Background Image Estimation with MRF and DBSCAN Algorithms

Pavel V. Skribtsov*, Sergey O. Surikov and Michael A. Yakovlev
PAWLIN Technologies Ltd, Dubna, Russian Federation; info@pawlin.ru

Abstract
New simple and fast method for robust background image estimation from video stream is offered. The method is based on Markov Random Field, Belief Propagation and DBSCAN algorithms. The usage of cyclic graph MRF model of non-overlapping image patches with narrow set of background hypotheses generated by DBSCAN algorithm provides faster convergence of Belief Propagation with accuracy comparable to the state-of-art methods.

Keywords: Background Estimation, Belief Propagation, DBSCAN, Markov Random Field

1. Introduction and Previous Work
Static background image estimation is important step for many automatic video analysis algorithms, for example, vehicle or pedestrians tracking. Ability to get a mask of foreground object significantly reduces the amount of computation required to run the complex object detectors per frame, taking into account that some background structures may cause continuous false detects present almost at each frame of the video. Foreground mask is also useful for limiting the directions of where the moving object could go that can be taken during the object tracking. The first essential step of foreground computation is estimation of the background image. Once the background image is built a number of approaches can be applied to compute the mask of foreground objects well described in a recent review. So this paper is focused only on the background image estimation step.

One of the possible easy ways of computing background image model is to collect pixel-wise statistics as it is done in MoG algorithm where pixel-wise Gaussian Mixture is used. This algorithm is implemented in OpenCV library. However such approach, being applied in the online manner has tendency to assign moving objects to background if they are still for a while. Moreover simple pixel wise statistics models do not work at such locations where foreground objects appear more frequently in video than background is revealed. Therefore a class of more complex algorithms is developed that try to find places in video sequence that correspond to background even if they were rarely exposed and most of the time were covered by foreground objects. They divide in two major groups – pixel-wise methods and patch-wise method. An example of the pixel-wise method is the work of that is based on the loopy belief propagation; however authors underline the speed concerns of their approach as it takes minutes to just process several manually picked frames from the video sequence. Other methods use image patches for spatial support as an additional heuristic in order to overcome some problems of the pixel-wised approaches. These methods can be based on overlapping patches and non-overlapping patches as the recent work of suggest its lower complexity. The contribution of this paper is further diminishing of the complexity of the background image estimation algorithm, providing faster convergence speed at the same or better quality of the result.

The work of was taken as the base for the comparison as it proves superior quality with respect to the low-complexity (fast) methods and the authors were nice to reveal C++ source code of the implementation that was successfully compiled and run. The differences

* Author for correspondence
of the proposed approach is in the following points:
1. “Uniformity” of the algorithm consisting purely of Belief Propagation iterations followed by the step of initial hypotheses collection and absence of the extra step of preliminary background computation 2. The MRF network is global, not local that allows information flow and support from distant locations of the image 3. DBSCAN clustering technique was deployed as it was found to be important component of the hypotheses collection process.

2. Proposed Algorithm

The outline of the algorithm is very similar to the Belief Propagation framework suggested for example-based super-resolution method suggested by therefore mathematical foundation of the approach is not repeated here. The key differences from are:

- Hypotheses (states) of the markov network nodes are chosen as centroids of a clustering algorithm applied to time-varying image patches picked from video stream at fixed locations.
- A-priory probability of the state in our problem is determined by the frequency of the hypotheses patch appearance in video.
- Compatibility matrix for each node and one of its 8-connected neighbor is determined by the edge matching metric.
- Belief propagation iterations (messages exchange process) are continued until either convergence (no belief change) is reached or the maximum number of iterations is exceeded (in our case – 50 at most, with convergence likely happening between 10th and 20th iterations).

The flowchart of the algorithm is given in Figure 1. First, the video to be processed is divided into time-varying patches using a fixed quadratic grid of blocks and the list of hypotheses is constructed (Step 1). An initial full list of hypotheses for each cell is considered representing the content of the corresponding video frame block on each frame. The patch hypotheses are filtered by the condition that a patch must be still for at least small number of frames (Step 2). The patch is considered to be still by an SAD metric with a small threshold dependent on the noise of the decompressed video content. If the patch is unique within a local time period, it is rejected. It is important step to reach high performance characteristics of the proposed method. For each of the time-varying sequence of patches the clustering method is applied (Step 3). The states of the Markov Network nodes are assigned to be the centroids of the patch-clusters. In our case we experimented with standard K-means and DBSCAN methods. Other clustering methods can be considered in future research without limitations. Then the graph of the markov network is created (Step 4) that is a standard static 8-connected graph of cell nodes, where each node corresponds to the block of the image grid. Side blocks obviously have fewer neighbors. Step 5 calculates compatibility matrices for adjacent nodes’ states based on the sum of L1-norm of the edge intensity difference vector and edge gradient difference vector. For diagonal neighbors the “edge” is considered to be a single corner pixel, therefore gradient component is not computed for side neighbors. The next step is computation of a-priory belief in markov network states which is proportional to the size of the cluster in the hypothesis space, which is equivalent to the frequency of the patch appearances in...
the video (Step 6). If the dependency of the frequency of appearance of the patch is not desired, all measurement beliefs of the states can be set to 1. If global iterations are enabled, the states that received low belief in the end of the previous global iteration can be discarded (Step 6b). Then belief propagation message passing iterations are run (Step 7) and the states having the best belief are selected after convergence (Step 8). Final resulting image is assembled from winner state patches at Step 9.

3. Practical Aspects of the Implementation

Some extra actions are omitted above to focus on the main aspects, nevertheless they worth mentioning.

- If the frame can’t be divided by cells without remainder, the resolution of the video is upscaled and in the end the remaining background image estimation is downscaled to the initial resolution.
- As the proposed method uses non-overlapping patch system, a natural edge in the background image which passes through the edge of a cell may cause true hypotheses to be rejected by belief propagation due to the low values of the corresponding compatibility matrix elements. In order to fix this issue, the frames were Gaussian-blurred with a 3x3 kernel to exchange border pixels information in adjacent patches. Non-blurred versions of patches are also maintained with a table of respective links. Later on the final assembly step this table of links is used to find the non-blurred winner patch in the cluster.
- Global iterations were found to be bringing too little improvement compared to the increase in overall computation time; therefore they were not used in experiments showed below.
- DCT-based (Discrete Cosine Transform\textsuperscript{12}) metric has been evaluated for computing compatibility matrix as in\textsuperscript{8}. However this approach did not give good results in our case. We assume this was due to the reason that in the work of\textsuperscript{8} – the test patch (“candidate label”) was evaluated with the 3 other fixed patches, so the test region constituted only 25% of the local image area where DCT was computed. Therefore the structure of the DCT coefficients was mostly defined by the 75% of the fixed part of the input image for DCT. In our case we compared DCT coefficients where test patch constituted 50% of the overall input image and the edge compatibility influenced spectral distribution in smaller amount than the content of the test image itself, so the simple intensity and gradient based edge compatibility metric worked much better (and way faster) in our case.
- DBSCAN provided much better results than K-means based on qualitative analysis. We assume this happens because of the resistance of DBSCAN to outliers and that it joins elements in cluster with more consistent “distances” between the elements. Examples of the background model built with two clustering methods are shown in Figure 2. Artifacts of K-means method are clearly seen.

4. Experiments

Overall the implementation of the algorithm showed fast convergence and good quality. The experiments reported in this paper were carried out using 4 publically available video sequences from CAVIAR project\textsuperscript{13} and 2 private video sequences. The video sequences “OneStopEnter1cor”, “Browse1”, “Browse4” and “ThreePastShop1front” are defined at the corresponding web pages of CAVIAR project. To give definition of the private videos sequences “Video 1” and “Video 2”, their typical frames are shown in Figure 3.

Qualitative results of the algorithms performance are depicted in Figure 5.

Table 1 shows the numerical comparison results. In all cases the proposed method works faster and slightly better than the nearest analog\textsuperscript{8}. Interestingly that due to the filtering (Step 2) of the proposed method in some cases it is possible to achieve substantial speedup (almost x7 times on the sequence #5).
Figure 3. Sample frames from the private video dataset “Video 1”. The video sequence contains multiple moving and quasi-static foreground objects, e.g. standing bicycle and people at the desk in right bottom corner that typically represent serious difficulties to simpler background estimators, e.g. MOG.

Figure 4. Sample frames from the private video dataset “Video 2” with moving and quasi-static foreground objects.
Table 1. Quantitative comparison

| Video sequence # (in the order of appearance in Fig. 5) | 1     | 2     | 3     | 4     | 5     | 6     |
|--------------------------------------------------------|-------|-------|-------|-------|-------|-------|
| Reddy et al. (R), Proposed Method (P)                  | R     | P     | R     | P     | R     | P     |
| Mean absolute pixel difference to Ground Truth if available, millions of grayscale levels (0..1) | 71    | 61    | 120   | 91    | 100   | 82    | 81    | 73    | N/A   | N/A   | N/A   | N/A   |
| No visible artifacts                                   | +     | +     | +     | +     | +     | +     | +     | -     | +     | -     | +     |
| Processing time, sec (Lenovo Laptop YA470, Intel Core i7, 2.6 Ghz, single core, no GPU used) | 72    | 19    | 45    | 15    | 45    | 16    | 60    | 25    | 1712  | 255   | 272   | 121   |
| FPS                                                    | 20.8  | 78.9  | 23.2  | 69.5  | 25.3  | 71.2  | 27.5  | 65.96 | 5.7   | 38.4  | 18.4  | 41.3  |

Figure 5. Qualitative results of the proposed method vs. Reddy et al approach. The results on the publically available datasets are almost indistinguishable. One can tell the different algorithms were applied by only analyzing the timestamp mosaic in the upper part of the image. The results on the private video sequences show considerable artifacts in case of Reddy et al approach at locations where foreground objects were standing still for longer period of time.
5. Discussion

The proposed algorithm is much simpler in code and faster than its nearest analog. At the same time it provides comparable or slightly better quality of the resulting background image estimation. In spite of that there is ready, open source C++ implementation of the background estimation algorithm available for download the efforts of developing our own implementation in this case happened to be justified as we got faster and simpler solution. We were also able to fully re-use existing standard belief propagation algorithm C++ implementation without any modifications as the core engine of the program; therefore it took less than 1 month to develop the software implementation for the proposed method.

6. Conclusion and Future Work

Novel background image estimation approach based on belief propagation algorithm was introduced that showed improvement on publically available benchmark dataset (CAVIAR) as well as in the private test-videos used in the project. Future efforts should be aimed at further increasing the speed of processing using GPU as Belief Propagation is highly parallelizable. We also see potential in exploring not only the winner states of the markov network nodes for more complex background model and apply more advanced foreground mask computation schemas that may make use of the second-order states as they may also represent the background in shadow or illumination change conditions.

7. Acknowledgements

The research was financed by the Ministry of Education and Science of the Russian Federation under the Grant Contract dated 24 November, 2014, (grant number 14.579.21.0070, contract unique identifier RFMEFI57914X0070), the grant was assigned for the research into “2D- and 3D-data fusion and processing software for object detection and recognition”.

8. References

1. Qu Z, Yu M, Liu J. Real-time traffic vehicle tracking based on improved MoG background extraction and motion segmentation. Proceedings of 3rd International Symposium on Systems and Control in Aeronautics and Astronautics (ISSCAA); Harbin. 2010 Jun 8-10. p. 676–80. Available from: 10.1109/ ISSCAA.2010.5633717.
2. Badie J, Bak S, Serban T-S, Bremon F. Recovering people tracking errors using enhanced covariance-based signatures. 14th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance; Beijing. China. 2012 Jul. p. 487–93.
3. Sobral A, Vacavant A. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. Computer Vision and Image Understanding. 2014; 122:4–21.
4. KaewTraKulPong P, Bowden R. An improved adaptive background mixture model for real-time tracking with shadow detection. Proceedings of 2nd European Workshop on Advanced Video Based Surveillance Systems, AVBS01; 2001 Sep.
5. Open Library (Open Source Computer Vision). Available from: www.opencv.org
6. Xu X, Huang TS. A loopy belief propagation approach for robust background estimation. IEEE Conference on Computer Vision and Pattern Recognition; Anchorage, AK. 2008 Jun 23-28. p. 1–7. Available from: 10.1109/CVPR.2008.4587543
7. Colombari A, Fusiello A, Murino V. Background initialization in cluttered sequences. IEEE Computer Society Proceedings of 5th Workshop on POCV (in conjunction with CVPR); New York. 2006 Jun 22. p. 197–202.
8. Reddy V, Sanderson C, Lovell BC. A low-complexity algorithm for static background estimation from cluttered image sequences in surveillance contexts. EURASIP Journal on Image and Video Processing. 2011. p. 1–14.. Available from: http://dx.doi.org/10.1155/2011/164956
9. Yedidia JS, Freeman WT, Y. Understanding belief propagation and its generalizations. In: Lakemeyer G, Nebel B, Editors. Exploring Artificial Intelligence in the New Millennium. Morgan Kaufmann; 2003 Jan. p. 239–6. ISBN: 1-55860-811-7.
10. Freeman WT, Jones TR, Pasztor EC. Example-based super-resolution. IEEE Computer Graphics and Applications. 2002; 22(2):56–65.
11. Ester M, Kriegel HP, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96); 1996. p. 226–31.
12. Ahmed N, Natarajan T, Rao K. Discrete Cosine Transform. Transactions on Computers. 1974; 100(23):90–3.
13. Caviar dataset. Available from: http://homepages.inf.ed.ac.uk/rbf/CAVIAR/
14. Brunton A, Shu Ch, Roth G. Belief propagation on the GPU for stereo vision. IEEE Proceedings of the 3rd Canadian Conference on Computer and Robot Vision; 2006 Jun 7-9. Available from: 10.1109/CRV.2006.19
15. GA Jones, N Paragios, CS Regazzoni. Video Based Surveillance Systems: Computer Vision and Distributed Processing. Kluwer Academic Publishers Springer; 2002. p. 135–44.