)$ corresponds to the intensity value at position (x, y) at time t. All the pixels are represented by K $(3 \leq K \leq 5)$ states, and each state can be approximated using a Gaussian distribution:

$$p(X_t) = \sum_{k=1}^{K} \omega_{k,t} \cdot \mathit{\Gamma}(X_t \left| \mu_{k,t}, \mathbf{\Sigma}_{k,t} \right.)$$

where $\omega_{k,t}$ is the weight of k^{th} Gaussian distribution at time t, and $\mathit{\Gamma}(X_t\,|\mu_{k,t},\mathbf{\Sigma}_{k,t})$ is the probability density function of the k^{th} Gaussian distribution. By $\mu_{k,t}$ and $\mathbf{\Sigma}_{k,t}$ we denote the mean and covariance. Thus,

$$\mathit{\Gamma}(X_t\,|\mu_{k,t},\mathit{\Sigma}_{k,t}) = \frac{1}{\sqrt{2\pi\cdot|\mathbf{\Sigma}_k|}}\exp\left[-\frac{1}{2}(\mathbf{x}_t-\mu_k)^\top\mathbf{\Sigma}_k^{-1}(\mathbf{x}_t-\mu_k)\right]$$

A comparison between pixel value X_t and the Gaussian cluster is given as follows for matching:

$$X_t \in \mathit{\Gamma}(X_t\,|\mu_{k,t},\mathbf{\Sigma}_{k,t}),\text{ if }|X_t-\mu_{k,t}|<2.5\cdot\sigma_{k,t}$$

where $\sigma_{k,t}$ is the variance of the k^{th} cluster. Let T be an indicative function,

$$T(\alpha)=\begin{cases}1 & \text{if } \alpha \text{ is true}\\ 0 & \text{otherwise}\end{cases} \tag{1}$$

Then, an updating process of weights is implemented by

$$\omega_{k,t}=(1-\varepsilon)\omega_{k,t-1}+\varepsilon\cdot T(X_{t-1}\in\mathit{\Gamma}(X_{t-1}\,|\mu_{k,t-1},\mathbf{\Sigma}_{k,t-1}))$$

where ε is the updating learning rate of the video. Further updating, for a matched k^{th} cluster, is implemented by

$$\mu_{k,t}=(1-\delta)\mu_{k,t-1}+\delta X_t$$

$${\sigma_{k,t}}^2=(1-\delta){\sigma_{k,t-1}}^2+\delta(X_t-\mu_{k,t-1})^T(X_t-\mu_{k,t-1})$$

$$\delta=\varepsilon\cdot\mathit{\Gamma}(X_t\,|\mu_{k,t-1},\mathbf{\Sigma}_{k,t-1})$$

The first H distributions are taken as a model for the background, with

$$H=\arg\min_l(\sum_{k=1}^{l}\omega_{k,t}\geq\tau)$$

for a threshold $\tau>0$. H corresponds to the minimum number of distributions to construct the model of the background. Pixel value M_V in the second layer equals

$$M_{\mathrm{V}}=\frac{G_{\max}}{2}\cdot T(X_t\notin H)$$

where $T(\cdot)$ is still the indicative function as defined above.

In this layer, we try to include the dynamic motion information which refers to a micro-event related to object motion. As the initial region proposal is given by GMM, feature re-organization will only take very little time.

Complementary Relative Scale Space. As we expect to detect all the multi-scale targets and events in a wide supervising range, we need to handle objects at different scales. For example, for a hypothesis “a human” we search for a vertical rectangle, and for “a forward-looking vehicle” we search for a square.

We identify each proposed region hypothesis with a rectangle, and then with a square of minimum size, denoted by $S \in \mathbb{R}^{k \times k}$, for $k = \max(w, h)$. Here, S is the proposed region for further event verification, and w and h refer to width and height of the detected foreground hypothesis. Then, in the third layer of the proposed pixel value representation space, we compress the scale information into

$$M_S = \frac{G_{\max}}{2} \cdot T((x, y) \notin S \,|S_o\,)$$

where (x, y) is the position of the current pixel, S is the generated square hypothesis region (i.e. a square block of *complementary scale information*; see Fig. 1), and S_o is the original multi-scale hypothesis region (i.e. the gray area in the square block of the complementary scale information in Fig. 1).

4 Deep CNN for Micro-Event Detection and Verification

This section explains our framework for event verification. The proposed method consists of two phases. The first phase addresses the region proposal process. Each hypothetical motion region is extracted via this phase for further processing (aimed at verification). In the second phase, we train a deep convolution neural network to solve the problem of integrated object detection and event verification.

Region Proposal. Aiming at reducing the time-complexity of event verification, we avoid scanning each pixel or generating a large number of hypotheses with methods like selective search [20] in such large sized frames. As we only focus on events in the supervised region corresponding to moving targets in the video, the proposed method in this paper relies on GMM for the region proposal. The region proposal process is illustrated in Fig. 2.

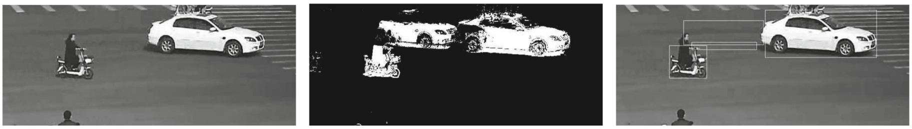

Fig. 2. *Left to right:* Illustration of the process of region proposal formation using GMM.

A detected (i.e. proposed) region is resized into 227 × 227 × 3 for further deep feature extraction and classification.

Network Layer Overview. In this paper, we use a convolutional layer, a max pooling layer, rectified linear units (ReLUs), and fully connected (FC) layers to construct our traffic event verification network.

Input data pass through all the organised layers to generate the final verification outputs. In the convolution layers, a group of kernels is used to filter the input such as to produce feature maps for deeper feature extraction. The function of the pooling layer is to calculate the overall response of a neighbour area in a feature map, which is one of the outputs of the convolution layer. Being aware of the problem of over-fitting, dropout layers are proposed for training optimisation. Finally, by using the softmax optimisation method, a multi-class identification result is given with an FC layer.

Network Architecture. We use a pre-trained model from the ImageNet Dataset [12]. During the training stage, over a million URLs of images have been used to obtain parameters for this network, and the whole architecture used in this paper is as shown in Fig. 3. At the end of this pre-trained network, the layers are designed to classify 1,000 objects.

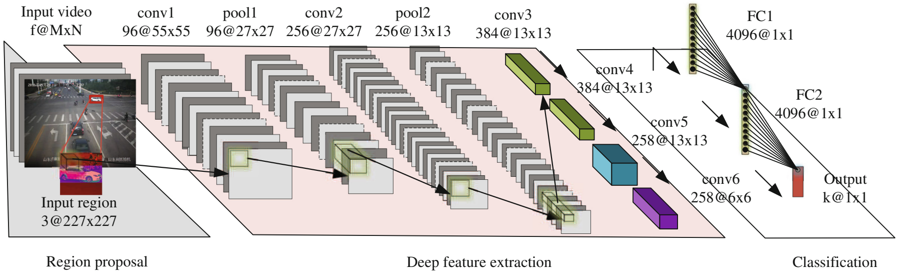

Fig. 3. Framework of the proposed micro-event verification using a deep CNN.

The detailed architecture of the deep neural network, adopted in this paper, consists of five convolution layers, seven ReLU layers, three max-pooling layers, and three FC layers. We generally divided this network into three main parts. First, the proposed region of interest is resized into $227 \times 227 \times 3$. In deep feature extraction, there are five main layers.The first convolutional layer has 96 kernels (all of size $11 \times 11 \times 3$).

After the convolution process with stride [4,4] and padding [0,0] and ReLU activity, we perform a normalisation with 5 channels per element. Then, a 3×3 max-pooling is used with stride [2,2] and padding [0,0]. Similarly, the second main layer of Part 2 consists of one convolutional layer sized $256@5 \times 5 \times 48$ with stride [1,1] and padding [0,0], ReLu activity, cross channel normalisation, and the same max-pooling as before. The third, fourth and fifth main layers all encompass one convolution layer ($384@3 \times 3 \times 256$, $384@3 \times 3 \times 192$, or $384@3 \times 3 \times 192$, respectively), and one ReLU activity layer.

After an additional max-pooling layer with stride [2,2] and padding [0,0], we extract a deep feature sized $1 \times 4,096$. There are three FC layers and two ReLU layers in the third part for multi-class classification.

Learning Details. Even though there is a big difference between the image in the ImageNet dataset for pre-training and the actually re-organised event representation in the augmented foreground motion space, it is still available to use the pre-trained network for further transfer learning. The reasons are:

The re-organised event representation in the augmented foreground motion space has three dimensions for each pixel. This is the same for the original colour image in the ImageNet dataset.

According to the examples of the event representation in the augmented foreground motion space, the skeleton of objects has been contained in the original image space. Information in this space can be easily learned from the pre-trained network.

Information in two other spaces (i.e. complementary foreground region and complementary relative scale space) also conclude relative edge and texture features for the original image. As a result, we can use a fine-tuning technology to abstract from the representation in these two augmented foreground motion spaces, respectively.

With such a deep convolutional neural network architecture, we use transfer learning technology to rebuild our own (event detection and verification) deep classifier for traffic scenes based on supervised learning. In this paper, we keep the architecture with the learned weights except for the last five layers. As we choose several events as our target, we segment the deep convolution neural network into two stages. The first 16 layers are taken for deep feature extraction, and the last 7 layers in our simulation are for classification.

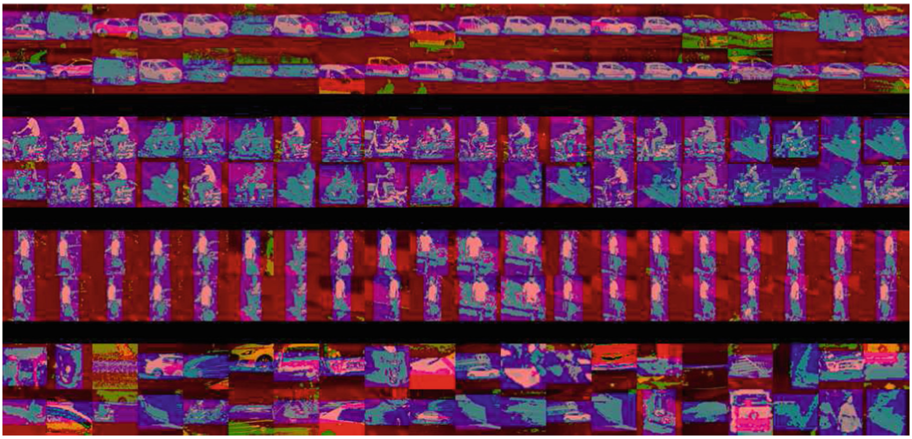

Fig. 4. Samples with some micro events in the augmented foreground motion space.

Totally, 3,622 event samples (see Fig. 4) are extracted and labeled manually from a video for further training for some special events such as vehicle horizontal traversal, motorcycle horizontal traversal, a motorcycle or pedestrian vertical traversal. The final seven layers are reorganized for event verification. First, we use an FC layer with 64 nodes, followed by an ReLU layer. Third, another FC layer with four nodes is given. The softmax optimisation process provides the final classification output.

Specifically, at the training stage, the batch size is set to be equal 128 in this paper, and the learning rate is set to be equal 0.0001 to fine-tune the network. There are 20 cycles being run at this training stage, each epoch means one complete pass through the training data. This training process costs 2,410.27 s with the assistance of a GPU in the used computer.

5 Experiments

The experimental report is divided into three segments. Detailed information of the dataset is given at the beginning. Then, we compare the performance of event localisation for different methods. Finally, we present the performance of event verification for extensive data recorded at a real traffic intersection using the proposed method.

Datasets. To evaluate the proposed method, we collect a dataset from downtown road intersections with a camera located about 8 meters over the road surface. Our videos record top or rear views of vehicles moving below the camera level. It is also possible to observe in the recorded data vehicles, motorcycles, and pedestrians on the other side of the intersection. We use four videos to evaluate the proposed method of moving object detection and behaviour verification. The videos were recorded at a frequency of 25 frames per second.

For event localisation, event descriptions of three time intervals of traffic videos are listed in Table 1. Each dataset contains 500 frames. The resolution of each frame is $2,592 \times 2,048$. We use each 10 adjacent frames to construct an event validation unit. The initial 5 frames are used for foreground region extraction and region proposals. Then, the last 5 frames are used for integrated object detection and behaviour verification.

Table 1. Validation datasets for traffic target detection and tracking, and event supervision.

	Resolution	Frames	Frames showing Event 1	Frames showing Event 2	Frames showing Event 3
Dataset 1	$2,592 \times 2,048$	500	48	44	35
Dataset 2	$2,592 \times 2,048$	500	107	129	27
Dataset 3	$2,592 \times 2,048$	500	68	50	-

Event Localisation. Normally, robust object detection and tracking are necessary for event localisation. In this paper, we compare the proposed integrated method and the traditional object detection method in a traffic scene at the intersection. In order to identify vehicle movements showing Event 1, we extract two hundred frames from the datasets and locate moving vehicles with the *active basis model* of [10], the *deformable part model* of [4], and our proposed method.

The active basis model performs well for rear-view vehicles, but it is difficult to train it for accurate vehicle event localisation in case of other viewing-angles. The deformable part based model is very accurate for detecting objects with a rigid structure, but it costs too much time to tackle one frame for one kind of targets even when using a cascade speed-up technology. Besides, we even need to cope with whole frames several times to extract different objects. The method proposed by us shows comparable results but proves to be much more time-efficient for extract multiple moving objects of interest. A further validation of verification accuracy is given in the next section.

Event Recognition. In this paper, we consider three types of significant events at a traffic intersection as examples, briefly identified as (i) vehicle horizontal traversal (i.e. left-to-right or right-to-left), (ii) motorcycle or bicycle horizontal traversal, or (iii) a motorcycle, bicycle, or pedestrian vertical traversal (i.e. top-down or bottom-up). We call those (i) Event 1, (ii) Event 2, and (iii) Event 3. Note that they may occur concurrently. They have been manually labeled, frame by frame, for having ground truth available. The total number of frames, showing each event, is given in Table 1.

Based on automatic detection and behaviour verification results, the performance of the proposed method is verified by using measures *Recall*, *Precision*, and a false-positive rate C_{FR}, defined as follows:

$$Recall = \frac{\text{detected events 1, 2, or 3}}{\text{total number of events 1, 2, or 3}}$$

$$Precision = \frac{\text{detected events 1, 2, or 3}}{\text{detected events 1, 2, or 3} + \text{false positives (per event)}}$$

$$C_{FR} = \frac{\text{false positives (per frame)}}{\text{total number of frames}}$$

Table 2. Comparisons of event localisation (object detection), computational costs, and available detection classes.

	Detection rate	Processing speed	Categories for verification
Active basis model	54%	0.51 fps	Vehicle
Deformable part model	94%	0.03 fps	Vehicle
Proposed method	91%	2.00 fps	Vehicle, motorcycle, and pedestrian

By using the proposed event verification framework, each moving object of interest showing a specific behaviour is detected, frame by frame. The performance is given in Table 2, and also illustrated in Fig. 5.

The entire algorithm is implemented in Matlab 2016a and CUDA in OS Windows 10 with 12 GB RAM and a GTX960M GPU processor. Processing is on average at 4.3 fps (Table 3).

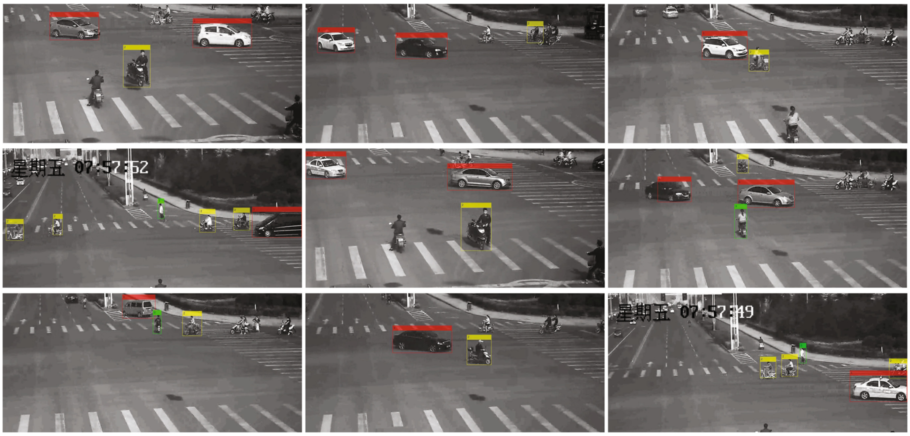

Fig. 5. Event verification at a road intersection. Three events are detected, recognized and labelled by a red rectangle (Event 1), a yellow rectangle (Event 2), and a green rectangle (Event 3). An object is detected when it moves; this generates an event to be classified; static objects are not labelled. (Color figure online)

Table 3. Performance of event verification in complex traffic scenes.

	Verification performance of Event 1		Verification performance of Event 2		Verification performance of Event 3		Comprehensive false alarm
	Recall	Precise	Recall	Precise	Recall	Precise	CFR
Dataset 1	91.6%	99.3%	95.5%	98.3%	82.9%	100.0%	1.8%
	(44/48)	(408/411)	(42/44)	(356/362)	(29/35)	(243/243)	(9/500)
Dataset 2	92.5%	99.9%	89.1%	99.7%	88.9%	84.8%	9.6%
	(99/107)	(1033/1034)	(115/129)	(1060/1063)	(24/27)	(228/272)	(48/500)
Dataset 3	88.2%	93.2%	80.00%	98.7%	-	-	10.8%
	(60/68)	(616/661)	(40/50)	(710/719)	-	-	(54/500)

6 Conclusions

This paper presents a novel integrated object-oriented event verification framework using a deep convolutional neural network. A new feature representation space is proposed to compress multi-frame and multi-object motion information into one colour image, which proved to be very helpful for integrated detection, tracking and behaviour understanding.

Considering the limited number of training samples from road-intersection traffic scenes, we initialised the neural network with a pre-trained ImageNet network. The experiments show that the proposed method outperforms previous work on multi-class object event localisation either in accuracy or in run-time. The accuracy of event classification has been improved as demonstrated for real data.

Acknowledgement. The experimental work was partially supported by Shandong Provincial Key Laboratory of Automotive Electronics and Technology, Institute of Automation, Shandong Academy of Sciences.

References

1. Bashir, F.I., Khokhar, A.A., Schonfeld, D.: Object trajectory-based activity classification and recognition using hidden Markov models. IEEE Trans. Image Process. **16**(7), 1912–1919 (2007)
2. Baccouche, M., Mamalet, F., Wolf, C., Garcia, C., Baskurt, A.: Action classification in soccer videos with long short-term memory recurrent neural networks. In: Diamantaras, K., Duch, W., Iliadis, L.S. (eds.) ICANN 2010. LNCS, vol. 6353, pp. 154–159. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15822-3_20
3. Dore, A., Regazzoni, C.: Interaction analysis with a Bayesian trajectory model. IEEE Trans. Intell. Syst. **16**(7), 1912–1919 (2007)
4. Felzenszwalb, P.F., Girshick, R.B., McAllester, D.: Cascade object detection with deformable part models. In: Proceedings of IEEE Conference on Computer Vision, Pattern Recognition, pp. 2241–2248 (2010)
5. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. **32**(9), 1627–1645 (2010)
6. Girshick, R.: Fast R-CNN. In: Proceedings of IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
7. Gupte, S.O., Masoud, O., Martin, R.F.K., Papanikolopoulos, N.P.: Detection and classification of vehicles. IEEE Trans. Intell. Transp. Syst. **3**(1), 37–47 (2002)
8. Hu, W., Tan, T., Wang, L., Maybank, S.: A survey on visual surveillance of object motion and behaviours. IEEE Trans. Syst. Man Cybern. Part C **34**(3), 334–352 (2004)
9. Hu, W., Xiao, X., Xie, D., Tan, T., Maybank, S.: Traffic accident prediction using 3D model-based vehicle tracking. IEEE Trans. Veh. Technol. **53**(3), 677–694 (2004)
10. Kamkar, S., Safabakhsh, R.: Vehicle detection, counting and classification in various conditions. IET Intel. Transp. Syst. **10**(6), 406–413 (2016)
11. Klette, R.: Concise Computer Vision. Springer, London (2014). https://doi.org/10.1007/978-1-4471-6320-6
12. Krizhevsky, A., Sutskever, I., and Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Proceedings of Advances Neural Information Processing Systems, pp. 1097–1105 (2012)
13. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)

14. Li, Y., Li, B., Tian, B., Yao, Q.: Vehicle detection based on the and-or graph for congested traffic conditions. IEEE Trans. Intell. Transp. Syst. **14**(2), 984–993 (2013)
15. Li, Y., Li, B., Tian, B., Yao, Q.: Vehicle detection based on the deformable hybrid image template. In: Proceedings of IEEE International Conference on Vehicular Electronics Safety, pp. 114–118 (2013)
16. Li, Y., Liu, W., Huang, Q.: Traffic anomaly detection based on image descriptor in videos. Multimedia Tools Appl. **75**(5), 2487–2505 (2016)
17. Niknejad, H.T., Takeuchi, A., Mita, S., McAllester, D.: On-road multivehicle tracking using deformable object model and particle filter with improved likelihood estimation. IEEE Trans. Intell. Transp. Syst. **12**(2), 748–758 (2012)
18. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: Proceedings of Advances Neural Information Processing Systems, pp. 91–99 (2015)
19. Sabokrou, M., Fayyaz, M., Fathy, M., Klette, R.: Deep-cascade: cascading 3D deep neural networks for fast anomaly detection and localization in crowded scenes. IEEE Trans. Image Process. **26**(4), 1992–2004 (2017). ieeexplore.ieee.org/document/7858798/
20. Uijlings, J.R.R., Van De Sande, K.E.A., Gevers, T., Smeulders, A.W.M.: Selective search for object recognition. Int. J. Comput. Vis. **104**(2), 154–171 (2013)
21. Viola, P., Jones, M.: Robust real-time face detection Int. J. Comput. Vis. **57**, 137–154 (2004)
22. Wu, Y.N., Si, Z., Gong, H., Zhu, S.-C.: Learning active basis model for object detection and recognition. Int. J. Comput. Vis. **90**(2), 198–235 (2010)
23. Xu, Y., Yu, G., Wu, X., Wang, Y., Ma, Y.: An enhanced Viola-Jones vehicle detection method from unmanned aerial vehicles imagery. IEEE Trans. Intell. Transp. Syst. **18**(7), 1845–1856 (2016). ieeexplore.ieee.org/document/7726065/
24. Zhang, Y., et al.: Vehicles detection in complex urban traffic scenes using Gaussian mixture model with confidence measurement. IET Intel. Transp. Syst. **10**(6), 445–452 (2016)