

# Moving Object Detection Using Energy Model and Particle Filter for Dynamic Scene

Wooyeol Jun<sup>(✉)</sup>, Jeongmok Ha, and Hong Jeong

Department of Electrical Engineering,  
Pohang University of Science and Technology (POSTECH),  
Cheongam-ro 77, Pohang, South Korea  
{wyjun, jmokha, hjeong}@postech.ac.kr

**Abstract.** We proposed an algorithm that uses an energy model with smoothness assumption to identify a moving object by using optical flow, and uses a particle filter with a proposed observation and dynamic model to track the object. The algorithm is based on the assumption that the dominant motion is background flow and that foreground flow is separated from the background flow. The energy model provides the initial label foreground object well, and minimizes the number of noise pixels that are included in the bounding box. The tracking part uses HOG-3 as an observation model, and optical flow as the dynamic model. This combination of models improves the accuracy of tracking results. In experiments on challenging data set that have no initial labels, the algorithm achieved meaningful accuracy compared to a state-of-the-art technique that needs initial labels.

**Keywords:** Moving Object Detection · Object tracking · Initial label estimation · Particle filter · Optical flow

## 1 Introduction

Use of computers to recognize moving objects in a dynamic scene video image is a challenging problem in computer vision. Moving Object Detection (MOD) in dynamic scene is a requirement before deployment of smart cars is feasible. The smart car industry demands accurate MOD in dynamic scene algorithms to detect objects that may pose a danger to a driver or vehicle. However many MOD algorithms cannot satisfy these demands.

The main objective of MOD is to find a moving an object in a video. Several approaches to solve this problem have been suggested: to use a detection algorithm in every video frame; or uses geometry or image feature to cluster background pixels and foreground pixels; or to track an object after it has been detected.

Part-based models assume that an object is a combination of its parts and generate a part model for detection [5]. Detection schemes based on neural networks train their own network to identify features that can be used to detect

a target object [11,18]. However these models consider every object, including immobile ones, and can find only predefined objects such as humans or vehicles.

Some algorithms use homography, or a combination of homography and parallax to find foreground objects [19]. However if a foreground object is relatively large, the background flow is distorted. These algorithms have the serious deficit that they lose detected objects if they stop moving. Some methods try to estimate the feature of a moving object such as motion estimation and appearance model by using information from several image frames [1,3,9,17]. This online background subtraction needs long-term trajectories, and therefore is unsuitable in rapidly-changing conditions. Another method is try to represent background in combinations of image column called low rank representation [2]. This approaches fails when background moves fast or camera moves fast.

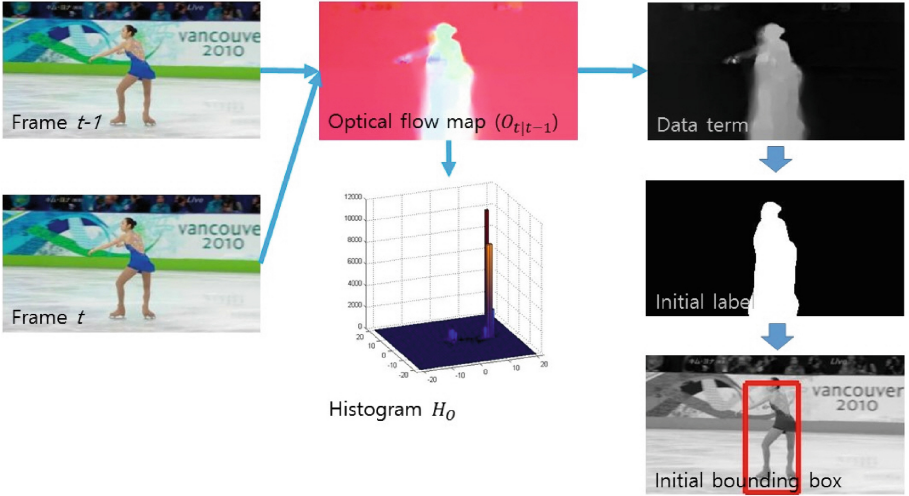
Tracking scheme can be divided into pixelwise tracking and objectwise tracking. Pixelwise tracking label every pixel as foreground or background in every frame. Objectwise tracking follow the foreground object as area like bounding box.

Pixelwise tracking schemes combined models of the foreground and of the background. Generalized background subtraction uses motion segmentation to find an initial label and to form foreground and background models for superpixels [4,12,13]. Each model classifies each pixel as foreground or background by matching each pixel to models. However if the initial label contains background pixels, the foreground label spreads out over time. Furthermore if the foreground objects has similar color to backgrounds, the foreground label also spreads out over time. Pixelwise tracking-based algorithms use an initial label assumption to find an object to track. Motion segmentation uses point trajectories for long term observation, and segmentation to detect moving objects [15,16]. These approaches can assign a good initial label but include some background pixels. Detection algorithms use the learned image features to find target object, and the result of detection contains many background pixels.

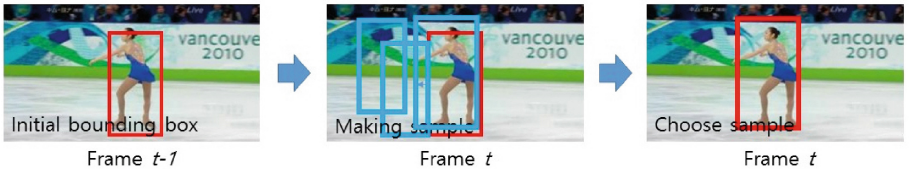
Objectwise tracking uses dynamic model and observation model to estimate objects position. Estimated objects is compared with the object in previous frame. A particle filter uses samples to track objects [7,8]. The filter assembles samples based on their dynamic models and uses the observation model to find the best matching sample. However if the dynamic model cannot follow an object's motion, tracking fails or if the observation model is ambiguous to given foreground object and background, tracking also fails.

We propose an algorithm that can detect a moving object and track it with low computation cost. Given frames and pixelwise optical flow, we use a smoothing assumption and the distance between foreground and background flow to form an energy function to decouple the moving object label from image. The decoupled initial labels are grouped, and each label is marked with a bounding box.

We used Histogram Of Gradients-3 (HOG-3) which is robust to gradient [10] as a particle filter's observation model to track each bounding box respectively with optical flow dynamic model. Because of it the proposed algorithm can find a stopped object that was previously moving. The main contribution and characteristics of the proposed algorithm are described below.



(a) The flow of initial label estimation procedure



(b) The flow of tracking procedure

**Fig. 1.** The flow of the proposed algorithm, optical flow map shows the movement vector of pixel in frame  $t - 1$ . (a) The red box represents estimated initial bounding box. (b) The red box represents tracked bounding box and the blue box represents samples expected to include target object (Color figure online).

1. We used a well-defined energy model to find the initial label of a moving object and to avoid motion smoothing that can cause the initial label to spread out.
2. We used HOG-3 in the particle filter's observation model; use of HOG-3 improves the tracking quality by ensuring that observations are accurate.
3. We apply an optical flow model to a particle filter's dynamic model to estimate the foreground's motion.

## 2 Overview

The proposed algorithm consists of initial label estimation and tracking (Fig. 1). To estimate the initial label, which we assume to be foreground, we used an optical flow map (Fig. 1a). At time  $t$ , optical flow map  $\mathbf{O}_{t|t-1}$  for frame  $t - 1$  to  $t$  is calculated [14].  $\mathbf{O}_{t|t-1}$  consists of background flows and foreground flows.

In our assumption, foreground flows are different from the background flow, but background flows are similar to each other. We adopt a histogram  $H_{\mathbf{O}}$  to distinguish foreground flows from  $\mathbf{O}_{t|t-1}$ . Because most background flows are included in the same bins, the bin with the largest number of flows represents estimated background optical flow  $\mathbf{m}_b$ , and foreground flows are included in bins that are far from the background bin  $B_b$ . Because of the smoothness assumption that most optical flow algorithms adopt, a blurred background flow near the foreground object is classified as foreground flow and some small irrelevant objects are classified as foreground objects. We adopt a Belief Propagation (BP) algorithm [6] to eliminate these unwanted foreground objects. For every pixel  $\mathbf{p}$ , we calculated the distance from the estimated background motion  $\mathbf{m}_b$  and each pixel's optical flow  $\mathbf{m}_{\mathbf{p}}$ . Using this assumption we construct MRF model of initial label.

Tracking applies an observation model and a dynamic model (Fig. 1b). The observation model scores the similarity between the target object and candidate object. The dynamic model estimates the position of the target object in frame  $t$  using optical flow  $\mathbf{O}_{t|t-1}$  and target position in frame  $t-1$ . The observation model calculated uses the HOG-3 feature. We score the similarity by calculating the distance between HOG-3 descriptor for the target object and HOG-3 descriptors for each candidate objects.

### 3 Initial Label Estimation

We assumed that foreground flow has relevant flow that differs from background flow. We used this assumption to guide construction of a foreground probability map. First, optical flow map  $\mathbf{O}_{t|t-1}$  for frame  $t-1$  to  $t$  is calculated. Each pixel  $\mathbf{p}$  has its own optical flow  $\mathbf{m}_{\mathbf{p}} = \mathbf{O}(\mathbf{p})$ . Denote background pixel as  $\mathbf{p}_b$  and foreground pixel as  $\mathbf{p}_f$ , then all  $\mathbf{m}_{\mathbf{p}_b}$  are similar to each other and differ from  $\mathbf{m}_{\mathbf{p}_f}$ .

Second, we construct a histogram of optical flow  $H_{\mathbf{O}}$  that has  $50 \times 50$  bins. Most background optical flows  $\mathbf{m}_{\mathbf{p}_b}$  are included in one bin  $B_b$  and foreground optical flows  $\mathbf{m}_{\mathbf{p}_f}$  are included in bin that is far from the background flow. A background bin can be estimated as

$$B_b = \Gamma(\arg \max_{\mathbf{m}} H_{\mathbf{O}}(\mathbf{m})), \quad (1)$$

where  $\Gamma(\cdot)$  is a function for input  $\mathbf{m}$  to output bin that includes  $\mathbf{m}$ . Estimated background flow is calculated by

$$\bar{\mathbf{m}}_b = \Theta(B_b), \quad (2)$$

where  $\Theta(\cdot)$  is a function for input bin to output bin's representative value.

Third, use the background flow  $\bar{\mathbf{m}}_b$  and flow map  $\mathbf{O}_{t|t-1}$  to we construct a label distribution. By the Bayesian Rule,

$$P(\mathbf{l}|\mathbf{o}) = \prod_{\mathbf{p}} P(o|\mathbf{l})P(\mathbf{l}), \quad (3)$$

where  $o$  means observation image at pixel  $\mathbf{p}$  and  $l$  means label at pixel  $\mathbf{p}$ . We assumed that  $p(o|l)$  has a Gaussian distribution

$$P(o|l) = \exp\left(\frac{-\|\mathbf{m}_{\mathbf{p}} - \bar{\mathbf{m}}_b\|^2}{2\sigma^2}\right), \quad (4)$$

where  $\sigma$  means variance. However due to the smoothness assumption,  $\mathbf{m}_{\mathbf{p}_b}$  near  $\mathbf{p}_f$  is blurred by the foreground flow that is misclassified as  $\mathbf{p}_f$ . Furthermore, the label for foreground can have hole or disconnection for one object. We used prior information  $P(l)$  to prevent these problem.  $l$  is assumed to be smooth, big enough and follow image data as

$$P(l) = \exp((1 - 2\eta)(l - l') + \eta), \quad (5)$$

where  $l'$  denotes label at neighbor pixel  $\mathbf{p}$  and  $\eta$  is a balance constant. Finally, an energy model of initial label problem is defined by

$$\begin{aligned} E &= E_d + E_s \\ &= D^d(\mathbf{m}_{\mathbf{p}}, \bar{\mathbf{m}}_b) + D^s(l, l'), \end{aligned} \quad (6)$$

where  $D^d(\mathbf{m}_{\mathbf{p}}, \bar{\mathbf{m}}_b)$  from Eq. (4) and  $D^s(l, l')$  from Eq. (5). To solve this equation, we adopt the BP algorithm and constant message weight. BP refines the smooth area of a foreground label and connects the label for each object. Then we count disconnected labels and surround each label by a bounding box.

## 4 Tracking

The initial calculated labels contain several background pixels. Other tracking algorithms [4, 12, 13] spread the foreground when the initial label is inaccurate. Our idea follows the particle filter sequence, in which the computation cost is small and a relatively inaccurate initial label is acceptable.

We adopt optical flow to reinforce the sample prior, and the HOG-3 feature to complement the observation model. The particle filter uses weighted samples to estimate the posterior probability  $P(x_t|z_{1:t})$  in a sequential Bayesian filtering manner, where  $x$  means state and  $z$  means observation. By sequential Bayesian filtering,

$$P(x_t|z_{1:t}) = P(z_t|x_t) \int P(x_t|x_{t-1})P(x_{t-1}|z_{1:t-1})dx_{t-1}, \quad (7)$$

where  $P(z_t|x_t)$  is the observation model and  $\int P(x_t|x_{t-1})P(x_{t-1}|z_{1:t-1})dx_{t-1}$  is the dynamic model. In a tracking manner, we must find the maximum posterior bounding box; i.e., the box in which  $x$  is the vector of the top-left pixel and the bottom-right pixel. Therefore every sample represents a bounding box, and a box's posterior means the probability that the box is suitable.

The dynamic model represents their assumed position. For this purpose we used optical flow with a Gaussian assumption. We assumed that most pixels in

a bounding box are foreground, and that the average optical flow of a bounding box is foreground optical flow  $\mathbf{O}_f$ .  $\mathbf{O}_f$  is used in the sample prior,

$$P(x_t|x_{t-1}) = \mathcal{N}(\mathbf{O}_f, \Sigma), \quad (8)$$

where  $\mathcal{N}(\cdot, \cdot)$  means Gaussian distribution and  $\Sigma$  means variance. When choosing the best sample in the Maximum a Posterior (MAP) rule, we eliminate other samples. Therefore we make samples based on one  $x_{t-1}$  sample with that bounding box's optical flow.

A particle filter can use any reasonable observation technique. We adopted HOG-3 feature which generates a histogram of zero-gradients, first-gradients, and second-gradients. The observation model  $P(z_t|x)$  can be substituted to

$$P(z_t|x_t^i) = \frac{1}{D(\mathcal{H}(z_{t-1}, \bar{x}_{t-1}), \mathcal{H}(z_t, x_t^i))}, \quad (9)$$

where  $D(\cdot, \cdot)$  means distance between the input descriptor,  $\mathcal{H}(a, b)$  calculates HOG-3 descriptor of input  $a$  cropped by left top of  $b$  and right bottom of  $b$ ,  $i$  means index of candidate object, and  $\bar{x}_{t-1}$  means the chosen sample in frame  $t - 1$ . Finally, we choose MAP sample  $x_t$  at frame  $t$ , remove other samples, and used it as  $x_{t-1}$  for the next frame.

## 5 Experiment

The proposed method was tested in many challenging videos that include fast background motion and complex foreground. We analyze our algorithm quantitatively by comparison to a simple particle filter, and qualitatively compare our algorithm to the state-of-the-art algorithm.

### 5.1 Experimental Environment

We performed our experiment on a quad core i7-3770CPU @3.40 GHz with 8 GB DDR3 RAM. The algorithms were tested in MATLAB 2015a. To compute dense optical flow maps, we used SIFT flow, which is a state-of-the-art optical flow algorithm [14]. The optical flow map is smooth at the boundary of the foreground and its result can affect the result of our algorithm. We can use any optical flow algorithm but we do not consider this option in our experiment.

The experiment was processed in two environment. The first experiment quantified the tracking accuracy of the proposed algorithm and the particle filter algorithm with changing parameters. The second experiment compared the tracking accuracy of the proposed algorithm qualitatively with that of Generalized background subtraction using superpixels with label integrated motion estimation (GBS-SP) and Generalized background subtraction based on hybrid inference by belief propagation and bayesian filtering (GBS-BP), and showed the characteristics of our algorithm. The proposed algorithm, GBS-SP and GBS-BP

**Table 1.** Comparison of the proposed algorithm and particle filter. PF represents the particle filter and Ours represents the proposed algorithm. All IOUs are determined by averaging the IOU of the whole frames.

Scenarios	The number of particles $n$									
	20		50		80		110		140	
	PF	Ours	PF	Ours	PF	Ours	PF	Ours	PF	Ours
<i>skating</i>	0.44	<b>0.68</b>	0.50	<b>0.72</b>	0.64	<b>0.74</b>	0.72	<b>0.76</b>	0.74	<b>0.75</b>
<i>car1</i>	0.87	<b>0.88</b>	0.88	<b>0.89</b>	0.88	<b>0.89</b>	0.89	<b>0.90</b>	0.89	<b>0.91</b>
<i>car2</i>	<b>0.93</b>	0.84	<b>0.94</b>	0.84	0.93	<b>0.94</b>	0.86	<b>0.95</b>	0.88	<b>0.96</b>
<i>people1</i>	0.82	<b>0.91</b>	0.87	<b>0.91</b>	0.80	<b>0.88</b>	0.88	<b>0.90</b>	0.88	<b>0.90</b>
<i>people2</i>	<b>0.56</b>	0.55	0.53	<b>0.56</b>	0.59	<b>0.61</b>	0.61	<b>0.62</b>	0.47	<b>0.64</b>
<i>PV person</i>	0.68	<b>0.69</b>	<b>0.66</b>	0.61	0.67	<b>0.69</b>	0.70	0.70	0.69	<b>0.70</b>

involve several free parameters. We fixed them for each algorithm in the first experiment, then used the best parameters in the second experiment. In the second experiment, because GBS-SP and GBS-BP does not inform the motion segmentation for the initial label in the paper, we used the ground truth initial label, as in their experiment environment. We also used our initial label for GBS-SP and GBS-BP, which works poorly and compared all of this in qualitative manner. To evaluate the proposed algorithm, we used the proposed video set in GBS-SP and GBS-BP which includes a variety of foreground objects and situations, and real car-embedded camera videos.

## 5.2 Performance Evaluation for Tracking

We compared our algorithm and particle filters. To evaluate the tracking accuracy we use Intersection of Union (IOU) for frame  $t$  as

$$I_t = \frac{area(E \cap G)}{area(E \cup G)} \quad (10)$$

where,  $E$  is the estimated bounding box and  $G$  is the ground truth bounding box. We calculate IOU of the proposed algorithm and particle filter with a range of sample number  $n$ . All IOUs were determined by averaging the IOU of the whole frame. To consider the effect of the proposed observation model and dynamic model, we tested them on a dynamic scene (Table 1). In all of the dataset, the proposed algorithm achieved better IOU than did the particle filter. In skating video that camera and foreground movement was fast, IOU of the proposed algorithm was much larger than that of particle filter. This is because as the scene continues, the particle filter missed the target object (Fig. 2). This result means that the proposed algorithm works well in dynamic scenes and that the model is better than the regular particle filter.



**Fig. 2.** Comparison with the proposed algorithm and particle filter for *skating*, *car1* and *car2* video. The images in each column represents the tracking result of each algorithm. The blue box represents the ground truth bounding box and the red box represents the tracked bounding box. Notice that the particle filter can not follow the foreground object in *skating* video (Color figure online).

### 5.3 Performance of Challenging Environment

We compared our algorithm, GBS-SP, and GBS-BP. The proposed algorithm track foreground in a bounding box, whereas GBS-SP and GBS-BP use pixelwise tracking, so quantitative comparison of the two algorithms is impossible. Instead we tested in variety of videos and showed characteristics of the proposed method with regard to GBS-SP and GBS-BP. Because GBS-SP and GBS-BP need a fine initial label, which is hard to pick out in real world scenes, the foreground area is spread out with a smoothed initial label. Furthermore, in a driving environment, the optical flow spreads out from the vanishing point, so the background optical flow will be ambiguous and the color of driving environment similar to foreground object that the foreground area is spread out.

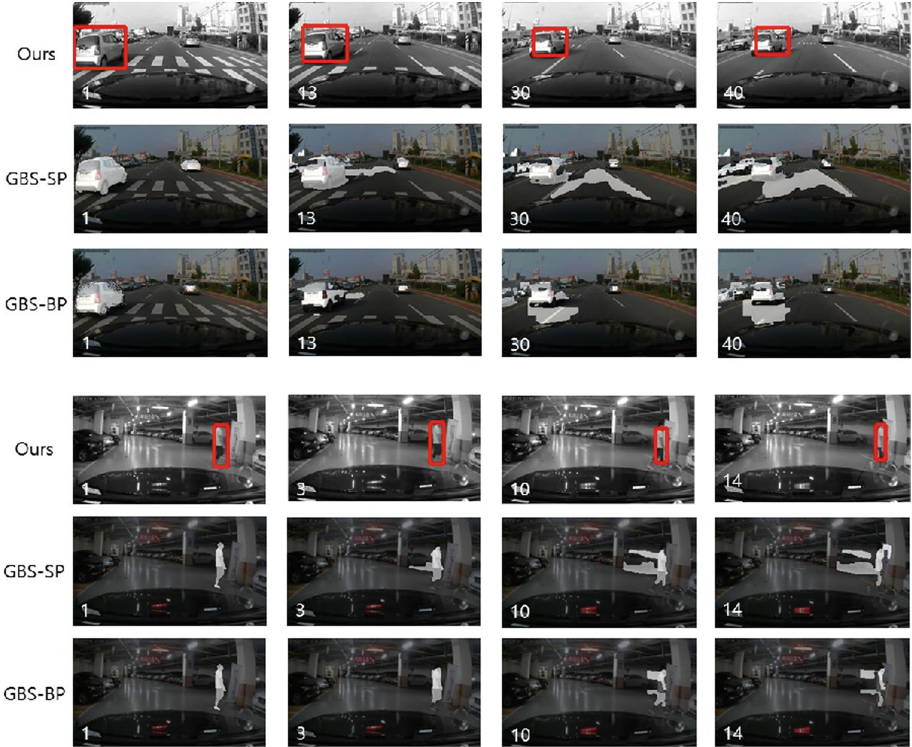




**Fig. 3.** Comparison with the proposed algorithm, GBS-SP, and GBS-BP for *skating* video with inexact initial label. The images in each column represents the tracking result of each algorithm. The blue box represents the ground truth bounding box and the red box represents the tracked bounding box. Notice that the foreground label of GBS-SP and GBS-BP are spread out over time (Color figure online).

We tested two environments: one with a smoothed initial label and one for a real car-embedded video scene in which the car moves and the foreground is relatively ambiguous. In the smoothed initial label experiment, we used the proposed initial label algorithm and compared the proposed algorithm, GBS-SP and GBS-BP. To compare in the car-embedded camera video data, we used the initial ground truth label to compare each algorithm.

The proposed algorithm followed the object well when the initial label was relatively smooth, whereas GBS-SP and GBS-BP spread the foreground label (Fig. 3). In the car-embedded video scene, our algorithm followed the target well, but GBS-SP and GBS-BP spread out the given ground truth foreground label (Fig. 4). These results mean that in a real car-embedded video scene the proposed method works better than GBS-SP and GBS-BP.



**Fig. 4.** Comparison with the proposed algorithm, GBS-SP, and GBS-BP for *real car-embedded video scene* video with exact initial label. The images in each column represents the tracking result of each algorithm. The red box represents the tracked bounding box of our algorithm and white area represent the tracked foreground label of GBS-SP and GBS-BP. Notice that the foreground label of GBS-SP and GBS-BP are spread out over time and the proposed method can follow the foreground object well (Color figure online).

## 6 Conclusions

For MOD, we showed a solution that uses a proposed initial label estimation method and an improved particle filter. The proposed method to estimate initial labels is based on the assumption that the foreground’s optical flow differs from the background optical flow. Using this assumption, we proposed a data term for the initial label, and a new energy model for it. The estimated initial label is tracked by the particle filter using the proposed observation model and dynamic model. The proposed method was evaluated on a video set in GBS-SP, GBS-BP, and a challenging car-embedded camera video data. The proposed algorithm achieved higher IOU than the particle filter and in challenging video tracked objects more accurately than did GBS-SP and GBS-BP. When a smoothed initial

label is given, the proposed algorithm can track the foreground. The proposed algorithm can be used in the challenging problem in smart car industry that need fast and accurate MOD algorithm.

**Acknowledgments.** This work was supported by the Human Resource Training Program for Regional Innovation and Creativity through the Ministry of Education and National Research Foundation of Korea (NRF-2014H1C1A1066380).

## References

1. Brox, T., Malik, J.: Object segmentation by long term analysis of point trajectories. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part V. LNCS, vol. 6315, pp. 282–295. Springer, Heidelberg (2010)
2. Cui, X., Huang, J., Zhang, S., Metaxas, D.N.: Background subtraction using low rank and group sparsity constraints. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part I. LNCS, vol. 7572, pp. 612–625. Springer, Heidelberg (2012)
3. Elqursh, A., Elgammal, A.: Online moving camera background subtraction. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part VI. LNCS, vol. 7577, pp. 228–241. Springer, Heidelberg (2012)
4. Ess, A., Leibe, B., Gool, L.V.: Depth and appearance for mobile scene analysis. In: IEEE 11th International Conference on Computer Vision, 2007. ICCV 2007, pp. 1–8. IEEE (2007)
5. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. *IEEE Tran. Pattern Anal. Mach. Intell.* **32**, 1627–1645 (2010)
6. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient belief propagation for early vision. *Int. J. Comput. Vis.* **70**, 41–54 (2006)
7. Gordon, N., Ristic, B., Arulampalam, S.: *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, London (2004)
8. Haug, A.: *A Tutorial on Bayesian Estimation and Tracking Techniques Applicable to Nonlinear And Non-gaussian Processes*. MITRE Corporation, McLean (2005)
9. Hayman, E., Eklundh, J.O.: Statistical background subtraction for a mobile observer. In: Ninth IEEE International Conference on Computer Vision, 2003. Proceedings, pp. 67–74. IEEE (2003)
10. Jiang, Y., Ma, J.: Combination features and models for human detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 240–248 (2015)
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*, pp. 1097–1105 (2012)
12. Kwak, S., Lim, T., Nam, W., Han, B., Han, J.H.: Generalized background subtraction based on hybrid inference by belief propagation and bayesian filtering. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2174–2181. IEEE (2011)
13. Lim, J., Han, B.: Generalized background subtraction using superpixels with label integrated motion estimation. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part V. LNCS, vol. 8693, pp. 173–187. Springer, Heidelberg (2014)

14. Liu, C., Yuen, J., Torralba, A., Sivic, J., Freeman, W.T.: SIFT flow: dense correspondence across different scenes. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part III. LNCS, vol. 5304, pp. 28–42. Springer, Heidelberg (2008)
15. Mittal, A., Huttenlocher, D.: Scene modeling for wide area surveillance and image synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2000, vol. 2, pp. 160–167. IEEE (2000)
16. Ochs, P., Malik, J., Brox, T.: Segmentation of moving objects by long term video analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **36**, 1187–1200 (2014)
17. Sheikh, Y., Javed, O., Kanade, T.: Background subtraction for freely moving cameras. In: 2009 IEEE 12th International Conference on Computer Vision, pp. 1219–1225. IEEE (2009)
18. Szarvas, M., Yoshizawa, A., Yamamoto, M., Ogata, J.: Pedestrian detection with convolutional neural networks. In: Proceedings of the IEEE, Intelligent Vehicles Symposium, 2005, pp. 224–229. IEEE (2005)
19. Yuan, C., Medioni, G., Kang, J., Cohen, I.: Detecting motion regions in the presence of a strong parallax from a moving camera by multiview geometric constraints. *IEEE Trans. Pattern Anal. Mach. Intell.* **29**, 1627–1641 (2007)