A Machine Learning Inspired Approach for Detection, Recognition and Tracking of Moving Objects from Real-Time Video

Anit Chakrabory[1,2](✉) and Sayandip Dutta[1,2]

[1] RCC Institute of Information Technology, Kolkata, India
ianitchakraborty@gmail.com, sayandip199309@gmail.com
[2] MCKV Institute of Engineering, Howrah, India

Abstract. In this paper, we address the problem of recognizing moving objects in video im-ages using Visual Vocabulary model and Bag of Words. Initially, the shadow free images are obtained by background modelling followed by object segmentation from the video frame to extract the blobs of our object of interest. Subsequently, we train a Visual Vocabulary model with human body datasets in accordance with our domain of interest for recognition. In training, we use the principle of Bag of Words to extract necessary features to certain domains and objects for classification, similarly, matching them with extracted object blobs that are obtained by subtracting the shadow free background from the foreground. We track the detected objects via Kalman Filter. We evaluate our algorithm on benchmark datasets. A comparative analysis of our algorithm against the existing state-of-the-art methods shows very satisfactory results to go forward.

Keywords: Background modelling · Bag of words · Object detection · Object recognition · Visual vocabulary

1 Introduction

Effective recognition of objects for tracking in video stream and processing of data involve integration of background modelling, shadow removal, analysis of segmented objects from the video frames and proper detection of objects. Subsequently, recognition of the detected objects is done by extracting the features adopting the machine learning inspired principle, bag of words.

In our paper, we use the Visual Vocabulary Model using Bag of Words to extract the necessary features of certain instances of objects through rigorous high-level training. Subsequently, we apply the extracted feature sets to the test domain to recognize and locate our objects of interest in the video scenes. Using visual instance occurrence and their probabilistic presence to imply a certain domain, we obtain optimum accuracy in domain recognition as well.

The contributions of this paper are:

- Background modelling and extraction of astute shadow free images using color invariant approach.

B.U. Shankar et al. (Eds.): PReMI 2017, LNCS 10597, pp. 170–178, 2017.
https://doi.org/10.1007/978-3-319-69900-4_22

- Extraction of the features of the objects captured in the blobs via the principle of Bag of Words.
- Classification of the objects in a certain domain of interest using probabilistic word occurrence for domain recognition.

The organization of the paper constitutes: Sect. 2 briefly explains the related works in the respective domain, Sect. 3 explains the proposed method for detection and recognition, specifically, Sect. 3.3 describes the concept of Visual Vocabulary Model for object recognition. Experimental results on several datasets and the comparative analysis with some state-of-the-art algorithms are presented in Sect. 4. Section 5 concludes the paper and discusses future possibilities for further improvements.

2 Brief Review of Related Works

Numerous color histograms based object detection algorithms have been proposed in recent years. He et al. [4] developed a locality sensitive histogram at each pixel for finer distribution of the visual feature points for object tracking in video scenes. Haar-like features have been proposed for appearance based tracking of objects [5–7, 9]. Spatiotemporal representation combined with genetic algorithm has also been used for feature extraction [1]. Recently pixel based segmentations have been applied [2] to handle tracking.

In recent years, the classifiers that have been extensively used for object tracking are: ranking SVM [7], semi-boosting [14], support vector machine (SVM) [12], boosting [13], structured output SVM [8], and online multi-instance boosting [6]. Various detection and tracking codes are available for evaluation with significant effort of the authors, e.g., MIL, IVT, TLD, FCT, VTD and likes.

3 Proposed Method

Initially, we model the segmented objects from the video frames and subtract the background model without shadow to obtain the blob of an object. Before recognizing the object inside the blob, we train a machine learning inspired Visual Vocabulary Model with a set of objects which can represent our domain of interest for recognition and tracking. We extract the features of the objects of both the training data and test data by principle of Bag of Words, in the training and testing phases respectively.

3.1 Background Modeling

In [10], Li et al. proposed an idea for background modelling. In our work, we introduce some modification over the same work and proceed as follows: At each time step an image I_m^t is obtained by subtracting two successive video frames and F_m^t can be obtained by subtracting the current video frame with the background model. To deal with sudden illumination variation an AND-OR operation is performed over I_m^t and F_m^t. The extracted frame I^t is compared with its previous frame I^{t-1} in order to obtain I_m^t by predicting the similarity between the two consecutive pixel values of frames I_t (x, y) and I_{t-1}(x, y). Pixel centers are compared between the succeeding images (I^t (x, y), I^{t-1} (x, y)). Temporal binary image of the moving object (I_m) has a radiometric similarity value, formally expressed as:

$$I_m(x, y) = \begin{cases} 1, & if\, R(x, y) > T_b \\ 0, & otherwise \end{cases}. \tag{1}$$

Similarly, F^t_m is formulated on a hypothesis based on the difference threshold (T_b), between background frame and the current frame, formally:

$$F^t_m = \begin{cases} 1, & if\, |I^t(x, y) - B^t(x, y)| > T_b \\ 0, & otherwise \end{cases}. \tag{2}$$

The pixels (x, y) of moving objects are formulated by operating on $I_m(x, y)$ and $F^t(x, y)$:

$$M^t(x, y) = \begin{cases} 1, & if\, (I_m(x, y) \cap F^t(x, y)) = 1) \\ 0, & otherwise \end{cases}. \tag{3}$$

The moving pixels in video frames are identified by $M^t(x, y)$.

In our implementation, a vector history V, with the six last values updated cumulatively, is considered as:

$$V = [E(t), E(t-1), E(t-2), E(t-3), E(t-4), E(t-5)]. \tag{4}$$

At time t, the mean value of pixel intensities in the frame is E(t). For each frame, we calculate proper learning rate α, based on this vector:

$$\alpha = a + b\frac{|E(t) - E(t-5)|}{\max(E(t), E(t-5))}, \tag{5}$$

Let d be a pixel of the image, the gray histogram of the pixel is *h(d)*, and background pixels and foreground pixels are denoted by I_B and I_F respectively. Probability of a background pixel misidentified as foreground pixel and vice versa are as follows:

$$P_{F|B} = \sum\nolimits_{d \in I_F} p(d|B) \,\text{and}\, P_{B|F} = \sum\nolimits_{d \in I_B} p(d|F), \tag{6}$$

where $P_{d|B}$ is the probability of background pixel and $P_{d|F}$ is the probability of foreground pixel.

Our goal is to minimize $P_{d|B}$ and $P_{d|F}$ as much as possible.

The Min $P_{F|B}$ is significant, as after morphological operation in the post-process, $P_{B|F}$ will be smaller.

$p(B)$ is the priori probability of the background as calculated from gray histogram of the image I^t_m.

$$p(B) = \sum\nolimits_{d=-T}^{T} h(d) \quad \mu = 0. \tag{7}$$

3.2 Shadow Removal

As mentioned in [11] by Xu *et al.*, by formally normalizing the pixels to r, g, b color space the shadow-free color invariant image can be constructed:

$$r' = \frac{r}{\sqrt{r^2 + g^2 + b^2}},\ g' = \frac{g}{\sqrt{r^2 + g^2 + b^2}},\ b' = \frac{b}{\sqrt{r^2 + g^2 + b^2}}. \tag{8}$$

where r, g, b are input image color channels, r', b', g'.

Application of Gaussian smooth filter suppresses the high frequency textures in both invariant and original images, formally:

$$E_{ori} = ||edge(I_{ori})||, E_{inv(i)} = ||edge\left(I_{inv(i)}\right)||, \tag{9}$$

where E_{ori} is the edge of the original image after applying smooth filter and I_{ori} is the original image. $E_{inv(i)}$ is the edge of the color invariant image after applying smooth filter and $I_{inv(i)}$ is the color invariant image. The hard shadow edge mask is constructed by choosing the strong edges of original images that are absent in the invariant images. Thus, we get:

$$HS(x,y) = \begin{cases} 1, & E_{ori}(x,y) > t1, \& \\ & \min_i(E_{inv(i)}(x,y) < t2), \\ 0, & otherwise \end{cases} \tag{10}$$

where $t1$, $t2$ are thresholds, set manually, based on the empirical analysis of datasets and assessed hard shadow edge mask is *HS(x,y)*. In (10), $t1$ maps the selected shadow edges to the strong edges of the subsequent hard shadows in images. $t2$ selects edges belonging only to shadows, as shown in Fig. 1.

3.3 Visual Vocabulary Model for Object Recognition

Visual Vocabulary Model is a machine learning based image classification model, specifically, handling images as documents, by labelling specific features as words by observing presence of such feature key words in an image.

First, we localize the key words by extracting the features of the object of interest such that they are distinct and invariant under different scale and illumination based conditions even with the presence of noise. We have used Nonlinear (cubic) Support Vector Machine (SVM) as the feature classifier. Polynomial kernel for cubic SVM is:

$$K(x,y) = \left(x^T y + c\right)^3. \tag{11}$$

Here x and y are input vector features, calculated from the training samples. A free parameter, $c \geq 0$, is indicating how far the equation is from homogeneity.

The following equation expresses the contribution of a feature f, at location l, at position x in the object class o_n with matching visual keyword index (C_i) indicating its potentiality of belonging to the class o_n. Thus, we get:

$$p(o_n, x|f, l) = \sum\nolimits_i p(o_n, x|C_i, l)\, p(C_i|f), \tag{12}$$

Mean-shift mode estimation with a kernel K, along with scale-adaptive kernel, is used to obtain the maxima in this space:

$$\hat{p}(o_n, x) = \frac{1}{V_{\mathrm{b}}(x_s)} \sum\nolimits_k \sum\nolimits_j p\big(o_n, x_j|f_k, l_k\big) K\left(\frac{x - x_j}{b(x_s)}\right). \tag{13}$$

Kernel bandwidth is denoted by b, and volume is denoted by V_b, which are varied over the radius of the kernel. In order to fix the hypothesized interest object, size and scale coordinate $\boldsymbol{x}_s$ is updated in parallel. This strategy makes it easier to deal with partial occlusions and also typically requires fewer training examples.

The pictorial structure model represents any object of interest as collection of parts, connected in pairs, and defined by a graph $G = (V,\ E)$, where the nodes $V = \{v_1, \ldots, v_n\}$ defines the parts and the edges $\big(v_i, v_j\big) \in E$ describes the corresponding connections.

$L = \{l_1, \ldots, l_n\}$ be a certain arrangement of part frame locations. Then the matching of the model to a video frame is formulated using an energy minimization function:

$$l_1^* = \arg\min\nolimits_{l_1}\left(m_1(l_1) + \sum\nolimits_{i=2}^{n} \min_{l_i} m_i(l_i) + ||l_i - T_{1i}(l_1)||_{M_{ij}}^2\right). \tag{14}$$

where M_{ij} is the diagonal covariance between transformed locations $T_{ij}(l_i)$ and $T_{ji}\big(l_j\big)$.

For further improvement of our validation score by approximating the similarity measures, we discriminatively model a linear time matching function, represented by the Pyramid Match Kernel (PMK) model to bridge the feature sets to the variable cardinalities. Let the input of a histogram pyramid be X ϵ S where $\Psi(\mathrm{X}) = [H_0(X), \ldots, H_{L-1}(X)]$, number of pyramid levels expressed as L. The histogram vector of point X is defined by $H_i(X)$.

Similarity between two input set of features Y and Z is expressed as:

$$\kappa_{PMK}(\Psi(Y), \Psi(Z)) = \sum\nolimits_{i=0}^{L-1} \omega_i(I(H_i(Y), H_i(Z)) - I(H_{i-1}(Y), H_{i-1}(Z))), \tag{15}$$

where $I(H_i(Y), H_i(Z))$ signifies the histogram intersection of two input set of features Y and Z at i^{th} level of the pyramid.

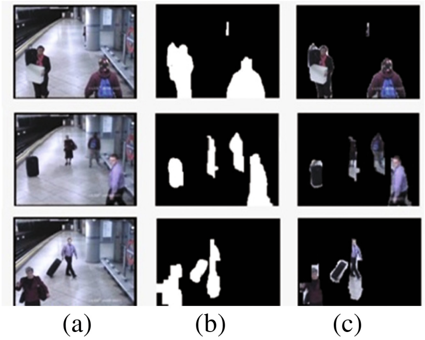

Fig. 1. (a) Video Frame, (b) Segmented Object Model, (c) Foreground Model.

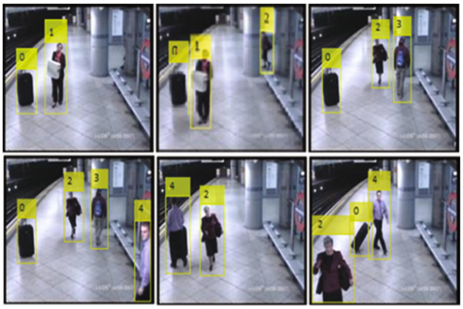

Fig. 2. Tracking results on INRIEA.

Finally, the features of the recognized objects are tracked via the classical Kalman Filter, which can also efficiently handle the tracking under partial occlusions as shown in Fig. 2. The performance measure of the proposed algorithm is done with respect to available benchmark datasets and we obtain very satisfactory and competitive results.

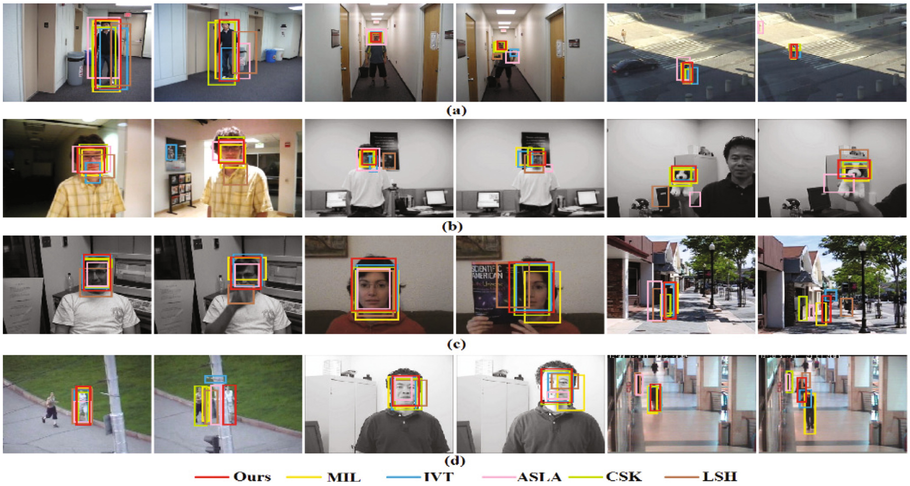

Fig. 3. Sample tracking results of the eight top performed trackers on challenging sequences. (a) Result samples on BlurBody, Boy and Crossing sequences. Challenging factors: background clutter and deformation. (b) Result samples on David, David2 and Dog1 sequences. Challenging factors: scale variation, motion blur, and occlusion. (c) Result samples on Dudek, FaceOcc1 and Human9 sequences. Challenging factors: deformation and occlusion. (d) Result samples on Jogging, Mhyang and Walking2 sequences. Challenging factors: fast motion, scale variation, and occlusion.

4 Experimental Results and Analysis

We test our algorithm on various benchmark datasets [3] with the aforementioned settings. Using the trained model as a reference to recognize newly arrived objects, we compare our algorithm with the other state-of-the-art algorithms, in other datasets as well for the validation our experiment. The tracking result of our algorithm on INRIA Person dataset and on other datasets in multiple frames handling various challenges, is shown in Figs. 2 and 3 respectively.

The overlap rate of tracking methods indicates stability of each algorithm by taking the pose and size of the target object into consideration in Table 1. Our algorithm achieves competitive, rather satisfactory results compared to the other state-of-the-art tracking algorithms [3]. Figure 4 represents a comparative analysis of the overlap rate in video frames against the other state-of-the-art methods showing competitive as well as satisfactory outcomes.

Table 1. Average overlap rate of tracking methods. The red, yellow and orange colors indicate the results ranked at the first, second and third places, respectively.

Sequences	Ours	CSK	FCT	HT	IVT	ASLA	MIL	PT	SPT	TLD	VTD
Blurbody	**0.59**	0.43	0.31	0.35	0.25	0.35	0.22	0.51	0.46	**0.56**	**0.72**
Boy	**0.74**	0.69	**0.77**	**0.78**	0.65	0.51	**0.72**	0.46	0.34	**0.78**	0.29
Crossing	0.39	0.58	0.69	0.36	0.68	0.75	0.29	0.49	**0.76**	**0.80**	**0.79**
David	**0.74**	0.57	**0.71**	0.31	**0.73**	0.65	0.38	0.28	0.52	0.47	0.34
David2	**0.67**	0.41	0.53	0.61	**0.75**	0.44	0.21	**0.71**	0.21	0.21	**0.67**
Dog1	0.70	0.34	**0.74**	0.25	0.58	0.39	**0.67**	0.27	0.33	**0.80**	0.24
Dudek	**0.73**	0.23	0.56	**0.79**	0.26	0.37	0.68	0.56	0.41	0.57	**0.72**
FaceOcc1	0.57	0.58	**0.73**	0.24	0.22	**0.75**	**0.73**	0.54	0.61	**0.73**	**0.71**
Gym	**0.50**	0.48	0.45	0.22	0.33	0.20	0.52	**0.66**	0.47	0.39	**0.77**
Jogging2	0.70	0.23	0.44	**0.77**	0.34	0.24	0.74	0.55	0.61	**0.75**	0.40
Mhyang	**0.51**	**0.65**	0.38	0.25	**0.73**	0.43	0.54	0.32	0.37	0.35	0.37
Walking2	**0.76**	**0.80**	0.51	0.37	0.23	0.45	**0.75**	0.31	0.40	0.58	0.33
BlurFace	**0.72**	0.43	0.31	0.26	0.40	**0.63**	0.37	0.21	**0.61**	0.22	0.33
Deer	0.55	0.63	0.51	0.36	0.65	0.40	**0.80**	0.28	0.60	**0.73**	**0.78**
Dog	**0.74**	0.45	0.56	0.44	0.34	0.30	**0.73**	0.52	**0.67**	0.49	0.38
Football	**0.67**	0.21	0.24	**0.80**	0.35	0.54	0.39	0.65	0.66	**0.77**	**0.67**
Jump	0.48	**0.64**	0.63	0.56	**0.75**	0.55	0.56	0.26	**0.75**	0.48	**0.79**
Dancer	**0.69**	0.30	**0.46**	0.26	0.32	0.41	0.31	0.33	**0.46**	0.37	**0.66**
Couple	0.63	**0.76**	**0.76**	0.72	**0.80**	**0.79**	0.24	0.43	0.73	0.68	0.32
Trellis	0.50	0.27	0.51	0.41	**0.61**	0.48	0.30	**0.68**	**0.64**	0.35	0.47
Woman	**0.62**	0.40	0.49	0.61	0.22	**0.68**	0.34	**0.62**	0.50	**0.69**	**0.68**
Girl2	0.52	0.43	0.31	0.35	0.25	0.35	0.22	**0.51**	0.46	**0.56**	**0.72**
Average	**0.62**	0.48	0.53	0.46	0.47	0.48	0.49	0.46	0.53	**0.56**	**0.55**

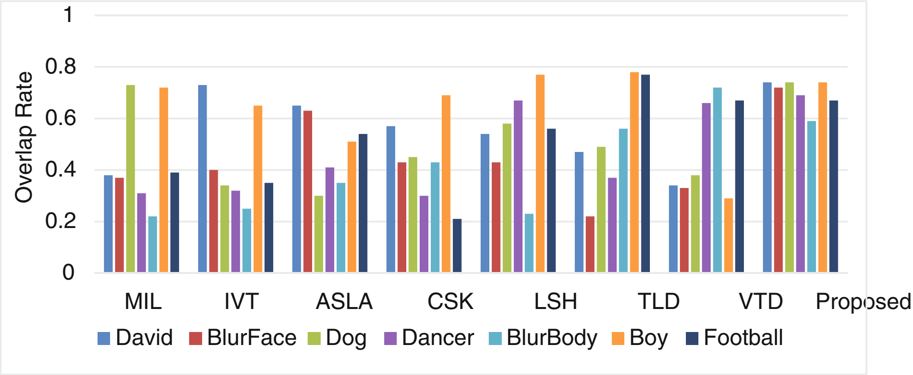

Fig. 4. Comparative analysis of overlap rate against the state-of-the-art methods on various benchmark datasets and challenges.

5 Conclusion

This paper presents object detection and recognition of the detected objects based on Visual Vocabulary Model. We train different objects separately in several images with multiple aspects and camera viewpoints to find the best key word points for recognition. Subsequently, we verify the extracted features of the training images after classification of the feature sets. These key word points are applied to the regions based on visual feature point analysis. The performance measure of the proposed algorithm is analyzed with respect to available benchmark data and we obtain very satisfactory and competitive results. This has great potentials in the field of problem solving integrating vision and pattern recognition with more robustness and variability, with exciting opportunities to explore in near future.

References

1. Learning spatio-temporal representations for action recognition: a genetic programming approach, IEEE Trans. Cybern. **46**(1), November 2016
2. Xiao, F., Lee, Y.J.: Track and segment: an iterative unsupervised approach for video object proposals (2016)
3. Wu, Y., Lim, J., Yang, M.: Object tracking benchmark. IEEE Trans. Pattern Anal. Mach. Intell. **37**(9), 1837–1838 (2015)
4. He, S., Yang, Q., Lau, R.W.H., Wang, J., Yang, M.-H.: Visual tracking via locality sensitive histograms. In: Proceedings of IEEE Conference Computer Vision Pattern Recognition, pp. 2427–2434 (2013)
5. Zhang, K., Zhang, L., Yang, M.-H.: Real-time compressive tracking. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7574, pp. 864–877. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33712-3_62
6. Babenko, B., Yang, M.-H., Belongie, S.: Robust object tracking with online multiple instance learning. IEEE Trans. Pattern Anal. Mach. Intell. **33**(7), 1619–1632 (2011)

7. Li, H., Shen, C., Shi, Q.: Real-time visual tracking using compressive sensing. In: CVPR, pp. 1305–1312 (2011)
8. Hare, S., Saffari, A., Torr, P.H.S.: Struck: structured output tracking with kernels. In: Proceedings IEEE International Conference Computer Vision, pp. 263–270 (2011)
9. Kalal, Z., Matas, J., Mikolajczyk, K.: P-N learning: bootstrapping binary classifiers by structural constraints. In: Proceedings of IEEE Conference Computer Vision Pattern Recognition, pp. 49–56 (2010)
10. Li, G., Wang, Y., Shu, W.: Real-time moving object detection for video monitoring systems. In: International Symposium on Intelligent Information Technology Application (2008)
11. Xu, L., Qi, F., Jiang, R.: Shadow removal from a single image. In: Proceedings of IEEE International Conference on Intelligent Systems Design and Applications, pp. 1049–1054 (2006)
12. Avidan, S.: Support vector tracking. IEEE Trans. Pattern Anal. Mach. Intell. **26**(8), 1064–1072 (2004)
13. Grabner, H., Grabner, M., Bischof, H.: Real-time tracking via on-line boosting. In: Proceedings of British Machine Vision Conference, pp. 6.1– 6.10 (2006)
14. Sevilla-Lara, L., Learned-Miller, E.: Distribution fields for tracking. In: Proceedings of IEEE Conference Computer Vision Pattern Recognition, pp. 1910–1917 (2012)