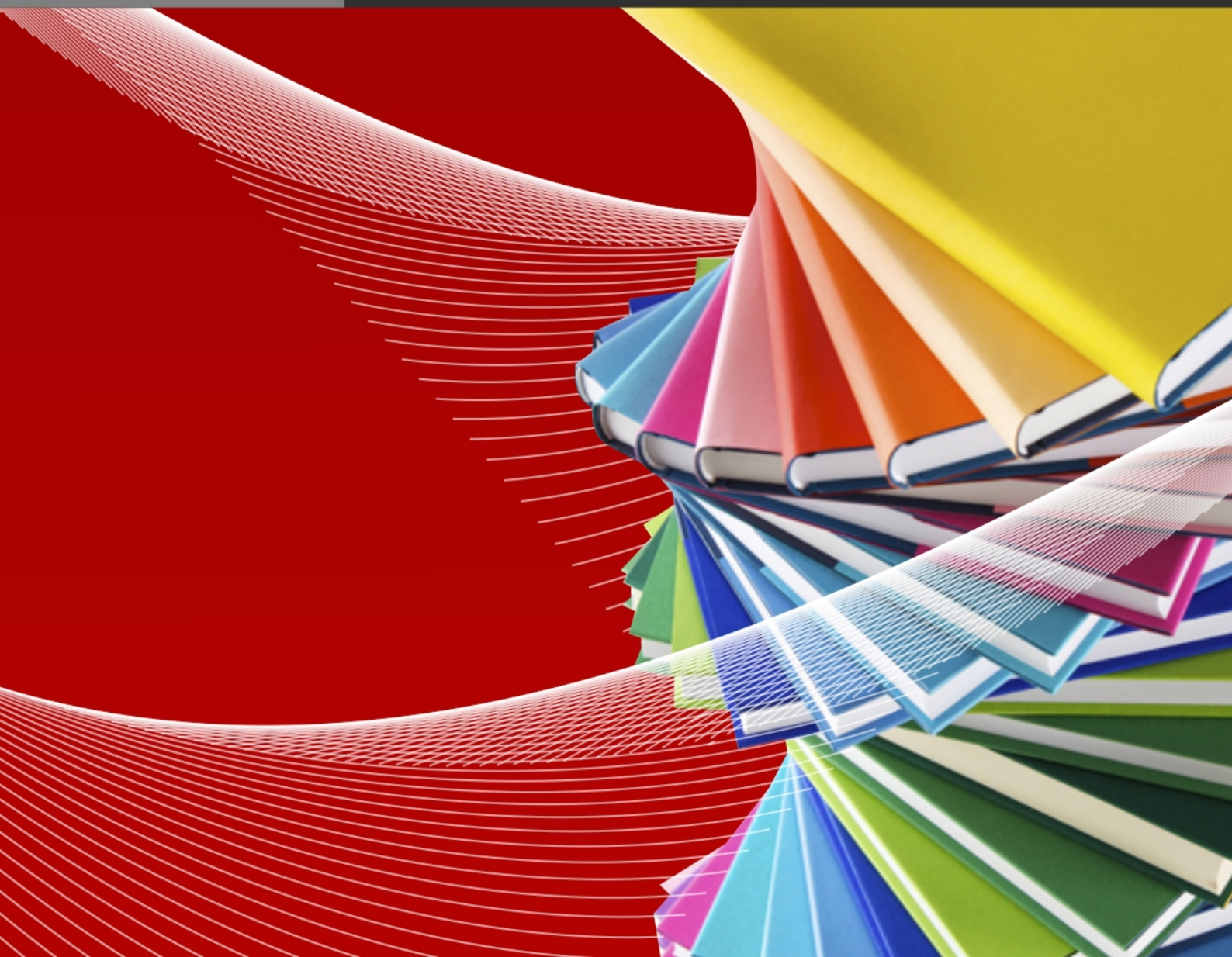
澳門大學
UNIVERSIDADE DE MACAU
UNIVERSITY OF MACAU

Outstanding Academic Papers by Students
學生優秀作品



Comparison of Visual Tracking Algorithms and Realization of Online Tracking System

by

ZHOU YINGSI

Final Year Project Report submitted in partial fulfillment
of the requirements for the Degree of

Bachelor of Science in Electrical and Computer Engineering

2015

Faculty of Science and Technology
University of Macau

********* Bachelor's Thesis Quote (OPTIONAL) **********

Bachelor's Thesis (or Final Report of ECEB420 Design Project II)

In presenting this Final Report of Design Project II (ECEB420) in partial fulfillment of the requirements for a Bachelor's Degree at the University of Macau, I agree that the **UM Library** and **Faculty of Science and Technology (FST)** shall make its copies available strictly for internal circulation or inspection. No part of this thesis can be reproduced by any means (electronic, mechanical, visual, and etc.) before the valid date (usually less than 3 years) limit listed below. Copying of this thesis before the valid date from other parties is allowable **only** under the prior written permission of the author(s).

Printed name: __Zhou Yingsi_____

Signature: _____

Student number: _____

Date: _____

Reliable Contact information (address, tel. no., email, etc.) of author:

_____

_____

_____

_____

Valid date until _____

********* End of Bachelor's Thesis Quote **********

DECLARATION

I declare that the project report here submitted is original except for the source materials explicitly acknowledged and that this report as a whole, or any part of this report has not been previously and concurrently submitted for any other degree or award at the University of Macau or other institutions.

I also acknowledge that I am aware of the Rules on Handling Student Academic Dishonesty and the Regulations of the Student Discipline of the University of Macau.

Signature : _____ (e-signature OK)

Name : _____

Student ID : _____

Date : _____

APPROVAL FOR SUBMISSION

This project report entitled "**Comparison of Visual Tracking Algorithms and Realization of Online Tracking System**" was prepared by Yingsi Zhou (DB12932) in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Computer Engineering at the University of Macau.

Endorsed by,

Signature : _____

Supervisor : Dr. Junda Zhu

ACKNOWLEDGEMENTS

I would like to express my gratitude and appreciation to my final year project supervisors Dr. Junda Zhu who gave me the wonderful opportunity to complete this project and gave me suggestion and guidance in project management and presentation skills. A special thanks to a graduate student, Mr. Ze Wang whose help, stimulating suggestions and encouragement, helped me in Matlab programming. I would also like to acknowledge with much appreciation the crucial role of a graduate student Mr. Yajun Huang of data center research Laboratory by his comment and tips.

ABSTRACT

Visual tracking is one of the essential application in computer vision. Many visual tracking algorithms have been designed and realized in practical application. By employing these visual tracking algorithms, a lot of trackers can be used in target tracking in different conditions. In order to utilize these tracker more efficiently, we need to estimate the performance of them and apply them into proper environment respectively. Due to the various characteristics and principle of these algorithms, their performances in robustness evaluation should be analyzed specially for test sequences with different attributes. This project brief introduces and estimates the performance of the following nine trackers: CSK (Circulant Structure of Tracking-by-Detection with Kernels), CT（Real-time Compressive Tracking）, CXT (Context Tracker), MTT (Multi-Task Sparse Learning), L1APG (L1 Tracker Using Accelerated Proximal Gradient Approach), DFT (Distribution Fields for Tracking), IVT (Incremental Learning for Robust Visual Tracking), SCM (Robust Object Tracking via Sparsity-based Collaborative Model), LOT(Locally Orderless Tracking) with categorizing them as generative or discriminative trackers. The robustness test includes one-pass evaluation (OPE), temporal robustness evaluation (TRE) and spatial robustness evaluation (SRE), which estimate the trackers' performance comprehensively. The evaluation on tracking speed is also given, so as to apply the high speed trackers into online tracking system. Ultimately, an online tracking system is developed with three different trackers, to cater the need of internal and external tracking attributes.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

In the field of computer vision, a lot of work have been done to the design and improvement in the tracking method. In actual applications, target tracking is a very challenging problem, because a lot of factors can interfere the detection for object. These factors can be external or internal. Some trackers are robust to these condition changes, while some others are not, since their mathematical principles and working principles are different, the ability to cope with disturbance will be different respectively. In order to take full use of the merits in every trackers, we need to find out their most fitting application conditions.

Visual tracking is searching the position of a moving target which we are interested in from a video or image sequences and the same target would be corresponded within different frames. The research and application of visual tracking algorithms is a useful technology in many fields, such as video monitoring system, human body detection, automobile navigation and underwater exploration. The actual scene in visual tracking system tend to be complex and capricious. Deformation, illumination variation, blur, fast motion and background clutter are challenges during tracking the object and may cause missing of targets. While much progress of research and realization of many visual tracking algorithms has been attained in recent years, a benchmark is employed to estimate their performance in target tracking, especially in the cases that external or internal condition changes. Since the visual trackers are working in different principles and having various features, their strengths are not possible to be demonstrated in every conditions and environments. In this project, the tracking accuracy evaluation of visual trackers is analyzed by test sequences with nine attributes, which is illumination variation, scale variation, occlusion, deformation, motion blur, fast motion, in-plane rotation, out-of-plane rotation, background clutters. I did the attribute analysis in this project aiming at find out the relatively better operation occasion for every tracker. Moreover, tracking processes includes object representation, searching

mechanism and model update, the analysis of trackers' performance will be also based on these processes.

On the strength of the machine learning principle, tracking algorithms can be divided into two categories. One is generative tracker, the other one is discriminative tracker. Generative tracker uses the appearance model to judge the objects' position, while the discriminative tracker makes decision of the samples and labels them into positive samples and negative samples. In this project, we will use the benchmark to evaluate and analyze the performance of this two sorts of algorithms.

We will analyze nine trackers: CSK (Circulant Structure of Tracking-by-Detection with Kernels), CT（Real-time Compressive Tracking）, CXT (Context Tracker), MTT (Multi-Task Sparse Learning), L1APG (L1 Tracker Using Accelerated Proximal Gradient Approach), DFT (Distribution Fields for Tracking), IVT (Incremental Learning for Robust Visual Tracking), SCM (Robust Object Tracking via Sparsity-based Collaborative Model), LOT(Locally Orderless Tracking). Among these nine trackers, MTT , L1APG,IVT , SCM are generative trackers, CT, CSK , CXT , DFT , LOT are discriminative trackers.

Generative tracker is generated the model in the aspect of the data distribution, even though the calculation and learning process is complex, the strategy can reflect the similarity of data in the same category. This kind of tracker can train out a representative appearance model by inputting an amount of samples, and then determine the location of the object by checking region with highest similarity to the model in that image, thereby, to realize visual tracking. However, the generative trackers only consider the target's appearance instead of decision bound, errors in judgement are easily to be made. Discriminative trackers take the context around targets in to consideration and train the classifier. It can efficiently reflect the discrepancy between samples. The learning process aims to find out a decision bound to separate the object and the context, which makes the discriminative tracker more robust when scale variation, occlusion and

rotation or other interference happen on the target. But the discriminative trackers are not able to give precise description to the target's appearance, which is a drawback in some applications.

In this project, one-pass evaluation (OPE), temporal robustness evaluation (TRE) and spatial robustness evaluation (SRE) would be proceeded on the tracker's performance evaluation in different attributes, which represents the challenging aspects in visual tracking.

I estimated not only the correct rate of the visual trackers, but also the speed of tracking in this project. Because, fast speed tracking is very important in online system, if the tracker is operated slowly, frame loss would be caused during tracking, significantly influence the accuracy. The analysis on tracking speed is aim at finding out the high-speed trackers and apply them into online tracking system.

Finally, I will demonstrate an online tracking system equipped with three different trackers that are with high-speed and strong robustness after referring the results from evaluation in the correct rates, robustness and tracking speed of all the trackers. The online tracking system I developed is a combination of tracking system in three fast and robust trackers, to satisfy the requirement of various environments.

CHAPTER 2 INTRODUCTION OF ALGORITHMS

In target tracking, we need 2D appearance models for describing and searching the tracking objects. Tracking algorithms can be generally categorized as either generative or discriminative based on their appearance models. They label the sample in different approach. Both of them can identify the most likely labels and their likelihoods.

Generative model is judging the sample by feature matching. It describe how the hidden labels "generated" the observed input as joint probabilities, represented as P(class, data). Generative tracking algorithms typically learn a model to represent the target object and then use it to search for the image region with minimal reconstruction error (Zhang, 2012). The generative algorithms do not use the background information which is likely to improve tracking stability and accuracy.

Discriminative model is to classify the samples into different types. It only predict or discriminate the hidden labels conditioned on the observed input, represented as P(class| data). Discriminative algorithms pose the tracking problem as a binary classification task in order to find the decision boundary for separating the target object from the background. These models need learning approach developed in selecting positive and negative samples via an online classifier.

2.1 GENERATIVE MODELS

2.1.1 MULTI-TASK SPARSE LEARNING

This algorithm Structured Multi-Task Tracking to formulate object tracking in a filter framework as a structured multi-task sparse learning problem. Since we model particles as linear combinations of dictionary templates that are updated dynamically, Multi-Task

Tracking (MTT) is employed in learning the representation of each particle (Zhang, 2012).

Multitasking sparse learning method is proposed in this algorithm to explore the global and local structure between different tasks. Principle of sparse representation is following a linear expression: $y = Ax$, where $A \in R^{m \times n}$ $(m \ll n)$, $x \in R^n$, and $y \in R^m$. That means when a completed set A in m dimension space is given, sparse representation can choose least number of x to reconstruct the vector y. In detecting, we have k type models in total. Every sample in each type is described as a vector array as aij. If type number i includes ni samples, then we have

$$A_i = \left[a_{i1}, \quad a_{i2}, \quad \cdots, \quad a_{in_i}\right] \in R^{m \times n_i}$$

$$A = \left[A_1, \quad A_2, \quad \cdots, \quad A_k\right]$$

If y belongs to type number i

$$y = x_{i1}a_{i1} + x_{i2}a_{i2} + \cdots + x_{in_i}a_{in_i}$$

Ideally, x will be a matrix as

$$x = \left[0, \quad \cdots, \quad 0, \quad x_{i1}, \quad x_{i2}, \quad \cdots, \quad x_{in_i}, \quad 0, \quad \cdots, \quad 0\right]^{\mathrm{T}} \in R^n$$

A and y is given, the result x will be figured out, namely, we are judging the classification of y in A according to the type of x.

As for MTT and SMTT, MTT is the spatial representation of different particles. The S-MTT formulation can be viewed as a generalization of MTT, since local structural information endows MTT with another layer of robustness in tracking. The S-MTT objective is composed of a convex quadratic term and a non-smooth regularizer, and thus we conventionally adopt the APG method for optimization.

2.1.2 TRACKER USING ACCELERATED PROXIMAL GRADIENT APPROACH

The same as MTT, L1 Tracker Using Accelerated Proximal Gradient Approach also use the idea of sparse representation. It model the target appearance using a sparse

approximation over a template set. This algorithm is closely related to the L1 tracker. The main differences lie in a different minimization model and a much faster numerical solver for the resulting Ll norm minimization problems (Mei, 2011). Besides sparse representation, this algorithm also employed the particle filter. The particle filter provides an estimate of posterior distribution of random variables. It consists of two steps: prediction and update, which gives an important tool for estimating the target of next frame without knowing the concrete observation probability.

The tracking process is guided by the particle filtering (Bao, 2012). We set the true target location of the initial frame. In the current frame, tracking results are close to the previous frame, are obtained according to the candidate target sampling by the particle filter. We assumed that trivial templates are included in the template dictionary. Therefore, in all possible target, our choice is those with minimal error bound resampling in the sparse reconstruction. In the process of target moving, because of the influence of internal transformation (rotation, greyscale) and external factors (shade and illumination change), target appearance may change. In order to adapt to this change, we need to change the appearance of the tracking target in time. L1 tracker replaces the outdated items in the dictionary by using the current tracking results.

2.1.3 INCREMENTAL LEARNING FOR ROBUST VISUAL TRACKING

Incremental learning for robust visual tracking adapts to the change of target facade in practical visual tracking availably, it is a kind of incremental algorithms that represents low-dimensional subspace by increasingly learning.

The update of appearance model in this tracking algorithm is based on principal component analysis, before obtaining the current frame, several frames from the tracking results can build an image space, operating PCA to the image space, then the mean and eigenvector of history tracking results can be output. PCA is used for multivariate statistical analysis, it can use less number of features to describe samples, reducing feature space dimension. The essence of it is Karhunen-Loeve Transform (K-

L Transform). Karhunen-Loeve Transform means optimal orthogonal transformation. It is a common feature extraction method and minimum mean square error of the optimal orthogonal transformation. K-L transformation have optimal effect on highlighting the differences between different pattern characteristics.

2.1.4 ROBUST OBJECT TRACKING VIA SPARSITY-BASED COLLABORATIVE MODEL

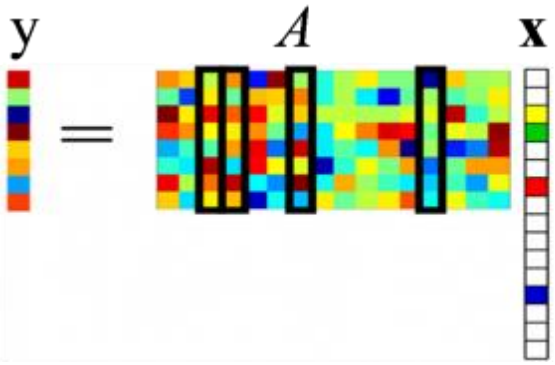


Figure 2.1.4 Sparse representation

A sparsity-based discriminative classifier (SDC) and a sparsity-based generative model (SGM) are developed in this tracker with sparsity-based collaborative model, and the appearance model employs both holistic templates and local representations. The holistic templates use the intensity in every frame to generate holistic templates, which is serving in target detection and solve the problem of target deformation. As for the local histogram, it change the tracking problem as a mapping problem. The function is s=T(r), whereas r is the pixel gray scale in original image, and s is the pixel gray scale in a new frame.

The sparsity-based comparison model adopt sparse representation idea with a linear expression: $y = Ax$, whereas $A \in R^{m \times n}$ $m \ll n$, $x \in R^n$ $y \in R^m$

As Figure 2.1.4 shows, given a completed set A in m dimension space， choose least number of x，we can reconstruct the vector y, this is the key ides of sparse representation. In visual tracking, assumed that we have k types of sample in total, each sample in every

type represent as aij with a column vector in A set. If type number i includes ni samples，then

$$A_i = \left[a_{i1}, \quad a_{i2}, \quad \cdots, \quad a_{in_i}\right] \in R^{m \times n_i}$$
$$A = \left[A_1, \quad A_2, \quad \cdots, \quad A_k\right]$$

If y belongs to type number i:

$$y = x_{i1}a_{i1} + x_{i2}a_{i2} + \cdots + x_{in_i}a_{in_i}$$

Therefore, the tracking problem become finding the projection of category classifier y in A according to the value of non-zero x, with A and y are known.

2.2 DISCRIMINATIVE MODELS

2.2.1 CONTEXT TRACKER

The reason for raising up this algorithm and the problem which is solving for discrimination of similar appearance of the target, including the condition of changes in appearance, varying lighting conditions, cluttered background and frame-cuts. It is even more challenging when the target leaves the field of view (FoV) leading the tracker to follow another similar object, and not reacquire the right target when it reappears (Dinh, 2011).

The two main point of this algorithm is distracters and supporter. The first one is detected by randomized ferns classifier, which are regions that have similar appearance as the target, while the latter one is consist of Hessian Detector and surf descriptor. Supporters are local key-points around the object having motion correlation with target in a short time span.

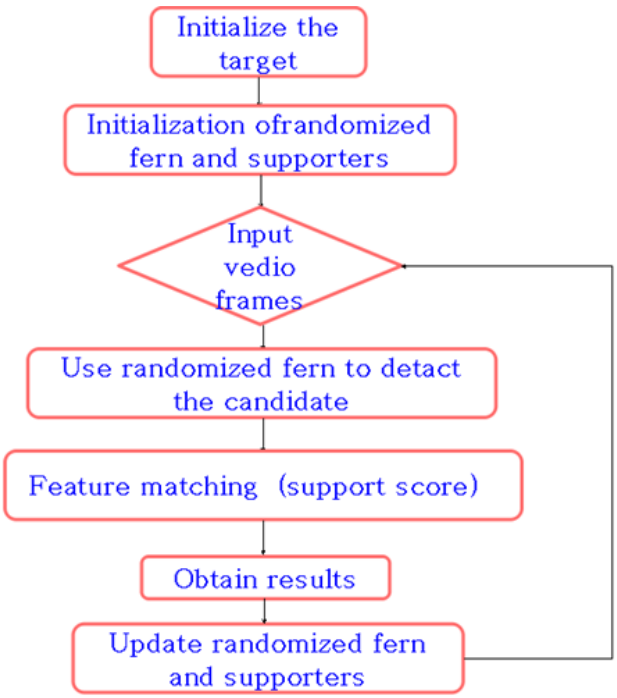


Figure 2.2.1 Flow chart of Context tracker

Context tracker is a P-N Tracker, as the tracking-learning-detection concept is adopted, in order to label the positive and negative structures. It uses scanning window to search for all of possible candidates in the whole image which helps to explore the context at the same time. P-N Tracker purely relies on template matching to find the best match among several candidates. It is vulnerable to switching to another similar object.

As for their working procedures, firstly initialization of the target is necessary. Then, we also need to initialize the distracters and supporters. Next, input video frames to begin tracking. After the detection of candidates by randomized fern and feature matching, we can obtain the result. Whereas, for the accuracy of the next detecting, the result attained in the previous step should be used to update the randomized fern and supporters.

2.2.2 REAL-TIME COMPRESSIVE TRACKING

This is an algorithm based on compressive sensing, with dimensionality reduction to image feature in very sparse measurement matrix，then the dimensionality reducted feature will be discriminated by a simple naive Bayes classifier. The tracking algorithm

is very simple, but the result of the experiment is very robust, the speed can reach about 40 frames per second.

The frame work of its classification method is general, to extract the image feature firstly and use the classifier to discriminate the samples. The difference is that CT tracker use compressed sensing for feature extraction, and use naïve Bayes classification to label the sample, and the classifier is updated by online learning

As for the principle of compressive sensing, it uses a very sparse measurement matrix to facilitate efficient projection from the image feature space to a low-dimensional compressed subspace. The low-dimension signal from dimensionality reduction can keep the characteristics of high-dimension ones. Moreover, the signal can be reconstructed if condition of restricted isometry property is satisfied.

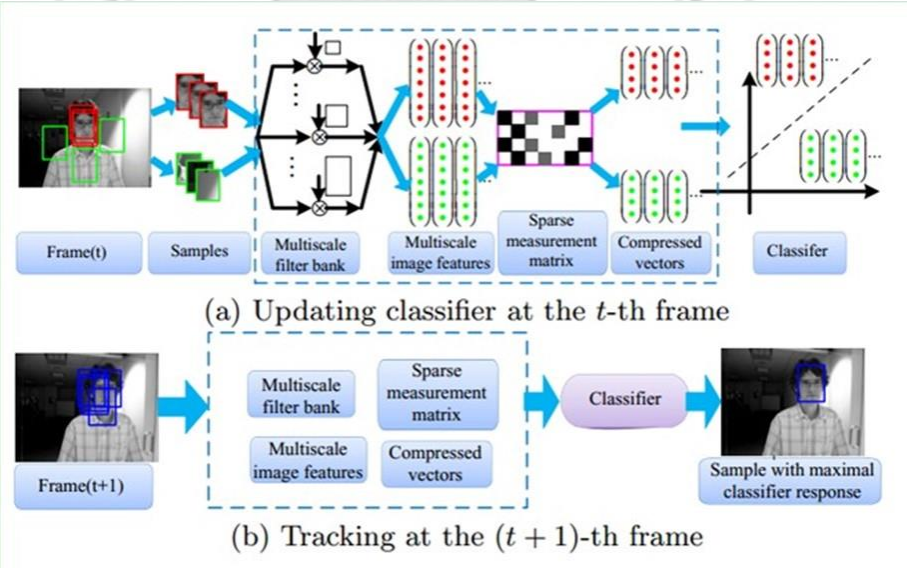


Figure 2.2.2 main components of our compressive tracking algorithm

Main idea in tracking (Zhang, 2012):

1. In t–th frame sampling and get object(positive) and background(negative) model，after multi-scale transformation and dimensionality reduction with sparse measurement matrix, naive Bayes classifier can be training.

2. In t+1-th frame，sample n bounding box around the object position in last frame (avoid the whole picture), and do dimensionality reduction from sparse measurement matrix fetch its feature, then use the trained naive Bayes classifier in t–th frame to make classification. Thus, tracking from t frame to t+1 done.

2.2.3 CIRCULANT STRUCTURE OF TRACKING-BY-DETECTION WITH KERNELS

In visual tracking, the potentially number of samples is large, which becomes a computational burden. And also become an obstacle in building a real time system. Using the well-established theory of circulant matrices, we provide a link to Fourier analysis that opens up the possibility of extremely fast learning and detection with the Fast Fourier Transform [Hare, 2011]. This can be done in the dual space of kernel machines as fast as with linear classifiers. We derive closed-form solutions for training and detection with several types of kernels, including the popular Gaussian and polynomial kernels.

The essential component in tracking-by-detection is a classifier. The function of the classifier is to label the samples as positive or negative. However, with the restriction of calculation, only handful of random samples are collected. This algorithm opt to train a classifier with all samples: dense sampling. It contributes to an efficient training. For efficient learning, CSK used circulant matrices. Circulant matrices is an n×n matrix whose rows are composed of cyclically shifted versions of a length-n list l. They encode the convolution of vectors, which is conceptually close to what we do when evaluating a classifier at many different sub-windows.

2.2.4 LOCALLY ORDERLESS TRACKING

Locally Orderless Tracking (LOT) is an algorithm that involve in automatic operating the evaluation in the sum total of local order or disorder in the object in visual tracking. This feature make the tracking focus on online rigid and deformable objects without prior assumption. The tracker have provided an object probability model changing over

time. The model uses the two parameters in earth mover's distance (EMD) to control the cost of moving pixels, and change its color. We adjust the cost of online tracking process according to the proportion of number of local order or local disorder.

In visual tracking, tracking object types often need to do an explicit or implicit assumptions, in order to judge whether we should consider it as a rigid deformable objects or non-rigid object. For example, if you want to keep tracking of a rigid object, the only possible thing that will be transferred in the appearance of this rigid object is caused by the mathematical geometry transformation, then, when the pixel point is fixed, and controlled by the geometric transformation, to simplify the similarities to pixel intensity difference per-pixel by using methods such as template matching should be justified. On the other hand, if the object is strictly a deformable objects, then histogram matching tracking based on color difference may be more suitable for reduces similar candidate on the color distribution.

The locally orderless tracking technology adopts a joint space, which can estimates the number of local order or partial disorder in the online target. Therefore, if the target is rigid, and there are only a few sample locally, the LOT tracking algorithm will save space information as template matching. However, if the target is non-rigid, then this tracking algorithm (LOT) will ignore the spatial information, as same as the histogram matching.

2.2.5 Distribution Fields for Tracking

This algorithm realize an image representing methods by using a template to represent the object. This template consists of the intensity values, gradient information, or other features. But limitations also exist, overly sensitive to the spatial structure of the object, so changes in appearance will cause error. Luckily, robust metrics alleviate this problem, but performance would decay in long sequences.

Based on the fact that objective function might not be smooth enough to reach the global optimum. Generally, the function is smoothed by blurring the image. This process can employ Gaussian pyramid or alternative blur kernels.

Blurring the image may destroy the image information in traditional way, but in the DF framework, the layered, or channel-by-channel blurring technique allows smoothing the objective function without loss of information that occurs in traditional blurring. When an image is blurred, the new pixel values are a combination of the neighboring pixels around them. The comparison is demonstrated in the following figure.

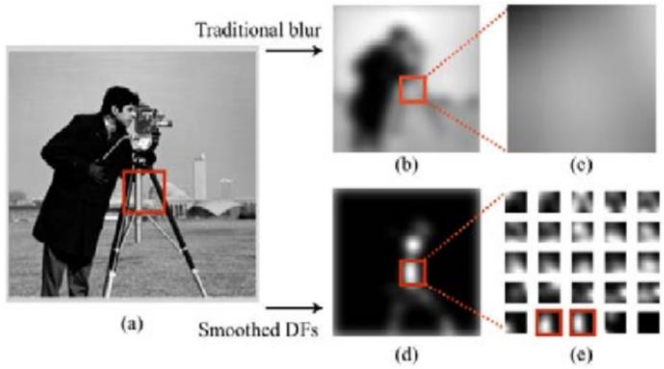


Figure 2.2.5 Comparison between traditional blur and smoothed DFs

Distribution Fields (DFs) composed of a set of probability distribution of image characteristic. First of all, it using Kronecker Delta function to extend the image into DFs:

$$f(i,j,k) = \begin{cases} 1, if I(i,j) == k \\ 0, otherwise \end{cases}$$

Whereas, i, j is the image pixel coordinate; K is the parameter of image characteristics, different k values represent different field or layer. For gray image, the image size is mXn (m and η refer to rows and columns of the image respectively). Take the gray value of the image characteristics, a three-dimensional distribution can be produced with a size of mXnXb, b is the number of values in the image grey scale.

CHAPTER 3 TRACKER PERFORMANCE EVALUATION

In this part, the comparison of nine algorithms are made, which are specified as MTT, L1APG, IVT, SCM, CT, CSK, CXT, DFT, LOT, and 22 test sequences are employed to test all the trackers. These test frames are carrying with the ground truth information of the tracked object. When processing visual tracking, a bounding box will be served as the marking for object that we are interesting in being tracing. However, due to the different robustness of various trackers, the bounding boxes are unable to overlap with the ground truth ones without error. The bounding box may covers only part of the ground truth object point. The judgement for whether the object is successfully tracked is the coverage of overlapping in bounding box area and ground truth region is no less than a certain threshold. The threshold we set in this project is form 0 to 1. To measure the strength and weakness of the tracking result, we should set up a benchmark, in this project, I use successful grate as the referencing measurement. The successful rate is define as the ratio of the number of successful frames and total frames. Firstly, we will estimate the successful rates of these trackers in one-pass evaluation. The successful rate is changing with threshold. If the successful rate of a tracker drops very fast with the increase in threshold, we can say that the performance of this tracking result is fairly bad. Moreover, as we have categorize these nine trackers as generative trackers and discriminative trackers, we will also evaluate their performances with the consideration on the features of generative models and discriminative models.

In visual tracking allocation, there are many challenges that are likely to influent the accuracy, i.e., change in background, object movement and similar objects interference. In order to make our word more adaptive to practical cases, we are analyze the successful rate of all the trackers under different attributes, which are Background Clutters (BC),Deformation (DEF), Fast Motion case (FM), In-Plane Rotation (IPR), Out-of-Plane Rotation (OPR), Illumination Variation (IV), Motion Blur (MB) Occlusion (OCC) and Scale Variation (SV). Also, our test sequences have been tagged

with this nine attributes, in order to represents the challenging aspects in actual cases of visual tracking. Every attributes are estimate in the integration result of 10 test sequences.

3.1 ONE-PASS EVALUATION

3.1.1 BACKGROUND CLUTTERS CASE (BC)

In this case, the attributes of the test sequence is background clutters, which defines as the background surrounding the target object has similar texture or color as the target. This is a case that may cause misjudgment of our target.

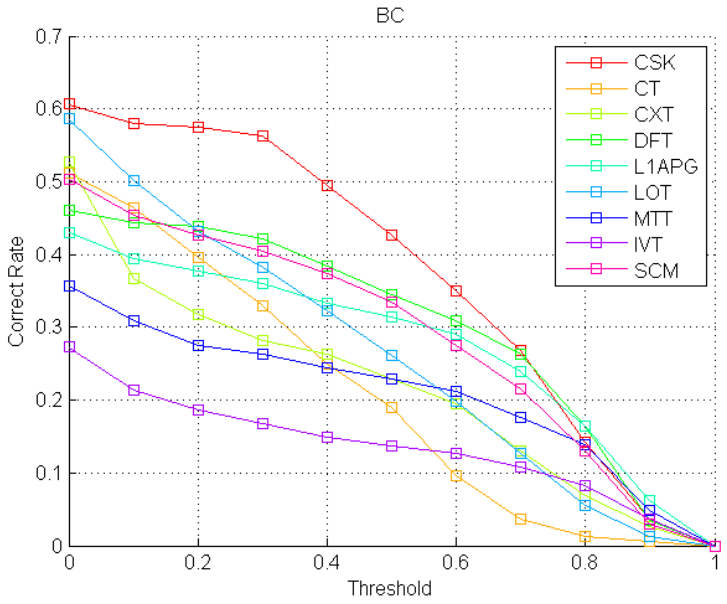


Figure 3.1.1.1 OPE success rate between nine trackers in BC case

Figure 3.1.1.1 shows the OPE success rate between nine trackers. In this case, CSK tracker (Circulant Structure of Tracking-by-Detection with Kernels) performs best, its curve have not dropped very fast with threshold.

The reason can be drawn from the way of its tracking principle. The function of the tracker can be concluded as a decision maker, to determine whether the object we catch is our target or not. This decision making relies on a decision function, the decision

function of CSK is a function of structural risk minimization. Then the tracking problem is transformed as a result mapping question in a function as:

$$\min_{\mathbf{w},b} \sum_{i=1}^{m} L\left(y_i,\ f(\mathbf{x}_i)\right) + \lambda \|\mathbf{w}\|^2 \tag{1}$$

In this function, the first item is a loss function, f(x) is the decision function we need. The last item is a structured penalty factor, which adopt to the Regularized Least Squares(RLS) with Kernels. The result of this function will be :

$$f^*(\mathbf{x}) = \sum_{i=1}^{\ell} c_i K(\mathbf{x}, \mathbf{x}_i). \tag{2}$$

In its coding, we find the processing principle. Firstly, the test frames are loaded, and we get the ground truth information, which is the position and scale information of the object, then we get a distribution function of the target object, in this algorithm, the distribution function is Gaussian distribution.

Then, the first test frame will be read by the program and converted to grayscale. We will do the filtering processing to the data and get a result with less fringe effect. Next, we get the kernel function from the previous information. The parameter c in function 2 can be calculated by this kernel. For each subsequent frames, they are all transform into gray scale image, and use hann window to process the data. The tracker will calculate the kernel again with combination of the next frame's image information. And now the value of c_i and Kernel can figure out the function response, choose the location with maximum response value. Finally, according to the location information with maximum response value, the tracker will be able to update the kernel and make decision for the next frame.

Thus, the reason that CSK performs the best in background clutters is that the tracker is reading the test sequence after gray scale processing. When the background information is complex, we can make the regional characteristics of the target much more clear through image gray processing, and to facilitate subsequent process more efficiently.

3.1.2 DEFORMATION AND OCCLUSION CASE (DEF and OCC)

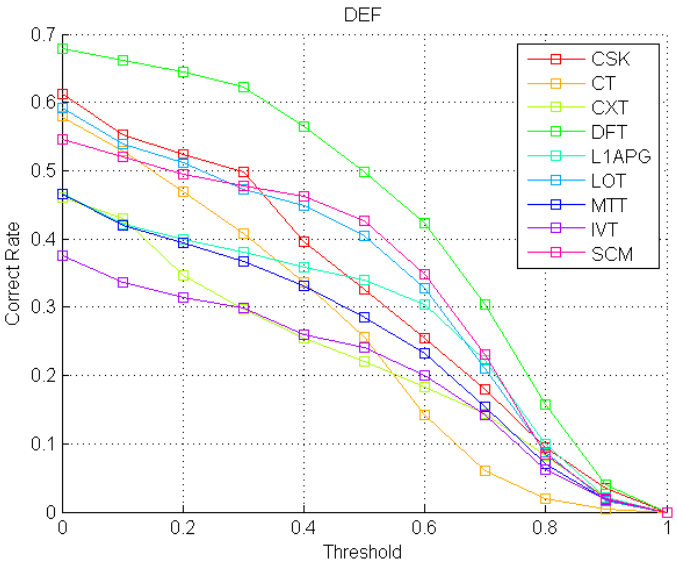


Figure 3.1.2.1 The OPE success rate between nine trackers in DEF case

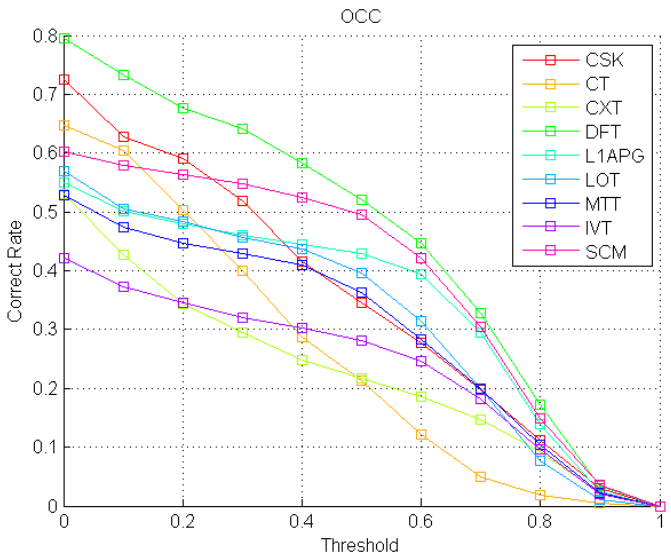


Figure 3.1.2.2 The OPE success rate between nine trackers in OCC case

This attributes represents that the test sequence includes non-rigid object deformation. Occlusion case in under the condition that the target is partially or fully occluded (Zhang, 2007).

To analyze this tracking problem, we give a definition to rigid object tracking and non-rigid object tracking firstly. According to the structural properties of the tracked target,

tracking target can be divided into rigid and non-rigid. Rigid object refers to the object with rigid structure, which is not easy to be deformed, such as vehicles; Non-rigid objects usually refer to easy-deformed objects, such as fabric, clothing surface. As for non-rigid target tracking include distortion and self-occlude phenomenon, tracking algorithms based on rigid target cannot be directly applied into non-rigid target tracking, which is more difficult and challenging. Occlusion is often include in non-rigid object tracking.

From Table 2.1.1.2, we can find that DFT performs the best over the other trackers. DFT is the short form of Distribution Fields for Tracking. Image descriptor: distribution fields DFs allow the representation of uncertainty in the descriptor, Small and non-aligned object occlusion are expressed as unlikely events and distinguished with impossible event, in order to reduce the influence of sensitivity.

On the other hand, CSK is using another image descriptor called Histogram of oriented gradient (Hog).

In conditions of cursory airspace sampling, refined direction sampling and the strong local optical normalization, as long as a pedestrians can keep upright posture, subtle body movements can be ignored and will not affect the detection effect. Therefore HOG feature is particularly suited to do human body tracking. But some objects of our test sequence are not human body, or accurately, not a linear model, moreover, and changes in the object are not subtle movement, so CSK did not perform better than DFT in this case.

So another conclusion can be made, in the aspect of generalizability in application, image descriptor DF out-performs Hog.

3.1.3 ROTATION CASE (IPR AND OPR)

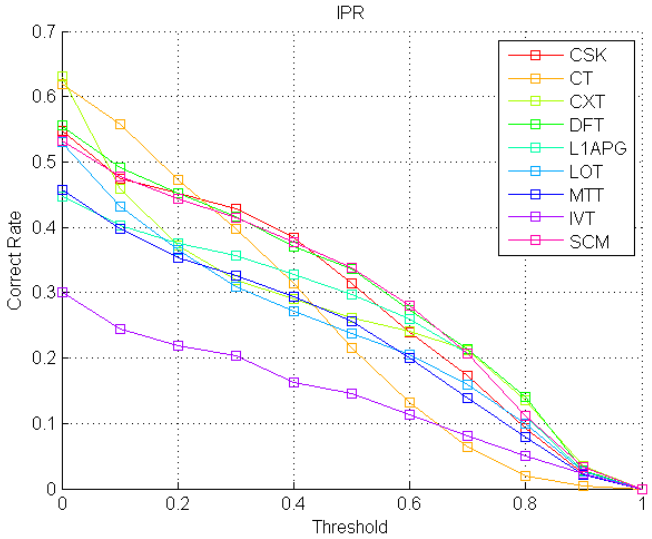


Figure 3.1.3.1 The OPE success rate between nine trackers in IPR case

There are two kind of rotational attributes, the first one is In-Plane Rotation (IPR), it refers to the rotational movement of the target in its image plane. The second one is Out-of-Plane Rotation (OPR), respectively, rotation of target object will be out of the image plane. The correct rate of all the trackers in IPR and OPR are demonstrated as Figure 3.1.3.1 and Figure 3.1.3.2.

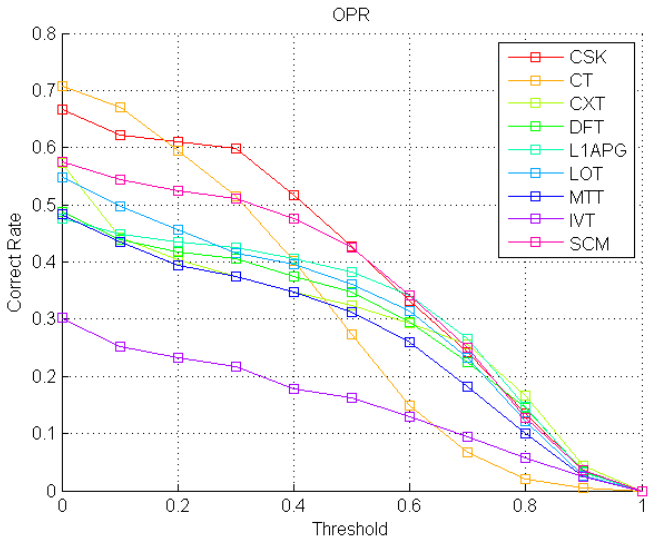


Figure 3.1.3.2 The OPE success rate between nine trackers in OPR case

We can find that CSK performs obviously better in out of plane target rotation. And actually, CSK is in good condition in two rotation attributes. Because CSK can not only use Hog as the image descriptor but also Scale-invariant feature transform (SIFT). It has the characteristic of local image feature description and detect, which can help to identify objects. SIFT is based on some interested local points the in object appearance and has nothing to do with the size and rotation of the target. Its tolerance for the light, noise, small angle change is quite high. Based on these features, it is highly significant and relatively easy to capture the characteristics of the object. If the database is large enough, it will be easy to recognize objects and very few mistakes may be made. Using SIFT feature description for rotational object detection has a high successful rate. Only three or more SIFT feature objects are enough to calculate the position and orientation. In today's computer hardware condition, with such a high speed and small feature database, SIFT tracking can be close to real-time operation. SIFT consist of a large amount of information, it is suitable for fast and exact mapping in the database.

The core of SIFT is searching the significant feature point in spaces with different scale and calculating their orientation (Lowe, 1999). The significant point SIFT searched are some very outstanding points that will not be influenced by light, affine transformation and noise factors, such as corner, edge points, light spot in dark area and the dark spots of bright background, etc.

3.1.4 SCALE VARIATION (SV)

The Scale Variation attribute refers to the tracking cases that the ratio between the bounding boxes of the first frame and the current frame is out of the range , whereas [1/ts, ts], ts > 1 (ts=2). In this condition, SCM keep a good accuracy under different threshold. As for CT, even though it correct rate is so high when threshold is 0, however, the accuracy drops too fast when we change the threshold.

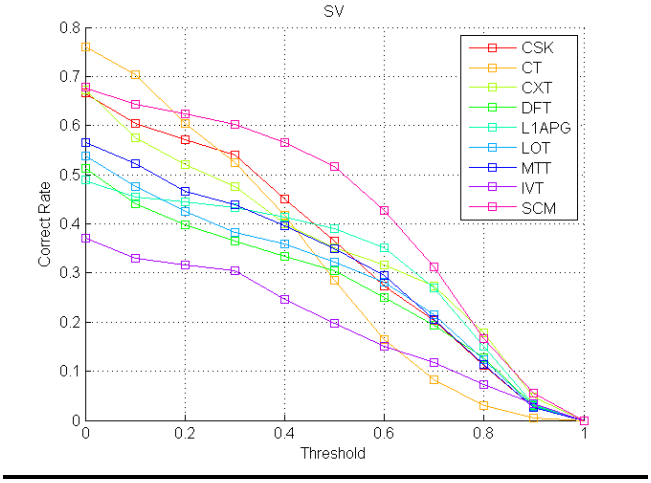


Figure 3.1.4 The OPE success rate between nine trackers in SV case

The succeed of SCM can be attributed to the merit of sparse representation. The most important idea of sparse representation is that a class of objects can be broadly represented by the training samples of the same kind in linear subspace if the training sample space is large enough. Therefore, when the object is represent by the whole sample space, its coefficient is sparse. This is one of the most important assumption in sparse representation, and is also the basis of further analysis. Through the general description of sparse representation, which can be abstracted as an equations: y=Ax，whereas y is the target samples, A is the training sample space. Sparse, of course, refers to the coefficient of the equation of the vector x is sparse.

The basic steps of sparse representation for image recognition is:

1. Sampling, namely to get the training samples and testing samples.
2. Dimensionality reduction to training samples and testing samples.
3. Set up the error upper-bound
4. Classification of outputs.

Its merit is obvious, firstly, the appearance model is simple and easy to be operated. Compared to the previous method of identification, it grasps the images from overall aspect aspects. In the large scale tracking, such as scale variation attribute test sequences, all the training sample will be considered and get a correlation coefficient, sparse

representative method will do classification according to this coefficient. Scale variation instead increases the sample dimension and contributes to the correct rate. The number of extracted feature is more important than the feature extraction method relatively.

3.1.5 MOTION BLUR (MB)

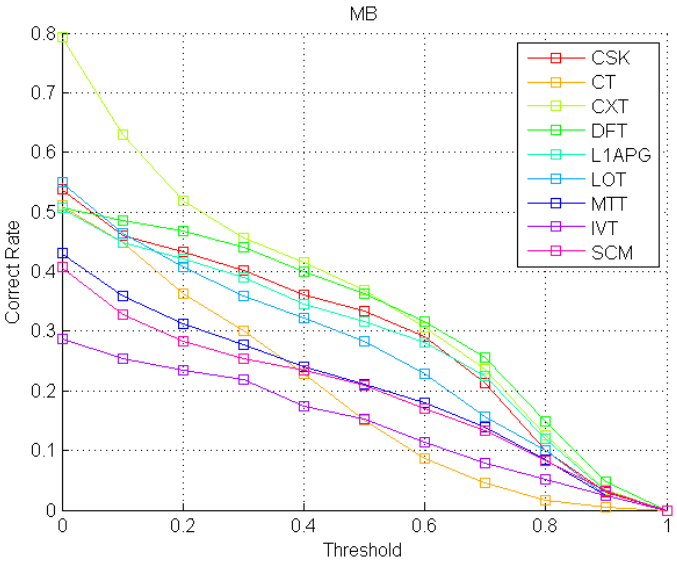


Figure 3.1.5.1 The OPE success rate between nine trackers in OCC case

Motion blur cases are frequently found in practical cases, because images catch by cameras are often blurry, and on the account of the motion of target, the target region may be blurred too.

CXT tracker is significantly outperform the other trackers, the main reason can be figure out from the P-N learning method it employed. P-N learning algorithm is widely used in visual tracking, it can accurately and efficiently label the sample negative or positive, negative sample means that the tracking result is not matching the feature of the actual object, vice versa. CSK use the principle of P-N learning and realize it in application.

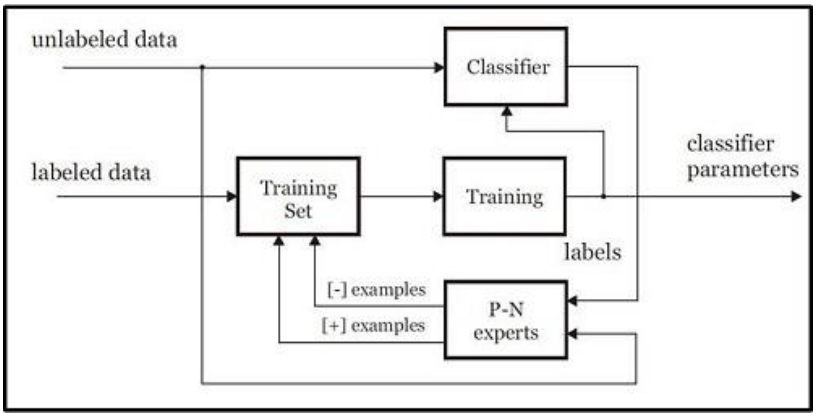


Figure 3.1.5.2 Flow chart of P-N learning algorithm

P-N Learning includes two experts, which is P-expert and N-expert. The previous one is use for recognizing the missed detection (false negatives), which is the positive example; While the N-expert is used for recognizing false alarms (false positives), which is negative example. The result of learning is constructing a smaller label sets as the initial training set, to classify the unlabeled data and rectify the result through P-N expert. This two experts may cause errors, if the error rate is lower than 0.5, compensation of the detector can make this result trustworthy.

P-expert is based on continuation of time, the position in current frame should be close to the previous one, if not, it will be labeled as positive example. So P-expert is using the position information and tracking frame by frame, while N-expert is based on space information. It adopt to single target strategy, allowing only one possible position in a frame. Therefore, N-expert can combine results from detector and classifier, then out put the most possible position, without overlapping with the negative samples.

All in all, as a binary classifier, PN learning has admirable classification performance on judging the structured unlabeled data. In motion blur case, the bound of sample is fuzzy, P-N learning will be appropriate for accurate discrimination.

3.1.6 ILLUMINATION VARIATION (IV)

The result of visual tracking can be not only influenced by internal interference, but also external disturbance, such as illumination variation, which refers to significantly changes of illumination in object tracking region.

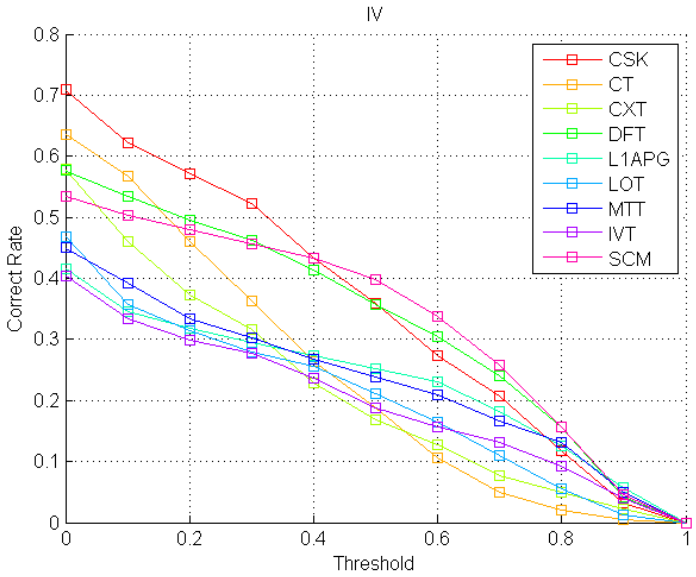


Figure 3.1.6 The OPE success rate between nine trackers in IV case

In this condition, the correct rate of CT are decreased very quickly, at the same time, the correct rate of CSK has a high starting point. As aforementioned, CSK is wearing a histogram of oriented gradient image descriptor. Because of Hog's operation on the local grid of consistently spaced cells in the image, HOG can keep good invariance to the geometric transformation and optical changes of the image, these two kind of deformation will only appear on larger spaces.

The accuracy of DFT also maintains well under different threshold. DFT uses Kronecker Delta function to construct the image, if the feature of the image is gray scale, the constructed distributed fields will not loss the image information from the original ones. Then, a Gaussian filter is employed to smooth the image, which can decrease the sensitivity to illumination change and background noise of the target model.

3.1.7 FAST MOTION (FM)

The bounding box in tracking is comparing with the target's ground truth information to judge whether the tracker has miss the object. If the movement of the ground truth is larger than tm pixels (tm=20), the tracking result would have possibility to be disturbed.

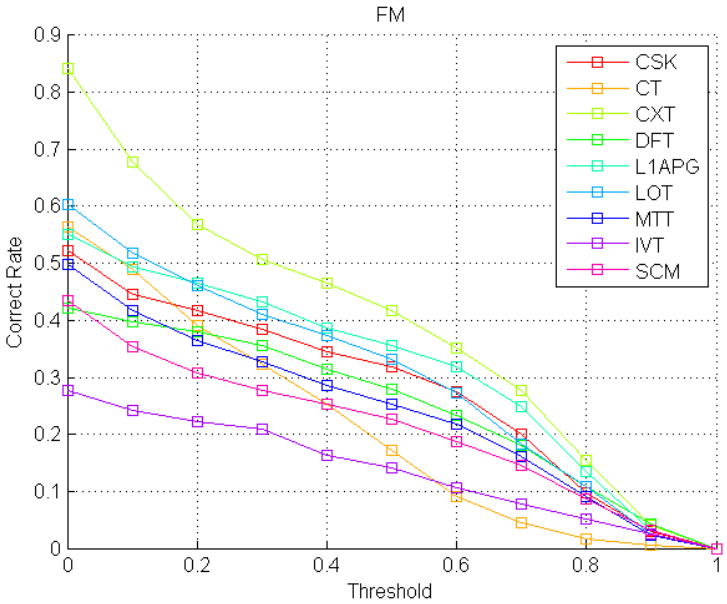


Figure 3.1.7 The OPE success rate between nine trackers in FM case

Figure 3.1.7 shows the success rate in one-pass evaluation between nine trackers in fast motion case. We notice that, among all attributes, the performance of L1APG is the best in this case. It is the only time that the success rate can reach up 0.56 when the threshold is set as 0. Moreover, the correct rate of L1APG is in a stable trend in all attributes, without dropping too rapidly with variation of threshold.

L1 tracker using accelerated proximal gradient approach (L1APG) use l1 norm minimization to minimize the error while regularizing the parameters of the model. Error minimization is for the sake of making our model fitting our training data, while parameters regularization is for the purpose of preventing our model over- fitting for our training data. Because too many parameters will rise the complexity of our model, leading to over-fitting, namely our training error will be small, but that is not our final goal. The ultimate aim is to minimize the test error of the model, which can accurately

predict the new samples. So, we need to make sure that our model is basically simple, and meanwhile, the training error should be minimizing, so that the model parameters will obtain good generalization performance.

In fast motion case, the model will be more and more complicated as the appearance model changes very fast, L1APG can efficiently solve the over-fitting between training data and our model, so it can keep a stable correct rate in the evaluation.

3.2 SPATIAL ROBUSTNESS EVALUATION

In visual tracking evaluation, we are aiming to find out a tracker that is not only very accurate in tracking but also has a good robustness under distraction. So we made the spatial robustness evaluation. In offline tracking, the initial bounding box is catch by ground truth information, in spatial robustness evaluation case, the initial bounding box shifts or scales the ground truth one of the first frame. In the evaluation of this project, we use 12 spatial shifts containing 4 centre shifts and 4 corner shifts, and 4 scale variations. The target size shift in an amount of 10% (Wu and Yang, 2013).

3.2.1 BACKGROUND CLUTTERS CASE (BC)

To test the spatial robustness in visual tracking algorithm when background clutters occurs is useful in engineering aspects. The actual environment, ever-changing in background factors will affect the tracking result. For instance, in the outdoor and high-traffic locations, the constant motion of people or vehicles around will caused serious interference to the specified target pedestrians or vehicles. Trees and construction on both sides of the road will also cause interference to the tracking target. What is more, data-captured camera equipment can be disturbed in indoor circumstance, such as camera dithering by wind and the vibration of moving vehicle. How to extract the target accurately under the condition of complex interference is an important index for measuring the tracking algorithm's performance.

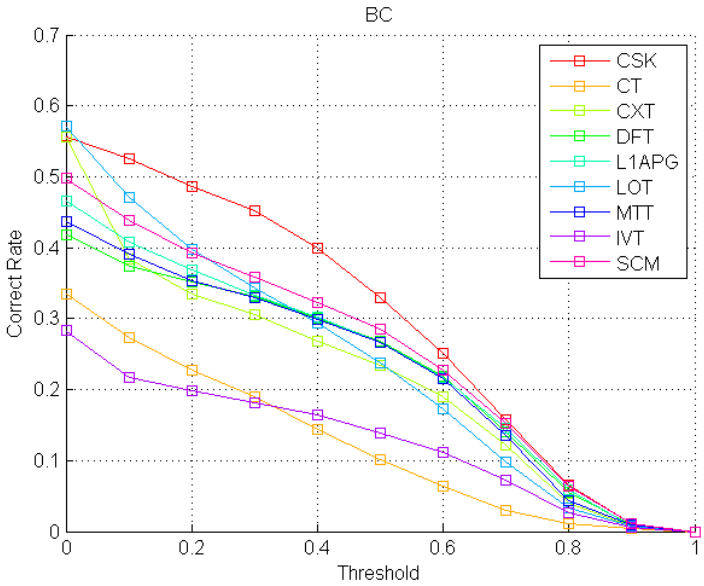


Figure 3.2.1 The SRE success rate between nine trackers in BC case

When initial frame shift 10 percent in size in 12 spatial variation, the correct rate of CXT falls significantly. Because after shifting, the bounding box contains both a part of the subject and a part of the background. Because CXT trackers will take the background near the target into classification consideration, background clutter can make significant reductions in the correct rate. After comprehensive considerations of one-pass evaluation and spatial robustness evaluation, CSK is very robust in background clutter case beyond doubt. Therefore, CSK can be employ in the monitoring system of residential area or traffic.

3.2.2 DEFORMATION AND OCCLUSION CASE (DEF AND OCC)

The introduction of object deformation and occlusion have been describe in chapter 3.1.2. Now we will also make spatial robustness evaluation on it, as the target object for deformation and occlusion are often human body, which is generally considered as linear model, self-shelter and reshaping of the human appearance may have alteration in appearance model or classifier. The comparison result is showing as following.

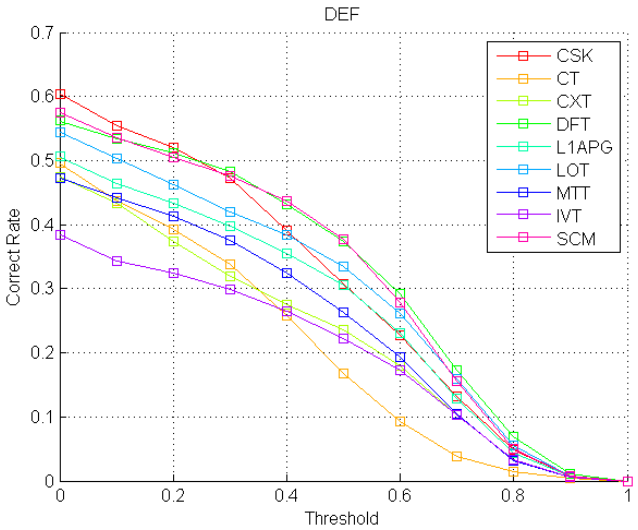


Figure 3.2.2.1 The SRE success rate between nine trackers in DEF case

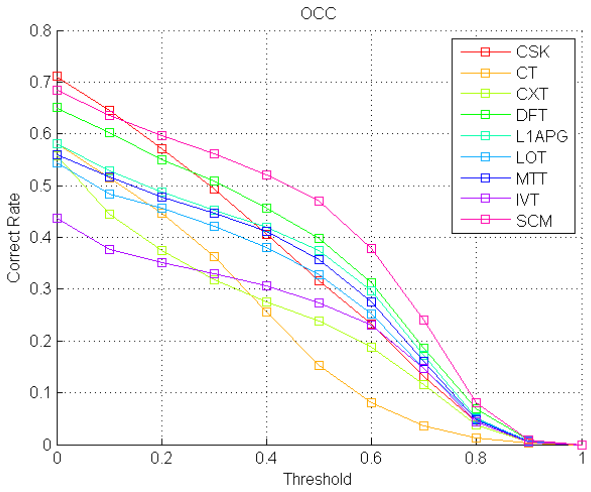


Figure 3.2.2.2 The SRE success rate between nine trackers in OCC case

The correct rate of CSK diminishes sharply, meanwhile, the one of SCM keep fairly good. Analysis of this trend will be given in mathematical aspect.

CSK use Gaussian distribution for object cognition. Gaussian function is a uniform function. The Gaussian filter use weighted average of neighbourhood pixel instead of the pixel value of that point. And every weighted average of neighbourhood pixel is monotonic increase or decrease with the distance between that point and the centre point. This property is very essential, because the image edge is a kind of local image characteristics, if the smooth operation still has very big effect on pixels far away from

the centre, which will cause the image distortion. In the same word, the image characteristics of neighbourhood pixel are distinguished from each other. In the case of spatial robustness evaluation, the spatial shifts including scale variations, centre shifts and corner shifts influence the image characteristics of neighbourhood pixel. So CSK does not perform well in the attributes of background clutter.

As for SCM tracker, it is adaptive in occlusion case, as its robustness is very strong from the correct rate comparison in Figure 3.2.2.1 and Figure 3.2.2.2. Because, sparse representation use the linear combination of all images in the data base to present the target object, this strategy is very robust to noise.

3.2.3 ROTATION CASE (IPR AND OPR)

In visual tracking process, the angle of the object in an image will be changed significantly due to the rotation of the target itself or camera rotating around the lens axis. In the SRE correct rate comparison, we will analyse the performance of these nine trackers synthesize two kind of rotation, which is in-plane rotation and out-plane rotation.

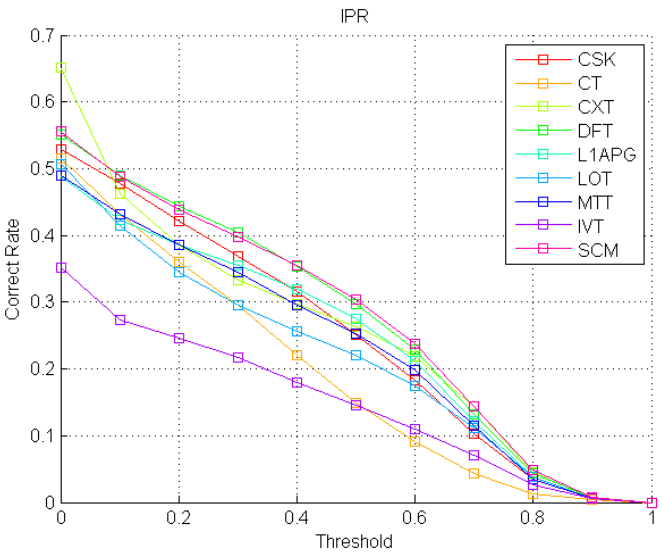


Figure 3.2.3.1 The SRE success rate between nine trackers in OPR case

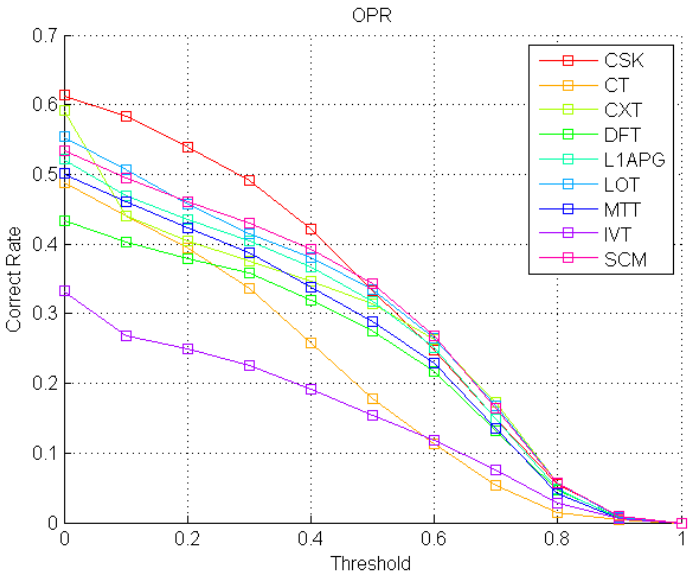


Figure 3.2.3.2 The SRE success rate between nine trackers in IPR case

Combined the result in Figure 3.2.3.1 and Figure 3.2.3.2 and the information of OPE case, the accuracy of CT dramatic decline. Thus it can be seen that CT is not robust enough if the target object has rotation.

CT tracker have characteristics of both generative model and discriminative model. The object is represented by a generative appearance model in CT, this model extracts the features in the compressive domain. A naive Bayes classifier is adopted in this tracker and play the role as distinguish the target from surrounding area, so it is also discriminative.

The core of CT tracker is naive Bayes classifier, its working principle is as following:

1. Division on each attributes appropriately, and then classify part of the samples manually, forming a training sample set. The input of this phase is all the unlabelled data, and the output is feature properties and the training sample.

2. Generation of classifier. The main job is to calculate the occurrence frequency of every attribute and every category in the training sample, and record the results. Its input is the characteristic properties and the training sample, the output is a classifier.

3. Using the classifier to label the sample, its input is classifier and samples, the output is the mapping relationship between the samples and categories.

Therefore, the calculation of conditional probability of each category P(a|y) is the key step in naive Bayes classifier. When the feature is discrete value, we only need to figure out the occurrence of each feature category in the training sample can we estimate the result of P(a|y). When the characteristic properties is continuous values, we generally assume it obeying Gaussian distribution. Thus, the average and variation value of characteristic properties can be calculated.

However, in rotation cases, P(a|y) will sometimes equal to 0, because a certain feature have not appear, this phenomenon decrease the precision of the classifier. This kind of feature lacking need some pre-processing, in CT tracker, Laplace calibration is introduce into the classifier. Just add a one in the feature counting that has not appear. If the amount of the sample is large enough, the result will not be influenced. But our test frame sample is not large enough.

Therefore, on account of sample loss and the decrease of sample quality in rotation, CT does perform well in the SRE test.

3.2.4 SCALE VARIATION CASE (SV)

Scale variation appears in practical tracking cases frequently, Because of the zoom in camera during tracking process, or target movement along the axis of the lens, target size in the image will change significantly.

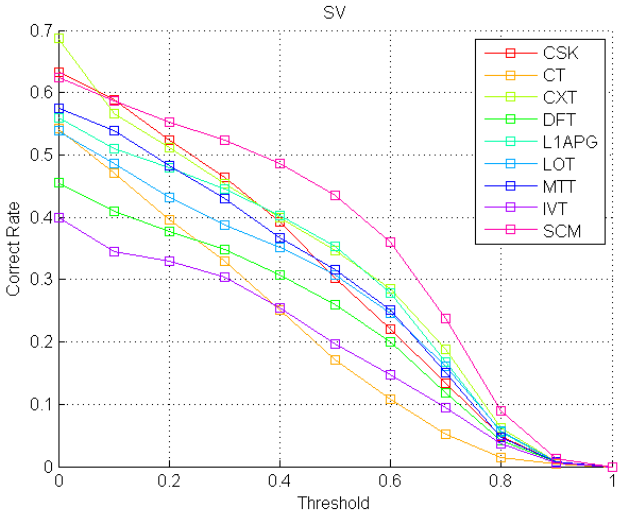


Figure 3.2.4.1 The SRE success rate between nine trackers in SV case

We can see from Figure 3.2.4.1 that SCM keeps ranking the first among all the trackers' performance in the aspect of correct rate, no matter in OPE case or SRE case.

Comprehensive utilized the result in OPE and SRE, SCM is adaptive to the case that the image centre is roughly aligned, the effect of sparse representation would be very good. That is the so-called linear model. For example, the facial features can be described by the rectangle property simply, colour of eyes should be darker than cheek, the colour of the mouth should be heavier than the surrounding and so on. Rectangular characteristics is sensitive to many simple graphical structure, such as edge and line, so it can describe the structure of particular trend (horizontal, vertical and diagonal).

The test sequence we used in evaluation, such as 'Boy', 'Girl', 'Woman' and 'Fleetface', are mostly consist of linear characteristics. We can find that the objects in the red bounding box are symmetric or nearly symmetric from Figure 3.2.4.2

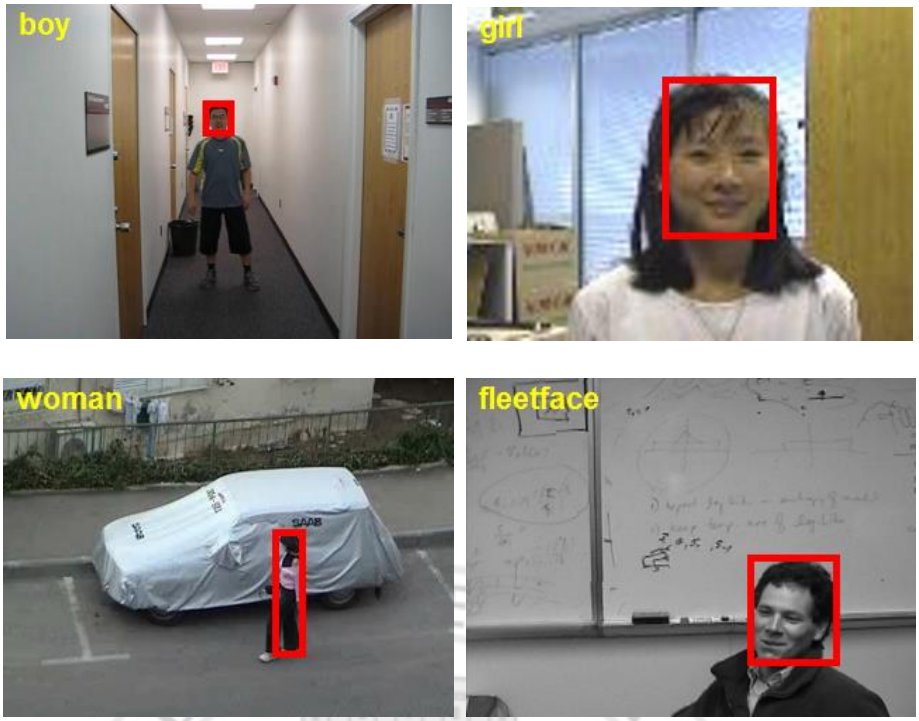


Figure 3.2.4.2 The captured image of some test sequence we used in evaluation

3.2.5 MOTION BLUR CASE (MB)

From the information presenting in Figure 3.2.5, the precision of DFT rise up to some extent than the OPE test.

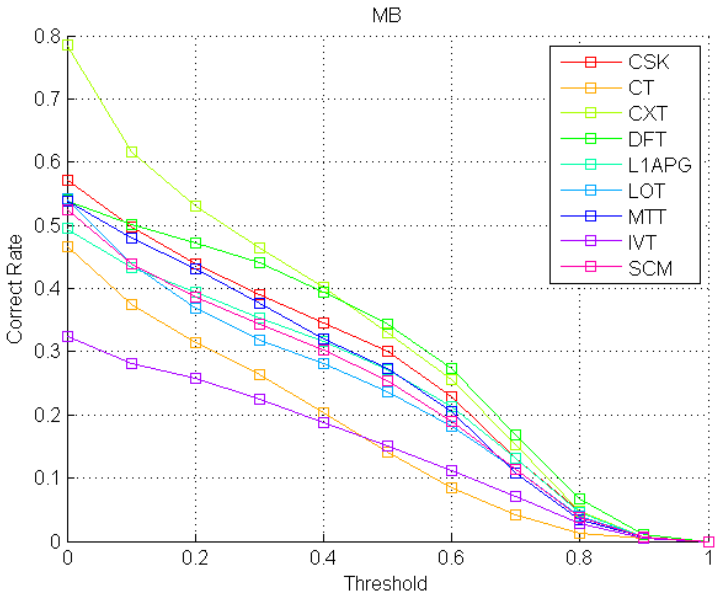


Figure 3.2.5 The SRE success rate between nine trackers in MB case

A significant feature of DFT is image smoothing, General image smoothing can lead to loss of space information, destroyed the information in source images. The Gaussian kernel function is adopted to distribution field smoothing, the information of each pixel will not lost, but pixel position becomes uncertain. Such smooth was conducted in each domain, the feature space can also be smooth and smooth after the DFs can describe movement in the pixel level, shadows, and illumination changes on the influence of the target model. If the blur degree can well serve its image smoothing, that may help in accuracy increasing. Plus, DFT is using sparse representation, the spatial shift is in 12 different direction or scale, which enlarge the sample space, this is a contribution to image smoothing too.

3.2.6 ILLUMINATION VARIATION CASE (IV)

Figure 3.2.6 demonstrates the SRE success rate between nine trackers in illumination variation case, after changes in illumination condition, generative trackers is significantly affected

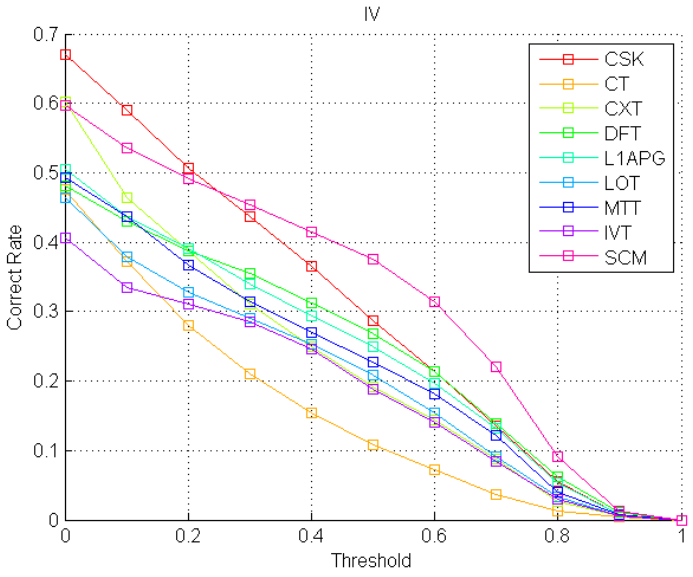


Figure 3.2.6 The SRE success rate between nine trackers in IV case

Generative trackers make full use of the appearance characteristics, and choose a proper model to describe the appearance change. It judges the target location by the similarity

degree between image sample and appearance model. The drawback of generative trackers is that it have not took full advantage of background information and make the distinction bad and susceptible to interference of complex environment.

By contrast, the discriminative trackers have better robustness in this case. The discriminative trackers take both the appearance of the target and the background into account, combining the appearance and foreground to train a classifier. The classifier can present the differences of data in various category. It set up a decision bounding to separate the object and the background by machine learning. The strength of discriminative trackers is reveal sufficiently in external changes case of robustness test.

3.2.7 FAST MOTION CASE (FM)

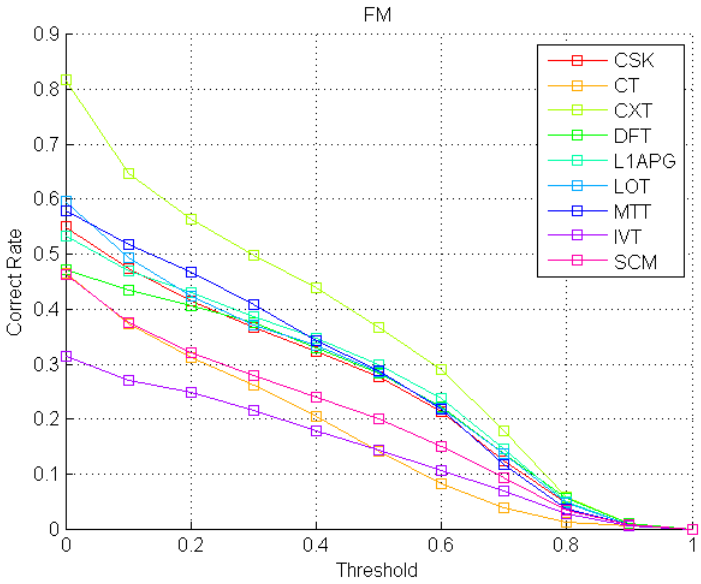


Figure 3.2.7 The SRE success rate between nine trackers in FM case

In the SRE success rate between nine trackers in fast motion case, the correct rate ranking between trackers have not much differences compared to OPE test. As the object move very fast, the background information will be faded out relatively. Even though the discriminative trackers that consider context information have a process of catching background data, fast movement of targets will be surrounded by different context in the subsequent frames. The merit of context consideration will be

inconspicuous except CXT. CXT has another prominent feature: high speed. The tracking speed of CXT can reach 0.03 second per frame. Namely, the update of appearance model of the context tracker can be updated in pace of the fast movement of the object.

3.3 TEMPORAL ROBUSTNESS EVALUATION

In a tracking evaluation on temporal robustness, the test sequence are divided in to 20 segments, the initial frame from each segment will be given with the ground-truth bounding box of the target object, every tracker is initialized and runs to the end of the sequence. We evaluate the tracker's performance on each segment. In order to make the result more convincing, 10 test sequences are included in test data, this ten videos are grouped into different attributes and I get the correct rate figures from two ways: to use the average of correct rate in every 20 segments of all the 10 test sequence in one attribute category, and to take the maximum value of every 20 segment out of all the test sequence. We found that the results are quite different.

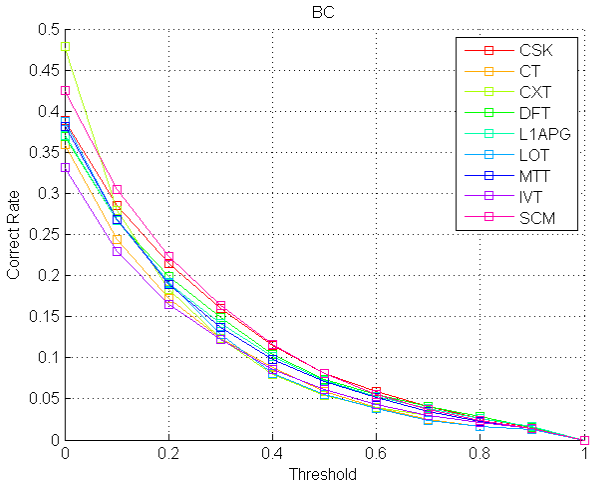


Figure 3.3.1 TRE success plot in BC case with mean value of segments

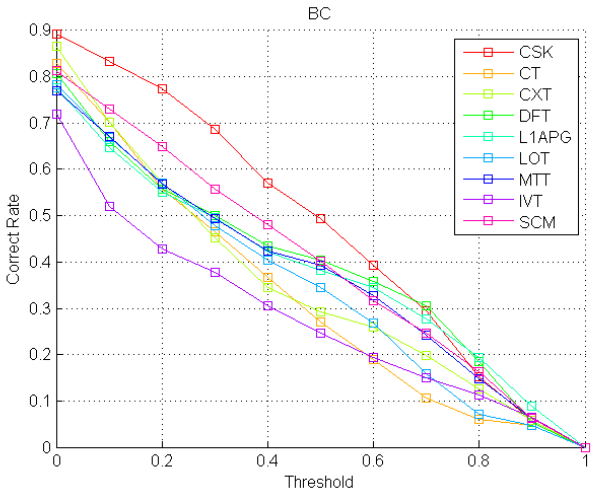


Figure 3.3.2 TRE success plot in BC case with max value of segments

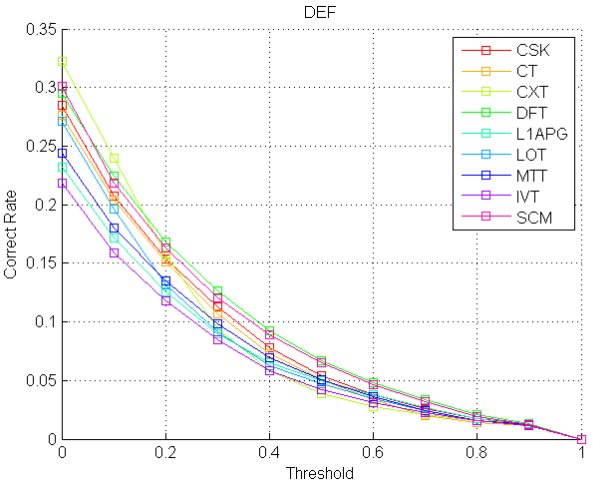


Figure 3.3.3 TRE success plot in DEF case with mean value of segments

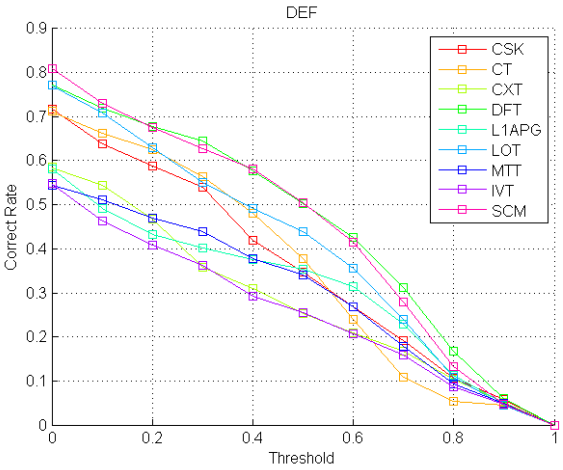


Figure 3.3.4 TRE success plot in DEF case with max value of segments

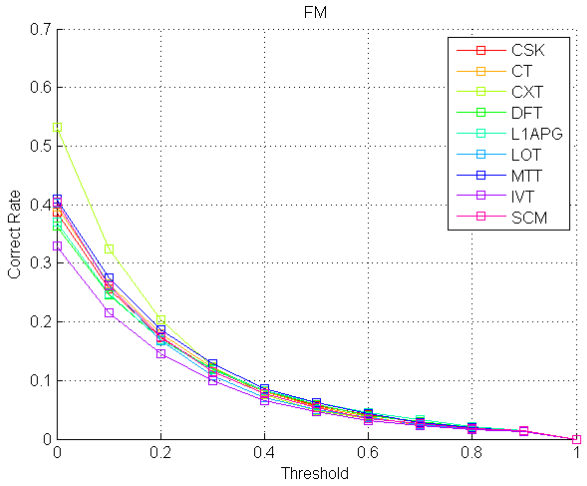


Figure 3.3.5 TRE success plot in FM case with mean value of segments

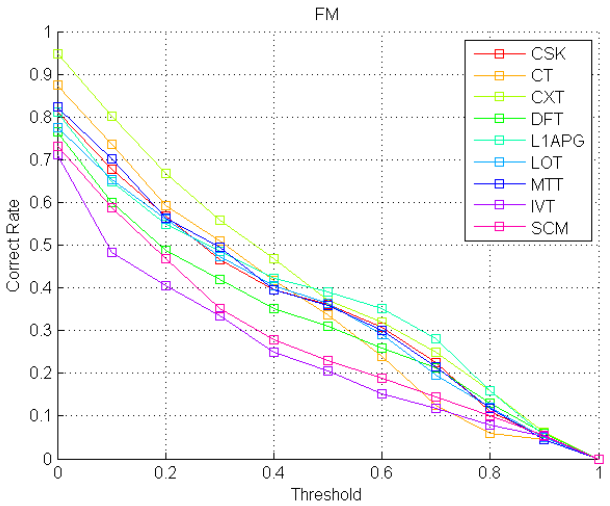


Figure 3.3.6 TRE success plot in FM case with max value of segments

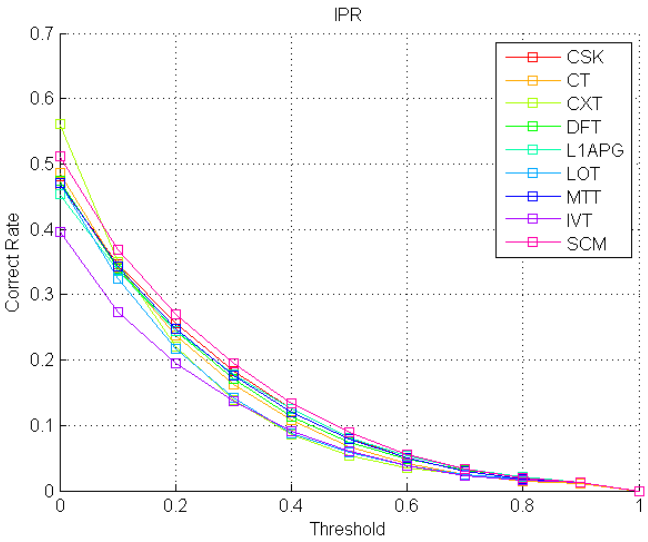


Figure 3.3.7 TRE success plot in IPR case with mean value of segments

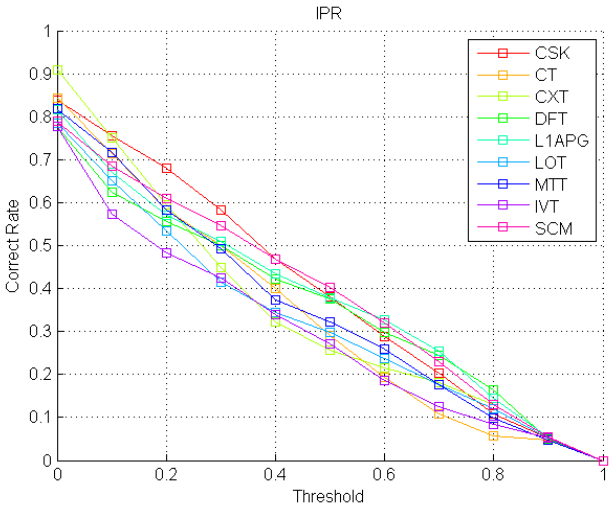


Figure 3.3.8 TRE success plot in IPR case with max value of segments

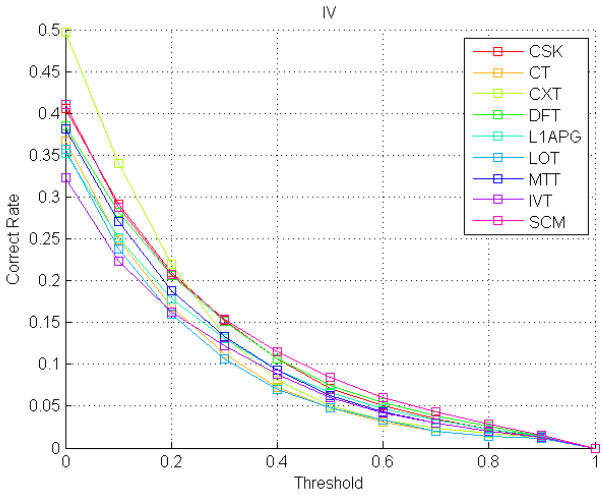


Figure 3.3.9 TRE success plot in IV case with mean value of segments

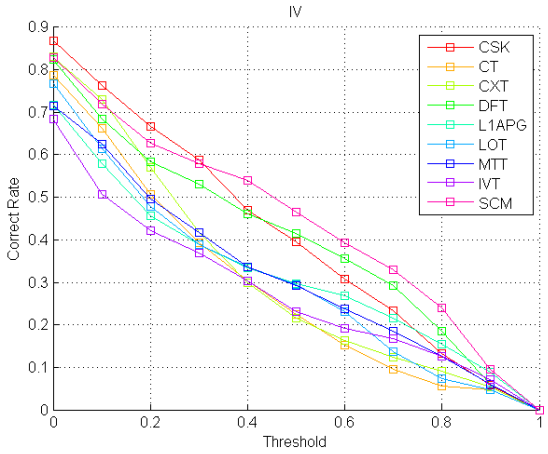


Figure 3.3.10 TRE success plot in IV case with max value of segments

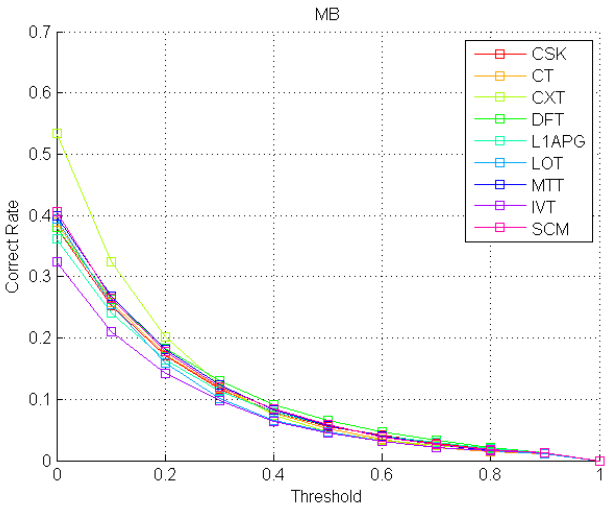


Figure 3.3.11 TRE success plot in MB case with mean value of segments

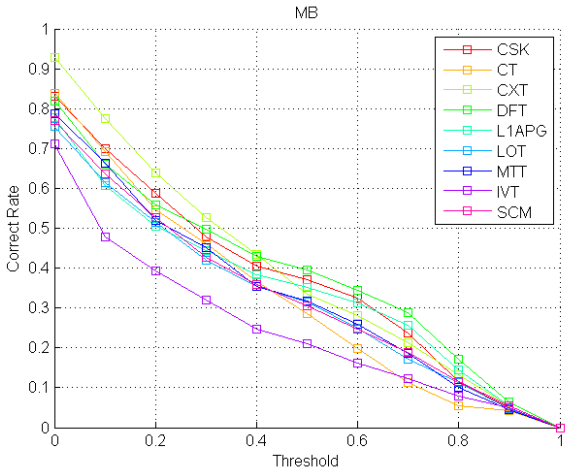


Figure 3.3.12 TRE success plot in MB case with max value of segments

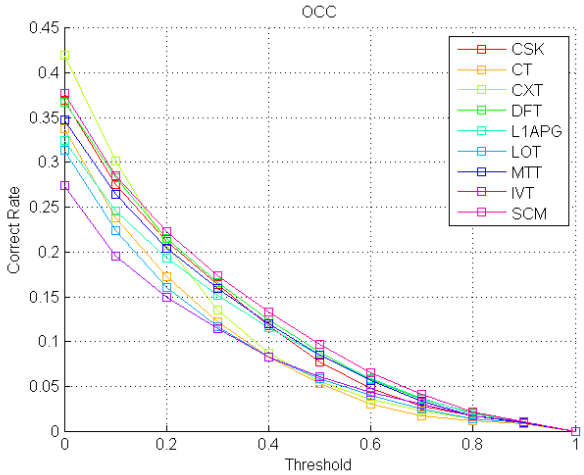


Figure 3.3.13 TRE success plot in OCC case with mean value of segments

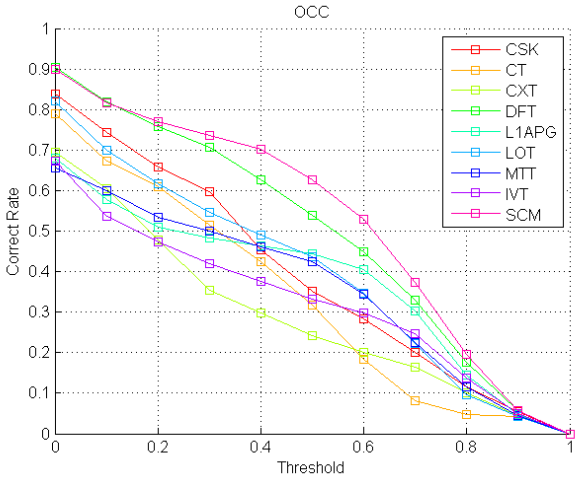


Figure 3.3.14 TRE success plot in OCC case with max value of segments

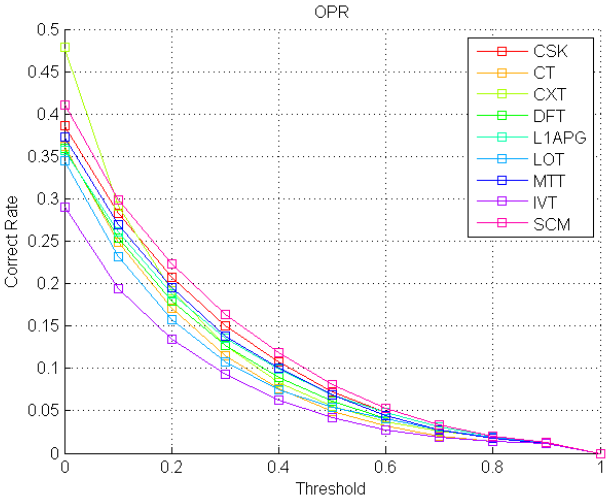


Figure 3.3.15 TRE success plot in OPR case with mean value of segments

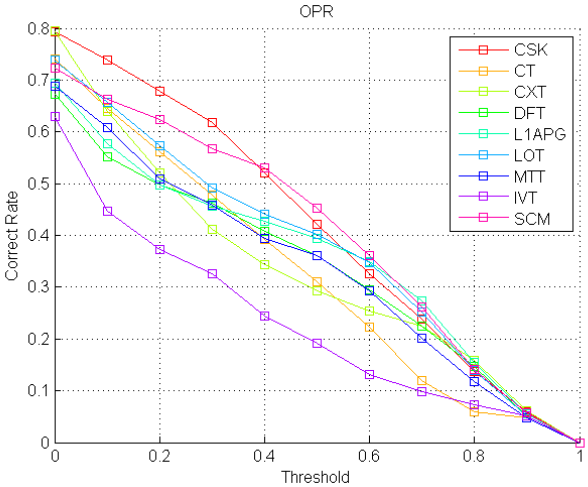


Figure 3.3.16 TRE success plot in OPR case with max value of segments

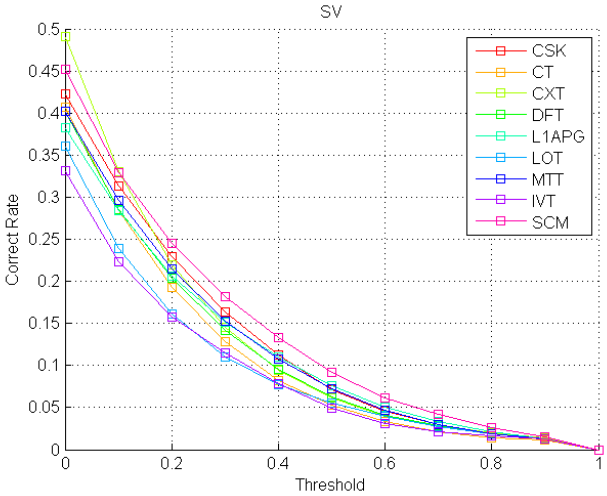


Figure 3.3.17 TRE success plot in SV case with mean value of segments

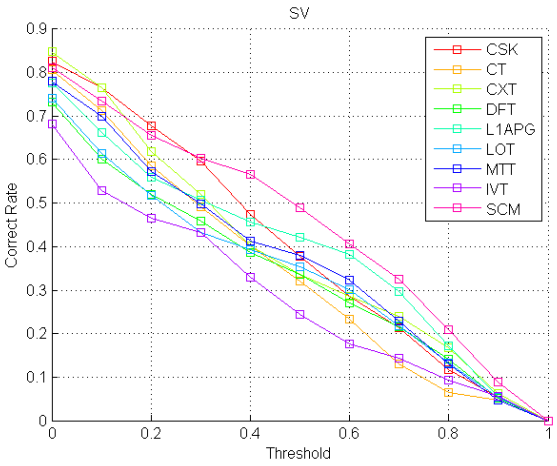


Figure 3.3.18 TRE success plot in SV case with max value of segments

The significant discrepancy between the result that taking mean value and maximum among segments shows that the correct rates between different segments in the test sequent with same attribute are very different. Then I set the threshold as 0, which is the milder conditions for tracking accuracy judgment, to evaluate the tracking precision in detailed. Data of 5 test typical sequences are listed with tracker name and correct rates.

Table 3.3 Best and worst tracker with correct rate in TRE test among five segments

#of segment	1st	5th	10th	15th	20th
basketball					
best	LOT/CSK 1	CXT 0.4894	CXT 0.09367	CXT 0.2691	all 0.05

worst	CXT 0.05241	IVT 0.1444	IVT 0.03291	IVT 0.0045	all 0.05
boy					
best	CT/CXT 1	DFT 0.4146	CXT 0.3159	DFT 0.3179	MTT 0.6
worst	IVT 0.3338	CT 0.0021	CT 0.0031	IVT 0.0058	CT 0.35
couple					
best	CT 0.7071	L1APG 0.5130	SCM 0.3855	DFT 0.2353	all 0.05
worst	IVT 0.1	DFT 0.0609	CXT 0.1928	IVT 0.0588	all 0.05
deer					
best	CSK/CXT 1	SCM 0.6393	L1APG 0.5532	CXT 0.4242	Half 0.45
worst	CT 0.0563	CSK 0.2295	DFT 0.1064	IVT 0.0909	CT 0.35
ironman					
best	CXT 0.4518	LOT 0.2302	MTT 0.3269	MTT 0.0724	CXT 0.4
worst	DFT 0.0964	DFT 0.0216	DFT/IVT 0.0096	IVT 0.0145	Others 0.05

From Table 3.3, we can obtain that correct rate variance between the best tracker and the worst one can be as large as 94.34 percent. And generally, accuracy in segments are declining from the 1st segment to the 20th segment.

Precision dropping during the whole tracking may be caused by accumulation of internal storage. In the other aspect, if a tracker has large difference of tracker's correct rate ranking between TRE test with mean value and max value of segments, we can tell that this tracker is not robust enough.

The evaluation result of temporal robustness is not very successful, another reason can be drawn from the test sequence. I used ten test sequence for evaluation in every. But these test sequence contains too many attribute at the same time, for example, test sequence 'Tiger' contains 7 attributes: (Illumination Variation) IV, (Occlusion) OCC, (Deformation) DEF,(Motion Blur) MB, (Fast Motion) FM, (In-Plane Rotation) IPR and (Out-of-Plane Rotation) OPR. Intermixed information may influence the judgement.

3.4 CONCLUSION OF TRACKERS PERFORMANCE EVALUATION

After estimate the trackers' performance of one-pass evaluation (OPE) temporal robustness evaluation (TRE) and spatial robustness evaluation (SRE) in correct rate and

robustness respectively, we find that incremental learning for robust visual tracking (IVT) is always ranking the last in all the success plot. Updating the online model by using particle filters, which IVT introduced to visual tracking, is still not very robust and precise (Ross, 2008). All in all, both generative trackers and discriminative trackers have their merit and demerit. Even though the discriminative trackers out-perform the generative trackers in many cases, but generative model still have strength in visual tracking that the discriminative one do not have.

Generative trackers aim at choosing proper model to simulate the changes in targets' appearance, to make decision with accordance to the similarity between image sample and the appearance model. The drawback of this kind of trackers is that the tracker take advantage of object's appearance information but ignore the context information, so the complex background can easily interfere the tracking result and give the tracker a bad distinguishability.

As for discriminative trackers, its advantage lies in the flexible classification boundary. Compared to pure probability method or generated model tracking, discriminative trackers are more distinct in classification, it can clearly distinguish the characteristics between different classes. In the condition of perspective change in target appearance, partial shade, scale change, the tracking effect is better. Discriminant model is more simple and easy to learn than the generation model. Discriminant tracker also exist some shortcomings: one is that it can't reflect the characteristics of the training data itself, which can only judge the category of target samples without describing the target appearance.

Machine learning is an essential part in visual tracking, most of the machine learning are proceeding with model. Generally, we divide target tracking into two part: feature extraction and target tracking algorithm. The extraction of target characteristics can be roughly divided into the following kinds:

1. Use colour histogram of the target area as the features, with rotation invariance, and it is not affected by the change of the target size and shape, it distributes roughly the same in the colour space.
2. Contour features of the target, this kind of algorithms are relatively fast, and also have better effect when the target has a small part being shaded
3. Texture feature of targets, texture feature improves the contour feature in tracking results.

In the aspect of classifier, the trackers that we estimate in this project have roughly the following four kind of classifier, their function is adaptive to different circumstance.

Naive Bayes (NB)

The classifier of compressing tracker is naïve Bayes classification. Its working principle is like counting numbers. If the conditional independence assumption is satisfied, Naive Bayes will converge much faster than discriminant models. Thus when the sample is not large, this classifier performs accurately.

Logistic Regression (LR)

CSK is employing logistic regression for classification. It is using least-squares method but the solution for classifying is iteratively reweighted least squares, equivalently to the logistic regression (Cizek, 1999). Compared to the conditional independence assumption of Naive Bayes, Logistic Regression need not to consider the correlation between samples. If you want some probability information or want to have a bigger data base in order to update and improve the model, LR is suitable for using.

Decision Tree (DT)

Boosting method is employed by SCM, which is a kind of improved decision tree. An important feature of DT is that it contains no parameters. Thus, you need not to consider the data is linear or not. The main drawback of DT is over-fitting. So ensemble learning algorithm such as Boosted Tree are developed afterward.

3.5 ONLINE SYSTEM

In the previous chapter, we have make comparison of the correct rate in nine trackers. High speed tracking is essential in online system, low tracking speed of the tracker would cause frame loss during tracking and significantly influence the precision.

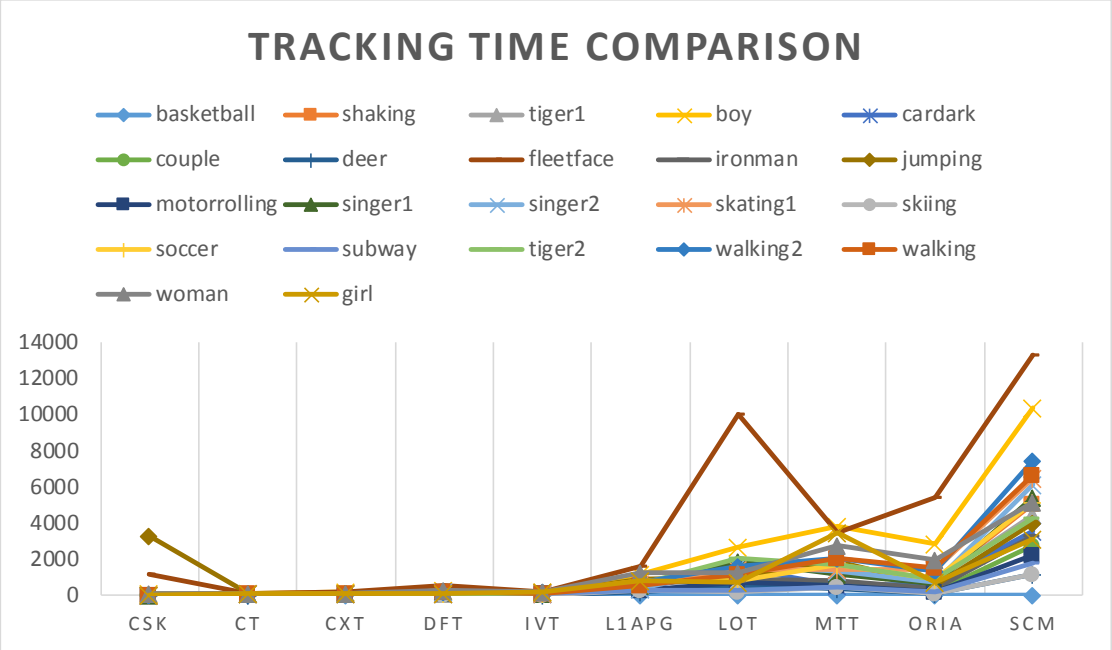


Figure 3.5.1 Tracking time comparison

Figure 3.5.1 compares the tracking time of all these nine trackers, which are measured in second. Each curve represents the tracking time in different test sequence. Even though the number of frames in every test sequence varies from 71 to 500, but the tracker of CSK, CT, CXT, DFT and IVT are using less time compared to the other trackers. As the curves of these five trackers are closed to each other with some overlapping. I will show the time information of these five high-speed tracker in more detail from Figure 3.5.2. Then we can see that CT is the fastest one among all the high-speed tracker.

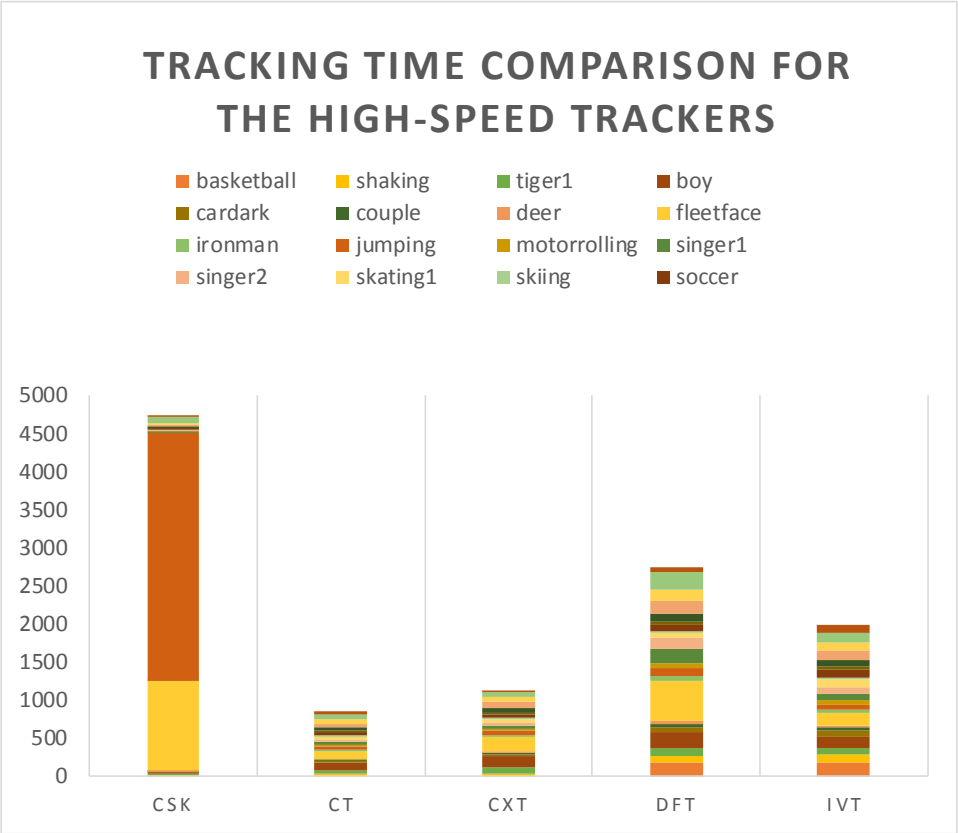


Figure 3.5.2 Tracking time comparison for the high-speed tracker

From the previous robustness and tracking speed analysis, we find that the correct rate of CT, CXT, CSK and DFT are fairly high compared to the other trackers. Thus, I determine to employ these trackers to build an online tracking system through.

The online system of DFT is not in good condition. A very tiny occlusion will cause object missing during tracking. As for the CSK online tracking system, frame loss is very serious, it can only operate in the condition that the object move very slowly. The CT online system performs the best, I use a stuff animal with similar colour as the background (the door is also in brown colour), the object can be successful tracking with movement, scale variation and angle variation. The tracking result is showing as following.

A conclusion can be drawn from the building of online system. The performance of visual tracking online system are mostly depending on the tracking speed, the overall correct rate of CSK is better than CT, but due to the higher speed of CT, whose online

tracking system is more successful in tracing the object, while CSK will miss the target even though the object movement is very small.

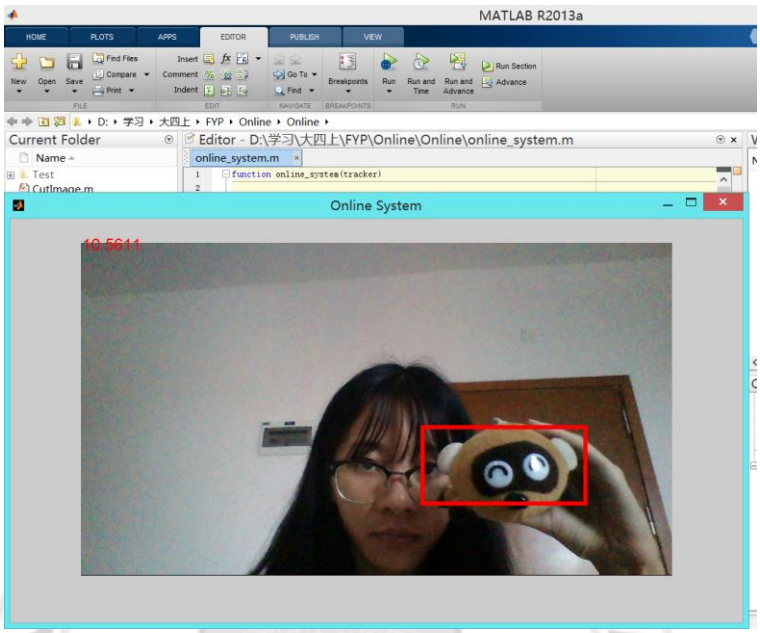


Figure 3.5.3 Initialization by a bounding box in tracking

Figure 3.5.4 Object movement in tracking

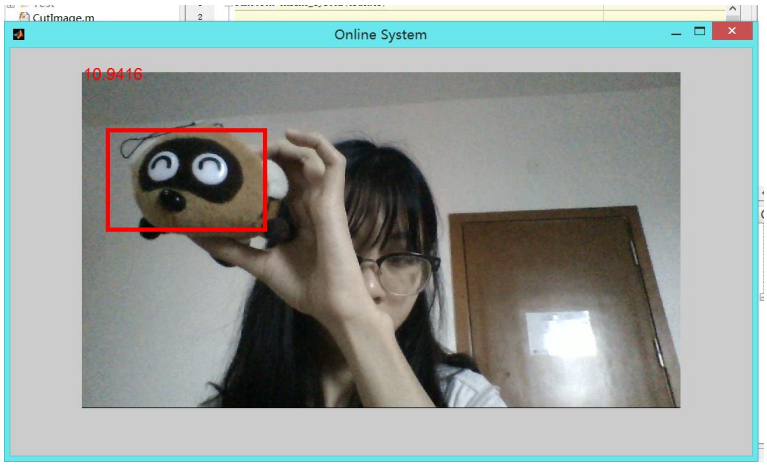


Figure 3.5.5 Scale variation in tracking

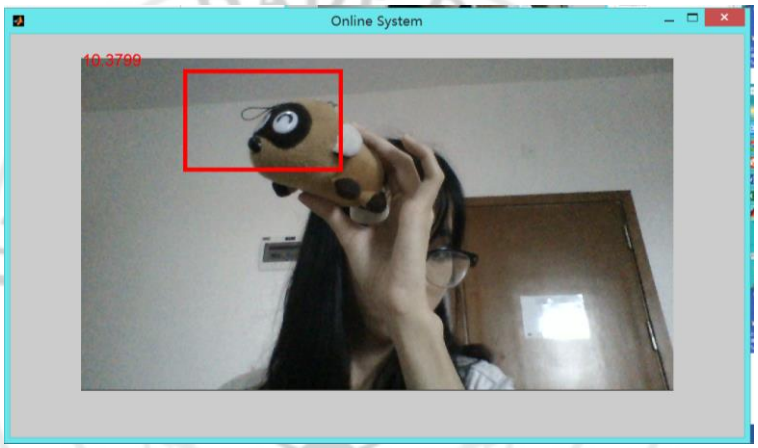


Figure 3.5.6 Angle variation in tracking

Concluding, there are two key points in the realization of online system based on Matlab. Firstly, the difference between online system and offline system is the way of fetching bounding box in the initial frame, in offline system, the initialization is based on the object's ground truth information, while in online system, the initial bounding box is fetching by the user. In my work, I made the system read the position of the mouse of the computer with Matlab online system, and use two diagonal point to get the initial position. Secondly, the speed of the tracker should be high enough to avoid frame loss during tracking

CHAPTER 5 CONCLUSION AND FUTURE WORK

In this project, nine visual tracking algorithms in two categories were discussed, then I gave the comparison between generative algorithms and the discriminative-model ones. From their different success rates, we can find the influence of test sequence attributes to tracking results. Generally, discriminative trackers out-perform the generative trackers. However, if the test sequence has very few frames, the results will be opposite. The online system is made with high speed algorithm with consideration of the robustness performance analysis and tracking speed analysis.

As for future works, I find that in the online system based on Matlab programming can only be adaptive in tracking that the object movement is slow, the CT tracker is in highest speed in the offline tracking system, but when I employ the conception in offline programming into our online system, the speed can not reach as high as the offline one any more. Actually, many online tracking system are based on OpenCV software in elipse. (Zhang, 2013) Online system programing is better by C language than Matlab code, the previous one would attain a rapid operation.

REFERENCES

Bao, C., Wu, Y., Ling, H. and Ji, H., 2012. "Real time robust l1 tracker using accelerated proximal gradient approach". In 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1830-1837.

Cizek, G. J. and Fitzgerald, S. M., 1999. "An Introduction to Logistic Regression". Measurement and evaluation in counseling and development.

Dinh, T. B., Vo, N. and Medioni, G., 2011. "Context tracker: Exploring supporters and distracters in unconstrained environments". In 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1177-1184.

Gonzalez, R. C., and Richard E. W., 2002. Digital Image Processing, 3ed. Prentice Hall.

Hare, S., Saffari, A. and Torr, P. H., 2011. "Struck: Structured output tracking with kernels". In 2011 IEEE International Conference on Computer Vision (ICCV), pp. 263-270.

Lowe, D. G., 1999. "Object recognition from local scale-invariant features". In Seventh IEEE International Conference on Computer Vision, Vol. 2, pp. 1150-1157, 1999.

Mei, X., Ling, H., Wu, Y., Blasch, E. and Bai, L., 2011. "Minimum error bounded efficient ℓ 1 tracker with occlusion detection". In 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1257-1264.

Ross, D. A., Lim, J., Lin, R. S. and Yang, M. H., 2008. Incremental learning for robust visual tracking. International Journal of Computer Vision, Vol. 77, pp. 125-141.

Wu, Y., Lim, J. and Yang, M. H., 2013. "Online object tracking: A benchmark". In 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2411-2418.

Zhang, K., Zhang, L. and Yang, M. H., 2012. "Real-time compressive tracking". Computer Vision–ECCV 2012, Vol. 7574, pp. 864-877.

Zhang, T., Ghanem, B., Liu, S. and Ahuja, N., 2012. "Robust visual tracking via multi-task sparse learning". In 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2042-2049.

Zhang, Z. and Piccardi, M., 2007. "A review pf tracking methods under occlusions". In MVA2007 IAPR Conference on Machine Vision Applications, pp. 146-149.

Krutika A Veerapur, Ganesh V. Bhat,2013. “Colour Object Tracking On Embedded Platform Using Open CV”. International Journal of Recent Technology and Engineering (IJRTE) Vol.2, Issue-3

APPENDIX

The Matlab code of online system is attaching in the appendix. It combines the online tracking system of tracking-by-detection with kernels (CSK), real-time compressive tracking (CT) and distribution fields for tracking (DFT).

```
clc;close all;
%
Continue=1; % Continue Control
State=1;
P1=[0,0];
res=[0,0,0,0];
init_image=zeros(452,452,3);
if strcmpi(tracker,'DFT')
    init_last_motion=[0,0];
elseif strcmpi(tracker,'CT') || strcmpi(tracker,'CSK')
    first_image=1;
end
%% Init Camera
hard=imaqhwinfo;
name=hard.InstalledAdaptors;
vid=videoinput(name{2});
vid.FramesPerTrigger=1;
vid.TriggerRepeat=Inf;
vid.ReturnedColorspace='rgb';
%% Init Figure
h=figure('NumberTitle','off','Name','Online System',...
    'MenuBar','none',...
    'Visible','on');
set(h,'doublebuffer','on');
set(h,'Position',[0,0,1,1]);
set(h,'WindowButtonDownFcn',@FigureButtonDown)
hold on
axis off;
%% Main Part
% Start camera
start(vid);
%
duration=0;
while 1
    % If Quit
    if ~Continue
```

```
        stop(vid);
        delete(vid);
        break;
    end
    % If Continue
    % get frame
    frame=CutImage(getdata(vid,1,'uint8'),452,452);
    %disp(vid.FramesAvailable)
    if vid.FramesAvailable>=2
        flushdata(vid);
    end
    %Plot frame
    imshow(frame);
    %Plot tracker
    if State==2
        rectangle('Position',[P1(1) P1(2) 5 5],'LineWidth',10,'EdgeColor','r');
    elseif State==3
        if strcmpi(tracker,'DFT')
            target_image=double(frame);
            tic
            [res,init_last_motion] =
run_DFT_change(floor(res),init_image,init_last_motion,target_image);
            duration=toc;
            init_image=target_image;
        elseif strcmpi(tracker,'CT')
            if first_image==1
                init_image=frame;
                init_rect=floor(res);
                %%
                para=paraConfig_CT('Nothing');

                ftrparams=para.ftrparams;

                trparams =para.trparams;

                M = para.M;% number of all weaker classifiers, i.e,feature pool
                %------------------------Learning rate parameter
                lRate = para.lRate;

                initstate = init_rect;%initial tracker

                img = init_image;

                [h,w,ch] = size(img);
```

```
if ch==3
    img = double(rgb2gray(img));
else
    img = double(img);
end
img = img - mean(img(:));

trparams.initstate = initstate;% object position [x y width height]

% Sometimes, it affects the results.
%------------------------
% classifier parameters
clfparams.width = trparams.initstate(3);
clfparams.height= trparams.initstate(4);

%------------------------
posx.mu = zeros(M,1);% mean of positive features
negx.mu = zeros(M,1);
posx.sig= ones(M,1);% variance of positive features
negx.sig= ones(M,1);

%------------------------
%compute feature template
[ftr.px,ftr.py,ftr.pw,ftr.ph,ftr.pwt] =
HaarFtr(clfparams,ftrparams,M);
%------------------------
%compute sample templates
posx.sampleImage =
sampleImg(img,initstate,trparams.init_postrainrad,0,100000);
negx.sampleImage =
sampleImg(img,initstate,1.5*trparams.srchwinsz,4+trparams.init_postrainrad,50);
%----------------------------------
%--------ClfMilBoost update
%--------extract haar features
iH = integral(img);%Compute integral image
selector = 1:M;% select all weak classifier in pool
posx.feature = getFtrVal(iH,posx.sampleImage,ftr,selector);
negx.feature = getFtrVal(iH,negx.sampleImage,ftr,selector);
%-----------------------------------------------
%-----------------------------------------------
[posx.mu,posx.sig,negx.mu,negx.sig] =
clfStumpUpdate(posx,negx,posx.mu,posx.sig,negx.mu,negx.sig,lRate);% update
distribution parameters
```

```
                    posx.pospred = weakClassifier(posx,negx,posx,selector);% Weak
classifiers designed by positive samples
                    negx.negpred  = weakClassifier(posx,negx,negx,selector);%  ... by
negative samples
                    %%

                    first_image=first_image+1;
                else
                     target_image=frame;
                     tic

[init_rect,posx,negx]=run_CT_change(target_image,init_rect,trparams,ftr,selector,pos
x,negx,lRate);
                     duration=toc;
                     res=init_rect;
%                    first_image=first_image+1;
%                    if first_image==100
%                          disp('Reverse')
%                          first_image=1;
%                    end
                end
            elseif strcmpi(tracker,'CSK')
                if first_image==1
                    init_image=frame;
                    init_rect=floor(res);

[~,alphaf,z,pos,sz,resize_image,yf,target_sz]=run_CSK_change(init_rect,init_image,1,
0,0,0,0,0,0,0,0);
                    first_image=first_image+1;
                else
                    target_image=frame;
                    tic

[init_rect,alphaf,z,pos,sz,resize_image,yf,target_sz]=run_CSK_change(init_rect,target
_image,0,alphaf,z,pos,sz,resize_image,yf,target_sz);
                    duration=toc;
                    res=init_rect;
                    %disp(res)
                end
            end
            rectangle('Position',res,'LineWidth',4,'EdgeColor','r');
            if duration~=0
                text(5,5,num2str(1/duration),'fontsize',15,'color','r');
```

```
            end
        end
        % Draw Buffer
        drawnow;
end

function FigureButtonDown(src,event)
        pt=get(gca,'CurrentPoint');
        if State==1
            P1(1)=pt(1,1);
            P1(2)=pt(1,2);
            State=2;
        elseif State==2
            P2(1)=pt(1,1);
            P2(2)=pt(1,2);
            if P1==P2
                P2=P2+1;
            end
            res(1)=min(P1(1),P2(1));
            res(2)=min(P1(2),P2(2));
            res(3)=abs(P1(1)-P2(1));
            res(4)=abs(P1(2)-P2(2));
            init_image=double(frame);
            State=3;
        elseif State==3
            Continue=0;
        end
end

end
```