Background subtraction with convolutional neural network and oversegmentation

F. Morozov[1], A. Konushin[1,2]
fedor.morozov@graphics.cs.msu.ru|anton.konushin@graphics.cs.msu.ru
[1] Lomonosov Moscow State University, Moscow, Russia;
[2] NRU Higher School of Economics, Moscow, Russia

Background subtraction in video is an important computer vision problem as it serves as a first step in many video analysis algorithms, yet it cannot be considered to be fully solved due to complexity and variety of input data from surveillance cameras. In this work we propose a background subtraction method based on a random sample background model [14] using a convolutional neural network to compare image patches with the background model. To improve the quality of stopped objects detection we utilize oversegmentation of input frames, obtained superpixels are also used at postprocessing step. Experimental evaluation on ChangeDetection.net dataset shows superiority of the proposed algorithm compared to analogues.

Keywords: *background subtraction, convolutional neural networks, oversegmentation.*

1. Introduction

This work considers the problem of background subtraction in video, it is formulated as follows: each pixel of video sequence frame has to be classified as belonging either to background or to a moving object. In video analysis moving objects represent the greatest interest so they have to be separated from the mostly static background.

Background subtraction is often used as one of the first steps in video analysis algorithms based on object tracking and recognition. It can be used for detection of objects left behind and objects removed from the scene, people counting and crowd detection, tracking objects in security systems, control of parking spaces or traffic analysis such as congestion detection. In figure 1 an example of object left behind detection is presented with corresponding background subtraction mask.

Fig. 1. Object left behind detection with background subtraction mask.

Many of the mentioned problems use video from surveillance cameras as an input. The use of video analysis algorithms allows to automatically detect potentially dangerous situations and respond to them in time. It also substantially reduces the amount of works that has to be done by surveillance systems operators and improve the recall of detected incidents. That is especially important considering that the amount of video data to be processed is rapidly increasing.

Background subtraction algorithms have to be robust to sudden or gradual lighting changes, detect camouflaging objects and objects moving in shadows, while glares, shadows and reflections shouldn't cause false positive detections. Scenes that feature periodic background motion like waving leaves or water ripple or intermittent object motion are of particular complexity as are scenes with small camera displacements caused by wind, vibration or other reasons. Videos captured in difficult weather conditions like rain and snow or videos obtained at night with artificial illumination are also to be considered.

Result of state of the art algorithms are still inferior to manual labeling by experts due to the challenges described above so the problem of background subtraction cannot be considered to be fully solved.

Methods based on artificial neural networks are broadly used for various computer vision problems and show state of the art results in object detection, image classification, semantic segmentation, face recognition etc. Use of deep learning for background subtraction has not been widely researched yet while it may significantly improve existing algorithms or provide some novel approaches.

2. Related work

Most background subtraction algorithms rely on building a background model at each pixel. Then every pixel of a frame being processed is compared to the model and classified as background if it fits the model and as a moving object otherwise. Various algorithms differ in model construction, matching and updating approaches. They can be separated into the following groups based on background modeling:

1. Methods based on mixture of Gaussians [9]
2. Methods based on a set of samples [14]
3. Methods based on codewords [7].

Gaussian mixture models estimate the mean and variance of a number of Gaussians which also have weights indicating their persistence. If observed value matches a Gaussian, it's weight increases and parameters are updated using running average. If the sum of matched Gaussians' weights is above a given threshold, the pixel is classified as background.

Algorithm [15] uses Gaussian mixture modeling augmented by flux tensor motion detection which significantly improves the quality of stopped objects detection. Algorithm [2] performs matching in rectangular area around the pixel and random update strategy. In [5] it is proposed to use local binary patterns to leverage neighborhood information, algorithm [3] uses minimal spanning trees to average results of similar pixels.

Algorithms based on a set of values store a number of previously observed values for each point. One of the most popular models proposed in [14] uses random update policy and consensus based classification. It the observed value is similar to a given number of model values it is classified as background and may replace one of the stored elements. This way the probability that a sample will be presented in the model decreases monotonically with time.

Algorithm [1] proposes a number of heuristics improving the base model: connected components analysis, model update limitation, blinking pixel detection and adaptive thresholds. In [6] an adaptive model is proposed that tunes parameters of the algorithm at the runtime.

Algorithms [10] and [11] use a local feature descriptor – local binary similarity patterns. In [10] it is demonstrated that the use of area descriptor can drastically improve the performance of sample-based algorithm while [11] combines this idea with adaptive parameter tuning.

Models based on code words consist of complex elements – background words with some aggregate statistics like frequency, mean values, maximum negative run length etc. These statistics are used to estimate persistence of each background word similar to Gaussian mixture methods. Algorithm [16] uses a small number of code words obtained by clustering observed values using running average and efficacy counters. Algorithm [12] uses mixed code words with both color and texture information in the form of LBSP descriptors.

Several methods based on convolutional neural networks were proposed for segmentation and matching problems. In [17] different neural network architectures are compared for the task of local feature matching. In [18] a Siamese neural network is used to compare image patches for stereo matching. In [8] a similar approach was proposed for background subtraction with a trivial model, in this article we expand this idea with a more complex algorithm.

3. Proposed method

Proposed algorithm is based on background model proposed in [14]. This model is robust in presence of many typical challenges and has relatively small number of parameters that require tuning. It has been thoroughly studied in literature with many modifications and implementation details available. The background model does not use any form of clustering and allows almost voluntary feature representations.

The background model consists of a fixed number of samples at each pixel location. In [14] it is initialized using only the first frames which leads to errors if there are moving objects present at the moment of initialization. We propose to initialize the model using a number of first frames of video sequence and choose only persistent background values.

On each iteration a frame being processed is compared to the elements of the background model at corresponding point. Is its similar to a given number of elements according to the chosen metric, a background label is selected. Otherwise the pixel is classified as part of a moving object.

Since values of individual pixels are sensitive to noise and illumination changes we leverage information stored in a neighborhood around the pixels. To compare these image fragments a Siamese neural network [8] described in section 4 is used, that calculates descriptors for each patch and compares them using dot product.

Pixels classified as background are used to update the background model. Since the original values of patch pixels are not needed for matching only calculated descriptors can be stored for each pixel. Following [14] we use a random update policy: with a given probability a randomly selected background model element is discarded and replaced with the observed descriptor value.

For better tolerance to dynamic background and illumination changes a background pixel may replace a random element in one of neighboring background models with the same probability. This propagation is limited using superpixel boundaries as described in section 6. For postprocessing we use a median filter with kernel size equal to 5, the results are further corrected by filling homogenous superpixels.

4. Neural network descriptor

To compare image patches we use a Siamese neural network proposed in our previous work [8]. Its architecture is illustrated in figure 2: the network consists of two branches with shared weights, each branch has three convolutional layers with rectified linear activations and calculates a descriptor of length 16 for the input patch. To compare the patches a dot product is used, its result indicates a similarity score for the two patches.

The main advantage of the proposed architecture is that it allows to perform most of the computations (network branches) once for each fragment and each time we match two fragments a relatively simple dot product operation is only required. This is especially important for background subtraction, because multiple matching with background model elements is required for each pixel and descriptor values can be stored instead of the original patches.

To train the network mini-batch gradient descent is used with batch size of 128 and 0.001 learning rate. Pairs of negative and positive examples are used to train the network optimizing margin loss with margin value of 0.2.

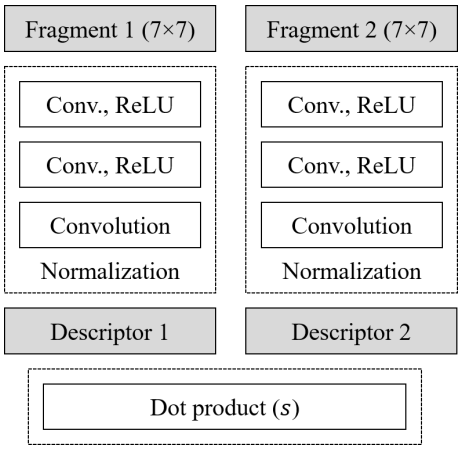

Fig. 2. Proposed neural network architecture.

5. Training the network

To train the network one needs pairs of positive and negative examples. We use a dataset of video sequences with ground truth labeling to generate pairs of patches. Pairs of background patches in the same point at different moments in time form the positive examples – assumingly similar patches, while pairs of background and a moving object patches in the same point form the negative ones, that the algorithm has to differentiate.

The number of background pixels is much greater than the number of moving objects' pixels so we construct as many negative examples as possible and generate the same amount of positive ones. So for every patch centered around a moving object pixel we need to pick two background patches around the same point. These fragments also shouldn't be too far in time since the background may change significantly over time, and may be centered around neighboring pixels to improve robustness in cases of dynamic background and camera displacement.

To generate the patches, it is proposed to use a sample-based background model. The use of background models allows to select background patches on the fly while random update makes the data more diverse and removes older samples, neighbor updates improve displacement tolerance.

6. Oversegmentation

Updating models in neighboring pixels allows the base background model to better cope with noise and illumination changes. On the other hand, this approach may lead to degradation in the presence of intermittent object motion: stopped objects are rapidly incorporated into the background models, detection errors appear on object boundaries.

One of the ways to deal with this problem is to limit neighbor updates on object boundaries as proposed in [1]. We propose to use oversegmentation to achieve this and use superpixel boundaries to limit the model elements propagation. For oversegmentation an algorithm [13] is used as it allows

real-time video processing with state of the art segmentation results.

The resulting superpixels are also used for postprocessing after median filtering: if a large enough share (0.96) of pixels grouped in a superpixel have similar classification results, the whole superpixel is classified with the prevailing value. An example of superpixel segmentation is presented in figure 3.

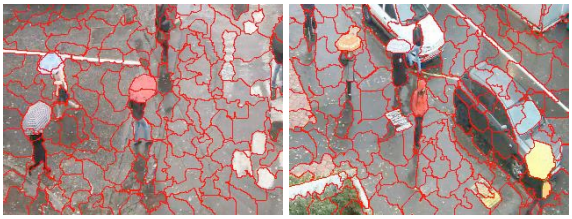

Fig. 3. Examples of oversegmentation.

7. Evaluation

To evaluate the proposed algorithm, we used ChangeDetection.net dataset [4]. This dataset consists of 31 video sequences with manual pixel perfect ground truth labeling split into five categories corresponding to typical background subtraction challenges (shadow, dynamic background, intermittent object motion, camera jitter, thermal) and a baseline category.

In table 1 are presented F-measure values calculated for detection results by the proposed method, base algorithm [14] and two its modifications: [10] that uses local binary similarity patterns and [6] that proposes adaptive parameter tuning. Proposed method outperforms analogues in all the dataset categories and in average.

We used OpenCV library for algorithm implementation and caffe framework for neural network training. Performance of the prototype version is about 6 frames per second for 320×240 resolution.

	ViBE [14]	LOBSTER [10]	PBAS [6]	Proposed
Baseline	0,87	0,92	0,92	**0,94**
Shadow	0,80	0,87	0,68	**0,90**
Dynamic background	0,57	0,57	0,72	**0,78**
Intermittent object motion	0,51	0,58	0,57	**0,63**
Camera jitter	0,60	0,74	0,81	**0,82**
Thermal	0,66	0,82	0,76	**0,84**
Average	0,67	0,75	0,75	**0,82**

Table 1. F-measure values on ChangeDetection.net dataset.

8. Conclusion

In this work a new background subtraction method is proposed based on sample background model with random update [14]. Neural network descriptor is used to compare patches on the frame being processed with the background model representations. Oversegmentation is used at model update and postprocessing steps to improve quality at object boundaries and fill in erroneous regions.

The proposed modifications of the base algorithm provide superiority compared to the base algorithm and its modifications in terms of f-measure as demonstrated on ChangeDetection.net. Examples of background subtraction with the proposed algorithm are presented in figure 4.

In the future we are planning to develop a more optimized implementation of the algorithm and further improve its precision and recall with data augmentation and adaptive parameter tuning.

Fig. 4. Examples of background subtraction using proposed algorithm.

9. Acknowledgements

This work was supported by RFBR grant no. 14-01-00849 and by the Skolkovo Institute of Science and Technology, the contract 081-R, Annex A2.

10. References

[1] Barnich O., Van Droogenbroeck M. Background subtraction: Experiments and improvements for ViBe.
[2] Chen Y., Wang J., Lu H. Learning sharable models for robust background subtraction.
[3] Chen M., Yang Q., Li Q., Wang G., Yang, M. Spatiotemporal background subtraction using minimum spanning tree and optical flow.
[4] Goyette N. Changedetection. net: A new change detection benchmark dataset.
[5] Heikkilä M., Pietikäinen M., Heikkilä J. A texture-based method for detecting moving objects.
[6] Hofmann M., Tiefenbacher P., Rigoll G. Background segmentation with feedback: The pixel-based adaptive segmenter.
[7] Kim K. Real-time foreground–background segmentation using codebook model.
[8] Morozov F., Konushin A. Background subtraction using a convolutional neural network
[9] Stauffer C., Grimson W. Adaptive background mixture models for real-time tracking.
[10] St-Charles P., Bilodeau G. Improving background subtraction using local binary similarity patterns.
[11] St-Charles P. L., Bilodeau G. A., Bergevin R. Flexible background subtraction with self-balanced local sensitivity.
[12] St-Charles P. L., Bilodeau G. A., Bergevin R. A self-adjusting approach to change detection based on background word consensus.
[13] Van den Bergh M. et al. SEEDS: Superpixels extracted via energy-driven sampling.
[14] Van Droogenbroeck M., Paquot O. ViBe: a powerful random technique to estimate the background in video sequences.
[15] Wang R. Static and moving object detection using flux tensor with split gaussian models.
[16] Wang B., Dudek P. A fast self-tuning background subtraction algorithm.
[17] Zagoruyko S., Komodakis N. Learning to compare image patches via convolutional neural networks.
[18] Zbontar J., LeCun Y. Computing the stereo matching cost with a convolutional neural network.

About the authors

Fedor Morozov is a Master student at Lomonosov Moscow State University, Department of Computational Mathematics and Cybernetics. His contact email is fedor.morozov@graphics.cs.msu.ru.

Anton Konushin is an associate professor at Lomonosov Moscow State University and at National Research University Higher School of Economics. His contact email is anton.konushin@graphics.cs.msu.ru.