STUDY OF MULTI-CUE OBJECT TRACKING IN VIDEO SEQUENCES

A THESIS

SUBMITTED TO THE DELHI TECHNOLOGICAL UNIVERSITY

FOR THE AWARD OF THE DEGREE OF

DOCTOR OF PHILOSOPHY

IN

Computer Science & Engineering

SUBMITTED BY
ASHISH KUMAR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
DELHI-110042 (INDIA)
October 2020

DELHI TECHNOLOGICAL UNIVERSITY

Certificate

This is to certify that the thesis entitled **"Study of Multi-Cue Object Tracking in Video Sequences"** being submitted by Ashish Kumar (Reg. No.: 2K16/PhD/CO/11) for the award of degree of Doctor of Philosophy to the Delhi Technological University is based on the original research work carried out by him. He has worked under our supervision and has fulfilled the requirements which to our knowledge have reached the requisite standard for the submission of this thesis. It is further certified that the work embodied in this thesis has neither partially nor fully submitted to any other university or institution for the award of any degree or diploma.

Dr. Gurjit Singh Walia
Supervisor
Scientist 'E'
Software Analysis Group
DRDO

Prof. Kapil Sharma
Supervisor
Professor & Head
Dept. of Information Technology
Delhi Technological University

Prof. Rajni Jindal
Head of the Department
Dept. of Computer Science & Engineering
Delhi Technological University

Declaration of Authorship

I hereby declare that all information in the thesis entitled "Study of Multi-Cue Object Tracking in Video Sequences" has been obtained and presented in accordance with academic rules and ethical conducts as laid out by Delhi Technological University. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

(Ashish Kumar)

Research Scholar

Acknowledgements

First and foremost, thanks to the *Almighty* for giving me strength and inspiration to carry out this research work. I owe a deep sense of gratitude to all his *Comprehensive Soul* whose *Divine Light* has enlighted my path throughout the complete journey of my research.

I would like to express my sincere and heartfelt thanks to my research supervisor **Prof. Kapil Sharma** for his valuable guidance, enthusiastic encouragement and persistent support. I am truly grateful from the core of my heart for his meticulous approach and wonderful assistance of his perspective and fruitful discussions on the research topics. His careful supervision and personal attention has given me a lot of confidence and enthusiam, during the different stages of my doctoral investigations. I invariably fall short of words to express my sincere gratitude for his patience and motivation.

I place on record my heartfelt gratitude and sincere thanks to **Dr. Gurjit Singh Walia** who has been my supervisor, advisor and mentor. He is the one whose expertise in the field is widely acclaimed. I thank him for his valuable advice, fruitful discussions, numerous suggestions, constructive criticism, and constant support during the course of my research. With great pleasure, I specially thank him for being kind and considerate towards me and allowing me to happily encroach upon his time, even at odd hours. It was a felicitous and unforgettable experience to work under his intellectual and revered guidance.

I lay my indebtedness to my current organisation where I am working, Bharati Vidyapeeth's College of Engineering (BVCOE), New Delhi for exhibiting a faith

in me and extending coorporation during the process of carrying out my research work along with my professional responsibilities in the organisation. I am thankful to HOD-ECE and all staff members of BVCOE, New Delhi for their kind help and support during the entire period of my research.

Above all, my deepest thanks goes to my parents and family members for their constant support, encouragment and blessings. Their inspiration and moral support has enlighten my path throughout the entire journey of doctoral research. A special thanks to my wife, Kanchan Bansal for her unflinching support, affection, understanding and always being there for me. She has endured so much over the years but never complained about the time that i could not spend with her.

Finally, my sincerest and heartful thanks to all the people who have extended their helping hands directly or indirectly at DTU and BVCOE. I am immensely grateful to each and every person who enriched this journey of intellectual pursuit.

My apologies if I have fortuitously omitted anyone to whom acknowledgement is due.

Ashish Kumar

List of Publications

1. Ashish Kumar, Gurjit Singh Walia, Kapil Sharma (2020), "A Novel Approach for Multi-cue Feature Fusion for Robust Object Tracking", Applied Intelligence 50, pp. 3201–3218, Springer, (IF: 3.325), DOI: https://doi.org/10.1007/s10489-020-01649-9.

2. Ashish Kumar, Gurjit Singh Walia, Kapil Sharma (2020), "Real-time Visual Tracking via Multi-cue based Adaptive Particle Filter Framework", Multimedia Tools and Applications 79, pp. 20639–20663, Springer, (IF: 2.313), DOI: https://doi.org/10.1007/s11042-020-08655-6.

3. Ashish Kumar, Gurjit Singh Walia, Kapil Sharma (2020), "Recent trends in multicue visual tracking: A review", Expert System with Applications 162, pp.113711, Elsevier, (IF: 5.452), DOI: https://doi.org/10.1016/j.eswa.2020.113711.

4. Ashish Kumar, Gurjit Singh Walia, Kapil Sharma, "Robust object tracking based on adaptive multicue feature fusion", Journal of Electronic Imaging, SPIE, (IF: 0.884), (Accepted).

5. Ashish Kumar, Gurjit Singh Walia, Kapil Sharma (2020). “Real-time Multi-cue Object Tracking: Benchmark”, In: Int. conf. on IoT Inclusive Life (ICIIL-2019), (Paper Published & Presented).

6. Ashish Kumar, Gurjit S Walia, Kapil Sharma (2019). “A Novel Approach to Overcome Sample Impoverishment Problem of Particle Filter using Chaotic Crow search algorithm”, In: International Conference on Futuristic Technologies (ICFT-2019). (Paper Published & Presented).

Abstract

Multi-cue object tracking is a challenging field of computer vision. In particular, the challenges originate from environmental variations such as occlusion, similar background and illumination variations or due to variations in target's appearance such as pose variations, deformation, fast motion, scale and rotational changes. In order to address these variations, a lot of appearance model have been proposed but developing a robust appearance model by fusing multi-cue information is tedious and demands further investigation and research. It is essential to develop a multi-cue object tracking solution with adaptive fusion of cues which can handle various tracking challenges. The goal of this thesis is to propose robust multi-cue object tracking frameworks by exploiting the complementary features and their adaptive fusion in order to enhance tracker's performance and accuracy during tracking failures.

A real-time tracker using particle filter under stochastic framework has been developed for target estimation. The inherent problems of particle filter namely, sample degeneracy and impoverishment have been addressed by proposing a resampling method based upon meta-heuristic optimization. In addition, an outlier detection mechanism is designed to reduce the computational complexity of the developed tracker.

A robust tracking architecture has been proposed under deterministic framework. Fragment-based tracker with a discriminative classifier has been designed that can enhance tracker's performance during dynamic variations. Periodic and temporal update strategy is employed to make tracker adaptive to changing environment. Extensive experimental analysis has been performed to prove the effectiveness of the developed tracking solution.

Multi-stage tracker based on adaptive fusion of multi-cue has been developed for multi-cue object tracking. The first stage of target rough localization improves the accuracy of tracker during precise localization. In the appearance model complementary cues are considered to handle illumination variations and occlusion. Classifier mechanism and fragment based appearance model are proposed to improve the tracker's accuracy during background clutters and fast motion. Experimental validation on multiple datasets validates the performance and accuracy of the proposed tracker.

Table of Contents

Certificate i

Declaration of Authorship iii

Acknowledgements iv

List of Publications vi

Abstract viii

Table of Contents x

List of Figures xii

List of Tables xiv

1 Introduction 1
- 1.1 Multi-Cue Object Tracking 1
- 1.2 Thesis Overview 5

2 Literature Review 8
- 2.1 Stochastic Framework 8
- 2.2 Deterministic Framework 13
- 2.3 Multi-stage Framework 15
- 2.4 Performance Validation 17
  - 2.4.1 Evaluation Metrics 18
  - 2.4.2 Benchmark Dataset 19
- 2.5 Research Gap 22
- 2.6 Research Motivation 24
- 2.7 Research Objective 25
- 2.8 Significant Findings 26

3 Stochastic Framework for Multi-cue Object Tracking 29
- 3.1 Introduction 29
- 3.2 Proposed Tracker Architecture 31
- 3.3 Core Design of Proposed Tracker 32
  - 3.3.1 Multi-cue Particle Filter 33
  - 3.3.2 Multi-cue Feature Extraction 35
  - 3.3.3 Proposed Adaptive Multi-cue Fusion 38
  - 3.3.4 Proposed Optimum Resampling Approach 41
- 3.4 Experimental Validation and Discussion 43
  - 3.4.1 Attribute based Evaluation 43

- 3.4.2 Analysis of Feature Contribution . . . 53
- 3.4.3 Computational Complexity . . . 53
- 3.4.4 Overall Performance Evaluation . . . 54
- 3.5 Significant Findings . . . 57

4 Deterministic Framework for Multi-cue Object Tracking 60

- 4.1 Introduction . . . 60
- 4.2 Proposed Tracker Architecture . . . 62
  - 4.2.1 Multi-cue Feature Extraction . . . 64
    - 4.2.1.1 Proposed Multi-cue Feature Fusion . . . 69
- 4.3 Experimental Analysis and Discussion . . . 72
  - 4.3.1 Qualitative Analysis . . . 73
    - 4.3.1.1 Illumination & Scale variations . . . 74
    - 4.3.1.2 Full occlusion (FOC) or Partial occlusion (POC) . 76
    - 4.3.1.3 Deformation & Background clutter . . . 76
    - 4.3.1.4 In-plane and Out-of-plane rotation . . . 78
    - 4.3.1.5 Fast motion & Motion blur . . . 80
    - 4.3.1.6 Low resolution & Out-of-view . . . 80
  - 4.3.2 Quantitative Analysis . . . 82
- 4.4 Significant Findings . . . 86

5 Multi-stage Framework for Multi-cue Object Tracking 89

- 5.1 Introduction . . . 89
- 5.2 Proposed Tracker Architecture . . . 91
  - 5.2.1 Stage I: Rough Localization . . . 93
  - 5.2.2 Stage II: Precise Localization . . . 94
    - 5.2.2.1 Multi-cue Feature Fusion . . . 97
- 5.3 Experimental Validation . . . 101
  - 5.3.1 Qualitative analysis . . . 102
    - 5.3.1.1 Illumination Variation . . . 102
    - 5.3.1.2 Full or Partial Occlusion . . . 105
    - 5.3.1.3 Background Clutters . . . 108
    - 5.3.1.4 Fast Motion and Deformation . . . 109
  - 5.3.2 Quantitative Analysis . . . 111
- 5.4 Significant Findings . . . 116

6 Conclusion and Future Directions 119

- 6.1 Summary of Major Contributions . . . 119
- 6.2 Directions of Future Work . . . 125

References 127

Appendix 146

List of Figures

3.1 Architecture of the proposed approach. At each time step t, the particles are initialized and evolved through the state model. Each particle is evaluated for three cues and reference histogram dictionary is extracted. Cues are integrated through an adaptive fusion model with context sensitive cue reliability and outlier detection. Particles are resampled and final state is estimated as the weighted mean of the outliers and important particles. 33
3.2 Representative frames from the considered challenging video sequences. Each frame has been labeled with # frame number and video sequence name on the left and right corner, respectively. . . . 45
3.3 Success plot under tracking challenges as (a) Illumination variations (b) Background Clutters (c) Fast Motion and Motion Blur (d) Scale variations and Deformation (e) In-plane and Out-of-plane rotation (f) Full or partial occlusion. Legend includes Area Under Curve (AUC) in the bracket. 47
3.4 Precision plot under tracking challenges as (a) Illumination variations (b) Background Clutters (c) Fast Motion and Motion Blur (d) Scale variations and Deformation (e) In-plane and Out-of-plane rotation (f) Full or partial occlusion. Legend includes average Distance Precision (DP) score in the bracket. 49
3.5 Overall performance comparison of the proposed tracker with other trackers: (a) Success plot (b) Precision plot. Legend includes the AUC for success plot and average DP for precision plot respectively, in the brackets. 55

4.1 Overview of the proposed tracker architecture. At each time step (t=1), robust unified feature is generated by the multi-cue fusion model for each positive and negative fragments. Complementary features namely, color and HOG are extracted and fragments are initialized on the target after rough localization. Tracker is made adaptive considering random forest classifier and reference dictionary update. 63
4.2 Sample Tracking frames on some critical video data with illumination & scale variation challenge: a) *Shaking* b) *Singer1* c) *Basketball* 74
4.3 Sample tracking frames on some critical video data with full or partial occlusion challenge: a) *Football* b) *Subway* c) *Jogging1* . . . 75
4.4 Sample tracking frames on some critical video data with deformation & background clutter challenge: a) *Bolt2* b) *Dancer2* c) *Walking2* 77
4.5 Sample tracking frames on some critical video data with IPR & OPR challenge: a) *Soccer1* b) *Pedestrian1* c) *Jogging2* 78
4.6 Sample tracking frames on some critical video data with fast motion & motion blur challenge: a) *Bolt2* b) *Crossing* c) *MountainBike* . . 79

4.7 Sample tracking frames on some critical video data with low resolution & out-of view challenge: a) *Tiger* b) *Walking2* c) *Walking* . 81

4.8 Success plot under tracking variations (a) Illumination & Scale variations (b) Full or partial Occlusion (c) Fast motion & Motion blur (d) In-plane & Out-of-plane rotation (e) Deformation & Background clutter (f) Low resolution & Out-of-view . Legend includes AUC in the bracket. 83

4.9 Precision plot under tracking variations (a) Illumination & Scale variations (b) Full or partial Occlusion (c) Fast motion & Motion blur (d) In-plane & Out-of-plane rotation (e) Deformation & Background clutter (f) Low resolution & Out-of-view. Legend includes mean precision overlap score in the bracket. 84

5.1 Overview of the proposed tracker. Tracker is initialized with positive and negative fragments in the reference dictionary. Stage I, target is roughly localized through optical flow. Then, fragments are generated around the localized target and LBP-HOG features are extracted. Unified feature is generated and transductive reliability is calculated for each localized fragment. Tracker is made adaptive by classifier score and transductive reliability. 91

5.2 Description of distance calculation for feature descriptor (a) LBP (b) HOG 97

5.3 Critical tracking frames from video sequence under Illumination variation challenge: a) *Human8* b) *Tiger* c) *Fish* d) *Tiger1* e) *Car2* f) *boat5* 103

5.4 Critical tracking frames from video sequence under FOC or POC challenge: a) *Jogging1* b) *Subway* c) *Walking2* d) *car15* e) *Walking* f) *Football* 104

5.5 Critical tracking frames from video sequence under Background Clutters challenge: a) *Singer2* b) *MountainBike* c) *Shaking* d) *Bolt2* e) *bike2* f)*Soccer1* 106

5.6 Critical tracking frames from video sequence under Fast Motion and Deformation challenge: a) *Dancer* b) *Jogging2* c) *person23* d) *Jumping* e) *Pedestrian1* f) *Surfer* 107

5.7 Precision Plot under visual tracking challenges: a) Illumination Variations (*Human8*, *Tiger1*, *boat5*, *Tiger2*) b) Full or Partial Occlusion (*Jogging1*, *Subway*, *Walking2*, *Walking*) c) Background Clutters (*Singer2*, *Soccer*, *Shaking*, *Bolt2*) d) Deformation and Fast Motion (*Dancer*, *Jumping*, *Couple*, *person23*) 113

5.8 Success plot under visual tracking challenges: a) Illumination Variations (*Human8*, *Tiger1*, *boat5*, *Tiger2*) b) Full or Partial Occlusion (*Jogging1*, *Subway*, *Walking2*, *Walking*) c) Background Clutters (*Singer2*, *Soccer*, *Shaking*, *Bolt2*) d) Deformation and Fast Motion (*Dancer*, *Jumping*, *Couple*, *person23*) 114

List of Tables

3.1 Tracking challenge and considered video sequence 44
3.2 Comparison of contribution of each feature 53
3.3 Tracking Speed Analysis (FPS) 54
3.4 Comparison results for average CLE. First, Second and Third results are shown, respectively. 54
3.5 Comparison results for average F-Measure. First, Second and Third results are shown, respectively. 55

4.1 Fuzzy inference rules 71
4.2 Comparative average centre location error tracking results (in pixels). First, second and third results are highlighted respectively. . . . 82
4.3 Comparative average F-Measure tracking results. First, second and third results are highlighted respectively. 82

5.1 CLE (in pixels). Red, blue and green represent the best results. . . . 111
5.2 F-Measure. Red, blue and green represent the best results. 112

Chapter 1

Introduction

Chapter 1

Introduction

Object tracking is substantially imperative in computer vision due to various realistic applications in video processing, medical imaging, robotics, human-machine interaction, traffic control, pedestrian tracking, augmented reality, and many more. However, tracking an object in a video sequence is very tedious due to dynamic environmental conditions which includes pose variations, illumination variations, full or partial occlusion, similar background, noise in video (dust, rain, snow, haze etc.), fast and abrupt object motion.

1.1 Multi-Cue Object Tracking

Earlier, the object tracking methods were proposed considering single cue only. Cues namely, color, texture, gradient, contour, spatial energy, motion, orientation or thermal profile were extracted in tracker's appearance model [1], [2]. However, now a days most of the research work emphasizes to extract multi-cue features for target representation in order to cater tracking challenges. Recently. many researchers has investigated and discussed at length that complementary multi-cues have addressed object tracking challenges in a better way in comparison to single cue based object tracking [3]. Complementary cues can compensate for each other when either of the cue performance degrades during tracking challenges. Color cue requires less computation and processing in comparison to other cues. However, trackers exploiting only color cue are not able to cope with change in illumination and full occlusion. On the other hand, HSV color space prove to be robust to illumination variations but its performance deteriorates in low saturation videos [4]. Texture cue extracts low level details of the target but fails to handle the scale and rotational variations in the sequence. Texture cue is augmented with color cue to improve tracker's performance during tracking challenges [5]. Gradient cue can cater scale variations and target's deformation but performance of the trackers exploiting only gradient cue degrades due to background clutters. Motion cue can be extracted from subsequent frames of the video sequences but lack shape and gradient information. Inclusion of shape or color cue with motion cue was proposed in order to improve tracker's performance during heavy cluttered environment [6],[7]. Shape cue can handle deformations in rigid object but is inefficient in managing the time varying deformation in non-rigid objects [8]. Depth cue can be extracted from Kinect sensors and RGB-D camera. It can prevent tracker's drift during scale variations but it lack's motion information. Due to which it is unable to handle fast motion and occlusion [9]. In [10], authors has combined motion cue, HSV color with depth information to address occlusion, fast motion and background clutters. Thermal cue extracts from thermal camera determines the thermal profile of the object and is robust to illumination variations. However, trackers based on thermal profile alone are not able to differentiate between the objects having similar thermal distribution and hence, their performance deteriorates under occlusion and background clutters. Complementary cue gray-scale has been proposed with thermal data in order to overcome its deficiency [11]. Audio cue can provide new inputs to object tracking as audio signals can handle out of view and occlusion. But, audio information is very sensitive to background noise and is not able to discriminate between the sounds coming from front and rear sources. Depth cue and visual cue information is incorporated with audio to beaten its shortcomings [12], [13]. Apart from the features extracted from various sensors namely, vision, thermal, Kinect and audio, deep features can also be extracted to improve tracker's accuracy [14], [15]. Deep features neglect the target's appearance and motion information and hence, low level features color and motion cues are embedded with deep features to avoid tracking failures [16]. In sum, multi-cues are more efficient and reliable in comparison to single cue. Fusion of complementary cue information not only handles environmental variations effectively but also enhances tracking accuracy. Hence, researchers are motivated towards exploiting multi-cue information for developing robust tracking solutions.

Multi-cue object tracking algorithms focus on predicting the target state in the video sequences. Multi-cue object tracking framework can be broadly categorized into stochastic framework, deterministic framework and multi-stage framework. Generally, tracking methods under stochastic framework consider Monte Carlo simulation for predicting the target state. Under this, particle filter, Kalman filter and their variants are exploited for providing the robust multi-cue object tracking solutions [17], [18]. These methods are widely explored in the object tracking area due to their wide potential of handling the challenges with simplicity. However, particle filter [19] is capable of handling the non-linear and non-Gaussian multi-cue tracking problems with great potential. But, the performance of the tracker under this approach is heavily affected due to the shortcomings of particle filter viz. particle degeneracy and sample impoverishment. On the other hand, tracking methods under deterministic framework work on cost minimization of objective function by extraction of foreground pixels from the background area. Deterministic framework includes tracking methods based on Mean shift, fragment based tracking, tracking by detection and tracking by parts [20], [21], [22]. Deterministic methods requires less computational load but most of considered methods are limited to local minima problems for optimal solutions. However, estimation of local and global minima for tracking solutions is still a open problem. Tracking methods under multi-stage framework improve tracking accuracy by incorporating two stage estimation [23]. The coarse to fine localization of target enhances tracker's performance during various tracking variations. Multi-cue are exploited for precise estimation of the target and are fused using either feature level fusion or score level fusion. Feature level fusion preserves high level relationship among features in comparison to score level fusion [24]. However, processing of a concatenated unified feature requires a lot of computational processing [25]. In sum, a lot of solutions have been provided under each category but there is still scope of improvement that can be considered for developing robust tracking frameworks.

1.2 Thesis Overview

The thesis comprises of six chapters and a brief description of these chapters is given below:

Chapter 1:- This chapter covers the introduction and purpose of the outlined research topic. It will also contain the main idea for the development of the thesis. In addition, the potential application areas and main challenges in multi-cue object tracking are covered.

Chapter 2:- This chapter covers the state-of-the-art techniques developed in existing research work on “Multi-cue object tracking under stochastic framework, deterministic framework, and multi-stage framework”. It will also highlights the research gaps in the existing work that has stimulated the development of research objectives. In addition, evaluation metrics and benchmark datasets require for the performance validation of the proposed trackers are discussed.

Chapter 3:- This chapter highlights the details of the methodology adopted to accomplish the multi-cue based object tracking under stochastic framework. In addition, it will also cover the observations and discussion of results.

Chapter 4:- This chapter highlights the details of the methodology adopted to accomplish the multi-cue based object tracking model under deterministic framework. The obtained experimental results will also be elaborated against the other compared state-of-the-art.

Chapter 5:- This chapter highlights the details of the methodology adopted to accomplish the multi-cue based object tracking under multi-stage tracking framework. The brief details highlighting the accuracy and the effectiveness of the proposed methodology will also be discussed.

Chapter 6:- This chapter contains the brief summary of all the ideas, observations and contributions of the resultants obtained in each objective. Also, the future directions are sketched in this chapter.

Chapter 2

Literature Review

Chapter 2

Literature Review

Over the past decade, multi-cue object tracking has been extensively explored and reviewed. In particular, the success of any object tracking method is highly dependent on developing an efficient and dynamic appearance model. In this direction, the multi-cue object tracking methods have been proposed under stochastic framework, deterministic framework or multi-stage framework. The various appearance models under each framework are briefly reviewed and are detailed as follows.

2.1 Stochastic Framework

Generally, multi-cue object tracking methods under stochastic framework consider Monte Carlo simulation for predicting the target state using particle filter, Kalman filter and its other variants [17], [26]. Particle filter has potential application in the field of multi-cue object tracking [27], [28]. In this direction, Xiao et al.[8] exploited color and shape feature of the target using a particle filter and then, fused them by normalized weighted feature fusion. Model was made adaptive by extracting the contextual information. Contextual information for each cue was determined by analyzing the particles in the search space based on the foreground and background information encoded in them. Experimental results revealed that there was no significant improvement in many video sequences by the adaptive fusion of feature saliency. Walia et al.[5] proposed to fuse color and texture using fuzzy based adaptive fusion model. The proposed outlier detection mechanism not only enhanced the performance of crow search based resampling strategy, but also reduced the computational complexity of the tracker. Also, transductive reliability was integrated at each time step for adaptive fusion and faster convergence. However, the method was analyzed on limited video sequences. In [29], authors proposed fusion of adaptive patches after extracting Color Histogram (CH) and Histogram of Oriented Gradient (HOG) features from object. Similarity between the reference patch and the object's appearance was calculated using weighted Bhattacharyya coefficient. Experimental results evaluated the performance of the method for similar objects only. Similarly, Sardari and Moghaddam [30] determined CH and HOG for target's appearance model. Occlusion handling and detection mechanism were exploited to maintain the tracker's performance during the challenge. MGbSA based resampling was employed on each particle to reduce the number of particles using local search and spiralchaoticmove procedure. However, in [31] blocking strategy was exploited to handle partial occlusion and least square method for severe occlusion. Integral histogram was proposed to reduce the computational cost of the method by integrating the colour cumulative histogram and Local Binary Pattern (LBP) features using deterministic coefficient. Color based particle filter in the local region and motion estimation in the global region were used to localize the target [32]. Number of particles were reduced and computational efficiency in the local-global estimation of the method was enhanced by deploying chaotic system in the framework. In [33], authors proposed particle filter for local state estimation and Interacting Multiple Models (IMM) for global state estimation. Color and texture features were extracted from appearance model. Color was determined through α-Gaussian mixture model and texture was extracted through distinctive uniform local binary pattern histogram based on the uniform LBP operator. Each feature was extracted through H-∞ filtering and combined using linear weighted fusion. On the other hand, Walia et al. [34] utilized Dezert-Smarandache Theory (DSmT) based PCR-6 for particle level fusion of the complementary cues viz. color, LBP and edge. Adaptive integration of the particles was performed in which each particle was discounted with transductive reliability before fusion. Similarly, Wen et al. [35] used color and motion information of the target but integrated them using linear weighted method. Firouznia et al. [32] used motion information to estimate the target's position globally and then the color information was extracted to localize target in the local region using chaotic particle filter. In [36], author exploited color, thermal and motion cue for observation model. Conflict amongst various cue was resolved using Dsmt based PCR-5 rules. Dou and Li [37] used Corrected Background-Weighted Histogram (CBWH), Completed Local Ternary Patterns (CLTP) and HOG for the target model. Multiple interactive model probabilities were used for particle importance. In [38], authors proposed a tracking algorithm using particle filter for tracking object in InfraRed (IR) video sequences. Co-occurrence matrix moments was proposed for the colour and texture features considering the fact that two similar colour objects could not have same moments. Sample particles weights were normalized before final state estimation. However, authors exploited sample constrained particle filter and sparse representation for tracking small objects in IR videos [39]. Saliency extraction was employed to determine the high frequency area in the image and hence, to reduce the computational load on the tracker. Sample constraint was integrated in particle filtering before estimating the final state. In [40], author used new 6 degree-of-freedom tracking technique in RGB-D images for 3D pose tracking. Particle Swarm Optimization (PSO) was embedded in particle filter for evaluation of particles and restricting the search space. To handle occlusion, depth was integrated in RGB to compute normal vector using the least-square plane fitting. In [41], authors has extracted color and intensity features to localize the target in visual attention integrated particle filter framework. Salient regions were searched to detect the target in the video but the proposed work was inefficient in handling target's appearance variations due to changing environment. In [42], color, texture and HOG cue were determined for each particle. Linear normalized weighted fusion of cues was proposed by integrating reliability at each frame. Linear fusion of cues was less efficient in capture the discriminative capability of each exploited cue. In addition, the proposed method was computationally slow and inefficient in handling the background clutters. Zhang et al. [43] proposed a tracking algorithm exploiting color, HOG and CNN features. Interdependencies between each feature was exploited using incremental strategy in the update model. The method had addressed the particle filter limitations to a great extent but was computationally expensive and not suitable for real-time tracking problems. In sum, the existing works have limitedly addressed real-time tracking problem which can handle various environmental variations namely, illumination variation, background clutters, full or partial occlusion, scale variations, deformation, fast motion and motion blur.

In addition, the performance of multi-cue object tracking methods using particle filter can be improved through resampling methods based on meta-heuristic approaches. In this direction, Zhang et al. [44] exploited PSO as resampling technique and determined intensity and gradient cues for each particle. Discriminative weight of each cue was fused adaptively at each frame. In [45], authors proposed immune genetic algorithm as resampling method to address PF problems. Crossover, mutation, promotion and inhibition parameters were used to ensure particle diversity in search space. However, the performance of this method was evaluated only on two video sequences. On the other hand, Yin et al. [46] used color and motion cues in particle filter based object tracking. CamShift exploited as optimization to determine the local maxima during environmental variations. However, the method was evaluated using limited performance metrics. In [47], authors proposed improved cuckoo search to solve the problems of particle filter for object tracking. Simulation results on video sequences were presented to prove the robustness of the proposed methods. In [35], author extracted two features from the target and fused them using linear weighted method. In sum, many resampling methods under multi-cue particle filter for tracking the object in video sequence has been proposed. However, the shortcomings of particle filter with adaptive fusion methodology are still need to be addressed for efficient object tracking.

2.2 Deterministic Framework

Multi-cue object tracking methods under deterministic framework focus on cost minimization of objective function by extraction of foreground pixels from the background area. Deterministic methods include tracking framework based on Mean shift, tracking by detection and tracking by parts [48], [49], [50]. Under this, contour based tracking was proposed by analyzing motion, shape and gabour features [51]. For background subtraction and to obtain foreground pixels Expectation Maximization based Effective Gaussian Mixture Model (EMEGMM) algorithm was exploited. The approach had limited applications and hence, could be used for tracking non-rigid objects only. In [48], HSV based color model and LBP based texture features were integrated to address the limitation of mean shift algorithm. Four neighbourhood search method was used to handle tracker's drift during occlusion. However, Circulant Structure Kernels (CSK) was exploited using patch based tracking to address partial to severe occlusion. Entropy minimization criterion was employed to redetect the target from the classifier pool after tracking failure. Yao et al. [49] proposed integration of semantic information with color and location using scale adaptive based energy minimization method. Segmentation of the target was done through energy minimization by graph cut algorithm for accurate localization. Position and shape of the target's appearance was modeled in tracklet-detection based tracking framework [6]. For each tracklet, confidence score was determined to identify the reliable and unreliable tracklet. Correlation filter was used to calculate the confidence score and to handle occlusion. Parate et al. [52] proposed a hybrid solution for efficient tracking by integrating global template and patch-based template. Integral Channel Features (ICF) were exploited for extracting colour and structural features for robust hybrid template. Clustering and vector quantization were used to handle the background clutters. However, to handle occlusion and background clutters, multiple variants of texture and color were integrated in mean shift based tracking framework [53]. Multiple textures features namely, LBP, Local Ternary Pattern (LTP) and Complete Local Binary Pattern (CLBP) were extracted to cater fast motion, scale and orientation variations. Alpha blending was used to integrate the various texture features in colour information. However, Medouakh et al. [54] proposed to combine texture feature based on Local Phase Quantization (LPQ) and HSV based CH in mean shift framework. Hybrid histogram was generated by integrating color-texture features jointly. Further, the information from two or more sensors can be extracted for providing efficient tracking under deterministic framework. For this, Li et al. [55] exploited RGB color features from vision sensor and thermal features from thermal sensor in a manifold ranking patch based tracking framework. Patch weights and modality weight extracted from structured SVM were combined for state estimation. Xie et al. [10] proposed object level segmentation to handle occlusion combining RGB and depth features in long video sequences. Embedding segmentation mask in keyframes improved the tracker's performance but this process is computationally complex and hence, reduces the efficiency in terms of speed. In order to improve efficiency, Xiao et al. [9] proposed to fuse color information with depth features directly to handle occlusion and fast motion. Spatial and temporal consistency constraints were exploited in update modal for continuous relearning of the tracker. However, color and depth were integrated in a 3-D mean shift framework to maintain tracker's performance during occlusion and background clutters [56]. Color feature was extracted through RGB method as well as HSV color space and the performance for each feature was compared. However, the method was failed to handle long-term occlusion. In sum, many solutions have been proposed to handle partial to severe occlusion, background clutters and fast motion under the framework. But, there is still scope of improvement in terms of methodology for multi-cue feature extraction, fusion of multi-cue information and computational speed.

2.3 Multi-stage Framework

Multi-cue object tracking methods under multi-stage framework aim to precise localization of the target. Initially, the target is roughly localized in the first stage of estimation. Based on the rough estimation, the target is precisely localized in the final stage of estimation. In this direction, multi-stage tracker [23], tracking under large motions [57], UGF tracker [25] were proposed. Multi-stage tracker [23] exploited motion cue for rough estimation of the target. RGB, LBP and PHOG were extracted in the tracker's appearance model for precise estimation.

Multi-cue were fused using non-linear weighted Borda count ranking. Tracker had addressed many challenges but failed to handle the background clutters, occlusion and fast motion efficiently. In order to address motion, Kim et al. [57] proposed coarse to fine estimation of the target. Superpixels were extracted from the target during coarse estimation which was followed by sampling and similarity determination for accurate localization. Walia et al. [25] utilized motion descriptor for initial localization and multi-cue visual descriptors for precise localization of the target. Multi-cue features were unified using graph cross-diffusion based feature fusion. The proposed graph fusion had shown efficient experimental results but was computationally complex. In sum, multi-cue object tracking approach under multi-stage framework improved the state estimation of the target. However, the integration of multi-cue can be explored to provide robust solutions.

Multi-cue features can be unified using feature level fusion or score level fusion. It has been well reviewed in literature that feature level fusion is better in comparison to score level fusion in preserving the relationship between the features [24]. In this direction, authors adaptively fused color and depth information along with spatio-temporal consistency constraints for tracking the object [9]. Kang et al. [58] exploited intensity, edge and texture features of the target and fused them using multi-feature fusion through using a Fisher discrimination criterion. Wang et al. [59] fused location, color and LBP features of the target. In [11], authors considered the thermal profile and grayscale features of the target and fused them using collaborative sparse representation. In [35], authors fused information of two channels and epipolar geometry for target estimation. Meng et al. [60] exploited Hue, Saturation and Value Histogram (HSV) features to cater occlusion and background clutters. In [61], authors integrated color histogram, HSV color, texture and motion features for real-time tracking. Wang et al. [62] extracted color and HOG features from superpixels of the target and used a random forest regressor to learn the superpixel features. In [63], authors proposed a weighted local sparse model considering the context information and reliability of each local patch. The information was fused using an adaptive update template strategy. Sun et al. [64] proposed a complementary measurement matrix by inducing color and texture features. The ensemble classifier was used to classify the previous and current samples in order to avoid tracking failures. In sum, many robust multi-cue tracking solutions considering feature level fusion under multi-stage framework has been proposed. However, developing a robust non-linear unified feature for an adaptive appearance model catering the environmental challenges can be further explored.

2.4 Performance Validation

For performance validation of the proposed architectures robust evaluation metrics are chosen. In addition, the challenging video sequences which include the environmental variations namely, illumination variation, scale variations, background clutters, deformation, full or partial occlusion, in-plane & out-of plane rotations, fast motion and motion blur are taken from publicly available datasets. The details related to the exploited evaluation metrics and the benchmark datasets is discussed in the following sections.

2.4.1 Evaluation Metrics

To prove the effectiveness and accuracy of the proposed trackers over other trackers, robust evaluations metrics viz. Center location error, F-Measure [65], AUC (Area Under Curve), DP (Distance Precision), OP (Overlap precision), Success plots and Precision plots [66] are considered. Center Location Error (CLE) is the Euclidean distance between the tracker's bounding box (BB_t) and the ground truth (BB_g). For each frame of a video sequence, CLE can be computed using Eq. (2.1).

$$CLE = \sqrt{\frac{\sum_{i=1}^{M}((X_t^i - X_b)^2 + (Y_t^i - Y_b)^2)}{M}} \tag{2.1}$$

where, (X_t, Y_t) is the coordinate of the BB_t, (X_b, Y_b) is the coordinate of BB_g and M is independent Monte carlo simulations evaluated for each frame. F-measure (F_m) is calculated as the harmonic mean of the precision (p_r) and Recall (r_c) using Eq. (2.2)

$$F_m = \frac{2 \times p_r \times r_c}{1 + p_r + r_c} \tag{2.2}$$

where, p_r is precision and can be defined as the ratio of overlap of target bounding box and ground truth bounding box with respect to target bounding box and r_c

is recall which is calculated as the ratio of overlap between target bounding box and ground truth bounding box with respect to ground truth bounding box.

Success plot depicts the success score at different overlap thresholds and mean precision at multiple center location error is depicted by the precision plots. AUC is average of success rate determined at different overlap threshold and DP is average precision score calculated at different location thresholds.

2.4.2 Benchmark Dataset

In the last decade, numerous object tracking algorithms have been proposed by different authors in different journals. Most of the reported work has been evaluated on publicly available datasets. These datasets provide a common base for evaluation of various tracking algorithm. The details of the publicly available benchmarked datasets is as follows:

- Amsterdam Library of Ordinary Videos dataset (ALOV++) was one of the largest dataset containing 316 video sequences with 89364 frames [1]. Dataset contained mainly real-time videos gathered from various online sources. Dataset focused on single object tracking with one situation per video. Mostly videos were of short duration with average length of 9.2 sec. Also, 10 long videos with average duration of 1 to 2 mins were included. 64 different object classes were categorized with 13 aspects of different challenges.

- OTB-15 dataset contained 100 fully annotated videos sequences with 59040 frames (in total) captured from stationary camera at 30 fps frame rate [66]. The dataset was suitable for single object tracking with video sequences analyzed on 11 attributed environmental challenges. The ground truth of the object in the scene was available in *.txt format having details of corresponding coordinates in each frame.

- TempleColor-128 dataset (TC-128) contained 128 colored annotated videos sequences with total 55346 frames captured from stationary camera at 30 fps frame rate [67]. Out of 128 sequences, 78 sequences were new and included tough tracking senarios while rest were collected from previous work. The ground truth information was given in *.txt file along with attributed challenge details in each video sequence.

- OTB-13 dataset contained 50 fully annotated videos sequences with 29491 frames (in total) captured from stationary camera at 30 fps frame rate [68]. The dataset has video sequences analyzed on 11 attributed environmental challenges. The ground truth of the object in the scene is available in *.txt format having details of corresponding coordinates of the target in each frame. OTB-50 dataset is a subset of both OTB-100 and TC-128 and hence, contains many common videos.

- UAV-123 dataset consisted of 123 new fully annotated HD videos with more than 110K frames [69]. Videos were captured from low altitude varying from 5 to 25 mtr. by a moving camera mounted on an unmanned aerial vehicle at resolution of between 720 px to 4K and average 30 fps frame rate. Dataset included wide range of scenes alongwith different classes of target in 12 different attributed environmental challenges.

- DTB dataset captured 70 outdoor videos from a moving camera mounted on an unmanned aerial vehicle or drone at 1280 X 720 resolution [70]. It includes 12 diversified challenges but majority of videos attributed with fast camera movements and rotational challenges. The dataset primarily aim to track two objects namely, people and cars.

- GOT-10k consisted of more than 10K videos with 1.5M frames in total having 563 object classes and 87 different motion classes captured at 10 fps frame rate [71]. This large dataset was developed to provide enough training data to the deep learning based trackers. The videos were split into training and testing datasets with zero overlap.

- Need for Speed dataset (NfS) consisted of 100 video fragments with more than 380K frames captured at higher data rate of 240 fps [72]. NfS mainly focused on fast motion with 8 other attributed challenges. Annotation Toolbox was used in each frame for annotating axis aligned bounding box. The dataset is suitable to analyze the performance of fast deep learning trackers.

- VIRAT consisted of 17 videos with 23 different events at maximum resolution of 1920X1080 and 24 fps datarate [73]. The dataset had videos captured from a stationary camera as well as from a aerial vehicles. Stationary camera videos were of 25 hours duration in total with 1080px or 720px resolution and frame rate upto 30 fps. Aerial camera dataset contained 4 hours of videos with 640X480 resolution at 30 fps frame rate.

- Large-scale Single Object Tracking (LaSOT) contained more than 1K sequences with 3.52M frames having 70 different object classes [74]. LaSOT provided both visual manual annotation as well as lingual annotation from natural language. Authors claimed it to be the largest tracking dataset with high quality annotated videos. Dataset provided large training data to meet the requirement of deep learning based trackers.

2.5 Research Gap

Based on the literature survey potential research gaps were identified. The details of the identified research gaps is as follows:

- There are various existing publicly available datasets but these datasets could not able to cater all dynamic environmental variation in a single test video. Different object tracking challenges are addressed in different video sequences.
- The existing datasets are not suitably applicable to the countries like India with the dynamic environment changes and are not particularly calibrated as per the essentials of object tracking.

- The most of the available datasets consider that the object is moving and the camera is stationary. There is still no available datasets which has been created in which object stationary and camera moving.

- Most of the existing techniques under stochastic and deterministic model considered only single cue and considered the video sequences from single datasets.

- Requirement of a adaptive appearance model which can address the change in target's appearance in case of background clutters and abrupt motion.

- Requirement of a technique which can dynamically update the reliabilities of target, so that scaling can be done in a better way.

- Most of the techniques considered limited performance matrices for the evaluation of tracker performance.

- Most of the tracking methods based on particle filter limitedly addressed the resampling drawbacks namely, sample impoverishment and sample degeneracy.

- Object Representation model needs to cater for deformation of object due to fast motion and pose variations for better identification of object in backgrounds clutters.

- Existing techniques not able to estimate the object location precisely when its motion is fast and abrupt.

- Requirement of a better multi-stage tracking framework which can estimate object location from coarse to fine, in order to achieve high precision.
- Requirement of a adaptive fusion model which can incorporate modalities on the basis of requirement.

2.6 Research Motivation

Object tracking is an imperative field of computer vision which aim to keep track of the target's displacement in the subsequent frames. A lot of work under various framework has been proposed to keep target's track but it is still open and challenging due to dynamic environmental conditions that include pose variations, scale variations, illumination variations, full or partial occlusion, fast motion and background clutters. To adapt such variations, single cue is not sufficient to provide robust tracking solutions. Also, it has well acknowledged by many researchers that the integration of multi-cue with discounting cue reliability is tedious in presence of tracking challenges. Most of the present proposed work are not efficient enough to address more than one environmental challenge concurrently. Hence, development of a robust appearance model is paramount that can address multiple tracking challenges. This work is motivated by the fact that multi-cues are necessary for building a robust appearance model. The adaptive fusion of multi-cue with online estimation of cue reliability is another direction that can be evaluated with the aim to provide robust tracking framework. Under particle filter framework, robust tracking solutions were provided but the drawbacks of resampling technique namely, sample impoverishment and degenercy were limitedly addressed. Meta-heuristics optimization based resampling methods can be explored to provide better tracking algorithms. Multi-stage object tracking model can be explored further with the aim to cater fast motion and motion blur by diffusing multi-cue in appearance model. Deterministic based solutions suffered from the local minima problem and tracking samples. This problem can be solved by multi-cue appearance model with adaptive fusion methods and transductive reliability.

2.7 Research Objective

This research was focused to develop a robust, adaptive multi-cue object tracking framework. The objectives which were considered in the current studies are as follows.

- To review various state-of-the-art techniques and frameworks for multi-cue object tracking. Experimental statistical comparison of 10 existing tracker on benchmarked video sequences.
- Review of various available datasets and performance metrics for multi-cue object tracking. Creation of self-dataset for multi-cue object tracking.

- Design and development of multi-cue object tracking under stochastic framework. Experimental validation of proposed framework on different datasets. Performance comparison of proposed method with state-of-the-art techniques.
- Design and development of multi-cue object tracking under deterministic framework. To validate the proposed framework on different datasets. Performance comparison of proposed framework with state-of-the-art techniques.
- Design and development of a multi-cue object tracking under multi-stage object tracking framework. Experimental validation of proposed technique on different datasets. Performance comparison of proposed technique with state-of-the-art techniques.

2.8 Significant Findings

The following were the key findings of the present work.

- Multiple video sequences were captured and analyzed in a self generated dataset.
- Captured video sequences were of adequate length and annotated to compare the performance of various trackers.
- Reviewed the latest trends in multi-cue object tracking frameworks in which the complementary cue information was extracted either from single sensors or multiple sensors.

- Multi-cue object tracking frameworks were elaborated and recent work was briefly reviewed and investigated.

The self generated dataset along with others significant findings were published in [75].

In addition, multi-cue object tracking frameworks had been reviewed and categorized under various methods. We had briefly analyzed the various tracking benchmark and tabulated their substantial parameters. Also, the experimental evaluation of the recent state-of-the-art had been performed and results were compared. The literature survey of the recent multi-cue object tracking methodology and benchmark along with experimental results were published in [76].

Chapter 3

Stochastic Framework for Multi-cue Object Tracking

Chapter 3

Stochastic Framework for Multi-cue Object Tracking

The aim of this work is to propose a real-time tracker under stochastic framework. Stochastic framework includes linear & Gaussian state estimation and non linear & non-Gaussian state estimation. Kalman filter and its other variants addressed linear and Gaussian state estimation [17], [18]. While particle filer, condensation filter can be utilized for non-linear and non-Gaussian estimation [19], [77]. Under stochastic framework, particle filter has shown superior performance for state estimation in multi-cue object tracking [5], [34].

3.1 Introduction

Particle filter (PF) can be defined as bootstrap filter [19] and used for state estimation using Sequential Monte Carlo methods. It consists of two steps: 1) Prediction, which evolves particle using state model 2) likelihood calculation, which determines particles weight during update state. The main advantage of PF is that it reduces the batch of sample patches during tracking and handle the dynamic environment challenges. Due to this, PF has been widely used in many real-time tracking frameworks [8],[46]. However, the performance of PF is restricted by its inherent shortcoming of sample degeneracy. Sample degeneracy is the problem when most of the particles weight become so negligible that they do not contribute much towards state estimation. Many resampling techniques [78], [79] were studied to address the PF drawback. Generally, resampling techniques replace the low weight particles but this process reduce the diversity in the search space. Hence, most of the resampling methods contribute to sample impoverishment where maximum particles accumulated to small area. To solve this, many nature-based optimization viz. Firefly algorithm [80], Improved cuckoo search [47], Ant optimization [81], PSO [44] were explored to a great extent. These methods improved PF based trackers efficiency by addressing its problems. These techniques were applied as resampling methods in particle filter tracking framework in which the appearance model was constructed either through single cue or multi-cue. In this direction, authors proposed modified galaxy-based search algorithm as resampling technique for estimating the target's optimum state [30]. However, the speed of the method was relatively slow. Zhang et al. [44] exploited PSO as resampling technique and determined intensity and gradient cues for each particle. But, the performance of the tracker was degraded during environmental variations. In [45], authors proposed immune genetic algorithm (IGA) as resampling method to address PF problems. However, the IGA based resampling method provided limited solution to PF drawbacks. In [47], authors proposed improved cuckoo search to solve the problems of PF for object tracking. Limited experimental results were presented to prove the efficiency of the method. In sum, many resampling techniques has been proposed to address PF shortcoming and to provide robust tracking methods. However, the shortcomings of particle filter with adaptive fusion methodology are still need to be addressed for efficient real-time object tracking. In order to address the shortcomings of the particle filter, an adaptive real-time multi cue object tracker under stochastic framework has been proposed and is detailed as follows.

3.2 Proposed Tracker Architecture

The proposed method utilizes complementary cues using multi-cue PF framework tracking the object in a video sequence. Adaptive update model with context sensitive cue reliability have been proposed for tracker's quick adaptation to environmental variations. In addition, problems of PF are addressed by proposing butterfly search optimization based resampling method. Architecture of the proposed tracker, update model and the proposed resampling technique are depicted in Fig. 3.1. Initially, the target is segmented from the background using the GMM subtraction [82]. Particles (N) are instantiated around the centroid of the detected target. All of these particles are evolved through the state vector (S_t) defined by the multi-component state model. Predicted particles are evaluated for each features descriptor namely color histogram (F_i), texture (F_e) and edge (F_g) individually. These cues are further subjected to adaptive fusion approach to obtain the fused weight ($\hat{W}_f$). This model ensures the automatic boosting and suppression of the particles during environmental challenges. At each time step t, context sensitive cue reliability (re_t^l) is also calculated to discount the particles based on each cue performance. The particles so obtained are passed through an outlier detection mechanism to detect outliers ($\hat{U}_x$) and important particles ($\hat{I}_y$). Outliers are passed through a butterfly optimization based resampling technique (BOA) to obtain (U_x). BOA based resampling diversifies the unimportant particles in the search area with its two variables viz. sensor modality and switch probability. Sensor modality helps in sensing the location of the particles and switch probability propagates the particles in the high likelihood region. Final state (G_t) of the target is estimated by the weighted sum of the resampled outliers (U_x) and the important particles ($\hat{I}_y$). This process is repeated iteratively during the entire video sequence. Reference dictionary ($\hat{D}_r$) is updated by selective replacement of the important particles. This ensures consistent update of the appearance model at each time step in accordance with environmental variations. Core design of the proposed multi-cue tracker is as follows.

3.3 Core Design of Proposed Tracker

In this section, core design of the multi-cue based adaptive tracker is discussed. The details of the multi-cue particle filter alongwith multi-cue feature extraction and adaptive multi-cue fusion model are as follow.

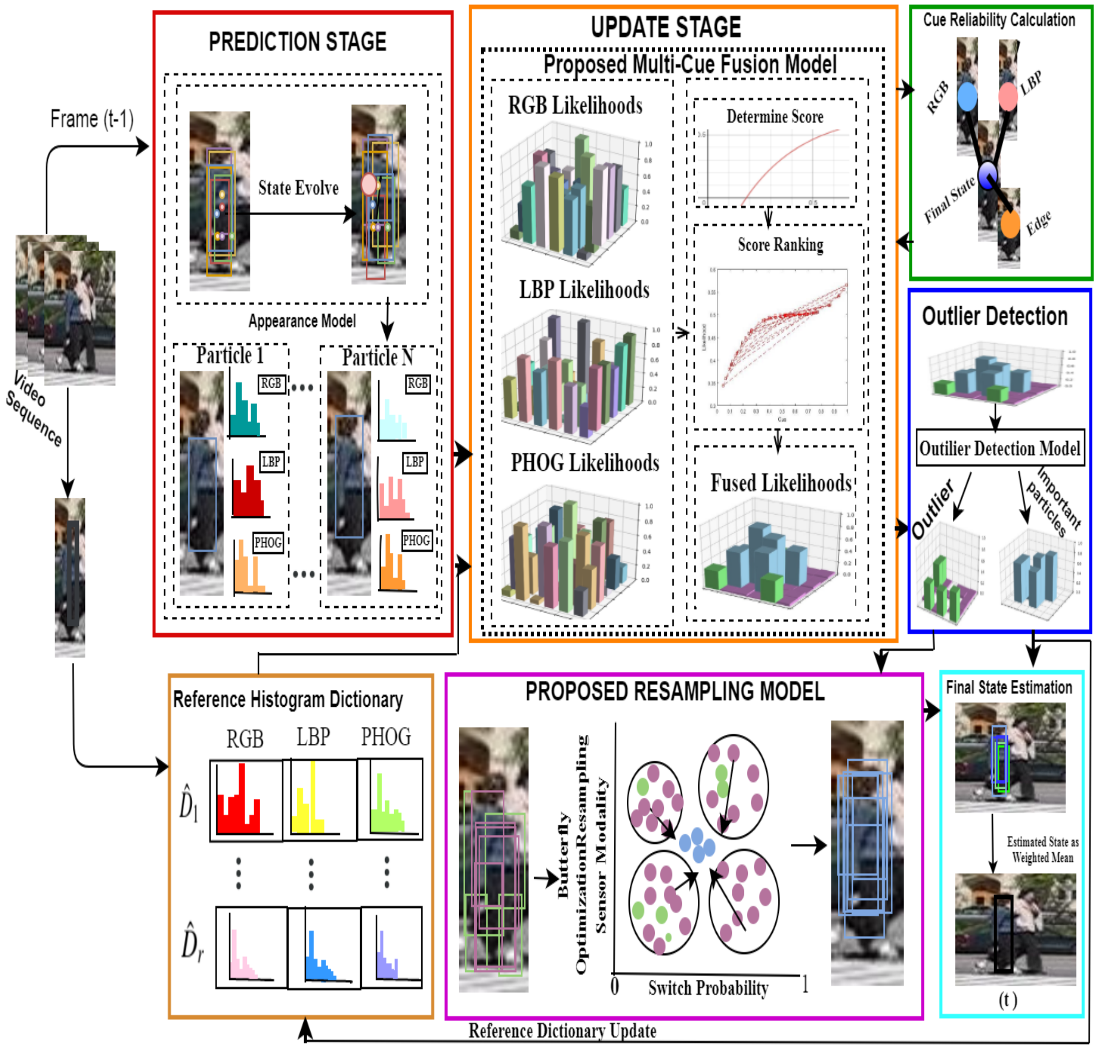

Fig. 3.1 *Architecture of the proposed approach. At each time step t, the particles are initialized and evolved through the state model. Each particle is evaluated for three cues and reference histogram dictionary is extracted. Cues are integrated through an adaptive fusion model with context sensitive cue reliability and outlier detection. Particles are resampled and final state is estimated as the weighted mean of the outliers and important particles.*

3.3.1 Multi-cue Particle Filter

Particle filter utilizes Monte Carlo algorithm for solving the state estimation problems. Particle filter has two stages: prediction stage and update stage to obtain the desired PDF. During the prediction stage, particle state is evolved through the system model at time t using Eq.(3.1).

$$S_t = f_{t-1}(S_{t-1}, \sigma_{t-1}), \tag{3.1}$$

where, f_{t-1} is non-linear state transition function and σ_{t-1} is zero mean white noise. Particles are distributed on the target to estimate the posterior density distribution $p(S_t|x_t)$ where, $x_t = x_1, x_2 ... x_t$ is the set of available information at time t. The measurement x_t at time t is determined by the observation model using Eq. (3.2).

$$x_t = w_t(S_t, K_t) \tag{3.2}$$

where, w_t is the non-linear function and K_t is zero mean white noise, independent of present and past state. If PDF $p(S_{t-1}|x_{t-1})$ is available at time $t-1$, then the prior PDF is obtained using the Eq.(3.3).

$$p(S_t|x_{t-1}) = \int p(S_t|S_{t-1}) p(S_{t-1}|x_{t-1}) dS_{t-1}, \tag{3.3}$$

Finally, when the measurement x_t is available at time t, then the state is updated using Baye's rule given by Eq.(3.4).

$$p(S_t|x_t) = \frac{p(x_t|S_t)p(S_t|x_{t-1})}{p(x_t|x_{t-1})} \tag{3.4}$$

Multi-component model defines state of the particle by the state vector S_t which included variables namely $X_p, Y_p, V_{xp}, V_{yp}, ro_p$ and α_p. (X_p, Y_p) is the center of rectangle of bounding box. V_{xp} and V_{yp} are the velocities in respective directions for p^{th} particle modeled through constant velocity model. ro_p and α_p are the rotation and scaling variables of the particle which used random walk model. Predicted particles are obtained by evolving the particles through the state model using Eq.(3.1). Each predicted particle is subjected to multi-cue evaluation namely, color histogram (F_i), texture (F_e) and edge (F_g). Color cue is stable during change in scale and partial occlusion, texture cue can handle change in illumination and similar background, and edge cue is invariant object's deformation and rotation. These are complementary cues i.e. if one fade others may compensate during the tracking challenges. Details of the likelihood calculation for these cues is discussed in the following section.

3.3.2 Multi-cue Feature Extraction

The proposed tracking approach considers three complementary cues namely, color, LBP and PHOG for each predicted particles. The details of each cue alongwith their likelihood calculation is as follows: RGB color model is exploited for calculating the color histogram. Color histogram for p^{th} particle is $F_i^p = H_1, H_2 ... H_{\hat{N}_b}$ and determine using Eq. (3.5).

$$H_c = Z \sum_{q=1}^{\hat{N}} B(m_q, n_q), c = 1, 2....\hat{N}_b \tag{3.5}$$

where Z is the normalizing constant and $B(.)$ represents binning function that assigned pixel (m_q, n_q) to one of the $\hat{N}_b$ histogram bins. For each particle, color histogram (F_i^p) is calculated by mapping it to normalized H_c. Texture cue measures intensity variations of the targets surface [83]. It is calculated for each particle using scale-invariant LBP depicted by Eq.(3.6).

$$LBP_{\alpha,\hat{N}_\eta} = \begin{cases} \sum_{r=0}^{\hat{N}_\eta - 1}(L(G^k - G^c)), & \text{if} U(LBP_{\alpha,\hat{N}_\eta}) \leq 2 \\ \eta + 1, & \text{otherwise} \end{cases} \tag{3.6}$$

where, G^k is grey value for k^{th} pixel and G^c is grey value for central pixel(c) having α radius with $\hat{N}_\eta$ as equally spaced neighboring pixels. $L(.)$ is step function. $U(LBP_{\alpha,\hat{N}_\eta})$ determine the uniform pattern in the image of the pattern labels. For each particle, LBP histogram (F_e) is determined by normalizing the value of the $LBP_{\alpha,\hat{N}_\eta}$. Spatial distribution of edges is considered for the representation of target's shape. Pyramid of Histogram of Oriented Gradients (PHOG) [84] is used for the extraction of shape information of the target with its spatial distribution. In this work, PHOG is extracted for each particle considering the target's shape and edge orientation. For this, Region Of Interest ($\mathcal{I}$) is extracted and then intensity gradient values are computed for pixel (m_q, n_q) using Eq.(3.7).

$$E(m_q, n_q) = \sqrt{(\mathcal{I}(m_q, n_{q+1}) - \mathcal{I}(m_q, b_{q-1}))^2 + (\mathcal{I}(m_{q+1}, n_q) - \mathcal{I}(m_{q-1}, n_q))^2} \tag{3.7}$$

For a given image, 20 bins are utilized using orientation which can be depicted using Eq. (3.8)

$$\theta = \tan^{-1}(\frac{\mathcal{I}(m_q, n_{q+1}) - \mathcal{I}(m_q, n_{q-1})}{\mathcal{I}(m_{q+1}, n_q) - \mathcal{I}(m_{q-1}, n_q)}) \tag{3.8}$$

For an image, the final PHOG descriptor is obtained by concatenating all HOG vector at each pyramid level for each p^{th} particle as $E_h^p = H_1, H_2 ... H_{\hat{N}_b}$. This depicts the spatial information for the image. Each HOG is normalized to obtain the final vector considering all the pyramid level. Zero level represents original image corresponding to b bins of each b HOG vector, level 1 by a $4b$ vector and the n^{th} level is represented as $\sum_{d=0}^{n} 2^{2d} = \sum_{d=1}^{n} 4^{2d}$. For each particle, Edge histogram (F_g) is determined by normalizing the value of the so obtained PHOG descriptor. Similarity between each cue histogram (F_l) for p^{th} particle and the corresponding reference histogram dictionary $\hat{D}_r$ is determined using Bhattacharya's distance. For this, Bhattacharya's coefficient is calculated using Eq. (3.9).

$$\beta_l^p(\hat{D}_{r,l}, F_l^p) = \sum_{n=1}^{\hat{N}} \sqrt{(\hat{D}_{r,l})^n \times (F_l^p)^n}, \quad l \in i, e, g \tag{3.9}$$

This Bhattacharya's coefficient is used for calculating the Bhattacharya's distance [85] between cue histogram (F_l) for p^{th} particle and its corresponding reference histogram $\hat{D}_r$ in the dictionary using Eq. (3.10).

$$D_l^p(\hat{D}_{r,l}, F_l^p) = \sqrt{1 - \beta_l^p(\hat{D}_{r,l}, F_l^p)}, \qquad l \in i, e, g \tag{3.10}$$

Bhattacharya's distance corresponding to each component of dictionary is averaged. Further, the likelihood for each cue for p^{th} particle is calculated using Eq.(3.11).

$$\gamma_l^p(\hat{D}_{r,l}, F_l^p) = \frac{1}{\sigma_l \sqrt{2\pi}} e^{-\frac{D_l^p(\hat{D}_{r,l}, F_l^p)^2}{2\sigma_l^2}}, \quad l \in i, e, g \tag{3.11}$$

where, σ_l represents the Standard deviation for Gaussian noise for each cue. The next section will detail the adaptive multi-cue fusion model.

3.3.3 Proposed Adaptive Multi-cue Fusion

Each cue likelihood is subjected to adaptive fusion model to obtain the fused weight $\hat{W}_f$. Non-linear ranking based score fusion method has been proposed for adaptive fusion of the cues. This model boosts the important particles and suppresses the unimportant particles automatically. For this, the likelihoods are obtained for each cue through Eq.(3.11) for p^{th} particle is assigned as initial weights using Eq.(3.12).

$$\hat{C}_{l,p} = \gamma_l^p(\hat{D}_{r,l}, F_l^p), \quad p = 1, 2, ...N \tag{3.12}$$

If N is the total number of particles and $l \in i, e, g$. Then, the individual weights for each particles of cue can be represented using Eq. (3.13).

$$\hat{C}_{l,N} = \begin{bmatrix} \hat{C}_{i1} & \hat{C}_{i2} & \dots & \hat{C}_{iN} \\ \hat{C}_{e1} & \hat{C}_{e2} & \dots & \hat{C}_{eN} \\ \hat{C}_{g1} & \hat{C}_{g2} & \dots & \hat{C}_{gN} \end{bmatrix} \tag{3.13}$$

Context reliability value for each cue is calculated at $t-1$ as $\hat{r}_t^l = (\hat{r}_{t-1}^i, \hat{r}_{t-1}^e, \hat{r}_{t-1}^g)$. These values are multiplied with each row corresponding to each cue is obtained using Eq.(3.14).

$$CS_{l,N}^{\hat{r}} = \begin{bmatrix} \hat{r}_{t-1}^{i} * \hat{C}_{i1} & \hat{r}_{t-1}^{i} * \hat{C}_{i2} & \dots & \hat{r}_{t-1}^{i} * \hat{C}_{iN} \\ \hat{r}_{t-1}^{e} * \hat{C}_{e1} & \hat{r}_{t-1}^{e} * \hat{C}_{e2} & \dots & \hat{r}_{t-1}^{e} * \hat{C}_{eN} \\ \hat{r}_{t-1}^{g} * \hat{C}_{g1} & \hat{r}_{t-1}^{g} * \hat{C}_{g2} & \dots & \hat{r}_{t-1}^{g} * \hat{C}_{gN} \end{bmatrix} \tag{3.14}$$

After this, particles are ranked based on the obtained score $CS_{l,N}^{\hat{r}}$ for each cue individually. For each particle, the rank is obtained as $R_j = R_N, R_{N-1}...R_1$. In each row, particle with highest score is ranked as the highest rank (R_N) and the particle with lowest score is given lowest rank (R_1). The rank matrix $W_{l,N}$ is obtained using Eq.(3.15).

$$W_{l,N} = \begin{bmatrix} R_j * \hat{C}_{i1}^{\hat{r}} & R_j * \hat{C}_{i2}^{\hat{r}} & \dots & R_j * \hat{C}_{iN}^{\hat{r}} \\ R_j * \hat{C}_{e1}^{\hat{r}} & R_{ij} * \hat{C}_{e2}^{\hat{r}} & \dots & R_j * \hat{C}_{eN}^{\hat{r}} \\ R_j * \hat{C}_{g1}^{\hat{r}} & R_j * \hat{C}_{g2}^{\hat{r}} & \dots & R_j * \hat{C}_{gN}^{\hat{r}} \end{bmatrix} \tag{3.15}$$

These scores are normalized row-wise using the min-max normalization. The normalized score matrix is represented by Eq. (3.16).

$$\bar{W_{l,N}} = \begin{bmatrix} W_{i1} & W_{i2} & \dots & W_{iN} \\ W_{e1} & W_{e2} & \dots & W_{eN} \\ W_{g1} & W_{g2} & \dots & W_{gN} \end{bmatrix} \tag{3.16}$$

The normalized score are subjected to $f : \bar{W_{l,N}} \rightarrow W_{fus}^{p}$ to obtain the fused score. $W_{fus}^{p} = [f(\bar{W_{i1}}, \bar{W_{e1}}, \bar{W_{g1}}), ...f(\bar{W_{iN}}, \bar{W_{eN}}, \bar{W_{gN}})]$. The function f for p^{th} particle is given by Eq.(3.17). Hence, using this, each particle is assigned weight after its evaluation over three cues.

$$\begin{aligned} W^p_{fus} = f(\bar{W_{ip}}, \bar{W_{ep}}, \bar{W_{gp}}) &= \frac{\bar{W_{ip}}}{1+\bar{W_{ip}} \times \bar{W_{ep}} \times \bar{W_{gp}}} \\ +\frac{\bar{W_{ep}}}{1+\bar{W_{ip}} \times \bar{W_{ep}} \times \bar{W_{gp}}} &+ \frac{\bar{W_{gp}}}{1+\bar{W_{ip}} \times \bar{W_{ep}} \times \bar{W_{gp}}} \end{aligned} \tag{3.17}$$

Next, the fused weight are passed through a non-linear function to boost the concordant cues and suppress the discordant cues using Eq.(3.18).

$$\hat{W}^p_{f,t} = \frac{e^{W^p_{fus}} - e^{-W^p_{fus}}}{e^{W^p_{fus}} + e^{-W^p_{fus}}} \tag{3.18}$$

Where, $\hat{W}^p_{f,t}$ is the final weight assigned to p^{th} particle at time t. These final weights are further subjected to an outlier detection mechanism to divide the particles into important particles ($\hat{I}_y$) and unimportant particles ($\hat{U}_x$). Unimportant particles are mainly affected due to environmental variations and are termed as outliers. These particles are detected by the outlier detection mechanism using Eq.(3.19).

$$\hat{U}_x = \quad \hat{W}^p_f, \qquad \textit{where} \quad \hat{W}^p_f < \tau_s \tag{3.19}$$

$$\hat{I}_y = \quad \hat{W}^p_f, \qquad \textit{where} \quad \hat{W}^p_f \geq \tau_s \tag{3.20}$$

where, τ_s is a defined threshold, $|x \cup y = N|$, $\hat{U}_x \in [X_{t,x}, Y_{tx,}]$ and $\hat{I}_y \in [X_{t,y}, Y_{t,y}]$. These outliers are further subjected to Butterfly Optimization Algorithm (BOA) [86] based resampling technique, which has been discussed in the next section.

3.3.4 Proposed Optimum Resampling Approach

BOA [86] is a nature-inspired meta-heuristic approach which disperses the outliers in the high likelihood area by its two parameters namely sensor modality ($\hat{m}_o$) and switch probability ($prob$). Sensor modality controls the search space and determines the convergence speed. Switch probability is used to switch between the local search and global search. BOA initializes the outliers as butterfly population (x) and their global position in the solution space is updated using Eq. (3.21).

$$U_x = \begin{cases} \hat{U}_x + (rand^2 \times b^{'} - \hat{U}_x) \times \mu_x & if \quad rand \leq prob \\ \\ \hat{U}_x + (rand^2 \times \hat{U}_j - \hat{U}_k) \times \mu_x, & \qquad otherwise \end{cases} \tag{3.21}$$

Here, $j, k \in x$, $rand \in [0, 1]$ and $U_x \in [X_{x,t}, Y_{x,t}]$. $b^{'}$ represents the best global position of the particles in the iteration. μ_x is calculated as: $\mu_x = \hat{m}_o I^a$. Here, I is correlated with the current position of the particles, $\hat{m}_o$ and a are the search space controlling parameters and varies from 0 to 1. Final centroid G_t of the target is estimated using fused weight $\hat{W}_f^p$ to that of outliers weight $\hat{W}_x$ and the important particles weight $\hat{W}_y$ using Eq.(3.22).

$$G_t = \frac{\sum_x \hat{W}_x \times U_x + \sum_y \hat{W}_y \times \hat{I}_y}{\sum_{p=1}^{N} W_p}, \quad |x \cup y = N| \tag{3.22}$$

where, $G_t = [X_t, Y_t]$ and U_x, $\hat{I}_y$ are the state determined by outliers and important particles respectively. Further, in order to ensure the quick adaptation of the proposed method during dynamic environment the reliability values are determined for each cue. Each cue reliability is estimated by calculating the L2-norm distance between the final estimated state and the state estimated through individual cue. L2-norm distance is calculated using (3.23).

$$d_{t,l} = ||G_t - G_t^l|| = \sqrt{(X_t - X_t^l)^2 + (Y_t - Y_t^l)^2} \quad \forall \quad l \in i, e, g \tag{3.23}$$

where, (X_t, Y_t) is the centroid of the final estimated state (G_t) and (X_t^l, Y_t^l) is the centroid of state determined through individual cue at time t. Using $d_{t,l}$ the cue reliability is calculated by Eq.(3.24).

$$\hat{r}_t^l = \frac{\tanh(-u(d_{t,l}) + h)}{2} + 0.5 \quad \forall \quad l \in i, e, g \tag{3.24}$$

where, u and h are constants. The context cue reliability $(\hat{r}_t)$ is calculated at state t and used for discounting the particles at state $t+1$. These reliability values lead to adaptive fusion of color, texture and edge likelihoods in the update model of the proposed tracker. Important particles $(\hat{I}_y)$ and Outliers particles $(\hat{U}_x)$ are updated at each state. Also, reference dictionary is updated by selective replacement of the important particles. This process is repeated iteratively to keep target's track until it is visible in the scene. Once it is lost, the whole tracker is re-initialized to re-detect the target. Details about the experimental validation of the proposed tracker are discussed in the next section.

3.4 Experimental Validation and Discussion

In order to analyze the performance of the proposed approach, we have chosen publicly available sample video sequences taken from OTB-100 dataset [66] and VOT dataset [87]. The chosen video sequences are rich in various environmental challenges such as background clutters, scale variations, deformation, full or partial occlusion, illumination variations, fast motion and motion blur. Results of the proposed tracker are analyzed against 13 other trackers viz. ASLA [88], MTT [89], CT [90], FRAG [91], IVT [92], MIL [93], WMIL [94], DFT [95], PF-PSO [44], PF [19], CSPF [5], MCPF [43] and STAPLE [96] . The proposed tracker is implemented in Python2 on a $i5$ quadcore 2.4 GHz processor with 8 GB RAM. Initially, the target is detected through GMM subtraction model [82] and $N = 49$ particles are instantiated on it. The dictionary is updated by selective replacement of the important particles. Tracker is executed 5 times iteratively to handle the probabilistic nature of the particle filter. Results depicted are average obtained over 5 iteration of the algorithm.

3.4.1 Attribute based Evaluation

In this section, the performance of the proposed tracker had been evaluated on 11 attributed challenges namely, illumination variations,background clutters, fast motion & motion blur, scale variations & deformation, in-plane & out-of-plane rotations and full or partial occlusion. The details of the tracking challenges

Table 3.1: Tracking challenge and considered video sequence

Tracking Challenge	**Video Sequences**
Illumination variations	*Fish, Human8, Skating1, Singer1, Car2, Soccer1, Singer2, Tiger, Shaking, Basketball, Crossing*
Background clutter	*Basketball, Car2, Pedestrian1, Football, Shaking, Singer2, Bolt2, MountainBike, Skating1*
Fast Motion and Motion Blur	*CarScale, Pedestrian1, Jumping, Soccer1, Jogging1, Tiger, Human7, Crossing*
Scale variations & Deformation	*CarScale, Shaking, Dancer, Human8, Walking, Jogging1, Human7, Bolt2, Skating1, Basketball, Pedestrian1*
In-plane & Out-of-plane rotation	*Basketball, CarScale, Pedestrian1, Football, Shaking, Singer2, Skating1, Soccer1, Tiger, Dancer, MountainBike*
Full or partial occlusion	*Football, CarScale, Subway, Walking2, Jogging1, Singer1, Jogging2, Tiger, Basketball*

and the considered video sequence is tabulated in Table 3.1. Sample tracking frames from the considered challenging video sequences under various attributed challenges is depicted in Fig. 3.2.

Illumination Variations: In *Car2* sequence, at frame #380 most of the trackers have shown drift whilst OURS, Frag and STAPLE are able to manage the change in illumination in the whole sequence. At frame #128 of *Human8*, when there is a sudden variation in illumination, most of the trackers lose the target but the proposed tracker with STAPLE and MCPF are able to locate the target till the end. Here, the PHOG cue enhances proposed tracker's performance by compensating for color and LBP cues. Another challenging sequence under this challenge is *Shaking*. Initially, all the trackers are able to keep track of the target with the marginal error. However, at frame #358 CT, FRAG, PF-PSO, IVT and MTT

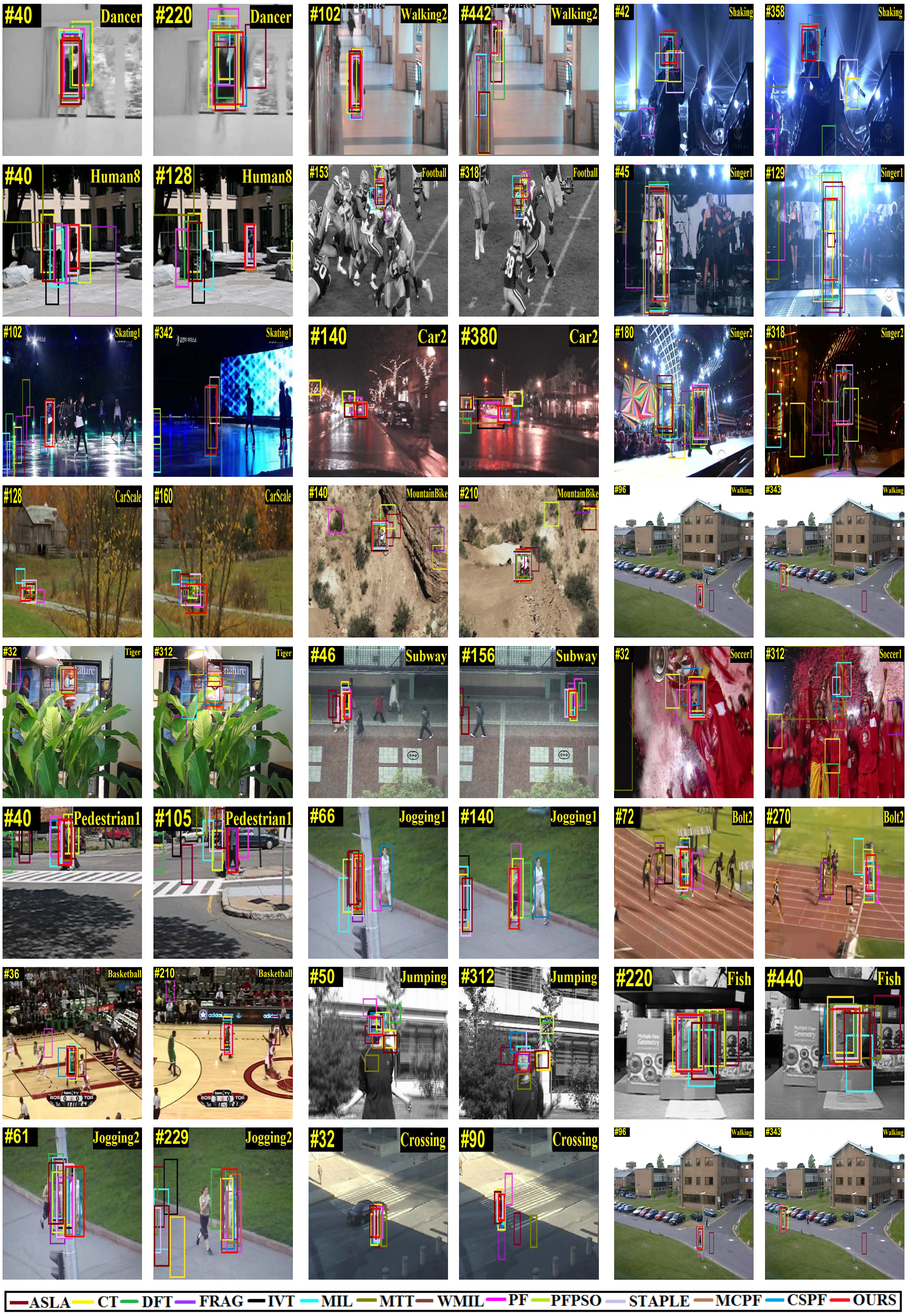

Fig. 3.2 *Representative frames from the considered challenging video sequences. Each frame has been labeled with # frame number and video sequence name on the left and right corner, respectively.*

have lost the target. OURS with MIL, WMIL, ASLA, and MCPF are able to keep track the target. This is due to the automatic boosting of good particles and suppression of the unimportant particles by the proposed fusion model. In *Singer2*, at frame #180, only our tracker, CSPF and STAPLE are able to locate the target while others have shown deflection. In *Singer1*, OURS, CSPF and MCPF have shown substantially good performance in comparison to other trackers. *Fish* and *Soccer1* sequences have abrupt illumination variations in which OURS and MCPF have much better results in comparison to other trackers. In sum, the proposed tracker have performed with substantially low error under this challenge which is due to the considered complementary cues in the appearance model. In addition, consistent updating of the reference dictionary with the important particles improve the performance of the proposed tracker under illumination variations. Also, the proposed tracker have achieved highest AUC of 0.696 depicted in Fig. 4.8 (a) and DP score of 0.761 illustrated in Fig. 4.9 (a) under the challenge.

Background Clutters: The sequences viz. *Crossing*, *Car2*, *Shaking* and *Singer2* have similar background to that of the target . For *crossing* sequence, when there is similar background at frame #90 ASLA, FRAG, MTT and PF lose the target whilst OURS, CSPF, MCPF and STAPLE locate the target with very small error. *Pedestrian* sequence have background clutter at frame #105 when the target suffers from background clutter then CT, FRAG, PF-PSO, MCPF, and MIL have lost the target and start tracking the similar objects in the background but OURS track the target successfully. This is due to the update of important particles and suppression of unimportant particles by adaptive multi-cue fusion. In *Tiger*

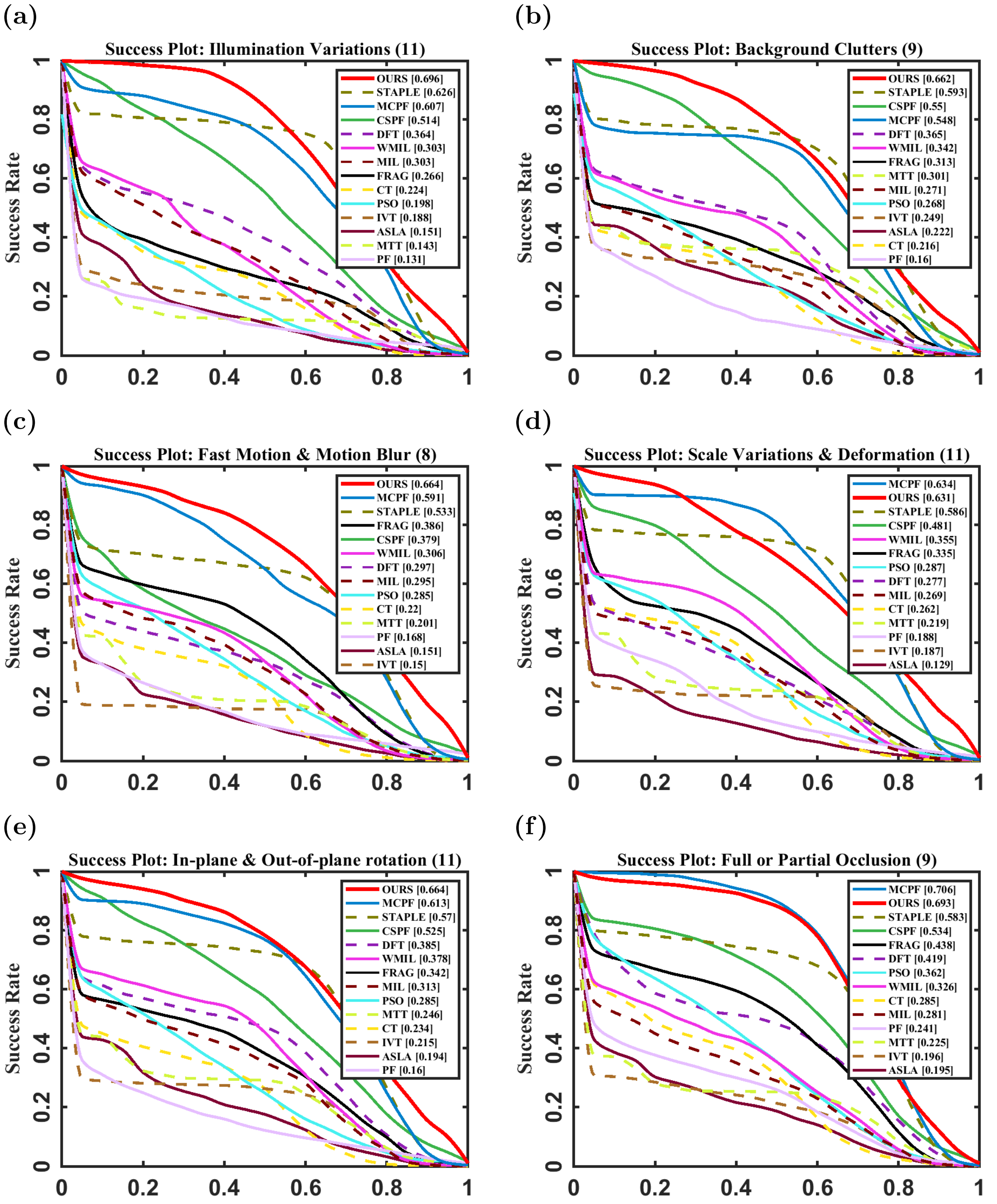

Fig. 3.3 *Success plot under tracking challenges as (a) Illumination variations (b) Background Clutters (c) Fast Motion and Motion Blur (d) Scale variations and Deformation (e) In-plane and Out-of-plane rotation (f) Full or partial occlusion. Legend includes Area Under Curve (AUC) in the bracket.*

sequence, trackers viz. CT, ASLA, FRAG, MTT and MCPF could not able to handle the tracking challenge. In *Walking* sequence, MTT, STAPLE and MCPF have shown better performance in comparison to the proposed tracker. This may be due to the false update of the particles due to similar background in the proposed tracker's appearance model during tracking. *Bolt2* and *Skating1* sequences have several pose variations with similar background. OURS keep track of the target in the whole sequence. This is owed by the robust complementary LBP and PHOG features over color. In sum, the robustness of the proposed tracker under background clutter is mainly due to the proposed adaptive fusion model which boosts the important particles and suppresses the unimportant particle during the challenge. Unimportant particles are detected as outliers by the proposed outlier detection mechanism and their position is further improved by the proposed resampling method. In addition, cue reliability is also discounted for the quick adaptation of target during background clutters. Under Background clutters challenge, the proposed tracker have attained highest AUC of 0.662 illustrated in Fig. 4.8 (b) and DP score of 0.668 illustrated in Fig. 4.9 (b).

Fast Motion and Motion Blur: Generally, fast motion and motion blur occur in a sequence either by abrupt target's motion or camera movement. *Pedestrian1* and *Human7* sequences have motion blur due to sudden camera motion. OURS have handled the challenge gracefully in these sequences. It is due the rotational factor in the state model and the detection of the affected particles by the proposed outlier detection mechanism. In *Tiger* and *Soccer1* sequences, target is moving very fast and trackers viz. STAPLE, CSPF, MIL, MTT and ASLA are not able to keep track of the target. But, OURS and MCPF have shown considerably good performance. *Jumping* sequence have tough tracking scenario with both fast motion and motion blur. The proposed tracker performs well in comparison to

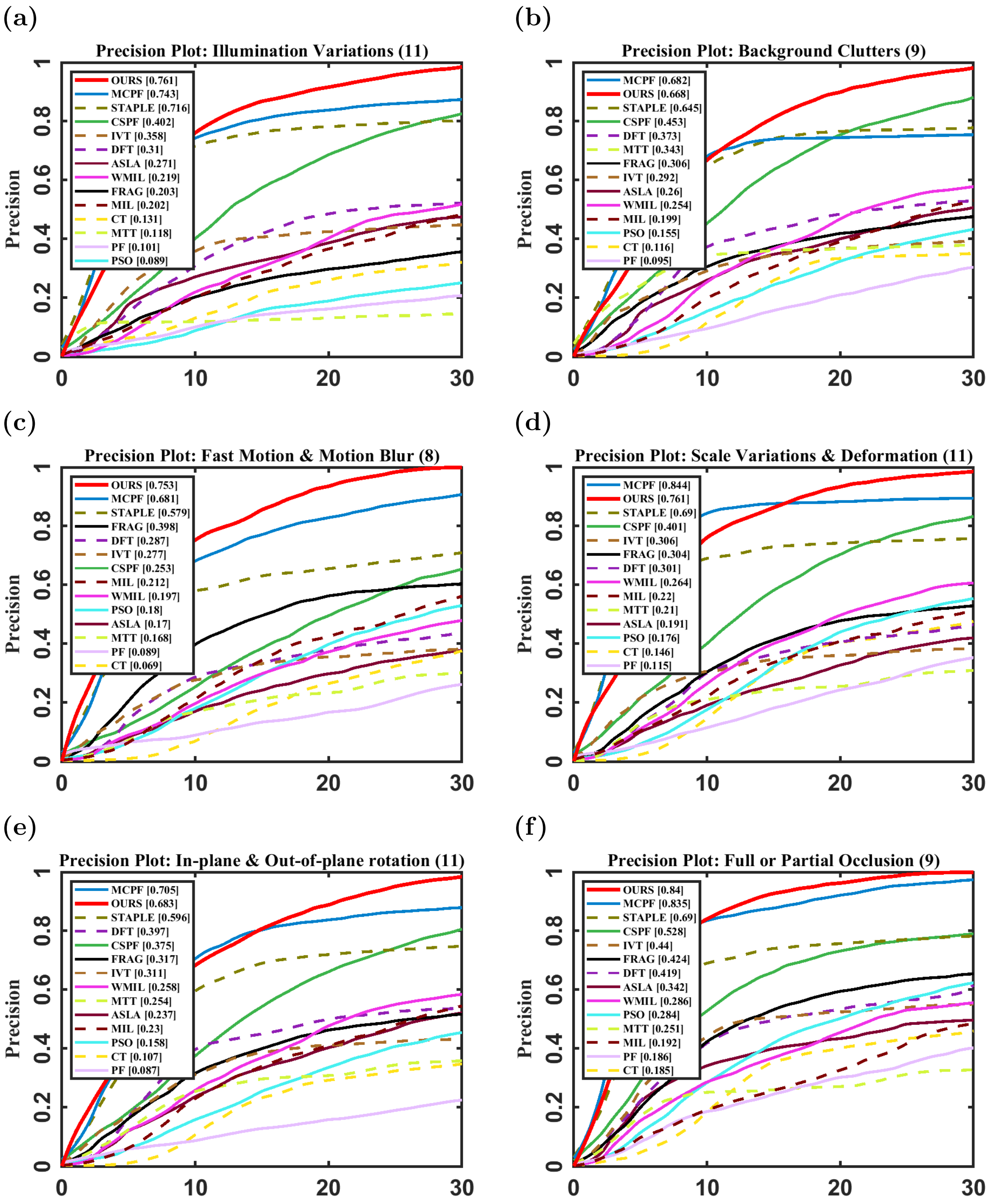

Fig. 3.4 *Precision plot under tracking challenges as (a) Illumination variations (b) Background Clutters (c) Fast Motion and Motion Blur (d) Scale variations and Deformation (e) In-plane and Out-of-plane rotation (f) Full or partial occlusion. Legend includes average Distance Precision (DP) score in the bracket.*

other trackers by the proposed fusion model which suppresses the affected particles and boosts the good particles. Under fast motion & motion blur challenge, the proposed tracker have achieved highest AUC of 0.664 as shown in Fig. 4.8 (c) and highest average DP score of 0.753, shown in Fig. 4.9 (c).

Scale Variations and Deformation: For sequence *Dancer*, the proposed tracker with MTT, IVT, CSPF, STAPLE and MCPF are able to handle the target deformation due to scale variations. However, MIL and WMIL are slightly deflected from the target. At frame #140, in *MountainBike* sequence target have scale variations and deformation due to its sudden movement which PF, FRAG, CT and WMIL could not able to handle. OURS, ASLA, MIL, MCPF and STAPLE have handled this variation with minimal error. In *CarScale* sequence, at frame #160 most of the tracker lose the target as these tracker's appearance model is inefficient to handle large scale variation. But, MCPF and STAPLE have performed marginally better than the proposed tracker. In *Basketball*, ASLA, MIL, CT, MTT and DFT lose the target whilst OURS with FRAG, WMIL,CSPF and MCPF have shown better performance. This is due to the scale-invariant LBP feature in the proposed tracker which have handled the target variations. OURS have shown better performance in comparison to others in *Walking2* throughout the whole sequence. It may be due to the scaling factor of the state model which have catered the scale variation in the sequence. In sum, OURS is robust under scale variations which is mainly attributed by the incorporation of scale-invariant LBP feature in the appearance model. Moreover, the incorporation of the scaling factor in the random walk model of the state model have strengthened our method under this challenge. Under the challenge, the proposed tracker have attained the second highest AUC score of 0.631 depicted in Fig. 4.8 (d) and second highest average DP score of 0.761 depicted in Fig. 4.9 (d).

In-plane and Out-of-plane rotation: These rotations appear frequently in a sequence when the target move in or out of the image plane. In sequences, *Basketball*, *Pedestrian1* and *Football* the target rotate in and out of the image plane suddenly. The proposed tracker has handled these variations by adaptive boosting of the LBP and PHOG features over color. In *Singer2* and *Soccer1*, most of the trackers are not able to keep track of the target. However, OURS has shown substantially better performance. OURS, CSPF, MCPF and STAPLE have shown good tracking results in comparison to other trackers in the sequence *Tiger* and *Shaking*. The proposed tracker is able to handle the challenge to a great extent. The rotational component in the state model of the proposed tracker and the complementary cues, color, LBP and HOG in the appearance model of the tracker enhances its performance under the challenge. Moreover, the proposed tracker outperforms the other trackers by achieving the highest AUC of 0.664 as illustrated in Fig. 4.8 (e) and second highest DP score of 0.683 as illustrated in Fig. 4.9 (e) under the challenge.

Full or Partial Occlusion: Severe occlusion have been noticed in the *Jogging1* and *Jogging2* sequence, when the pillar is occluded the moving target. Most of the trackers have shown drift while the proposed tracker and MCPF are able to track the target after it recover from the full occlusion. *Soccer1* sequence have multiple full and partial occlusion challenge. At frame #312 when the target is partially occluded by the similar objects most of the tracker are deflected from the target. However, the proposed method gracefully handle the challenge and track the target successfully. This is due to the complementary cues exploited in the appearance model. There are multiple full and partial occlusion of the target in *Subway* sequence. At frame #156 when the target is occluded, then ASLA, WMIL and MTT totally lose the target. Only OURS, CSPF, MCPF and STAPLE can keep track of the target even after occlusion. Generative based trackers are not able to handle occlusion as their appearance model is inefficient in handling this challenge. For *Football* sequence, when target is occluded by the other players then OURS with MCPF track the target successfully. This is due to the selective boosting of the color and LBP features and suppression of the PHOG by the proposed fusion model during occlusion. In *Tiger*, the proposed tracker with CSPF have shown better tracking results in comparison to other trackers. The adaptiveness of the proposed tracker under full or partial occlusion is primarily due to the proposed fusion model which suppressed the unimportant particles whose weight decreased due to occlusion and boosted the other important particle. These unimportant particles are detected by the outlier mechanism and hence, prevents erroneous update of the tracker due to occlusion. These particles are further subjected to the proposed resampling method in order to enhance their contribution to the state estimation. Also, as shown in Fig. 4.8 (f) and Fig. 4.9 (f) the proposed tracker have achieved AUC of 0.693 and average DP score of 0.840, respectively under full or partial occlusion challenge.

TABLE 3.2: Comparison of contribution of each feature

Video Sequence	**Color**	**LBP**	**PHOG**	**CH+LBP**	**LBP+PHOG**	**CH+PHOG**	**Ours**
Car2	41.82	25.06	18.29	28.5	19.61	41.12	5.15
MountainBike	49.36	165.76	85.64	18.36	16.22	15.81	7.69
Jogging1	29.12	26.77	24.55	11.24	13.13	13.55	6.85
Crossing	11.77	11.85	75.51	13.84	13.66	34.13	4.77
Shaking	121.48	183.42	49.08	29.6	31.97	27.51	18.25
CarScale	39.14	94.88	36.83	18.44	26.93	17.44	10.85
Soccer1	199.9	118.2	174.49	25.09	25.87	28.04	11.31
Pedestrian1	43.16	43.91	25.65	22.09	20.74	17.96	12.26
Human8	29.43	68.85	29.02	29.43	18.71	18.84	9.67
Dancer	34.23	69.06	40.43	17.8	16.74	26.16	4.1
Subway	46.98	44.98	38.56	10.5	11.82	7.98	6.19
Tiger	219.4	107.55	210.91	37.69	39.69	43.7	7.28

3.4.2 Analysis of Feature Contribution

The performance of the proposed method considering individual feature and all possible multiple combination of the features in terms of CLE has been tabulated in Table 3.2. Challenging video sequences are considered in order to prove the robustness of exploiting three complementary multi-cue in the appearance model of the proposed tracker. Results infer that single cue is inefficient in handling the various environmental variations. Multiple cues namely, color and LBP, LBP and PHOG or PHOG and color when integrated together then, performance of the tracker improves marginally. However, when these cues are combined using the proposed fusion approach, performance of the tracker substantially enhanced by addressing the various environmental challenges.

3.4.3 Computational Complexity

In order to analyze the real-time computational complexity of the proposed tracker, we have calculated the average computational speed score of the proposed tracker and compare the results with other trackers. Average speed score is calculated by average fps obtained over all the video sequences. Table 3.3 tabulates the computational speed analysis of the proposed method in comparison to the other methods. It has been evident from the table that the proposed tracker had shown real-time performance with high accuracy. The proposed tracker have shown comparable performance against other methods and outperforms trackers namely, ASLA, MIL, DFT, WMIL, CT, FRAG, STAPLE, MCPF, PSO and CSPF in terms of accuracy by 89.41%, 45.65%, 27.64%, 35.56%, 81.73%, 83.29%, 16.88%, 13.12%, 79.57% and 48.77% respectively. Although, MCPF have shown comparable performance to the proposed tracker but it is computationally slow.

TABLE 3.3: Tracking Speed Analysis (FPS)

Tracker	ASLA	MIL	DFT	WMIL	CT	FRAG	STAPLE	MCPF	PSO	CSPF	OURS
Speed (FPS)	16	7	9	18	19	14	18.4	0.57	15	**21**	20
Accuracy	0.0745	0.3823	0.5090	0.4533	0.1285	0.1175	0.5847	0.6112	0.1437	0.3604	**0.7035**

3.4.4 Overall Performance Evaluation

TABLE 3.4: Comparison results for average CLE. First, Second and Third results are shown, respectively.

Challenge	ASLA	MTT	MIL	DFT	WMIL	IVT	CT	FRAG	PSO	PF	MCPF	STAPLE	CSPF	OURS
Illumination Variations	70.93	162.236	62.16	63.66	63.32	79.07	83.00	78.75	69.43	178.317	23.36	29.27	17.53	7.87
Scale variations & Deformation	87.11	108.78	56.52	66.69	64.99	67.79	73.74	60.26	48.06	108.74	26.27	34.44	17.74	7.68
Fast Motion & Motion Blur	69.20	96.25	42.93	55.03	53.28	74.12	63.11	49.85	52.30	98.97	10.82	30.71	27.19	7.03
Background Clutters	73.06	111.88	64.17	68.28	81.35	78.98	103.13	87.98	56.88	135.82	52.24	33.11	14.56	9.22
Full or partial Occlusion	77.13	142.25	61.92	33.81	64.03	66.66	59.95	38.28	45.36	131.111	5.92	33.28	19.04	6.31
In-plane and Out-of-plane rotation	74.47	124.58	60.50	67.15	70.11	81.64	96.49	83.90	69.18	149.31	24.24	34.60	18.79	8.88
Overall	**76.92**	**108.08**	**50.49**	**47.91**	**66.43**	**65.59**	**70.35**	**61.10**	**46.75**	**110.27**	**24.19**	**28.26**	**16.55**	**6.89**

In order to evaluate the overall performance of the proposed approach, Table 3.4 and Table 3.5 have tabulated the average CLE (in pixels) and average F-Measure

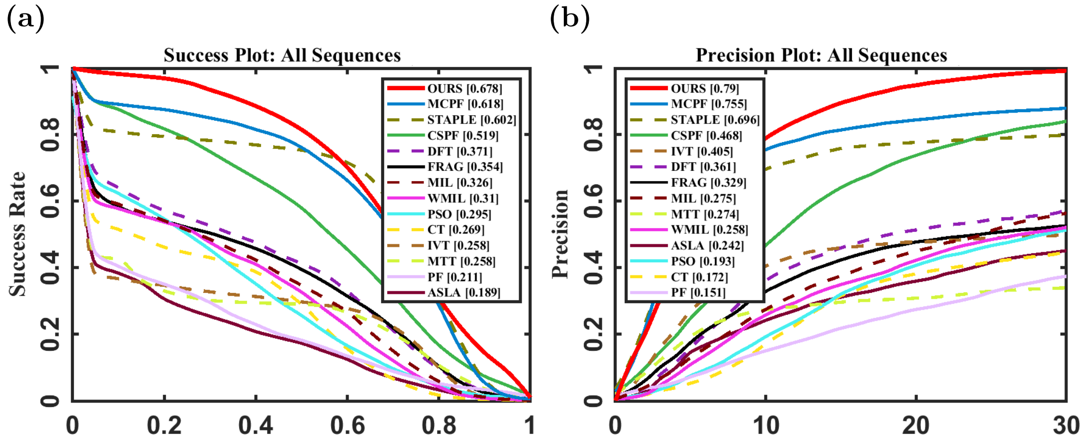

Fig. 3.5 *Overall performance comparison of the proposed tracker with other trackers: (a) Success plot (b) Precision plot. Legend includes the AUC for success plot and average DP for precision plot respectively, in the brackets.*

TABLE 3.5: Comparison results for average F-Measure. First, Second and Third results are shown, respectively.

Challenge	ASLA	MTT	MIL	DFT	WMIL	IVT	CT	FRAG	PSO	PF	MCPF	STAPLE	CSPF	OURS
Illumination Variations	0.193	0.153	0.378	0.434	0.389	0.205	0.278	0.319	0.259	0.148	0.708	0.701	0.634	0.804
Scale variations & Deformation	0.155	0.259	0.332	0.339	0.437	0.200	0.332	0.411	0.371	0.235	0.739	0.661	0.587	0.734
Fast Motion & Motion Blur	0.186	0.242	0.367	0.350	0.376	0.156	0.277	0.466	0.365	0.203	0.700	0.602	0.464	0.764
Background Clutters	0.271	0.334	0.332	0.439	0.417	0.277	0.268	0.367	0.344	0.294	0.630	0.671	0.669	0.772
Full or partial Occlusion	0.241	0.254	0.347	0.503	0.401	0.222	0.365	0.526	0.463	0.197	0.815	0.662	0.632	0.799
In-plane and Out-of-plane rotation	0.244	0.289	0.384	0.457	0.462	0.235	0.293	0.409	0.366	0.193	0.715	0.647	0.639	0.769
Overall	**0.232**	**0.294**	**0.402**	**0.450**	**0.385**	**0.288**	**0.338**	**0.425**	**0.381**	**0.258**	**0.716**	**0.681**	**0.628**	**0.786**

of the proposed tracker along with 13 other state-of-the-art, respectively. The proposed tracker has achieved an average mean CLE of 6.89 (in pixels) and average mean F-Measure of 0.786 when analyzed on challenging videos. Overall success and precision plots are depicted in Fig. 3.5. Success plot illustrates that the proposed tracker has achieved high AUC score of 0.678 and precision plot depicts the high precision score of 0.790 at 10px for our tracker.

In sum, low CLE value and high F-Measure of the proposed tracker reveals its robustness against various environmental challenges. Also, Success vs overlap threshold plots and precision vs CLE threshold (in pixel) plots indicates that the proposed tracker is able to locate the target precisely. The drawbacks of particle filter viz. particle degeneracy and sample impoverishment are handled by the proposed butterfly optimization based resampling technique. Our tracker is computationally efficient as it only resampled the outlier particles. Unimportant particles which contribute less in the state estimation or are affected due to illumination variation, occlusion, background clutters, fast motion and in-plane and out-of-plane rotations were detected as outliers by the outlier detection mechanism. In addition, the proposed adaptive fusion model integrates the multi-cue weight through context cue reliability which ensures consistent updation of the tracker. The fusion model boosts the important particles and suppresses the unimportant particles for the final state estimation. Our tracker can deal with various tracking challenges to a great extent. On the other hand, generative trackers viz. IVT, FRAG and MTT tend to drift under full occlusion and background clutters. Discriminative trackers viz. MIL and CT are prone to illumination and scale variations. Particle filter based trackers viz. PF and PF-PSO have limited performance due to the limitation of the resampling method. WMIL has exploited random Haar features and is not able to handle the object's deformation and rotation. STAPLE is not able to cater dynamic appearance variations. CSPF is susceptible to fast motion and motion blur. MCPF have shown comparable performance but it is not suitable for real-time tracking. However, Our tracker achieves real-time performance with high accuracy under various environmental variations.

3.5 Significant Findings

The following were significant findings of the proposed multi-cue object tracking model under stochastic framework.

- A real-time multi-cue object tracking solution using particle filter under stochastic framework by adaptive fusion of the cues had been proposed.
- Adaptive fusion model boosted the important particles and suppressed the unimportant particles to handle occlusion, background clutters and motion blur.
- Outlier detection mechanism was able to disparate the low performing particles affected due to environmental variations. These particles were relocated to high likelihood area by the proposed resampling method.
- The meta heuristic optimization based resampling method not only addressed the drawbacks of particle filter but also enhanced tracker's performance.
- Multi-cue appearance model was efficient in handling the illumination variations and scale variations. Inclusion of scaling and rotation factor in state model catered the change in scale, fast motion, in-plane and out-of-plane rotation of the target.
- Context sensitive cue reliability discounted the particles for the quick adaptation of the proposed approach to the dynamic environment.

- On average of the outcome, our tracker achieved precision score of 0.790, center location error of 6.89 (in pixels), F-measure of 0.786, and success rate of 0.678 when evaluated on OTB-100 and VOT datasets against 13 others state-of-the-art.
- The proposed tracker exhibits real-time computational performance of 20 frames/sec.

The experimental results alongwith other findings were published in [97].

In addition, the application of meta-heuristic optimization technique, chaotic crow search algorithm had been studied as a resampling approach for addressing particle filter drawbacks. Results pertaining to one dimensional problem and 2-D bearing only tracking problem were published in [98].

Chapter 4

Deterministic Framework for Multi-cue Object Tracking

Chapter 4

Deterministic Framework for Multi-cue Object Tracking

The aim of this work is to propose a multi-cue object tracking model under deterministic framework. Deterministic framework focus on cost minimization by exploiting both foreground and background information for providing robust tracking solutions. The framework includes tracking methods based on Mean shift, tracking by detection and fragment based tracking [56], [23]. Under this, multi-cue object tracking method in fragment based architecture has been proposed in which the features were adaptively fused using a fuzzy based fusion model.

4.1 Introduction

Recently, multi-cue object tracking algorithms under deterministic framework had been reviewed extensively. Trackers based on deterministic framework exploits both foreground as well as background information for providing efficient tracking solutions [20], [99]. In this direction, Zhang et al. [20] modeled the discriminative appearance of the target using Fisher vectors. Linear kernel classifier was used to classify the target from the background. In [21], authors proposed an observation model based on dual features and ICA with hierarchical reference maps. Stochastic and periodic update was used for keeping track of variations in target's appearance. Yu et al. [100] proposed an appearance model in which semi supervised random forest was used to preserve the similarity between the foreground and background. Target's structure information was presented using a patch based grid structure. Zhou et al. [101] proposed class specific dictionary learning to distinguish the target from the background and adopted a mechanism to capture the outliers. However, in [102], authors utilized dual discriminative dictionary to store the background template information and manifold regularization to calculate the similarity between the reference template and the target template. Dual dictionaries improved the discriminating ability of the tracker but their regular update reduced the processing of the tracker. Xu et al. [99] exploited supervised tensor flow learning with an support tensor classifier for developing discriminative tracker. Tracker demonstrated superior performance in presence of occlusion and background clutters. Despite this, online update, feature extraction and tracker's training required a lot of processing power.

In addition, the methods considering complementary multi-cue in the tracker's appearance model were also proposed under deterministic framework. Complementary multi-cue were visual cues which compensated for each other if performance of any cue deteriorated due to tracking challenges. Multi-cue based trackers integrated the multiple extracted cue utilizing either score fusion or feature fusion.

Score fusion combined the different scores obtained from multiple classifier to segment the target from the background. Under this, multicue tracker [35], multistage tracker [23], and weighted part model [103] considered score fusion. However, feature fusion integrated the different extracted feature vector from each cue into a single feature vector. In this direction, Lan et al. [24] exploited feature level fusion using joint sparse representation for integrating reliable features. Intensity, texture and HOG features were fused into a unified robust feature by graph cross diffusion mechanism [25]. Chi et al. [21] considered edge, shape and geometric features for object representation. Li et al. [104] performed weighted feature fusion on the basis of similarity variance between the target and candidate templates. In sum, feature fusion had been investigated in many recent object tracking works under deterministic framework. However, feature fusion is inefficient in capturing the target's non-linear variations that occur due to change in environment. Hence, feature fusion for developing a adaptive robust appearance model under deterministic framework can be further explored.

4.2 Proposed Tracker Architecture

The proposed object tracking architecture exploits multiple visual descriptors for the object's localization in the video sequences. Initially, the tracker is initialized with positive samples ($\mathcal{P}^+$) and two types of negative samples ($\mathcal{N}^-$, $\mathcal{N}^{--}$) in the reference dictionary ($\mathcal{D}_r$). $\mathcal{P}^+$ and $\mathcal{N}^-$ are obtained by sampling the target and the target's background area while $\mathcal{N}^{--}$ is captured by sampling the other target in the

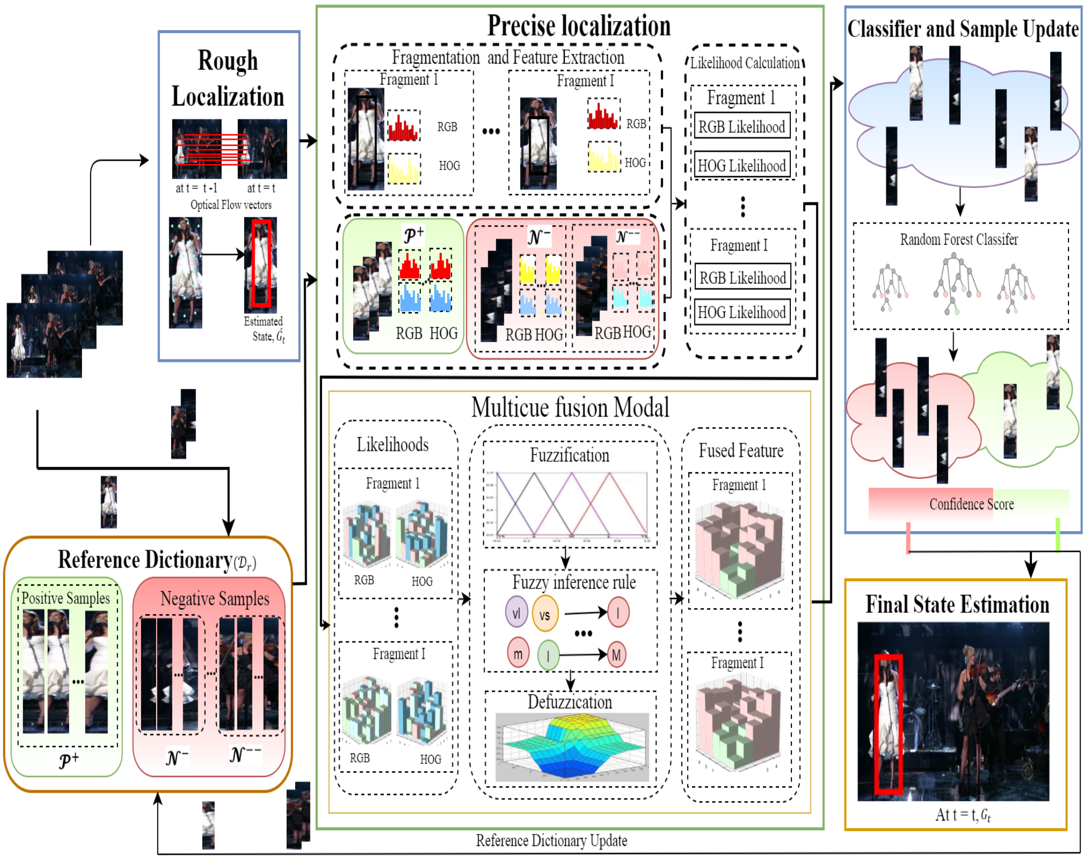

Fig. 4.1 *Overview of the proposed tracker architecture. At each time step (t=1), robust unified feature is generated by the multi-cue fusion model for each positive and negative fragments. Complementary features namely, color and HOG are extracted and fragments are initialized on the target after rough localization. Tracker is made adaptive considering random forest classifier and reference dictionary update.*

scene. This set of negative samples improves the tracker's discriminative ability in presence of the other target in the scene. In addition, the proposed tracker employs a robust fusion model and random forest classifier ($\mathcal{R}_c$) to adapt to the dynamic environmental variations. Fig. 4.1 illustrates a detailed methodology with the core design architecture of the proposed tracker. The proposed tracker adopts two stage estimation approach for better estimation of the target in the video sequences. During the first stage, the target is roughly localized ($\acute{G}_t$) exploiting the motion cue extracted using Horn-Schunk optical flow method. The next stage of precise localization improves the previously localized centroid of the target. For this, multi-cue features namely, color (M_r) and HOG (M_o) are extracted for target appearance model. These extracted features are further subjected to the proposed fuzzy based fusion model which assures that the target's eminent features are captured well in the robust unified feature ($\mathcal{US}$). The conflict between the cue is resolved by introducing two parameters namely, fuzzy nearness (F_n) and correlation coefficient (Cor) in the fuzzy inference model. After this, the fragments are passed through a random forest classifier ($\mathcal{R}_c$). This classifier classifies the fragments into positive and negative fragments. The final state (G_t) of the target is determined by the weighted mean of the centroid of fragments with high confidence score. The tracker is temporally made adaptive to environment variations with the selective replacement of the samples in the reference dictionary ($\mathcal{D}_r$). Detailed design of the rough localization is discussed in the next section.

4.2.1 Multi-cue Feature Extraction

During rough localization, the target's location is promptly estimated by evaluating the motion cue using the optical flow method. Optical flow method determines the motion descriptor by considering the change in pixel intensity and their related movement between the consecutive frames. In the proposed tracker architecture, Horn-Schunk (HS) optical method [105] has been utilized for quick estimation of the target centroid during initial stage. HS method assumes that there is no illumination variation between the consecutive frames and calculates $\mathcal{I}(x, y, t)$ using

Eq.(4.1).

$$\mathcal{I}_x p_x + \mathcal{I}_y q_y + \mathcal{I}_t = 0 \tag{4.1}$$

where, $\mathcal{I}_x$ and $\mathcal{I}_y$ are spatial gradient of intensity, p_x and q_y are change in pixel displacement w.r.t time in x-direction and y-direction, respectively. $\mathcal{I}_t$ is change in pixel intensity w.r.t time t. In order to minimize the total error in optical flow components, the HS method includes smoothness constraints with spatial gradient. The minimizing total error ϵ_t is given by Eq.(4.2).

$$\epsilon_t^2 = \int\int (\mathcal{I}_x p_x + \mathcal{I}_y q_y + \mathcal{I}_t)^2 + \upsilon^2 \left(\left(\frac{\partial p_x}{\partial x}\right)^2 + \left(\frac{\partial p_x}{\partial y}\right)^2 + \left(\frac{\partial q_y}{\partial x}\right)^2 + \left(\frac{\partial q_y}{\partial y}\right)^2 \right) dxdy \tag{4.2}$$

where, υ^2 is the weighting factor that measures the smoothness constraint term. If O_v optical flow vectors are considered for evaluation around the precise localized target centroid G_{t-1} in the previous frame, then the target's rough centroid $\acute{G}_t$ in the present frame is localized using the Eq. (4.3).

$$\acute{G}_{t,(x,y)} = G_{t-1,(x,y)} + \frac{1}{O_v} \sum_{j=1}^{N} (p_x^j, q_y^j) \tag{4.3}$$

where, (p_x^j, q_y^j) are the optical flow vectors for j^{th} pixel in x-direction and y-direction, respectively. After rough localization, during the next stage of precise localization the target's centroid is computed by extracting multiple features from the candidate fragments. Candidate fragments $\mathcal{F}_i \in \{\mathcal{F}_1, \mathcal{F}_2 ... \mathcal{F}_I\}$ are initialized by sampling the target around the previous localized centroid $\acute{G}_t$ through random walk model. Each random walk model has a scale factor(s) and angular displacement (ϕ) to address the scaling and rotational variation of the target. For this, the state $\acute{G_t} = (s, \phi)^c$ for each candidate fragment is propagated through random walk model using Eq.(4.4).

$$\acute{G}_{t+1} = \acute{G_t} + \vartheta \tag{4.4}$$

where, $\vartheta \sim g(0, c)$ is zero mean Gaussian noise and c is a covariance matrix depicting state vector uncertainty. Each candidate fragments is scaled and rotated using Eq.(4.4) and then subjected to feature extraction. In the proposed tracker, multi-cue complementary features namely, color and HOG are extracted for each candidate fragments to describe its appearance characteristics. The details of the multiple feature extraction are Color is an efficient feature for object tracking as it is invariant to object's scaling and can cater partial occlusion challenge. Also, color requires low computational power for its extraction. For each candidate fragment, color cue is extracted using RGB color histogram model (CH). The CH $M = \{M_1, M_2, ...M_{N_b}\}$ for each pixel locations $\{(a_1, b_1), (a_2, b_2)...(a_n, b_n)\}$ in the fragment is calculated using Eq.(4.5).

$$M^i = \alpha \sum_{j=1}^{n} \delta(I(a_j, b_j)), \qquad i = 1, 2, ...N_b \tag{4.5}$$

where, $I(.)$ assigns each pixel (a_j, b_j) to one of the N_b bin, $\delta(.)$ is Kronecker-delta function and α is the normalizing factor determined as $\sum_{i=1}^{N_b} M^i = 1$. For i^{th} fragment, the color histogram is determined as M_r^i.

Unlike color, the HOG feature is invariant to illumination variations and can handle background clutters efficiently. HOG [106] defines the shape structures of each candidate fragments and hence, can acquire the distribution of horizontal gradient intensity and vertical gradient intensity w.r.t edge directions. For this, the image is filtered with kernel functions depicted in Eq.(4.6).

$$[-1,0,1] \quad and \quad [-1,0,1]' \tag{4.6}$$

The magnitude ω and direction Φ for horizontal gradient values $\mathcal{I}_x$ and vertical gradient values $\mathcal{I}_y$ for each pixel (a,b) can be determined using Eqs.(4.7) and (4.8).

$$\omega(a,b) = \sqrt{(\mathcal{I}_x(a,b))^2 + (\mathcal{I}_y(a,b))^2} \tag{4.7}$$

$$\Phi(a,b) = \arctan \frac{\mathcal{I}_x(a,b)}{\mathcal{I}_y(a,b)} \tag{4.8}$$

Each candidate fragment region in the image is divided into small spatial regions called cell. Each cell has an edge orientation histogram and each pixel (a,b) in the cell casts a weighted vote for its orientation histogram and neighboring pixels orientation histogram. If N_p defines the number of pixels in a cell, then the bin (ξ) for each histogram cell (d) can be computed using Eq.(5.10).

$$\hat{M}_d(\xi) = \sum_{i=1}^{N_p} \omega(a,b)\delta(\dot{\Phi}(a,b) - \xi) \tag{4.9}$$

where, $\dot{\Phi}(a,b)$ is quantized orientation calculated from $\Phi(a,b)$ and $\delta(.)$ is Kronecker-delta function. Further, gradient values in each cell are normalized using L2-norm to cope with illumination variation and shading effect and L2-norm is computed by Eq. (5.11).

$$\hat{M}_o(\xi) = \frac{\dot{M}_o(\xi)}{\sqrt{\sum_{j=1}^{d\times d\times N_b} \dot{M}_o(j)^2 + \nu^2}} \tag{4.10}$$

where, ν is a regulation parameter, d and N_b are the total number of cells and number of bins per cell, respectively. After normalization, the gradient histogram of bins for i^{th} candidate fragment is stored as M_o^i as HOG feature.

Further, the similarity between each candidate fragment and the samples stored in the reference dictionary is determined. At $t = 1$, $\mathcal{P}^+$, $\mathcal{N}^-$ and $\mathcal{N}^{--}$ are saved in the dictionary $\mathcal{D}_r$. The Bhattacharya distance [**?**] is calculated between the extracted features $M_k \in \{M_r, M_o\}$ of the candidate fragment and the corresponding feature from each sample of the reference dictionary $s \in \mathcal{D}_r = \{\mathcal{P}^+, \mathcal{N}^-, \mathcal{N}^{--}\}$. For this, Bhattacharya's coefficient is computed using Eq. (4.11).

$$\beta_i(\mathcal{D}_r, M_k^i) = \sum_{n=1}^{N} \sqrt{(\mathcal{D}_r)^n \times (M_k^i)^n} \tag{4.11}$$

Using Eq.(4.11), the Bhattacharya's distance between i^{th} fragment feature and the corresponding feature from each sample of the reference dictionary $\mathcal{D}_r$ is determined using Eq. (4.12).

$$B_i(\mathcal{D}_r, M_k^i) = \sqrt{1 - \beta_i(\mathcal{D}_r, M_k^i)} \tag{4.12}$$

After this, the likelihood for each feature is determined and the similarity matrix $\Gamma_i \in \mathbb{R}^{s\times 1}$ for each feature k in i^{th} candidate fragment can be computed using

Eq.(4.13).

$$\lambda_{i,k}(\mathcal{D}_r, M_k^i) = \frac{1}{\sigma_k\sqrt{2\pi}} e^{-\frac{B_{i,k}^2}{2\sigma_k^2}}, \quad k \in r, o \tag{4.13}$$

where, σ_k denotes cue standard deviation. The discussions related to multi-cue feature fusion model will be detailed in the next section.

4.2.1.1 Proposed Multi-cue Feature Fusion

The proposed fusion model performs a non-linear fusion of multi-cue features in a fuzzy inference model [107]. The fusion model not only captures the eminent relationship between the features but also resolve the conflict between the features. For this, fuzzy nearness and correlation coefficient [108] are induced in a fuzzy fusion model. Fuzzy nearness matrix $F_n \in \mathbb{R}^{s\times s}$ measures the similarity between the features for i^{th} fragment using Eq. (4.14).

$$F_{n,i}(M_{r,u}, M_{o,v}) = \frac{\sum_{p=1}^{n} min(\lambda_{r,u}^p, \lambda_{o,v}^p)}{\sum_{p=1}^{n} max(\lambda_{r,u}^p, \lambda_{o,v}^p)}, \qquad u, v = 1, 2...s \tag{4.14}$$

Next, the correlation coefficient qualitatively measures the degree of conflict between two sources of evidence. The correlation coefficient matrix $Cor \in \mathbb{R}^{s\times s}$ measures the difference of conflict between two features for i^{th} fragment and determined by Eq. (4.15).

$$Cor_i(M_{r,u}, M_{o,v}) = \begin{cases} \frac{\lambda_{r,u}^{max} + \lambda_{o,v}^{max}}{2}, & if \quad \lambda_{r,u}^{max} = \lambda_{o,v}^{max} \\ \frac{\lambda_{r,u}^{min} + \lambda_{o,v}^{min}}{2}, & if \quad \lambda_{r,u}^{max} \neq \lambda_{o,v}^{max} \end{cases} \tag{4.15}$$

The non-linear and complex relationship between the fuzzy nearness and correlation coefficient can be modeled using fuzzy theory. For this, the obtained fuzzy nearness F_n and correlation coefficient Cor are fuzzified. For fuzzification process, the sample set $S = \{vl, l, m, s, vs\}$ defined the very large, large, medium, small and very small for $x \in \{F_{n,i}$ or $Cor_i\}$. Membership functions are computed for each set using Eqs.(4.16-4.20).

$$\mu_{vl}(x) = 1/1 + e^{-p_{vl}(x-q_{vl})} \tag{4.16}$$

$$\mu_l(x) = e^{\frac{-(x-p_l)^2}{q_l^2}} \tag{4.17}$$

$$\mu_m(x) = e^{\frac{-(x-p_m)^2}{q_m^2}} \tag{4.18}$$

$$\mu_s(x) = e^{\frac{-(x-p_s)^2}{q_s^2}} \tag{4.19}$$

$$\mu_{vs}(x) = 1/1 + e^{-p_{vs}(x-q_{vs})} \tag{4.20}$$

Where, $p_{vl}, q_{vl}, p_l, q_l, p_m, q_m, p_s, q_s, p_{vs}$ and q_{vs} are linguistic parameters defined for each membership function in the sample set respectively. In order to map the relationship between the fuzzy nearness F_n and correlation coefficient Cor, the fuzzy inference rules are defined in the Table 4.1.

The crisp values for the similarity matrix $\dot{\mathcal{S}} \in \mathbb{R}^{s \times s}$ are obtained by the COG method. COG values between the F_n and Cor can be determined using Eq.

Table 4.1: Fuzzy inference rules

$\dot{\mathcal{S}}$		F_n				
		vl	l	m	vs	s
Cor	vl	vl	vl	l	l	l
	l	vl	vl	m	m	m
	m	l	m	m	s	vs
	s	l	m	s	vs	vs
	vs	l	m	vs	vs	vs

(4.21).

$$COG_{F_n,Cor} = \frac{\sum_{F_n}\sum_{Cor}\mu_{\dot{\mathcal{S}}}(x)x}{\sum_{F_n}\sum_{Cor}\mu_{\dot{\mathcal{S}}}(x)} \tag{4.21}$$

After, the defuzzification process, the values for the similarity matrix $\mathcal{S} \in \mathbb{R}^{s\times s}$ for i^{th} fragment can be obtained using the Eq. (4.22).

$$\mathcal{S}_i = \frac{\sum_{F_n}\sum_{Cor}\Lambda_{F_n,Cor} \times COG_{F_n,Cor}}{\sum_{F_n}\sum_{Cor}\Lambda_{F_n,Cor}(\dot{\mathcal{S}})} \tag{4.22}$$

Where, $\Lambda_{F_n,Cor} = min(\mu_{Cor}(\dot{\mathcal{S}}_{Cor}), \mu_{F_n}(\dot{\mathcal{S}}_{F_n}))$ is fuzzy control rule. This is followed by column-wise normalization of $\mathcal{S}_i$ to obtain the final unified robust feature $\mathcal{US}_i \in \mathbb{R}^{s\times 1}$ for i^{th} fragment $\mathcal{F}_i$ as Eq.(4.23).

$$\mathcal{US}(\mathcal{F}_i) = \frac{\mathcal{S}_i(\mathcal{F}_i)}{\sum_{j=1}^{s}|\mathcal{S}_i(\mathcal{F}_i)|} \tag{4.23}$$

This robust feature $\mathcal{US}(\mathcal{F}_i)$ is subjected to random forest classifier $\mathcal{R}_c$ for prediction. Random forest classifier $\mathcal{R}_c$ creates multiple regression decision tree in which each subset tree votes for the candidate fragment based on its branching structure. Breiman et al. [109] bagging algorithm has been exploited which generates a strong classifier from many homogeneous weak classifiers. Random forest assigns each candidate fragment a value [0,1] based on its analogy to positive fragment and negative fragment learned during the classifier training. In addition, At each frame, I scores are generated as $\Psi = \{\psi_1, \psi_2 ... \psi_I\}$. The high confidence score candidates are selected as $\tilde{\mathcal{F}}_i = \{\tilde{\mathcal{F}}_1, \tilde{\mathcal{F}}_2 ... \tilde{\mathcal{F}}_{\tilde{I}}\}$ and their score is averaged using Eq.(4.24).

$$\Upsilon_i = \frac{e^{\psi_i}}{\sum_{j=1}^{\tilde{I}} e^{\psi_j}} \tag{4.24}$$

Using this, the target is precisely localized and the final estimated state at time t is given by Eq.(4.25).

$$G_{t,(x,y)} = \sum_{i=1}^{\tilde{I}} \Upsilon_i \tilde{\mathcal{F}}_i(x, y) \tag{4.25}$$

where, $\tilde{\mathcal{F}}_i(x, y)$ is the centroid of the i^{th} fragment. Further, to prevent tracker's drift during long-term tracking the proposed tracker adopted a re-detection strategy. Also, to improve tracker's performance during abrupt environmental variations when all the candidate fragments are classified as negative, the whole tracker is reinitialized to re-detect the target. The details related to the experimental analysis of the proposed tracker is as follows.

4.3 Experimental Analysis and Discussion

In this section, the experimental results related to the performance of the proposed tracker has been analyzed and discussed. For this, the challenging video sequences which include the tracking challenges namely, illumination & scale variations, background clutters & deformation, full or partial occlusion, in-plane (IPR)

& out-of plane rotation (OPR), fast motion & motion blur, and low resolution & out-of view are selected from OTB-2015 [66] and VOT-2017 [87] datasets. Experimental tracking results are compared with seven other trackers namely, MEEM [110], STAPLE [96], DCFCA [111], SiamDW [112], RT-MPF [97], SiamRPN++ [113], and UGF [25]. We have implemented the proposed tracker in Python3 on a machine with core $i5$ 2.53 GHz processor and 4 GB RAM. Firstly, the tracker is initialized by generating the candidate fragments on the target. The number of initialized fragments will substantially be based on the dimension of the target. During initialization, a small size target requires less fragments in comparison to a large and high-resolution target. Also, the positive and two kinds of negative samples in the reference dictionary $\mathcal{D}_r$ are initialized as $\mathcal{P}^+, \mathcal{N}^-$ and $\mathcal{N}^{--} = 40$. The rigorous experimental analysis has been performed to demonstrate the adaptability of the proposed tracker against various tracking variations. For this, the qualitative and quantitative analysis of the proposed tracker has been detailed in the following subsection.

4.3.1 Qualitative Analysis

In this section, the qualitative evaluation of our tracker has been discussed. The detail of each challenge with the analyzed video sequence is follow in turn.

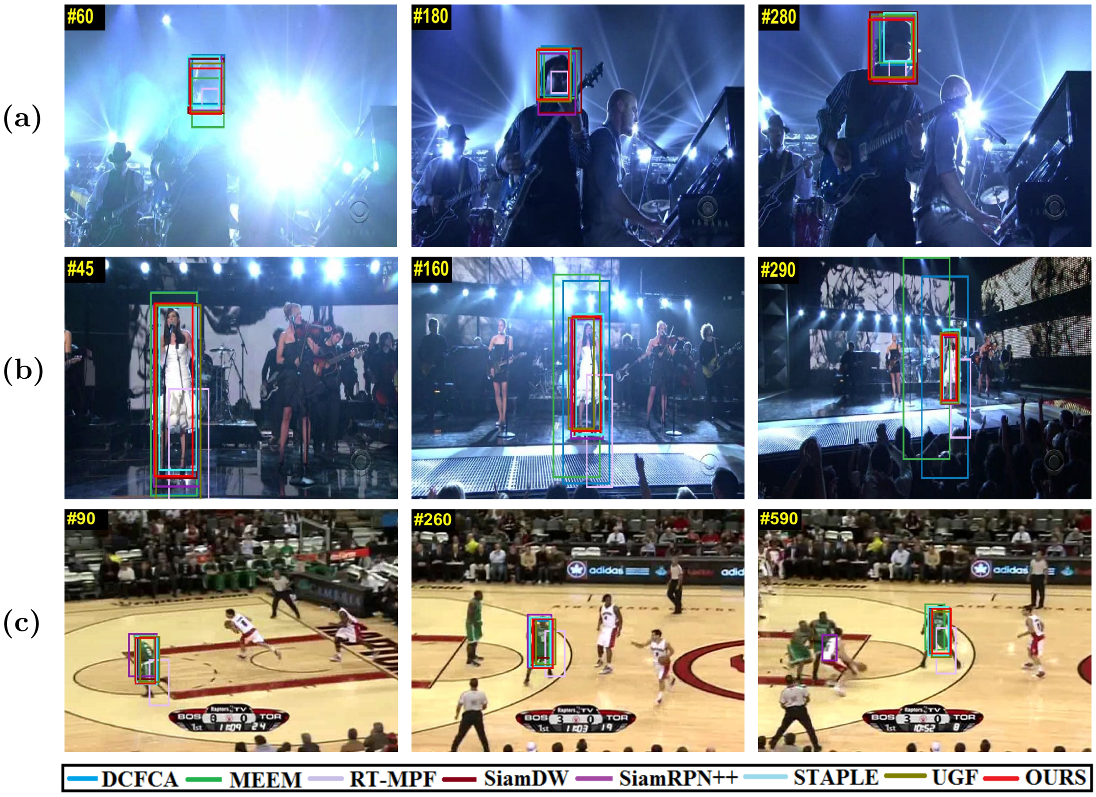

Fig. 4.2 *Sample Tracking frames on some critical video data with illumination & scale variation challenge: a)* Shaking *b)* Singer1 *c)* Basketball

4.3.1.1 Illumination & Scale variations

Under the challenge, video sequences viz. *Shaking*, *Singer1* and *Basketball* are evaluated. *Shaking* has sudden and abrupt changes in illumination throughout the whole sequence. Some representative tracking frames for the video sequences are shown in Fig. 4.2 (a). At frame #240, OURS, SiamDW and DCF-CA have performed significantly better against STAPLE, and RT-MPF. This is due to exploited multi-cue complementary features, viz. color and HOG in the proposed tracker's appearance model. Some frames for *Singer1* are depicted in Fig. 4.2 (b). Sequence has illumination variations with scale variations due to sudeen camera motion. OURS, STAPLE, SiamRPN++ and UGF track the target in the whole sequence. In *Basketball* sequence, SiamDW, DCF-CA and OURS have shown good results than the SiamRPN++, and MEEM. It is due to the consistent update of the reference dictionary with the candidate samples in the proposed tracker. In sum, OURS has shown good performance during the challenge due to complementary multi-cue features in the appearance model. When color cue fails to handle the illumination variation, HOG cue compensate during the challenge. Also, the inclusion of scale and rotation parameters in the proposed tracker's random walk model handle the variations and aids in the improved performance of the tracker.

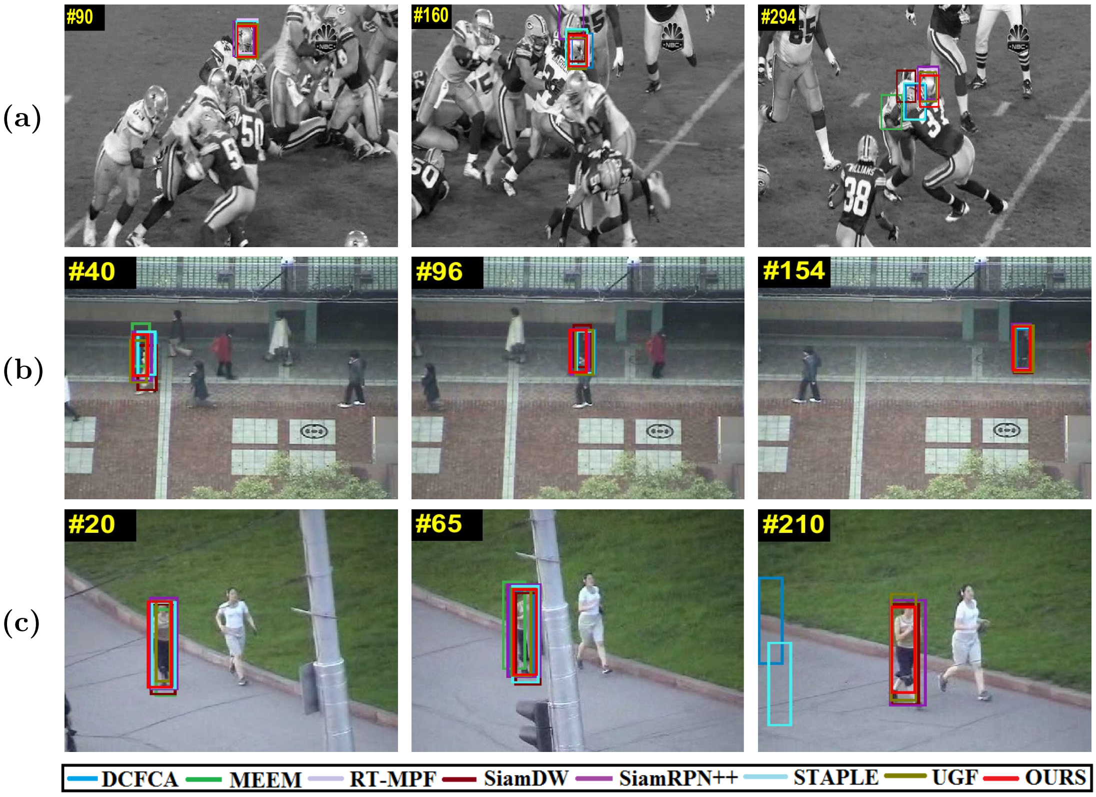

Fig. 4.3 *Sample tracking frames on some critical video data with full or partial occlusion challenge: a)* Football *b)* Subway *c)* Jogging1

4.3.1.2 Full occlusion (FOC) or Partial occlusion (POC)

Critical frames of video sequences having FOC or POC variations are illustrated in Fig. 5.4. In *Football* sequence, OURS, RT-MPF and SiamRPN++ are able to handle the challenge when the other players in the scene occlude the target. It is due to the multi-cue feature fusion, which ensures the high level relationship of the multi-cue is acquired well in the robust unified feature. At frame #40, in *Subway* sequence, when the object recover from multiple POC and FOC from the other persons, STAPLE, MEEM, and DCF-CA has shown failure in comparison to STAPLE, SiamRPN++ and OURS. In *Jogging1*, OURS, UGF and SiamDW are able to handle the challenge in comparison to other trackers and the same is illustrated in Fig. 5.4 (c). In sum, the proposed tracker has managed the challenge gracefully. It is due to the fuzzy based fusion of the proposed tracker which generates the robust unified feature. In addition, the random forest classifier classifies the affected fragments due to occlusion and prevents the inaccurate update of the proposed tracker.

4.3.1.3 Deformation & Background clutter

Deformation challenge occurs in the sequence due to the target's sudden pose variations or similar backgrounds. For this, video sequences viz. *Bolt2*, *Dancer2* and *Walking2* are considered and the representative frames are shown in Fig. 5.5. *Bolt2* has a target's deformation accompanied by a change in pose and background clutters. At frame #282, MEEM, SiamDW, and RT-MPF have drifted away

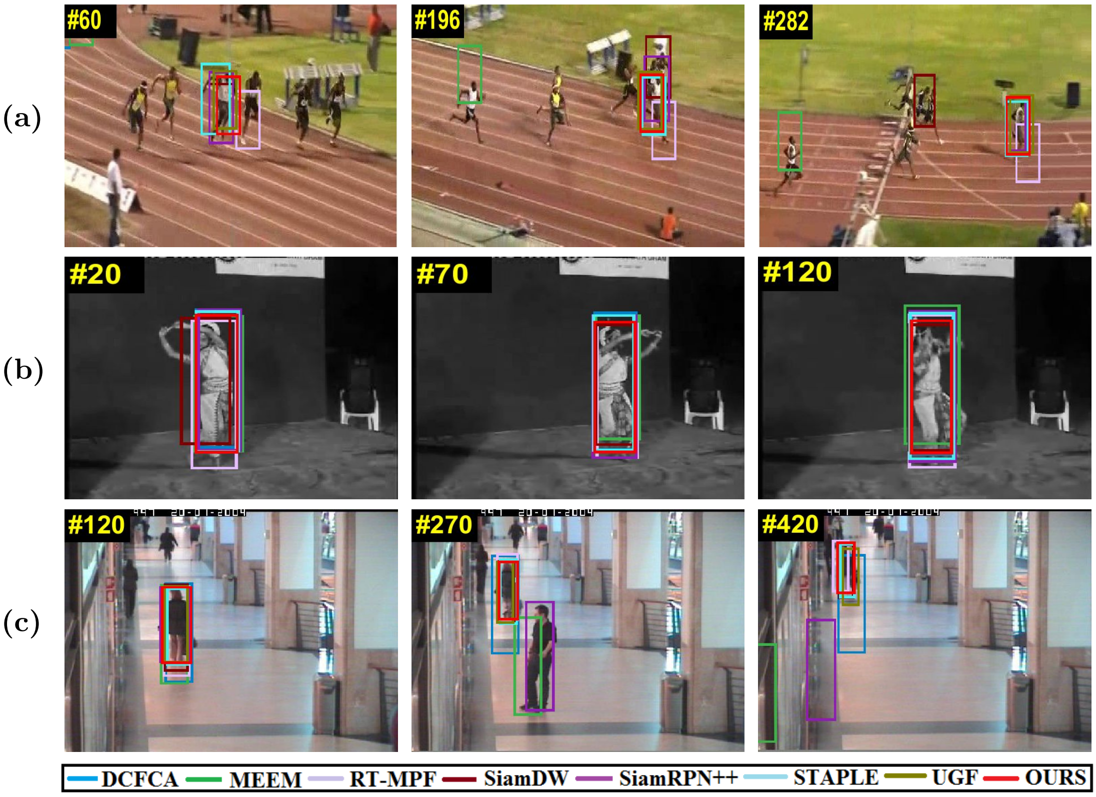

Fig. 4.4 *Sample tracking frames on some critical video data with deformation & background clutter challenge: a)* Bolt2 *b)* Dancer2 *c)* Walking2

from the target. However,OURS, STAPLE and UGF track the target successfully. Target has deformation due to its pose variations in *Dancer2* and is depicted in Fig. 5.5. Mostly trackers have shown better performance in the sequence but OURS tracks the target with minimal error. In *Walking2*, at frame #270, when another person is in the scene, the trackers viz. SiamRPN++, DCFCA, and MEEM keep its track and lost the target. OURS, STAPLE, UGF and RT-MPF keep track of the target. In sum, the tracker performance during the challenge primarily due to positive and negative samples in the reference dictionary. These sets improve the discriminating ability of the proposed tracker in the presence of the other objects. Also, the update of these samples in the subsequent frame enhances the adaptive ability of the tracker during the challenge.

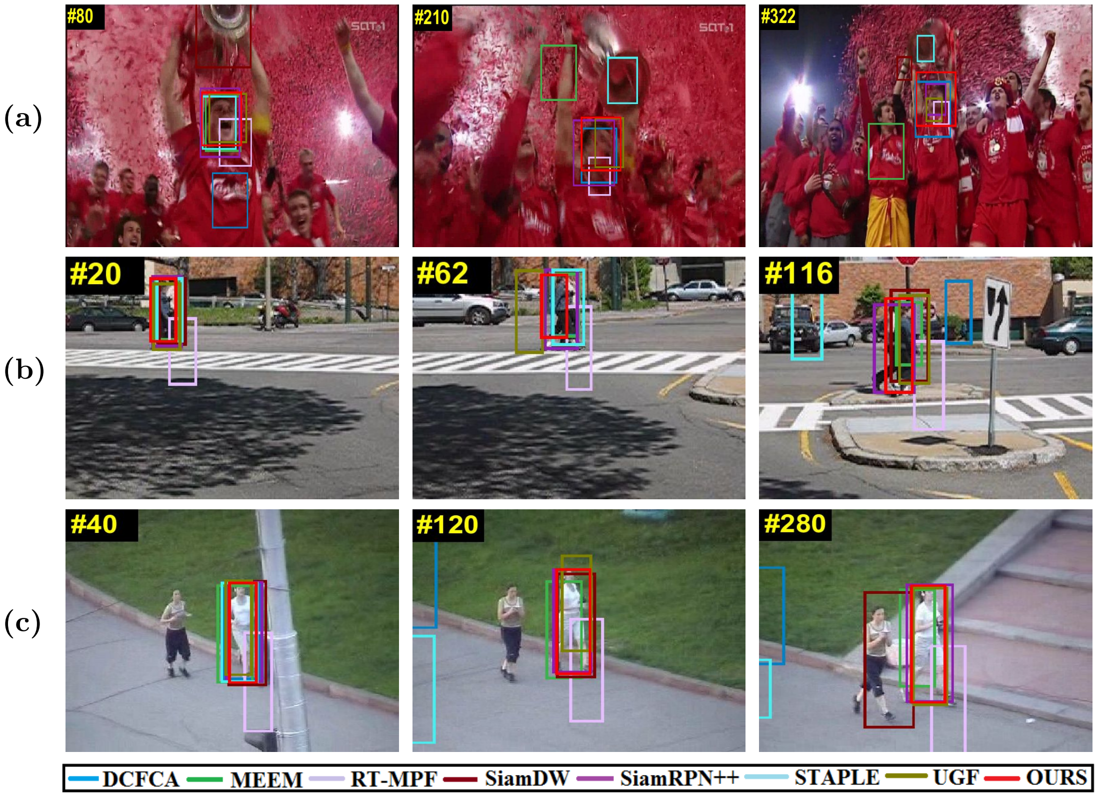

Fig. 4.5 *Sample tracking frames on some critical video data with IPR & OPR challenge: a)* Soccer1 *b)* Pedestrian1 *c)* Jogging2

4.3.1.4 In-plane and Out-of-plane rotation

Generally, in-plane (IPR) and out-of plane (OPR) rotation occur in the video sequence due to the frequent movement and rotation of the target in and out of the image plane. Fig. 4.5 (a) depicts the representative frames for *Soccer1* sequence. The sequence has IPR and OPR in the whole sequence due to target movement. The proposed tracker has handled the rotation with minimal error in comparison to the other trackers. This is due to the regular update of the samples of reference dictionary which addresses the eventual changes. *Pedestrian1* sequence has variations due to the camera's abrupt movement. In frame # 116, all trackers distract away from the target as shown in Fig. 4.5 (b). Only OURS, MEEM and

SiamDW successfully locate the target till end. In *Jogging2*, OURS with MEEM and SiamRPN++ has tracked the target successfully at frame #280 while the rest of the trackers failed to keep the track. In sum, OURS has performed considerably better in comparison to the other trackers. It is due to the rotational component of the random walk which handles the variations gracefully. The motion cue extracted during rough localization will cater the rotational changes. Also, the unified robust fused feature which exaggerates the complementary HOG feature over the RGB feature to accommodate the appearance variations under the challenge.

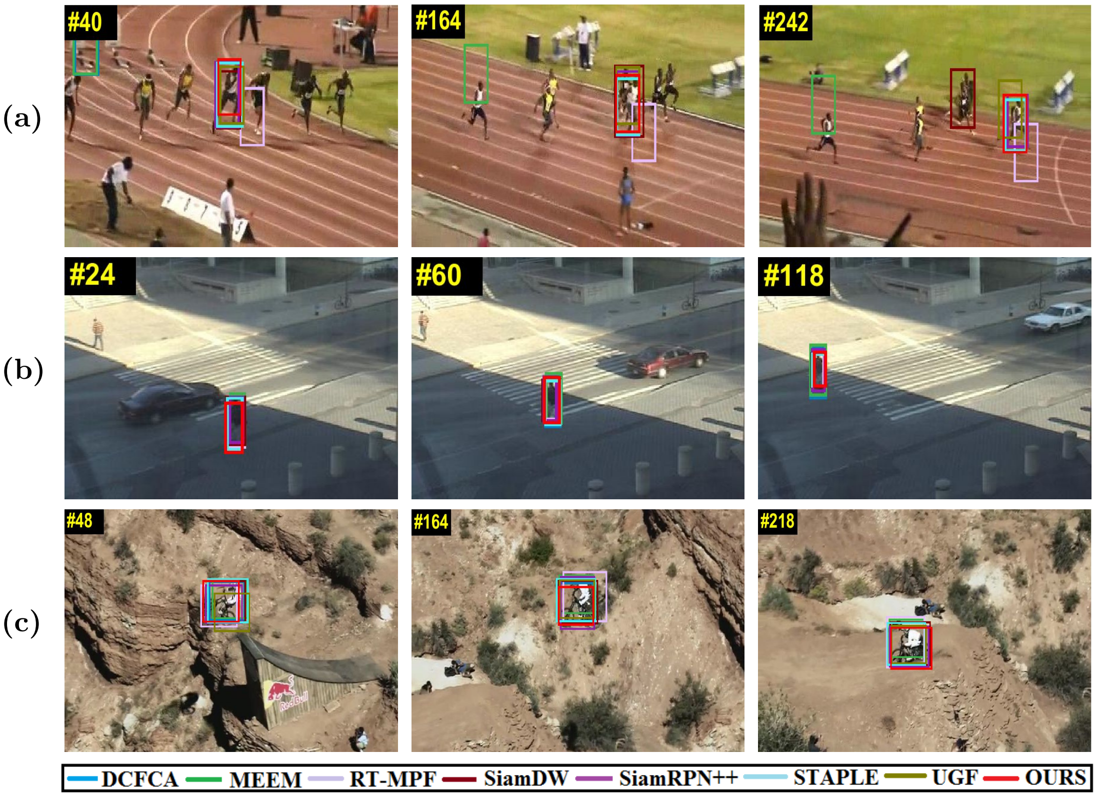

Fig. 4.6 *Sample tracking frames on some critical video data with fast motion & motion blur challenge: a)* Bolt2 *b)* Crossing *c)* MountainBike

4.3.1.5 Fast motion & Motion blur

Sample video frames under fast motion and motion blur challenge are presented in Fig. 5.6. In *Bolt2*, at frame #40 DCFCA and MEEM have lost the target while other trackers can locate the target. However, at frame #242 OURS, STAPLE and SiamRPN++ have shown better results than others. The performance of our tracker in the sequence is mainly attributed to the random forest classifier which classifies the affected fragments from the others. In *crossing* sequence, OURS and SiamDW have shown marginally less error in comparison to other state-of-the-arts. *MountainBike* sequence has fast motion challenge. At frame #164, OURS, SiamRPN++ and SiamDW track the target with a small error in comparison to RT-MPF, MEEM and UGF. In sum, the performance of our tracker during fast motion and motion blur is due to the proposed classifier mechanism. This classifier not only classifies the affected fragments but also prevents the false periodic update of the tracker with positive samples. Also, the rotational component in the random walk model ensures the high performance of the tracker under the challenge.

4.3.1.6 Low resolution & Out-of-view

In Fig. 4.7, representative frames under the low resolution and out-of-view challenge are illustrated. *Tiger* sequence, at frame #240 UGF, RT-MPF, MEEM and DGFCA fail to keep target's track but OURS and SiamDW keep track till the end. In *Walking2*, OURS has shown comparatively good performance with respect to other trackers. This is due to the proposed fusion of multi-cue which

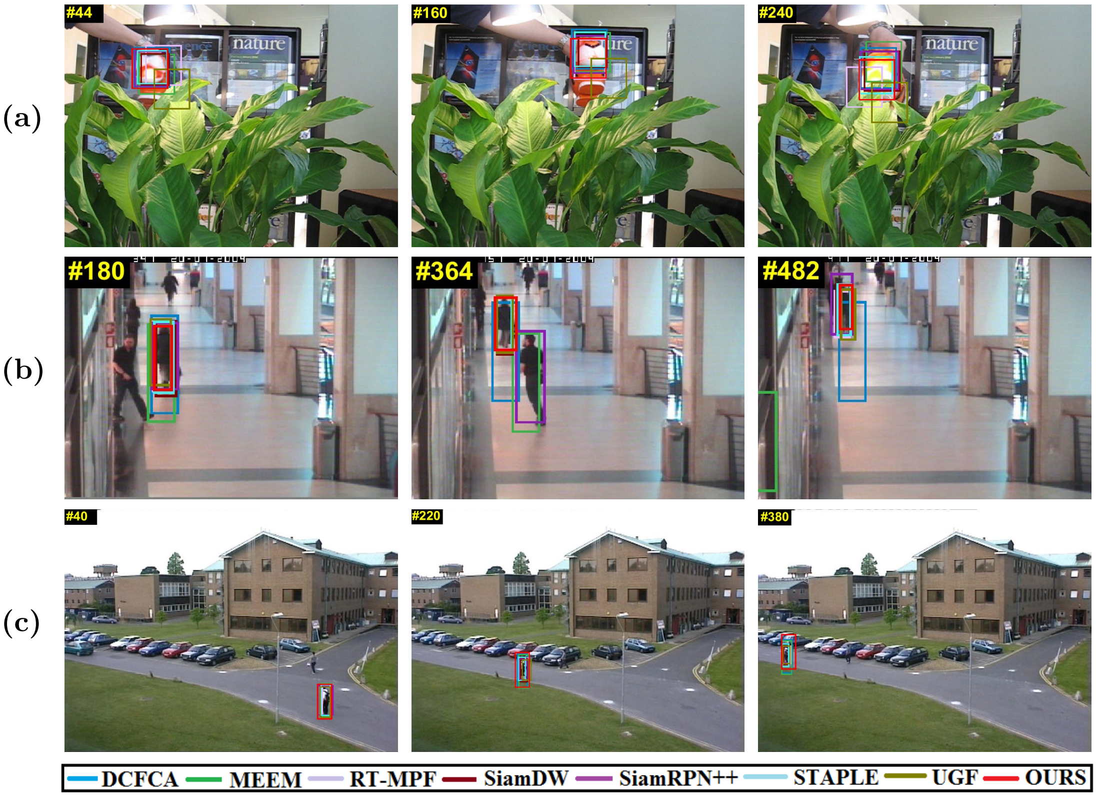

Fig. 4.7 *Sample tracking frames on some critical video data with low resolution & out-of view challenge: a)* Tiger *b)* Walking2 *c)* Walking

captures the eminent relationship between the cues well. OURS, SiamRPN++, and SiamDW have shown better results in comparison to others in the *Walking* sequence. In sum, the proposed tracker is able to address the challenge in the video sequences gracefully. It is due to the proposed fuzzy based fusion of multi-cue which captures the eminent relationship between cues and diminishes the low level relationship. In addition, the visual cues viz. color and HOG with motion descriptor in the proposed tracker's appearance model improve its performance under the low resolution and out-of-view challenge.

TABLE 4.2: Comparative average centre location error tracking results (in pixels). First, second and third results are highlighted respectively.

Challenge	DCFCA	MEEM	STAPLE	SiamDW	RT-MPF	SiamRPN++	UGF	OURS
Illumination & Scale variations	8.94	18.27	10.45	10.94	49.24	24.36	7.55	6.78
Full or partial Occlusion	35.52	14.05	35.60	5.72	7.53	5.38	7.33	4.69
Fast motion & Motion blur	102.92	85.79	5.82	15.46	13.83	4.30	7.78	4.95
In-plane & Out-of-plane rotations	63.44	37.93	82.40	25.27	47.44	10.06	14.18	4.93
Deformations & Background clutter	35.52	12.65	35.60	5.72	37.71	5.38	7.13	6.29
Low resolution & Out-of view	12.76	33.15	5.31	7.99	3.30	19.95	21.11	7.05
Average	43.18	33.64	29.20	11.85	26.51	11.57	10.85	5.78

TABLE 4.3: Comparative average F-Measure tracking results. First, second and third results are highlighted respectively.

Challenge	DCFCA	MEEM	STAPLE	SiamDW	RT-MPF	SiamRPN++	UGF	OURS
Illumination & Scale variations	0.718	0.683	0.821	0.813	0.289	0.765	0.834	0.851
Full or partial Occlusion	0.571	0.708	0.575	0.832	0.748	0.799	0.752	0.833
Fast motion & Motion blur	0.555	0.527	0.835	0.736	0.382	0.852	0.805	0.839
In-Plane & Out-of-plane rotations	0.386	0.554	0.339	0.609	0.256	0.765	0.613	0.768
Deformations & Background clutter	0.571	0.739	0.575	0.832	0.281	0.789	0.765	0.791
Low resolution & Out-of view	0.689	0.563	0.842	0.791	0.524	0.693	0.619	0.754
Average	0.582	0.629	0.664	0.769	0.413	0.777	0.731	0.806

4.3.2 Quantitative Analysis

For the quantitative evaluation of the proposed tracker, robust performance metrics are considered. Two performance metrics namely, Center location error (CLE) and F-Measure [65] are tabulated. Two plots, namely precision plot and success plot [66] are plotted to prove the effectiveness of the proposed tracker. In addition, Area under the curve (AUC) and mean precision are also computed for comparing the proposed tracker's performance with other state-of-the-arts. Center location error (CLE) can be computed as the distance between the tracked bounding box and the ground truth. F-Measure is described as $f'_m = (2 \times rc' \times pr')/(1 + rc' + pr')$. Here, rc' is recall defined as the overlap ratio between BB_t and BB_g with respect to BB_g and pr' is precision calculated as the overlap ratio between BB_t and BB_g with respect to BB_t. BB_t and BB_g are target's bounding box and ground truth bounding box, respectively. Precision plot illustrates mean precision at multiple location error thresholds. Success plot depicts the percentage of correctly tracked frames at different overlap thresholds.

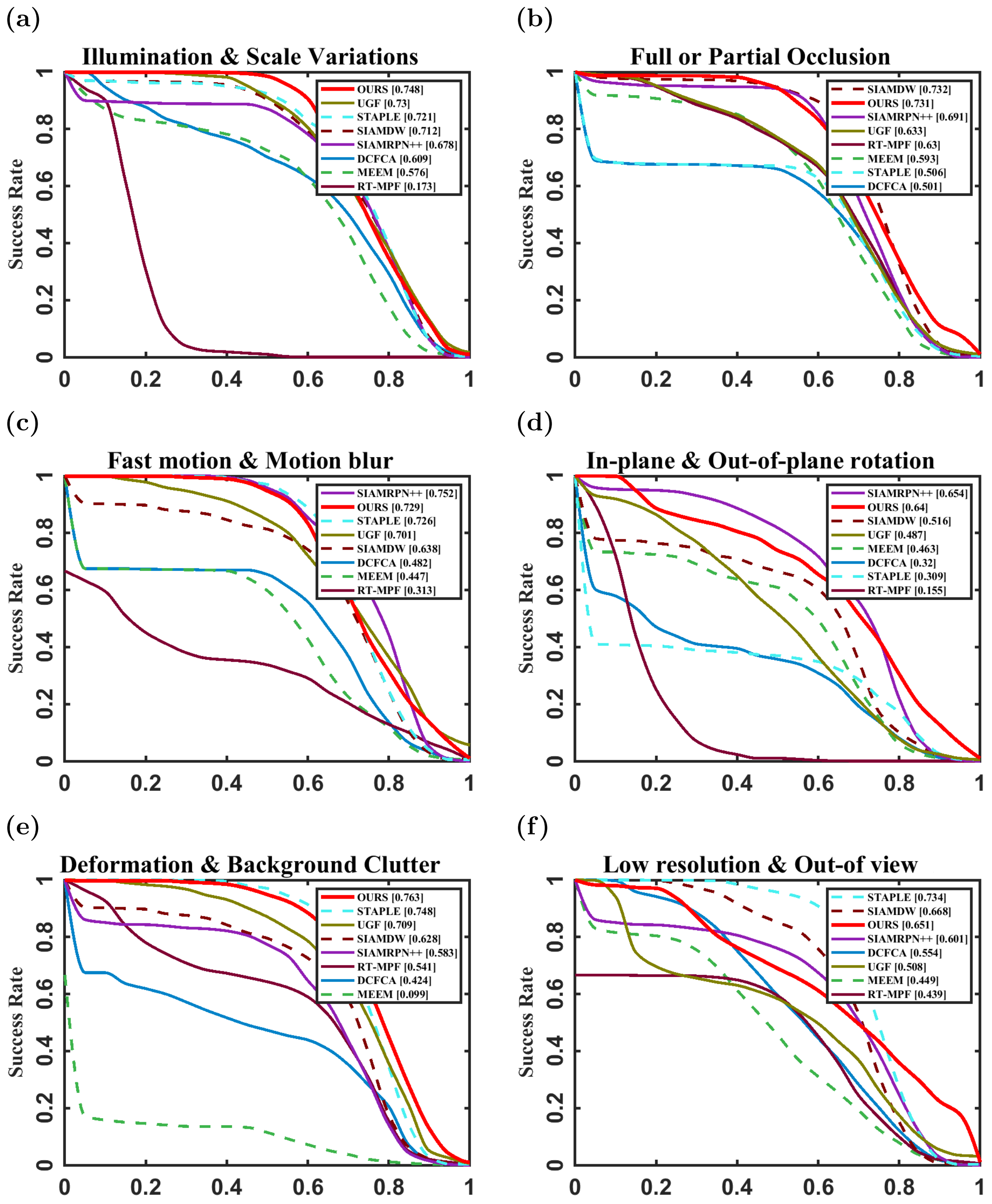

Fig. 4.8 *Success plot under tracking variations (a) Illumination & Scale variations (b) Full or partial Occlusion (c) Fast motion & Motion blur (d) In-plane & Out-of-plane rotation (e) Deformation & Background clutter (f) Low resolution & Out-of-view . Legend includes AUC in the bracket.*

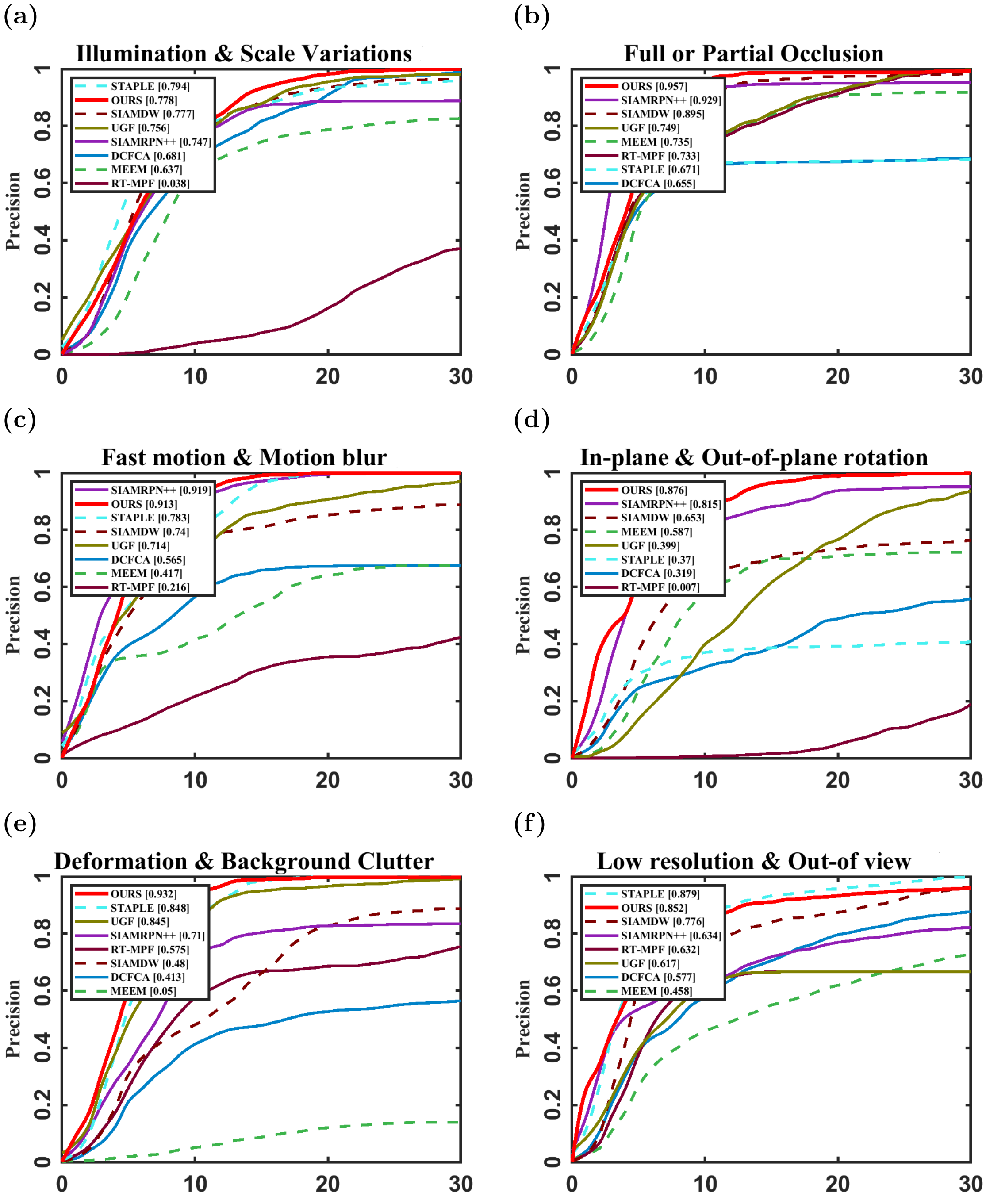

Fig. 4.9 *Precision plot under tracking variations (a) Illumination & Scale variations (b) Full or partial Occlusion (c) Fast motion & Motion blur (d) In-plane & Out-of-plane rotation (e) Deformation & Background clutter (f) Low resolution & Out-of-view. Legend includes mean precision overlap score in the bracket.*

Table 4.2 and Table 4.3 tabulate the average center location error (CLE) and average F-Measure of the trackers, respectively. The proposed tracker has attained average mean CLE of 5.78 (in pixels) and average mean F-measure of 0.806 when evaluated on challenging benchmark video sequences. Also, the success plots and precision plots, as per the attributed challenge are illustrated in Figs. 4.8 and 4.9, respectively. High mean precision score and AUC demonstrate the efficiency of the proposed tracker during various environmental variations. During illumination and scale variations, our tracker's outstanding performance proves the effectiveness of the exploited complementary multi-cue namely, RGB and HOG in the target's appearance model. Our tracker has achieved the highest AUC of 0.748, as depicted by Fig. 4.8 (a). In sequences with full occlusion and partial occlusion, the tracking results reveal the strength of the unified feature generated by the proposed fusion model against tracking variations. The proposed tracker has attained the highest mean precision of 0.957 during full and partial occlusion and the same is illustrated in Fig. 4.9 (b). During fast motion & motion blur, our tracker has attained second highest 0.729 AUC and 0.913 mean precision as depicted in Figs. 4.8 and 4.9 (c), respectively. The motion cue descriptor during rough localization alongwith sample update of effective fragment over the affected fragment ensures high mean precision of 0.876 during in-plane and out-of plane rotation. Highest AUC 0.763 and mean overlap precision 0.932 during deformation and background clutter confirm the robustness of the fused feature alongwith the adaptive appearance model which consistently updates the target's appearance. The tracker has attained 0.651 AUC and 0.852 mean precision during low resolution & out-of-view challenge depicted by Figs. 4.8 and 4.9 (f), respectively. In sum, the proposed tracker has shown efficient performance addressing various environmental variations.

The proposed tracker outperforms other state-of-the-art trackers. Discriminative tracker, UGF is not robust to target's rotation and deformations. Particle filter based tracker, RT-MPF is inefficient in handling the low resolution and out-of view challenge. STAPLE is unable to cope with target's deformation due to dynamic variations. Deep learning based trackers viz. SiamRPN++ and SiamDW have shown comparable performance but their realization requires specialized hardware for their execution. DCFCA is comparatively slow and not appropriate for real-time scenarios. MEEM exploits quantized color features in the target's appearance model and fail to cater background clutters when accompanied with out-of plane rotations. To summarize, the proposed tracker shows favorable performance against existing state-of-the-art in both object overlap and mean precision with leading average F-Measure and minimum average CLE on challenging benchmark sequences.

4.4 Significant Findings

The significant highlights of the present work are as follows.

- Proposed a robust object tracking architecture under deterministic framework based on adaptive feature fusion of complementary features and a discriminative classifier which was able to cope with various visual tracking challenges.

- Complementary cues namely, color and HOG build robust appearance model invariant to illumination variations and scale variations.
- The proposed tracker has exhibited real-time performance with 24fps.
- The proposed fusion model generated robust unified feature robust to background clutters and occlusion.
- Random forest classifier created discriminative decision boundary between the target fragments and background fragments. Hence, ensured tracker's quick adaptation under various visual tracking challenges.

In addition, experimental results along with other findings were published in [114].

Chapter 5

Multi-stage Framework for Multi-cue Object Tracking

Chapter 5

Multi-stage Framework for Multi-cue Object Tracking

The aim of this research is to propose an adaptive multi-cue object tracking model under multi-stage framework. For this, target is localized from coarse to fine using a discriminative approach [25]. The extracted multi-cue features are fused adaptively using feature level fusion.

5.1 Introduction

In multi-cue object tracking, feature level fusion has been extensively explored. In this direction, Fu et al. [115] proposed structural similarity in order to determine the sample importance with a bag probability function. Classifier scores were used to obtain the final fused vector. Feng et al. [116] constructed weighted map for each frame by incorporating target saliency. Weight map was updated dynamically using the level-set algorithm. Xiao et al. [9] fused contextual information with depth cue in a two layered target model. Adaptive multi-feature fusion was proposed to stabilize tracker's performance during camera motion and other tracking variations. Authors proposed multi-task kernel based tracking framework [58]. Similarity between features was determined to discriminate the reliable features from unreliable ones. In [59], authors proposed multi-feature fusion using regional covariance matrices. Covariance matrices provided accurate and reliable tracking results. However, direct fusion using covariance matrices was computationally complex. However, authors [64] proposed complementary measurement matrix for fusion of multi-cue features. The method was efficient in handling full or partial occlusion. However, scale variations and out-of-view challenges were neglected. Wang et al. [62] combined two random forest based discriminative model to provide robust discriminative appearance model. The method had shown efficient tracking performance under various tracking challenges. In [51], authors extracted optical flow and gabor features based on contours for the object's appearance model. On the other hand, Walia et al. [25] considered optical flow for coarse estimation of the target. After this, intensity, texture and edge features were exploited in the appearance model for the fine estimation. These features were fused using cross-diffusion mechanism to generate a unified robust feature. In sum, many robust multi-cue tracking solutions considering feature level fusion has been proposed. However, developing a robust non-linear unified feature for an adaptive appearance model catering the environmental challenges can be further explored.

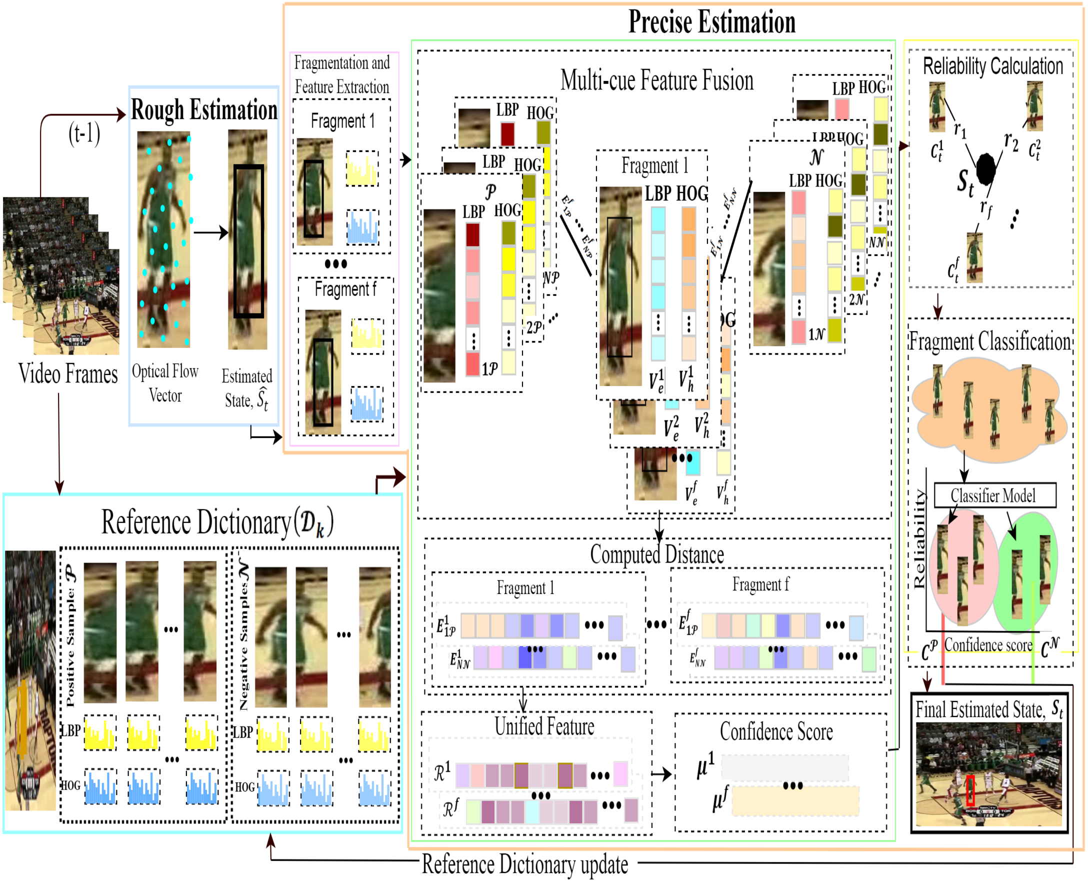

Fig. 5.1 *Overview of the proposed tracker. Tracker is initialized with positive and negative fragments in the reference dictionary. Stage I, target is roughly localized through optical flow. Then, fragments are generated around the localized target and LBP-HOG features are extracted. Unified feature is generated and transductive reliability is calculated for each localized fragment. Tracker is made adaptive by classifier score and transductive reliability.*

5.2 Proposed Tracker Architecture

The proposed multi-stage tracker considers both motion cue and visual feature descriptor for target localization in video sequences. At stage I, the target is roughly localized by exploiting the optical flow. This provided a much better feature sampling for the next stage of precise localization. During precise localization, candidate fragments are generated around the approximated centroid of the previous stage and complementary multi-cue features namely, LBP (V_e) and HOG (V_h) are extracted. For each candidate fragment, Euclidean distance (E^f) is calculated between each feature descriptor and the corresponding feature of the positive sample ($\mathcal{P}$) and negative samples ($\mathcal{N}$) from the dictionary. The robust unified feature vector ($\mathcal{R}^f$) is generated by averaging the obtained distances which ensured that the high level features are captured well and low-level features are suppressed. Further, the unified feature is averaged to determine the confidence score (μ_f) for each fragment. Fig. 5.1 represents the architecture of the proposed tracker. Initially, a set of positive samples ($\mathcal{P}$) and negative samples ($\mathcal{N}$) are stored in the reference dictionary ($\mathcal{D}_k$). These samples are updated with the reliable samples in the subsequent frames. Next, the fragments obtain by the proposed multi-cue feature fusion are passed through a K-Means classifier to classify them into the positive samples ($C^{\mathcal{P}}$) and the negative samples ($C^{\mathcal{N}}$). At each time step t, transductive reliability ($r(C^f)$) is calculated to assure the consistent updation of the tracker with reliable fragments catering the dynamic environment. The final state (S_t) of the target is obtained by the mean of the high confidence and high reliability fragments. Confidence score and tranductive reliability ensure that the tracker is updated with high confidence score reliable samples only. Also, the reference dictionary ($\mathcal{D}_k$) is updated with the selective replacement of the positive samples and the negative samples. The next section details the first stage of the proposed tracker.

5.2.1 Stage I: Rough Localization

During rough localization, the target is localized through optical flow. Optical flow considers the rate of movement of the pixel in the respective directions between the consecutive frames. In the proposed method, Horn-Schunck method (HS) is used to determine the optical flow [105]. Optical flow determines the change in pixel intensity $\mathcal{I}(x, y, t)$ at point (x, y) at time t assuming the change in brightness between the consecutive frames was constant and the same has been depicted using Eq.(5.1)

$$\mathcal{I}(x, y, t) = \mathcal{I}_x v_x + \mathcal{I}_y v_y + \mathcal{I}_t \tag{5.1}$$

where, $v_x = \frac{dx}{dt}$ and $v_y = \frac{dy}{dt}$ represents the change in pixel displacement in x and y direction w.r.t. change in dt respectively. Smoothness constraint is determined by minimizing the error given by Eq. (5.2).

$$\epsilon^2 = \int\int (\mathcal{I}(x, y, t)^2 + \alpha^2((\frac{\partial v_x}{\partial x})^2 + (\frac{\partial v_x}{\partial y})^2 + (\frac{\partial v_y}{\partial x})^2 + (\frac{\partial v_y}{\partial y})^2) dxdy \tag{5.2}$$

Where, α^2 scale the second term which is smoothness constraint. If we have selected N flow vectors around the precise localized centroid $S_{t-1,(x,y)}$ in the previous frame, then the centroid $\hat{S}_t$ in the current frame is roughly localized and depicted using the Eq. (5.3).

$$\hat{S}_{t,(x,y)} = \left(S_{t-1,x} + \frac{1}{N}\left(\sum_{i=1}^{N} v_x^i \right), S_{t-1,y} + \frac{1}{N}\left(\sum_{i=1}^{N} v_y^i \right) \right) \tag{5.3}$$

This roughly localized centroid is used for generating the candidate fragments for precise localization of the target. The next section will detail about the precise localization of the target.

5.2.2 Stage II: Precise Localization

During this stage, the target localization is accomplished through multi-cue feature fusion of each candidate fragments. For this, candidate fragments are generated by sampling the target around the centroid ($\hat{S}_t$) determined in stage I using random walk model. Random walk caters for object's scaling and rotational variations. For each fragment, state ($\hat{S}_t$) is propagated through the random walk model using Eq.(5.4).

$$\hat{S}_{t+1} = \hat{S}_t + \eta_t \tag{5.4}$$

Where, $\hat{S}_t = (c, \tau)^m$ in which c represents the scaling factor and τ is angular displacement for each fragment. m is a covariance matrix used to describe the uncertainty in state vector as $m = diag(\sigma_c, \sigma_\tau)$. η_t is zero mean Gaussian noise through a random walk model. Each localized fragment is scaled and rotated using the random walk model and multi-cue features are then extracted. In the proposed tracker, we have considered complementary cues viz. Texture and HOG and their extraction details are as follows.

Texture cue extracts the low level image features with spatial arrangement of pixels in the image plane. Texture cue is robust to scaling and rotational variations. For each fragment, texture cue is extracted using scale-invariant LBP [83]. Further, the normalized histograms are obtained by the uniformity LBP measure for each pixel. Uniformity LBP is determined considering central pixel (p_c, q_c) and neighborhood pixel (p_o, q_o) in radius γ using Eq. (5.5).

$$\begin{aligned} U(LBP_{\gamma,N_b}(p_c,q_c)) &= |\beta(\mathcal{I}_{j-1}(p_{j-1},q_{j-1}) - \mathcal{I}_c(p_c,q_c)) - \beta(\mathcal{I}_o(p_o,q_o) - \mathcal{I}_c(p_c,q_c))| \\ &+ \sum_{j=0}^{N_b-1} |\beta(\mathcal{I}_j(p_j,q_j) - \mathcal{I}_c(p_c,q_c)) - \beta(\mathcal{I}_{j-1}(p_{j-1},q_{j-1}) - \mathcal{I}_c(p_c,q_c))| \end{aligned} \tag{5.5}$$

$$LBP_{\gamma,N_b}(p_c,q_c) = \begin{cases} \sum_{j=0}^{N_b-1} |\beta((\mathcal{I}_j(p_j,q_j) - \mathcal{I}_c(p_c,q_c))|, & U(LBP_{\gamma,N_b}(p_c,q_c)) \leq 2 \\ j+1, & otherwise \end{cases} \tag{5.6}$$

Where, $\mathcal{I}_c$ is gray value of central pixel (p_c, q_c) in radius γ, $\mathcal{I}_j$ is gray value of j^{th} pixel (p_j, q_j) and $\mathcal{I}_o$ is gray value of N_b equally spaced neighboring pixels (p_o, q_o). These values are used to determine the bit wise transition in the uniform image pattern for calculating texture uniformity LBP. For each candidate fragment, these histograms are concatenated to obtain a single feature vector as V_e^f.

HOG feature [106] is used to represent the edge information of the target. HOG captures the edge directions w.r.t. distribution of horizontal and vertical gradient intensities. The distribution of gradient intensities are determined by filtering the image with the kernels given by Eq.(5.7).

$$[-1, 0, 1] \qquad \& \qquad [-1, 0, 1]^T \tag{5.7}$$

If $\mathcal{G}_x(p,q)$ and $\mathcal{G}_y(p,q)$ represent the gradient values in the horizontal and vertical directions respectively. then, the magnitude $(\hat{M})$ and orientation(θ) for each pixel is calculated using Eq.(5.8) and Eq. (5.9).

$$\hat{M}_{(p,q)} = \sqrt{(\mathcal{G}_x(p,q))^2 + (\mathcal{G}_y(p,q))^2} \tag{5.8}$$

$$\theta_{(p,q)} = arctan\left(\frac{\mathcal{G}_y(p,q)}{\mathcal{G}_x(p,q)}\right) \tag{5.9}$$

Further, ROI(Region of Interest) of the image is divided into 9 rectangular cells and for each pixel in a cell a weighted vote for each edge orientation is calculated. Votes are bilinearly interpolated between the magnitude and the orientation for each bin. Here, cell histogram for each bin is calculated and bin (γ) for each cell is determined using Eq.(5.10).

$$G_h(\gamma) = \sum_{l=1}^{N_s}(\hat{M}_{l,(p,q)}\delta(\theta^{`}_{l,(p,q)} - \gamma)) \tag{5.10}$$

Where, $\delta(.)$ is Kronecker delta function, $\theta^{`}_{(p,q)}$ is quantized orientation and N_s is the number of pixels in each cell. To address the illumination variations and contrast, gradient values are normalized using L2-norm and final concatenated feature vector is obtained by Eq.(5.11).

$$V_h^f = V_h(\gamma)/\sqrt{\left(\sum_{l=1}^{c\times c\times N_b} V_h(l)^2\right) + \aleph^2} \tag{5.11}$$

Here, c is the number of cells and N_b is the number of bins per cell and $\aleph$ is a constant. For each candidate fragment HOG feature vector is determined as V_h^f.

5.2.2.1 Multi-cue Feature Fusion

For each candidate fragment f, the individual feature vector of LBP (V_e^f) and HOG (V_h^f) are unified to form a robust feature vector ($\mathcal{R}^f$). Initially, a set of positive samples ($\mathcal{P}$) and negative samples ($\mathcal{N}$) are stored in the reference dictionary ($\mathcal{D}_k$). In each frame, for each fragment the distance is calculated between the feature descriptor and the corresponding feature from the positive samples and the negative samples of the reference dictionary. Fig. 5.2 illustrates the calculation of distance between the feature vector and the corresponding feature vector from the positive sample and the negative sample of the reference dictionary. Further, the Euclidean distance between the i^{th} LBP feature point (a_i) and the corresponding

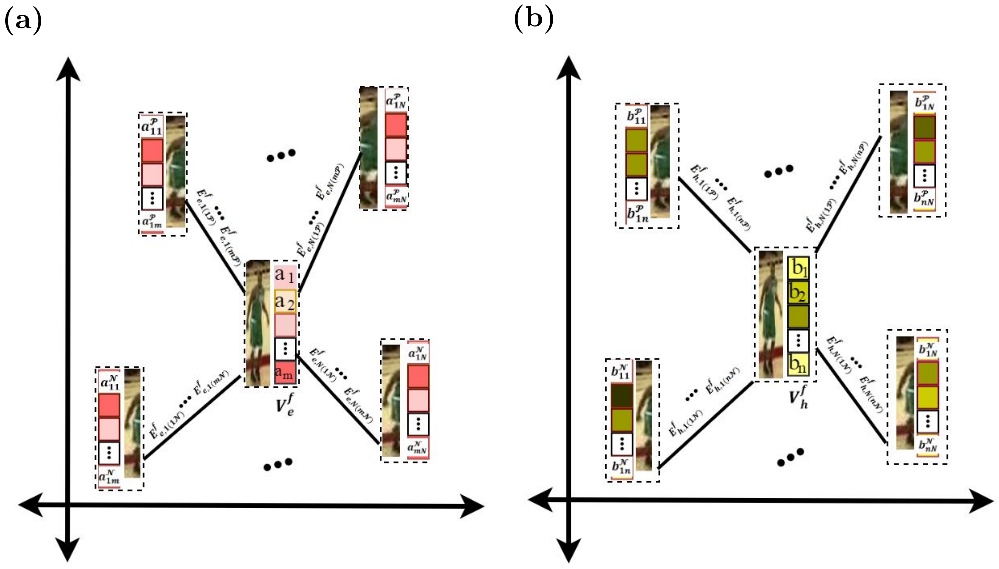

Fig. 5.2 *Description of distance calculation for feature descriptor (a) LBP (b) HOG*

feature point ($a_{i\mathcal{P}}$) of the positive sample and ($a_{i\mathcal{N}}$) the negative sample, respectively from the reference dictionary is depicted in Fig. 5.2 (a) and is calculated using Eqs.(5.12), (5.13).

$$E^f_{e,N(i\mathcal{P})} = (a_i - a_{i\mathcal{P}}), \quad for \quad 1 \leq i \leq m; 1 \leq \mathcal{P} \leq N \tag{5.12}$$

$$E^f_{e,N(i\mathcal{N})} = (a_i - a_{i\mathcal{N}}), \quad for \quad 1 \leq i \leq m; 1 \leq \mathcal{N} \leq N \tag{5.13}$$

Here, m is the length of LBP feature vector and N represents the number of positive samples and the negative samples in the reference dictionary. Similarly, distance between j^{th} HOG feature point (b_j) and the corresponding feature point ($b_{j\mathcal{P}}$) of the positive sample and ($b_{j\mathcal{N}}$) the negative sample, respectively from the reference dictionary is depicted in Fig. 5.2 (b) and is calculated using Eqs.(5.14), (5.15).

$$E^f_{h,N(j\mathcal{P})} = (b_j - b_{j\mathcal{P}}), \quad for \quad 1 \leq j \leq n; 1 \leq \mathcal{P} \leq N \tag{5.14}$$

$$E^f_{h,N(j\mathcal{N})} = (b_j - b_{j\mathcal{N}}), \quad for \quad 1 \leq j \leq n; 1 \leq \mathcal{N} \leq N \tag{5.15}$$

Here, n is the length of HOG feature vector and N represents the number of positive samples and the negative samples in the reference dictionary. Finally, the unified feature vector for f^{th} fragment between the features V_e and V_h is determined using Eq.(5.16).

$$\mathcal{R}^f = \sqrt{(E^f_{e,k})^2 + (E^f_{h,k})^2} \qquad k \in \mathcal{P}, \mathcal{N} \tag{5.16}$$

The obtained unified feature vector ($\mathcal{R}^f$) will preserve the high-level relationship and suppress the low-level relationship among the feature vectors. This feature vector is averaged to obtain $\bar{\mathcal{R}^f}$. For each fragment, confidence score is calculated using Eq.(5.17).

$$\mu^f = \frac{1}{2\pi\sigma_f} e^{-\frac{(\bar{\mathcal{R}^f})^2}{2\sigma_f^2}} \tag{5.17}$$

where, σ_f is the standard deviation of Gaussian noise in calculation process of the unified feature. Further, the obtained confidence scores of each fragments are subjected to K-Means classifier. The classifier provides the value [0,1] to the candidate fragments as per their resemblance into the positive samples and the negative samples. For f fragments, f scores are generated as $(\mu^1.\mu^2...\mu^f)$, K-Means clustering is used to partition the fragments into K classes as $C_1, C_2...C_K$. If f^{th} fragment belongs to K^{th} cluster then $f \in C_K$ and $C_K = [C_{\mathcal{P}}, C_{\mathcal{N}}]$. If $w(C_K)$ is the measure of within-cluster variations then the objective function which is to be minimized is given by Eq.(5.18).

$$argmin \sum_{i=1}^{K} \sum_{i \in C_K} ||w(C_K)|| \tag{5.18}$$

Here, the within-clusters variations $w(C_K)$ is determined using sum of squares method. For this, pairwise squared deviations of p points within the clusters is defined using Eq.(5.19).

$$w(C_K) = \frac{1}{|C_K|} \sum_{r=1}^{p} \sum_{a,b \in C_K} ||a_r - b_r||^2 \tag{5.19}$$

Where, $|C_K|$ represents the number of fragments in K^{th} cluster. Using Eqs.(5.18) and (5.19), the K-means clustering is defined using Eq.(5.20), Eq.(5.21).

$$argmin \sum_{i=1}^{K} \frac{1}{|C_K|} \sum_{r=1}^{p} \sum_{a,b \in C_K} ||a_r - b_r||^2 \tag{5.20}$$

$$argmin \sum_{a_r \in C_K} \sum_{r=1}^{p} ||a_r - \phi_f||^2 = \sum_{a_r \neq b_r \in C_K} \sum_{r=1}^{p} (a_r - \phi_f)(\phi_f - b_r) \tag{5.21}$$

Where, ϕ_f is mean of p points in cluster C_K. Clusters centroid is determined as the mean of all the fragments score assigned to each cluster. K-Means algorithm determines the local cluster and hence, executes iteratively to determine the nearest centroid. Also, the transductive reliability is calculated for each candidate fragment. The final state is estimated by taking the mean of the fragments with high confidence and high reliability using Eq.(5.22).

$$S_{t,(x,y)} = \sum_{n=1}^{\hat{N}} (C_n(x, y)) \tag{5.22}$$

Where, $C_{\hat{N}}(x, y)$ is the centroid and $\hat{N}$ is the number of candidate sample with high confidence score and high reliability, respectively. Reliability of each candidate fragment is calculated in order to ensure the tracker is adaptive with change in the environment. Reliability ($r(C^f)$) of the candidate fragment (C^f) can be calculated using Eq.(5.23).

$$r(C^f) = \frac{u(tanh(-(C^f - S_t + v)))}{2} + w \tag{5.23}$$

Where, u, v and w are constant to outline the shape of $tanh$ function. Further, to improve the performance during long term tracking a re-detection strategy has been employed in the proposed tracker. In case, the tracker drifts away from the target due to the sudden change in the environmental condition, the tracker is re-initialized to re-detect the target. The experimental validation of the proposed method will follow in turn.

5.3 Experimental Validation

For rigorous performance analysis, the proposed tracker is evaluated on challenging video sequences taken from OTB-100 dataset [66], VOT dataset [87] and UAV123 dataset [69]. The considered video sequences include various environmental challenges viz. illumination variation, full or partial occlusion (FOC or POC), background clutters, fast motion and deformation. The experimental results are compared against 13 others state-of-the-art trackers namely, ASLA [88], STRUCK [117], MTT [89], IDCT [118], DFT [95], L1-APG [119], CT [90], WMIL [94], fDSST [120], CSPF [5], BIT [121], STAPLE [96] and DCF-CA [111]. Publicly available author's release code of these trackers is used for fair performance comparison. The proposed tracker is implemented on MATLAB 2018 on a 2.4GHz processor machine with 6 GB RAM. Initially, the fragments are initialized on the target with minimal overlap. The number of fragments initialized is highly dependent on the size of the target. Small and low resolution target requires fewer fragment during initialization in comparison to high resolution target. Also, The reference dictionary is updated with $\mathcal{P} = 20$ and $\mathcal{N} = 20$. To prove the robustness of our tracker during various environmental challenges the qualitative and quantitative analysis is described in the following sections.

5.3.1 Qualitative analysis

In this section, the qualitative analysis of the proposed tracker was performed. The details of video sequence and the considered challenge is as follows.

5.3.1.1 Illumination Variation

Sample frames under the illumination variation challenge are shown in Fig.5.3. When there is sudden change in illumination at frame #48 for *Human8* sequence, the proposed approach and STAPLE successfully tracked the target. This is due to the HOG feature descriptor in the appearance model of the proposed model which compensated for the LBP feature which was susceptible to illumination variation. OURS, DFT, WMIL, BIT and STAPLE have tracked the target with minimal error in comparison to other trackers as shown in Fig. 5.3 (b). Here, the transductive reliability with proposed fusion of multi-cue will suppress the unreliable features during tracking and prevented drift. In *Fish* sequence, OURS, STAPLE, STRUCK and DCF-CA can keep track of the target successfully and the same is depicted in Fig.5.3 (c). In Fig.5.3 (d), when there is illumination variations alongwith In-plane rotation in frames (#148 and #314), then only the proposed method with fDSST and STAPLE have shown relatively good performance. It is due to

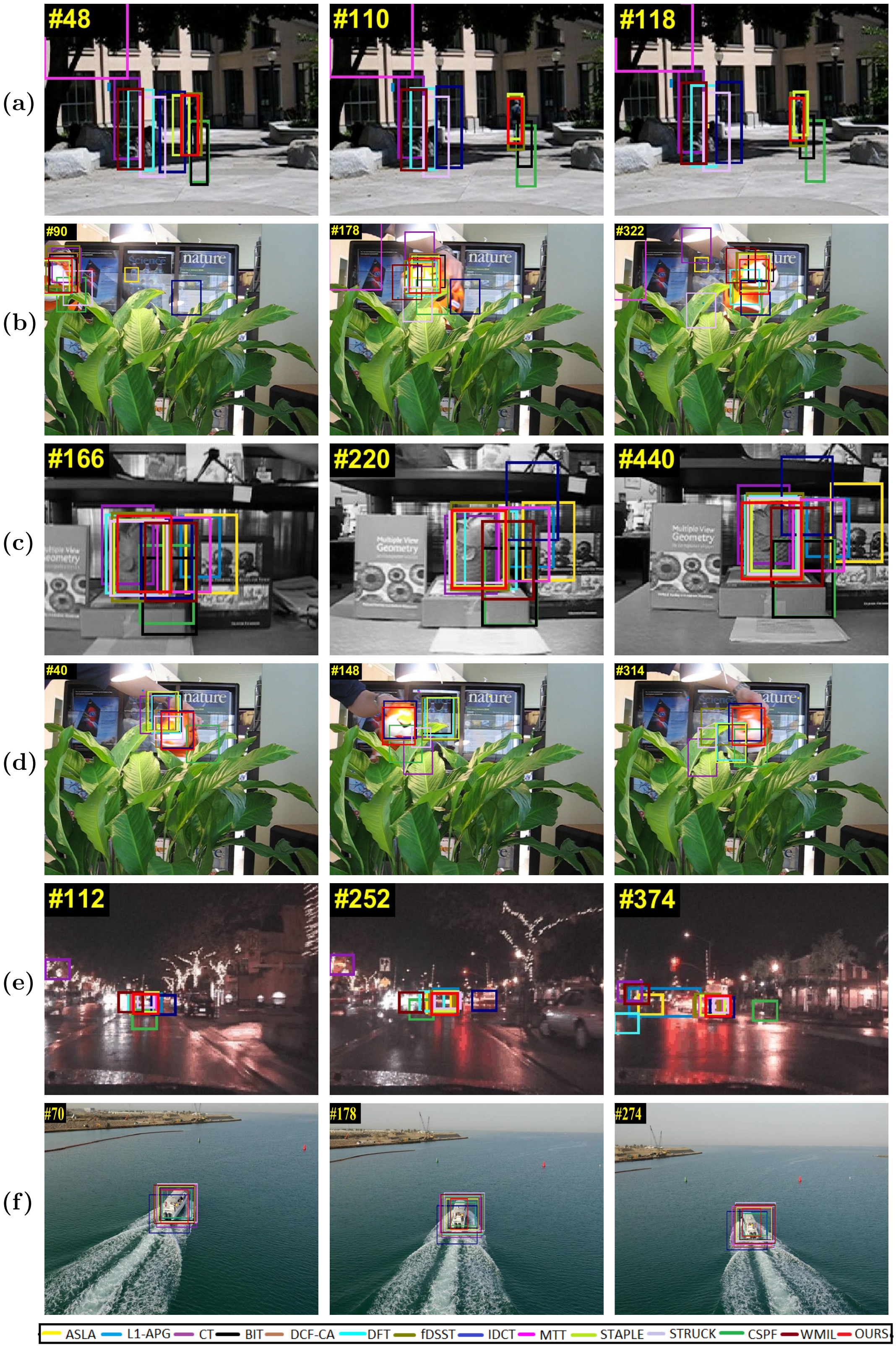

Fig. 5.3 *Critical tracking frames from video sequence under Illumination variation challenge: a)* Human8 *b)* Tiger *c)* Fish *d)* Tiger1 *e)* Car2 *f)* boat5

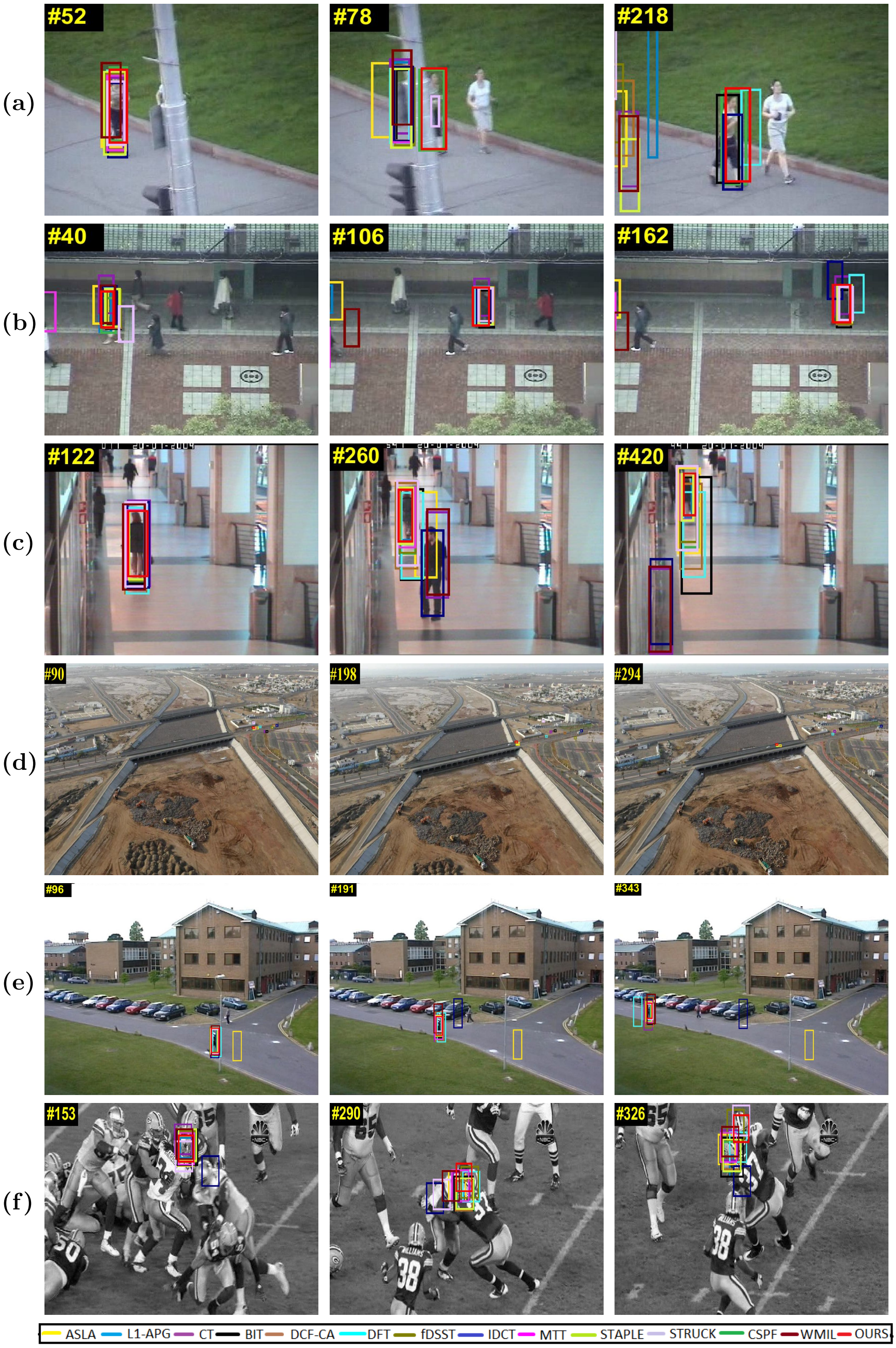

Fig. 5.4 *Critical tracking frames from video sequence under FOC or POC challenge: a)* Jogging1 *b)* Subway *c)* Walking2 *d)* car15 *e)* Walking *f)* Football

the exploited HOG and the rotation-invariant LBP features in the tracker which addressed the challenge gracefully. In *Car2* sequence, OURS, MTT and IDCT had performed substantially better in comparison to L1-APG, ASLA, WMIL, DFT and CSPF. Fig. 5.3 (f) has depicted that ASLA, OURS, DCF-CA and STAPLE have performed better in comparison to others. This is due to the robust unified feature that have been constructed by the fusion of multi-cue in the proposed tracker. Moreover, the complementary features LBP and HOG in the appearance model of the tracker improves the performance under the challenge.

5.3.1.2 Full or Partial Occlusion

Representative tracking results for *Jogging1* during full or partial occlusion are shown in Fig. 5.4 (a). At frame #78, target is fully occluded by the pillar, then OURS, IDCT, BIT and CSPF located the target successfully. This is due to the classifier mechanism which eliminated the non-reliable fragments from the tracker. In *Subway* sequence, target is occluded by various person several times. At frame #106, Generative trackers viz. ASLA, MTT and L1-APG have lost the trackers as their appearance model lack drift alleviation mechanism. However, at frame #162 trackers fDSST, STRUCK, BIT, STAPLE and OURS have performed considerably better. Here, the LBP feature has compensated for the HOG feature for developing robust appearance model to handle the challenge. Some tracking frames for *Walking2* sequence are depicted in Fig. 5.4 (c). In this sequence, target is partially occluded by the similar object. At frame #260 IDCT, ASLA

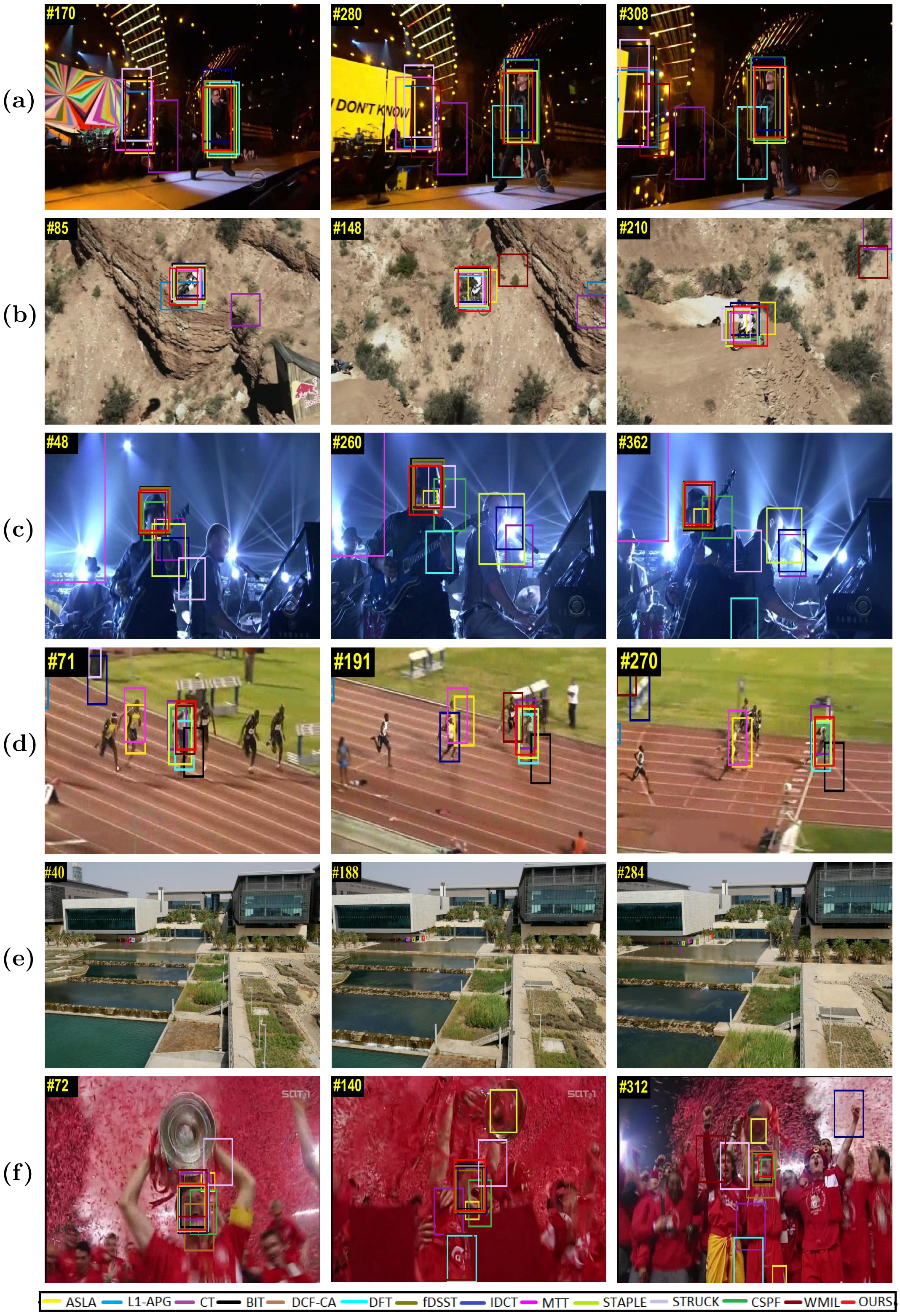

Fig. 5.5 *Critical tracking frames from video sequence under Background Clutters challenge: a)* Singer2 *b)* MountainBike *c)* Shaking *d)* Bolt2 *e)* bike2 *f)*Soccer1

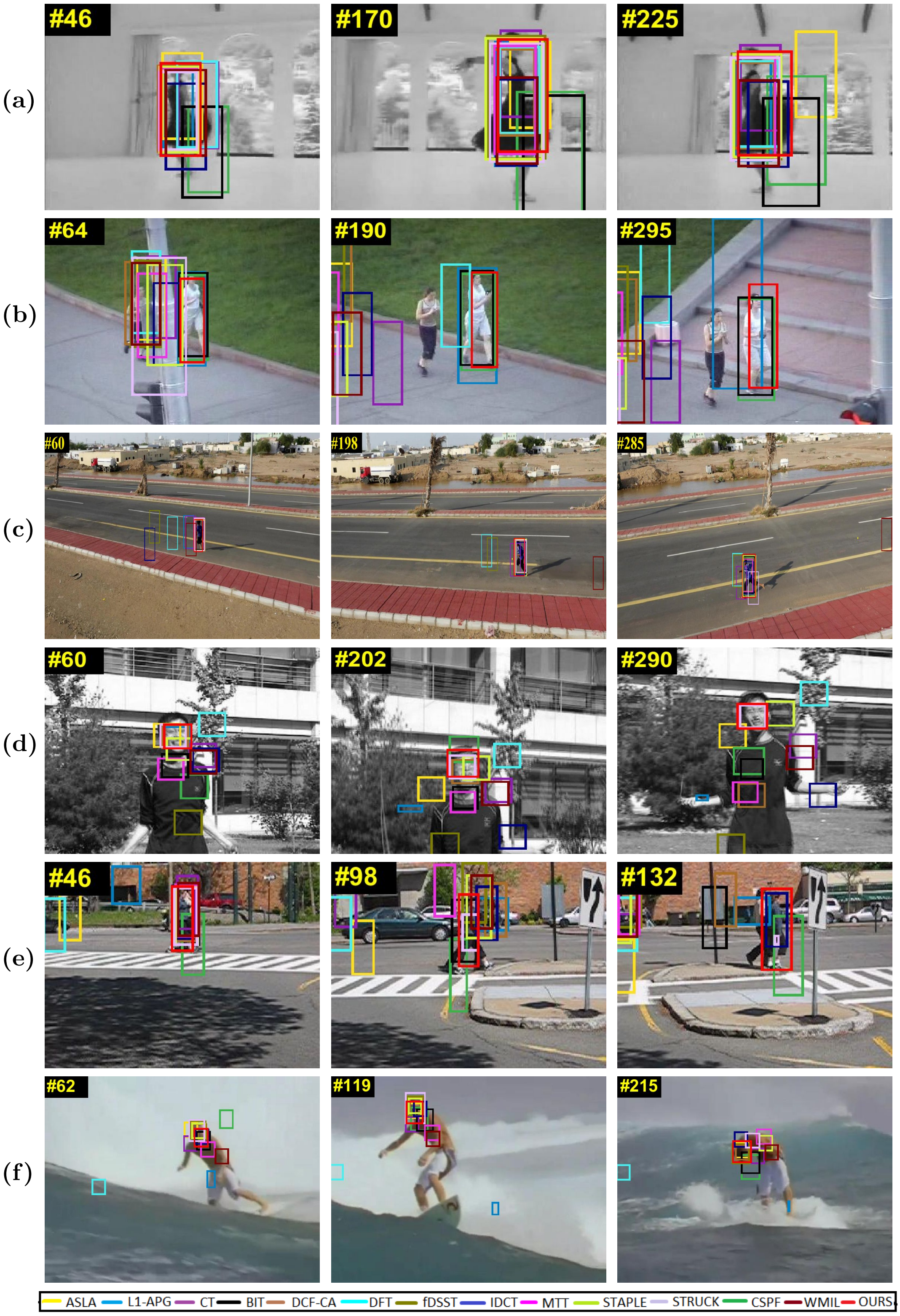

Fig. 5.6 *Critical tracking frames from video sequence under Fast Motion and Deformation challenge: a)* Dancer *b)* Jogging2 *c)* person23 *d)* Jumping *e)* Pedestrian1 *f)* Surfer

and STRUCK have lost the target and locate the other object in the scene. While OURS, STRUCK, MTT and L1-APG have kept track of the target till end. This is due to complementary LBP-HOG features exploited in the appearance model. In addition, the proposed classifier mechanism also prevents the false updation of the reference dictionary. In sequence *Walking*, ASLA, IDCT and DFT have lost the target while the rest of the trackers tracked the target with minimal error. In sequence *car15*, the target size is small, which OURS, STAPLE and STRUCK able to track while other trackers have deflected from the target. Tracking results for *Football* sequence are shown in Fig. 5.4 (d). In subsequent frames (#290 and #326), when target is occluded by other players OURS, fDSST and DFT have shown relatively good performance in comparison to STRUCK, IDCT, L1-APG, STAPLE and DCF-CA. The performance of the proposed approach under the challenge is mainly attributed due to the generated unified robust feature. Also, the proposed classifier mechanism which has classified the affected fragment during occlusion to prevent the drift of the tracker.

5.3.1.3 Background Clutters

Tracking results of the proposed tracker under this challenge are depicted in Fig. 5.5. Tracking frames results for *Singer2* are shown in Fig. 5.5 (a). At frame #280, OURS with STAPLE and DCF-CA has handled the challenge efficiently in comparison to other trackers. Here, the exploited features descriptor in the appearance model have supported the performance of the proposed approach under the challenge. In *MountainBike* sequence, at frame #210 STRUCK, L1-APG and CT have drifted away from the target whilst OURS, BIT and MTT kept track of the target in the whole sequence. *Shaking* sequence has similar background with illumination variations in the whole sequence. CT, MTT, CSPF and DFT are failed to handle the challenge and hence, drift away from the target. However, OURS, DCF-CA, BIT and WMIL have performed considerably better in the sequence. Fig.5.5 (d), at frame #276 OURS and STAPLE have tracked the target while the other trackers have located the other similar object in the scene. Critical tracking frames for *bike2* are depicted in Fig.5.5 (e), in which OURS has shown relatively better performance against other trackers. In *Soccer1* sequence, DCF-CA has performed better with marginal less error in comparison to the proposed tracker. It may be due to false updation of the proposed tracker by the unreliable fragment during the challenge. In sum, the proposed approach is able to handle the similar background as the proposed fusion of the multi-cue suppress the low level features and boost the high level features accompanied by the transductive reliability at each stage. In addition, the proposed classifier mechanism will replace the unreliable fragments and update the reference dictionary in order to maintain the performance of tracker during the challenge.

5.3.1.4 Fast Motion and Deformation

Critical tracking results illustrating the performance of the proposed approach under the challenge are shown in Fig.5.6. In *Dancer* sequence, the deformation of the target due to fast motion which is accompanied by pose variations. ASLA has lost the target but OURS with other trackers have shown better performance. Fig5.6 (b) presents the tracking results for *Jogging2* sequence. At frame #290, IDCT, ASLA, STRUCK, WMIL, STAPLE, DCF-CA and DFT have lost the target but OURS and L1-APG have tracked the target with minimal error till end. *Jumping* sequences has high target's deformation with motion blur and fast motion. OURS, STRUCK, and DCF-CA have catered this deformation to a great extent in comparison to other trackers. *person23* sequence has fast motion challenge which OURS, DCF-CA, STAPLE and CSPF have handled gracefully. In *Pedestrian1* sequence, at frame #132 DCF-CA, STAPLE, L1-APG, DFT and BIT have lost the target due to inefficient appearance model which is not able to handle the target's deformation due to fast motion and out-of plane rotation. OURS, CSPF and STRUCK have efficiently handled the deformation due to fast motion. It is due to rough localization of the target through optical flow which exploited the motion cue of the target. Tracking results for *Surfer* sequence is depicted in Fig.5.6 (e). At frame #119, the appearance model of FRAG, L1-APG, DFT, MTT and IDCT get corrupted and lose the target. OURS, STRUCK and DCF-CA have handled the variations due to fast motion and in plane rotations. This may be due to periodic updation of the tracker which removed the erroneous unreliable fragments from the tracker. In addition, the rough localization of the target by optical flow have exploited the motion cue in the appearance model. Transductive reliability is also integrated at each time step to prevent drift under target's deformation when accompanied by several pose variations, fast motion and motion blur.

Table 5.1: CLE (in pixels). Red, blue and green represent the best results.

Video Sequence	ASLA	STRUCK	MTT	DFT	L1-APG	STAPLE	WMIL	DCF-CA	CT	fDSST	IDCT	CSPF	BIT [121]	OURS
Walking2	19.98	4.21	3.96	31.70	3.99	3.79	59.90	22.07	63.20	10.67	64.78	7.21	28.77	3.73
Jogging1	99.36	123.00	107.90	34.86	88.47	91.20	94.37	89.68	92.95	102.65	15.65	12.38	3.62	7.43
Car2	20.32	3.41	1.35	31.51	12.68	1.35	42.54	5.25	102.98	3.15	21.32	23.87	5.56	5.03
Jogging2	143.90	162.08	152.78	44.01	11.59	146.76	138.36	148.57	116.98	155.17	123.06	8.07	2.84	10.11
Shaking	19.98	107.36	188.94	103.06	122.16	137.60	11.01	8.01	162.88	7.57	134.62	47.91	7.33	5.67
Singer2	193.91	166.91	197.55	39.58	182.60	7.57	160.98	10.82	121.21	12.25	27.38	14.85	180.55	9.43
Tiger	62.61	66.83	204.21	11.57	119.75	11.57	28.31	13.27	70.27	11.04	84.41	56.7	24.9	11.53
Jumping	34.13	8.31	58.89	69.40	77.69	24.82	53.84	2.94	50.85	125.46	79.40	33.99	41.43	17.26
Human8	65.56	56.44	162.03	71.05	112.73	5.77	79.25	3.02	82.65	2.55	43.13	5.58	33.61	4.02
Dancer	23.11	9.37	7.94	12.35	9.04	6.34	17.63	7.29	23.21	6.75	21.21	71.2	66.8	5.64
Pedestrian1	96.23	7.61	55.39	119.69	22.92	34.65	39.78	20.11	79.30	40.01	23.63	15.22	18.72	7.58
Soccer1	121.81	110.59	36.84	112.41	61.76	65.80	84.34	21.65	80.62	11.45	87.96	40.88	7.12	5.70
Football	15.30	13.37	12.39	9.93	17.55	13.02	14.51	13.96	15.63	6.41	70.14	10.62	14.55	9.71
Subway	145.04	4.01	165.30	5.88	149.41	2.59	136.73	2.92	11.58	2.81	8.45	4.97	3.87	2.44
MountainBike	21.73	12.36	5.62	10.30	171.98	9.03	7.11	8.20	192.44	8.59	15.43	14.89	9.54	107.06
Fish	70.91	5.98	43.88	8.51	57.35	4.29	30.99	4.51	13.06	3.59	79.34	54.08	54.83	4.26
car15	-	3.53	-	-	-	1.93	217.7	-	15.51	211.83	-	8.49	221.57	6.34
Tiger1	96.71	76.84	110.77	73.56	119.51	56.34	-	13.56	55.07	62.97	61.90	65.41	54.99	17.99
Surfer	10.56	9.78	44.31	215.73	139.05	27.39	63.74	4.98	45.58	4.11	47.36	36.96	24.01	5.25
bike2	150.65	104.8	138.67	-	153.4	206.82	284	-	-	153.22	209.31	67.08	176.43	29.89
boat5	14.81	28.67	25.36	29.11	23.84	13.67	26.48	13.74	29.11	29.08	24.85	27.69	29.52	17.89
Walking	247.76	3.13	3.41	14.29	2.73	2.09	8.63	2.94	8.55	2.09	116.83	7.35	4.27	2.70
Bolt2	102.92	-	110.46	15.20	-	6.96	119.07	-	13.03	-	150.00	9.08	36.23	11.20
person23	357.73	52.26	6.82	91.98	15.88	8.06	410.14	8.73	41.15	153.83	85.69	6.12	7.71	5.15
Average	**92.76**	**49.60**	**80.21**	**52.53**	**76.19**	**37.06**	**97.07**	**20.30**	**64.69**	**49.01**	**69.38**	**27.11**	**44.12**	**8.88**

5.3.2 Quantitative Analysis

In order to evaluate the performance of the proposed tracker under quantitative analysis four performance metrics are exploited. Two performance evaluations are tabulated as CLE and F-Measure [65]. In addition, Success plots and Precision plots are plotted to prove the robustness of the proposed approach. CLE and F-Measure comparison results on challenging benchmark video sequences are tabulated in Table 5.1 and Table 5.2 respectively. Our tracker has attained average

Table 5.2: F-Measure. Red, blue and green represent the best results.

Video Sequence	ASLA	STRUCK	MTT	DFT	L1-APG	STAPLE	WMIL	DCF-CA	CT	fDSST	IDCT	CSPF	BIT	OURS
Walking2	0.464	0.652	0.869	0.506	0.846	0.873	0.344	0.594	0.315	0.767	0.334	0.777	0.551	0.849
Jogging1	0.161	0.188	0.201	0.221	0.191	0.201	0.138	0.204	0.191	0.207	0.767	0.718	0.884	0.790
Car2	0.551	0.743	0.929	0.492	0.679	0.929	0.258	0.781	0.030	0.860	0.371	0.173	0.771	0.782
Jogging2	0.133	0.151	0.147	0.271	0.741	0.161	0.133	0.139	0.092	0.144	0.152	0.820	0.091	0.742
Shaking	0.323	0.081	-	0.236	-	0.057	0.781	0.821	0.051	0.848	0.020	0.292	0.051	0.903
Singer2	0.053	0.054	0.049	0.561	0.044	0.875	0.064	0.836	0.137	0.811	0.699	0.751	0.138	0.859
Tiger	0.127	0.262	-	0.813	-	0.805	0.587	0.786	0.191	0.807	0.251	0.249	0.588	0.809
Jumping	0.176	0.707	0.113	0.031	0.126	0.343	0.049	0.579	0.055	0.061	0.071	0.211	0.055	0.508
Human8	0.130	0.193	-	0.127	-	0.749	0.087	0.127	0.063	0.887	0.284	0.749	0.250	0.827
Dancer	0.631	0.811	0.814	0.745	0.788	0.872	0.732	0.773	0.662	0.874	0.703	0.227	0.250	0.881
Pedestrian1	0.056	0.601	0.334	0.020	0.360	0.596	0.526	0.527	0.237	0.530	0.434	0.542	0.524	0.738
Soccer1	0.059	0.156	-	0.200	-	0.260	0.182	0.494	0.197	0.663	0.302	0.213	0.722	0.766
Football	0.631	0.671	0.671	0.706	0.574	0.678	0.593	0.664	0.612	0.777	0.113	0.692	0.627	0.711
Subway	0.194	0.801	0.081	0.768	0.186	0.850	0.189	0.844	0.699	0.836	0.757	0.758	0.818	0.877
MountainBike	0.633	0.774	0.824	0.804	0.308	0.832	0.436	0.822	0.222	0.838	0.713	0.728	0.822	0.871
Fish	0.171	0.885	0.297	0.854	0.072	0.877	0.471	0.909	0.774	0.905	0.147	0.276	0.250	0.929
car15	-	0.643	0.066	0.062	-	0.676	0.092	0.059	0.418	0.072	0.043	0.292	0.032	0.519
Tiger1	-	0.213	-	0.303	-	0.303	-	0.820	0.125	0.268	0.732	0.221	0.311	0.736
Surfer	0.512	0.651	0.056	0.025	0.033	0.275	0.038	0.666	0.087	0.815	0.213	0.182	0.250	0.817
bike2	-	0.268	0.049	0.048	0.214	0.146	0.001	0.049	0.002	0.212	0.05	0.001	0.143	0.139
boat5	0.003	0.585	0.601	0.578	0.707	0.629	0.572	0.602	0.565	0.587	0.549	0.633	0.699	0.777
Walking	0.069	0.811	0.841	0.475	0.744	0.848	0.659	0.686	0.643	0.860	0.324	0.550	0.769	0.866
Bolt2	0.024	0.019	0.013	0.669	0.013	0.803	0.387	0.014	0.645	0.014	0.162	0.759	0.250	0.679
person23	-	0.528	0.808	0.047	0.664	0.844	0.035	0.776	0.272	0.06	0.453	0.862	0.836	0.875
Average	**0.261**	**0.477**	**0.409**	**0.398**	**0.405**	**0.603**	**0.319**	**0.566**	**0.303**	**0.571**	**0.360**	**0.487**	**0.445**	**0.760**

CLE of 8.88 (in pixels) and average F-Meaure of 0.760. In addition, in Figs. 5.7 and 5.8 precision plot and success plot are illustrated, respectively. Plots reveal that our tracker has attained high precision values and high success rate. In sequences with illumination variation challenge, the proposed tracker has achieved superior performance due to exploited complementary HOG and LBP features. High precision values during FOC and POC have confirmed the effectiveness of the proposed fusion of multi-cue features. During background clutters, deformation and fast motion, the suppression of deteriorating cue over the effective cues

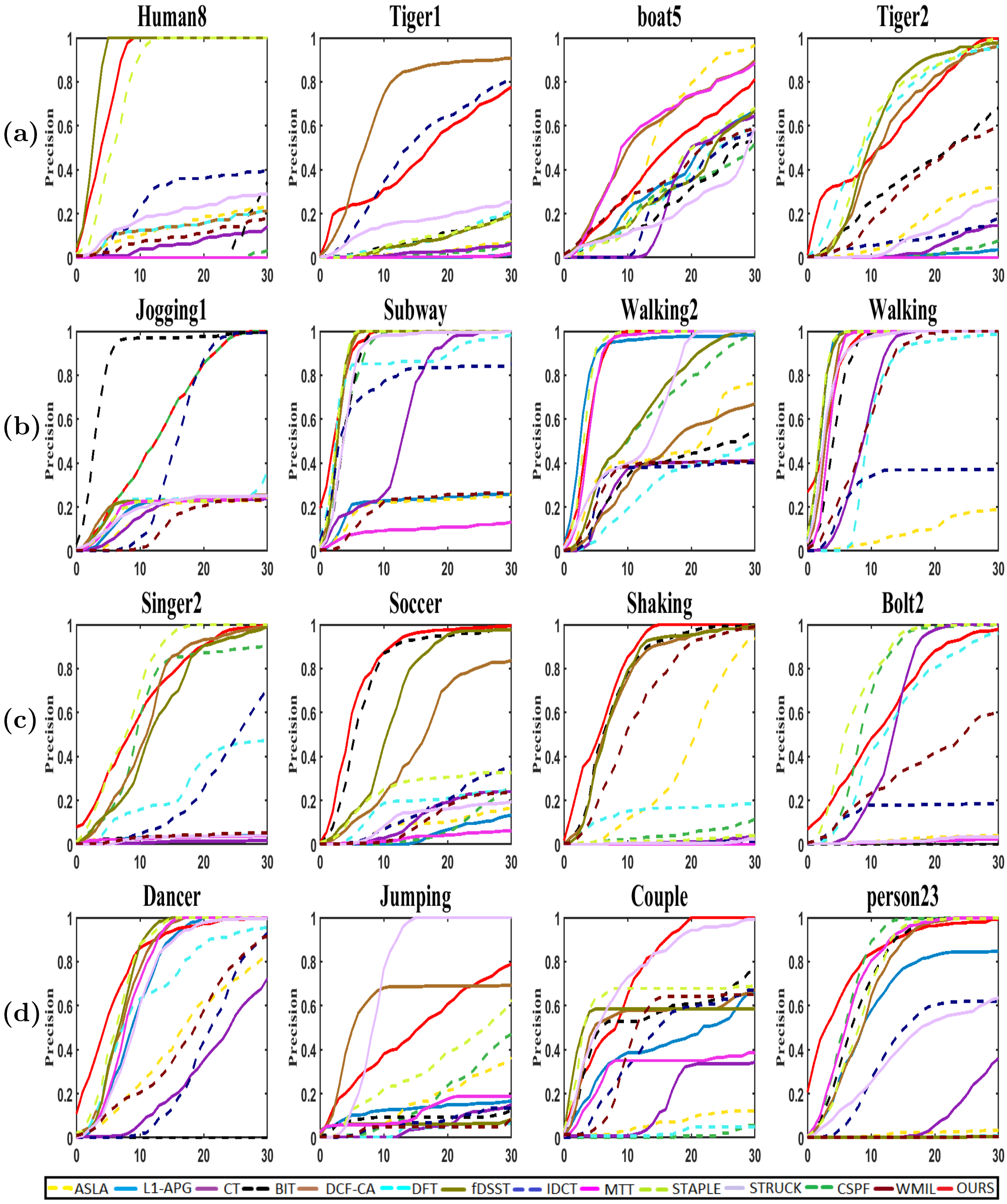

Fig. 5.7 *Precision Plot under visual tracking challenges: a) Illumination Variations (*Human8, Tiger1, boat5, Tiger2*) b) Full or Partial Occlusion (*Jogging1, Subway, Walking2, Walking*) c) Background Clutters (*Singer2, Soccer, Shaking, Bolt2*) d) Deformation and Fast Motion (*Dancer, Jumping, Couple, person23*)*

by the transductive reliability alongwith the proposed classifier mechanism have validated the strength of the proposed method. Also, rough localization of the target using optical flow has estimated the motion information more accurately.

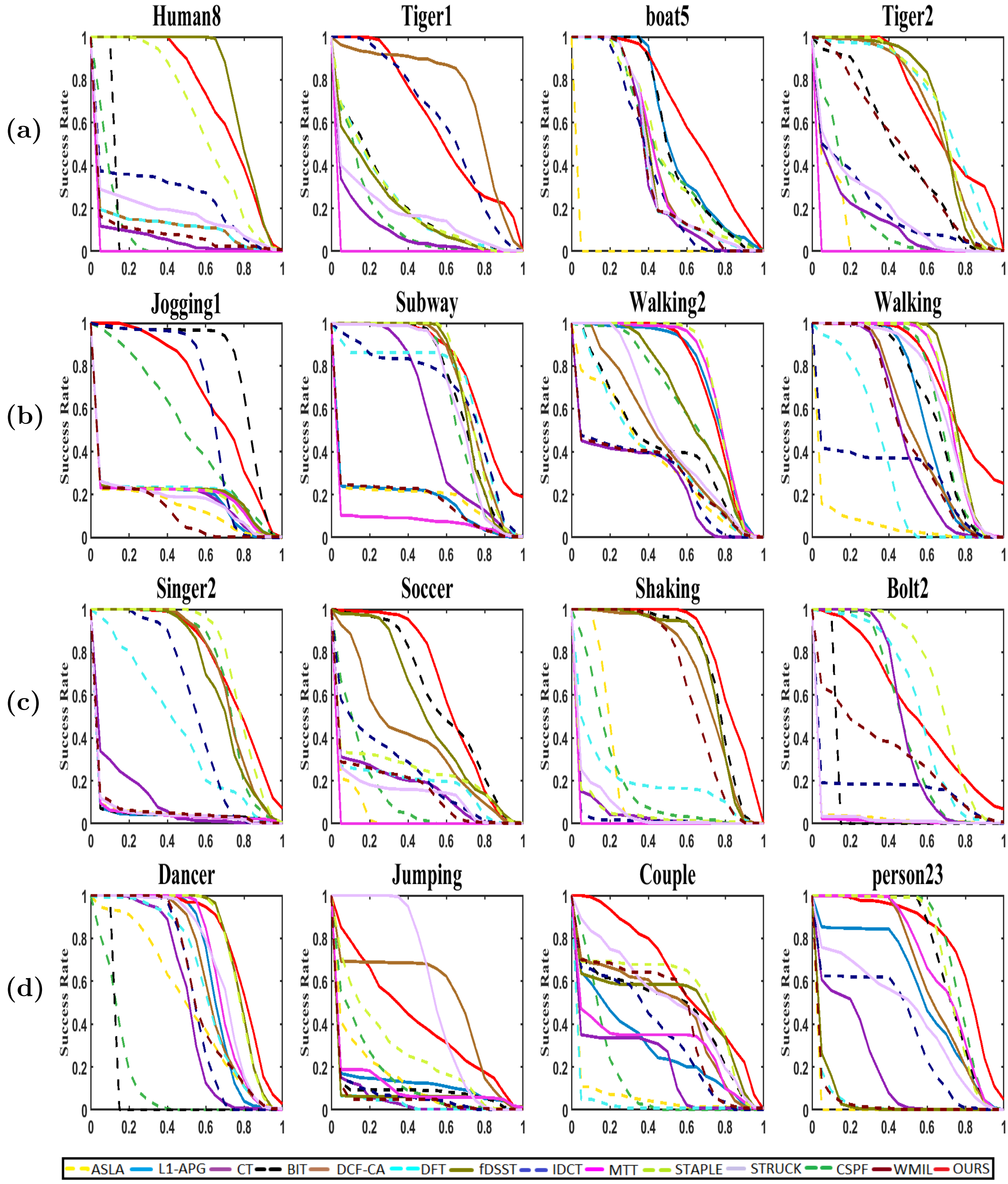

Fig. 5.8 *Success plot under visual tracking challenges: a) Illumination Variations (*Human8, Tiger1, boat5, Tiger2*) b) Full or Partial Occlusion (*Jogging1, Subway, Walking2, Walking*) c) Background Clutters (*Singer2, Soccer, Shaking, Bolt2*) d) Deformation and Fast Motion (*Dancer, Jumping, Couple, person23*)*

Our tracker has performed well in tough situations by catering various dynamic environmental challenges.

Our tracker has shown eminent performance in comparison to the existing state-of-the-art trackers. Low mean CLE and high mean F-Measures validate the strength of proposed tracker. In addition, Precision plot and success plot reveal that the proposed tracker outperformed the other trackers. Generative trackers, like MTT, ASLA, and L1-APG are not able to handle background clutters and FOC. Discriminative trackers, like WMIL and CT are failed to handle target's deformation due to pose variations. STAPLE and CSPF are not able to handle target's deformation due to fast motion and motion blur efficiently. DCF-CA has shown comparable performance but not suitable for real-time computation whereas fDSST has shown resistance to occlusion and fast motion. STRUCK is able to handle background clutters challenge to a great extent but was relatively slow due to complex learning methods in the update model. WMIL had random Haar-like features in appearance model and not able to cater target deformation due to rotation. MTT and L1-APG has exploited sparse representation using single feature in the appearance model and drift away from the target during tough environmental challenges. In sum, the proposed tracker has performed better than the other trackers by exploiting complementary multi-cue LBP and HOG in the appearance model during precise localization of the target. During rough localization, proposed approach has exploited the motion cue using optical flow to handle the target deformation due to fast motion. The fusion of multi-cue generated a robust unified feature to overcome occlusion and background clutters. Nevertheless, the proposed K-Means based classifier mechanism has classified the positive reliable fragments and hence, prevents the false updation of the reference dictionary during environmental challenges.

5.4 Significant Findings

The following were the key highlights of the proposed multi-cue tracker under multi-stage framework.

- A novel multi-stage tracking framework for object tracking.
- Optical flow exploited the motion cue for rough estimation of the target.
- During precise localization, complementary features viz. scale-invariant LBP and HOG in appearance model handled scale variations and occlusion.
- The proposed fusion of feature vectors improved the computational efficiency of the tracker during background clutters.
- K-Means based classifier mechanism classified the positive fragments and the negative fragments and hence, prevented the false updation of the reference dictionary.
- Transductive reliability of each fragment was integrated for quick adaptation of the target during environmental challenges.
- The proposed method has achieved the computational efficiency of 22 frames/sec.

- Quantitative and qualitative analysis on challenging video sequences inferred the competitive results of the proposed tracker against 13 other trackers.

In addition, the experimental results along with significant findings of the proposed work were published in [122].

Chapter 6

Conclusion and Future Directions

Chapter 6

Conclusion and Future Directions

This chapter will summarize the major contributions and achievements that come out of the present work. Despite the significant contributions, no research is said to be complete unless it directs to a few topics for future research. Hence, the potential work that can be explored further under present studies is briefly discussed as directions to future work in the Section. 6.2.

6.1 Summary of Major Contributions

The key idea of this thesis work was to design and develop a robust multi-cue object tracking model under various frameworks namely, stochastic, deterministic and multi-stage. To address the limitations of each individual framework, several novel contributions proposed under present work sre summarized as follows.

- Multiple video sequences were captured from vision camera and analyzed in a self generated dataset. Captured video sequences were of adequate length and annotated to determine the real-time performance of various trackers.

In addition, latest trends in multi-cue object tracking frameworks which exploited the complementary cue information were analyzed. Multi-cue object tracking under various frameworks was investigated and briefly reviewed.

- A real-time multi-cue tracker under particle filter framework was proposed in which likelihood model was constructed by considering three complementary multi-cues namely CH, LBP and PHOG. RGB cue was robust against change in scale and partial occlusion but deteriorates during illumination variations and background clutter. However, LBP texture cue was invariant to change in illumination and background clutters but could not handle scale variations and target's deformation. while the PHOG can efficiently handle the target's deformation and rotation. These complementary cues were considered with the aim so that if one cue fail the other may compensate during the dynamic environment. Resampling technique based on Butterfly search optimization was proposed to address the shortcomings of PF. This optimization had a fast convergence rate along-with its two parameters: 1) Switch probability 2) Solution vector. These two parameters diversified the particles in high likelihood area of search space to prevent particle degeneracy and sample impoverishment. An adaptive fusion method for fusion of multi-cue was proposed to cater dynamic environmental variations. This fusion model ensured the automatic boosting of the important particles and suppression of the unimportant particles during occlusion, background clutter and target's deformation. In addition, context sensitive cue reliability was estimated to ensure the quick adaptation of the tracker during real-time environmental challenges. Also, outlier detection mechanism was exploited to detect low performing particles as outliers. In each subsequent frame, outliers were detected which dispense less towards the state estimation. This mechanism not only helped in improving the efficiency of the proposed resampling method but also reduced the computational complexity of the proposed tracker.

- A robust object tracking framework was proposed which included fuzzy based fusion of multi-cue to create the clear determination boundary between the positive fragments and the negative fragment. The proposed fusion model ensured the high level relationships between the features were captured and their weak relationships were suppressed to make tracker robust to background clutters and pose variations. The multi-stage estimation considered the coarse to fine localization of the target. The two step localization of target's position not only enhanced tracker's accuracy during environmental challenges but also, reduced its computational load. During rough localization, the motion cue was calculated using optical flow to address the fast motion and rotational variations in target. This enhanced tracker's performance during next step of localization. During precise localization, an adaptive appearance model was proposed to maintain the target's temporal and context consistency. For this, complementary multi-cue namely, RGB and HOG were exploited in the tracker's appearance model. RGB cue was computationally inexpensive and robust to scale variations and partial occlusion but failed to address full occlusion, similar background and illumination variations. Unlike RGB cue, HOG was invariant to these challenges and had effectively improved the tracker's performance under the tracking variations. Random forest based discriminative classifier mechanism ensured the highly marginalized boundary was created between the positive fragments and the negative fragments. The classifier provided confidence score to each candidate fragment and helped in the decision to select the efficient fragments for precise localization of target. In addition, this also assured the periodic update of the reference dictionary with effective fragments and hence, prevented the eventual drift of the tracker during occlusion and background clutter.

- Multi-stage tracking framework was proposed which included the coarse to fine estimation of the target. This multi-stage estimation reduced the computational complexity of the tracker. For this, initially the target was roughly localized using optical flow. Optical flow estimated the motion cue for each pixel of the target and hence, improved the accuracy of tracker during precise localization. During precise estimation of the target complementary features namely, LBP and HOG were exploited in the appearance model to handle illumination variation, background clutters and occlusion. Scale-invariant LBP feature determined local pattern information while HOG feature calculated edge information of the target and hence, was complementary to each other. Strength of scale invariant LBP feature vector and HOG feature vector were fused by determining the Euclidean distance between the feature vectors. The distance is determined between the feature descriptors and the samples from the reference dictionary. The proposed fusion not only analyzed the high order relationship between the features but also diminished the weak relationship among them. At each step, the tracker was made adaptive by integrating transductive reliability for each fragment sample. This reliability ensured that the tracker was adaptive with the change in the environment. Discriminative classifier mechanism based on K-Means was proposed to classify the positive samples from the negative samples. Classifier score created a clear decision boundary between the positive fragment samples and negative fragment samples. This process maintained the tracker's accuracy during background clutters and occlusion by preventing the false updation of the tracker.

- The proposed tracker using stochastic approach under particle filter framework had shown efficient tracking performance by handling the various tracking challenges. Multi-cue based appearance model exploited complementary cue information in which each cue compensated for each other during change in environmental conditions. Outlier detection mechanism was able to identify the low performing particles which were affected due to visual tracking challenges and dispensed less towards state estimation. Meta-heuristic optimization based resampling technique addressed the shortcomings of the particle filter effectively. The proposed tracker under particle filter framework had shown effective and efficient performance under various tracking challenges. However, the tracker tended to drift under occlusion and pose variations due to lack of background information in the appearance model. In order to address this, we have proposed a tracker under deterministic framework. This tracker exploited fragment based approach in the appearance model. In addition, the reference dictionary was initialized with a set of positive and two set of negative samples to include both foreground as well as background information. The set of positive samples included the target region, one set of negative sample comprised of the nearby background area of the target and the another negative set contained the samples of the other target present in the scene. The periodic update of reference dictionary ensured the consistent performance of the tracker during tough environmental challenges. The appearance model was made robust to occlusion, deformations and background clutters by exploiting the multi-cue and background information. In order to further enhanced the tracker's performance under visual tracking challenges by improving the computational complexity, we have proposed a tracking architecture under multi-stage framework. The rough localization of the target during initial stage improved its precise localization during the final state estimation. This multi-stage estimation reduced the computational complexity of the tracker and handled fast motion and target's rotational challenges effectively. Also, reliability metric was computed to prevent the appearance model from updating with fragments containing background clutters. Nevertheless, using regular updates for a dynamic appearance model and periodic updates of reference templates prevented the eventual drift of tracker in long video sequences.

- Exhaustive experimental evaluation that included both qualitative and quantitative analyses on benchmark video sequences from OTB, VOT2017 and

UAV123 proved the robustness of the various proposed trackers against other state-of-the-art trackers during various object tracking challenges. Under particle filter framework, the proposed approach achieved superior performance against 13 other stat-of-the-art trackers. On average of the outcome, our tracker under stochastic approach achieved CLE 6.89 (in pixels) and F-measure of 0.786 against 13 others state-of-the-art. Further, the proposed tracker under deterministic approach attained average CLE of 5.78 and average F-Measure of 0.806 on challenging video sequences under various tracking variations. However, the tracker proposed under multi-stage framework exhibited real-time performance and attained average CLE of 8.88 (in pixels) and average F-measure of 0.760 on video sequences from multiple benchmarked datasets.

6.2 Directions of Future Work

In the present work, multi-cue object tracking model under various framework were investigated and explored at length to provide novel contributions to the domain. Despite that, there are certain research areas that emerge out of the present work which demand future investigation. These areas are summarized as directions to future work and are detailed as follows.

- Object tracking work under particle filter framework can be extended to track multiple people in the video. Also, adaptive online learning can be explored for directing particles under uncertain environmental variations.

- Spatial and context relationships between the subsequent frames to develop a holistic representation of the targeted object can be further investigated.
- Outlier detection procedure can be explored with fuzzy decision boundary for generating the clearer decision discriminability.
- Multi-stage tracker can be explored for integrating more visual cue at each stage to provide more robust tracking solution.
- Another possible extension can be made by utilizing the multi-modal information captured from multiple sensors in the tracker's appearance model.
- The fusion model can be explored by incorporating user-defined source importance information as another potential direction for the future undertaking.

References

[1] Arnold WM Smeulders, Dung M Chu, Rita Cucchiara, Simone Calderara, Afshin Dehghan, and Mubarak Shah. Visual tracking: An experimental survey. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1442–1468, 2014.

[2] Hanxuan Yang, Ling Shao, Feng Zheng, Liang Wang, and Zhan Song. Recent advances and trends in visual tracking: A review. *Neurocomputing*, 74(18):3823–3831, 2011.

[3] Gurjit Singh Walia and Rajiv Kapoor. Recent advances on multicue object tracking: a survey. *Artificial Intelligence Review*, 46(1):1–39, 2016.

[4] X. Qian, L. Han, Y. Wang, and M. Ding. Deep learning assisted robust visual tracking with adaptive particle filtering. *Signal Processing: Image Communication*, 60:183–192, 2018.

[5] Gurjit Singh Walia, Ashish Kumar, Astitwa Saxena, Kapil Sharma, and Kuldeep Singh. Robust object tracking with crow search optimized multi-cue particle filter. *Pattern Analysis and Applications*, 23(3):1439–1455, 2020.

[6] M. Liu, C. Jin, B. Yang, X. Cui, and H. Kim. Online multiple object tracking using confidence score-based appearance model learning and hierarchical data association. *IET Comp. Vis.*, 13(3):312–318, 2018.

[7] D. Du, H. Qi, W. Li, L. Wen, Q. Huang, and S. Lyu. Online deformable object tracking based on structure-aware hyper-graph. *IEEE Transactions on Image Processing*, 25(8):3572–3584, 2016.

[8] Jingjing Xiao, Rustam Stolkin, Mourad Oussalah, and Aleš Leonardis. Continuously adaptive data fusion and model relearning for particle filter tracking with multiple features. *IEEE Sensors Journal*, 16(8):2639–2649, 2016.

[9] Jingjing Xiao, Rustam Stolkin, Yuqing Gao, and Aleš Leonardis. Robust fusion of color and depth data for rgb-d target tracking using adaptive range-invariant depth models and spatio-temporal consistency constraints. *IEEE transactions on cybernetics*, 48(8):2485–2499, 2017.

[10] Q. Xie, O. Remil, Y. Guo, M. Wang, M. Wei, and J. Wang. Object detection and tracking under occlusion for object-level rgb-d video segmentation. *IEEE Transactions on Multimedia*, 20(3):580–592, 2017.

[11] C. Li, X. Sun, X. Wang, L. Zhang, and J. Tang. Grayscale-thermal object tracking via multitask laplacian sparse representation. *IEEE Trans. on Systems, Man, and Cybernetics: Systems*, 47(4):673–681, 2017.

[12] Qingju Liu, Teofilo de Campos, Wenwu Wang, Philip Jackson, and Adrian Hilton. Person tracking using audio and depth cues. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 22–30, 2015.

[13] Xinyuan Qian, Alessio Brutti, Maurizio Omologo, and Andrea Cavallaro. 3d audio-visual speaker tracking with an adaptive particle filter. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2896–2900. IEEE, 2017.

[14] Hanxi Li, Yi Li, and Fatih Porikli. Deeptrack: Learning discriminative feature representations online for robust visual tracking. *IEEE Transactions on Image Processing*, 25(4):1834–1848, 2016.

[15] Baojie Fan and Huizhi Chen. Context-aware long-term correlation tracking with hierarchical convolutional features. *Pattern Recognition Letters*, 2018.

[16] Martin Danelljan, Goutam Bhat, Susanna Gladh, Fahad Shahbaz Khan, and Michael Felsberg. Deep motion and appearance cues for visual tracking. *Pattern Recognition Letters*, 124:74–81, 2019.

[17] Shiuh-Ku Weng, Chung-Ming Kuo, and Shu-Kang Tu. Video object tracking using adaptive kalman filter. *Journal of Visual Communication and Image Representation*, 17(6):1190–1208, 2006.

[18] Neil Gordon, B Ristic, and S Arulampalam. Beyond the kalman filter: Particle filters for tracking applications. *Artech House, London*, 830(5):1–4, 2004.

[19] Neil J Gordon, David J Salmond, and Adrian FM Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. In *IEE Proceedings F (Radar and Signal Processing)*, volume 140, pages 107–113. IET, 1993.

[20] Bobin Zhang, Xiuyan Shao, Wei Chen, Fangming Bi, Weidong Fang, Tongfeng Sun, and Chaogang Tang. Visual tracking based on robust appearance model. *Image and Vision Computing*, 89:211–221, 2019.

[21] Zhizhen Chi, Hongyang Li, Huchuan Lu, and Ming-Hsuan Yang. Dual deep network for visual tracking. *IEEE Transactions on Image Processing*, 26(4):2005–2015, 2017.

[22] Zhangjian Ji, Kai Feng, and Yuhua Qian. Part-based visual tracking via structural support correlation filter. *Journal of Visual Communication and Image Representation*, 64:102602, 2019.

[23] Gurjit Singh Walia, Saim Raza, Anjana Gupta, Rajesh Asthana, and Kuldeep Singh. A novel approach of multi-stage tracking for precise localization of target in video sequences. *Expert Systems with Applications*, 78:208–224, 2017.

[24] Xiangyuan Lan, Andy J Ma, Pong C Yuen, and Rama Chellappa. Joint sparse representation and robust feature-level fusion for multi-cue visual tracking. *IEEE Transactions on Image Processing*, 24(12):5826–5841, 2015.

[25] Gurjit Singh Walia, Himanshu Ahuja, Ashish Kumar, Nipun Bansal, and Kapil Sharma. Unified graph-based multicue feature fusion for robust visual tracking. *IEEE Transactions on Cybernetics*, 50(6):2357–2368, 2019.

[26] Irene Anindaputri Iswanto and Bin Li. Visual object tracking based on mean-shift and particle-kalman filter. *Procedia computer science*, 116:587–595, 2017.

[27] Fasheng Wang, Baowei Lin, Junxing Zhang, and Xucheng Li. Object tracking using langevin monte carlo particle filter and locality sensitive histogram based likelihood model. *Computers & Graphics*, 70:214–223, 2018.

[28] Pojala Chiranjeevi and Somnath Sengupta. Rough-set-theoretic fuzzy cues-based object tracking under improved particle filter framework. *Ieee Transactions On Fuzzy Systems*, 24(3):695–707, 2016.

[29] Meng Cai-Xia and Zhang Xin-Yan. Object tracking method based on particle filter of adaptive patches combined with multi-features fusion. *Multimedia Tools and Applications*, 78(7):8799–8811, 2019.

[30] Faegheh Sardari and Mohsen Ebrahimi Moghaddam. A hybrid occlusion free object tracking method using particle filter and modified galaxy based search meta-heuristic algorithm. *App. Soft Comp.*, 50:280–299, 2017.

[31] Ruohong Huan, Shenglin Bao, Chu Wang, and Yun Pan. Anti-occlusion particle filter object-tracking method based on feature fusion. *IET Image Processing*, 12(9):1529–1540, 2018.

[32] Marjan Firouznia, Karim Faez, Hamidreza Amindavar, and Javad Alikhani Koupaei. Chaotic particle filter for visual object tracking. *Journal of Visual Communication and Image Representation*, 53:1–12, 2018.

[33] Younes Dhassi and Abdellah Aarab. Visual tracking based on adaptive interacting multiple model particle filter by fusing multiples cues. *Multimedia Tools and Applications*, 77(20):26259–26292, 2018.

[34] Gurjit Singh Walia and Rajiv Kapoor. Online object tracking via novel adaptive multicue based particle filter framework for video surveillance. *International Journal on Artificial Intelligence Tools*, 27(06):1850023, 2018.

[35] Mengjie Hu, Zhen Liu, Jingyu Zhang, and Guangjun Zhang. Robust object tracking via multi-cue fusion. *Signal Processing*, 139:86–95, 2017.

[36] Gurjit Singh Walia and Rajiv Kapoor. Robust object tracking based upon adaptive multi-cue integration for video surveillance. *Multimedia Tools and Applications*, 75(23):15821–15847, 2016.

[37] Jianfang Dou and Jianxun Li. Robust visual tracking based on interactive multiple model particle filter by integrating multiple cues. *Neurocomputing*, 135:118–129, 2014.

[38] Issam Elafi, Mohamed Jedra, and Noureddine Zahid. Tracking objects with co-occurrence matrix and particle filter in infrared video sequences. *IET Computer Vision*, 12(5):634–639, 2018.

[39] Xiaomin Zhang, Kan Ren, Minjie Wan, Guohua Gu, and Qian Chen. Infrared small target tracking based on sample constrained particle filtering and sparse representation. *Infrared Physics & Technology*, 87:72–82, 2017.

[40] José Guedes dos Santos Júnior and João Paulo Silva do Monte Lima. Particle swarm optimization for 3d object tracking in rgb-d images. *Computers & Graphics*, 76:167–180, 2018.

[41] Wanyi Li, Peng Wang, and Hong Qiao. Top–down visual attention integrated particle filter for robust object tracking. *Signal Processing: Image Communication*, 43:28–41, 2016.

[42] Jian-fang Dou and Jian-xun Li. Robust visual tracking base on adaptively multi-feature fusion and particle filter. *Optik-International Journal for Light and Electron Optics*, 125(5):1680–1686, 2014.

[43] Tianzhu Zhang, Changsheng Xu, and Ming-Hsuan Yang. Learning multi-task correlation particle filters for visual tracking. *IEEE transactions on pattern analysis and machine intelligence*, 41(2):365–378, 2018.

[44] Miaohui Zhang, Ming Xin, and Jie Yang. Adaptive multi-cue based particle swarm optimization guided particle filter tracking in infrared videos. *Neurocomputing*, 122:163–171, 2013.

[45] Hua Han, Yong-Sheng Ding, Kuang-Rong Hao, and Xiao Liang. An evolutionary particle filter with the immune genetic algorithm for intelligent video target tracking. *Comp. & Maths. with App.*, 62:2685–2695, 2011.

[46] Minghao Yin, Jin Zhang, Hongguang Sun, and Wenxiang Gu. Multi-cue-based camshift guided particle filter tracking. *Expert Systems with Applications*, 38(5):6313–6318, 2011.

[47] Gurjit Singh Walia and Rajiv Kapoor. Intelligent video target tracking using an evolutionary particle filter based upon improved cuckoo search. *Expert Systems with Applications*, 41(14):6315–6326, 2014.

[48] Jinhang Liu and Xian Zhong. An object tracking method based on mean shift algorithm with hsv color space and texture features. *Cluster Computing*, pages 1–12, 2018.

[49] Rui Yao, Guosheng Lin, Chunhua Shen, Yanning Zhang, and Qinfeng Shi. Semantics-aware visual object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(6):1687–1700, 2019.

[50] Xin Wang, Zhiqiang Hou, Wangsheng Yu, Lei Pu, Zefenfen Jin, and Xianxiang Qin. Robust occlusion-aware part-based visual tracking with object scale adaptation. *Pattern Recognition*, 81:456–470, 2018.

[51] S Kanagamalliga and S Vasuki. Contour-based object tracking in video scenes through optical flow and gabor features. *Optik*, 157:787–797, 2018.

[52] Mayur Rajaram Parate, Vishal R Satpute, and Kishor M Bhurchandi. Global-patch-hybrid template-based arbitrary object tracking with integral channel features. *Applied Intelligence*, 48(2):300–314, 2018.

[53] Oumaima Sliti, Habib Hamam, and Hamid Amiri. Clbp for scale and orientation adaptive mean shift tracking. *Journal of King Saud University-Computer and Information Sciences*, 30(3):416–429, 2018.

[54] Saadia Medouakh, Mohamed Boumehraz, and Nadjiba Terki. Improved object tracking via joint color-lpq texture histogram based mean shift algorithm. *Signal, Image and Video Processing*, 12(3):583–590, 2018.

[55] Chenglong Li, Chengli Zhu, Shaofei Zheng, Bin Luo, and Jing Tang. Two-stage modality-graphs regularized manifold ranking for rgb-t tracking. *Signal Processing: Image Communication*, 68:207–217, 2018.

[56] Ye Liu, Xiao-Yuan Jing, Jianhui Nie, Hao Gao, Jun Liu, and Guo-Ping Jiang. Context-aware three-dimensional mean-shift with occlusion handling for robust object tracking in rgb-d videos. *IEEE Transactions on Multimedia*, 21(3):664–677, 2019.

[57] Chansu Kim, Donghui Song, Chang-Su Kim, and Sung-Kee Park. Object tracking under large motion: Combining coarse-to-fine search with superpixels. *Information Sciences*, 480:194–210, 2019.

[58] Bin Kang, Wei-Ping Zhu, and Dong Liang. Robust multi-feature visual tracking via multi-task kernel-based sparse learning. *IET Image Processing*, 11(12):1172–1178, 2017.

[59] Howard Wang, Sing Kiong Nguang, and Jiwei Wen. Robust video tracking algorithm: a multi-feature fusion approach. *IET Computer Vision*, 12(5):640–650, 2018.

[60] Ong Kok Meng, Ong Pauline, Sia Chee Kiong, and Low Ee Soong. Effective moving object tracking using modified flower pollination algorithm for visible

image sequences under complicated background. *Applied Soft Computing*, page 105625, 2019.

[61] Stefan Duffner and Christophe Garcia. Using discriminative motion context for online visual object tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 26(12):2215–2225, 2015.

[62] Wei Wang, Chunping Wang, Si Liu, Tianzhu Zhang, and Xiaochun Cao. Robust target tracking by online random forests and superpixels. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(7):1609–1622, 2017.

[63] Yong Wang, Xinbin Luo, Shan Fu, and Shiqiang Hu. Context multi-task visual object tracking via guided filter. *Signal Processing: Image Communication*, 62:117–128, 2018.

[64] Shuifa Sun, Shichao Liu, Shiwei Kang, Chong Xia, Zhiping Dan, Bangjun Lei, and Yirong Wu. Improved dual-mode compressive tracking integrating balanced colour and texture features. *IET Computer Vision*, 12(8):1200–1206, 2018.

[65] Neda Lazarevic-McManus, JR Renno, Dimitrios Makris, and Graeme A Jones. An object-based comparative methodology for motion det. based on the f-measure. *Comp. Vis. and Image Un.*, 111:74–85, 2008.

[66] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Object tracking benchmark. *IEEE Transactions on PAMI*, 37(9):1834–1848, 2015.

[67] Pengpeng Liang, Erik Blasch, and Haibin Ling. Encoding color information for visual tracking: Algorithms and benchmark. *IEEE Trans. on Image Processing*, 24(12):5630–5644, 2015.

[68] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Online object tracking: A benchmark. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[69] Matthias Mueller, Neil Smith, and Bernard Ghanem. A benchmark and simulator for uav tracking. In *ECCV*, pages 445–461. Springer, 2016.

[70] Siyi Li and Dit-Yan Yeung. Visual object tracking for unmanned aerial vehicles: A benchmark and new motion models. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[71] Lianghua Huang, Xin Zhao, and Kaiqi Huang. Got-10k: A large high-diversity benchmark for generic object tracking in the wild. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[72] Hamed Kiani Galoogahi, Ashton Fagg, Chen Huang, Deva Ramanan, and Simon Lucey. Nfs: A benchmark for higher frame rate object tracking. In *Proc. of the IEEE Int. Conf. on Comp. Vis.*, pages 1125–1134, 2017.

[73] Sangmin Oh, Anthony Hoogs, Amitha Perera, Naresh Cuntoor, Chia-Chih Chen, Jong Taek Lee, Saurajit Mukherjee, JK Aggarwal, Hyungtae Lee, Larry Davis, et al. A large-scale benchmark dataset for event recognition

in surveillance video. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3153–3160. IEEE, 2011.

[74] Heng Fan, Liting Lin, Fan Yang, Peng Chu, Ge Deng, Sijia Yu, Hexin Bai, Yong Xu, Chunyuan Liao, and Haibin Ling. Lasot: A high-quality benchmark for large-scale single object tracking. In *Procd. of the IEEE Conf. on Comp. Vis. and Pattern Recog.*, pages 5374–5383, 2019.

[75] Ashish Kumar, Gurjit Singh Walia, and Kapil Sharma. Real-time multi-cue object tracking: Benchmark. In *Proceedings of International Conference on IoT Inclusive Life (ICIIL 2019), NITTTR Chandigarh, India*, pages 317–323. Springer, 2020.

[76] Ashish Kumar, Gurjit Singh Walia, and Kapil Sharma. Recent trends in multicue visual tracking: A review. *Expert System with Applications*, 162:113711, 2020.

[77] Michael Isard and Andrew Blake. Condensation—conditional density propagation for visual tracking. *International journal of computer vision*, 29(1):5–28, 1998.

[78] Gwangmin Choe, Tianjiang Wang, Fang Liu, Suchol Hyon, and Jongwon Ha. Particle filter with spline resampling and global transition model. *IET Comp. Vis.*, 9(2):184–197, 2014.

[79] Tiancheng Li, Miodrag Bolic, and Petar M Djuric. Resampling methods for particle filtering: classification, implementation, and strategies. *IEEE Signal Processing Magazine*, 32(3):70–86, 2015.

[80] Ming-Liang Gao, Li-Li Li, Xian-Ming Sun, Li-Ju Yin, Hai-Tao Li, and Dai-Sheng Luo. Firefly algorithm based particle filter method for visual tracking. *Optik-Int. J. for Light and Elect. Optics*, 126:1705–1711, 2015.

[81] Fasheng Wang, Baowei Lin, and Xucheng Li. An ant particle filter for visual tracking. In *2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS)*, pages 417–422. IEEE, 2017.

[82] Chris Stauffer and W Eric L Grimson. Adaptive background mixture models for real-time tracking. In *IEEE CS Conf. on Computer Vision and Pattern Recognition*, volume 2, pages 246–252, 1999.

[83] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. on pattern analy. and machine intelli.*, 24(7):971–987, 2002.

[84] Anna Bosch, Andrew Zisserman, and Xavier Munoz. Representing shape with a spatial pyramid kernel. In *Procd. of the 6th ACM int. conf. on Image and video retrieval*, pages 401–408, 2007.

[85] Anil Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bull. Calcutta Math. Soc.*, 35:99–109, 1943.

[86] Sankalap Arora and Satvir Singh. Butterfly optimization algorithm: a novel approach for global optimization. *Soft Computing*, 23(3):715–734, 2019.

[87] Matej Kristan, Jiri Matas, Aleš Leonardis, Tomas Vojir, Roman Pflugfelder, Gustavo Fernandez, Georg Nebehay, Fatih Porikli, and Luka Čehovin. A novel performance evaluation methodology for single-target trackers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(11):2137–2155, Nov 2016.

[88] Xu Jia, Huchuan Lu, and Ming-Hsuan Yang. Visual tracking via adaptive structural local sparse appearance model. In *IEEE Conf. on Computer vision and pattern recognition*, pages 1822–1829, 2012.

[89] Tianzhu Zhang, Bernard Ghanem, Si Liu, and Narendra Ahuja. Robust visual tracking via multi-task sparse learning. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2042–2049. IEEE, 2012.

[90] Kaihua Zhang, Lei Zhang, and Ming-Hsuan Yang. Real-time compressive tracking. In *European conf. on computer vision*, pages 864–877. Springer, 2012.

[91] Amit Adam, Ehud Rivlin, and Ilan Shimshoni. Robust fragments-based tracking using the integral histogram. In *Comp. visi. and pattern recog., 2006 IEEE CS Conf. on*, volume 1, pages 798–805, 2006.

[92] David A Ross, Jongwoo Lim, Ruei-Sung Lin, and Ming-Hsuan Yang. Incremental learning for robust visual tracking. *Int.l J. of computer vision*, 77:125–141, 2008.

[93] Boris Babenko, Ming-Hsuan Yang, and Serge Belongie. Robust object tracking with online multiple instance learning. *IEEE transactions on pattern analysis and machine intelligence*, 33(8):1619–1632, 2011.

[94] Kaihua Zhang and Huihui Song. Real-time visual tracking via online weighted multiple instance learning. *Pattern Recog.*, 46:397–411, 2013.

[95] Laura Sevilla-Lara and Erik Learned-Miller. Distribution fields for tracking. In *2012 IEEE Conference on computer vision and pattern recognition*, pages 1910–1917. IEEE, 2012.

[96] Luca Bertinetto, Jack Valmadre, Stuart Golodetz, Ondrej Miksik, and Philip H. S. Torr. Staple: Complementary learners for real-time tracking. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[97] Ashish Kumar, Gurjit Singh Walia, and Kapil Sharma. Real-time visual tracking via multi-cue based adaptive particle filter framework. *Multimedia Tools and Applications*, 79:20639–20663, 2020.

[98] Ashish Kumar, Gurjit Singh Walia, and Kapil Sharma. A novel approach to overcome sample impoverishment problem of particle filter using chaotic

crow search algorithm. In *Proceedings of International Conference on Futuristic Technologies (ICFT-2019), GEC Panipat, India*, 2019.

[99] Guoxia Xu, Sheheryar Khan, Hu Zhu, Lixin Han, Michael K Ng, and Hong Yan. Discriminative tracking via supervised tensor learning. *Neurocomputing*, 315:33–47, 2018.

[100] Yuanhao Yu, Qingsong Wu, Thia Kirubarajan, and Yasuo Uehara. Robust discriminative tracking via structured prior regularization. *Image and Vision Computing*, 69:68–80, 2018.

[101] Tao Zhou, Fanghui Liu, Harish Bhaskar, Jie Yang, Huanlong Zhang, and Ping Cai. Online discriminative dictionary learning for robust object tracking. *Neurocomputing*, 275:1801–1812, 2018.

[102] Lingfeng Wang and Chunhong Pan. Visual object tracking via a manifold regularized discriminative dual dictionary model. *Pattern Recognition*, 91:272–280, 2019.

[103] Chaoyang Zhao, Jinqiao Wang, Guibo Zhu, Yi Wu, and Hanqing Lu. Learning weighted part models for object tracking. *Computer Vision and Image Understanding*, 143:173–182, 2016.

[104] Zhiyong Li, Song Gao, and Ke Nai. Robust object tracking based on adaptive templates matching via the fusion of multiple features. *Journal of Visual Communication and Image Representation*, 44:1–20, 2017.

[105] Berthold KP Horn and Brian G Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981.

[106] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognization, IEEE Computer Society Conf.*, volume 1, pages 886–893, 2005.

[107] Ying Bai and Dali Wang. Fundamentals of fuzzy logic control—fuzzy sets, fuzzy rules and defuzzifications. In *Advanced Fuzzy Logic Technologies in Industrial Applications*, pages 17–36. Springer, 2006.

[108] Mengmeng Ma and Jiyao An. Combination of evidence with different weighting factors: a novel probabilistic-based dissimilarity measure approach. *Journal of sensors*, 2015, 2015.

[109] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees.* CRC press, 1984.

[110] Jianming Zhang, Shugao Ma, and Stan Sclaroff. Meem: robust tracking via multiple experts using entropy minimization. In *ECCV*, pages 188–203. Springer, 2014.

[111] Matthias Mueller, Neil Smith, and Bernard Ghanem. Context-aware correlation filter tracking. In *CVPR*, volume 2, page 6, 2017.

[112] Zhipeng Zhang and Houwen Peng. Deeper and wider siamese networks for real-time visual tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4591–4600, 2019.

[113] Bo Li, Wei Wu, Qiang Wang, Fangyi Zhang, Junliang Xing, and Junjie Yan. Siamrpn++: Evolution of siamese visual tracking with very deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4282–4291, 2019.

[114] Ashish Kumar, Gurjit Singh Walia, and Kapil Sharma. Robust object tracking based on adaptive multicue feature fusion. *Journal of Electronic Imaging*, 2020.

[115] Changhong Fu, Ran Duan, and Erdal Kayacan. Visual tracking with online structural similarity-based weighted multiple instance learning. *Information Sciences*, 481:292–310, 2019.

[116] Wei Feng, Ruize Han, Qing Guo, Jianke Zhu, and Song Wang. Dynamic saliency-aware regularization for correlation filter-based object tracking. *IEEE Trans. on Image Processing*, 28(7):3232–3245, 2019.

[117] Sam Hare, Stuart Golodetz, Amir Saffari, Vibhav Vineet, Ming-Ming Cheng, Stephen L Hicks, and Philip HS Torr. Struck: Structured output tracking with kernels. *IEEE Tran. Patt. Anal. Mach. Intelli.*, 38(10):2096–2109, 2016.

[118] Alireza Asvadi, Hami Mahdavinataj, Mohammad Reza Karami, and Yassar Baleghi. Online visual object tracking using incremental discriminative color learning. *The CSI Journal on Comp. Sc. and Engg.*, 2014.

[119] Chenglong Bao, Yi Wu, Haibin Ling, and Hui Ji. Real time robust l1 tracker using accelerated proximal gradient approach. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1830–1837. IEEE, 2012.

[120] Martin Danelljan, Gustav Häger, Fahad Shahbaz Khan, and Michael Felsberg. Discriminative scale space tracking. *IEEE transactions on pattern analysis and machine intelligence*, 39(8):1561–1575, 2017.

[121] Bolun Cai, Xiangmin Xu, Xiaofen Xing, Kui Jia, Jie Miao, and Dacheng Tao. Bit: Biologically inspired tracker. *IEEE Transactions on Image Processing*, 25(3):1327–1339, 2016.

[122] Ashish Kumar, Gurjit Singh Walia, and Kapil Sharma. A novel approach for multi-cue feature fusion for robust object tracking. *Applied Intelligence*, 50:3201–3218, 2020.

Appendix

Biodata

Ashish Kumar was born on 22nd May 1986 in Delhi, India. He received his B.E. degree (with HONS) in Electronics and Communication Engineering from Vaish College of Engineering, M.D. University, Rohtak (Haryana) in 2007. He received his M.Tech degree (with Distinction) in Computer Science and Engineering from University School of Information Technology, Guru Gobind Singh Indraprastha University, New Delhi in 2009. Presently, he holds the position of Assistant Professor in Department of Electronics and Communication Engineering at Bharati Vidyapeeth's College of Engineering since 2009. He joined Delhi Technological University, New Delhi as part time Ph.D Scholar in Computer Science and Engineering Department under the supervision of Prof. Kapil Sharma and Dr. Gurjit Singh Walia in 2016. His current research focuses on object tracking, video processing and machine learning. He works on multi-cue object tracking solutions and has proposed various robust tracking frameworks.