MULTIPLE FEATURES BASED SHARED MODELS FOR BACKGROUND SUBTRACTION

Yingying Chen[1], *Jinqiao Wang*[1], *Jianqiang Li*[2] *and Hanqing Lu*[1]

[1]National Laboratory of Pattern Recognition, Institute of Automation
Chinese Academy of Sciences, Beijing, China, 100190
[2]School of Software Engineering, Beijing University of Technology
{yingying.chen, jqwang, luhq}@nlpr.ia.ac.cn, lijianqiang@bjut.edu.cn

ABSTRACT

Background modeling is a fundamental problem in computer vision and usually as the first step for high-level applications. Pixel based approaches usually ignore the spatial coherence, while region based approaches are sensitive to region size and scene complexity. In this paper, we propose a robust background subtraction approach via multiple features based shared models. Each shared model is represented by a sequence of samples based on sample consensus. Each pixel dynamically searches a matched model around the neighborhood. This shared mechanism not only enhances the robustness for background noise and jitter but also significantly reduces the number of models and samples for each model. Besides, we concatenate color and texture features as multiple features according to the discriminability and complementarity, so that each pixel can find a proper model more easily. Finally, the shared models are updated by random selecting a pixel matched the model with an adaptive update rate. Experiments on ChangeDetection benchmark 2014 show that the proposed approach outperforms the state-of-the-art methods.

***Index Terms*—** Background modeling, shared model

1. INTRODUCTION

As the first step in many computer vision tasks such as object tracking, classification, re-identification and retrieval, background subtraction has experienced a rapid development over the past decades. Background subtraction has moved forward from simply comparing a static background frame with current frame to establishing a sophisticated background model of the scene with periodic updates.

Toward a convenient and high-speed implementation, most modern approaches of background subtraction are based on pixel level modeling which assumes adjacent pixels are independent and builds a separate model for each pixel, such as Gaussian Mixture Model (GMM) [1, 2], Kernel Density Estimation (KDE) [3], and non-parametric approaches based on sample consensus (ViBe [4] and PBAS [5]). Color intensities are the most common choice as feature representation or distribution estimation. These pixel based approaches can fully utilize temporal information, and meanwhile partly or totally ignore spatial information between adjacent pixels. Therefore, pixel based approaches are not robust enough to background noise and camera jitter though they are effective and easy to bootstrap. Since color intensities are sensitive to the changes of background illumination, some approaches introduced texture information to enhance the discriminability like LBP [6], SILTP [7] and LBSP [8]. In addition, to incorporate more information around, some region based approaches [9, 10] were proposed by combining a central pixel with the adjacent pixels. This context information enhanced the robustness for background noise, illumination and camera jitter. Some other approaches [11, 12] established models by clustering pixels into different clusters. However, they often lead to performance degradation for foreground object extraction since the region based approaches are sensitive to region size and scene complexity.

Based on our widely observation, it is not necessary to build a background model for all positions since a model can be easily shared by the neighbor pixels with similar appearance. Therefore, in this paper we present a robust background subtraction approach via multiple features based shared models. A given observation is considered as foreground or background based on whether to dynamically find a matched model around the neighborhood. This kind of shared mechanism allows different pixels to share the same model in current frame and different models in next frame. In this way, the shared models could fully exploit spatial-temporal consistency to enhance models robustness as well as reduce the number of models. Each shared model is represented by a sequence of samples based on sample consensus similar to ViBe [4].

In addition, how to select visual features to represent a sample in shared models is critical for the accuracy and stability of foreground object extraction. Color features cannot solve camouflage and illumination variation on the ground that color intensity itself is sensitive to illumination changes while texture features cannot separate smooth foregrounds from smooth backgrounds in most cases [13], the integration of color and texture features is an effective way to enhance the robustness while capture the subtle local changes in complex

978-1-4799-8339-1/15/$31.00 ©2015 IEEE

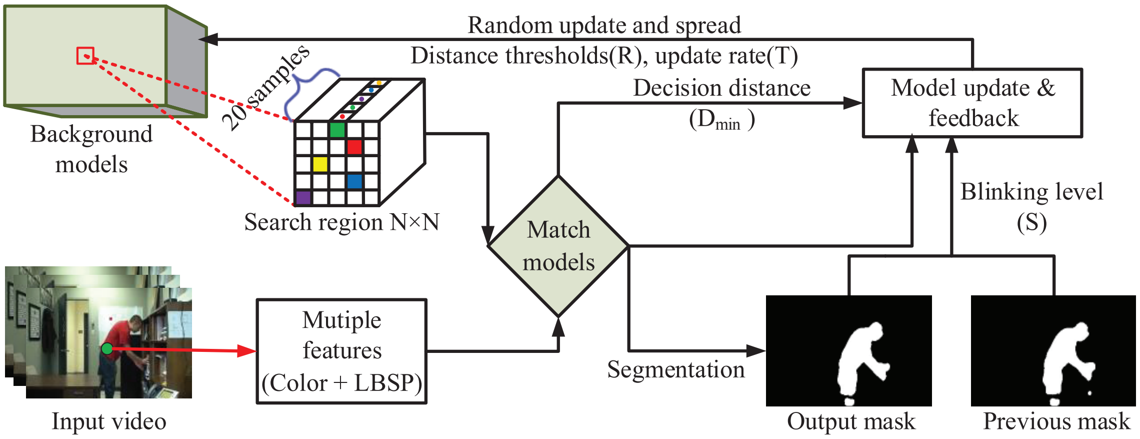

Fig. 1. Overview of the proposed approach.

scenes. Hence, we concatenate color and LBSP features as a rich descriptor to model each sample in the shared models. With the discriminative and complementary descriptor, the proposed shared model needs less samples than pixel-based approaches. Furthermore, due to the dynamic change of the correlation between pixels and models, we update our models by a random sampling strategy. In consideration of the the effectiveness of feedback scheme in PBAS [5] and SuBSENSE [14], we also add an adaptive control of the update rate and segmentation threshold to keep sensitivity and generalization ability for shared model learning.

2. METHODOLOGY

The overview of the proposed approach is illustrated in Fig.1. For a given pixel, we extract the color and texture features. Then we dynamically search a matched model for each pixel around the shared region. The shared mechanism makes dynamic many-to-one correspondence substitute for fixed one-to-one correspondence between pixels and models. Moreover, we update the shared model by random selecting a pixel that matches with the model. To enhance the sensitivity and generalization, we adopt an adaptive threshold and feedback strategy. Like SuBSENSE, we utilize two indicators: decision distance D_{min} and blinking level S to monitor background dynamic and segmentation noise respectively. Then we can realize the automatical control of update rate T for shared models and distance threshold R for sample matching according to these two indicators.

2.1. Multiple features based shared model

In our approach, not all positions need to build a background model since a model can be shared by the neighbor pixels. We establish models by utilizing a sample consensus approach similar to ViBe [4]. Each model is represented by a sequence of historical samples based on sample consensus. Each pixel dynamically searches a matched model around the shared regions. We denote a shared model located at x as $\mathcal{B}(x)$, which contains a collection of K historical samples $B(x)$ noted as:

$$\mathcal{B}(x) = \{B_1(x), B_2(x), ..., B_K(x)\} \tag{1}$$

Each sample is represented by color values $F_{color}(x)$ and LBSP descriptors $F_{LBSP}(x)$ of three color channels. We denote the concatenated multiple features as $F(x)$. Given a pixel x^t in time t, we dynamically search a matched model with the feature $F(x^t)$ from background shared models in a $N \times N$ region. $\mathcal{L}(x^t)$ is defined as a binary label for a pixel x^t. If x^t matches a shared model, $\mathcal{L}(x^t) = 1$, otherwise $\mathcal{L}(x^t) = 0$. $\mathcal{L}(x^t)$ is computed as,

$$\mathcal{L}(x^t) = \begin{cases} 1, & if\ \exists s, L_s(x^t) = 1,\ |s - x| \leq N/2 \\ 0, & otherwise \end{cases} \tag{2}$$

where s is the position of the shared models around pixel x^t in a $N \times N$ region. The similarity $L_s(x^t)$ between a pixel and a model is calculated as Eq. 3.

$$L_s(x^t) = \begin{cases} 1, & if\ \#\{dist(F(x^t), B_n(s)) < R, \forall n\} > \#_{min} \\ 0, & otherwise \end{cases} \tag{3}$$

where R is the maximum distance threshold. $\#_{min}$ is the minimum number of the matched samples between a model and a pixel. We fix $\#_{min} = 2$ in all the experiments. $dist(F(x^t), B_n(s))$ represents the distance between $F(x^t)$ and a given background sample $B_n(s)$ calculated as,

$$\begin{aligned} dist(F(x^t), B_n(s)) = \ & |F_{color}(x^t) - F^{Bn}_{color}(s)| + \\ & D_h(F_{LBSP}(x^t), F^{Bn}_{LBSP}(s)) \end{aligned} \tag{4}$$

where $D_h(\cdot,\cdot)$ is the hamming distance. With the shared mechanism and search strategy, different pixels could share the same model in current frame and different models in next frame. In this way, we can exploit the spatial-temporal correlation between pixels by searching the matched model around a pixel, which can enhance the robustness while significantly decrease the number of models.

2.2. Shared model update

We divide the update into three parts: model update and spread, model feedback of T (update rate for shared models) and R (distance threshold for sample matching), and foreground spread.

Model update and spread: Since a background model is shared by the neighbor pixels, we randomly choose a pixel that matches with the model to update. That is, we randomly select a pixel matched the model to update a random selected sample of the model, then the multiple features of this pixel has $1/T$ probability to replace that sample. Meanwhile, the pixel also has $1/T$ probability to replace a sample of a neighbor model in the search region.

Model feedback: The adaptive control of the update rate T and distance threshold R is critical to affect model sensitivity and generalization ability for shared model updating. To automatically adjust the update rate T and distance threshold R, we add the decision distance D_{min} and blinking level S similar to SuBSENSE [14]. Decision distance D_{min} is the minimal distance between the pixel and samples of the model that it matches, which reflects the degree of background dynamics. Blinking level S is an indicator changing with consistency of consecutive segmentation masks, which reflects segmentation noise. Based on these two indicator, we increase update rate T and distance threshold R in the area that changes dramatically and decrease them in the flat area. More details can be found in [14].

Foreground spread: If a background model is surrounded by foregrounds, i.e., if more than a half pixels closed to this model are foregrounds, we offer a $1/T$ probability to replace a sample by the feature of this foreground pixel.

3. EXPERIMENTS

To evaluate the performance of the proposed approach, we run the experiments on the public ChangeDetection benchmark 2014 [15], which provides a realistic, camera-captured (no CGI), diverse set of videos. A total of 53 video sequences with human labeled ground truth are used for testing.

3.1. Experiments on different shared region size

In this section, we report the performance of the proposed approach with different size of shared regions. Generally, the number of models is proportional to image size and inversely proportional to the size of shared regions. The models are dynamically shared by the pixels so that complexity of image sequences also has some effect on the number of models. Take the sequence "port_0_17fps" as an example, we give the model number of our approach with different size of shared regions in Table 1. In addition, we compare with GMM [1], adaptive GMM [2], RMoG [10] and SuBSENSE [14]. When the size of shared region is 7×7, our approach achieves much better performance with about 10% models compared to original GMM, adaptive GMM and SuBSENSE. Compared to the region based approach, our approach achieves better results with less models than RMoG With 7×7 and 9×9 shared regions. As the increase of shared region size, the model number is reduced. But too large size of shared regions will lower the speed of model matched and cause unnecessary sharing.

Table 1. Comparison results of different shared region size.

	Recall	Precision	F-Measure	Total ♯ of Models
GMM1 [1]	0.4487	0.0121	0.0236	307200
GMM2 [2]	0.4971	0.0095	0.0187	307200
RMoG [10]	0.3558	0.0040	0.0080	34134
SuBSENSE [14]	0.7481	0.0827	0.1490	307200
Ours(5×5)	0.7036	0.4110	0.5189	43815
Ours(7×7)	0.6245	0.5502	0.5850	29174
Ours(9×9)	0.5397	0.6362	0.5840	22901

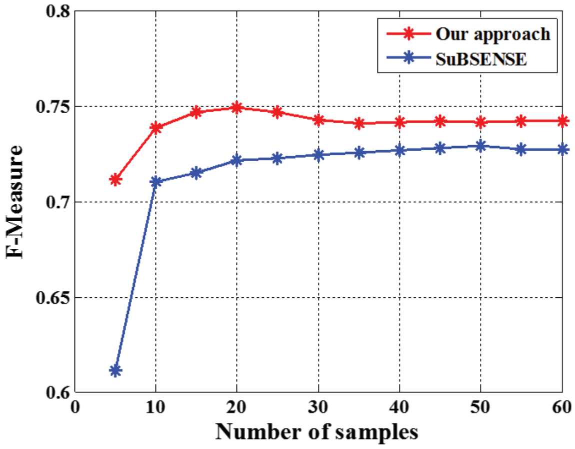

Fig. 2. Average F-Measure on ChangeDetection benchmark 2014 upon different numbers of background samples.

3.2. Experiments on different samples

We further analyze the effect of the sample number on the performance. The comparison results with SuBSENSE on ChangeDetection benchmark 2014 is shown in Fig. 2.

Generally, increasing samples increases precision but decreases recall. Therefore, effective control for the number of samples is critical to balance the precision and recall. As illustrated in Fig. 2, our approach obtains a comparable performance to SuBSENSE when the number of samples is 5. Moreover, the performance of our approach with 10 samples outperforms the best performance of SuBSENSE with 50 samples. Our approach achieves the optimal result with 20 samples, which has a 2% gain than SuBSENSE. Furtherly, we fix the shared region size at 5×5, which means that we use

Table 2. F-Measures for ChangeDetection benchmark [15]. Ba: Baseline; BW: Bad Weather; CJ: Camera Jitter; DB: Dynamic Background; IOM: Intermittent Object Motion; LF: Low Framerate; NV: Night Video; Sh: Shadow; Th: Thermal; Tu: Turbulence; Overall is the average F-Measure of 11 categories.

Approach	Ba	BW	CJ	DB	IOM	LF	NV	PTZ	Sh	Th	Tu	Overall
SuBSENSE[14]	**0.9503**	**0.8619**	0.8152	0.8177	0.6569	0.6445	**0.5599**	0.3476	**0.8986**	0.8171	0.7792	0.7408
FTSG[16]	0.9330	0.8228	0.7513	**0.8792**	0.7891	0.6259	0.5130	0.3241	0.8832	0.7768	0.7127	0.7283
CwisarDH[17]	0.9145	0.6837	0.7886	0.8274	0.5753	0.6406	0.3735	0.3218	0.8476	0.7866	0.7227	0.6812
RMoG[10]	0.7848	0.6826	0.7010	0.7352	0.5431	0.5312	0.4265	0.2470	0.7212	0.4788	0.4578	0.5736
GMM1[1]	0.8245	0.7380	0.5969	0.6330	0.5207	0.5373	0.4097	0.1522	0.7370	0.6621	0.4663	0.5707
GMM2[2]	0.8382	0.7406	0.5670	0.6328	0.5325	0.5065	0.3960	0.1046	0.7322	0.6548	0.4169	0.5566
Spec-360[18]	0.9330	0.7569	0.7142	0.7766	0.5609	0.6437	0.4832	0.3653	0.8187	0.7764	0.5429	0.6732
KDE[3]	0.9092	0.7571	0.5720	0.5961	0.4088	0.5478	0.4365	0.0365	0.7660	0.7423	0.4478	0.5688
Proposed	0.9474	0.8422	**0.8159**	0.8214	**0.6733**	**0.7295**	0.4551	**0.4196**	0.8885	**0.8337**	**0.8445**	**0.7519**

about one-tenth models of SuBSENSE.

We obtain better results with less samples, which benefits from the shared mechanism. Since adjacent pixels can share the same model, models can absorb the discrepancy of pixels thus increasing the diversity of samples. Utilizing the same number of samples, our model can accommodate more context information and adapt more background changes. With the spatial-temporal relationship embedded in the models, incoming pixels can find a matched model more easily than those based on pixel level models. Our approach is more robust to background movements, noise and slight camera jitter thereby improving precision without reducing recall.

3.3. Evaluation on ChangeDetection benchmark

We run the experiments on ChangeDeteciton benchmark 2014 [15] to compared with the state-of-the-art approaches. The shared region is set to 5×5 and the number of sample is 20.

Table 2 presents the quantitative comparison of the proposed approach in terms of F-Measure to several state-of-the-art approaches. Our approach achieves the best performance in six of eleven categories. Note that the proposed approach outperforms all the other approaches on the average F-measure of 11 categories. For PTZ, $LowFramerate$, and $Turbulence$ categories, our approach improves about 5% than the second results. The reason is that shared models effectively remove the background noise and camouflage. Moreover, with the effective sharable mechanism, our approach reduces by about nine-tenths models than SuBSENSE and achieves the best performance at the expense of acceptable computational cost. The average processing time is 45ms per frame with a 5×5 shared region. All programs run on an Intel i7 CPU at 3.4 GHz.

Fig.3 shows some visual comparisons of foreground detection results. The detection results of GMM and SuBSENSE are obtained with BGSLibrary [19]. In Fig.3, the sequences are “intermittentPan” from PTZ,“sofa” from $intermittentObjectMotion$, “sidewalk” and “badminton” from $Camera\ \ Jitter$. From the visual comparison on these sequences, our approach presents effectiveness on removing nonstatic background and acquiring complete foreground.

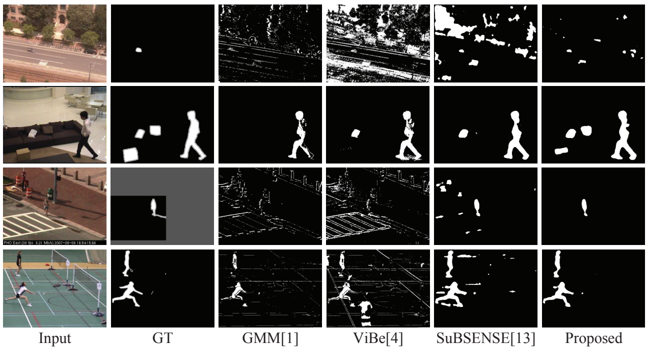

Fig. 3. Visual comparison of foreground detection results.

4. CONCLUSION

We propose to learn multiple features based shared models for robust background subtraction. Each pixel dynamically searches a matched model around the neighborhood. Multiple features are fused for effective appearance modeling of each sample in the shared models, which helps search for suitable models. With the shared mechanism, we allow pixels having similar feature to share the same model, which enhances accuracy and robustness as well as reduces the number of models and samples for each model. A random update strategy and an adaptive update are utilized to keep sensitivity and generalization ability for shared model learning. Experimental results show that our approach outperforms the state-of-the-art methods on ChangeDetection Benchmark 2014.

5. ACKNOWLEDGMENT

This work was supported by 863 Program 2014AA015104, and National Natural Science Foundation of China 61273034, and 61332016.

6. REFERENCES

[1] Stauffer Chris and Grimson W Eric L, “Adaptive background mixture models for real-time tracking,” in *CVPR*. IEEE, 1999, vol. 2.

[2] Zivkovic Zoran, “Improved adaptive gaussian mixture model for background subtraction,” in *ICPR*. IEEE, 2004, vol. 2, pp. 28–31.

[3] Elgammal Ahmed, Harwood David, and Davis Larry, “Non-parametric model for background subtraction,” in *ECCV*, pp. 751–767. Springer, 2000.

[4] Barnich Olivier and Van Droogenbroeck Marc, “Vibe: a powerful random technique to estimate the background in video sequences,” in *ICASSP*. IEEE, 2009, pp. 945–948.

[5] Martin Hofmann, Philipp Tiefenbacher, and Gerhard Rigoll, “Background segmentation with feedback: The pixel-based adaptive segmenter,” in *CVPRW*. 2012, pp. 38–43, IEEE.

[6] Marko Heikkila and Matti Pietikainen, “A texture-based method for modeling the background and detecting moving objects,” *PAMI*, vol. 28, no. 4, pp. 657–662, 2006.

[7] Shengcai Liao, Guoying Zhao, Vili Kellokumpu, Matti Pietikainen, and Stan Z. Li, “Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes,” in *CVPR*. 2010, pp. 1301–1306, IEEE.

[8] Guillaume-Alexandre Bilodeau and Jean-Philippe Jodoin, “Change detection in feature space using local binary similarity patterns,” in *CRV*. 2013, pp. 106–112, IEEE.

[9] XH Fang, W Xiong, BJ Hu, and LT Wang, “A moving object detection algorithm based on color information,” in *Journal of Physics: Conference Series*. IOP Publishing, 2006, vol. 48, p. 384.

[10] Sriram Varadarajan, Paul Miller, and Huiyu Zhou, “Spatial mixture of gaussians for dynamic background modelling,” in *AVSS*. IEEE, 2013, pp. 63–68.

[11] Harish Bhaskar, Lyudmila Mihaylova, and S. Maskell, *Automatic Target Detection Based on Background Modeling Using Adaptive Cluster Density Estimation*, pp. 130–134, Gesellschaft fr Informatik, 2007.

[12] Brian Valentine, Senyo Apewokin, Linda Wills, and Scott Wills, “An efficient, chromatic clustering-based background model for embedded vision platforms,” *CVPR*, vol. 114, no. 11, pp. 1152–1163, 2010.

[13] H. Han, J. Zhu, S. Liao, Z. Lei, and S.Z. Li, “Moving object detection revisited: Speed and robustness,” *Circuits and Systems for Video Technology*, vol. PP, no. 99, pp. 1–1, 2014.

[14] Pierre-Luc St-Charles and Guillaume-Alexandre Bilodeau, “Improving background subtraction using local binary similarity patterns,” in *WACV*. 2014, pp. 509–515, IEEE.

[15] Yi Wang, Pierre-Marc Jodoin, Fatih Porikli, Janusz Konrad, Yannick Benezeth, and Prakash Ishwar, “Cdnet 2014: An expanded change detection benchmark dataset,” in *CVPRW*. 2014, pp. 387–394, IEEE.

[16] Rui Wang, Filiz Bunyak, Guna Seetharaman, and Kannappan Palaniappan, “Static and moving object detection using flux tensor with split gaussian models,” in *CVPRW*. IEEE, 2014, pp. 420–424.

[17] Massimo De Gregorio and Maurizio Giordano, “Change detection with weightless neural networks,” in *CVPRW*. IEEE, 2014, pp. 409–413.

[18] Sedky Mohamed, Chibelushi CC, and MONIRI Mansour, “Image processing: Object segmentation using full-spectrum matching of albedo derived from colour images,” 2010.

[19] Andrews Sobral, “BGSLibrary: An opencv c++ background subtraction library,” in *IX Workshop de Visao Computacional (WVC'2013)*, Rio de Janeiro, Brazil, Jun 2013.