KERNERLIZED CORRELATION FILTERS PARAMETERS OPTIMIZATION FOR ENHANCED VISUAL TRACKING

By:

Ong Chor Keat

(Matrix No.: 120583)

Supervisor:

Dr. Parvathy A/P Rajendran

June 2017

This dissertation is submitted to
Universiti Sains Malaysia
As partial fulfillment of the requirement to graduate with honors degrees in
BACHELOR OF ENGINEERING (AESOSPACE ENGINEERING)

School of Aerospace Engineering
Engineering Campus
Universiti Sains Malaysia

KERNERLIZED CORRELATION FILTERS PARAMETERS OPTIMIZATION FOR ENHANCED VISUAL TRACKING

ABSTRACT

Visual tracking has become one of the most important components in computer vision as the knowledge in this field can be applied into a wide range of applications in computer vision such as medical imaging, pattern recognition, video surveillance, industrial robot, computer-human interaction, etc. A lot of researches have been conducted and many types of state-of-the-art methods and modifications such as sparse representation, online similarity learning, self-expressive, spatial kernel phase correlation filter and others are proposed in order to increase the robustness of the tracking. Despite of many methods has been demonstrated successfully, but there are several issues that still need to be addressed. There still have some unsolvable difficulties in which they become a challenging task to track an object effectively and robustly and it will tend to decrease the accuracy of the results and hence. Until now, there are still no perfect algorithm to track the target flawlessly. In order to improve the performance, the main idea proposed is implementing optimization technique on the selected parameters and obtain a better performance. In this research, the tracking is proposed by using the overlap ratio (OR) and centre location error (CLE). In our case, our target is to obtain a better accuracy, which is higher overlap ratio and lower centre location error than the result from the algorithms available in public. A simple optimization is used in here, where the global best results with respect to the value of the parameters are selected through a range of values defined in our work. Through the optimization, the overall overlap ratio is increased to 0.554 and overall centre location error is decreased to 19.803 pixels. Thus, the proposed method had increased the accuracy and robustness of the visual tracking on many of the video sequences.

OPTIMASI PARAMETER PENAPISAN KERNERLIZED KORELASI UNTUK MEMPERTINGKATKAN PENGESANAN VISUAL

ABSTRAK

Pengesanan visual telah menjadi salah satu komponen yang penting dalam bidang penglihatan computer kerana ilmu pengetahuan dalam bidang ini dapat digunakan dalam pelbagai aplikasi seperti pengimejan dalam bidang perubatan, pengesahan corak, pengawasan video, robot dalam industri, interaksi antara komputer-manusia dan lain-lain. Penyelidik-penyelidik telah menjalankan pengesanan visual dengan pelbagai keadah and pengubasuaian juga dicadangkan untuk menanmbahbaikan keputusan pengesanan. Walaupun kebanyakan kaedah telah mencapai keputusan yang memuaskan, tetapi masih wujud beberapa isu yang perlu diutamakan kerana masalahnya tidak dapat diselesai sepenuhnya dan ini telah menjadi cabaran dalam pengesanan visual. Oleh sebab ini, masih tidak berwujudnya kaedah yang dapat mengesan sasaran dengan sempurnanya. Idea utama yang dicadangkan untuk mempertingkatkan keseluruhan keputusan pengesanan adalah menjalankan teknik optimasi pada parameter yang dipilih. Di sini, prestasi dinilai dengan mengunakan kadar pertindihan (OR) dan ralat lokasi (CLE). Untuk mendapat prestasi yang terbaik, kadar pertindihan perlu ditingkatkan ke tahap maksimum dan ralat lokasi perlu dikurangkan ke tahap minimum berbanding dengan algoritma yang didapati dalam awam. Satu optimasi yang mudah digunakan di sini, keputusan yang terbaik akan dipilih bersama dengan nilai parameter daripada jarak yang ditentukan dalam cara kita. Dengan optimasi, kadar pertindihan purata telah ditingkatkan kepada 0.554 dan ralat lokasi purata dikurangkan kepada 19.803 pixels. Oleh, itu, keadah yang dicadangkan telah mencapai prestasi yang diharapkan dari segi ketepatan dan kemantapan di pengesanan visual pada pelbagai video.

ACKNOWLEDGEMENTS

First of all, I would like to express my deepest gratitude to all that have been help me a lot from the beginning until the end of my final year project especially to my supervisor, Dr. Parvathy Rajendran. As she gives me an opportunity to work with her, provides helps and guidance along the way, thus I need to express my deepest and most sincere gratitude to her again. The second most important person I would like to thank is Mr. Joao F. Henriques, because of his coding that is available in the public, thus provide me a chance to improve his result in my project the Besides, I also want to express my sincere appreciation to some lecturers, Dr. Halim, Dr. Elmi, Dr. Nurulashikin, Dr. Aslina and others as they had conducted a short briefing on our final year project which help us a lot on how to use endnote in Microsoft Word and how to write our thesis. In addition, I want to thank to my seniors, Mr. Kok Kai Yit and Ms. Law Hooi Mee because they gave some technical advises, motivations and suggestions on my final year project when I encountered some difficulties. Last but not least, I would like to thank to all of my friends and my family for their support, motivation, encouragement and caring, as their existence gives the strength for me to complete this final year project successfully.

DECLARATION

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

_____

ONG CHOR KEAT
Date: 5th June 2017

STATEMENT 1

This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by giving explicit references. Bibliography/references are appended.

_____

ONG CHOR KEAT
Date: 5th June 2017

STATEMENT 2

I hereby give consent for my thesis, if accepted, to be available for photocopying and for interlibrary loan, and for the title and summary to be made available to outside organizations.

_____

ONG CHOR KEAT
Date: 5th June 2017

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

IV	Illumination Variation
SC	Scale Variation
OCC	Occlusion
DEF	Deformation
MB	Motion Blur
FM	Fast Motion
IPR	In-Plane Rotation
OPR	Out-of-Plane Rotation
OV	Out-of-View
BC	Background Clutters
LR	Low Resolution
OR	Overlap Rate
CLE	Centre Location Error
KCF	Kernerlized Correlation Filters
KCFop	Kernerlized Correlation Filters Optimized (Proposed Method)
Struck	Structure Output Tracking
DFT	Distribution Field for Tracking
CSK	Tracking by Detection with Circulant Structure
MIL	Multiple Instance Learning

Frag	Fragment based Tracking using the Integral Histogram
CT	Compressive Tracking
OAB	Tracking via Online Boosting
LOT	Locally Orderless Tracking
CXT	Context Tracking
SemiT	Semi-supervised Online Boosting for Tracking
VTD	Visual Tracking Decomposition
IVT	Incremental Learning for Robust Visual Tracking
L1APG	L1 Tracker using Accelerated Proximal Gradient
SVM	Support Vector Machine
TRE	Temporal Robustness Evaluation

NOMENCLATURE

λ	Regularization Parameter
X^H	Hermitian Transpose
X^*	Complex-Conjugate of X
I	Identity Matrix
F	DFT Matrix
K	Kernel Matrix
α	Vector of Coefficient α_i
k^{XZ}	Kernel Correlation
σ	Kernel Sigma
p	Padding
s	Spatial Bandwidth
w_1, w_2	Frame Width and Target Width
h_1, h_2	Frame Height and Target Height
x_t, y_t	Centre Location of the Target
x_g, y_g	Ground truth Location of the Target
ROI_t	Area of Target's Region
ROI_g	Area of Ground Truth's Region

CHAPTER 1

INTRODUCTION

1.1 GENERAL OVERVIEW

Visual tracking is one of the research field in computer science that increases in popularity due to the importance to many applications in computer vision such as medical imaging, pattern recognition, video surveillance, industrial robot, computer-human interaction, etc.

Basically, the main objective of visual tracking is to estimate and locate the target objects in consecutive video frames. In general, the basis working principle of visual tracking is, after inputting a video sequence, we require a description for the object to be tracked. For example, shape, colour model, texture or others can use to be the template image of the object. Next, some context is applied into the object as implementing a good and proper integration of such context information into a tracking framework will bring some positive effects to visual tracking. After context information integration, the classifier classifies the image patches, then updated from time to time which is also known as online learning in order to handle and adapt the new appearance changes in the subsequent frames. These steps are repeated to track the object and stop when it reaches the last frame of the video. Although visual tracking has been studied for several decades, but it is remaining as a challenging topic to be researched as mainly due to abrupt object motion, appearance pattern change, non-rigid object structures, occlusion and camera motion. And thus there are no a single comprehensive method to handle all these destabilizing factors where These destabilizing factors are mainly consist of 11 attributes, which are shown in Table 1.1 with description respectively [1, 2].

Table 1.1: 11 attributes annotated to test sequences with threshold values provided

Name	**Description**
Illumination Variation (IV)	the illumination in the target region is significantly changed

Scale Variation (SV)	the ratio of the bounding boxes of the first frame and the current frame is out of the range $[1/t_s, t_s]$, $t_s > 1$ (t_s=2)
Occlusion (OCC)	the target is partially or fully occluded
Deformation (DEF)	non-rigid object deformation
Motion Blur (MB)	the target region is blurred due to the motion of target or camera
Fast Motion (FM)	the motion of the ground truth is larger than tm pixels (t_m=20)
In-Plane Rotation (IPR)	the target rotates in the image plane
Out-of-Plane Rotation (OPR)	the target rotates out of the image plane
Out-of-View (OV)	some portion of the target leaves the view
Background Clutters (BC)	the background near the target has the similar color or texture as the target
Low Resolution (LR)	the number of pixels inside the ground-truth bounding box is less than t_r (t_r =400)

In recent years, different algorithms have been proposed in order to solve the challenging issues. One of the methods is choose the right features or the most desirable property of a visual feature in order to be distinguished in the feature space easily. So, feature descriptors are playing an important role in selecting the right features. For instances, gradient feature is proved to have advantageous in human detection [3, 4]; color features which are robust against certain photomatic changes; texture features where texture is used to measure the intensity of a surface and quantifies properties such as smoothness and regularity [5-7]; spatio-temporal features which used as representation for action recognition and visual detection; multiple features fusion which is more robust for image and video retrieval, visual tracking and detection.

Despite these feature descriptors, visual tracking still requires online learning based tracking methods to handle appearance variations of a target object. Online learning is required in for the tracker to adapt these appearance changes, adjust and update to new situations from time to time. There are 2 types of appearances variations, which are intrinsic (pose changing, shape deformation) and extrinsic (occlusion, camera motion, camera viewpoint and illumination variation). Thus, these appearance variations must be handled by the online

learning algorithm, which is divided into 2 categories: generative method and discriminative method.

1.2 PROBLEM STATEMENT

As mentioned above, although dozens of tracking algorithm have being proposed, but it still cannot achieve the best performance. We believe that the tracking results from the algorithm that we are focusing now can be better than original results through optimization. According to the Visual Tracking Benchmark, a successful tracking performance requires overlap ratio more than 0.5 or 50% and centre location error less than 20 pixels. Imagine that if the tracker's performance does not achieve the range of these two standard threshold values, worst case scenario, the bounding box will become further apart with the target and even loss their target during the tracking process. Sometimes it will not able to recover back to its track. Thus, the tracker will become less effective to be used by user which will cause a huge loss of its contribution to many of the visual tracking applications.

1.3 OBJECTIVE

The research is studied based on the objectives as stated below, which are:

- To study a novelty idea or method for optimization in enhancement of visual tracking.
- To improve the tracking accuracy by maximize the overlap ratio and minimize the centre location error.

1.4 THESIS LAYOUT

This thesis will consist of six chapters. Chapter 1 introduces the general overview of visual tracking in this modern era with some common problems that encountered during the tracking process. Chapter 2 reviews on the previous work done by the researchers related to this field. This literature highlights about generative model, discriminative model, collaboration between generative and discriminative model and correlation filters. The proposed methods by these researchers will be discussed in this chapter. Chapter 3 describes

the methodology that being used in this project. The proposed method for optimization and implemented general algorithm will be discussed in detail. Parameter chosen for optimization, mathematical equations used in “High Speed Tracking with Kernelized Correlation Filters (KCF)” [46] are discussed and the calculation to evaluate the visual tracking performance are presented in this chapter. Chapter 4 presents on the tracking results after optimization. The work of the optimization is discussed with average tracking performance in term of OR and CLE. The results between our proposed method and other tracking algorithms is compared also in term of OR and CLE. Lastly, chapter 5 concludes the work and results achieved by our proposed method and proposes future work that will be conducted to achieve the best tracking performance for our proposed method.

CHAPTER 2

LITERATURE REVIEW

2.1 APPEARANCE MODEL

Dozens of visual tracking method have been proposed and reviewed in [2]. Recently, online learning has become important element in visual tracking to elevate the tracking performances. Therefore, online learning is required in for the tracker to adapt these appearance changes, adjust and update to new situations from time to time. There are 2 types of appearances variations, which are intrinsic (pose changing, shape deformation) and extrinsic (occlusion, camera motion, camera viewpoint and illumination variation). Thus, these appearance variations must be handled by the online learning algorithm, which is divided into 2 categories: generative method and discriminative method.

Generally, generative online learning method will learn the appearance of the object, then it will update online on the object model in order to adapt the appearance changes. Adam et al. [8] represented the target using integral histogram and robust in target with partial occlusions or pose changes. Ross et al. [9] presented an appearance-based tracker to gradually learn a low dimensional eigen basis representation for tracking the target that with changing pose, illumination and appearance from time to time. Performance of [9] can said to be satisfying but it will encounter drifting problem. Jia et al. [10] implemented a template update strategy which incremental subspace learning and sparse representation are combined together. The adaption of the template reduces possibility of drifting and the effect of the occluded target template. Bao et al. [11] proposed, by adding an l_2 norm regularization on the coefficients associated with the trivial templates into a new l_1 norm related minimization model, it can achieved a better tracking accuracy than other l_1 tracker, [12, 13]. Mei et al. [12] casted the tracking as a sparse approximation problem in a particle filter framework and achieved a very

promising tracking result. Mei et al. [13] presented a new approach known as Bounded Particle Resampling (BPR)-L1 tracker to enhance the template updates by detect occlusions and lessen the drifting problem. Liu et al. [14] proposed a new selection-based dictionary learning method known as K -Selection and modelled the target appearance by using a sparse coding histogram based on a learned dictionary. By this way, it can adapt to appearance changes and drifting problem is reduced. Liu et al. [15] proposed a two stage sparse optimization to minimize the reconstruction error of the target and select a sparse set of features to maximize the discriminative power. Tian et al. [16] gathered the sparse coefficients of all patches in an object into a histogram based on their spatial distribution. The candidates are predicted for object verification during tracking by using particle filter methodology. Sparse coding is implemented to evaluate degree of changes of the appearance model and thus reduced the drifting problem. Cheng et al. [17] had conducted both generative and discriminative trackers under the particle filter framework. Common method implemented by [10, 12, 14-17] is utilizing sparse representation to represent the target, and their work prove that sparse representation is more powerful tool to handle and analysis appearance representation during online tracking, where it had overcame many challenging attribute such as heavy occlusions, illumination changes and pose variation. Li et al. [18] embedded "Online Reconstruction Error Prediction (OREP)" into the IVT [9] framework to predict appearance reconstruction error, and proven that OREP greatly improved the performance of some video sequences as compared with [9, 11].

Meanwhile, discriminative learning method required a classifier to be trained and updated online to differentiate the object from the background. It is also known as tracking-by-detection because it requires the user to manually identify the target in the first frame to generate a set of features of target. Then, another separate set of features is generated automatically to describe the background. Next, the target will be separated from the

background in the subsequent frames. Similarly, it must be updated continuously to handle the appearance changes. Support vector tracking [19] proposed SVM to optimize the classification score by generating a Gaussian pyramid from every support vector, known as "Support Vector Pyramid" to account large motions in the image plane. The experiment shows that it performs better in long period of vehicles tracking. Babenko et al. [20] proposed online MIL algorithm for object tracking and achieves promising performance with real-time tracking. Henriques et al. [21] proposed Fourier analysis that capable for extremely fast learning and detection with the Fast Fourier Transform. Closed-form solutions for training and detection with several types of kernels, including the popular Gaussian and polynomial kernels are derived and the algorithm achieved competitive performance. Yang et al. [22] proposed superpixels in an appearance model that gives flexible and effective mid-level cues to distinguish the background and the foreground target. [22] is more capable to handle the situations with big changes of pose and scale, shape deformation, occlusion and camera shake. Zhang, Song et al. [23] presented online weighted multiple instance learning (WMIL) to integrate the sample importance into the learning procedure, compute a new bag probability function that combines the weighted instance probability. Patras, Hancock et al. [24] presented a discriminative framework that coupled the predictor to a probabilistic classifier to predict the target accurately. Yuan et al. [25] proposed a robust superpixel-based tracker via depth fusion, developed sufficient structural information and high flexibility of mid-level features, depth-map's discriminative ability for the target and background separation, thus generated stronger discriminative ability. Fan et al. [26] presented a supervised approach to learn and update a structured, sparse, and discriminative representation that alternating between robust sparse coding and dictionary updating. Zhuang et al. [27] presented discriminative sparse similarity map (DSS map) to find the candidate with highest score in the evaluation model based upon a matrix, and thus obtain the best tracking results effectively.

Chen et al. [28] presented a robust discriminative local collaborative (DLC). DLC encodes the candidates by an efficient local regularized least square solver with the l_2 norm minimization by using the local image patches of both the target templates and the ones on the background cooperatively. Yang et al. [29] applied Laplacian regularized least squares (LapRLS) to learn a robust classifier for exploiting unlabeled data and preserving the local geometrical structure of the feature space adequately. Qian, Xu et al. [30] presented Subclass Discriminant Constraint (SDC) for visual tracking. The SDC searches for a discriminative subspace to allow linear separation of image blocks that connected with the object and the background. Two dictionaries are constructed and learned in such subspaces for tracking and detection. A transformation matrix and sparse coefficient codes are being found out during dictionary learning. The similarity between the target candidate and the template is determined over sparse coefficients according to the histogram intersection. Hare et al. [31] presented a new adaptive tracking-by-detection framework based on structured output prediction. By using an online structured output SVM [19] learning framework, image features and kernels are comprised easily.

2.2 CORRELATION FILTER

Correlation Filter Based Tracking [32] utilized filters trained on example images to model the appearance of objects. The object is initially selected by a tracking window that centreed on the object in the first frame. By correlating the filter over a search window in next frame tracking and filter training collaborate to track the target. Next the new position of the target is indicated from the location respective to the maximum value in the correlation output. Based on this new location, appearance variation is updated online. Fourier domain Fast Fourier Transform (FFT) is applied to compute correlation to generate a fast tracker. Zhang et al. [33] presented spatial kernel phase correlation based tracker (SPC) that only implements phase correlation filter on adoption of the phase spectrum to estimate the object's translation.

SPC achieves favorable tracking performance as it is more robust to noise and cluster. Liu et al. [34] presented a part-based representation tracker via kernelized correlation filter for visual tracking and Spatial–Temporal Angle Matrix (STAM) that used to select reliable patches from parts via multiple correlation filters to obtain stable patches effectively. Combination of this framework increases the diversity of affine matrices and related candidates. Chen et al. [35] proposed a patch based tracker which adaptively integrates the kernel correlation filters with multiple effective features to handle occlusion challenges . The effective patches are selected by using an adaptive weight selection scheme, and thus improves the robustness of algorithm. Li et al. [36] presented a multi-view correlation tracker, where multi-view model fuses various features and more discriminative features is selected. Fast training and efficient target locating provided by correlation filter framework had enhanced stability of scale variation tracking.

2.3 OTHERS TRACKING METHOD

Sevilla-Lara et al. [37] proposed distribution fields (DFs) to build an image that allows smoothing the objective function and the information about pixel values is keep intact. DFs descriptor has the advantage on slow changes in appearance and pose and minor occlusions. Zhang et al. [38] proposed compressive tracking (CT) to preserve the structure of original image space based on non-adaptive random projections. By adopting a very sparse measurement matrix, features from the foreground and background targets are compressed efficiently. Generative and discriminative appearance models are combined in CT algorithm to encounter for scene variations. Grabner et al. [39] presented an on-line AdaBoost feature selection algorithm that has a advantages on its capability of on-line training, allowing the adaption of the classifier while tracking the object. Thereby appearance changes of the object such as out of plane rotations, illumination changes are handled effectively. Oron et al. [40] proposed Locally Orderless Tracking (LOT) that will estimate the amount of local (dis)order

in the target automatically, allows the tracker specific in both rigid and deformable objects on-line without prior assumptions. Dinh et al. [41] proposed auto exploration on the context information in two semantic terms, which are distracters and supporters by using a sequential randomized forest, an online template-based appearance model, and local features. The tracker able to handle some challenges in tracking in uncontrolled environments with abrupt motion, occlusion, motion blur and frame-cut. Grabner et al. [42] proposed a novel on-line semi-supervised boosting method to reduce drifting problem in tracking applications. The update process is formulated in a semi-supervised fashion as combined decision of a given prior and an on-line classier without adjusting any parameters. Kwon et al. [43] proposed visual tracking decomposition scheme that efficiently highlights the object with drastic changes of motion and appearance or both. Zhuang et al. [44] proposed a shallow and deep collaborative model that collaborates generative model to construct a local binary mask for handling occlusion tracking and a discriminative classifier to learn generic features. Cooperation between of these two models is more favorable to overcome occlusion and target appearance change. Hu et al. [45] proposed a deep metric learning (DML) approach for under the particle filter framework that utilizes a feed-forward neural network architecture to classify the target object and background regions. A set of hierarchical nonlinear transformations in the feed-forward neural network is learned in order to project both the template and particles into the same feature space. The marginal between objects and backgrounds are maximized, and thus that objects are separated from the background regions easily.

2.4 REVIEW ON KCF

The “High Speed Tracking with Kernelized Correlation Filters (KCF)” [46] is the extension to the previous research, which is “Exploting the Circulant Structure of Tracking-by-Detection with Kernals” [21] that is used to handle numerous channels in order to greatly

improve the tracking performance. In [21], the relationship between Ridge Regression with cyclically shifted samples and classical correlation filters is developed and connected together. Instead of using expensive matrix algebra, fast learning is achieved by using *O (n log n)* Fast Fourier Transforms. However, it is limited to single channel. For this reason, [46] is developed to work in multiple channels with using a much simpler diagonalization technique that will be discussed later in this section.

2.4.1 BUILDING BLOCKS

In Ridge Regression, Support Vector Machines [47] is applied to obtain a good result of the performance that is near to more complex methods. The squared error over samples x_i and their regression targets y_i is minimized through the trained a function $f(z) = w^T z$. The regularization parameter λ manages the overfitting, with a closed-form minimizer, w.

$$\min_{w} \sum_{i} (f(x_i) - y_i)^2 + \lambda \|w\|^2 \tag{2.1}$$

$$w = (X^T X + \lambda I)^{-1} X^T y \tag{2.2}$$

In one row of x_i, the data matrix X has one sample and regression target y_i is represented in each element of y.

2.4.2 CYCLIC SHIFTS

Base sample is referred in here by considering an $n \times 1$ vector to represent a patch with the object of interest, expressed as x. A classified is trained to collect both the positive and negative examples of the base samples. One-dimensional translations of this vector are then modelled by a permutation matrix, P which is called cyclic shift operator.

$$P = \begin{bmatrix} 0 & 0 & 0 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \tag{2.3}$$

Small translation is modelled through shifting one element of x is by the $Px = [x_n, x_1, x_2, \cdots, x_{n-1}]^T$. By using the matrix power $P^u x$, a bigger translation is achieved with chaining *u* shifts. A shifting of reverse direction is done with negative *u*. The signal *x* is obtained periodically for every *n* shifts because it is repeated due to the cyclic property. The full set of shifted signals is acquired through Equation 2.4.

$$\{P^u x | u = 0, \dots, n-1\} \tag{2.4}$$

2.4.3 CIRCULANT MATRICES

A regression with shifted samples is calculated by using Equation 2.5, as the rows of data matrix *X*.

$$X = C(x) = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_n & x_1 & x_2 & \cdots & x_{n-1} \\ x_{n-1} & x_n & x_1 & \cdots & x_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & x_4 & \cdots & x_1 \end{bmatrix} \tag{2.5}$$

Equation 2.6 is used to express all circulant matrices diagonally by the Discrete Fourier Transform regardless of the generating vector *x*, without creating vector *x* [48].

$$X = F\ diag(\hat{x})F^H \tag{2.6}$$

F is a constant matrix independent of *x* while $\hat{x}$ represents the generating vector of DFT, $\hat{x} = \mathcal{F}(x)$. The hat symbol ^ is used as shorthand for the DFT vector. *F* is a special matrix that compute any input vector of DFT as $\mathcal{F}(z) = \sqrt{n}Fz$. The eigendecomposition of a general circulant matrix is expressed in Equation 2.6.

2.4.4 COMPILING IT ALL TOGETHER

A non-centreed covariance matrix is taken as the term $X^H X$. Substituting it into Equation 2.6 will yield Equation 2.7.

$$X^H X = F\ diag(\hat{x}^*)F^H F\ diag(\hat{x})F^H \tag{2.7}$$

Factor $F^H F = I$ can be eliminated and cancelling the unitarity of F in many expression and yield Equation 2.8.

$$X^H X = F\ diag(\hat{x}^* \odot \hat{x}) F^H \tag{2.8}$$

The above steps are applied to the full expression for linear regression and most quantities can be put into the diagonal and thus, Equation 2.9 is obtained.

$$\hat{w} = \frac{\hat{x}^* \odot \hat{y}}{\hat{x}^* \odot \hat{x} + \lambda} \tag{2.9}$$

2.4.5 KERNEL TRICK

Kernel trick [49] is applied to allow more robust, non-linear regression function *f(z)*. Kernel trick consists of the following in order to map the inputs of linear problem to a non-linear feature-space $\varphi(x)$:

$$w = \sum_i \alpha_i \varphi(x_i) \tag{2.10}$$

Using alternative representation α as dual space instead of primal space *w*.

$$\varphi^T(x)\varphi(x') = \kappa(x, x') \tag{2.11}$$

Using kernel function κ such as Gaussian to compute in terms of dot-products.

$n \times n$ kernel matrix *K* stores the dot-products between all pairs of samples with elements.

$$K_{ij} = \kappa(x_i, x_j) \tag{2.12}$$

The biggest drawback of kernel trick is complexity elevates when number of samples increases.

$$f(z) = w^T z = \sum_{i=1}^{n} \alpha_i \kappa(z, x_i) \tag{2.13}$$

2.4.6 FAST KERNEL REGRESSION

In order to avoid this kernel trick's weakness, a solution is given to the kernelized version of Ridge Regression [47].

$$\alpha = (K + \lambda I)^{-1} y \tag{2.14}$$

To preserve the circulant structure for a kernel, all dimensions of data must be treated equally. After making the kernel to become *K* circulant, Equation 2.14 is diagonalized in linear case and Equation 2.15 is obtained

$$\hat{a} = \frac{\hat{y}}{\hat{k}^{xx} + \lambda} \tag{2.15}$$

where k^{xx} is the first row of the kernel matrix $K = C(k^{xx})$. A more common kernel correlation is defined as elements of vector $k^{xx'}$ with two arbitrary vectors, x and x', which can be referred as kernel auto-correlation analogy with the linear case. Equation 2.17 is expressed in another way that is equivalent to a dot-product in a high-dimensional space $\varphi(\cdot)$.

$$k_i^{xx'} = \kappa(x', P^{i-1}x) \tag{2.16}$$

$$k_i^{xx'} = \varphi^T(x')\varphi(P^{i-1}x) \tag{2.17}$$

2.4.7 FAST DETECTION

K^Z is defined as asymmetric kernel matrix between all training samples and all candidate patches. It is given by $\kappa(P^{i-1}z, P^{j-1}x)$ as due to the cyclic shifts of base sample *x* and base patch *z* of the samples and patches respectively. The first row of kernel matrix is defined in Equation 2.18.

$$K^z = C(k^{xz}) \tag{2.18}$$

Thus, the regression function for all candidate patches can be computed from Equation 2.13 with

$$f(z) = (K^z)^T \alpha \quad (2.19)$$

Diagonalization of Equation 2.19 will obtain Equation 2.20 in order to determine Equation 2.19 efficiently.

$$\hat{f}(z) = \hat{k}^{xz} \odot \hat{\alpha} \quad (2.20)$$

f(z) is evaluated at all locations that can be referred as a spatial filtering operation over the kernel values k^{xz}. The linear combination of the neighbouring kernel values is obtained from each *f(z)* from k^{xz}, measured by the learned coefficients. Thus, formulation of the *f(z)* can become more effective since this is a filtering operation.

2.4.8 RADIAL BASIS FUNCTION AND GAUSSIAN KERNELS

$$k^{xx'} = \exp\left(-\frac{1}{\sigma^2}\left(\|x\|^2 + \|x'\|^2 - 2\,\mathcal{F}^{-1}(\hat{x}^* \odot \hat{x}')\right)\right) \quad (2.21)$$

The full kernel correlation is computed from Equation 2.21 in only *O (n log n)*.

CHAPTER 3

METHODOLOGY

In this section, we will discuss on the basic concept of high speed tracking with Kernelized Correlation Filters (KCF) [46]. The parameters and idea of optimization technique used to optimize the performance of KCF in our work is presented with initial setting and procedures required. The work flow is presented in a flow chart to highlight the crucial procedures in our framework. The proposed method will be evaluated by its tacking performances, which is OR and CLE, where the calculation for both evaluator is discussed in section below.

3.1 PROPOSED METHOD

As mentioned by the author of the "High-Speed Tracking with Kernelized Correlation Filters", there is greater space of improvement for the algorithm to perform more accurately and robustly. Thus, the proposed method is basically a simple modification on the KCF. Instead of the default values, two parameters are selected to be varied in the optimization process, which are *p* (padding) and *s* (spatial bandwidth). Padding is the extra area surrounding the target while spatial bandwidth is used to predict the response of an imaging system to very small objects which directly related to the size of the image and its object.

The range of values are defined at first in order to determine the global best values that are corresponding to the best result obtained. At first, the original setting for padding and spatial bandwidth is 1.5 and 0.1 respectively. By altering the values of padding from 1 to 2 with increment of 0.1 while 0.05 to 0.5 with increment of 0.05 for spatial bandwidth, and then combining these different sets of values of padding and spatial bandwidth, the OR and CLE will be computed based on each combination. Thus, the global best tracking performance is obtained from all the combinations and recorded for all different sequences. However, for some video sequences, the results obtained cannot achieved any improvement within the

combination of these values. Thus, we increase the range of values to be optimized, where padding is increased from 0.1 to 4.0 with increment of 0.1 while spatial bandwidth is increased from 0.01 to 0.4 with increment of 0.05. This modification is shown with yellow highlighted text at Appendices Section 7.3 Matlab Coding under 7.3.1 run_tracker.m. The reasons of choosing 0.1 and 0.05 as the increment value for padding and spatial bandwidth respectively are because of some limitations, which are time constraint and the value of the OR and CLE are not affected even when the number of decimal is increased further for the increment value. We also need to ensure the time to compute the tracking results to be as fast as possible. This new input value of padding and spatial bandwidth will be called to another subroutine to computed the new window size and to create regression labels, gaussian shaped with a bandwidth proportional to the target respectively. This is shown with yellow highlighted text at Appendices Section 7.3 Matlab Coding under 7.3.14 tracker.m. The changes will eventually affect the tracking results and these results are all recorded and tabulated as shown in Appendices Section 7.2 Table of Results.

With all the global best values that correspond to their best results, we can analyze the global best tracking results and obtained the highest overlap ratio and the lowest centre location error. The analysis will be conducted by using colour mapping method, where the colour will provide the information of the results to us. The density of the colour will be varied according to the values which can provide a better visualization for us to obtain the best parameters with the best tracking performance. The tracking pipeline is presented in this section with a flow chart, where the main procedures conducted to perform the parameters enhancement of the visual tracking is listed out here step by step. The performance evaluator, overlap ratio (OR) and centre location error (CLE) are used to measure the tracking performance of all video sequences because OR and CLE are the most used and common performance evaluators in visual tracking and thus provide an easier way to compare with other tracking method. The

techniques to calculate OR and CLE will be discussed in section below. Nevertheless, in order to evaluate the performance obtained from these optimized values of padding and spatial bandwidth, global best results are selected and compared with the others tracking methods, which the OR and CLE can be obtained from the website of visual tracking benchmark [50]. Comparison is done by using Temporal Robustness Evaluation (TRE), which is used when the tracker is initialized by a given first ground truth bonding box's location of first image frame, and then is evaluated on each segment with tallied overall statistics. The algorithm is run by using Matlab 2012a with the computer's specification of Intel(R) Core (TM) i7-3520M CPU @ 2.90GHz. In our proposed method, there are no mathematical modelling, equations, cost or fitness function are involved or created in this optimization. However, we are manually applying some modifications on the KCF coding in order to obtained the desired improvement, where this optimization can also be said that is done manually. The reasons of doing this simple modification are due to some constraints such as time required to finish this whole research within the due date and low performance of computer.

3.1.1 FLOW CHART

Initially the padding and spatial bandwidth is set to be 1.5 and 0.1. However, this default setting will limit the performance of the tracking. Thus, in our proposed method, padding and spatial bandwidth will be varied corresponding to frame size and the target size in the first frame for every different sequence. First, the algorithm of KCF is modified so that it can run through the different set of combinations of padding and spatial bandwidth. As the first set of combination is finished, the second combination is looped and proceeded the same as before and continue until the last combination. The tracking begins with the first sequence until the last sequences one by one. Next, it will undergo several processes that are similar with the original KCF tracking algorithm, where a model is trained with the image patch at the initial

position of the target with the feature descriptor. Thus, a feature template is created to extract the feature of image patches. Then, the tracking process is started from the first frame until the last frame. The patch is detected at the previous position and the target position with maximum value of tracking performance is created and updated with a new model is trained at the new position. Based on the obtained value, it is interpolated linearly and classified with kernel correlation filters which is similar with the default algorithm by [46].

Thereby, a bounding box with green outline is plotted on the desired target location and the target will be tracked continuously with the newly updated desired tracking location until the end. The overlap ratio and centre location error for each frame is computed and the overall OR and CLE is calculated by averaging all consecutive frames of that particular video sequences. Next, colour mapping analysis is conducted in order to determine the improved results. Figure 3.1 below shows the simple flow chart of the tracking algorithm as discussed above. All the video sequence datasets are obtainable from the Visual Tracking Benchmark [50] under the category of TB-100 sequences.

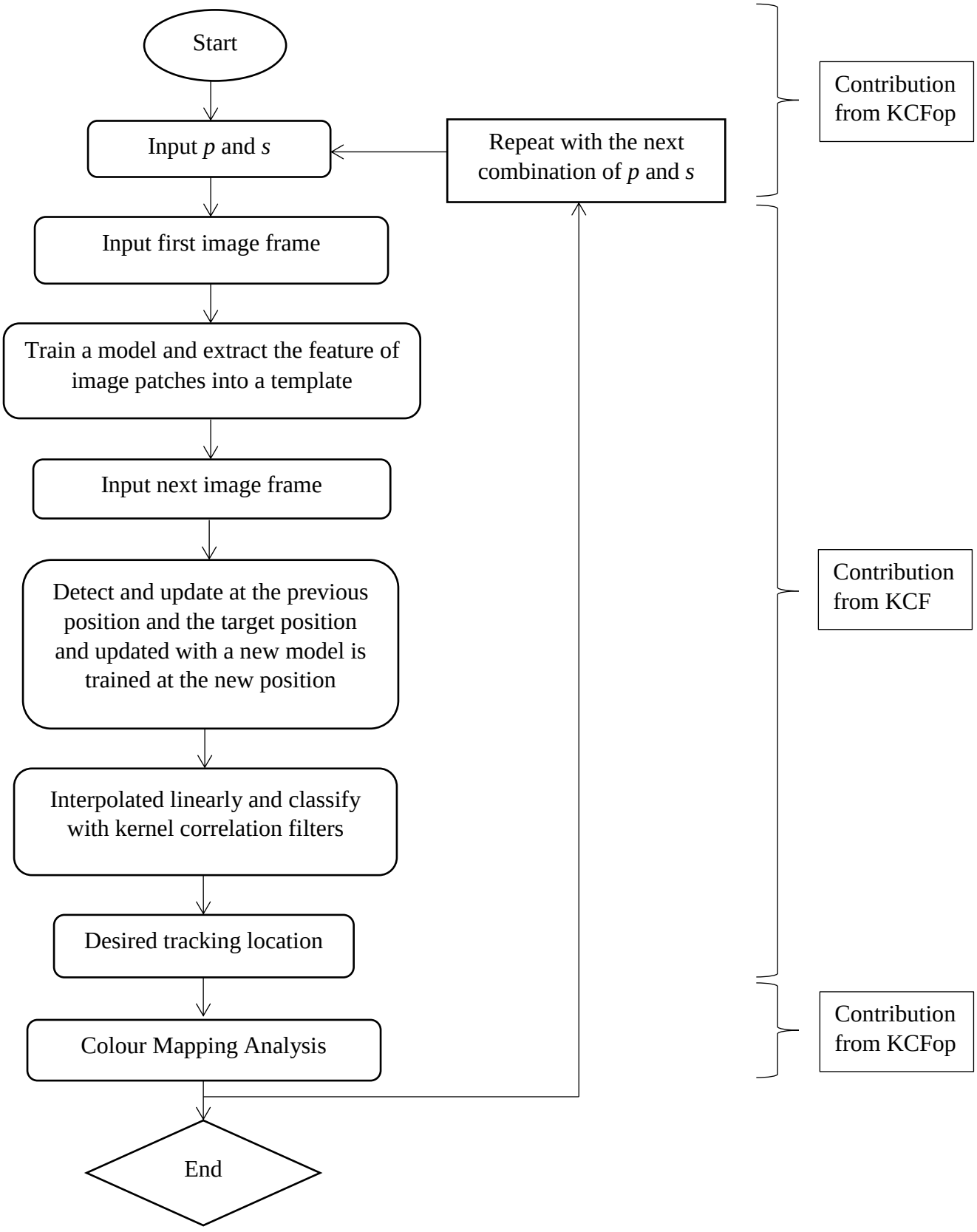

Figure 3.1: The flow chart of the tracking algorithm

3.1.2 PERFORMANCE EVAULATION

Overlap ratio (OR) and centre location error (CLE) are chosen as the performance evaluation in visual tracking. Both are qualitative evaluation and OR and CLE of our proposed method will be compared with other algorithms proposed in this field. Overlap ratio is defined as the percentage of the overlapping area between the region of ground truth and target. OR is calculated based on Equation 3.1. Based on this concept, OR is calculated by using a Matlab command, which is "rectin" to calculate the intersecting area of these two boxes.

$$AOR = \frac{area\{ROI_t \cap ROI_g\}}{area\{ROI_t \cup ROI_g\}} \tag{3.1}$$

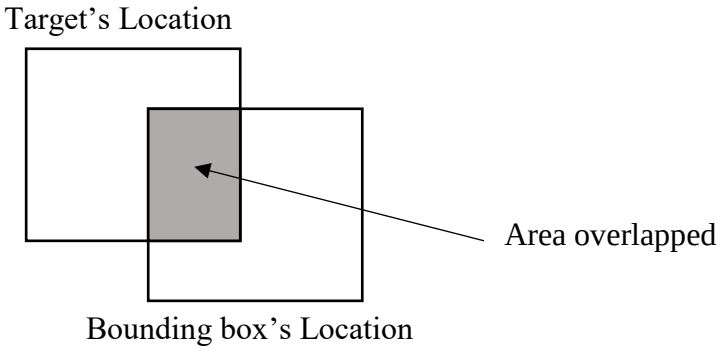

Figure 3.2: Area overlapped between target's location and bounding box's location

Centre location error is defined as the Euclidean distance between the centre location of a target size and the ground truth where it is measured in pixel, as shown in Equation 3.2.

$$CLE = \sqrt{\left(x_t - x_g\right)^2 + \left(y_t - y_g\right)^2} \tag{3.2}$$

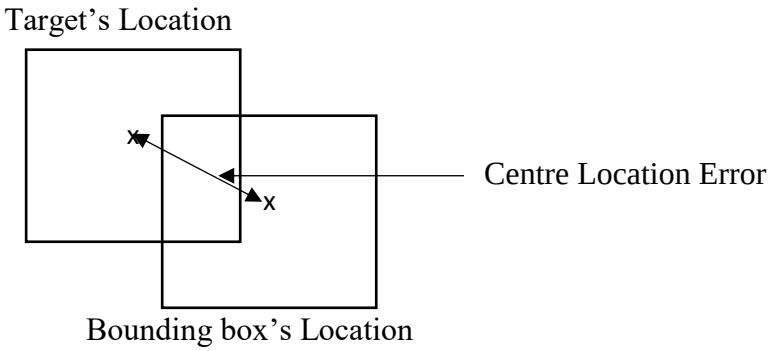

Figure 3.3: Centre location error between target's location and bonding box's location

This concept is basically the same with calculating the distance between 2 points using the *x* and *y* coordinate, where in our case is the coordinates at the centre of the box. Similar to OR, "sqrt()" command is used in Matlab in order to calculate the CLE.

The higher the overlap ratio indicated that the target is tracked more accurately. OR score with more than 0.5 will only be considered as a successful tracking. While for the centre location error, the lower the score it is, the better the tracking is.

CHAPTER 4

RESULT AND DISCUSSION

The results obtained from the optimization and tracking performance of the proposed algorithm will be discussed in this section. A total of 86 sequences have been run through the experiment and the results of the global best performance corresponding with its own padding *p* and spatial bandwidth *s* are recorded. Meanwhile, the global best values of *p* and *s* obtained for each sequence are also discussed and tabulated with respect to their best tracking performance in term of OR and CLE. The results of the proposed algorithm are compared with other algorithms.

4.1 OPTIMIZATION ANALYSIS

As mentioned above, optimization has been conducted and we obtained the global best performance with the its own padding and spatial bandwidth corresponding to its respective video sequence. Once the tracking performances of one sequence is obtained, we had conducted a map data study, where colour scale is applied in order to determine the best tracking results. Table 4.1 and 4.2 are plotted in order to display the colour mapping for one of the example of the results of OR and CLE, which is Walking sequence. In our case, there are 3 colours used as the scale in this colour mapping analysis. The best value, which is corresponding to the highest OR and the lowest CLE will be indicated with light green while the worst value, which is corresponding to the lowest OR and highest CLE will be indicated with dark red. The mean value between the best and the worst value will be indicated with yellow. The density of the colour is varied through all according to the colour scale used. For example, Walking sequence acquires the best tracking result of 0.486 OR and 4.280 CLE at 1.0 padding and 0.05 spatial bandwidth. Thus, after obtaining these values of padding and

spatial bandwidth, Table 7.1 is constructed to record the global best OR and CLE with respect with its own padding and spatial bandwidth for all 86 video sequences.

Table 4.1: Colour map for deciding the padding and spatial bandwidth for Walking sequence in term of overlap ratio

84. Walking - Overlap Ratio										
p \ s	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
1.0	**0.486**	0.440	0.250	0.055	0.017	0.010	0.010	0.010	0.010	0.010
1.1	0.466	0.435	0.252	0.086	0.017	0.010	0.010	0.010	0.010	0.010
1.2	0.472	0.457	0.319	0.101	0.020	0.012	0.010	0.010	0.010	0.010
1.3	0.462	0.441	0.265	0.135	0.022	0.012	0.010	0.010	0.010	0.010
1.4	0.475	0.455	0.327	0.183	0.048	0.018	0.011	0.011	0.011	0.010
1.5	0.471	0.456	0.347	0.211	0.090	0.036	0.011	0.011	0.011	0.011
1.6	0.466	0.445	0.342	0.220	0.101	0.037	0.011	0.011	0.011	0.011
1.7	0.472	0.454	0.356	0.236	0.107	0.054	0.026	0.011	0.011	0.011
1.8	0.459	0.448	0.350	0.260	0.105	0.074	0.027	0.011	0.011	0.011
1.9	0.471	0.453	0.353	0.264	0.115	0.077	0.037	0.012	0.012	0.011
2.0	0.467	0.456	0.355	0.288	0.118	0.086	0.056	0.011	0.011	0.011

Table 4.2: Colour map for deciding the padding and spatial bandwidth for Walking sequence in term of center location error

84. Walking - Centre Location Error										
p \ s	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.50
1.0	**4.28**	4.37	14.57	302.72	382.68	415.30	415.58	416.35	416.45	417.23
1.1	5.17	4.45	13.96	219.35	377.06	415.26	415.26	415.42	416.12	416.38