



Field programmable Gate Array based Real Time Object Tracking using Partial Least Square Analysis

D. Somasundaram^{*}, N. Kumaresan^b, S. Vanitha^c

^a Department of Biomedical Engineering, Sri Shakti institute of Engineering and Technology, Coimbatore - 641062, Tamil Nadu, India

^b Department of ECE, Anna University, Regional Center, Coimbatore, Tamil Nadu, India

^c Department of ECE, PSG Institute of Technology and Applied Research, Coimbatore - 641 062, Tamil Nadu, India

* Corresponding Author: somgce@gmail.com

Received : 12th August 2020, Accepted : 30th August 2020

Abstract: In this paper, we proposed an object tracking algorithm in real time implementation of moving object tracking system using Field programmable gate array (FPGA). Object tracking is considered as a binary classification problem and one of the approaches to this problem is that to extract appropriate features from the appearance of the object based on partial least square (PLS) analysis method, which is a low dimension reduction technique in the subspace. In this method, the adaptive appearance model integrated with PLS analysis is used for continuous update of the appearance change of the target over time. For robust and efficient tracking, particle filtering is used in between every two consecutive frames of the video. This has implemented using Cadence and Virtuoso software integrated environment with MATLAB. The experimental results are performed on challenging video sequences to show the performance of the proposed tracking algorithm using FPGA in real time.

Keywords: Object Tracking, Feature Extraction, Adaptive Appearance Model, Partial Least Square Analysis, FPGA.

1. Introduction

Object tracking in video processing follows the segmentation step and is more or less equivalent to the 'recognition' step in the image processing. Detection of moving objects in video streams is the first relevant step of information extraction in many computer vision applications, including traffic monitoring, automated remote video surveillance, and people tracking. In this, the appearance of the object may change due to various factors such as motion, angle, occlusion, etc. Especially in real time tracking, these challenges are more when compared to offline tracking. To avoid these difficulties in real-time object tracking, efficient image segmentation algorithms

are needed. So, to overcome these problems, this paper proposed an efficient method for tracking in real time which adapts to the different appearance change of the moving object. Many researchers have been undertaken in the area of video object tracking in offline and in real time. Many theoretical and experimental results have been used for various objects tracking analysis in the video. A video consists of multiple numbers of frames and each frame has large amount of information; hence video tracking is a time-consuming process. So, the efficient methods can be used to extract the features in the video sequences consuming less time.

In recent years, many ways are used to extract features from the foreground and background of the object in the video for accurate tracking. The features are extracted from a densely sampled grid structure leads to a high degree of multicollinearity. But the use of high dimension features leads to computational complexity. To reduce these high dimensionality features to low dimensionality features, PLS analysis is used which preserves discriminative information to project the data onto a lower dimensional subspace. After extracting the low dimensional features from the foreground and background of the object, the tracking process starts from the first frame of the object in the video. For real time tracking, the appearance of the object changes over time, so adaptive appearance model is implemented to adaptively change to the appearance change of the object during tracking. Then the particle filtering is applied in between every two consecutive frames for robust tracking. The particle filters may be a serious alternative for real-time applications classically approached by model-based Kalman filter techniques.

The rest of the paper is organized as follows: Section 2 describes the literature survey on real time object tracking. Section 3 explains about the partial least square (PLS) analysis which is low dimension reduction technique. Section 4 describes how to adapt to appearance change of the object over time. Section 5 deals with the use of real time particle filtering method. Section 6 shows the experimental setup and the challenging performances on different tracking sequences.

2. Related Work

In this section we focus on various models and techniques for video-based object tracking. Traditionally real-time object tracking have been achieved using a new search technique for finding the best match among feature vectors of the reference block and feature vectors of the search area in the wavelet domain[4]. Although searching the feature vectors in wavelet domain is a complex process in real time tracking, Abu-Bakar, S.A.R. et al. [5], presents an efficient technique for real-time tracking of a single moving object in terrestrial scenes based on the linear prediction (LP) solved by the maximum entropy method (MEM). The selection of features based on color, color frequency is convenient to extract than by wavelet domain. So Yasushi Yagi et al. [9] presented a mean-shift tracking algorithm to an adaptive tracker by selecting reliable features from color and shape-texture cues according to their descriptive ability. In these methods the selection of feature vectors are in high dimension. So tracking the objects in the high-dimensional feature space is not only computationally expensive but also

functionally inefficient. Selecting a low-dimensional discriminative feature set is a critical step to improve tracker performance. Bohyung Han et al. proposed that multiple heterogeneous features are brought together, and likelihood images are constructed for various subspaces of combined feature space and the features are extracted by principal component analysis (PCA) based on those likelihood images. Yanbin Han et al.[11] presents that the real time object tracking by Camshift combining color information and improved LBP. The color based Camshift is suitable for tracking targets in simple cases, it fails to track objects in more complex situations. Similarly, Emami. E et al.[14] proposed an improved Cam shift algorithm to cope with Camshafts tracking problems. Tracking multiple objects in real time is a real complex thing. Qin, Wan et al.[10], proposed to track multiple objects in a real-time visual surveillance system. Abdel-Hadi .A et al.[12], presented a method for real time tracking of the moving target objects which is characterized by a color probability distribution. Patra, D. et al. [13], proposed a new algorithm for the tracking of target object from the video based on segmentation and Kernel based procedure. The computational complexity becomes very high in those kernel-based techniques. The target localization problem is minimized using segmentation technique, instead of using mean shift tracking algorithm. Robust real-time tracking of non-rigid object is a challenging task in object tracking. So the integration of the particle filter with the main tracking algorithm leads to better computational efficiency in tracking. Recently, Qing Wang et al.[17] presents that the object can be tracked using partial least square (PLS) analysis with particle filtering. Particle filtering has proven very successful for non-linear and non-Gaussian estimation problems.

3. Partial Least Square (PLS) Analysis

Partial least square (PLS) analysis is a low dimension reduction technique as well as a wide class of methods for modeling relations between sets of observed variables which is related to principal components regression. instead of finding hyper planes of minimum covariance between the response and independent variables, it finds a linear regression model by projecting the predicted variables and the observable variables to a new space by means of latent variables. PLS analysis is used to reduce the high dimensional feature vectors into low dimensional feature vectors which are extracted by means of latent variables in the subspace. The use PLS in object tracking is very convenient for real time because of its low dimensionality reduction technique.

3.1 Feature Extraction

It is mandatory to develop an efficient method that continually evaluates and updates the set of features used for real time tracking. The features used for tracking need to be only locally discriminative for real time and the object need to be clearly separable from its current surroundings. For tracking purposes, different types of features like color, shape, motion and texture can be used to track objects. To track the target object, the object must be distinguishable from its background. In the learning stage, the target object appearance 'i' is denoted in the form

of an image window. This image window is decomposed into overlapping blocks and a set of features is extracted for each block to construct a feature vector. For each object appearance 'i', the feature vectors are extracted. To capture texture in the object appearance we extract features by means of co-occurrence matrices. The Co-occurrence features are useful for the detection of people in motion. Once the feature extraction process is performed for all blocks inside an image window, features are concatenated creating a high dimensional feature vector. For each appearance represented by an image window, the features are extracted and the PLS analysis method is applied to reduce the dimensionality of the feature vectors in the subspace.

3.2 PLS For Dimension Reduction

The basic idea of PLS is to construct new predictor variables called latent variables, as linear combinations of the original variables are stated in a matrix \mathbf{X} of descriptor variables (features) and a vector \mathbf{Y} of response variables (class labels). Consider the general linear PLS algorithm to model the relation between two data sets (blocks of variables) and both the \mathbf{X} and \mathbf{Y} data are represented to new spaces. PLS[18] is used to find the fundamental relations between two matrices (\mathbf{X} and \mathbf{Y}), i.e. a latent variable approach to modeling the covariance structures in these two spaces. A PLS model will try to find the multidimensional direction in the \mathbf{X} space that explains the maximum multidimensional variance direction in the \mathbf{Y} space. PLS analysis is particularly suited when the matrix of predictors has more variables than observations, and when there is multicollinearity among \mathbf{X} values.

Let $\mathbf{X} \in R^N$ and $\mathbf{Y} \in R^M$ be N-dimensional space of feature vectors represents the first block and M-dimensional space of class labels represents the second block. PLS analysis correlates the relations between these two blocks. Let the number of samples be n . PLS decomposes the $(n \times N)$ matrix of zero-mean variables \mathbf{X} and the $(n \times M)$ matrix of zero-mean variables \mathbf{Y} into the form

$$\begin{aligned}\mathbf{X} &= \mathbf{T}\mathbf{P}^T + \mathbf{E} \\ \mathbf{Y} &= \mathbf{U}\mathbf{Q}^T + \mathbf{F} \quad (1)\end{aligned}$$

Where \mathbf{T} and \mathbf{U} are score and component matrices, \mathbf{P} and \mathbf{Q} are loading matrices, and \mathbf{E} and \mathbf{F} are the error terms. The two blocks of variables \mathbf{X} and \mathbf{Y} are decomposed by computing the weight vectors $\mathbf{W} = \{w_1, w_2, \dots, w_i\}$. The weight vectors are computed by using likelihood function. The covariance between the two latent vectors t_i and u_i is

$$[\text{Cov}(t_i, u_i)]^2 = \max [\text{Cov}(\mathbf{X}\mathbf{w}_i, \mathbf{y})]^2 \quad (2)$$

where t_i and u_i are the i -th columns of \mathbf{T} and \mathbf{U} , respectively.

The matrix \mathbf{X} and vector \mathbf{y} are deflated by subtracting their rank-one approximations based on the extraction of the latent vectors t_i and u_i . This process is repeated until the desired number of latent vectors had been extracted.

$$\mathbf{X} \leftarrow \mathbf{X} - \mathbf{t}_i \mathbf{p}_i^T$$

$$\mathbf{Y} \leftarrow \mathbf{Y} - \mathbf{u}_i \mathbf{q}_i^T \quad (3)$$

Where \mathbf{p}_i and \mathbf{q}_i are the i -th columns of \mathbf{P} and \mathbf{Q} loading matrices. With the resulting latent vectors, the dimensionality of the object is reduced in the learned subspace.

3.3 Target Representation

The PLS analysis [14] is used to analyse the mutual relationship between the object appearance and class label of the object for dimensionality reduction and also for classification. In this paper, object tracking is defined as a classification problem which labels the target as positive sample and background as negative sample. This method considers the two classes, feature vectors and class label for tracking the object. In the surveillance videos, the target object is located in the first frame of the video. After targeting the object in the first frame of the video, the target area is considered as positive sample \mathbf{x}_1 , where the feature of the warped image is specified by some state parameter. The negative samples are collected around the target area in the subspace randomly.

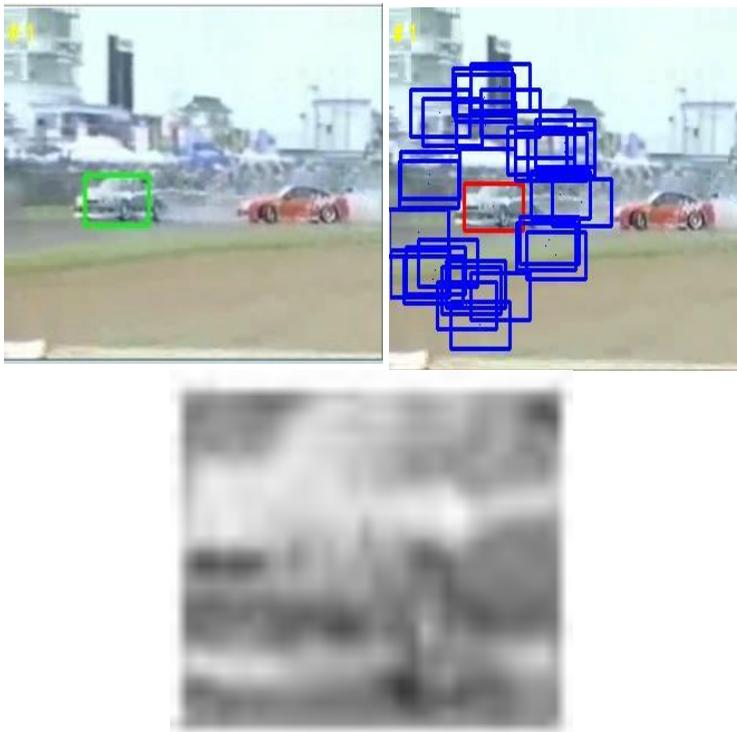


Figure 1. Samples in the frame (a) Target object is marked by a green polygon from which a positive sample can be collected. (b) Negative samples are collected from the blue polygons. (c) Target object

From the obtained positive and negative samples, PLS analysis is used to determine an appearance model of the target object.

The latent features are extracted from the target of the object and these data are decomposed. For decomposition of the target object first the weight vectors are computed. After computing the weight matrices, the initial appearance of the target is modeled as A_1 .withthe help of latent feature space a target object can be easily discriminated from the background.The recognition of the target from the background depends on the feature variables. These feature variables are based on the pixel intensity of the object, which makes no difficulty in discriminability between the target and the background. For example, if each feature variable has the intensity of one pixel and the feature vector represents the group of pixel intensities in an object region, a subspace can be learned by PLS analysis by means of positive and negative samples.

4. Adaptive Based Appearance Model

In real time tracking, one major problem is that the tracking continuous for long period of time. So there occurs a computational complexity that the appearance of the objects changes. In offline video tracking, the appearance change is not expected to be large. But in online video tracking, there occurs a drastic change in the appearance of the moving objects. So here proposed an adaptive appearance model for tracking, i.e. it adapt to the current situation. By the adaptive appearance model, the appearance of a target object may be represented by means of multiple appearance models $\bar{A} = \{A_1 \dots A_k\}$, where k is the number of appearances of the object.

4.1 Distance Minimization

The first appearance model A_1 is estimated by PLS analysis in the first frame of the object. For the further frames the appearance may change due to motion, occlusion, etc. In the next frame, the appearance will change and the target is located by calculating the minimum distance between the target in original region and the target in the new region. The distance between a target \mathbf{x} and the learned appearance model set A_k is defined by

$$d = \min\{d_i \mid i = 1, \dots, k\} \quad (4)$$

Where d_i is the distance between the target \mathbf{x} and the appearance model A_i

$$d_i = \left\| W_i^T (X - \bar{X}_i) - W_i^T (\bar{X}_i^p - \bar{X}_i) \right\|^2 \quad (5)$$

Where \bar{X}_i^p is the mean of the positive samples, \bar{X}_i is the mean of all the samples and $\|\cdot\|$ is the Euclidean norm. After calculating the minimum distance between the original target and the new target, the new appearance will be updated in place of A_1 .The updating of the new appearance of the object for tracking in real time is based on the mean of positive samples, mean of positive and negative samples and the weight matrix.

5. Filtering Method

The filtering problem is the process of estimating a system's current state which is hidden, based on past and current observations. This is represented by the probability density function $p(x_t | x_{t-1}, z_{0:t})$. For visual tracking sequences, the state can be position, velocity, orientation, or scale of an object.

5.1 Particle Filtering

For real time tracking, kalman filters are commonly used. The problem in Kalman filters is that they represent the state of the system using only single Gaussians. But the particle filters can keep track of as many hypotheses as there are particles, so if new information shows up that causes you to shift your best hypothesis completely. So for robust and continuous tracking in real time particle is more efficient. Generally, by the use of particle filters in object tracking, all samples can be updated whenever new sample information arrives. For real time situations, there occurs a difficulty in the update of the new information cannot be completed. This leads to computational complexity in tracking. The majority of the filtering methods deal with this problem by skipping the information that arrives during the update of the filter.

5.2 Real Time Particle Filtering (RTPF)

The real time particle filter (RTPF) considers all the information's by distributing the samples among the observations within an update window. RTPF weights the different samples sets in the target object by means of likelihood function. The likelihood function is used to discriminate between the original target and the new appearance of the target. It determines the quality of the image during tracking. It assigns larger weight to the target which best matches according to the observation model. This method focuses on samples on the most valuable observations. This "virtual sample set", or belief, is a mixture of the distributions represented in it.

The optimal belief is the belief we would get if there was enough time to compute the full posterior probability within the update window. The optimal belief $Bel_{opt}(x_{t_k})$ at the end of an estimation window results from iterative application of the Bayes filter updating on each observation.

$$Bel_{opt}(x_{t_k}) \propto \int \dots \int \sum_{i=1}^k p(z_{t_i} | x_{t_i}) p(z_{t_i} | x_{t_{i-1}}, u_{t_{i-1}}) Bel(x_{t_0}) dx_0 \dots dx_{t_{k-1}} \quad (6)$$

Where $Bel(x_{t_0})$ denotes the belief estimated in the previous window.

RTPF computes the optimal weights of the mixture distribution at the end of each estimation window. This is done by gradient descent method using the Monte Carlo estimates of the gradients. The resulting weights are used to generate samples for the individual sample sets of the next estimation of the window. Between every two consecutive frames, RTPF and affine transformation is applied and the target state at time's 's' is denoted by $\mathbf{S}_t = (x_t, y_t, s_t, \alpha_t, \theta_t)$ are x,

y translations, scale, aspect ratio, and in-plane rotation angle, respectively. Hence the real time particle filtering is efficiently implemented for real time object tracking especially in the case the challenging factors like illumination, occlusion, motion, etc.

6. FPGA Architecture for implementation

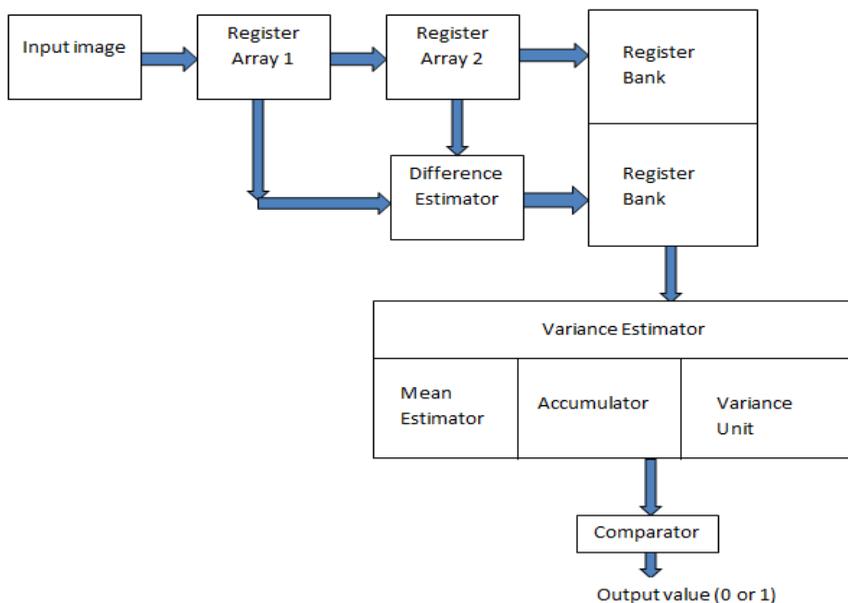


Figure 3. FPGA Architecture for proposed method

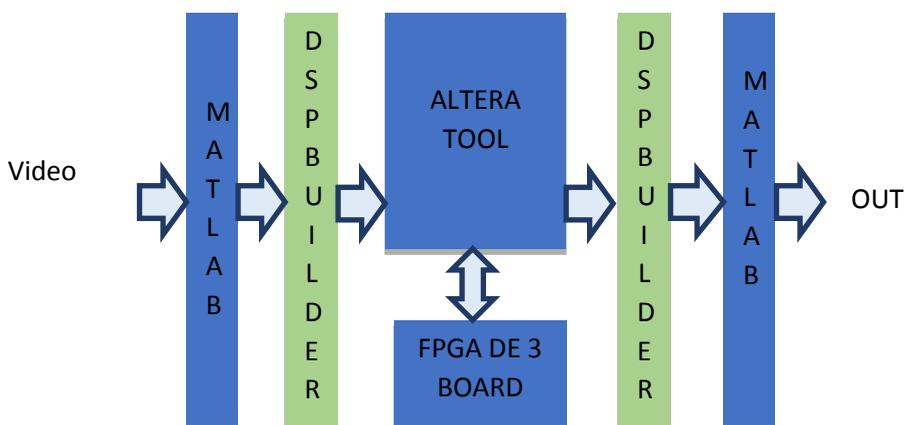


Figure 3. Interfacing FPGA and MATLAB environment

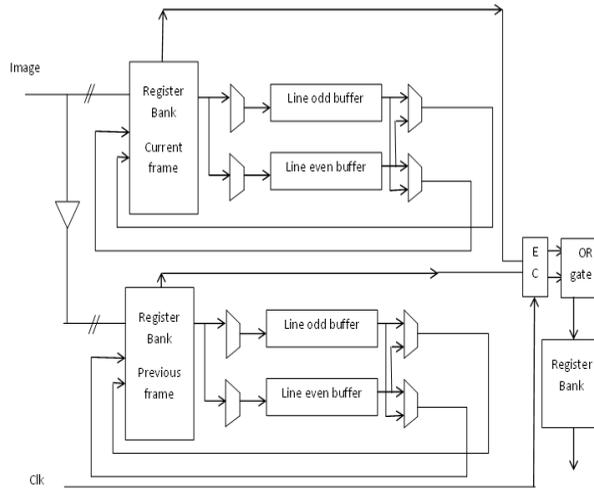


Figure 4 .High level architecture for frame Read/Write operation

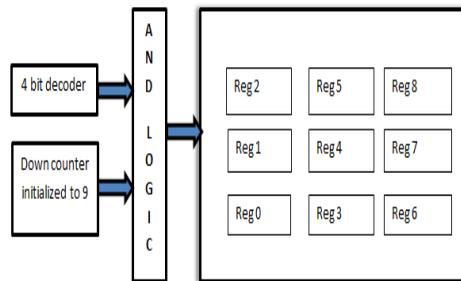


Figure 5. Register Bank

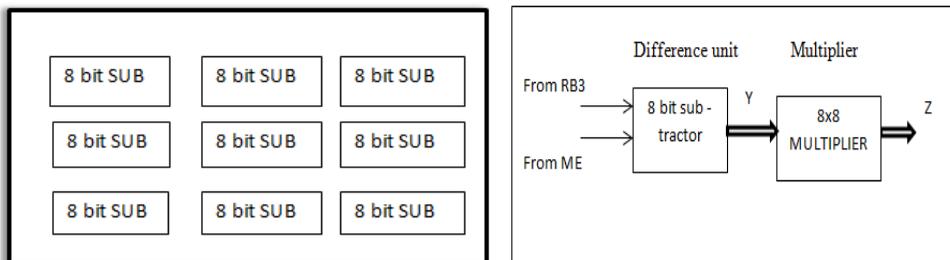


Figure 6. Subtraction unit

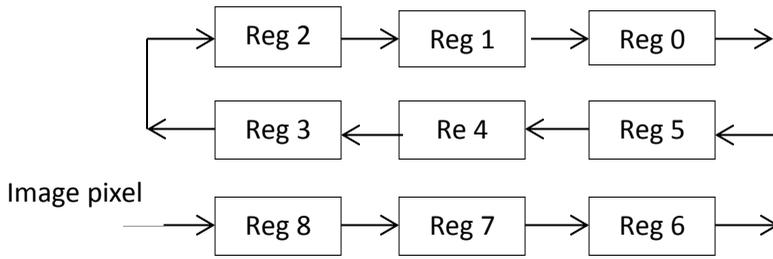


Figure 7. Parallel (PI)

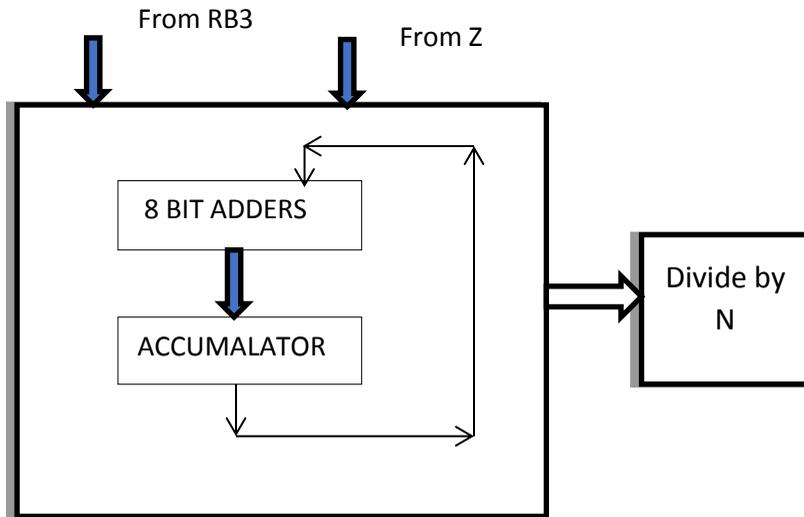


Figure 8. Mean / variance Estimator (ME)

6. Experimental Results

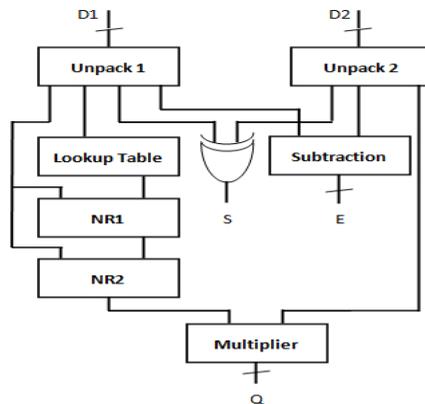


Figure 9. Iteration unit

In this section we will demonstrate some experimental results on several challenging video sequences. We have implemented the proposed method in MATLAB on Microsoft windows 7 platform. Here illustrated the performance of proposed algorithm on different challenging sequences. In each tracking sequences the target object is labeled in the first frame. Each image sequence of the target object is normalized to a 32×32 patch. In the first frame one positive sample and 25 negative samples are taken by PLS analysis to initialize an appearance model. The weight vectors, p , is set to 15, and the maximum number of appearance models, K is set to 10. The tracking area is drawn by a rectangle window which has the dimensional state vector $S = [x, y, w, h, \theta]$, where (x, y) represents the position of the tracking window, (w, h) represents the width and height of the tracking window, and θ represents the rotation angle of the tracking window.

To demonstrate the robustness and efficiency of the proposed method, we have tested the tracking algorithm on many challenging image sequences. These sequences consists of many difficult scenarios such as changes in appearance, pose variations, shadowing, occlusion, scale changes, cluttered backgrounds, and quick motion resulting in motion blur. Figure 2-6 shows the tracking results for different sequences. The tracking result for the surfing sequences are shown in figure 2. In these surfing sequences, the targeted object is being tracked for long duration. It shows the change in appearance and our algorithm tracks the object efficiently without any distortion. Figure 3 shows the tracking sequences for the drifting car sequences. In these sequences, the car drifts gradually and it shows the drastic change in the appearance of the target from one frame to another and it adapts to the conditions. Also there is a motion blur in the sequences. But there is no distortion in tracking the target in the video by our proposed algorithm. Figure 4 shows tracking results for the skating sequences. In these sequences, there occurs only less motion blur but the appearance change of the target is high. Our algorithm tracks the target without any distortion over long period of time. Figure 5 & 6 shows the tracking result for the people walking in the mall.

Parameters for comparison:

i. Object detection rate

The rate at which tracked objects are matched to the ground truth without reference to the labels assigned. OBR value varies between 0 and 1. For poor object detection OBR value is zero and for perfect matched objects ground truth is one.

$$OBR = \frac{1}{GTL} \sum_{g=0}^{GTL} \frac{GTM_g}{GTT_g} \text{-----}(1)$$

ii. Average Size Detection Rate

The rate at which the tracked object size differs from the matched ground truth. ASDR value varies between 0 and 1, where 0 means poor object size detection and 1 means accurate

object size detection. The sensitivity of the size detection rate can be varied if required, for the result presented sensitivity of ASDR is set at $\pm 20\%$.

$$ASDR = \frac{1}{L} \sum_{i=0}^L \frac{SDR_i}{TPM_i} \text{-----}(2)$$

iii. Label Tracking Detection Rate

The rate at which uniquely labeled objects matches the uniquely labeled ground truth. LTRD value varies between 0 and 1, 0 implies that the tracking algorithm has poor object matching to the unique labels and 1 means accurate object tracking with the same unique label and location between the tracked object and ground truth.

$$LTDR = \frac{1}{L_m} \sum_{i=0}^{L_m} \frac{TPM_i}{OAF_i} \text{-----}(3)$$

(iv). Non-Label Tracking Detection Rate

It is used to account for the tracked objects that are not matched. NTDR value varies between 0 and 1. If NTDR is equal to 1 this indicates that all the non-matched objects are tracked without affecting the matched objects. This situation occurs when objects are detected temporarily especially from reflective surfaces.

$$NTDR = \frac{1}{L_n} \sum_{i=0}^{L_n} \frac{FPM_i}{OAF_i} \text{-----}(4)$$

Table 1. Performance comparison for Real time video

	OBR	ASDR	NTDR	LTDR	Success rate (%)
MILT	0.968	0.857	0.108	0.953	97.4
VRT	0.987	0.834	0.201	0.843	88.1
BoostT	0.884	0.827	0.303	0.829	92.3
LIT	0.681	0.842	0.202	0.764	86.4
PLS	0.981	0.973	0.053	0.976	98.6

Table 1 gives the comparative results of OBR, ASDR, NTDR and LTDR based on the successive rate.

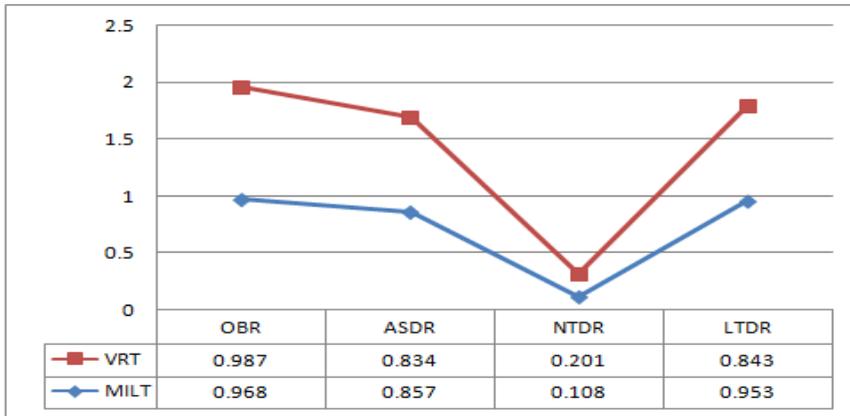


Figure 10. VRT Vs MILT

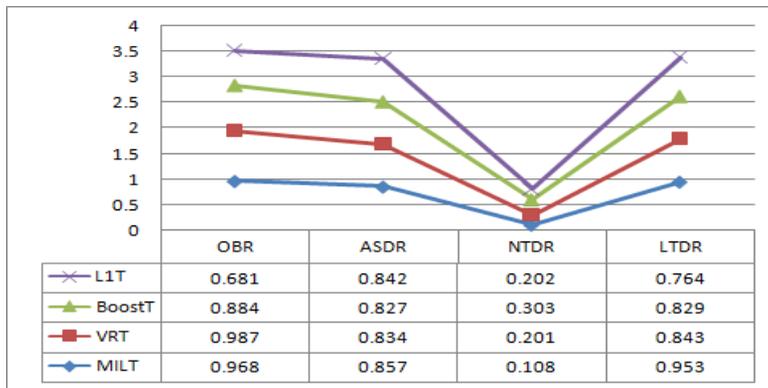


Figure 10. VRT Vs MILT Vs BOOSTT Vs L1T

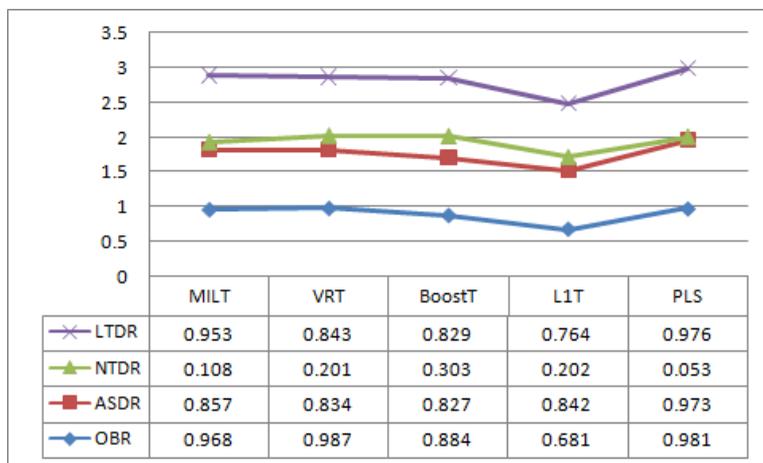


Figure 10. LTDR Vs NTDR Vs ASDR Vs OBR

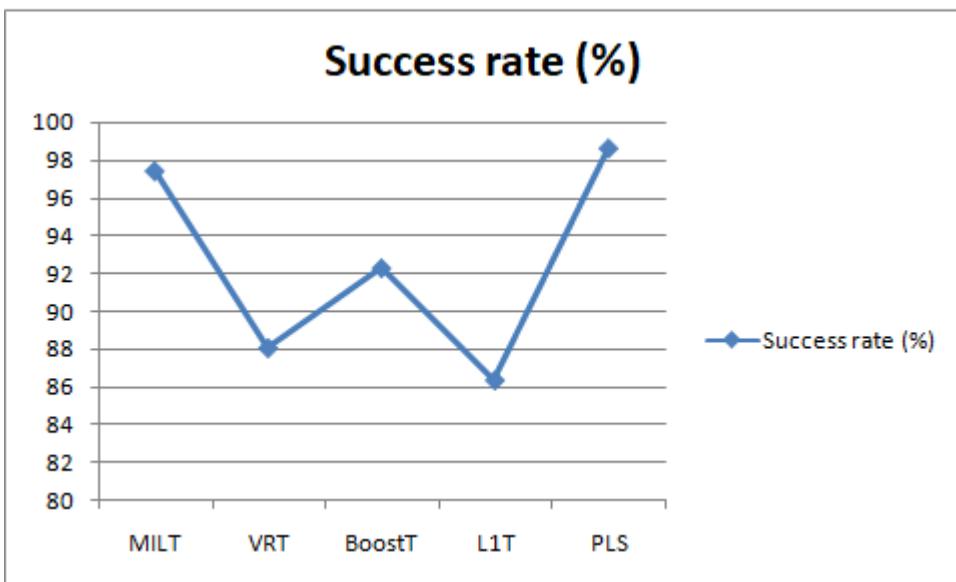


Figure 10. Success Rate

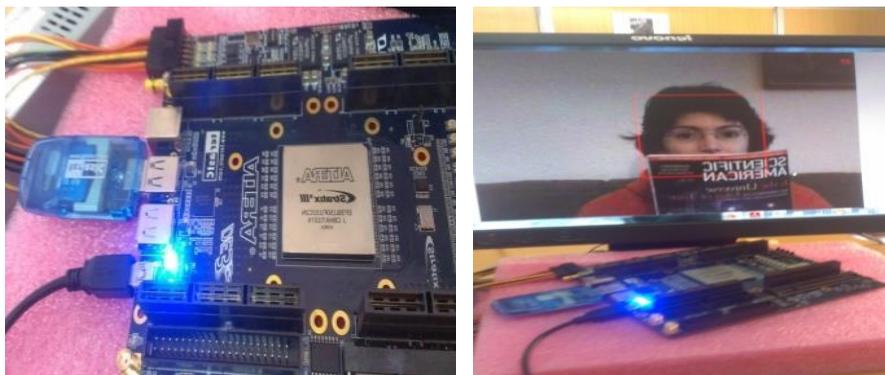


Figure 11. Altera DE3 interface and real time tracking

In the above Figure 11. It explains the Altera DE3 interface and real time tracking system.

7. Conclusion

In this paper, we presented a real time object tracking method by using PLS analysis along with real time particle filtering (RTPF) algorithm. In the proposed method, the PLS

algorithm is used to best discriminate between the target and background and also to adapt for the appearance change of the target. Real time tracking needs robust tracking algorithm for all challenging factors. So RTPF method is integrated with PLS analysis for robust tracking. Compared to the particle filtering method, this method has the advantage of not skipping any observations during filter update, because, particle filtering method skips out some of the observed samples over long period of time. The proposed algorithm shows that tracking objects in real time with high success rate and low error sequence and it is mainly applicable for real time surveillance system.

References

- [1] D. A. Ross, J. Lim, R. S. Lin, Yang, M. H. Incremental learning for robust visual tracking, *International journal of computer vision*, 77 (2008) 125-141.
- [2] Q. Li, Y. Yan, H. Wang, Discriminative weighted sparse partial least squares for human detection, *IEEE Transactions on Intelligent Transportation Systems*, 17 (2015) 1062-1071.
- [3] H. Medeiros, J. Park, A. Kak, Distributed object tracking using a cluster-based kalman filter in wireless camera networks, *IEEE Journal of Selected Topics in Signal Processing*, 2 (2008) 448-463.
- [4] M. Amiri, H. R. Rabiee, F. Behazin, M. Khansari, (2003) A new wavelet domain block matching algorithm for real-time object tracking, In *Proceedings 2003 International Conference on Image Processing*, IEEE, 3.
- [5] P. Y. Yeoh, S. A. R. Abu-Bakar, (2003) Accurate real-time object tracking with linear prediction method, In *Proceedings 2003 International Conference on Image Processing*, IEEE, 3.
- [6] Durand, J., & Hutchinson, S. (2003). Real-time object tracking using multi-res. critical points filters, In *2003 IEEE International Conference on Robotics and Automation*, IEEE, 2, 1682-1687
- [7] P. Lanvin, J. C. Noyer, M. Benjelloun, M. Yearly, Y. Zhai, (2005) Hybrid particle filtering for real time object tracking, In *Conference Record of the Thirty-Ninth Asilomar Conference on Signals, Systems and Computers*, IEEE, 761-764.
- [8] J. U. Cho, S. H. Jin, X. Dai Pham, J. W. Jeon, J. E. Byun, H. Kang, (2006) A real-time object tracking system using a particle filter, In *2006 IEEE/RSJ international conference on intelligent robots and systems*, IEEE, 2822-2827.
- [9] J. Wang, Y. Yagi, Integrating color and shape-texture features for adaptive real-time object tracking, *IEEE Transactions on Image Processing*, 17 (2008) 235-240.
- [10] W. Yaonan, W. Qin, Y. Hongshan, Effective method for tracking multiple objects in real-time visual surveillance systems, *Journal of Systems Engineering and Electronics*, 20 (2009) 1167-1178.
- [11] J. Yin, Y. Han, J. Li, A. Cao, (2009) Research on real-time object tracking by improved Cam Shift, In *2009 International Symposium on Computer Network and Multimedia Technology*, IEEE, 1-4.

- [12] A. Abdel-Hadi, (2010) Real-time object tracking using color-based Kalman particle filter. In The 2010 International Conference on Computer Engineering & Systems, IEEE, 337-341.
- [13] D. Chakraborty, D. Patra, (2010) Real time object tracking based on segmentation and distance minimization, In 2010 Annual IEEE India Conference, IEEE, 1-4.
- [14] E. Emami, M. Fathy, (2011) Object tracking using improved camshift algorithm combined with motion segmentation, In 2011 7th Iranian Conference on Machine Vision and Image Processing, IEEE, 1-4.
- [15] S. Nasrullah, D. A. Khan, (2011) A novel algorithm for real time moving object tracking, In 2011 International Conference on Image Information Processing, IEEE, 1-5.

Acknowledgements: NIL

Conflict of interest: NIL

About the License: © 2020 The Author. This work is licensed under a Creative Commons Attribution 4.0 International License which permits unrestricted use, provided the original author and source are credited.