A Generic Feature Selection Method for Background Subtraction Using Global Foreground Models

Marc Braham and Marc Van Droogenbroeck

INTELSIG Laboratory, Department of Electrical Engineering and Computer Science,
University of Liège, Belgium
{m.braham, M.VanDroogenbroeck}@ulg.ac.be

Abstract. Over the last few years, a wide variety of background subtraction algorithms have been proposed for the detection of moving objects in videos acquired with a static camera. While much effort have been devoted to the development of robust background models, the automatic spatial selection of useful features for representing the background has been neglected. In this paper, we propose a generic and tractable feature selection method. Interesting contributions of this work are the proposal of a selection process coherent with the segmentation process and the exploitation of global foreground models in the selection strategy. Experiments conducted on the ViBe algorithm show that our feature selection technique improves the segmentation results.

Keywords: background subtraction, feature selection, foreground modeling, change detection, ViBe.

1 Introduction

Detecting moving objects in video sequences provides a valuable information for various applications such as video coding [1], depth extraction from video [2], or intelligent vision systems [3]. A straightforward approach for motion detection is background subtraction, typically used in tracking systems, for its ability to detect moving objects without any assumption about their appearance, size or orientation [4,5]. The process consists in building a model of the static scene, which is named *background*, and updating this model over time to account for luminance and structural changes in the scene. The background model is subtracted from the current frame, and pixels or regions with a noticeable difference are assumed to belong to moving objects (the *foreground*). A background subtractor is thus a two-class classifier (foreground or background).

While conceptually simple, background subtraction still remains a difficult task because of the variety of challenging situations that occur in real world scenes. For instance, pixels may be erroneously classified as foreground in the presence of camera jitter, dynamic background (such as swaying trees or sea waves) or illumination changes. On the other hand, camouflage (*i.e.* foreground and background sharing similar colors) generate many false negative detections.

To deal with these challenging situations, a plethora of background subtraction techniques have been proposed (see [6] or [7] for a recent and comprehensive classification of these techniques including more than 300 references). They differ, among other things, in the features chosen to build the background model. According to Bouwmans [7], five categories of features are frequently used: color features, edge features, texture features, motion features, and stereo features. Color features refer mainly to the color components of color spaces which can be processed separately or jointly. The RGB domain is the most popular [8,9], but some authors exploit other color spaces such as HSV [10] or YCbCr [11] in order to increase invariance with respect to brightness changes, and thus illumination changes and shadows. Gradient features [12] and texture features such as LBP histogram [13] or its variants [14,15] offer a robust solution to illumination changes but might be unsuitable in image areas with poor texture. Motion features, such as optical flow [16], should be particularly interesting for scenes where the foreground is moving continuously (absence of temporarily stopped objects) and depth features acquired with range cameras such as the Kinect camera [17] are (to some degree) insensitive to lighting conditions but cannot be computed when objects are far from the camera.

In conclusion, there seems to be no agreement on a unique feature performing better than any other feature independently of the background and foreground properties. Surprisingly, this is not considered in practice. Almost all existing background subtraction techniques operate with a uniform feature map in the sense that the features used to model the background and perform the detection are the same for all pixels of the image, thus ignoring the non-uniformity of the spatial distribution of background properties and neglecting the foreground properties. Recently, Bouwmans concluded in a comprehensive survey paper [7] that feature selection in background subtraction is still an open problem and may be one of the main future developments in this field.

In this paper, we propose a generic method for the spatial selection of the most significant features and we assess it for the ViBe technique. The main novelties of our work are: (1) a dedicated feature selection process that is coherent with the background subtraction process, and (2) the use of global foreground models in the selection strategy. The rest of the paper is organized as follows. Section 2 reviews the related work and presents the main limitations of existing selection methods. We provide the details of our method in Section 3. Experiments showing the benefits of our method are described in Section 4. Finally, Section 5 concludes the paper.

2 Related work

Li *et al.* [18] were among the first to express the need for modeling distinct part of the image with different features. They describe the background image as consisting of two pixel categories, static pixels and dynamic pixels, and incorporate this model in a Bayesian framework. Static pixels belong to stationary objects such as walls or room furniture whereas dynamic pixels refer to non stationary objects such as swaying trees or sea waves. Color and gradient statistics are used to describe static pixels while color co-occurrence statistics describe dynamic pixels. As the correspondence between a pixel category and the features used to describe it is fixed, the selection strategy reduces to an identification process of static and dynamic pixels. A temporal difference between two consecutive video frames is used for this purpose. Unfortunately, this method lacks of generality as the cardinality of the feature set, *i.e.* the number of candidate features or group of features, is limited to two (one group of features for static pixels and one other for dynamic pixels).

Parag *et al.* [19] introduced a more general feature selection framework requiring a set of training images free of foreground objects. The set of training images is divided into two parts. The first one generates a background statistical model for each feature of the feature set. These statistical models are used for the computation of background probability estimates (for each candidate feature) with the Kernel Density Estimation (KDE) [8] on the remaining part of the training set and on synthetic foreground examples. The computed estimates feed the training of a RealBoost classifier [20]. The resulting classifier selects the most useful features for each pixel and assigns a confidence weight to each of them. While this framework can be used with an arbitrary number of candidate features, it has a serious drawback. Indeed, the synthetic foreground examples used for boosting are generated randomly from a uniform distribution. In other words, all candidate features are assumed to be uniformly distributed for foreground objects. This assumption is not valid because of the wide variety of foreground statistical distributions among different features. For instance, gradient has a foreground probability density function (FG-pdf) concentrated around low values while RGB color components have a wider FG-pdf. This may explain why the classification performance of Parag's framework [19] decreases when gradient features are added to the feature set. In our work, we overcome this major limitation by means of global statistical foreground models.

Another boosting-based approach for feature selection was proposed by Grabner *et al.* [21,22]. The main novelty of their work is the spatio-temporal nature of the selection through on-line boosting. However, since the method is based on self-learning (*i.e.* classifier predictions feed model updates), the background model can end up in catastrophic state, as mentioned in [23]. Once again, unrealistic assumptions about the statistical distributions of foreground features are used (a uniform distribution is assumed for the gray value of foreground objects and serves as a basis for computing other feature distributions).

Recently, Javed *et al.* [24] included a dynamic feature selection strategy for background subtraction in an *Online Robust Principal Component Analysis* (OR-PCA) framework. Feature statistics, in terms of means and variances, are used as a selection criterion. Unlike aforementioned purely spatial or spatio-temporal selection approaches, their method is exclusively temporal-based, which means that all pixels use the same features for the foreground segmentation. The non-uniformity of the spatial distribution of background properties is thus ignored.

3 Proposed selection strategy

A schematic view of our feature selection strategy is given in Fig. 1. Like in Parag's work [19], our selection process occurs during a training phase, which allows to avoid extra computations during normal background subtraction operations. This training phase is divided into three parts, represented by boxes [1], [2], and [3] in Fig. 1. The first part consists to accumulate images free of foreground objects, typically a few hundreds at a framerate of 30 frames per second, which are further processed to build local background statistical models. The second training part requires another sequence of images, this time including foreground objects. These images are not saved into memory, but processed by a background subtraction algorithm in order to build a global foreground statistical model for each candidate feature of the feature set. The third part of the training phase selects the best feature/threshold combination; this process is *local*, meaning that it is performed for each pixel individually. The goal of the selection process consists to detect which feature is most appropriate to discriminate between the local background and the foreground at the frame level. To guarantee some consistency with the segmentation process, the selection step is fed by supplementary information relating to the background model of the segmentation algorithm, and to the application performance metric (see box [4] in Fig. 1). All these steps are detailed in the following subsections.

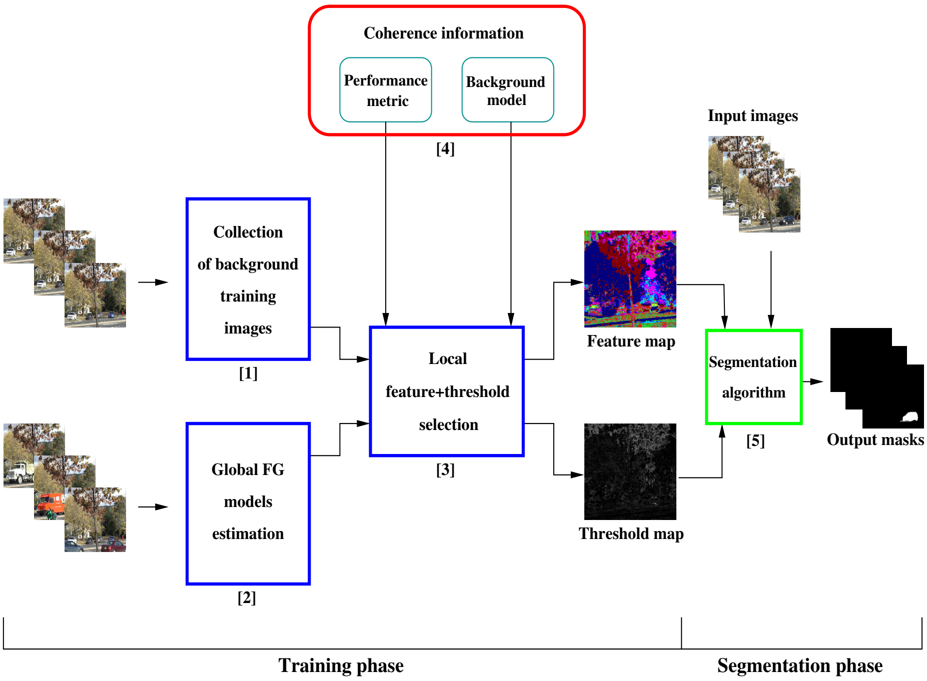

Fig. 1. Schematic representation of the proposed feature selection framework.

3.1 Estimation of the foreground: local vs global foreground models

Background subtraction is a classification problem between the background or foreground classes. It is well known that the classification performance of a classifier is dependent on the discriminating power of the used features. Therefore, an optimal classifier should select the best feature locally, that is for each pixel separately, to maximize its capability to distinguish between the local background values and the local foreground values, which indirectly implies that we are able to estimate the corresponding statistical distributions for any feature.

Obtaining reliable local background statistical models is realistic and common. Indeed, local background feature values are relatively stable (or at least predictable). Moreover, the pixel-wise probability to observe a background sample (we name this as the local BG-prior) is much higher than the local FG-prior. Subsequently, it suffices to collect a small set of background training images to estimate the local BG feature distributions. However, estimating local foreground distributions is more complex. The two main difficulties are that (1) most features have a wide FG-pdf, and (2) local FG-priors are often low with respect to BG-priors, which means that foreground might not be encountered locally during the training phase. In fact, estimating the local foreground distributions reliably would require an extremely long observation period, which is clearly intractable. There is thus an imbalance between the estimation of local background or foreground characteristics.

To overcome this estimation problem, we assume that the *local* foreground distribution of a feature can be approximated by its *global* foreground distribution. There is an important advantage to assume this. Indeed, it might be that a moving object is very different from the background for some areas of the image, but close for other areas. A global estimation will help to learn from the foreground values detected in the easiest areas of the scene. Mathematically, assuming that the distributions are represented by histograms of observed values, and denoting by $\mathbf{LH}_{\mathrm{f}}$ the matrix of local histograms for a feature f, the *global* histogram $\mathbf{GH}_{\mathrm{f}}$ of feature f is given by:

$$\mathbf{GH}_{\mathrm{f}}(b) = \sum_{(x,y)} \mathbf{LH}_{\mathrm{f}}(x, y, b) \ , \tag{1}$$

where b is a particular bin and (x, y) refers to the coordinates in the image. For a multi-dimensional feature (for instance the RGB feature which is a tridimensional feature, to the contrary of the individual color components: R,G, or B), equation (1) becomes:

$$\mathbf{GH}_{\mathrm{f}}(b_1, b_2, ..., b_D) = \sum_{(x,y)} \mathbf{LH}_{\mathrm{f}}(x, y, b_1, b_2, ..., b_D) \ , \tag{2}$$

where D is the number of dimensions of the feature. Note that our assumption of a uniform spatial distribution of foreground feature values is not perfectly true, especially because of the modification of geometrical and colorimetric properties of foreground objects in the image as they move across the scene. Despite that, we believe that the approach is appropriate. The main benefit results from the considerable reduction of the observation time required for estimating the local foreground distributions. A practical estimation becomes thus feasible.

In summary, the second training stage (named "Global FG models estimation" in Fig. 1) is performed as follows. We feed a background subtraction algorithm (named "estimator" in the remainder of the paper) with a sequence of training images including foreground objects. The choice of the estimator is not critical; state-of-the-art techniques, such as SuBSENSE [25] or PAWCS [26], are good candidates to segment foreground objects. Note that a fast estimator is not necessary because there is no real-time requirement during the training phase. The segmentation masks of the estimator are processed to build the $\#_\mathrm{f}$ global foreground models, $\#_\mathrm{f}$ denoting the cardinality of the feature set.

3.2 Coherence between selection, segmentation, and application

While some authors have expressed the need for modeling distinct part of the image with different features for background subtraction [18,19,21], none of them noticed that the best feature map may depend on elements unrelated to background or foreground statistical properties. We innovate in this paper by considering the type of background model used by the segmentation algorithm (see box [5] in Fig. 1). Some features may be adapted to specific background models, while performing badly for others. This statement was confirmed recently by López-Rubio *et al.* [27]. They showed that, while Haar-like features [28] were reported to be adapted to KDE background models in the context of feature combination [29], they have an insufficient discriminating power in a probabilistic mixture-based approach. A generic algorithm should therefore select, locally, the most discriminating feature *for each background model individually.*

Another important element of our framework is the inclusion of coherence with respect to the application. Some applications require to minimize the false alarm rate while maintaining a desired recall, whereas others need to maximize the recall while maintaining a relatively low false alarm rate. As each feature choice tends to favor particular metrics, the performance metric of the application should be considered during the selection process. Therefore, we use the application performance metric to evaluate the capability of a feature to discriminate between local background samples and global foreground samples. Note however that the framework is incompatible with prior-dependent performance metrics, such as the popular F_1 measure. The origin of this limitation is the modification of the local FG-prior due to the construction of the global foreground histograms. Indeed, the number of global foreground samples collected during the second step of the learning phase (see box [2] in Fig. 1) is much higher than the number of local background samples of a pixel (in fact equal to the number of background training images). Therefore, as the selection process introduces a large bias in the estimation of the local FG-prior, it is strongly recommended to use prior-independent metrics, such as the geometrical mean [30] or the Euclidean distance in ROC space introduced in [31].

3.3 Local feature/threshold selection process

Our generic feature selection method for background subtraction is presented in Algorithm 1. This algorithm is implemented in the [3] box of Fig. 1. Note that the background subtraction step involves the choice of matching thresholds (one threshold per feature) for comparing current input feature values with local background feature values. For instance, in the case of ViBe [9], the radius of the color sphere is the matching threshold used to compare RGB feature values. These thresholds affect the results of the corresponding features. For this reason, we feed the generic selection algorithm with $\#_\mathrm{f}$ sets of $\#_\mathrm{t}$ candidate thresholds (one set of candidate thresholds per feature), where $\#_\mathrm{t}$ denotes the cardinality of the threshold sets (assumed equal for each threshold set, for convenience) and let the algorithm select the best feature/threshold combination, locally.

The selection process consists in a simulation of the target background model for each feature. Firstly, the simulated local feature background models are build. In the case of ViBe's background model [9], this is performed by selecting, randomly and for each feature, 20 values among the K background feature values available at coordinates (x, y), K denoting the number of background training images. Then, each model is assessed for its capability to predict the correct class of input samples, these one being both (1) all local samples of the background training images, and (2) all global samples collected in the corresponding global FG histogram. As the predictions depend on the matching threshold, this capability is assessed for all candidate thresholds of a given feature, and described by a confusion matrix [32]. The performance metric is computed from the confusion matrix and the best feature/threshold combination is selected.

4 Experiments

4.1 Description of the methodology

We particularized our generic method for the ViBe background subtraction algorithm [9], and evaluated it for the CDnet 2012 dataset [33]. The feature set contains 9 individual color components ($\#_\mathrm{f} = 9$), taken from the 3 common color spaces RGB, HSV, and YCbCr:

$$CF = \{R, G, B, H, S, V, Y, Cb, Cr\},\ \#_\mathrm{f} = 9\ . \tag{3}$$

Each feature is quantized into 256 bins. For this purpose, the saturation S and the value V (resp. the hue H) are rescaled linearly from the range $[0, 1]$ (resp. $[0, 360)$) to the range $[0, 255]$. The pre-defined set of possible thresholds is

$$CT^f = \{2, ..., 10, 12, 14, 16, 18, 20, 25, 30, 35, 40, 45, 50\}\ \forall f,\ \#_\mathrm{t} = 20\ . \tag{4}$$

The Euclidean distance in the ROC space, introduced in [31] and defined as

$$d_{euc} = \sqrt{FNR^2 + FPR^2}, \tag{5}$$

Algorithm 1 Generic algorithm for local feature/threshold selection.

```
Inputs
(1) A set of $\#_{\mathrm{f}}$ candidate features CF = $\{F_1,F_2,...,F_{\#_{\mathrm{f}}}\}$
(2) $\#_{\mathrm{f}}$ sets of $\#_{\mathrm{t}}$ candidate thresholds $CT^f = \{T_1^f,T_2^f,...,T_{\#_{\mathrm{t}}}^f\}$
(3) A performance metric P
(4) A target background model BM
(5) A table of K background training images BTI where BTI(x,y,k) is the background
    RGB value at pixel coordinates (x,y) and training image k
(6) A table FGGH where FGGH(f,b) is the global FG histogram value
    for bin b of feature $F_f$ (see Section 3.1 for more details)
Outputs
(1) A feature map $\mathbf{F}_{map}$ with $\mathbf{F}_{map}(x,y) \in$ CF, ∀(x,y)
(2) A threshold map $\mathbf{T}_{map}$ with $\mathbf{T}_{map}(x,y) \in CT^{F_{map}(x,y)}$, ∀(x,y)
Internal data
(1) $\#_{\mathrm{f}}$ tables $\mathbf{Class}_f$ where $\mathbf{Class}_f$(b) is a class assigned to bin b of feature $F_f$
(2) A confusion matrix CM
(3) A table PT where PT(f,t) is a performance assigned to the combination $F_f/T_t^f$
(4) $\#_{\mathrm{f}}$ feature background models $FBM_f$
Pseudo-code of the selection algorithm
for each (x,y) do
    for each $F_f \in$ CF do
        Build the feature background model $FBM_f$ using BM and BTI(x,y,:) ;
        for each $T_t^f \in CT^f$ do
            for each bin b of feature $F_f$ do
                Predict $\mathbf{Class}_f$(b) given $FBM_f$ and $T_t^f$ ;
            end
            Compute CM given $\mathbf{Class}_f$(:), BTI(x,y,:), and FGGH(f,:) ;
            Compute PT(f,t) given CM and P ;
        end
    end
    Find (f*,t*) optimizing PT(f,t) ;
    If several f optimize PT(f,t), then select f* randomly among the optimum f ;
    If several t optimize PT(f*,t), then select t corresponding to the lowest $T_t^{f^*}$ as t* ;
    $\mathbf{F}_{map}$(x,y) = $F_{f^*}$ ; $\mathbf{T}_{map}$(x,y) = $T_{t^*}^{f^*}$ ;
end
```

where FNR and FPR denote respectively the false negative rate and the false positive rate, is our prior-independent performance metric. Its behavior is close to that of the geometrical mean of Barandela *et al.* [30], but it is easier to interpret because it measures the distance between a classifier and the theoretically best classifier, also called *oracle* (top left corner of the ROC space). We want to minimize this distance, because lower distances mean being closer to the oracle.

For the purposes of our analysis, we have selected three categories of videos of CDnet [33]: "*Baseline*", "*Camera jitter*", and "*Dynamic background*". Other categories are less relevant, because either they are inappropriate ("*Thermal*" only contains grayscale images), or because our algorithm is not designed to deal with the particularities of that category ("*Shadow*" for example). In fact, it is designed to be robust against camouflage and against background changes that occur during the first part of the training phase (see box [1] in Fig. 1), such as camera jitter, dynamic backgrounds, or illumination changes. Intermittent object motion and shadows are related to foreground objects, and thus cannot be learned during the background learning step.

CDnet [33] provides videos split for the needs of two phases: (1) an initialization phase (ground-truth images are not provided) and (2) an evaluation phase (ground-truth images are provided). Images of the initialization phase are used to train the background model. For videos containing moving objects during this phase, we have manually selected background images, when it was possible. Otherwise, we have manually defined bounding boxes around moving objects and their shadows, to help discarding these areas for building the background models. The first half of images of the evaluation phase is used to estimate the global foreground models, using PAWCS [26] as an estimator. After this step, we perform the local feature/threshold selection. The computed feature and threshold maps are provided to ViBe (designated by ViBe-FT), which then detects the foreground for the second half of images of the evaluation phase. We evaluate the segmentation masks of ViBe-FT on this second half.

4.2 Results and discussion

The results of ViBe-FT are compared to those of the original ViBe [9] algorithm when applied on the 9 color components separately, *i.e.* with uniform feature and threshold maps, denoted by {ViBe-R, ViBe-G, . . ., ViBe-Cb, ViBe-Cr}. The uniform thresholds have been set to {11,11,11,11,11,11,11,3,3} (these values were found to be well suited to the corresponding features). Moreover, we report the results of ViBe when only the feature selection process of Algorithm 1 is activated (thresholds are fixed to the aforementioned values), designated by ViBe-F.

Results displayed in Fig. 2 show that the proposed local feature selection framework significantly improves the detection results. For each category, our feature and threshold maps reduce the mean Euclidean distance of ViBe. This means that our framework pushes the background subtraction algorithm towards the oracle of the ROC space, and thus improves the detection. The conclusion is similar when the threshold selection mechanism is deactivated (denoted by "ViBe-F" on the graphic), which proves the robustness of the local feature selection process with respect to the threshold values.

Fig. 3 shows several feature maps obtained after the local feature selection process, as well as the resulting improved segmentation masks. Note that these improvements are significant despite that (1) a small set of candidate features was used, and (2) the number of training frames used to estimate the global foreground distributions is relatively small (744 on average), which confirms the importance of a *global* estimation for obtaining reliable foreground distributions. The computational complexity of the selection process is $O(\#_{\mathrm{f}}\#_{\mathrm{t}})$. Note however that, as this process occurs during an off-line training phase, it does not affect the computational complexity of the detection task.

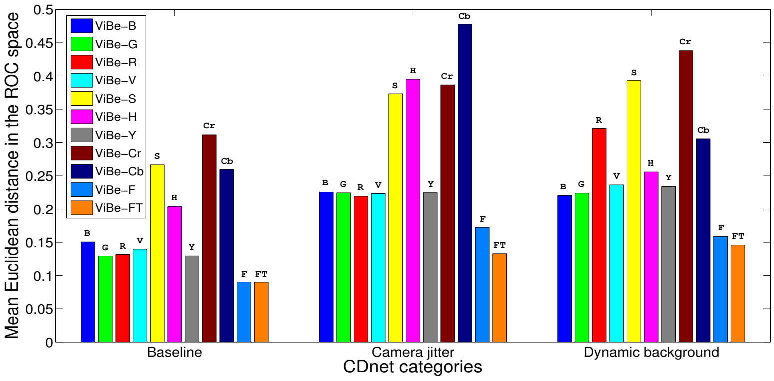

Fig. 2. Mean Euclidean distances in the ROC space for relevant CDnet [33] categories.

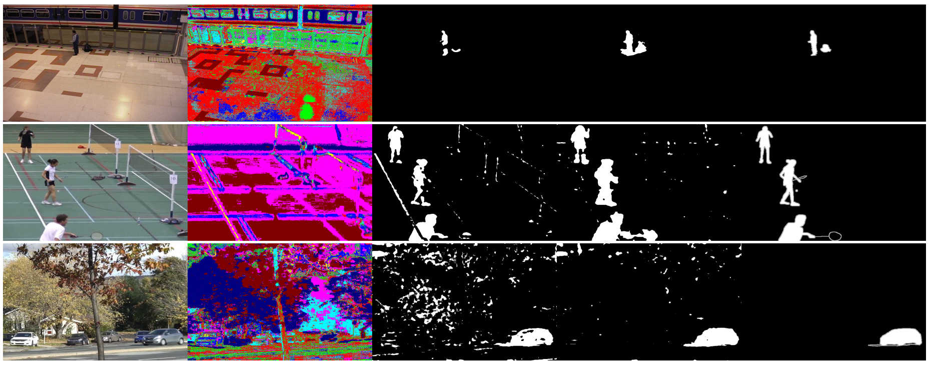

Fig. 3. Improvements obtained with our selection framework for three videos of CDnet [33]. Columns from left to right show: the input images, the computed feature maps (see legend of Fig. 2 for the color-feature matching), the results of ViBe-G (lowest mean d_{euc} among the uniform feature maps), the results of ViBe-FT, and the ground-truths. Note that both ViBe-G and ViBe-FT are post-processed by a 9x9 median filter.

5 Conclusions

In this paper, we present a generic feature selection method for background subtraction. Unlike previous approaches, we estimate the foreground feature statistical distributions *at the frame level*. Features are selected locally depending on their capability to discriminate between *loca*l background samples and *global* foreground samples *for a specific background model* and *for a chosen performance metric*. Experiments led for the ViBe algorithm show that our method significantly improves the performance in the ROC space.

Acknowledgments Marc Braham has a grant funded by the FRIA (`www.frs-fnrs.be`). We are also grateful to Jean-François Foulon for his help.

References

1. Xiao, J., Liao, L., Hu, J., Chen, Y., Hu, R.: Exploiting Global Redundancy in Big Surveillance Video Data for Efficient Coding. Cluster Computing 18(2), 531–540 (2015)
2. Karsch, K., Liu, C., Kang, S.B.: DepthTransfer: Depth Extraction from Video Using Non-Parametric Sampling. IEEE Trans. Pattern Anal. Mach. Intell. 36(11), 2144–2158 (2014)
3. Oliver, N.M., Rosario, B., Pentland, A.P.: A Bayesian Computer Vision System for Modeling Human Interactions. IEEE Trans. Pattern Anal. Mach. Intell. 22(8), 831–843 (2000)
4. Kim, Z.: Real Time Object Tracking Based on Dynamic Feature Grouping with Background Subtraction. In: IEEE Int. Conf. Comput. Vision and Pattern Recognition (CVPR). pp. 1–8 (2008)
5. Jodoin, J.-P., Bilodeau, G.-A., Saunier, N.: Urban Tracker: Multiple Object Tracking in Urban Mixed Traffic. In: IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 885–892 (2014)
6. Jodoin, P.-M., Piérard, S., Wang, Y., Van Droogenbroeck, M.: Overview and Benchmarking of Motion Detection Methods. In: Bouwmans, T., Porikli, F., Hoferlin, B., Vacavant, A. (eds.) Background Modeling and Foreground Detection for Video Surveillance, chap. 24. Chapman and Hall/CRC (2014)
7. Bouwmans, T.: Traditional and Recent Approaches in Background Modeling for Foreground Detection: An Overview. Computer Science Review 11, 31–66 (2014)
8. Elgammal, A., Harwood, D., Davis, L.: Non-Parametric Model for Background Subtraction. In: Vernon, D. (ed.) ECCV 2000, Part II. LNCS, vol. 1843, pp. 751–767. Springer (2000)
9. Barnich, O., Van Droogenbroeck, M.: ViBe: A Universal Background Subtraction Algorithm for Video Sequences. IEEE Trans. Image Process. 20(6), 1709–1724 (2011)
10. Cucchiara, R., Grana, C., Piccardi, M., Prati, A.: Detecting Moving Objects, Ghosts, and Shadows in Video Streams. IEEE Trans. Pattern Anal. Mach. Intell. 25(10), 1337–1342 (2003)
11. Kristensen, F., Nilsson, P., Öwall, V.: Background Segmentation Beyond RGB. In: Narayanan, P.J., Nayar, S.K., Shum, H.-Y. (eds.) ACCV 2006, Part II. LNCS, vol. 3852, pp. 602–612. Springer (2006)
12. Gruenwedel, S., Van Hese, P., Philips, W.: An Edge-Based Approach for Robust Foreground Detection. In: Blanc-Talon, J., Kleihorst, R., Philips, W., Popescu, D., Scheunders, P. (eds.) ACIVS 2011. LNCS, vol. 6915, pp. 554–565. Springer (2011)
13. Heikkilä, M., Pietikäinen, M.: A Texture-Based Method for Modeling the Background and Detecting Moving Objects. IEEE Trans. Pattern Anal. Mach. Intell. 28(4), 657–662 (2006)
14. Bilodeau, G.-A., Jodoin, J.-P., Saunier, N.: Change Detection in Feature Space Using Local Binary Similarity Patterns. In: International Conference on Computer and Robot Vision (CRV). pp. 106–112 (2013)
15. Silva, C., Bouwmans, T., Frélicot, C.: An EXtended Center-Symmetric Local Binary Pattern for Background Modeling and Subtraction in Videos. In: Int. Conf. Comput. Vision Theory and Applicat. (VISAPP). pp. 395–402 (2015)

16. Chen, M., Yang, Q., Li, Q., Wang, G., Yang, M.-H.: Spatiotemporal Background Subtraction Using Minimum Spanning Tree and Optical Flow. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part VII. LNCS, vol. 8695, pp. 521–534. Springer (2014)
17. Freedman, B., Shpunt, A., Machline, M., Arieli, Y.: Depth Mapping Using Projected Patterns, US patent 8150142 (2012)
18. Li, L., Huang, W., Gu, Y.-H., Tian, Q.: Statistical Modeling of Complex Backgrounds for Foreground Object Detection. IEEE Trans. Image Process. 13(11), 1459–1472 (2004)
19. Parag, T., Elgammal, A., Mittal, A.: A Framework for Feature Selection for Background Subtraction. In: IEEE Int. Conf. Comput. Vision and Pattern Recognition (CVPR). vol. 2, pp. 1916–1923 (2006)
20. Schapire, R.E., Singer, Y.: Improved Boosting Algorithms Using Confidence-rated Predictions. Machine Learning 37(3), 297–336 (1999)
21. Grabner, H., Bischof, H.: On-line Boosting and Vision. In: IEEE Int. Conf. Comput. Vision and Pattern Recognition (CVPR). vol. 1, pp. 260–267 (2006)
22. Grabner, H., Roth, P., Grabner, M., Bischof, H.: Autonomous Learning of a Robust Background Model for Change Detection. In: IEEE Int. Workshop on Performance Evaluation of Tracking and Surveillance (PETS). pp. 39–46 (2006)
23. Grabner, H., Leistner, C., Bischof, H.: Time Dependent On-line Boosting for Robust Background Modeling. In: Int. Conf. Comput. Vision Theory and Applicat. (VISAPP). pp. 612–618 (2007)
24. Javed, S., Sobral, A., Bouwmans, T., Jung, S.: OR-PCA with Dynamic Feature Selection for Robust Background Subtraction. In: ACM Symposium on Applied Computing. pp. 86–91. ACM, New York (2015)
25. St-Charles, P.-L., Bilodeau, G.-A., Bergevin, R.: SuBSENSE: A Universal Change Detection Method with Local Adaptive Sensitivity. IEEE Trans. Image Process. 24(1), 359–373 (2015)
26. St-Charles, P.-L., Bilodeau, G.-A., Bergevin, R.: A Self-Adjusting Approach to Change Detection Based on Background Word Consensus. In: IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 990–997 (2015)
27. López-Rubio, F., López-Rubio, E.: Features for Stochastic Approximation Based Foreground Detection. Comp. Vision and Image Understanding 133, 30–50 (2015)
28. Viola, P., Jones, M.: Rapid Object Detection Using a Boosted Cascade of Simple Features. In: IEEE Int. Conf. Comput. Vision and Pattern Recognition (CVPR). vol. 1, pp. 511–518 (2001)
29. Han, B., Davis, L.S.: Density-Based Multifeature Background Subtraction with Support Vector Machine. IEEE Trans. Pattern Anal. Mach. Intell. 34(5), 1017–1023 (2012)
30. Barandela, R., Sánchez, J., García, V., Rangel, E.: Strategies for Learning in Class Imbalance Problems. Pattern Recognition 36(3), 849–851 (2003)
31. Braham, M., Lejeune, A., Van Droogenbroeck, M.: A Physically Motivated Pixel-Based Model for Background Subtraction in 3D Images. In: International Conference on 3D Imaging (IC3D). pp. 1–8 (2014)
32. Fawcett, T.: An Introduction to ROC Analysis. Pattern Recognition Letters 27(8), 861–874 (2006)
33. Goyette, N., Jodoin, P.-M., Porikli, F., Konrad, J., Ishwar, P.: changedetection.net: A New Change Detection Benchmark Dataset. In: IEEE Int. Conf. Comput. Vision and Pattern Recognition Workshop (CVPRW). pp. 1–8 (2012)