International Journal of Computer Science and Informatics

Volume 2 | Issue 2 Article 8

October 2012

A Comparative Analysis of Kernel-Based Target Tracking Methods using Different Colour Feature Based Target Models

Nirja Magoch Thakur
Department of Computer Engineering, Defence Institute of Advanced Technology, Pune, India, nirja@rde.drdo.in

MM Kuber
Department of Computer Engineering, Defense Institute of Advanced Technology, Pune, mmkuber@diat.ac.in

Recommended Citation

Thakur, Nirja Magoch and Kuber, MM (2012) "A Comparative Analysis of Kernel-Based Target Tracking Methods using Different Colour Feature Based Target Models," *International Journal of Computer Science and Informatics*: Vol. 2 : Iss. 2 , Article 8.
DOI: 10.47893/IJCSI.2012.1074
Available at: https://www.interscience.in/ijcsi/vol2/iss2/8

A Comparative Analysis of Kernel-Based Target Tracking Methods using Different Colour Feature Based Target Models

Nirja Magoch Thakur & M.M.Kuber

Department of Computer Engineering, Defence Institute of Advanced Technology, Pune, India
E-mail : nirja@rde.drdo.in, mmkuber@diat.ac.in

Abstract - An effective target modeling is the root of a robust and efficient tracking system. Color feature is widely used feature space for target modeling in real time tracking applications because of its computational efficiency and invariance towards change in shape, scale and rotation. The effective use of this feature with kernel-based target tracking can lead to a robust tracking system. This paper provides a comparative analysis of the performance of three variants of kernel-based tracking system using color feature. The simulation results show that the target modeling using transformed background weighted target model will perform efficiently when initialized target has similar color feature with background while the combination of color-texture will be more accurate and robust when texture features are prominently present.

Keywords - *Mean shift, Target Modeling, Histogram, Bhattacharya coefficient, RGB, Local Binary Pattern.*

I. INTRODUCTION

Real-time and automatic moving target tracking is a key technique in many vision applications in robotics, target recognition and automatic security surveillance in military as well as civil applications [1]. The moving target tracking is realized mainly through video sequence analysis, prediction of the position of target and tracking trajectories of the target in successive video frames.

The robustness of a target tracking method has significant dependence on features that are used to model target. A discriminative representation of target is required for target modeling so that target can be specified with more confidence. The color probability density of target object is one of the most widely used features to model a target. However it has got a major drawback of not associating any structural information about pixels [7]. This shortcoming leads to divergence from actual target tracking when background and foreground have lot of similar color feature. This paper will cover a comparative simulation analysis of three methods that use kernel-based tracking i.e. the basic mean shift tracking to iteratively find the next probable position of target in video sequence using different models of target. RGB model that uses only color probability density function to specify a target. RGB-BW method uses a transformed background weighted target model. RGB-TXT model is realized for tracking that uses combination of texture and color probability densities to represent a target.

This paper is organized as follows. Section II will cover three target models used in comparative simulation. In section III, describes the mean shift tracking algorithm used for tracking target. Section IV will include the results of simulation analysis and observation made on the basis of results. Finally, conclusion and future work are given in Section V and VI.

II. TARGET MODELING

Locating and tracking the target object successfully is highly dependent on how target is represented or modeled. Target can be modeled using local features of object of interest like color, texture, edges (point of locally maximal contrast), lines, and corners (points where contrast is high along two directions). These local features offer some invariance to image changes caused by scene or illumination changes, improving detectability over time.

A. *RGB Model*

RGB method uses the color probability density function in all three channels Red, Green and Blue to model object that is to be tracked. Both target as well as target candidates are modeled in RGB color space using color probability distribution taken over a specified region. An isotropic kernel Epanechnikov [2] that assigns smaller weights to pixels farther from the centre is also used to increase the robustness of estimation. To calculate the color probability distribution, instead of adding one to the corresponding bin of pixel the weight of that pixel calculated by kernel is added.

The advantage of using color probability density for target representation is its computational efficiency and insensitivity to rotation and scaling.

B. RGB-BW : background weighted target model

RGB-BW (RGB with background weighted/transformed target model) method [3] associates background information with target modeling to reduce background interference in localization of target in successive frames. In cases where background have similar color features as present in target the discrimination on the basis of only RGB density become inefficient. This leads to target loss during tracking in cases where target is not initialized (selected) accurately. A transformed background model is calculated by modeling background using color probability density of background region and then using it to transform target model so as to suppress the effect of background color features in target model. So even if certain features of background are included during target selection, the use of transformed target model will nullify the effects of those features during target modeling.

Let $\{b_k\}_{k=1,2 \ldots n}$ is color probability density of background region. The least non-zero value (i.e. $\{b\}$) is found out from this probability distribution and used to calculate a weighted background model. The transformed background model is calculated by using this coefficient

$$P_k = \left\{ \min\left\{ \left(\frac{b}{b_k} \right), 1 \right\} \right\} \quad (1)$$

This coefficient will reduce the weight of all color features which are having higher values of probability densities in actual background model. The result is basically a transformed background model which will reduce all prominent features of background present in target.

C. RGB-TXT : combined texture and color target model

RGB-BW method's performance is satisfactory as long as the background is not having much complexity like moving objects, improper lighting conditions and less color information. In such conditions a more robust target representation is required. RGB-TXT method overcome the limitation of above mentioned target models by associating structural information of pixels using texture feature with color to represent a target and hence provide more specific target model. To extract texture information from frame Local Binary Pattern (LBP) operator is used and target model is formed by combined histogram using color and texture [4]. LBP operator labels each pixel with its corresponding texture value. The use of rotation invariant version of LBP operator provide unique labels to only uniform patterns which share the common characteristics of having total at most two transitions from one to zero or from zero to one[5]. Other non-uniform patterns are put into miscellaneous category and hence are given same label. Each uniform pattern is the representative of a local primitive like spots, lines, edges, corners etc. The patterns which represent lines, edges, corners are only considered for histogram creation with color features.

III. MEAN SHIFT TRACKING

Mean shift tracking algorithm [6] is a popular one due to its simplicity and efficiency among various object tracking methods. It is a nonparametric density estimator which iteratively computes the nearest mode of a sample distribution. Basically the algorithm involves the computation of an offset from the current location y_0 of target to a new location y_1 by performing iterations. The color histogram is used to represent the target because of its robustness to scaling, rotation and partial occlusion.

Let the distribution of the target object q measured around center position y_0 i.e. the position of target in previous frame. Let p is the candidate target model distribution. It is assumed that in subsequent frames target does not change much in position so candidate target model is represented by color probability distribution measured over an area expanded by some constant factor around previous position of target.

- To check how well p matches q the target distribution, a similarity metric e.g. Bhattacharyya coefficient is used. The basic idea is to minimize the distance between the two distributions or alternatively maximizing Bhattacharyya coefficient. The coefficient is calculated as:

$$\rho\,[p, q] = \sum_{l=1}^{L} \sqrt{p_l \bullet q} \quad (2)$$

- Estimate the objects next location y_1 using

$$y_1 = \frac{\sum_{i=1}^{n} x_i w_i}{\sum_{i=1}^{n} w_i} \qquad (3)$$

Where n is the total number of pixels and w_i is the weight corresponding to pixel. This equation (3) is a measure of weighted mean of probability distribution.

- Measure the color distribution around the new position and also check for convergence conditions i.e. minimum shift/offset and maximum number of iterations. If offset is less than minimum shift considered (by default equal to 1), color distributions are iteratively calculated by reinitializing until we find a better match (or reaches maximum number of iterations).
- If the distribution at y_1 matches q well enough (similarity matrix is nearly equal to 1) the target object has found. Otherwise set y_0 to y_1 and repeat the mean shift calculation procedure.

IV. SIMULATION ANALYSIS AND RESULTS

Tracking results using the models described in Section II are studied using Matlab simulation (Matlab version 7.0.9.529(R2009b)). Three cases are presented here where results are verified using Performance Evaluation of Tracking and Surveillance (PETS) video data sets.

The texture is characterized by using Matlab function Entropy which is statistical measure to calculate the randomness and used for texture analysis. The parameters used for the comparison are: Average number of iterations required for mean shift convergence, Computational time and Similarity using Bhattacharya coefficient. The conclusion is based on both statistical as well as visual results. Visual results are also included at the end of paper (Ref. Figure 4, 5, 6 and 7).

Case-1: Target is not initialized properly; target object has got lot of background information.(Ref. Fig. 4,5)

Texture characteristics: Entropy: 6.297

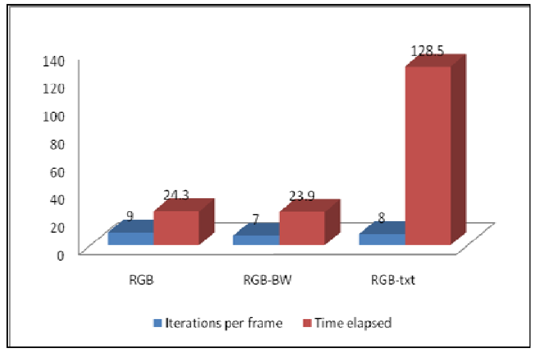

Fig. 1(a) : Simulation results :Case 1

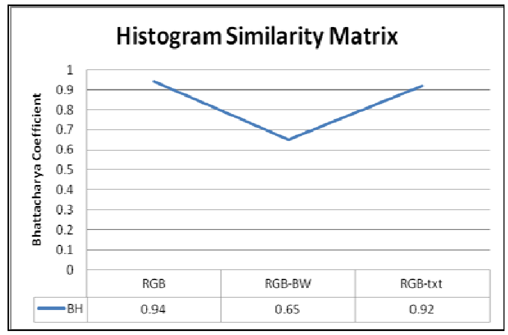

Fig. 1(b) Simulation results :Case 1

Figure 1(a) show that RGB-BW method takes less average no. of iterations to converge and least computation time as compared to RGB-TXT and RGB method.

Results presented by figure 1(b) shows that on an average similarity of target model and target candidate model is found the most in RGB and RGB-TXT. But the divergence is more in RGB as per the results showed in Fig 1(a). (As no. of iterations are maximum for convergence). Hence chances of target loss are more in RGB.

So we can choose either RGB-BW or RGB-TXT of modeling target. RGB-TXT will provide us with more accurate and close results as compared to RGB-BW but at the cost of computation load.

Case-2: Image (video frame) has got sufficient texture and color information.

Texture Characteristics: Entropy: 7.9686

Figure 2(a) shows that RGB-TXT method requires less number of iterations to converge than RGB, but more than RGB-BW. Computational time is almost same in all methods (5-6 seconds). RGB-TXT will consume slightly more time as compared to other two methods.

As per texture and color statistics, both texture and color features are prominent in the frame.

Figure 2(b) indicates that RGB-TXT will generate more similar target candidate model, hence more discriminative as compared to RGB and RGB-BW.

Hence the RGB-TXT algorithm will best perform among the three methods.

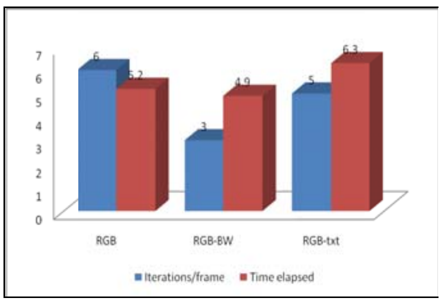

Fig. 2(a) Simulation results :Case 2

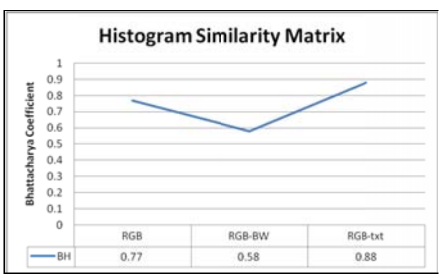

Fig. 2(b) Simulation results : Case 2

Case-3: Tracking a person inside shopping mall (complex background with improper lighting, people moving in background) with prominent texture(Ref. Fig. 6,7)

Texture Characteristics: Entropy: 6.9233

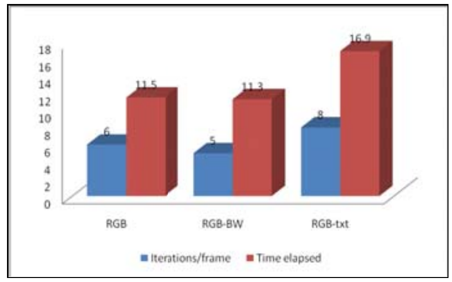

Fig. 3(a) Simulation results :Case 3

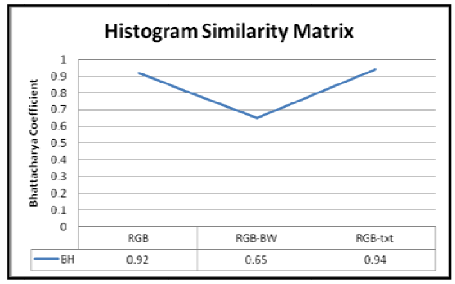

Fig.3(b) Simulation results:Case 3

Figure 3(a) indicates that the both RGB and RGB-BW methods are having same computational time and convergence rate as compared to RGB-TXT method.

As the entropy in frame is higher, texture features are prominent; hence RGB-TXT method will perform best in such scenario. Figure 3(b) indicates that the similarity matrix has high value for RGB-TXT as compared to RGB and RGB-BW.

V. CONCLUSION

The RGB-TXT feature based method will perform best if sufficient texture quality (as per statistics like Entropy i.e. the non-uniformity of texture as well as statistics derived from Gray Level Co-occurrence Matrix like contrast, correlation) is present in the frames. Though the computational time in RGB-TXT method is more as compared to RGB and RGB-BW, but results will be more accurate and the chances of deviation from the target object will also be less. The RGB-BW method can be used instead of RGB-TXT when texture features are not sufficient as it takes less number of iterations for localization of target model and also it is more computationally efficient. RGB based method will give the most similar target match on the basis of Bhattacharya coefficient but the convergence rate is not good i.e. no. of iterations are quite high. This can leads to deviation from the actual target i.e. false target tracking. Results show that there is always a tradeoff between accuracy and computational speed. But for critical real time tracking applications where both speed and computational efficiency are required, an adaptive real time tracking system is required which use a pre-analysis of few starting frames to predict the prominent feature and switch to appropriate algorithm based on the results of analysis.

VI. FUTURE WORK

For real time tracking applications, an adaptive tracking algorithm is required to be developed, which will adapt to appropriate target modeling and tracking algorithm on the basis of prominence of feature. The prominent feature will be used for target representations. There is scope for incorporation of machine learning techniques to provide tracker with intelligence to make decision so as to adapt to most suitable modeling and tracking algorithm.

REFERENCES

[1] M. J. Swain, and D. H. Ballard, "Color indexing," International Journal of Computer Vision, 7(1), pp. 11-32, 1991.

[2] Jifeng Ning, Lei Zhang1, David Zhang and Chengke Wu, "Robust Mean Shift Tracking with Corrected Background-Weighted Histogram".

[3] D. Comaniciu, V. Ramesh, and P. Meer. "Kernel-based object tracking". IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(5):pp.564-577, May 2003.

[4] Jifeng Ning, Lei Zhang, David Zhang, "Robust Object Tracking using joint color-texture histogram", International Journal of Pattern Recognition and Artificial Intelligence Vol. 23, No. 7,pp. 1245–1263,2009.

[5] Timo Ojala, Matti PietikaÈ inen, Senior Member, IEEE, and Topi MaÈenpaÈa, "Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns", IEEE Transactions on pattern analysis and machine intelligence, Vol. 24, No. 7, July 2002

[6] D. Comaniciu, V. Ramesh and P. Meer, "Real-Time Tracking of Non-Rigid Objects using Mean-Shift," Computer Vision and Pattern Recognition, Vol. 2, pp. 142-149, 2000.

[7] An Zhao, Eveleigh, "Robust Histogram-based Object Tracking in Image Sequences", IEEE Computer Society of India, pp.45-42, 2007.

Fig. 4 : Performance of tracker using RGB Target Model.

Fig. 5: Performance of tracker using RGB-TXT Target Model

Fig. 6 : Performance of tracker using RGB-BW Target Model

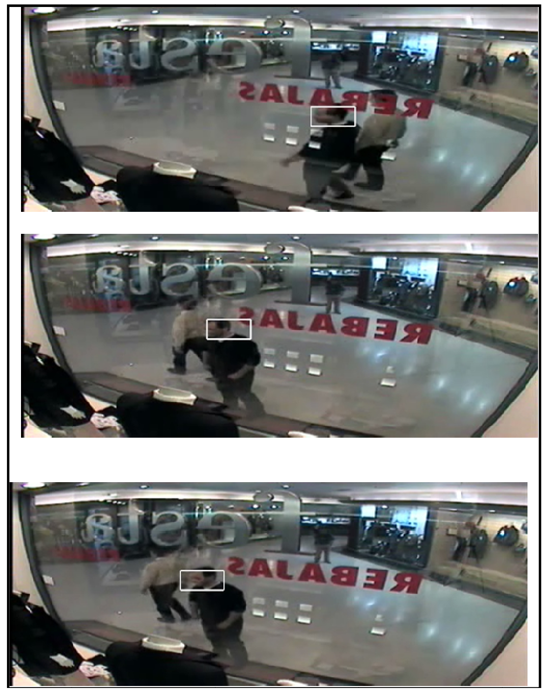

Fig. 7: Performance of tracker using RGB-TXT Target Model

◈◈◈