LEARNING SHARABLE MODELS FOR ROBUST BACKGROUND SUBTRACTION

Yingying Chen, Jinqiao Wang and Hanqing Lu

National Laboratory of Pattern Recognition, Institute of Automation
Chinese Academy of Sciences, Beijing, China, 100190
{yingying.chen, jqwang, luhq}@nlpr.ia.ac.cn

ABSTRACT

Background modeling and subtraction is a classical topic in compute vision. Gaussian mixture modeling (GMM) is a popular choice for its capability of adaptation to background variations. Lots of improvements have been made to enhance the robustness by considering spatial consistency and temporal correlation. In this paper, we propose a sharable GMM based background subtraction approach. Firstly, a sharable mechanism is presented to model the many-to-one relationship between pixels and models. Each pixel dynamically searches the best matched model in the neighborhood. This kind of space-sharing way is robust to camera jitter, dynamic background, etc. Secondly, the sharable models are built for both background and foreground. The noises resulted by local small movements could be effectively eliminated through the background sharable models, while the integrity of moving objects is enhanced by the foreground sharable models, especially for small objects. Finally, each sharable model is updated through randomly selecting a pixel which matches this model. And a flexible mechanism is added for switching between background and foreground models. Experiments on ChangeDetection benchmark dataset demonstrate the effectiveness of our approach.

Index Terms— Background modeling, GMM

1. INTRODUCTION

Background modeling is one of the classical and essential problems in computer vision. It plays a significant role on many applications, including object detection, video surveillance, tracking, video compression, etc. As the most commonly used solution, background subtraction is widely applied in above tasks to distinguish foreground and background. In consideration of the first step of the whole process, the performance of background subtraction heavily affects the subsequent steps and the overall results.

Over the past years, diverse approaches, benchmarks and libraries have been developed, which witnesses the importance of background subtraction. Among these approaches, temporal information is fully utilized to build background models. Based on the assumption that a pixel is independent to its adjacent pixels, pixel based background modeling is widely investigated, 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]). Although pixel based approaches are effective and easy to bootstrap, they ignore the spatial relationship between pixels, thus they are not robust enough to the noise and background movement. In order to harness the information around each pixel, some region based approaches [6, 7] were proposed by incorporating the adjacent pixels around a central pixel. This context information enhanced the robustness for background noise and illumination, e.g., block descriptors [7] and local binary similarity patterns (LBSP) [8]. Some other approaches [9, 10] clustered pixels into different classes to build models. However, the region based approaches are usually sensitive to the region size and the complexity of the video scene, which inevitably leads to precision loss.

To fully exploit the spatial-temporal correlation across different pixels, we propose a novel approach to learn sharable models for background substraction. On one hand, we argue that a model can be dynamically shared by different pixels in different frames because adjacent pixels have similar pattern in space and time. It is not necessary to build a background model for each pixel in a texture-consistent region. This sharable strategy could response the relation between pixels and models to reduce the total number of models. On the other hand, we build the sharable models both for background and foreground pixels. As pointed out by [11], the use of explicit foreground models along with background models can be useful. With the sharable strategy, the noises resulted by local small movements can be effectively eliminated by the sharable background models, while the integrity is enhanced by the sharable foreground models. Moreover, due to many-to-one relationship between pixels and models, the sharable models are updated through a randomly sampling strategy, then a flexible mechanism is added for switching between foreground and background models to obtain completed foreground.

In short, the contribution of our work is two-fold: one is that we present a sharable GMM model to exploit the spatial-temporal correlation between pixels, which is robust to cam-

era jitter, dynamic background, etc; the other is that combining sharable background and foreground models with a random update strategy presents the effectiveness on removing nonstatic background and obtaining integral foreground.

2. RELATED WORKS

Among various background modeling approaches, the Gaussian mixture modeling (GMM) is known to be effective in sustaining background variations. In original GMM algorithm, Stauffer *et al.* [1] presented an online EM algorithm which estimated the maximum likelihood of the historical values for GMM background modeling. Their approach had two assumptions: historical values of each pixel in background can be represented by a set of Gaussian distributions; adjacent pixels are entirely independent. Though many improvements have been made [12], this pixel independent assumption makes the approaches vulnerable to the noise.

To alleviate this, some researchers presented to leverage the local context information around a pixel, such as region [6, 13] or block [7], cluster [9, 10] and other spatial-temporal methods [14]. In [6], Fang *et al.* proposed a block-wise GMM method which consisted on a vector of 3×3 neighbors of the current pixel. Latecki *et al.* [13] divided each image into spatial-temporal blocks and obtained compact vector representation of each block to provide a joint representation of motion patterns and textures. These block based approaches are robust to the noise and background movement, but provide less precise foreground detections. In [7], Varadarajan *et al.* modeled regions as mixture distributions to meliorate the flaw of original GMM that could not effectively deal with dynamic background and camera jitter challenged the performance of GMM. In general, region-based model uses local context around a pixel for noise suppression, whereas is relatively complex and vulnerable to pixel changes periodically.

Bhaskar *et al.* [9] clustered pixels with similar features in HSV color space into different regions, and each cluster was modeled by a GMM. Similarly, in [10], they utilized clustering approaches to reduce the number of background models while making models more robust to noise. It is obvious that cluster-wise models are more robust to the noise and pixel-wise models provide more accurate foreground segmentations. The clustering approaches assumes that pixels in the same cluster should experience the same background changes, so that clusters need to initialize again when background changes. Although the clustering approaches reduce the memory usage and the computational cost, the performances are not superior to the pixel based approaches. A case base GMM was proposed in [14] to allow some pixels to share a background model. The pixel value in two consecutive frames and position cues as check features to find a background model. However, this complex features were sensitive to pixel changes periodically and limit the effects of model sharing.

3. LEARNING SHAREABLE MODELS

To exploit the spatial-temporal correlation between pixels, we propose a sharable mechanism that makes each pixel dynamically search for the optimal model from neighboring models according to its color feature. The overall flow of the proposed approach is illustrated in Figure 1.

3.1. Pixel Model

We model the pixel values of an image as a mixture of Gaussians both for background and foreground estimation. This is beneficial for two aspects. One is that foreground models can deal with stopped foreground and moved background objects. The other is that foreground tend to match foreground models so that the completeness of foreground is easily guaranteed. Define x^t as an input pixel value at the time t, and the matching probability of the pixel belonging to a model is:

$$P(x^t) = \sum_{i=1}^{K} \omega_i^t \mathcal{N}_i(x^t, \mu_i^t, \Sigma_i^t) \tag{1}$$

where $\mathcal{N}$ is a Gaussian density function, K is the number of distributions and ω_i^t is the weight of the i^{th} distribution.

$$\mathcal{N}_i(x^t, \mu_i^t, \Sigma_i^t) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma_i|^{\frac{1}{2}}} e^{-\frac{1}{2}(x^t - \mu_i^t)^T (\Sigma_i^t)^{-1} (x^t - \mu_i^t)} \tag{2}$$

where μ_i^t and Σ_i^t are the mean value and the covariance matrix of the i^{th} Gaussian in the mixture, respectively. As Equation (1), we calculate the probability of each pixel matching with all background and foreground models around it.

3.2. Sharable Model

We adopt a sharable mechanism to exploit the spatial-temporal correlation between pixels. Given a pixel x^t, we dynamically search an optimal model from background and foreground models in a $N \times N$ region. We set $\mathcal{B}$ as background models and $\mathcal{F}$ as foreground models. $\mathcal{L}_{\mathcal{B}}(x^t)$ is defined as a binary label for a pixel x^t. If x^t matches a background model, $\mathcal{L}_{\mathcal{B}}(x^t) = 1$, otherwise $\mathcal{L}_{\mathcal{B}}(x^t) = 0$. In a similar fashion, if x^t matches a foreground model, $\mathcal{L}_{\mathcal{F}}(x^t) = 1$, otherwise $\mathcal{L}_{\mathcal{F}}(x^t) = 0$. The labeling criterion is computed as follows,

$$\mathcal{L}_{\mathcal{B}}(x^t) = \begin{cases} 1, & if \quad |x^t - \mu_{\mathcal{B},k}^t| < 2.5\sigma_{\mathcal{B},k}^t \\ 0, & otherwise \end{cases} \tag{3}$$

$$\mathcal{L}_{\mathcal{F}}(x^t) = \begin{cases} 1, & if \quad |x^t - \mu_{\mathcal{F},k}^t| < 2.5\sigma_{\mathcal{F},k}^t \\ 0, & otherwise \end{cases} \tag{4}$$

where $\mu_{\mathcal{B},k}$ and $\sigma_{\mathcal{B},k}$ are the mean and standard deviations of the k^{th} distribution of a background model. $\mu_{\mathcal{F},k}$ and $\sigma_{\mathcal{F},k}$ are the mean and standard deviations of the k^{th} distribution

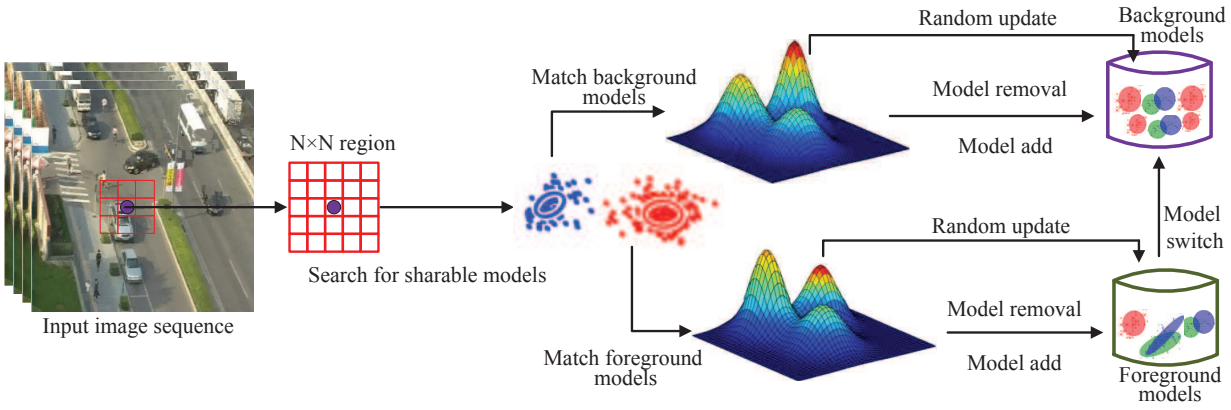

Fig. 1. Overview of the proposed approach.

of a foreground model. Then we calculate the matching probability with background and foreground models respectively:

$$P(x^t|\mathcal{B}) = \begin{cases} \sum_{i=1}^{K} \omega_i \mathcal{N}_{\mathcal{B},i}(x^t, \mu_i^t, \Sigma_i^t) & if \ \ \mathcal{L}_{\mathcal{B}}(x^t) = 1 \\ 0 & if \ \ \mathcal{L}_{\mathcal{B}}(x^t) = 0 \end{cases} \tag{5}$$

$$P(x^t|\mathcal{F}) = \begin{cases} \sum_{i=1}^{K} \omega_i \mathcal{N}_{\mathcal{F},i}(x^t, \mu_i^t, \Sigma_i^t) & if \ \ \mathcal{L}_{\mathcal{F}}(x^t) = 1 \\ 0 & if \ \ \mathcal{L}_{\mathcal{F}}(x^t) = 0 \end{cases} \tag{6}$$

To find an optimal model, we perform an exhaustive search for all the matched models in a $N \times N$ region around the pixel x^t. The model of maximum probability is obtained with the following selection function.

$$P(x^t) = \arg\max\{P_i(x^t|\mathcal{B}), P_j(x^t|\mathcal{F})\}, \quad 1 < i < m, 1 < j < n \tag{7}$$

$$\mathcal{L}(x^t) = \begin{cases} 1, & if \ a \ background \ model \ is \ chosen \\ 0, & if \ a \ foreground \ model \ is \ chosen \end{cases} \tag{8}$$

where m and n are the number of models that need to match within the $N \times N$ region. $\mathcal{L}(x^t)$ is the final pixel label, and $P(x^t)$ is the maximum matching probability.

In this way, a pixel is labeled into foreground or background by finding an optimal matched model in the neighborhood. If no model is chosen, then the pixel is regarded as foreground and a new model is established. To prevent the overlapping of background and foreground distributions, we add distance constraint when establishing the foreground models. If $\mathcal{L}(x^t) = 1$ and no matched foreground model, a new foreground model is built when $|x^t - \mu^t| > 5\sigma^t$. The benefit is two-fold. One is to reduce the number of foreground models. The other is to effectively suppress the noise pixels.

To illustrate the validity of sharable models, we output the mean value of the optimal matched model for a pixel to generate a shared map. Figure 2 shows some examples of shared maps, where we can observe distinctly that the shared maps reconstruct the original image. Actually, the shared maps look like a blur effect of the original image. The wider the search range is, the more blurry the shared maps look.

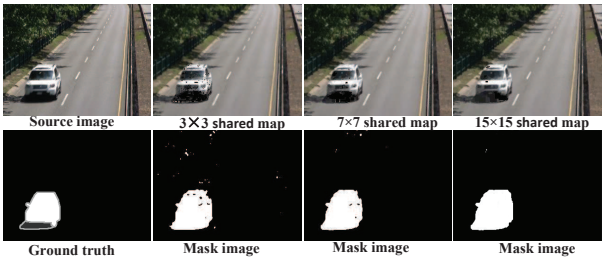

Fig. 2. Examples of shared maps with different search ranges.

3.3. Update

Original GMM adopts blind update to incorporate all samples into the background models. Since our approach has two kinds of sharable models, the update confronts with two questions: the many-to-one relationship between pixels and models; the switch between background and foreground models.

Model update: We select a conservative update strategy with a random sampling mechanism. Through randomly selecting a pixel that matching a model, the background and foreground models are updated respectively. This is simple but effective for the many-to-one relationship between pixels and models.

Model switch: The conservative update may lead to deadlock situations and everlasting ghosts. For example, the background incorrectly classified as foreground prevents its background model from being updated. To solve this problem, we add a switch to transform foreground models into background models if a foreground model is used for a long period. We set the switch time to 500 frames.

Model remove: If a sharable model is not used for a long time, it is deleted. Here, we set the life time to 500 frames for background models and 50 frames for foreground models.

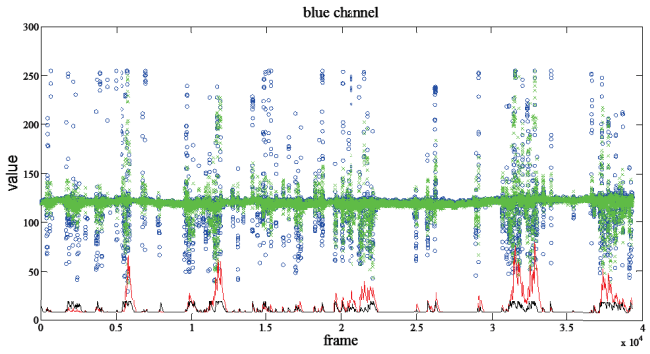

Fig. 3. Variance changes for the blue channel of a background model in "highway" sequence. The green points representing random pixel values are much more smooth than the blue ones standing for pixel values in original GMM. The back line shows that the variance degeneracy effectively controlled.

Table 1. Comparison results of different shared region size. bg: background; fg: foreground.

	Recall	Precision	F-measure	Total ♯ of Models
GMM1 [1]	0.4904	0.7463	0.5919	86400
GMM2 [2]	0.5075	0.9290	0.6564	86400
RMoG [7]	0.4296	0.9232	0.5864	9600
Our bg (3×3)	0.9654	0.8924	0.9275	22034
Our bg+fg (3×3)	0.9157	0.9619	0.9347	24185
Our bg+fg (5×5)	0.9431	0.9549	0.9489	10106
Our bg+fg (9×9)	0.8623	0.9637	0.9102	4511

3.4. Variance Control

Original GMM is faced with variance degeneracy as pointed out in [15]. The phenomenon is that the variance increases until the component saturates the mixture by spanning over the entire pixel color range. As a consequence, pixels are constantly classified as either foreground or background, depending on the weight of the Gaussian component. In our approach, the pixels tend to be classified as background or foreground models with the sharable mechanism, instead of as different distributions of the same model. Hence, variance degeneracy can be suppressed to some extent. However, due to many-to-one relationship between pixels and models, we adopt a random sampling strategy to update, which may aggravate the variance degeneracy at some point. According to our experiments, the change of pixels that are randomly chosen is more smooth. Figure 3 gives an example of variance changes for the blue channel of a background model in $40k$ frames. Take a background model as example, when the background is stable, smooth pixels are conducive to effectively restrain the variance degeneracy. However, when the background changes continuously or foreground objects have similar color with background, these pixels are classified as the same distribution with the background and variance increases gradually. This is shown in Figure 3. Hence, we set an upper limit for each model.

4. EXPERIMENTS

To evaluate the performance of the proposed approach, we perform the experiments on the public ChangeDetection benchmark 2014 [16], which provides a realistic, camera-captured, diverse set of videos. A total of 53 video sequences with human labeled ground truth are used for testing. The video sequences are separated into eleven categories based on different types of challenges shown in Table 3.

4.1. Experiments on Different Shared Region Size

This section reports the performance of our approach with different size of shared regions. Since models are shared dynamically with a random update mechanism, the number of sharable models changes with time. Take the sequence "office" as an example, we compare our approach with different size of shared regions in Table 1. Additionally, we compare with GMM [1], adaptive GMM [2], RMoG [7]. With $1/3$ models compared to original GMM and adaptive GMM, our approach achieves much better performance. with 9×9 shared region, our approach achieves better results with a half number of models than RMoG. Furtherly, in Table 1 we can see that the addition of foreground sharable models improves precision but decreases recall a little. On the whole, the F-measure is greatly improved due to more integral foreground. With the increase of shared region size, the model number is reduced. But too large shared region will degrade the performance. In this paper, we set the shared region size as 5.

4.2. Experiments on Variance Control

In Table 2, we show the effects of variance control. We can find that the recall increases with variance control, while the precision in "highway" sequence decreases a little and the precision in "turbulence3" sequence increases a little. The F-measures of the two sequences are improved about 1% respectively. This indicates that the effectiveness of variance control against variance degeneracy. Generally, the variance degeneracy occurs in background sharable models when the background changes continuously. Thus, with the introduce of variance control, the performance is improved.

4.3. Evaluation on ChangeDetection Benchmark

In this section, we report the performance of the proposed approach on ChangeDetection benchmark 2014 [16]. The learning rate α is 0.01 for both background and foreground. The shared regions of $Camera\ \ jitter$ and $Dynamic\ \ Background$ are set as 9 and others are set as 5. And the other parameters are given in Section 3.

Table 3. F-measures for ChangeDetection benchmark [16]. BW: Bad Weather; Ba: Baseline; 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; Overall* is the average F-measure except Shadow.

Approach	Ba	BW	CJ	DB	IOM	LF	NV	PTZ	Sh	Th	Tu	Overall	Overall*
SuBSENSE[17]	**0.9503**	**0.8619**	0.8152	0.8177	0.6569	0.6445	**0.5599**	0.3476	**0.8986**	0.8171	0.7792	0.7408	0.7250
FTSG[18]	0.9330	0.8228	0.7513	**0.8792**	0.7891	0.6259	0.5130	0.3241	0.8832	0.7768	0.7127	0.7283	0.7128
CwisarDH[19]	0.9145	0.6837	0.7886	0.8274	0.5753	0.6406	0.3735	0.3218	0.8476	0.7866	0.7227	0.6812	0.6635
RMoG[7]	0.7848	0.6826	0.7010	0.7352	0.5431	0.5312	0.4265	0.2470	0.7212	0.4788	0.4578	0.5736	0.5588
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	0.5541
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	0.5390
Spec-360[20]	0.9330	0.7569	0.7142	0.7766	0.5609	0.6437	0.4832	0.3653	0.8187	0.7764	0.5429	0.6732	0.6553
KNN[21]	0.8411	0.7587	0.6894	0.6686	0.5918	0.5491	0.4200	0.2126	0.7788	0.6046	0.5198	0.5937	0.5856
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	0.5454
SCSOBS[22]	0.9333	0.6620	0.7051	0.6865	0.5026	0.5463	0.4503	0.0409	0.7230	0.6923	0.4880	0.5961	0.5707
Proposed	0.9346	0.7791	**0.8173**	0.8673	**0.7979**	**0.6664**	0.4333	**0.3734**	0.8133	**0.8254**	**0.8556**	**0.7421**	**0.7350**

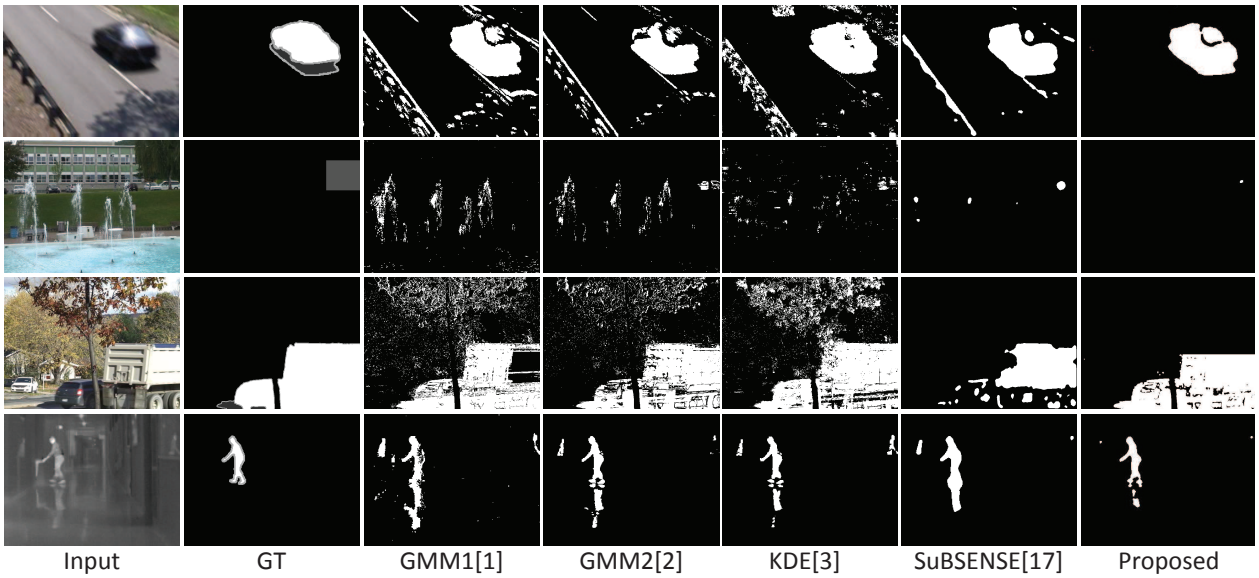

Fig. 4. Visual comparison of foreground detection results.

Table 2. Comparison results of variance control.

	Recall	Precision	F-measure
highway	0.9108	0.9305	0.9206
highway(with control)	0.9355	0.9270	0.9312
turbulence3	0.8000	0.8727	0.8348
turbulence3(with control)	0.8105	0.8741	0.8411

Table 3 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. However, the improvement is not significant. This is because we do not consider shadow modeling. With the $Shadow$ is excluded, the improvement over the state-of-the-art is more significant as shown in last column of Table 3. Moreover, the performance on $Shadow$ can be greatly improved with some shadow detection approaches, such as [23]. Although our approach only has a 1% gain than SuBSENSE [17]. SuBSENSE needs about $135ms$ to process a frame for 320×240 videos with the implementation [24], while our approach needs about $30ms$. With an effective sharable mechanism, our approach reduces by about two third models than original GMM and achieves the best performance at the expense of acceptable computational cost. All programs run on a second generation Intel i7 CPU at 3.4 GHz with no architecture-specific instruction and saving output frames on a local hard drive.

Figure 4 shows some visual comparisons of foreground detection results. The foreground detection results of other

approaches are obtained with BGSLibrary [24]. In Figure 4, the first sequence is "highway" from $Camera\ \ Jitter$; the second and third sequences are "fountain01" and "fall" from $Dynamic\ \ Background$; the last sequence is from $Thermal$. From these sequences, our approach presents effectiveness on removing nonstatic background and obtaining integral foreground.

5. CONCLUSION

We propose a simple but effective approach to learn sharable models for background subtraction. Through dynamically establishing many-to-one relationship between pixels and models, we allow pixels having similar feature to share the same model. By jointly foreground modeling and random update, our approach reduces more nonstatic background and obtains more complete foreground. Experimental results show that our approach outperforms the state-of-the-art methods on ChangeDetection Benchmark 2014.

6. ACKNOWLEDGMENT

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

7. REFERENCES

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

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

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

[4] Olivier Barnich and Marc Van Droogenbroeck, "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, IEEE.

[6] 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.

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

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

[9] 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.

[10] 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.

[11] Yaser Sheikh and Mubarak Shah, "Bayesian modeling of dynamic scenes for object detection," *PAMI*, vol. 27, no. 11, pp. 1778–1792, 2005.

[12] Thierry Bouwmans, Fida El Baf, and Bertrand Vachon, "Background modeling using mixture of gaussians for foreground detection-a survey," in *Recent Patents on Computer Science 2008*. 2008, Bentham Science Publishers.

[13] Dragoljub Pokrajac and Longin Jan Latecki, "Spatiotemporal blocks-based moving objects identification and tracking," *VS-PETS*, pp. 70–77, 2003.

[14] Atsushi Shimada, Yosuke Nonaka, Hajime Nagahara, and Rin ichiro Taniguchi, "Case-based background modeling: associative background database towards low-cost and high-performance change detection," *MVA*, vol. 25, pp. 1121–1131, 2014.

[15] P.L.M. Bouttefroy, A. Bouzerdoum, and S. L. Phung, "On the analysis of background subtraction techniques using gaussian mixture models," in *ICASSP*. 2010, IEEE.

[16] 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, IEEE.

[17] Pierre-Luc St-Charles, Guillaume-Alexandre Bilodeau, and Robert Bergevin, "Flexible background subtraction with self-balanced local sensitivity," in *CVPRW*. IEEE, 2014, pp. 414–419.

[18] 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.

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

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

[21] Zoran Zivkovic and Ferdinand van der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *PRL*, vol. 27, no. 7, pp. 773–780, 2006.

[22] Lucia Maddalena and Alfredo Petrosino, "The sobs algorithm: what are the limits?," in *CVPRW*. IEEE, 2012, pp. 21–26.

[23] Andrea Prati, Ivana Mikic, Mohan M Trivedi, and Rita Cucchiara, "Detecting moving shadows: algorithms and evaluation," *PAMI*, vol. 25, no. 7, pp. 918–923, 2003.

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