VISUAL OBJECT TRACKING USING CO-DIFFERENCE FEATURES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF
MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

By
Hüseyin Seçkin Demir
August 2017

VISUAL OBJECT TRACKING USING CO-DIFFERENCE FEATURES

By Hüseyin Seçkin Demir

August 2017

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Ahmet Enis Çetin(Advisor)

Süleyman Serdar Kozat

Rengül Atalay

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

VISUAL OBJECT TRACKING USING CO-DIFFERENCE FEATURES

Hüseyin Seçkin Demir
M.S. in Electrical and Electronics Engineering
Advisor: Ahmet Enis Çetin
August 2017

Visual object tracking has been one of the widely studied computer vision tasks which has a broad range of applications in various areas from surveillance to medical studies. There are different approaches proposed for the problem in the literature. While some of them use generative methods where an appearance model is built and used for localizing the object on the image, others use discriminative approaches that models the object and background as two different classes and turns the tracking task into a binary classification problem. In this study, we propose a novel object tracking algorithm based on co-difference matrix and compare its performance with the recent state-of-the-art tracking algorithms on two specific applications. Experiments on a large class of datasets show that the proposed co-difference based object tracking algorithm has successful results in terms of track maintenance, success rate and localization accuracy.

The proposed algorithm uses co-difference matrix as the image descriptor. Extraction of co-difference features is similar to the well known covariance method. However the vector product operator is redefined in a multiplication-free manner. The new operator yields a computationally efficient implementation for real time object tracking applications.

For our experiments, we prepared a comparison framework that contains over 70000 annotated images for visual object tracking task. We conducted experiments for two different application areas seperately. The first one is infrared surveillance sytems. For this application, we used a thermal image dataset that contains various objects such as humans, cars and military vehicles. The second application area is cell tracking on time-lapse microscopy images. Image sequences for the second application contain cells of different shapes and sizes. For both applications, datasets include a considerable amount of rotation and

background clutter.

Performance of the tracking algorithms are evaluated quantitatively based on three different metrics. These metrics measure the track maintenance score, success rate and localization accuracy of an algorithm. Experiments indicate that the proposed co-difference based tracking algorithm is among the best performing methods by having the highest localization accuracy and success rate for the surveillance dataset, and the highest track maintenance score for the cell motility dataset.

Keywords: Object Tracking, Co-difference Matrix, Covariance Matrix, Video Processing, Image Processing.

ÖZET

ORTAK FARK ÖZNİTELİKLERİ KULLANARAK GÖRSEL NESNE TAKİBİ

Hüseyin Seçkin Demir
Elektrik-Elektronik Mühendisliği, Yüksek Lisans
Tez Danışmanı: Ahmet Enis Çetin
Ağustos 2017

Görsel nesne takibi, gözetlemeden tıbbi çalışmalara kadar birçok alanda geniş bir uygulama alanına sahip, yaygınca çalışılan bilgisayarla görü alanlarından birisi olmuştur. Literatürde problemin çözümüne yönelik farklı yaklaşımlar sergilenmektedir. Bazıları, nesnenin görüntü üzerindeki konumunu bulmak için bir görüntü modelinin oluşturulup kullanıldığı üretici yöntemleri kullanırken, diğerleri nesne ve arkaplanı iki ayrı sınıf olarak modelleyerek takip işlemini ikili sınıflandırma problemine dönüştüren ayırıcı yaklaşımları kullanmaktadır. Bu çalışmada, ortak-fark matrisine dayalı yeni bir nesne takip algoritması sunuyor ve bu algoritmanın performansını iki farklı uygulama üzerinde güncel takip algoritmaları ile karşılaştırıyoruz. Geniş bir veri seti ile yapılan deneyler, sunduğumuz ortak-farka dayalı obje takip algoritmasının takip koruma, başarı oranı ve konumlandırma hassasiyeti yönlerinden başarılı sonuçlar verdiğini göstermiştir.

Sunulan algoritma, imge betimleyici olarak ortak-fark matrisini kullanmaktadır. Ortak-fark özniteliklerinin çıkarılması yaygınca bilinen kovaryans metoduna benzemektedir. Ancak, vektör çarpım operatörü, çarpma işlemi içermeyecek şekilde tanımlanmıştır. Bu yeni operatör, gerçek zamanlı nesne takip uygulamaları için işlemsel olarak verimli bir çözüme imkan vermiştir.

Deneylerimizde kullanmak üzere, görsel nesne takibi için işaretlenmiş 70000'den fazla imgeyi içeren bir karşılaştırma altyapısı oluşturduk. Deneylerimizi iki farklı uygulama alanı için ayrı ayrı gerçekleştirdik. İlk ugulama alanı kızılötesi gözetleme sistemleriydi. Bu uygulama için, insanlar, arabalar ve askeri araçlar gibi çeşitli nesneleri içeren bir termal görüntü veri seti kullandık. İkinci uygulama alanı hızlı çekim mikroskop görüntülerinde hücre takibiydi. Bu uygulama için kullanılan görüntü dizileri, farklı boyut ve şekillerdeki hücreleri

içermekteydi. İki uygulama için kullanılan veri setleri de önemli miktarda dönme ve arkaplan karmaşıklığı içermektedir.

Takip algoritmalarının performansları üç farklı ölçüt aracılığı ile niceliksel olarak değerlendirildi. Bu ölçütler, bir algoritmanın takip koruma skorunu, başarı oranını ve konumlandırma hassasiyetini ölçmektedir. Deney sonuçları, ortak-farka dayalı takip algoritmasının, gözetleme veri seti için en yüksek başarı oranı ve konumlandırma hassasiyetine, hareketli hücre dataseti için en yüksek takip koruma oranına sahip olarak en başarılı algoritmalar arasında olduğunu göstermiştir.

Anahtar sözcükler: Nesne Takibi, Ortak-Fark Matrisi, Kovaryans Matrisi, Video İşleme, Görüntü İşleme.

Dedicated to my beloved family...

Acknowledgement

I would like to express my deepest gratitude to my thesis advisor Prof. Dr. Ahmet Enis Çetin for his enlightening guidance, encouragement and for his continuous support in the development and completion of this study.

I would also like to thank to my thesis committee members, Assoc. Prof. Serdar Kozat and Prof. Dr. Rengül Atalay for their valuable comments and suggestions.

I offer my sincere thanks to Dr. Maria Carla Parrini from Institute Curie, Paris for her guidance about cell biology.

I express my gratitude to TÜBİTAK BİDEB "National Scholarship Program for MSc Students".

I would like to thank to ASELSAN for supporting me during my Master's studies. I also thank my colleagues in ASELSAN for their valuable feedbacks and suggestions.

A special gratitude and love goes to my family; my father (Mehmet Demir), my mother (Emine Sibel Demir) and my brother (Okay Demir) for their unconditional support and belief in me. Finally, I want to express my deepest love and thanks to my beloved wife, Nida Demir, for her support and patiance.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Visual object tracking has been one of the widely studied problems in computer vision field. Surveillance systems [1], human-computer interaction [2, 3], autonomous driving [4, 5] and computerized assistance systems in medical image processing [6–9] are among the numerous application areas. Although there are various methods proposed for the problem [10–14], it is hard to say that one single algorithm can cover all the problems in various scenarios. The task involves many application dependent challenges that need to be solved such as rotation, shape deformations, scale and illumination changes. Therefore, the performance of the tracking algorithms may vary dramatically depending on the scenario and the video database used in the scenario. In this study, we propose a novel visual object tracking algorithm and compare its performance with various state-of-the-art tracking methods in two different application areas.

1.1 Problem Statement and Motivation

In this thesis, our objective is to develop an algorithm to track a visual object throughout an image sequence where only the bounding box covering the object for the initial frame is given. The problem is known as short term single object

tracking where trackers are supposed to have no prior information from a pre-learned appearance model for the object being tracked. The algorithms may have update mechanisms for adapting the appearance changes on the object. However, for the online learning of an appearance model during tracking, only a single label from the initial frame is provided. There is no other positive or negative samples provided to the tracker explicitly.

In recent years, region covariance features have been used for different applications such as object detection [15], classification [16] and tracking [17]. Although region covariance is a successful descriptor and efficient approach when compared to most other feature based methods, its computational complexity is still high for the systems with restricted processing power. A more efficient alternative to covariance matrix, the so-called co-difference matrix, was proposed in [18] and used in various applications [19].

In this thesis, we employ the co-difference matrix in the visual object tracking problem and compare its performance with covariance matrix method as well as other recent state-of-the-art trackers [20–29].

Surveillance systems and biomedical studies are among the main fields where visual object tracking applications are utilized. Therefore, we decided to build a framework where we can quantitatively evaluate the performance of our algorithm on datasets related to these fields. For our experiments we used nearly 70000 annotated images in which we compare the performance of our proposed algorithm with other recent trackers. In the next sections, we briefly summarize the application areas that we study in this thesis.

1.2 Tracking Applications in Surveillance

Video surveillance systems have been utilized for monitoring critical areas since 1960's. We can divide the history of video surveillance systems into three generations [1, 30].

The first generation video surveillance systems (1960-1980) consisted of analog video outputs collected from different analog cameras and videos were displayed to a human observer. They required large bandwidths and storage spaces. Real time detection of unusual events purely depended on the operator's abilities. Furthermore, offline scanning of the past events were not easy due to the large quantity of the data.

In the second generation surveillance systems (1980-2000), basic video processing algorithms such as motion detection started to be utilized for assisting the operator. The videos were still being transmitted to a center and processed by a central unit.

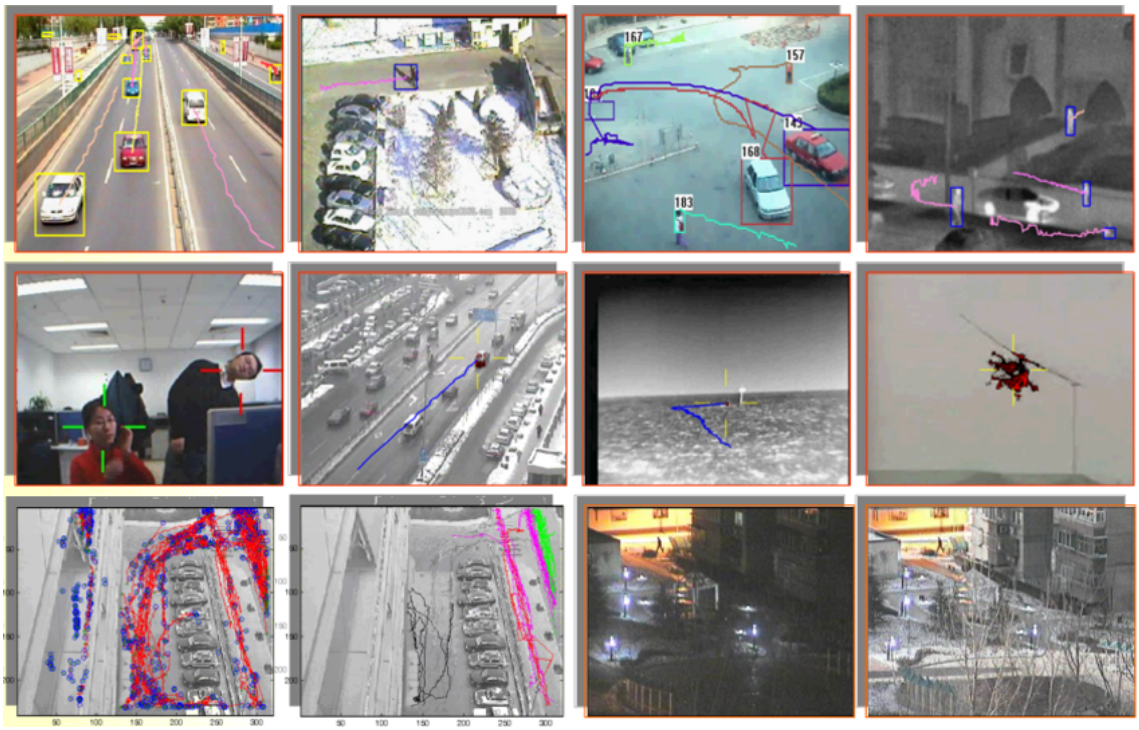


Figure 1.1: Visual object tracking is utilized in surveillance systems for various applications

With the advent of technology, the third generation surveillance systems (2000-) started to employ more complex video processing algorithms with the goal of minimizing the need for human attention. In order to decrease the bandwidth

requirements, surveillance systems started to utilize sensor level image processing applications. Thus, the need for transmitting the whole data is replaced with transmitting the results of intelligent decisions given in the sensor level. However, processing power limitations on these sensors require more efficient and robust algorithms.

Object tracking methods for surveillance applications in the literature generally focus on visual spectrum [10, 11, 13]. On the other hand, decline in the cost of infrared (IR) sensors turned IR cameras into a valuable option for surveillance applications. As the surveillance systems started to utilize IR cameras more and more common, a need for targeting IR specific challenges has emerged. Even if some studies specifically address the issue [20, 31], visual object tracking in IR spectrum, especially with a restricted computational power, presents a challenging task that needs to be studied.

Since surveillance applications mostly require real-time processing, efficiency of the algorithm must be one of the major concerns. Memory, processing power and energy consumption requirements become especially important in embedded platforms located in sensor suites. Efficient implementation of our method due to its multiplier-less nature makes it a good candidate for surveillance applications. Therefore, we included IR spectrum surveillance scenarios in our comparison framework and evaluate the performance of our method on IR datasets containing relevant scenarios.

1.3 Tracking Applications in Cell Motility

Visual object tracking task plays a key role in dynamic cell behavior studies where the migration analysis of cell populations has a significant place [32]. Cell migration is a fundamental process in regular tissue development and recovery [33]. Speed, direction and morphological changes of the cell during the movement are closely related to the structure of the environment [34]. In order to move through extracellular spaces or over the surfaces of other cells, special mechanisms

are employed by the individual cells [35]. These motility patterns are investigated using the microscopic image sequences in a wide range of cell types with different morphological properties. Some of the applications include red blood cell speed measurement [36], cancer cell tracking [37], Bovine Pulmonary Artery Endothelial (BPAE) cell motility tracking [38], leukocytes tracking [39, 40] and embryo cell tracking [38].

In recent years, different approaches have been proposed for such analyses [41, 42]. Benchmarks for comparing various methods exist for fluorescent microscopy [43,44]. However, very few studies focus on analyzing images taken by differential interference contrast (DIC), phase contrast or other label-free microscopy, which are used commonly for observing living cells [45]. In this thesis, we use DIC microscopy images for comparing the cell tracking performance of our algorithm with other state-of-the-art trackers.

Label-free microscopic images (especially in DIC microscopy) are usually have low contrast gray scale images with deformable cell shapes, as demonstrated in Figure 1.2.

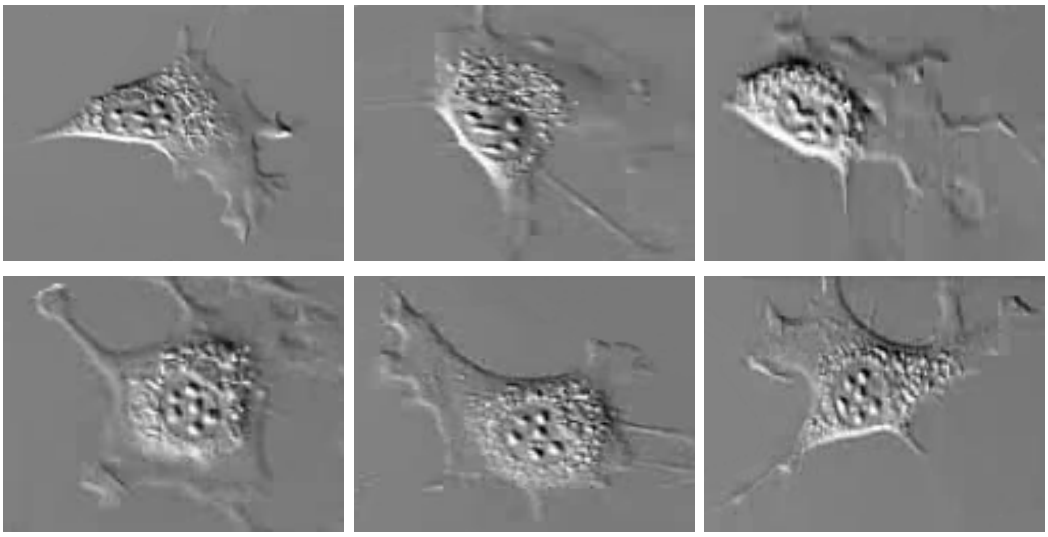


Figure 1.2: An example time-lapse differential interference contrast microscopy image sequence with a challenging object rotation and deformation scenario. The shape of a sample cell changes over time as shown in image frames 1,45,63,97,160,208 and 230, respectively.

Due to this property of DIC microscopy, automatic cell tracking becomes a hard task. In addition to low contrast, there exist several other challenges in cell tracking including;
1) Similar morphological structure of the cells makes it difficult to differentiate one cell from another in dense scenes.
2) Shape deformations and random rotations during the cell motion require adaptive models which are robust to these changes.

In this thesis, we present a comparative study which evaluates the robustness of our algorithm against these specific challenges in cell tracking scenarios.

1.4 Scope and Organization of the Thesis

In this thesis, we focus on short-term tracking scenarios where the only information given to a tracking algorithm is the object bounding box for the initial frame. The methods are evaluated based on their tracking performance until the end of the image sequence. Initialization method of the bounding box or reinitialization after a track loss is not within the scope of this study.

The organization of the thesis is as follows;

In the second chapter, we summarize the different visual object tracking approaches in the literature and introduce the well-known algorithms that we use for our comparative study.

In Chapter 3, we first explain the covariance features and its use in different computer vision applications. Then we introduce the co-difference matrix and give the details of the proposed tracking method with the parameters used for the tracking task.

In Chapter 4, we introduce the properties of the datasets we use for our comparison. Then, we explain the metrics that are utilized for quantitative evaluation. In the same chapter, we also demonstrate the results of experiments for two

different application areas.

In Chapter 5, we conclude the thesis with the final remarks and present ideas for future work.

Chapter 2

Visual Object Tracking

2.1 Background Information

Visual object tracking algorithms are typically grouped under two main categories: generative and discriminative tracking approaches.

Generative methods form an appearance model for representing the target and search for the closest match in the next frames. In general, these approaches are preferred for their computational efficiency. Normalized cross correlation can be considered as a very simple example for understanding this model. By using the image patch as the appearance model directly and finding the object location in the next frames by maximizing the normalized cross correlation result gives a simple generative tracking method. More sophisticated methods exploit the idea of maximizing the correlation output and aims to find the optimum filter for this task [20, 26, 46]. IVT [29] is an example generative method which uses eigenbasis representation for the appearance model and updates the eigenbasis vectors model using an incremental PCA algorithm. In [47], object state is modeled by combining different appearance and motion models. Correlation based generative methods are also popular for their efficiency. In [26], the authors propose a correlation based tracking algorithm where they aim to find the optimal filter for

a desired correlation output. In some generative methods, L1-norm minimization is used for sparse representation of the object. For example, [25] utilizes a fast numerical solver for an L1- norm related minimization to model the object appearance by using a sparse approximation over a template set.

Steps of an example correlation based tracking algorithm are demonstrated in Figure 2.1. As the figure shows, generative methods form an appearance model for the given initial image patch. Then, in the next frame, they search for the best match with respect to an objective function. For the correlation filter case, object location is selected as the place where the correlation result has its maximum value. After finding the new position of the object, the model is updated to adapt the appearance changes on the object. This process is repeated in each frame.

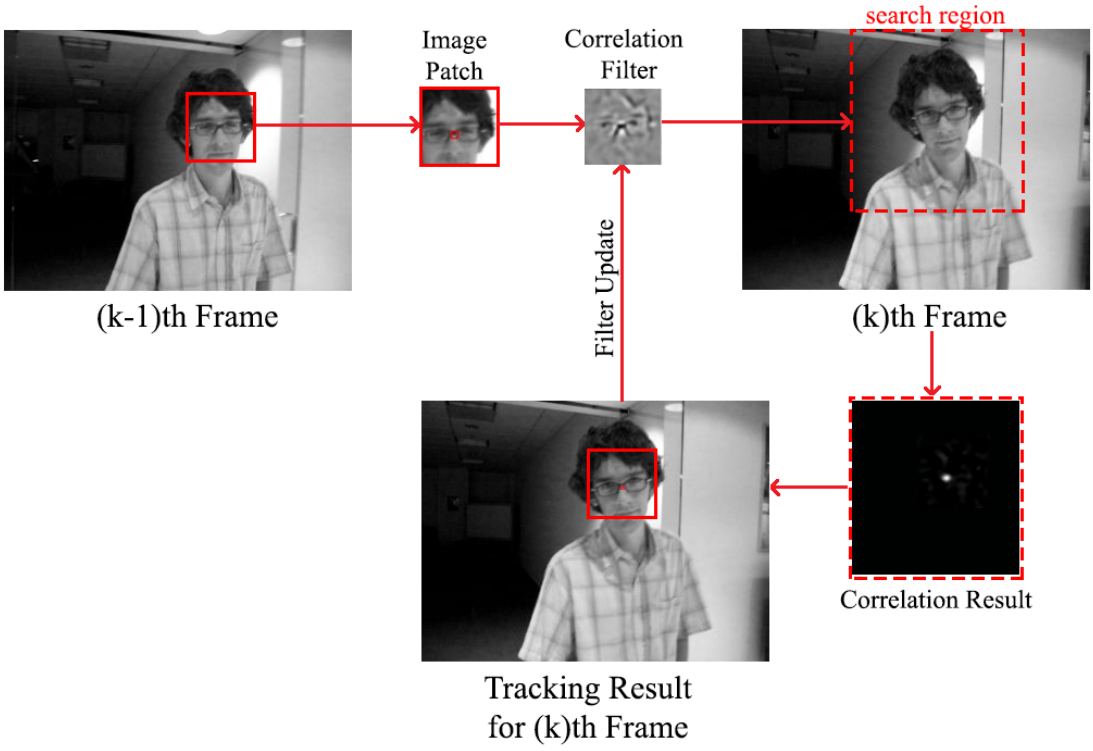


Figure 2.1: Algorithm flow of a correlation filter based tracker

As it is seen from the process, main objective of the generative models is to achieve the best appearance model that represents the foreground object for

localizing it in the next frames.

Discriminative methods, on the other hand, train a classifier to separate the object and background, and approach the tracking task as a binary classification problem. These methods are also referred as track-by-detection methods. They collect samples from background and foreground to form a classifier using visual descriptors such as HOG features [23,27,48]. MILTrack algorithm is an example of this approach [21]. It utilizes HOG features and creates a classifier by combining weak classifiers. In [22], MILTrack algorithm is further improved by employing a feature selector. In [23], a track-by-classification method is proposed in the compressed domain where the random projection idea is used. In [24], the authors employ a kernelized structured output support vector machine (SVM) to make a classification.

Steps of a discriminative method is depicted in Figure 2.2. As the figure summarizes, the method firstly collects sample patches from background and foreground for the given initial frame. It uses these samples for training the classifier. After training process, it classifies the candidate patches in the next frame and selects the best match among the positive outputs. After finding the current object location, it collects new samples from object and background, and continue training the classifier for adapting the foreground and background appearance changes.

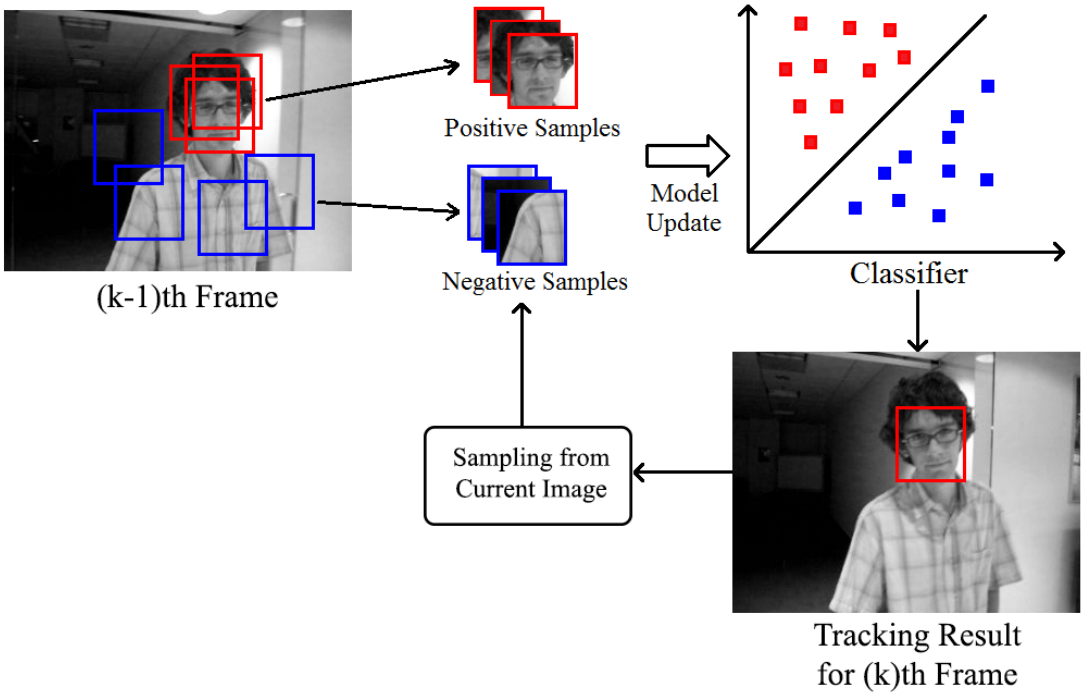


Figure 2.2: Algorithm flow of a discriminative object tracker

For our experiments, we included well-known object tracking methods in the literature from these two different approaches. Most of the algorithms we used have publicly available codes. For all the methods, we used default parameters suggested by the authors. Brief summary of the algorithms are given in the next subsections.

2.2 Compared Algorithms

2.2.1 Discriminative Scale Space Tracker (DSST)

In [46], the MOSSE tracker [26] is extended with a robust scale estimation. In this method, a one-dimensional discriminative scale filter is used for estimating the target size. Another contribution of the paper is employing a pixel-dense representation of HOG-features in combination with the intensity features used

in the MOSSE tracker for translation filter. (Source code available at [49])

2.2.2 Fast Compressive Tracking (FCT)

Zhang et al. used a classification based approach in compressed domain for object tracking. In this approach, they firstly extract features from multi-scale image feature space [23]. Then, using a sparse measurement matrix, they calculate the compressed features that preserve the structure of image feature space. They use the same measurement matrix for compressing the foreground and the background samples. Thus, the tracking task is converted into a binary classification problem that will be solved with a naive Bayes classifier with online update in the compressed domain. (Source code available at [50])

2.2.3 Incremental Learning for Robust Visual Tracking (IVT)

In [29], Ross et al. presented a method that uses a low dimensional subspace representation of the target object for tracking purpose. Proposed method employs an incremental PCA algorithm for adapting the appearance changes by updating the eigenbasis vectors incrementally. (Source code available at [51])

2.2.4 Structured Output Tracking with Kernels (STRUCK)

In [24], Hare et al. propose a tracking by detection method that adaptively estimates the translation of target object via structured-output SVM. Most tracking by detection algorithms firstly generate a set of training examples by sampling the image and assigning binary labels without giving different weights to different training samples. Then, they train the classifier and estimate the location of the object based on the result of the classifier. In the proposed method, however, rather than employing a binary classifier, a prediction function which finds the

translation of the object between frames is trained via structured-output SVM. When updating the prediction function, both the image samples and associated translation information is provided that allows the model to link the tracking and learning tasks. (Source code available at [52])

2.2.5 Kernelized Correlation Filter Tracker (KCF)

In [27], Henriques et al. used a Kernelized Correlation Filter that operates on HOG features. The key idea is to use all the cyclic shift versions of the target patch for training the classifier, instead of using dense sliding windows. Each training sample is assigned with a score generated by a Gaussian function depending on the shift amount. Using the advantages of circulant structure, the classifier is trained in Fourier domain efficiently. (Source code available at [53])

2.2.6 L1 Tracker using Accelerated Proximal Gradient Approach (L1APG)

Bao et al. employed the idea of modeling the target by using a sparse approximation over a template set [25]. In this approach, they solve an ℓ-1 norm related minimization for many times to achieve the sparse representation. Although this approach was used for object tracking successfully in the past, the main drawback was the demanding computational power requirement. In contrast to other ℓ-1 trackers, Bao uses a fast numerical solver that has a guaranteed quadratic convergence. Moreover, they claim that the tracking accuracy is also improved by including an ℓ-2 norm regularization on the coefficients associated with the trivial templates.(Source code available at [54])

2.2.7 Multiple Instance Learning Tracker (MILTrack)

Babenko et al. [21] utilizes the multiple instance learning framework for object tracking, where image patches are bagged into positive and negative sets to discriminate the target from background. MILTrack method uses Haar-like features for representing the image patches. Target and background samples are, then, discriminated by using a boosting based algorithm where a set of weak classifiers are combined to make a classification decision. (Source code available at [55])

2.2.8 Minimum Output Sum of Squared Errors Tracker (MOSSE)

Correlation based approaches are widely used for object tracking especially for their computational efficiency. MOSSE is an adaptive correlation based algorithm that calculates the optimal filter for the desired Gauss-shaped convolution output [26]. The method has an update mechanism that adaptively changes the correlation filter depending on the target shape. This method has the lowest computational burden amongst the compared algorithms. (Source code available at [56])

2.2.9 Online Discriminative Feature Selection Tracker (ODFS)

In [22], an online discriminative feature selection approach is proposed where the classifier score is coupled with the importance of the patch samples. ODFS employs an feature selection mechanism where the features that optimize the objective function in steepest ascent direction for positive samples and steepest descent direction for negative samples are selected. (Source code available at [57])

2.2.10 Spatially Regularized Discriminative Correlation Filter Tracker (SRDCF)

Discriminatively learned correlation filters (DCF) utilize a periodic assumption of the training samples to efficiently learn a classifier on all patches in the target neighborhood. The main contribution of [48] is mitigating the problems arising from assumptions of periodicity in discriminative correlation filters by introducing a spatial regularization function that penalizes filter coefficients residing outside the target region. By selecting the spatial regularization function to have a sparse Discrete Fourier Spectrum, the filter is efficiently optimized directly in the Fourier domain. For the classification of the candidate patches, SRDCF employs HOG and gray-scale features giving a 42 dimensional feature vector at each 4x4 HOG cell. (Source code available at [58])

2.2.11 Sum of Template and Pixel-wise Learners (Staple)

In order to construct a model that is robust to intensity changes and deformations, [59] combines two different image patch representations which are sensitive to complementary effects. Correlation based algorithms have robust results on illumination change scenarios, but they are sensitive to deformations because of their dependency on the object shape. Color-based approaches, on the other hand, handle shape variations well, but their dependency on color hurts the performance on illumination changes. This study combines the translation results of two approaches in a weighted manner based on their reliability scores to achieve a higher accuracy. (Source code available at [60])

2.2.12 Ensemble of MOSSE Trackers (TBOOST)

In [20], an ensemble based object tracking method is proposed. This algorithm creates and updates an adaptive ensemble of simple correlation filters and generates tracking decisions by switching among the individual correlators in the

ensemble depending on the target appearance in a computationally highly efficient manner.

Chapter 3

Co-difference Matrix and Object Tracking in Video

Before we get into the details of co-difference based object tracking method, we first review the region covariance based feature extraction from videos. In recent studies, covariance features have been utilized for various vision applications such as object detection [15], classification [16] and tracking [17]. Region covariance is a successful image descriptor and efficient approach when compared to most other feature based methods. However, its computational complexity is still high for the systems with restricted processing power. A more efficient alternative to covariance matrix, so-called co-difference matrix, was proposed [18] and used in different applications [19, 61]. In our study, we employed co-difference features for visual object tracking task.

In the next sections, we first explain the extraction of covariance features. Then, we introduce the co-difference matrix and its use as image descriptor. Finally, we describe the co-difference based object tracking method.

3.1 Covariance Features

Given a two dimensional intensity image I, let R be a rectangular sub-window consisting of N pixels and let $(\mathbf{f_k})_{k=1...n}$ be the d-dimensional feature vectors in R. These features can be intensity, image gradients, edge responses, high order derivatives etc. Then, we calculate the covariance matrix for region R as follows:

$$\mathbf{C_R} = \frac{1}{N-1}\sum_{k=1}^{N}(\mathbf{f_k} - \mu_\mathbf{R})(\mathbf{f_k} - \mu_\mathbf{R})^T \tag{3.1}$$

where μ_R is the d-dimensional mean vector of the features calculated in region R. The covariance matrix is a symmetric positive-definite matrix with a matrix size of d-by-d.

Extraction process of covariance descriptor is demonstrated in Figure 3.1. Firstly, a set of features that will represent the given image patch are selected. In the figure, pixel luminance, color intensities, vertical and horizontal derivatives are depicted as visual features. After calculating these features for a given image patch, resulting matrices are reshaped and concatenated. For an image patch that contains N pixels, we obtain a N-by-d concatenation result where d is the number of features. After this step, covariance of this matrix is calculated where the result is a d-by-d matrix that will be used as the image descriptor.

Although it seems a convenient way to fuse information coming from different features, its computational cost is relatively high due to multiplications especially for large regions.

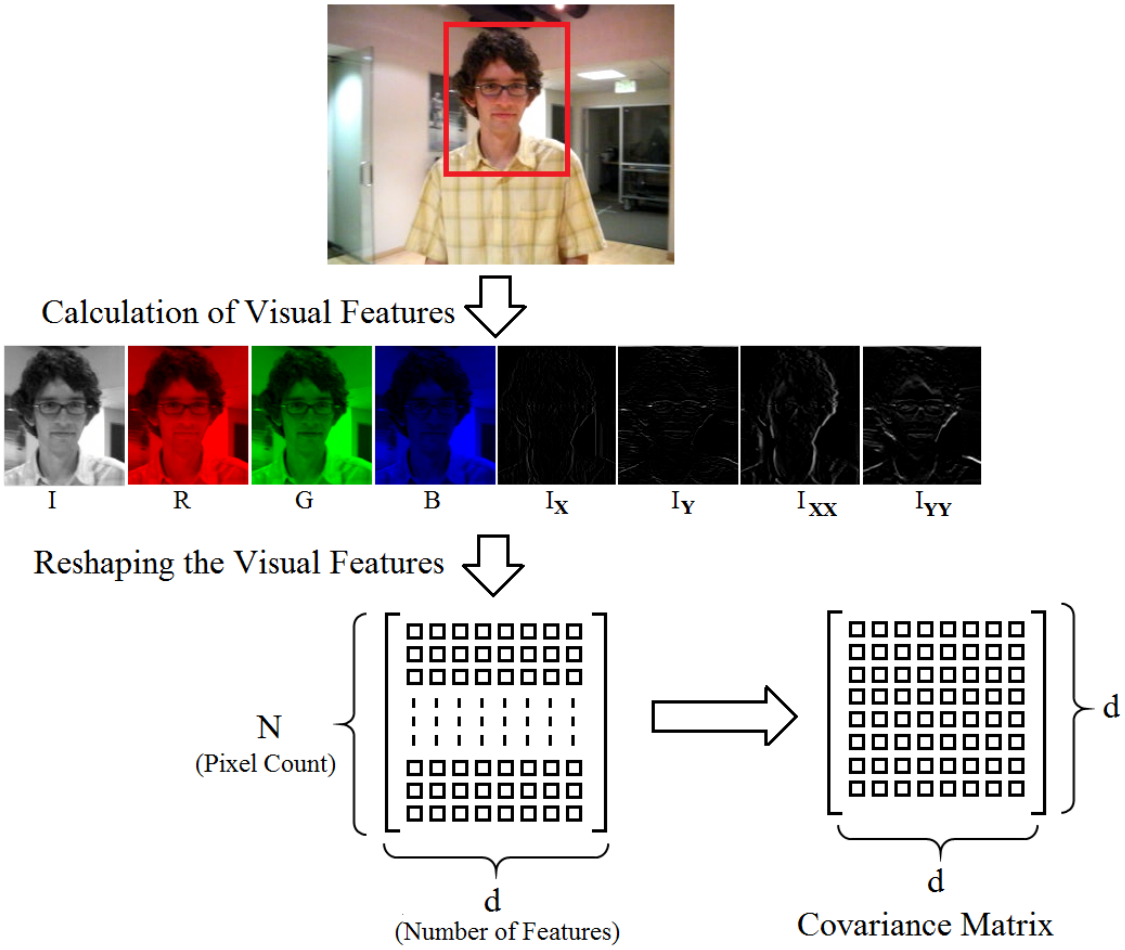


Figure 3.1: Calculation of covariance matrix

A more efficient method for extracting an image descriptor is co-difference matrix which can be calculated without any multiplication. This method is explained in the following section.

3.2 Co-difference Features

In [18], a new efficient method is introduced for calculating the "covariance-like" descriptors. The main difference that boosts the performance is the

multiplication-free nature of the method. Instead of the multiplications in covariance method, this implementation uses an operator based on additions. Let a and b be two real numbers. The new operator is defined as follows:

$$a \oplus b = \begin{cases} a + b & \text{if } a \geq 0 \text{ and } b \geq 0 \\ a - b & \text{if } a \leq 0 \text{ and } b \geq 0 \\ -a + b & \text{if } a \geq 0 \text{ and } b \leq 0 \\ -a - b & \text{if } a \leq 0 \text{ and } b \leq 0 \end{cases} \tag{3.2}$$

which can also be expressed as;

$$a \oplus b = sign(a \times b)(|a| + |b|) \tag{3.3}$$

This operator basically performs a summation operation, but the sign of the result is the same as the multiplication operator. In [19], it is stated that the co-difference descriptor can be calculated about 100 times faster than the covariance matrix in some processors. Using the operator defined in (3.2), a new vector product of two vectors $\mathbf{x_1}$ and $\mathbf{x_2}$ of size N is given as;

$$< \mathbf{x_1}, \mathbf{x_2} >= \sum_{i=1}^{N} x_1(i) \oplus x_2(i) \tag{3.4}$$

where $x_k(i)$ is the i-th entry of the vector $\mathbf{x_k}$. Now , we can define the co-difference matrix for a region R as follows;

$$\mathbf{C_d} = \frac{1}{N-1} \sum_{k=1}^{N} (\mathbf{f_k} - \mu_\mathbf{R}) \oplus (\mathbf{f_k} - \mu_\mathbf{R})^T \tag{3.5}$$

3.2.1 Co-difference Features for Visual Object Tracking

We employed a generative approach in which we use the co-difference matrix given in Equation 3.5 as the region descriptor for our visual tracking algorithm. In our implementations, we defined the feature vector as

$$\mathbf{f_k} = [x(k)\ \ y(k)\ \ I(k)\ \ I_x(k)\ \ I_y(k)\ \ I_{xx}(k)\ \ I_{yy}(k)] \tag{3.6}$$

where the elements of the feature vector are horizontal and vertical pixel positions within the region, intensity, gradients in both directions and second derivative values in both directions, respectively. Calculation of co-difference based descriptor is demonstrated in figure 3.2.

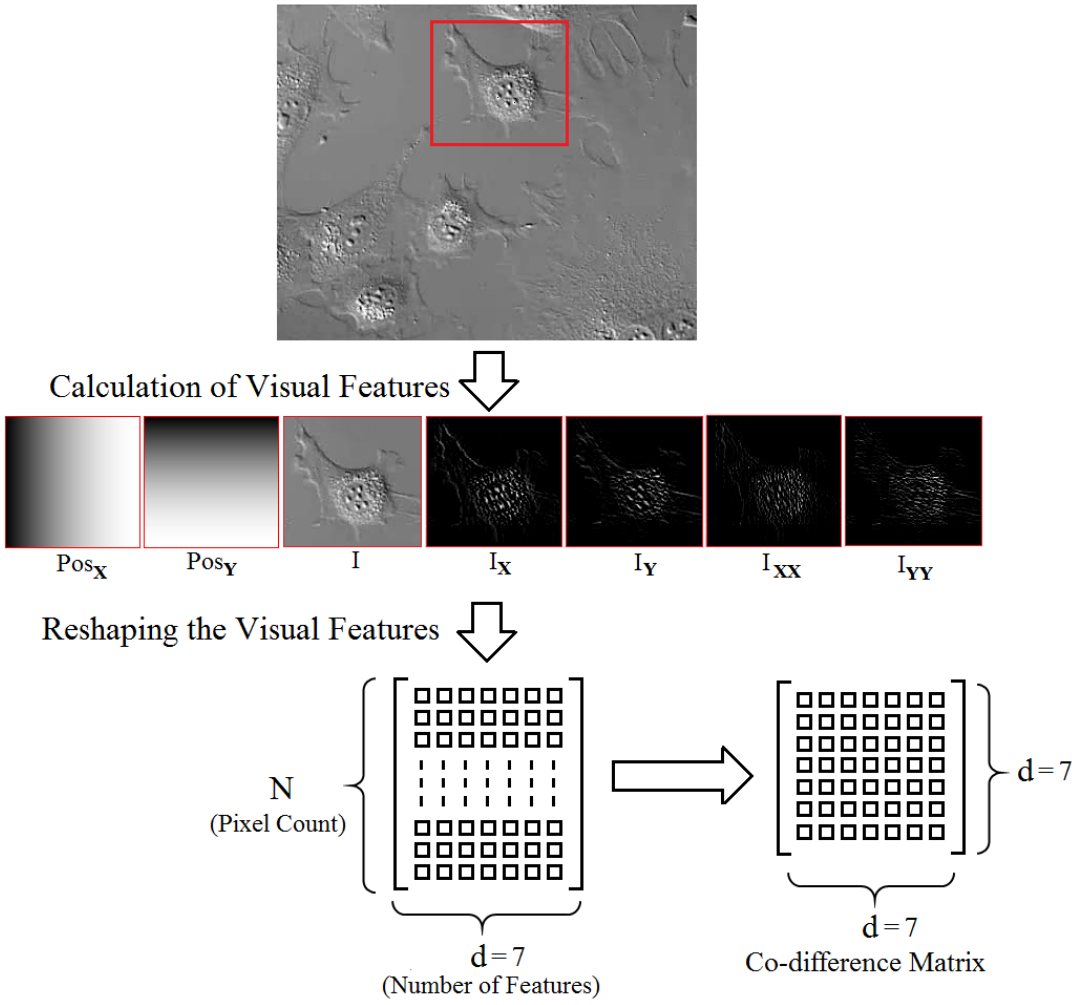


Figure 3.2: Calculation of co-difference matrix

As it is seen from the figure, each pixel in the region is represented by a 7-dimensional feature vector. As a result, we calculate a 7x7 co-difference descriptor. The co-difference matrix is symmetric as the covariance matrix.

The co-difference matrix has advantages similar to that of covariance matrices as region descriptors. The co-difference matrix has a natural way of combining multiple features without normalizing features or using blending weights. It contains the information embedded within the histograms as well as the information that can be derived from the appearance models.

In general, a single co-difference matrix extracted from a region is enough to match the region in different views and poses. The noise corrupting individual samples are largely filtered out because of the averaging operation during co-difference computation. The co-difference matrix of any region has the same size, thus it enables comparing regions without being restricted to a constant window size. It also has a scale invariance property over the regions in different images provided that raw features (image gradients and orientations) used during the computation of the covariance matrix are extracted according to the to scale difference. In addition, the co-difference matrix has a robust behavior to rotations because of the averaging. It should be pointed out that the co-difference is invariant to the mean changes such as identical shifting of color values. This becomes an important property when objects are tracked under varying illumination conditions. It is possible to compute the co-difference matrix from feature images in a fast way using "integral" image representations as the covariance matrix [5].

Tracking process using co-difference features is summarized in Figure 3.3.

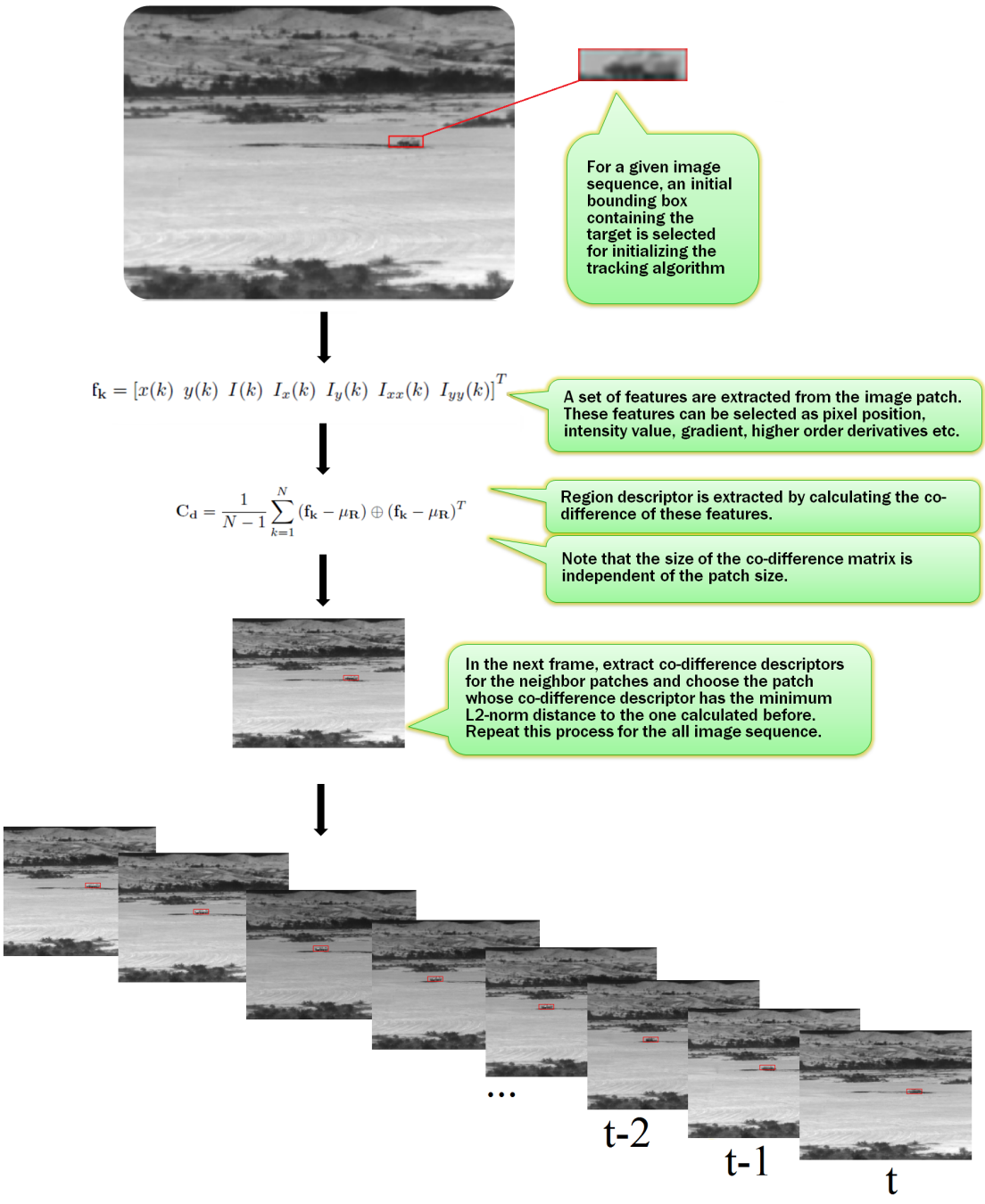


Figure 3.3: Summary of co-difference based object tracking algorithm

As the figure demonstrates, after extracting the descriptor for the object patch, we advance to next frame and search for the closest match. To obtain the most

similar region to the given object, we need to compute distances between the co-difference matrices corresponding to the target object window and the candidate regions during object tracking. This can be done by computing the generalized eigenvalues of the current matrix of the target window and the matrices of the target window. The generalized eigenvalue based distance matrix is given by;

$$\rho(C_1, C_2) = \sqrt{\sum_i ln^2 \lambda_i} \tag{3.7}$$

where λ_i are the generalized eigenvalues of the matrices C_1 and C_2.

Although, the covariance and co-difference matrices do not lie on the Euclidean space they can be compared using the arithmetic subtraction of two matrices and computing the Euclidean norm of the difference. We experimentally observed this arithmetic approach gives successful results. Euclidean norm based comparison actually reduces the computational cost of the tracker.

After finding the closest match based on the Euclidean distance, the appearance model is updated using the new co-difference matrix as formulated in Equation 3.8.

$$\mathbf{C_{d_t}} = \alpha \mathbf{C_{d_{t-1}}} + (1 - \alpha)\mathbf{C_{d_{new}}} \tag{3.8}$$

where $\mathbf{C_{d_{new}}}$ is the co-difference matrix calculated in the current frame, $\mathbf{C_{d_{t-1}}}$ is the co-difference matrix before update and $\mathbf{C_{d_t}}$ is the updated co-difference matrix. We picked α parameter as 0.95 for our appearance model.

The covariance matrix is Euclidean ℓ_2 norm based because each entry is the inner-product of two vectors. It is well-known that the inner-product induces the ℓ_2 norm. On the other hand, the co-difference matrix is an ℓ_1 norm based matrix, because the vector-product defined in Equation 3.4 induces the ℓ_1 norm.

$$< \mathbf{x}, \mathbf{x} >= \sum_{i=1}^{N} x(i) \oplus x(i) = 2||\mathbf{x}||_1 \tag{3.9}$$

As a result the co-difference matrix is "sparser" than the covariance matrix. The ℓ_1 norm based methods usually produce better image processing algorithms

[62–65]. This may be the reason why the co-difference matrix produces better tracking results compared to the covariance matrix.

Chapter 4

Experimental Studies

The aim of our experiments is to quantitatively evaluate the performance of our tracker on two different application areas and compare its performance with various state-of-the-art trackers.

For the surveillance application, the algorithms are compared on the IR band image sequences of SENSIAC dataset[1]. For the cell motility application, tracking algorithms are compared on NIKON cell motility videos [66].

The performances of the tracking algorithms are compared using the metrics described in the following section.

4.1 Performance metrics

In all the following experiments, we use two main evaluation metrics, i.e., success and precision rates, used in [10].

The first metric is the success rate which indicates the percentage of frames, in which the overlap ratio between the ground truth and the tracking result is sufficiently high with respect to an appropriate threshold. A success rate plot

[1]SENSIAC: www.sensiac.org

can be generated by varying the overlap threshold between 0 and 1. In order to rank the tracking algorithms based on their success rates, we use the area under curve (AUC) and track maintenance (TM) scores, which are derived from success plots. AUC refers to the total area under a success rate plot and TM is the ability of a tracker to maintain a track, i.e., the percentage of frames where a non-zero overlap ratio is maintained.

The second evaluation metric is the precision value. It denotes the percentage of the frames in which the Euclidean distance between the estimated and the actual target centers is smaller than a given threshold. The precision value demonstrates the localization accuracy (LA) of a given tracking method. In order to rank the algorithms based on their precision value, a distance threshold of 20 pixels is used in Table 4.6.

These two metrics are showed in Figure 4.1 and Figure 4.2.

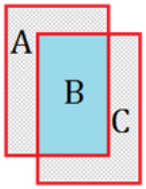


Figure 4.1: First performance measure indicates the percentage of frames where the overlap ratio B/(A+B+C) is higher than a certain threshold. (Region A∪B defines the region covered by ground truth where the region B∪C shows the region defined by the tracking result)

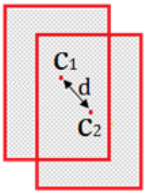
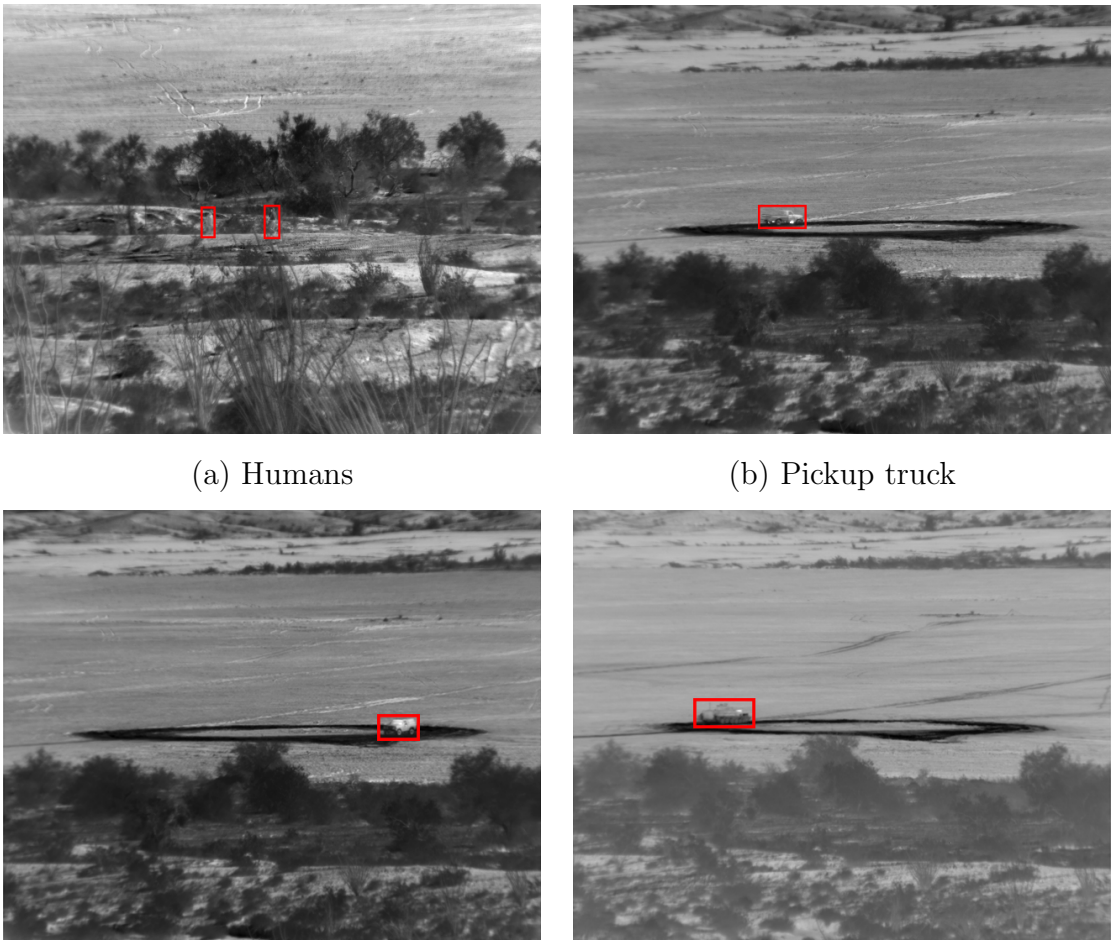


Figure 4.2: Second performance measure indicates the percentage of frames where the distance d between the centers of ground truth and tracking result is lower than a certain threshold.

4.2 Surveillance Experiments

4.2.1 Surveillance Dataset

The SENSIAC dataset includes mid-wave IR image sequences of various scenes containing different types of target objects with different sizes such as walking pedestrians, trucks, tanks and others. A ground truth that defines the bounding box around the target for each frame is also provided. Our experiments are performed on 20 IR image sequences, which contain considerable amount of background clutter, rotation and a few occlusion instances (Figure 4.3).



(a) Humans

(b) Pickup truck

(c) SUV

(d) Tank

Figure 4.3: Example IR image frames from the SENSIAC dataset

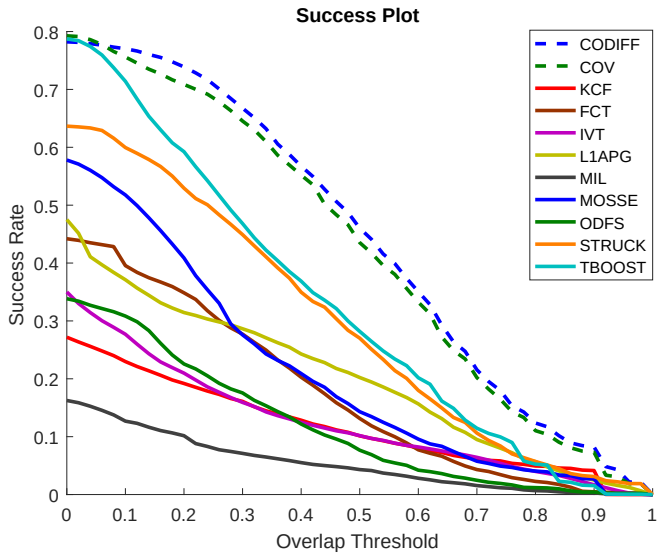
Table 4.1: Video content of the surveillance dataset used in experiments

Video Name	Target Object	Range	Initial Bounding Box Size	
			Horz. (px)	Vert. (px)
Seq 01	Human moving at slow pace	500	41	13
Seq 02	Pickup	1000	19	63
Seq 03	Sport Utility Vehicle	1000	19	37
Seq 04	Armored Personnel Carrier	1000	27	77
Seq 05	Anti-Aircraft Weapon	1000	27	33
Seq 06	Human moving at slow pace	1000	27	7
Seq 07	Pickup	2000	9	31
Seq 08	Sport Utility Vehicle	2000	11	23
Seq 09	Armored Personnel Carrier	2000	13	39
Seq 10	Infantry Scout Vehicle	2000	11	13
Seq 11	Main Battle Tank	2000	13	41
Seq 12	Self-Propelled Howitzer	2000	15	39
Seq 13	Self-Propelled Howitzer	2500	13	31
Seq 14	Human moving at fast pace	2500	9	3
Seq 15	Pickup	3000	7	19
Seq 16	Sport Utility Vehicle	3500	7	11
Seq 17	Armored Personnel Carrier	3500	7	25
Seq 18	Armored Personnel Carrier	3500	9	23
Seq 19	Infantry Scout Vehicle	4000	7	13
Seq 20	Infantry Scout Vehicle	4500	5	13

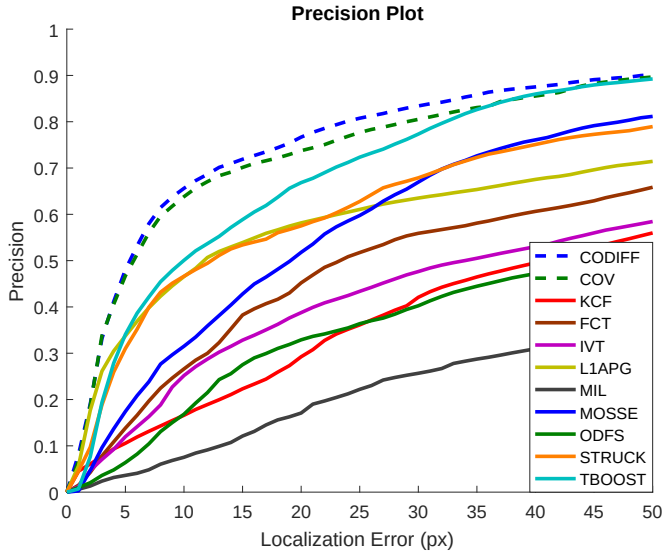
Table above summarizes the content of the surveillance dataset. As it can be seen from the table, videos include scenarios with different target distances and object sizes.

4.2.2 Surveillance Results

Overall performance results of compared visual object trackers are depicted in Figure 4.4 and quantitative results are listed in Table 4.2. Success and precision plots for individual experiments are provided in the Appendix A.



(a) Succes vs overlap threshold plots of compared methods



(b) Precision vs. localization error plots of compared methods

Figure 4.4: Success and precision plots for Surveillance Dataset.

Table 4.2: Success and Precision rate comparison for Surveillance Dataset

	Success		Precision
	AUC	*TM*	*LA*
CODIFF [67]	**0.445**	**0.782**	**0.767**
COV [17]	0.4292	0.793	0.738
TBOOST [20]	0.327	0.787	0.668
STRUCK [24]	0.297	0.637	0.575
MOSSE [26]	0.211	0.578	0.518
L1APG [25]	0.202	0.475	0.581
FCT [23]	0.178	0.442	0.452
IVT [29]	0.127	0.350	0.388
ODFS [22]	0.120	0.338	0.329
CRC [28]	0.119	0.272	0.292
MIL [21]	0.055	0.163	0.171

As we see from the Table 4.2, the proposed co-difference based object tracking algorithm achieves a localization accuracy of 76.68% which is the highest among the compared methods. In success metric, it also gets the best result by having an AUC value of 0.445. When we compare the track maintenance scores, it gets the third highest performance among the compared methods by having a track maintenance score of 78.22%.

When we examine the results, we see that the generative methods have a better performance behaviors in surveillance dataset. This may be caused from the fact that IR spectrum has tendency to contain less texture content than visible spectrum [20]. Lack of texture content may affect the performance of feature based discriminative approaches in a negative manner, since they try to use visual features such as corners and blobs to differentiate the object from background.

Successful results of co-difference and covariance tracking algorithms possibly come from the representation power of these descriptors since they combine a set

of image features.

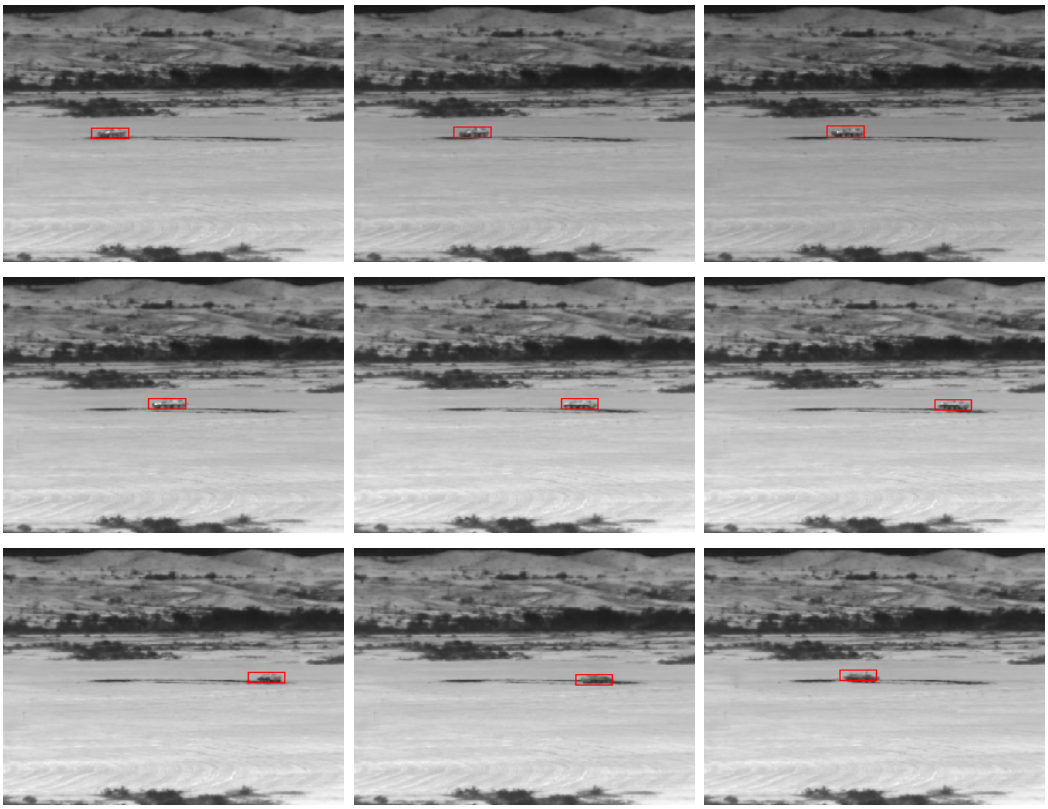


Figure 4.5: Tracking results of the co-difference algorithm for a sample scene, in which significant amount of rotation is present.(Frame numbers from top left to bottom right: 1,33,85,129,234,291,348,545,710)

For a more detailed analysis, we divided the surveillance experiments into three subgroups based on the target distance and present the individual results for these subgroups.

Figure 4.6, 4.7 and 4.8 demonstrate the performance of compared algorithm for three different cases. In the first one, target distance is below 2000m. Compared to other cases, we expect to have larger objects on the image plane, since they are closer to the camera. Success and precision plots in Figure 4.6 show that the performance difference of discriminative and generative approaches is smaller in this set of experiments. This result supports the argument given in the previous paragraph, since visual features become stronger with the increasing object size.

On the other hand, when the distance between the object and camera increases, projection of the target on the image plane gets smaller which results in lack of texture details. In this case, feature based discriminative methods perform worse than their generative counterparts as expected. (see Figure 4.8)

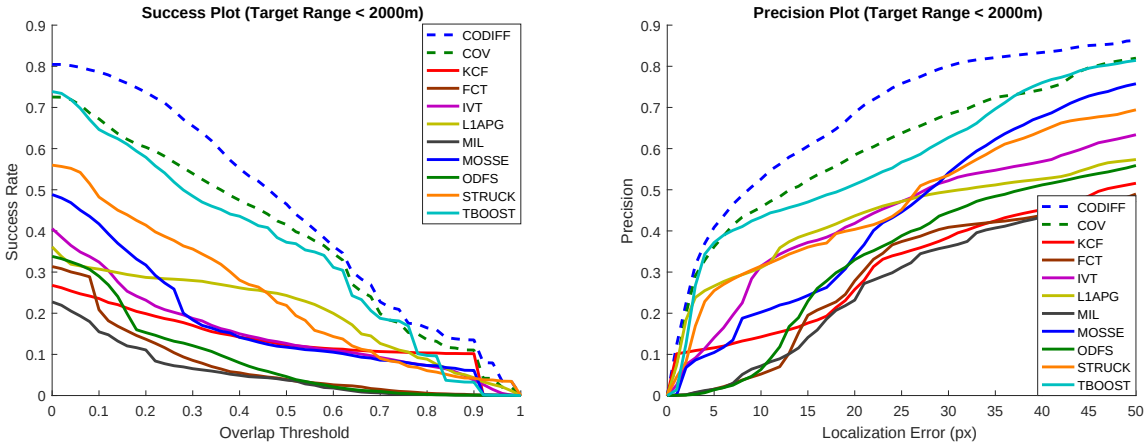


Figure 4.6: Results for the surveillance videos with object distance below 2000m

Table 4.3: Best performing trackers in surveillance videos with target distance below 2000m

	Rank	Seq 01	Seq 02	Seq 03	Seq 04	Seq 05	Seq 06
Success Rate	#1	CODIFF	TBOOST	CODIFF	COV	CODIFF	CODIFF
	#2	COV	CODIFF	COV	CODIFF	COV	TBOOST
	#3	KCF	STRUCK	STRUCK	IVT	STRUCK	COV
Precision	#1	CODIFF	CODIFF	CODIFF	COV	COV	CODIFF
	#2	COV	TBOOST	COV	CODIFF	CODIFF	TBOOST
	#3	TBOOST	STRUCK	STRUCK	L1APG	STRUCK	COV

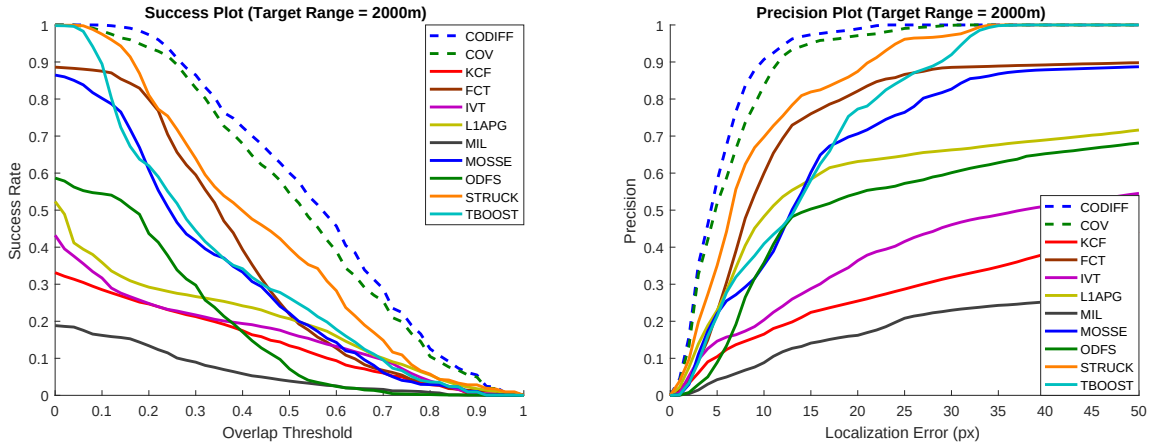


Figure 4.7: Results for the surveillance videos with object distance of 2000m

Table 4.4: Best performing trackers in surveillance videos with target distance of 2000m

	Rank	Seq 07	Seq 08	Seq 09	Seq 10	Seq 11	Seq 12
Success Rate	#1	MOSSE	CODIFF	COV	COV	CODIFF	CODIFF
	#2	CODIFF	COV	CODIFF	CODIFF	COV	FCT
	#3	COV	STRUCK	STRUCK	FCT	STRUCK	COV
Precision	#1	CODIFF	CODIFF	COV	FCT	CODIFF	FCT
	#2	COV	STRUCK	CODIFF	COV	COV	L1APG
	#3	STRUCK	MOSSE	STRUCK	ODFS	FCT	CODIFF

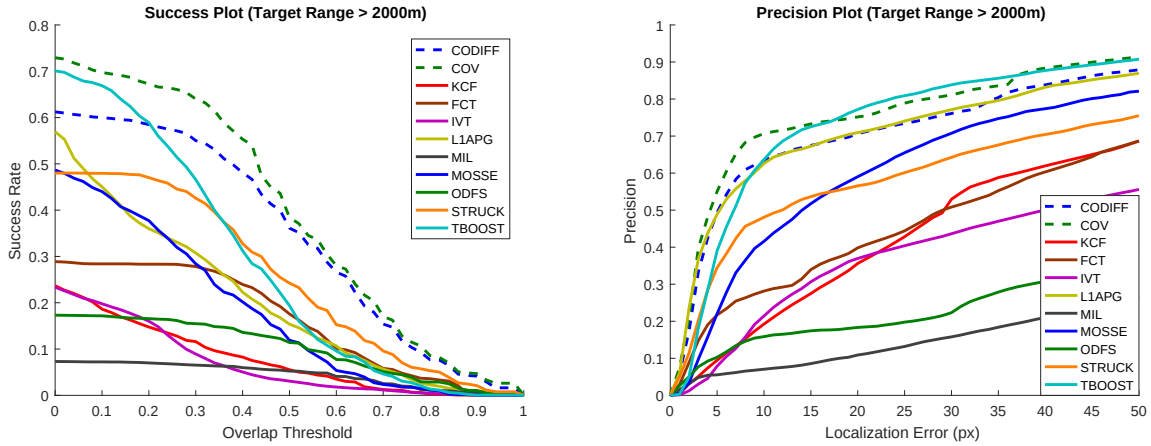
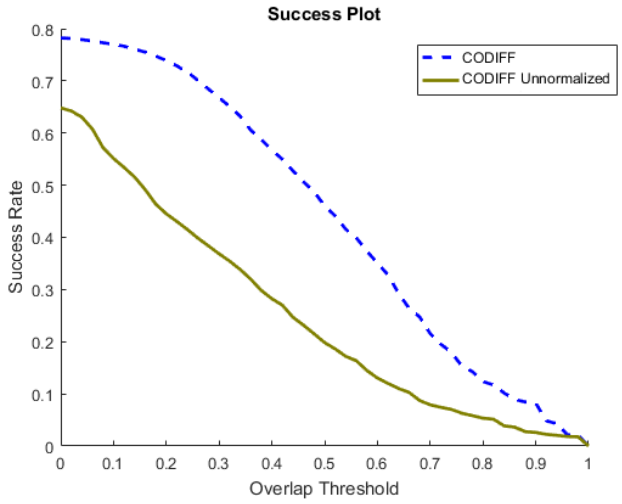


Figure 4.8: Results for the surveillance videos with object distance above 2000m

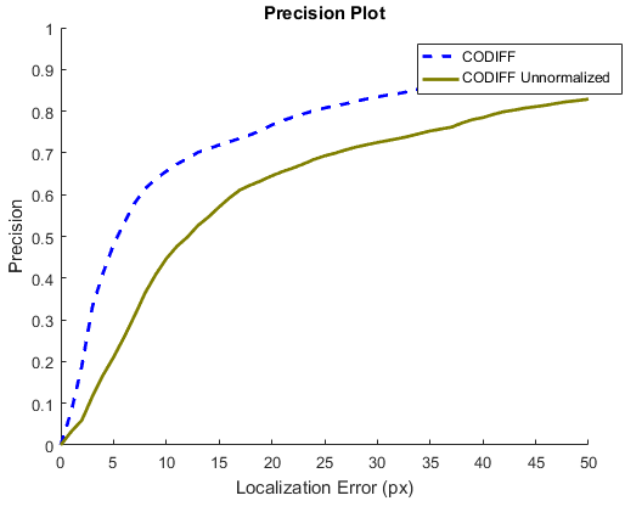
Table 4.5: Best performing trackers in surveillance videos with target distance above 2000m

	Rank	Seq 13	Seq 14	Seq 15	Seq 16	Seq 17	Seq 18	Seq 19	Seq 20
Success Rate	#1	TBOOST	COV	MOSSE	COV	CODIFF	CODIFF	FCT	STRUCK
	#2	STRUCK	L1APG	STRUCK	CODIFF	COV	STRUCK	COV	TBOOST
	#3	COV	CODIFF	L1APG	TBOOST	TBOOST	COV	CODIFF	MOSSE
Precision	#1	TBOOST	L1APG	L1APG	COV	CODIFF	CODIFF	COV	TBOOST
	#2	STRUCK	COV	MOSSE	CODIFF	COV	STRUCK	CODIFF	MOSSE
	#3	COV	MOSSE	COV	TBOOST	TBOOST	COV	FCT	STRUCK

SENSIAC dataset contains high dynamic range images where the image pixels are represented by 14-bit values. We saw that normalizing the pixel values between 0 and 1 increases the performance of our descriptor. The results of co-difference tracking algorithm for both normalized and unnormalized intensity values are given in Figure 4.9. Because of this result, we used normalized intensity values in the co-difference descriptor for all the surveillance experiments reported in the thesis.



(a) Effect of normalization on the succes



(b) Effect of normalization on the precision

Figure 4.9: Effect of normalizing intensity values for infrared surveillance dataset

4.3 Cell Motility Experiments

4.3.1 Cell Motility Dataset

For our cell tracking experiments, we used Nikon cell motility dataset [66]. In order to make a comparison between tracking algorithms, we firstly generated the ground truth data by annotating the cells in each frame, where every cell is considered as a new object. The dataset contains 5 different image sequences and 40 annotated objects that compose nearly 35000 bounding box data in ground truth. The duration between two consecutive frames is 30 seconds. Dataset contains image sequences with challenging rotation and deformation scenarios as well as different object sizes. Various cell image sequences used for evaluation are depicted in Figure 4.10 - 4.14. Brief information about the sequences is given in the following subsections.

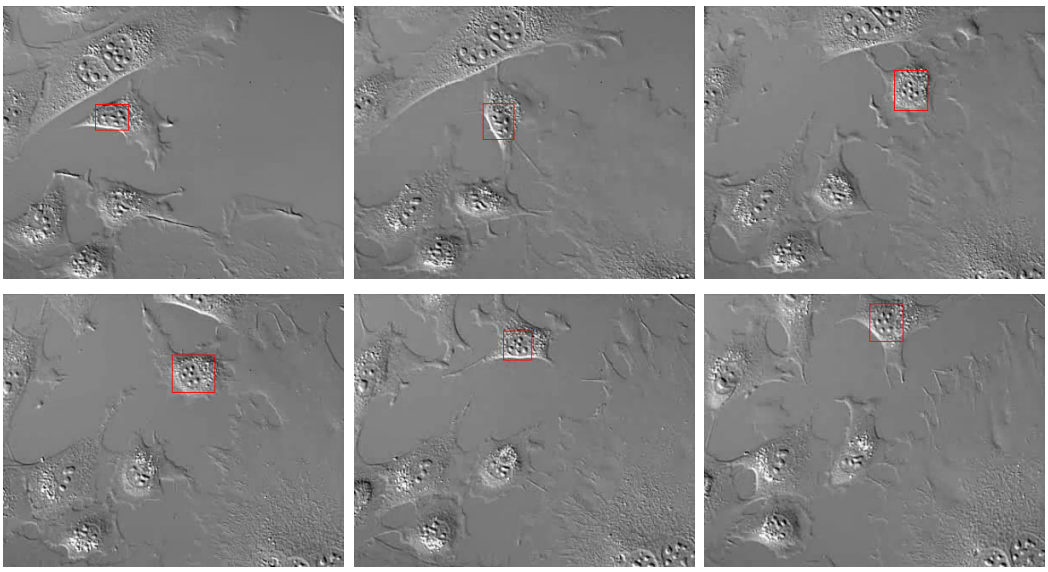


Figure 4.10: Sample images from Albino Swiss Mouse Embryo Fibroblasts (3T3) sequence

4.3.1.1 Albino Swiss Mouse Embryo Fibroblasts (3T3)

The 3T3 cell line was established from Albino Swiss Mouse Embryo tissue. It is a widely utitized fibroblast culture in laboratory research. 3T3 cells helped scientists study the differences between cell's ability to go under oncogenic transformation and cell mortality because 3T3 cells could grow indefinitely, while they are unable to boost the tumor growth.

4.3.1.2 Bovine Pulmonary Artery Endothelial Cells (BPAE)

BPAE cells contain an enzyme that is involved in the maintanace of blood pressure. Because of this reason, they are important in hypertension and coronory heart disease research.

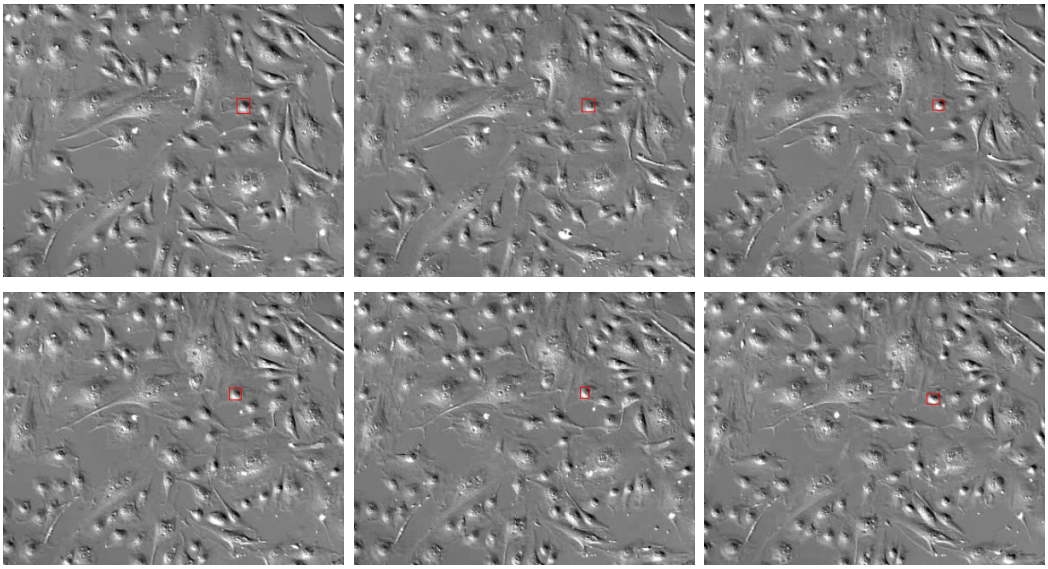


Figure 4.11: Sample images from Bovine Pulmonary Artery Endothelial Cells (BPAE) sequence

4.3.1.3 Rhesus Monkey Kidney Epithelial Cells (LLC-MK2)

LLC-MK2 cells are important since they have been used in the production of mumps vaccines and in the isolation of parainfluenza viruses.

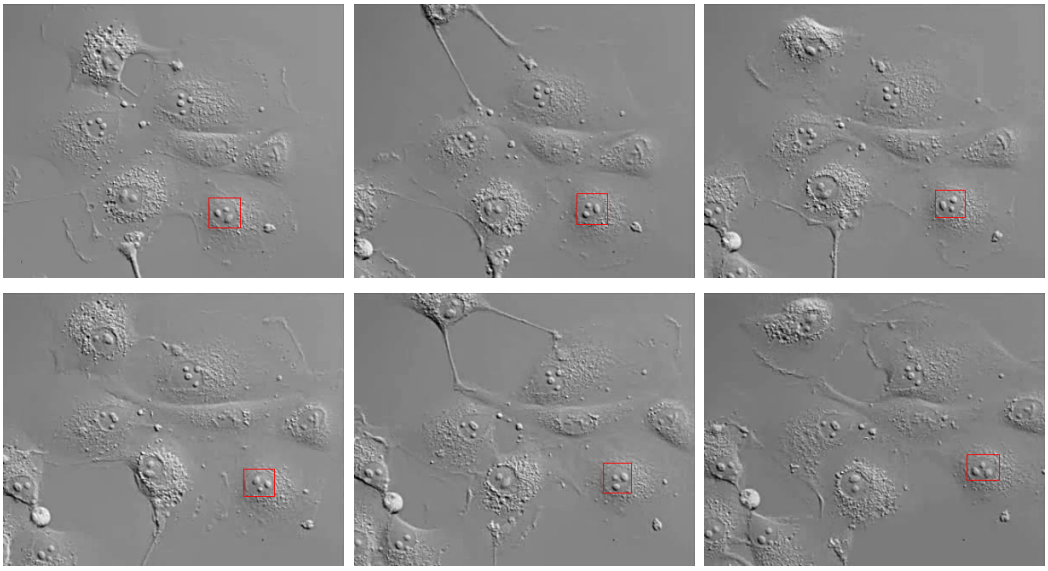


Figure 4.12: Sample images from Rhesus Monkey Kidney Epithelial Cells (LLC-MK2) sequence

4.3.1.4 Human Bone Osteosarcoma Epithelial Cells (U2OS)

The U2OS cell line was cultivated from the bone tissue of a human female suffering from osteosarcoma. Osteosarcoma is the most common type of bone cancer in the world and is the sixth most frequently occurring cancer in children [68]. Behaviour analysis of this cell is, therefore, has a significant place in cancer research.

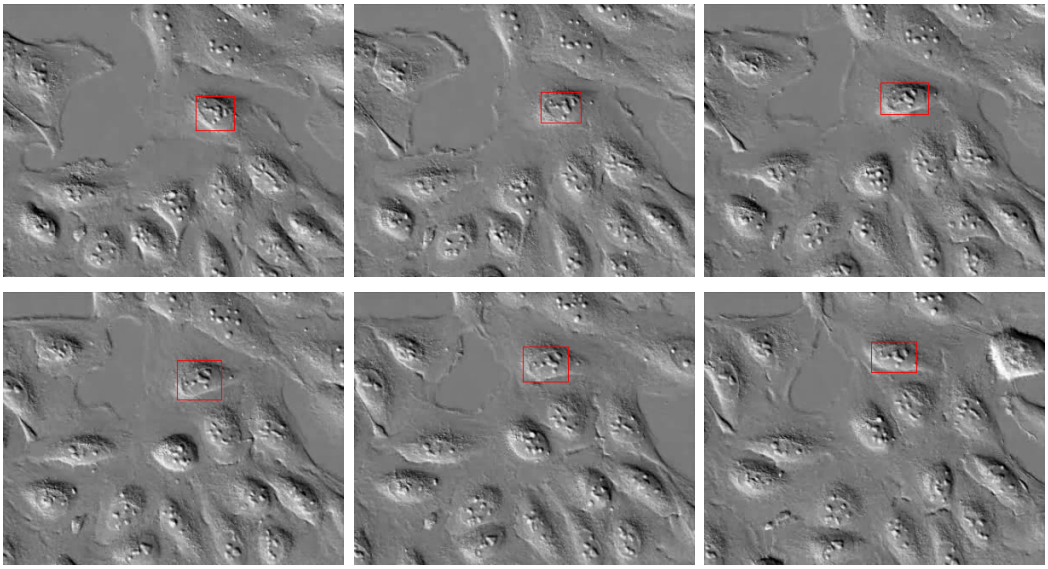


Figure 4.13: Sample images from Human Bone Osteosarcoma Epithelial Cells (U2OS) sequence

4.3.1.5 Embryonic Rat Thoracic Aorta Medial Layer Myoblasts (A-10)

A-10 cells are derived from the thoracic aorta which transports blood from the heart to the other organs and parts of the body.

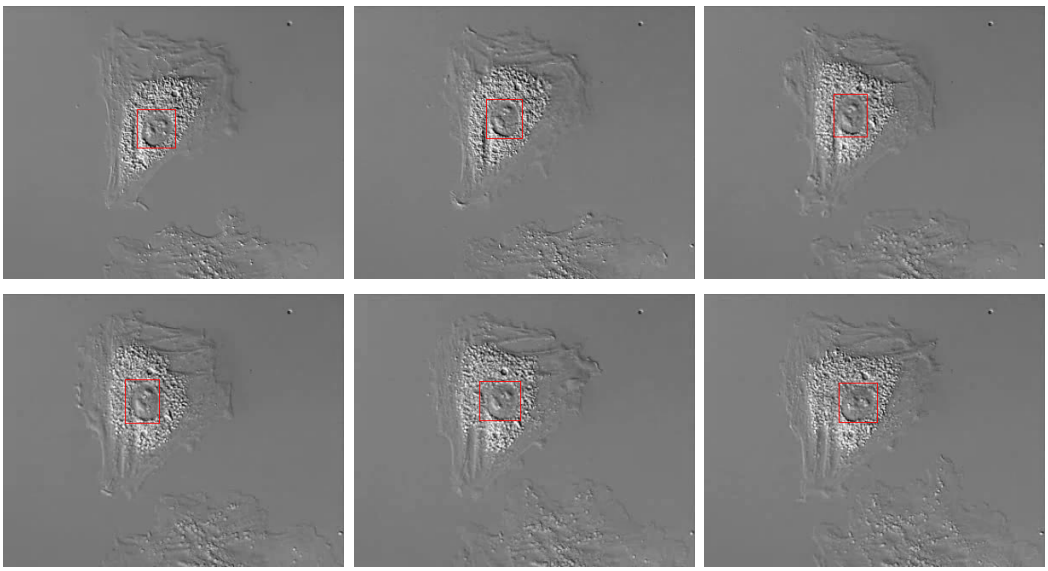


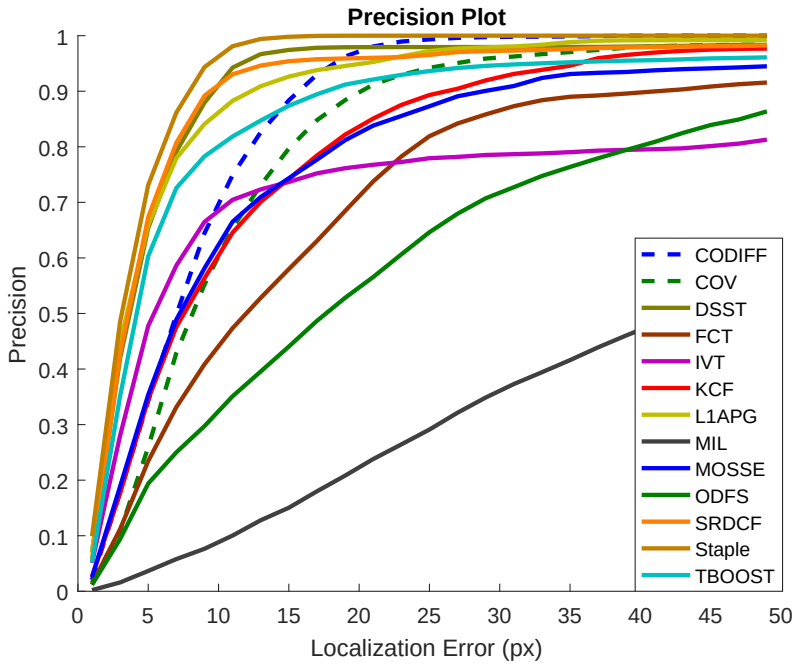
Figure 4.14: Sample images from Embryonic Rat Thoracic Aorta Medial Layer Myoblasts (A-10) sequence

4.3.2 Cell Motility Results

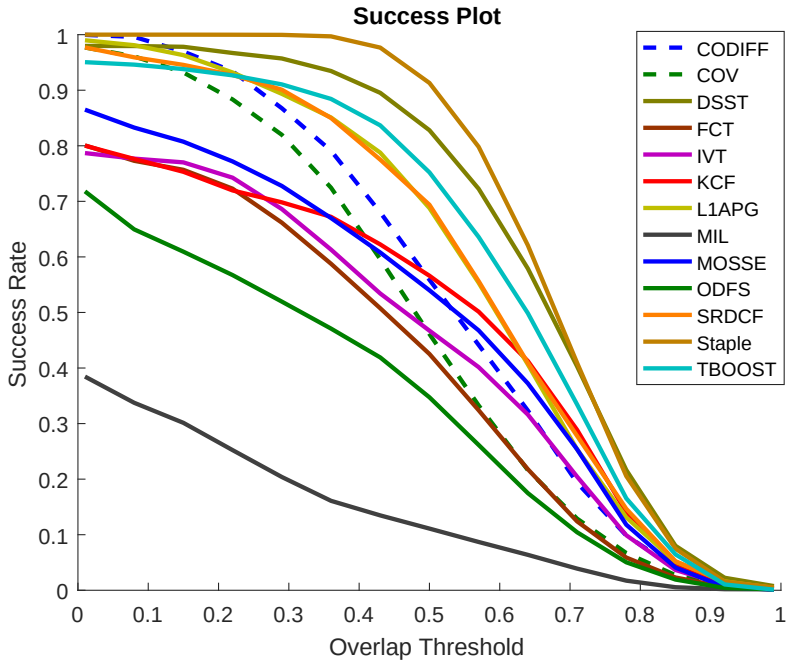
Overall performance results of compared visual object trackers are depicted in Figure 4.15 and quantitative results are listed in Table 4.6. Success and precision plots for each video sequence are provided in the Appendix B.

Cell motility results show that Staple,DSST and CODIFF algorithms have a better precision behaviour than other algorithms with a localization accuracy higher than 97 percent.

When the track maintenance scores are examined, best performing tracking algorithms are Staple, CODIFF and L1APG. AUC scores show that Staple, DSST and TBOOST algorithms have the most successful results in terms of average success rate.



(a) Succes vs overlap threshold plot



(b) Precision vs. localization error plot

Figure 4.15: Success and precision plots for Cell Motility Dataset

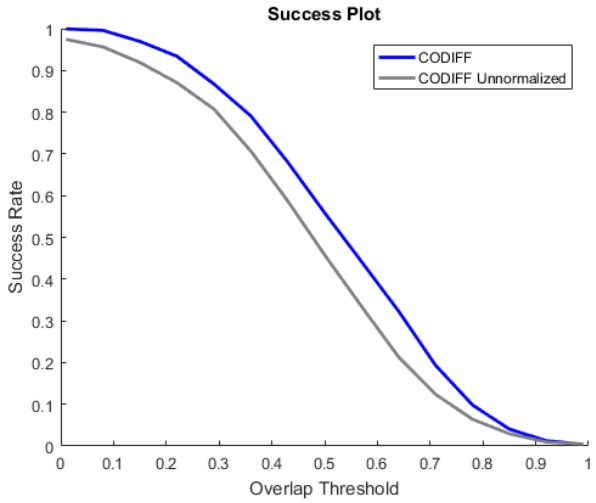
Table 4.6: Success and Precision rate comparison for Cell Motility Dataset

	Success		Precision
	AUC	*TM*	*LA*
Staple [59]	0.67	1.000	1.000
DSST [46]	0.63	0.978	0.978
CODIFF [67]	0.52	1.000	0.972
SRDCF [48]	0.56	0.976	0.959
L1APG [25]	0.57	0.989	0.949
TBOOST [20]	0.58	0.947	0.911
COV [17]	0.48	0.977	0.898
KCF [27]	0.45	0.787	0.826
MOSSE [26]	0.46	0.856	0.823
IVT [29]	0.43	0.803	0.783
FCT [23]	0.38	0.788	0.693
ODFS [22]	0.31	0.700	0.522
MIL [21]	0.11	0.345	0.193

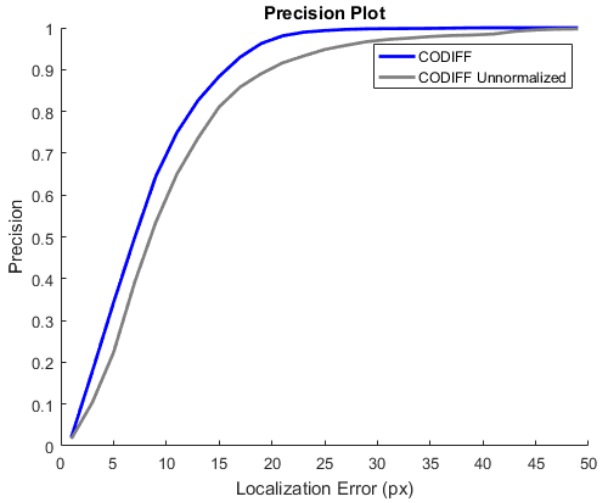
Table 4.7: Best performing trackers in cell motility video sequences

	Rank	3T3	BPAE	LLC-MK2	U2OS	A-10
Success Rate	#1	Staple	Staple	KCF	DSST	TBOOST
	#2	DSST	L1APG	Staple	Staple	KCF
	#3	CODIFF	DSST	DSST	KCF	CODIFF
Precision	#1	DSST	Staple	Staple	DSST	Staple
	#2	Staple	L1APG	SRDCF	Staple	DSST
	#3	CODIFF	DSST	KCF	L1APG	KCF

In cell motility videos, image pixels are represented by 8-bit values. We saw that normalizing the pixel values between 0 and 1 increases the performance of our descriptor. Results of co-difference tracking algorithm for both normalized and unnormalized intensity values are given in Figure 4.16. Because of this result, we used normalized intensity values in the co-difference descriptor for all the cell motility experiments reported in the thesis.



(a) Effect of normalization on the succes



(b) Effect of normalization on the precision

Figure 4.16: Effect of normalizing intensity values for cell motility dataset

4.4 Extension of multiplier-less operator to other trackers

In this thesis, we defined a multiplier-less operator and used it in the calculation of covariance-like features to achieve a better performance. We can extend this idea to various tracking algorithms. To show this with a simple example, we utilized the proposed multiplier-less operator in the normalized cross correlation (NCC) tracker.

For this experiment, we created a NCC tracking code that uses an image template as the appearance model and localizes the object by finding the maximum value of normalized cross correlation result within a search window. We also created a multiplier-less variant of this code where the multiplication operator in the correlation is replaced with the function given in Equation 3.2. Performance results of these two tracking methods for cell motility videos are given in Figure 4.17.

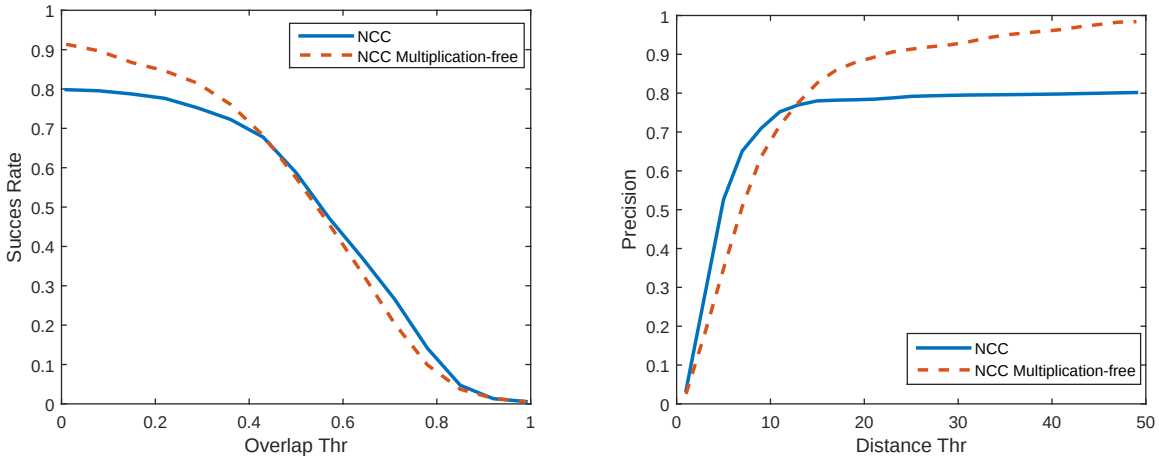


Figure 4.17: Effect of using multiplier-less operator in NCC tracker

As the figure demonstrates, we can achieve even better results without requiring multiplications. Similar to this example, we can extend this idea to other methods that contains a large number of multiplications.

Chapter 5

Conclusion

This thesis presents a novel visual object tracking algorithm based on the co-difference matrix. The calculation of co-difference descriptor is similar to covariance method, but it is more efficient to compute than the covariance matrix, because it can be implemented without performing any multiplications.

The co-difference matrix is based on a vector operator related with the ℓ_1 norm. On the other hand the covariance matrix is based on the inner-product operations. This is the fundamental difference between the two matrices. As a result the co-difference matrix of a given image region is sparser than the corresponding covariance matrix.

In our experiments, we compared our proposed method with various state of-the-art object tracking algorithms on two different applications. The experiments show that the proposed co-difference based tracking algorithm is among the best performing methods by having the highest localization accuracy and success rate for the infrared surveillance dataset, and it has the highest track maintenance score in the cell motility dataset.

5.1 Future Work

As a next step, a search mechanism in the scale space could be added to the current algorithm for scale-invariance. Moreover, box filter can be utilized in the implementation to improve the performance of the tracking. We can also extend the idea of utilizing multiplier-less operator to other successful tracking methods that require a large number of multiplications.

Bibliography

[1] Y. Dedeoğlu, "Moving object detection, tracking and classification for smart video surveillance," Master's thesis, Bilkent University, 2004.

[2] R. Jacob and K. S. Karn, "Eye tracking in human-computer interaction and usability research: Ready to deliver the promises," *Mind*, vol. 2, no. 3, p. 4, 2003.

[3] P. Majaranta and A. Bulling, "Eye tracking and eye-based human–computer interaction," in *Advances in physiological computing*, pp. 39–65, Springer, 2014.

[4] M. A. Sotelo, F. J. Rodriguez, L. Magdalena, L. M. Bergasa, and L. Boquete, "A color vision-based lane tracking system for autonomous driving on unmarked roads," *Autonomous Robots*, vol. 16, no. 1, pp. 95–116, 2004.

[5] A. Petrovskaya and S. Thrun, "Model based vehicle detection and tracking for autonomous urban driving," *Autonomous Robots*, vol. 26, no. 2-3, pp. 123–139, 2009.

[6] C. Cao, C. Li, and Y. Sun, "Motion tracking in medical images," *Biomedical Image Understanding, Methods and Applications*, pp. 229–274, 2015.

[7] D. Li, D. Winfield, and D. J. Parkhurst, "Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches," in *Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on*, pp. 79–79, IEEE, 2005.

[8] M. Mehrubeoglu, L. M. Pham, H. T. Le, R. Muddu, and D. Ryu, "Real-time eye tracking using a smart camera," in *2011 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, pp. 1–7, Oct 2011.

[9] M. Maška, V. Ulman, D. Svoboda, P. Matula, P. Matula, C. Ederra, A. Urbiola, T. España, S. Venkatesan, D. M. Balak, *et al.*, "A benchmark for comparison of cell tracking algorithms," *Bioinformatics*, vol. 30, no. 11, pp. 1609–1617, 2014.

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

[11] Y. Wu, J. Lim, and M.-H. Yang, "Object tracking benchmark," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 9, pp. 1834–1848, 2015.

[12] M. Kristan, R. Pflugfelder, A. Leonardis, J. Matas, L. Čehovin, G. Nebehay, T. Vojíř, G. Fernández, and et al, *The Visual Object Tracking VOT2014 Challenge Results*, pp. 191–217. Cham: Springer International Publishing, 2015.

[13] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Cehovin, G. Fernández, T. Vojir, G. Hager, G. Nebehay, and R. Pflugfelder, "The visual object tracking vot2015 challenge results," in *Proceedings of the IEEE international conference on computer vision workshops*, pp. 1–23, 2015.

[14] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. Čehovin, T. Vojír, G. Häger, A. Lukežič, G. Fernández, and et al, *The Visual Object Tracking VOT2016 Challenge Results*, pp. 777–823. Cham: Springer International Publishing, 2016.

[15] F. Porikli and T. Kocak, "Robust license plate detection using covariance descriptor in a neural network framework," in *Video and Signal Based Surveillance, 2006. AVSS '06. IEEE International Conference on*, pp. 107–107, Nov 2006.

[16] M. Faraki, M. Harandi, and F. Porikli, "Approximate infinite-dimensional region covariance descriptors for image classification," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pp. 1364–1368, April 2015.

[17] F. Porikli, O. Tuzel, and P. Meer, "Covariance tracking using model update based on lie algebra," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1, pp. 728–735, June 2006.

[18] H. Tuna, I. Onaran, and A. E. Cetin, "Image description using a multiplier-less operator," *Signal Processing Letters, IEEE*, vol. 16, pp. 751–753, Sept 2009.

[19] A. Suhre, F. Keskin, T. Ersahin, R. Cetin-Atalay, R. Ansari, and A. E. Cetin, "A multiplication-free framework for signal processing and applications in biomedical image analysis," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pp. 1123–1127, May 2013.

[20] E. Gundogdu, H. Ozkan, H. S. Demir, H. Ergezer, E. Akagunduz, and S. K. Pakin, "Comparison of infrared and visible imagery for object tracking: Toward trackers with superior ir performance," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2015 IEEE Conference on*, pp. 1–9, June 2015.

[21] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 983–990, June 2009.

[22] K. Zhang, L. Zhang, and M.-H. Yang, "Real-time object tracking via online discriminative feature selection," *Image Processing, IEEE Transactions on*, vol. 22, pp. 4664–4677, Dec 2013.

[23] K. Zhang, L. Zhang, and M.-H. Yang, "Fast compressive tracking," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, pp. 2002–2015, Oct 2014.

[24] S. Hare, A. Saffari, and P. Torr, "Struck: Structured output tracking with kernels," in *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 263–270, Nov 2011.

[25] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust l1 tracker using accelerated proximal gradient approach," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 1830–1837, June 2012.

[26] D. Bolme, J. Beveridge, B. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pp. 2544–2550, June 2010.

[27] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2015.

[28] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 2015.

[29] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1, pp. 125–141, 2007.

[30] F. Oberti, G. Ferrari, and C. S. Regazzoni, "A comparison between continuous and burst, recognition driven transmission policies in distributed 3gss," in *Video-Based Surveillance Systems*, pp. 267–278, Springer, 2002.

[31] H. İ. Cüce, "Mean-shift analysis for image and video applications," Master's thesis, Bilkent University, 2005.

[32] K. Li, E. D. Miller, M. Chen, T. Kanade, L. E. Weiss, and P. G. Campbell, "Cell population tracking and lineage construction with spatiotemporal context," *Medical image analysis*, vol. 12, no. 5, pp. 546–566, 2008.

[33] P. Friedl and D. Gilmour, "Collective cell migration in morphogenesis, regeneration and cancer," *Nature reviews Molecular cell biology*, vol. 10, no. 7, pp. 445–457, 2009.

[34] P. Friedl and S. Alexander, "Cancer invasion and the microenvironment: plasticity and reciprocity," *Cell*, vol. 147, no. 5, pp. 992–1009, 2011.

[35] R. M. Jiang, D. Crookes, N. Luo, and M. W. Davidson, "Live-cell tracking using sift features in dic microscopic videos," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 9, pp. 2219–2228, 2010.

[36] D. Gerlich, J. Mattes, and R. Eils, "Quantitative motion analysis and visualization of cellular structures," *Methods*, vol. 29, no. 1, pp. 3–13, 2003.

[37] O. Debeir, P. Van Ham, R. Kiss, and C. Decaestecker, "Tracking of migrating cells under phase-contrast video microscopy with combined mean-shift processes," *IEEE transactions on medical imaging*, vol. 24, no. 6, pp. 697–711, 2005.

[38] G. A. Dunn and G. E. Jones, "Cell motility under the microscope: Vorsprung durch technik," *Nature reviews. Molecular cell biology*, vol. 5, no. 8, p. 667, 2004.

[39] N. Ray and S. T. Acton, "Motion gradient vector flow: An external force for tracking rolling leukocytes with shape and size constrained active contours," *IEEE transactions on medical Imaging*, vol. 23, no. 12, pp. 1466–1478, 2004.

[40] Y. Sato, J. Chen, R. A. Zoroofi, N. Harada, S. Tamura, and T. Shiga, "Automatic extraction and measurement of leukocyte motion in microvessels using spatiotemporal image analysis," *IEEE Transactions on Biomedical Engineering*, vol. 44, no. 4, pp. 225–236, 1997.

[41] A. Hand, T. Sun, D. Barber, D. Hose, and S. MacNeil, "Automated tracking of migrating cells in phase-contrast video microscopy sequences using image registration," *Journal of microscopy*, vol. 234, no. 1, pp. 62–79, 2009.

[42] E. Meijering, O. Dzyubachyk, I. Smal, *et al.*, "9 methods for cell and particle tracking," *Methods in enzymology*, vol. 504, no. 9, pp. 183–200, 2012.

[43] N. Chenouard, I. Smal, F. De Chaumont, M. Maška, I. F. Sbalzarini, Y. Gong, J. Cardinale, C. Carthel, S. Coraluppi, M. Winter, *et al.*, "Objective comparison of particle tracking methods," *Nature methods*, vol. 11, no. 3, pp. 281–289, 2014.

[44] M. Maška, V. Ulman, D. Svoboda, P. Matula, P. Matula, C. Ederra, A. Urbiola, T. España, S. Venkatesan, D. M. Balak, *et al.*, "A benchmark for comparison of cell tracking algorithms," *Bioinformatics*, vol. 30, no. 11, pp. 1609–1617, 2014.

[45] F. Piccinini, A. Kiss, and P. Horvath, "Celltracker (not only) for dummies," *Bioinformatics*, vol. 32, no. 6, pp. 955–957, 2016.

[46] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *British Machine Vision Conference, Nottingham, September 1-5, 2014*, BMVA Press, 2014.

[47] J. Kwon and K. M. Lee, "Visual tracking decomposition," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1269–1276, June 2010.

[48] M. Danelljan, G. Hger, F. S. Khan, and M. Felsberg, "Learning spatially regularized correlation filters for visual tracking," in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 4310–4318, Dec 2015.

[49] Linkoping University Computer Vision Laboratory, "Discriminative Scale Space Tracker." `http://www.cvl.isy.liu.se/en/research/objrec/visualtracking/scalvistrack/DSST_code.zip`. Linkoping University, Accessed 2017-08-02.

[50] K. Zhang, "Fast Compressive Tracking." `http://www4.comp.polyu.edu.hk/~cslzhang/FCT/FCT.htm`. Accessed 2017-08-02.

[51] D. Ross, "Incremental Learning for Robust Visual Tracking." `http://www.cs.toronto.edu/~dross/ivt/`. Accessed 2017-08-02.

[52] S. Hare, "Struck: Structured Output Tracking with Kernels." `https://github.com/samhare/struck`. Accessed 2017-08-02.

[53] T. Vojir, "Tracking with Kernelized Correlation Filters." https://github.com/vojirt/kcf. Accessed 2017-08-02.

[54] H. Ling, "L1 tracking using accelerated proximal gradient." https://github.com/lukacu/visual-tracking-matlab/tree/master/l1apg. Accessed 2017-08-02.

[55] L. Cehovin, "Multiple Instance Learning Tracker." https://github.com/lukacu/mil. Accessed 2017-08-02.

[56] A. Q. Delgado, "Tracking using Adaptive correlation filters." https://github.com/albertoQD/tracking-mosse. Accessed 2017-08-02.

[57] K. Zhang, "Real-time Object Tracking via Online Discriminative Feature Selection." http://www4.comp.polyu.edu.hk/~cslzhang/ODFS/ODFS.htm. Accessed 2017-08-02.

[58] Linkoping University Computer Vision Laboratory, "Learning Spatially Regularized Correlation Filters for Visual Tracking." https://www.cvl.isy.liu.se/en/research/objrec/visualtracking/regvistrack/. Linkoping University, Accessed 2017-08-02.

[59] L. Bertinetto, J. Valmadre, S. Golodetz, O. Miksik, and P. H. S. Torr, "Staple: Complementary learners for real-time tracking," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[60] L. Bertinetto, "Staple: Complementary Learners for Real-Time Tracking." https://github.com/bertinetto/staple. Accessed 2017-08-02.

[61] K. Duman, "Methods for target detection in sar images," Master's thesis, Bilkent University, 2009.

[62] B. Rao, "Signal processing with the sparseness constraint," in *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, vol. 3, pp. 1861–1864 vol.3, May 1998.

[63] R. Baraniuk, "Compressive sensing [lecture notes]," *Signal Processing Magazine, IEEE*, vol. 24, pp. 118–121, July 2007.

[64] P. Combettes and J. Pesquet, "Image restoration subject to a total variation constraint," *Image Processing, IEEE Transactions on*, vol. 13, pp. 1213–1222, Sept 2004.

[65] M. Tofighi, O. Yorulmaz, K. Kose, D. Yildirim, R. Cetin-Atalay, and A. E. Cetin, "Phase and tv based convex sets for blind deconvolution of microscopic images," *IEEE Journal of Selected Topics in Signal Processing*, to be published in February 2016.

[66] Nikon Instruments, "Cell Motility." `https://www.microscopyu.com/galleries/cell-motility`, 2016. Nikon's MicroscopyU, Accessed 2017-06-02.

[67] H. S. Demir and A. E. Cetin, "Co-difference based object tracking algorithm for infrared videos," in *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 434–438, Sept 2016.

[68] G. Ottaviani and N. Jaffe, *The Epidemiology of Osteosarcoma*, pp. 3–13. Boston, MA: Springer US, 2010.

Appendix A

Individual Results for Surveillance Dataset

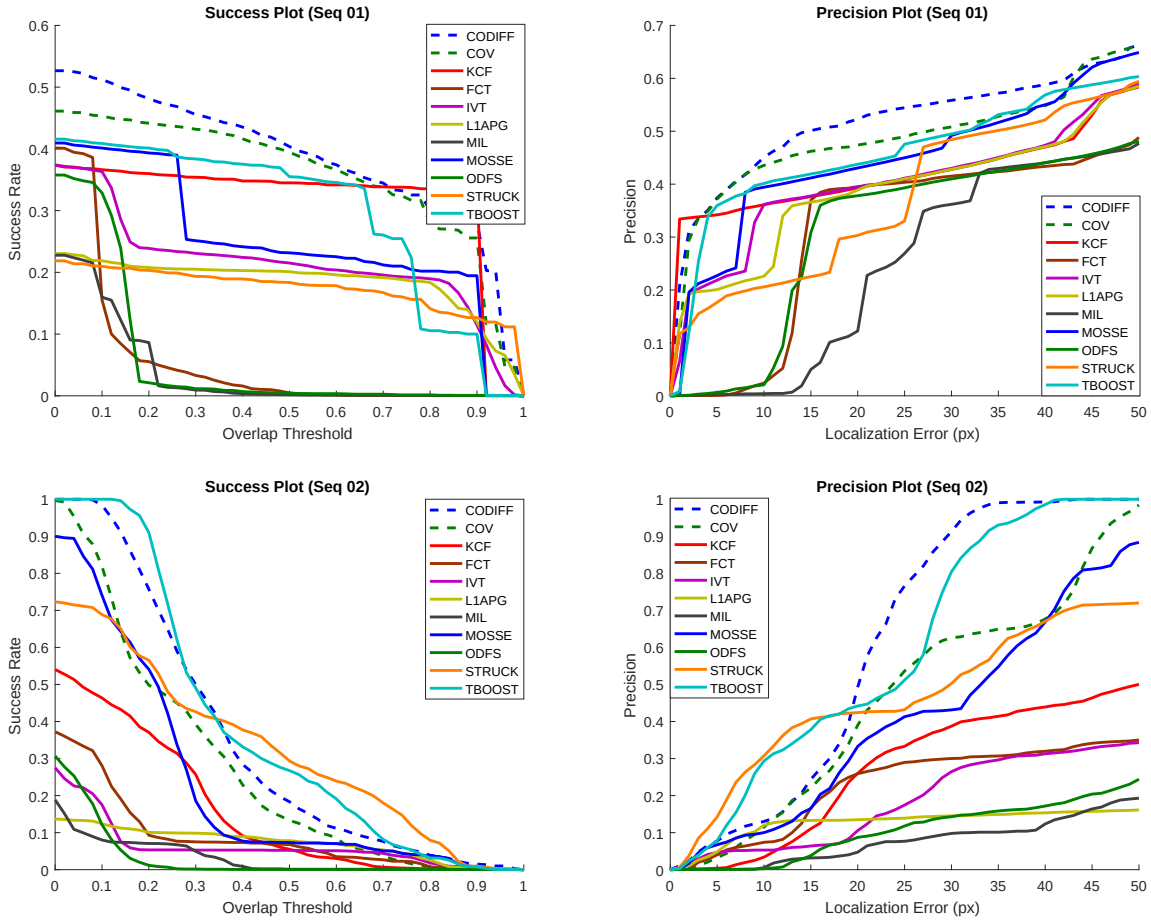


Figure A.1: Results for Video Sequences 01 and 02

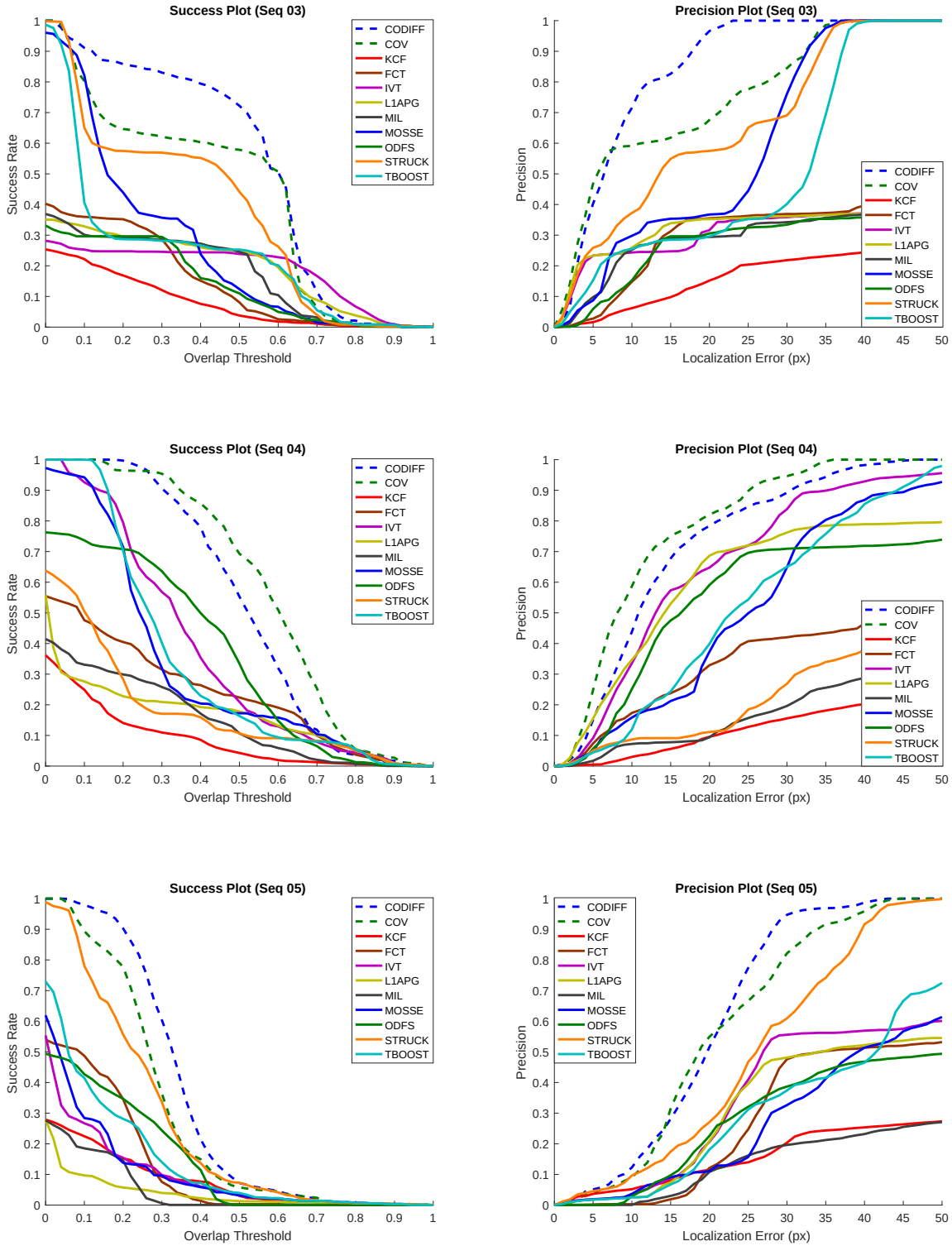


Figure A.2: Results for Video Sequences 03-05

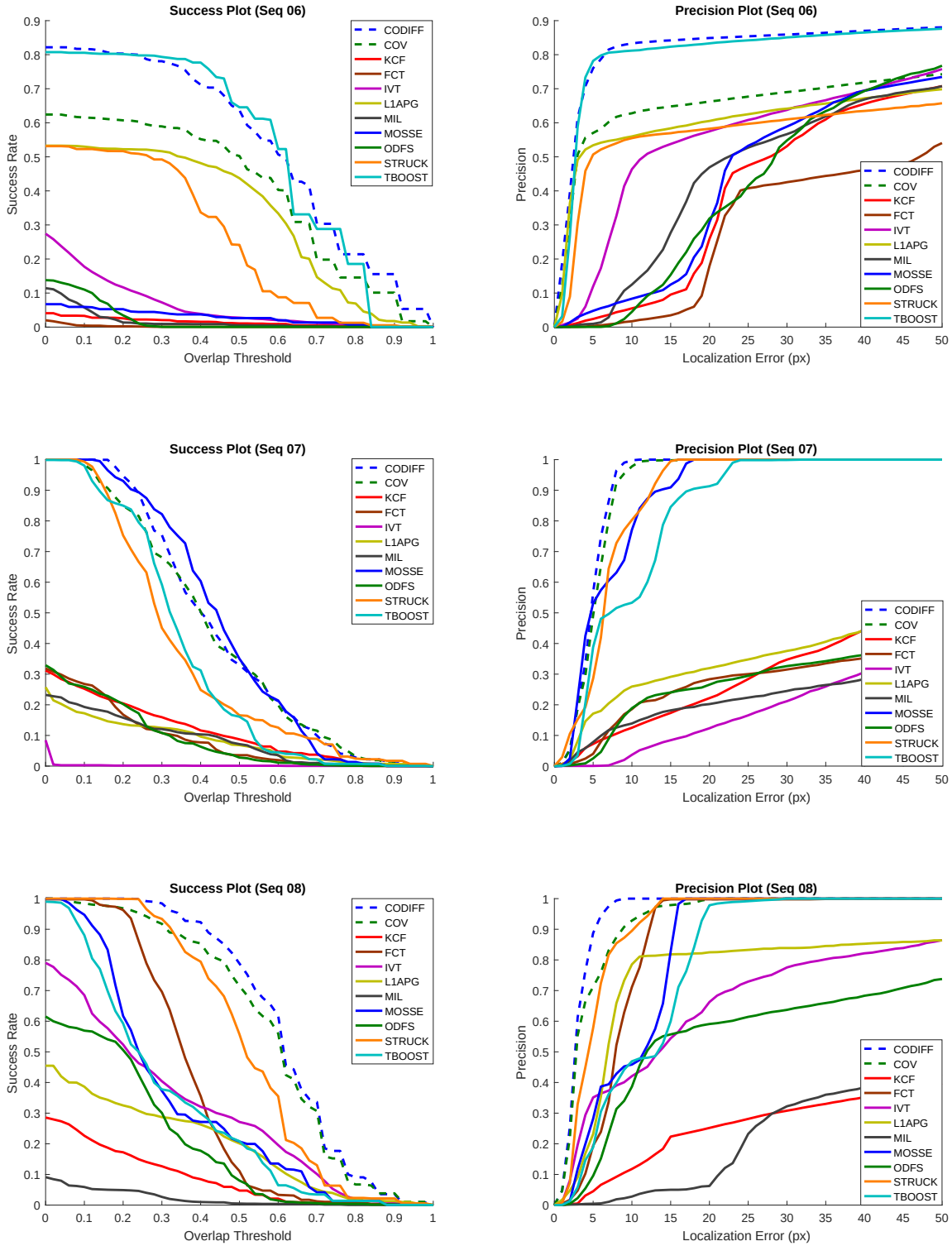


Figure A.3: Results for Video Sequences 06-08

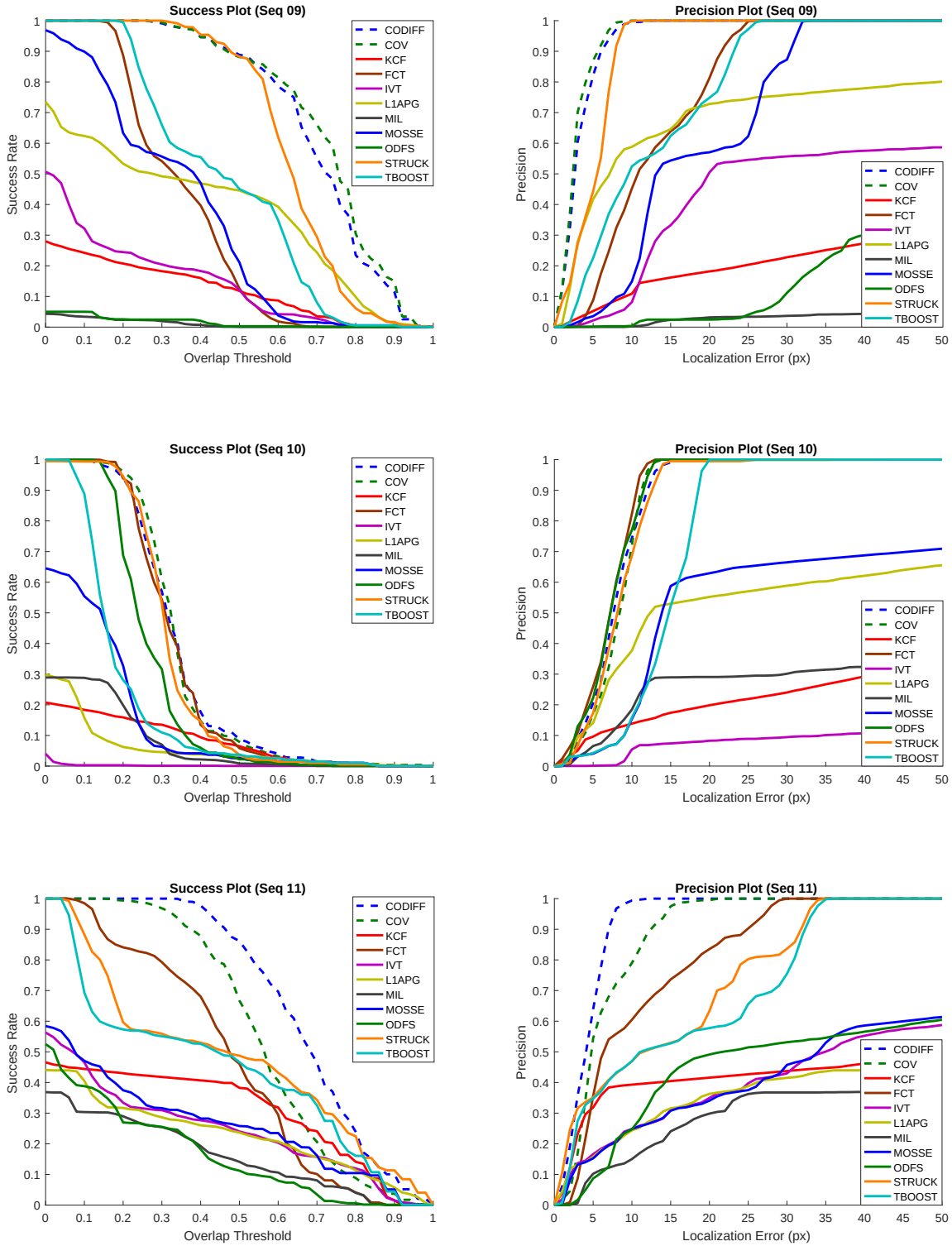


Figure A.4: Results for Video Sequences 09-11

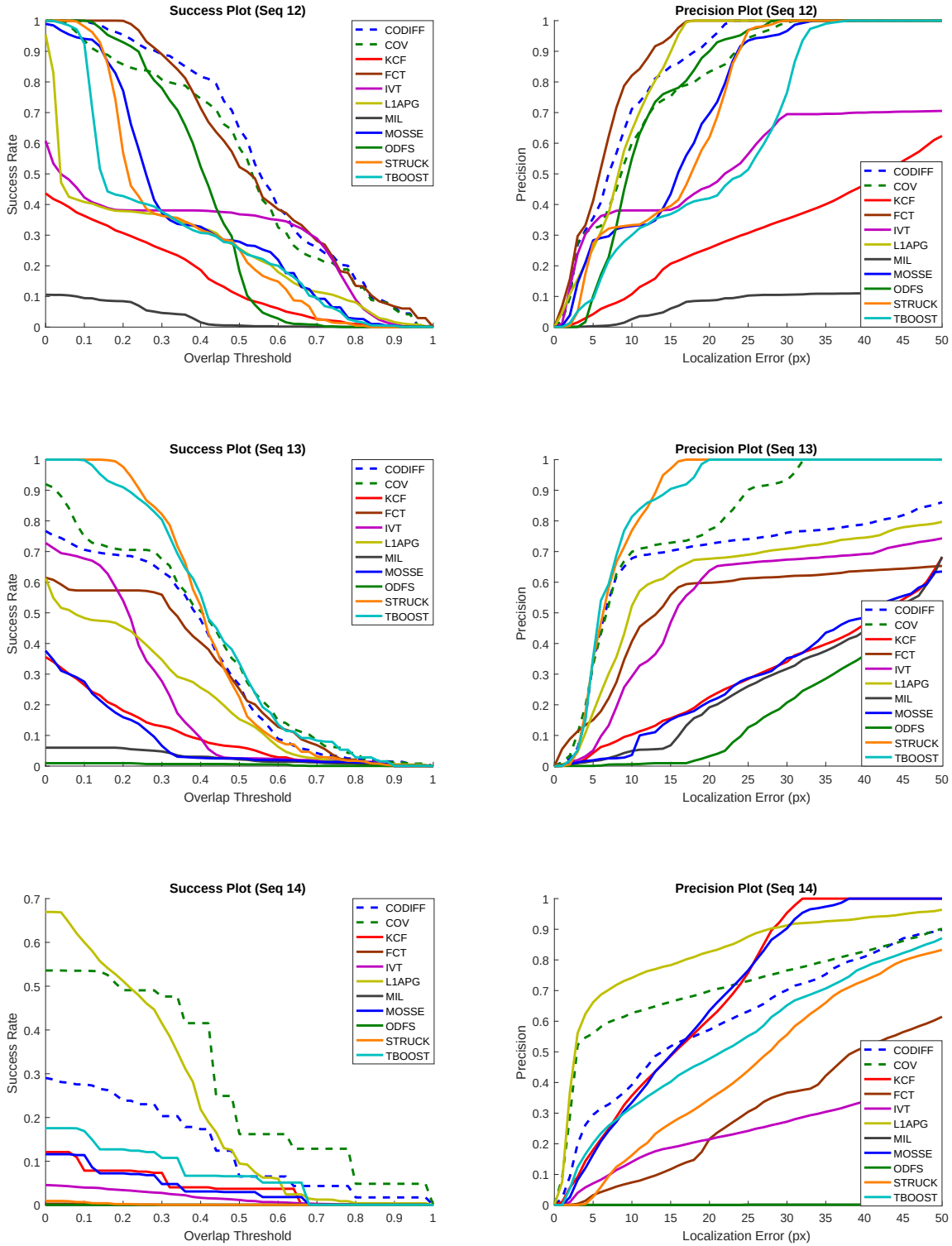


Figure A.5: Results for Video Sequences 12-14

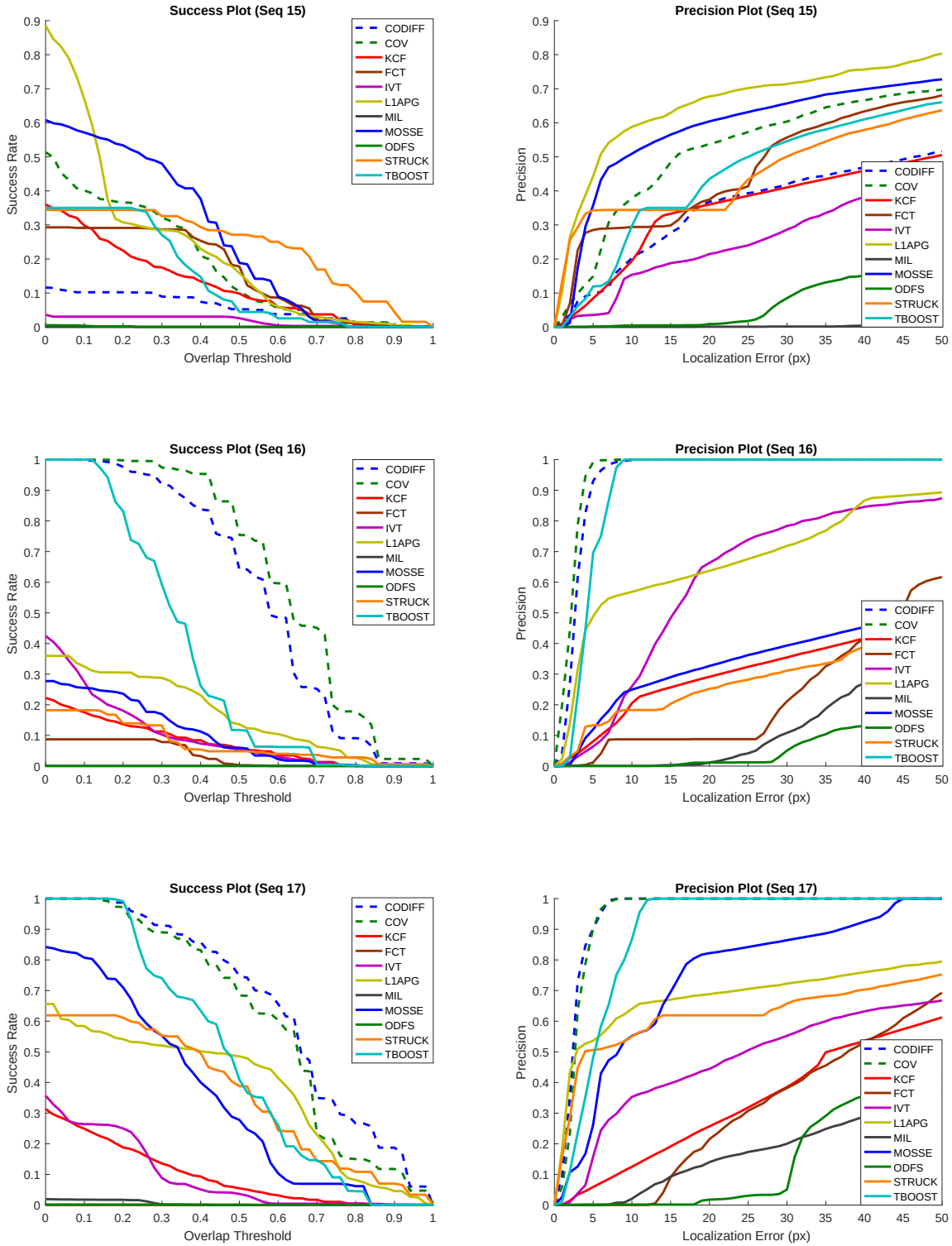


Figure A.6: Results for Video Sequences 15-17

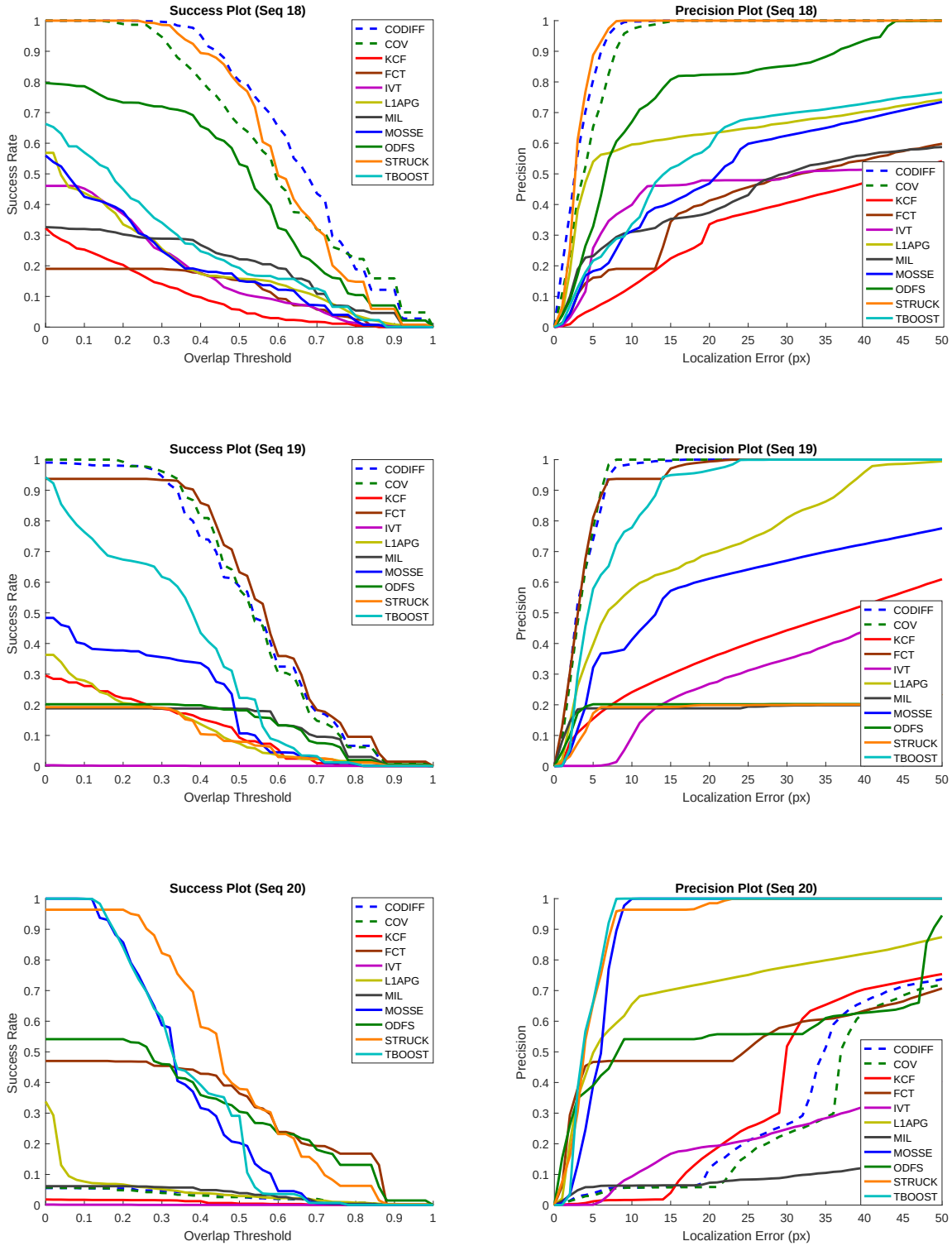


Figure A.7: Results for Video Sequences 18-20

Appendix B

Individual Results for Cell Motility Dataset

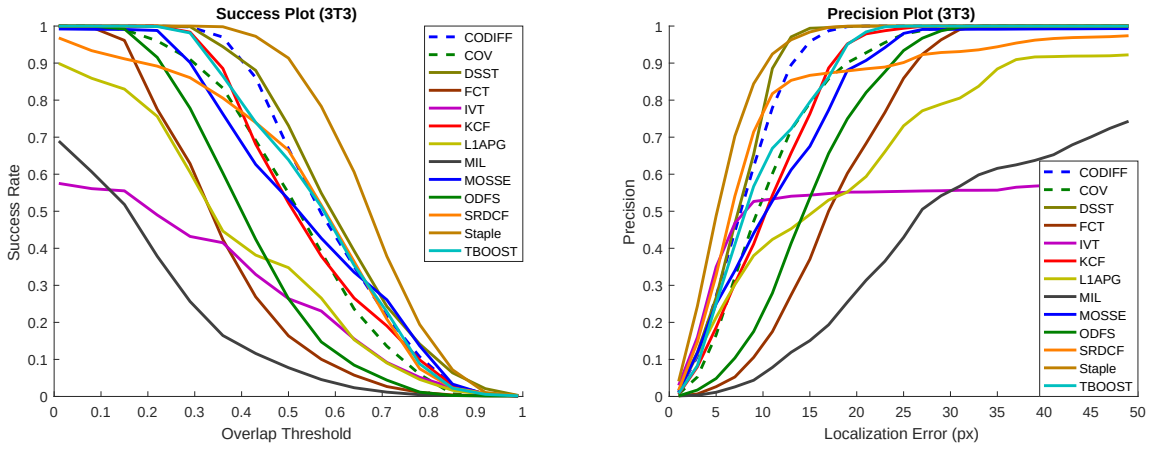


Figure B.1: Results for 3T3 Image Sequence

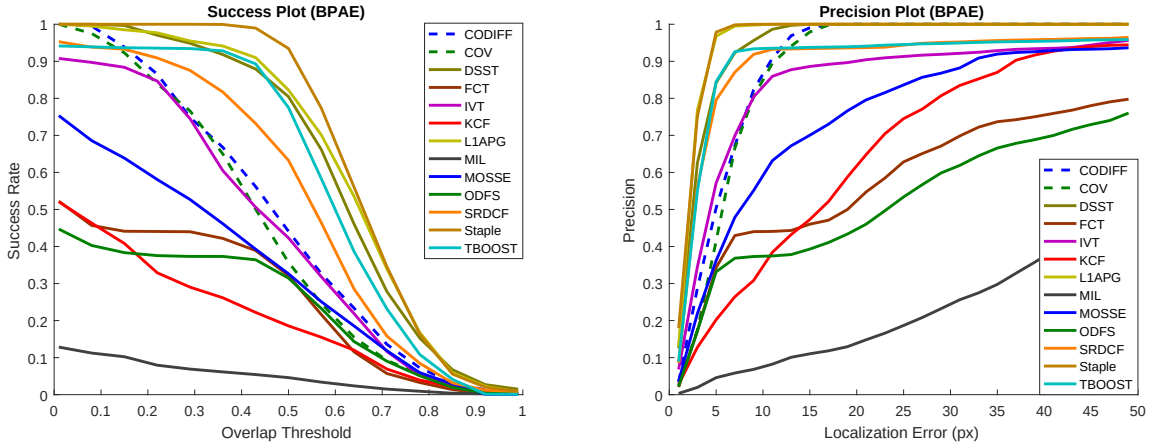


Figure B.2: Results for BPAE Image Sequence

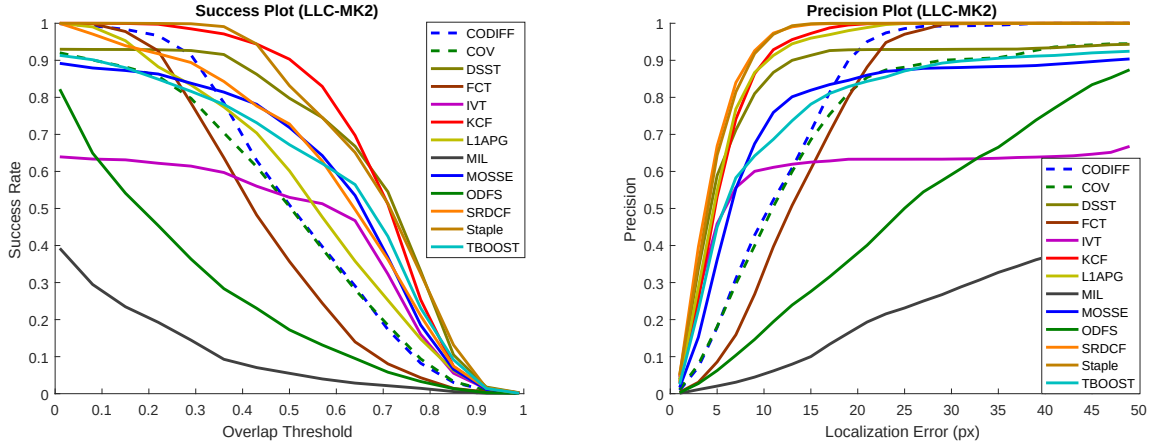


Figure B.3: Results for LLC-MK2 Image Sequence

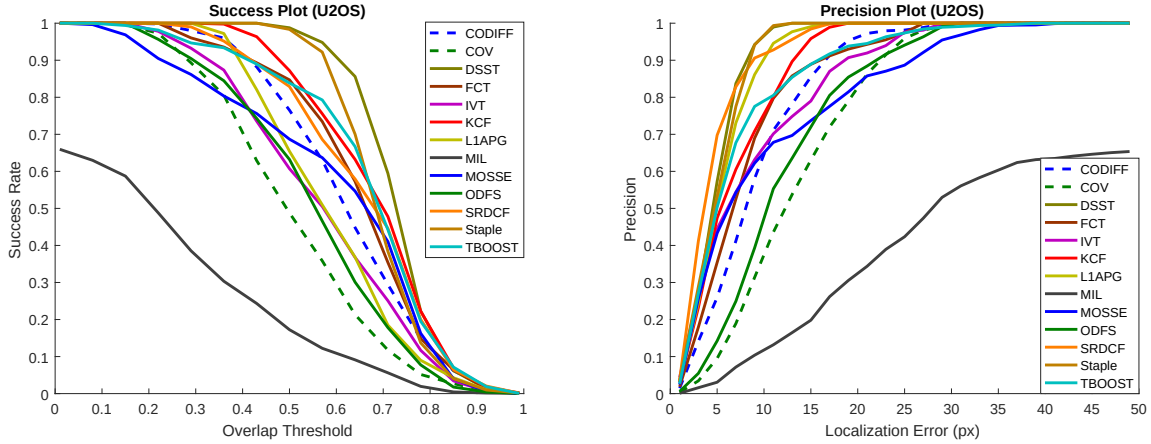


Figure B.4: Results for U2OS Image Sequence

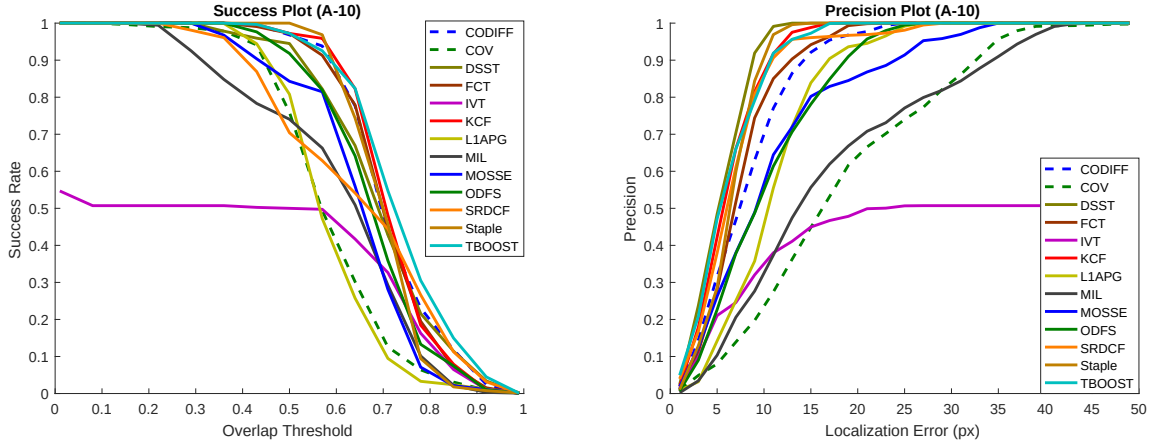


Figure B.5: Results for A-10 Image Sequence