PERFORMANCE ASSESSMENT OF OBJECT TRACKING ALGORITHMS IN DIM-SMALL INFRARED IMAGE USING OpenCV

Toukir Ahmed

Exam Roll: 610

Class Roll: 2014

Registration Number: 1711

Session: 2017-18

Fazla Rabbi Bindu

Exam Roll: 611

Class Roll: 2012

Registration Number: 1709

Session: 2017-18

Md. Shakil Sarker

Exam Roll: 624

Class Roll: 2020

Registration Number: 1719

Session: 2017-18

Department of Computer Science & Engineering

Mymensingh Engineering College

(Affiliated With University Of Dhaka)

Mymensingh-2208, Bangladesh July-2023

PERFORMANCE ASSESSMENT OF OBJECT TRACKING ALGORITHMS IN DIM-SMALL INFRARED IMAGE USING OpenCV

A THESIS
Submitted In Partial Fulfillment of the
Requirements for the Degree
of
BACHELOR OF SCIENCE
IN
COMPUTER SCIENCE & ENGINEERING

By

Toukir Ahmed
Registration Number: 1711
Session: 2017-18

Md. Shakil Sarker
Registration Number: 1709
Session: 2017-18

Fazla Rabbi Bindu
Registration Number: 1719
Session: 2017-18

Under Supervision of

Prof. Md. Alamgir Hossain
Principal, Mymensingh Engineering College

Department of Computer Science & Engineering

Mymensingh Engineering College
(Affiliated With University Of Dhaka)
Mymensingh-2208, Bangladesh July-2023

Declaration

This is to certify that this final thesis report is submitted by the authors for the purpose of obtaining the degree of Bachelor of Science in Computer Science, and the degree of Bachelor of Engineering in Computer Science and Engineering. We hereby declare that all the instances of work presented in this thesis are original and inspirations for the work that we have made use of have been duly accredited with proper referencing.

Signature of Supervisor	Signature of Authors
Prof. Md. Alamgir Hossain	Toukir Ahmed
	Fazla Rabbi Bindu
	Md Shakil Sarker

Acknowledgement

First of all, we would like to thank Almighty Allah for giving us enough mental and physical strength to complete our thesis properly. We would like to express our sincere gratitude to our supervisor Professor MD Alamgir Hossain for his cooperation, suggestion, guidance, and continuous encouragement through the course of the study. Besides our supervisor, we would like to acknowledge our honorable teachers of Computer Science and Engineering for their encouraging support and discussion. We also thank our parents for their encouragement, support, and attention. We are also thankful to our classmates for their moral support which helped us to accomplish our thesis.

Dedication

To our parents and our family. Both our parents give enough inspiration and encouragement to complete our thesis work.

ABSTRACT

A fundamental computer vision job, object tracking has a variety of uses in robotics, autonomous vehicles, and surveillance. Due to its capacity to take pictures in poor lighting or inclement weather, infrared imaging has recently attracted a lot of interest. However, due to decreased visibility and increased noise, object tracking in dim-small infrared pictures presents particular difficulties.

In the setting of dim-small infrared pictures, this thesis gives a thorough performance analysis of object tracking methods utilizing OpenCV. These tracking algorithms are taken into consideration: Tracking-Learning-Detection (TLD), Kernelized Correlation Filters (KCF), Boosting, Median Flow, Minimum Output Sum of Squared Errors (MOSSE), Multiple Instance Learning (MIL), and Discriminative Correlation Filter with Channel and Spatial Reliability (CSRT).

On a dataset of dim, tiny infrared pictures with a variety of objects and motion situations, the algorithms are assessed. The tracking accuracy, resilience to noise and occlusion, computational efficiency, and capability to handle low-contrast and low-resolution infrared pictures are some of the performance criteria employed.

Each method is developed using the OpenCV library, which offers a variety of computer vision features, to assure the accuracy of the assessment. To get relevant findings, the tests are carried out under carefully monitored settings, and the outcomes are statistically examined.

The results of this study provide important light on the advantages and disadvantages of various object tracking methods in the setting of dim-small infrared pictures. The outcomes can help in the creation of more reliable and precise tracking systems for infrared imaging applications as well as in assisting in the selection of suitable algorithms for certain tracking scenarios.

Overall, this thesis builds upon previous research and advancement in the field to better comprehend object tracking in dim-small infrared pictures. The insights gathered from this work can help advance the capabilities of computer vision systems and object tracking algorithms in difficult imaging environments.

Keywords: Computer Vision, Infrared Image, KCF, CSRT, MIL, TLD, MOSSE, Boosting, OpenCV.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

New issues and challenges are arising as a result of the development of computer emerging technologies, which presents obstacles and challenges for experimenters. As obstacles and challenges increase, a variety of fresh exploration opportunities are also emerging.

When it comes to computer vision , the impact of object tracking cannot be overstated. It's an operation of computer vision where a program detects objects and also tracks their movements in space or across different camera angles[1].

It is used in widespread vary of functions such as surveillance and protection systems,traffic control and analysis, robotics and self sufficient system, automobile driver assistance and self driving, clinical imaging and surgical navigation, cell tracking, sports activities evaluation and participant tracking, visitors manipulate and analysis, face and full-body recognition, image matching,Industrial automation, video compression, augmented and virtual reality, monitoring of wildlife and endangered species etc. [2]

Visual imaging (images of visual band light) is good for feature extraction due to its high feature value.[21] But for night images this imaging technique couldn't find much feature from an image due to low light. Problems like surveillance, rescue missions, wildlife tracking during night and so on can be solved by IR imaging.

1.2 Motivation

Even though there have been significant developments and advancements in this area recently, a sizable number of original algorithms have yet to be created, making it a very challenging topic. It's getting impossible to create a universal tracking algorithm because of the vast array of environmental conditions and other variables.

Additionally, selecting the optimal algorithm depends not only on how it was designed but also on how it was implemented. In order to examine some of the freely accessible techniques, we used the OpenCV library and a variety of difficult videotape sequences. The Python OpenCV package has the algorithms for comparison fluidly set up. With over 3000 algorithms included in it and all the necessary structures and tools for object tracking techniques, it is a widely used library. [01]

1.3 Objectives

The objectives of comparing object tracking algorithms can include:

- Evaluating the delicacy of the algorithms in terms of success point, precision and runtime.
- Assessing the robustness of the algorithms in the presence of in-plane rotation, occlusion, out of view, background clutter, scale variation and fast motion.
- Analyzing the computational effectiveness of the algorithms in terms of runtime.
- Identifying the optimal algorithm for a specific application or dataset.
- Finding potential improvements to existing algorithms..
- Identifying the scenarios where the algorithm fails and trying to understand the cause of the failure.

1.4 Problem Definition

An algorithm's performance cannot be fully captured by a single statistic or collection of criteria. Different algorithms may perform better in other environments or scripts, and the selection of criteria may also vary depending on the particular task or use situation. Delicacy, robustness, and computing efficiency are three factors frequently used to evaluate object tracking systems.

However, it's crucial to keep in mind that an algorithm's performance may also be influenced by the particular dataset or task that it is being estimated on. Different datasets may present unique challenges, such as occlusion, background clutter, varying spatial scales, stir blur, fast stir, low resolution, etc.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The bottom point of the thesis is to evaluate object tracking algorithms using various factors. The same kind of work has already been done, but in a different approach. What we've opted to do is to analyze the strengths and weaknesses of the algorithms used in various industries and make recommendations for the best algorithm in each subject.

2.2 Related Works

The KCF and CSRT algorithms were put to the test against jointly implemented KCF (Kernel Correlation Filter) and CSRT (The Channel and Spatial Reliability Tracker) algorithms by **Akshat Mittal (2021) et al.** The monitoring success rate and tracking consistency were the performance metrics they utilized.
When employed individually, CSRT produces lower FPS output but with higher object tracking precision, whereas KCF produces higher FPS output but with somewhat lower object tracking precision.

They claimed that combining The Channel and Spatial Reliability Tracker (CSRT) and Kernel Correlation Filter (KCF) produced a tracking success rate measured by frame skips that was much higher than either KCF or CSRT alone. The hybrid implementation's tracking speed was faster than CSRT but a little slower than KCF. To develop practical answers for diverse use cases, they must concentrate more on trials with various combinations of these methods.[02]

To follow a person's head and entire body using OpenCV's overhead view, **Kaleem Ullah et al (2019)** examined seven different tracking algorithms (such as Boosting [12], MIL [10], KCF [11], TLD [02], MedianFlow [15], MOSSE [15], and CSRT[15]). They discovered that, when it came to complete body tracking, all algorithms produced worse outcomes to head tracking. On the other hand, the CSRT and Boosting algorithms did well, with a head tracking accuracy of 85% and 83%, respectively. The paper's flaw is that it only compares single-person tracking; it does not compare tracking for several individuals.[03]

In an effort to compare the tracking algorithm implementations found in the OpenCV library, **Peter Janku (2016) et al.** They employed three feature detectors—SURF (Speeded Up Robust Features), SIFT (Scale-Invariant Feature Transform), and ORB (Oriented FAST and Rotated BRIEF)—as well as three pure trackers—MIL (Multiple Instance Learning), Boosting, and MedianFLow—as well as one sophisticated tracking framework, TLD (Tracking-Learning-Detection) [02]. They demonstrated that MIL and Boosting had higher success rates than TLD. They also demonstrated that TLD is as sluggish as SIRF and SURF for the time it takes to process a single frame, whereas the other algorithms are quicker.
The paper's drawback is that a thorough study of the algorithms is not provided, despite the fact that doing so is crucial given the flaws and restrictions in method implementation that have been discovered. [04]

The experimental evaluation of several object tracking algorithms, including TLD, Boosting, MIL, KCF, and MedianFlow, was done by **NS Raghava et al. in 2020** with the goal of addressing object tracking-related difficulties. With the exception of quick motion and poor resolution, KCF proved to be the most effective in resolving all issues that arose. Although KCF outperformed TLD in the event of inadequate resolution, KCF demonstrated to be the most effective strategy for resolving the bulk of the testing difficulty aspects. MIL is the slowest algorithm when compared in terms of time using the Frames Per Second (FPS) parameter, whereas MedianFlow comes out on top. Due to the TLD's poor pace, which is thought to be brought on by its implementation using OpenCV, which lacks optimizations, a surprise result was achieved. The paper's drawback is that it does not provide a thorough analysis of how these algorithms are used, how their efficiency are calculated or how other strategies might be used. [05]
Due to routine item change, diversity in scene size, obstructions, appearance variants, sensation of self movement, and light changes, **Mukesh Tiwari (2017) et al.** suggested object recognition and tracking as one of the fundamental areas of exploration. The crucial step in object tracking is

feature determination in particular. It is known for having several ongoing uses including video reconnaissance and vehicle insight. Tracking focused on object evolution and appearance in order to solve the detection problem. The tracking calculation, which smooths out the video arrangement, takes up the majority of the computation. Again, very few solutions make advantage of the previously available information regarding object shape, shading, surface, etc. This investigation discusses and breaks down tracking calculations that consolidate over specified object bounds. This paper's goal is to analyze and review previous methods for object tracking and detection using video groups at various stages. Recognize the gap and suggest a different approach to dealing with enhancing the tracking of objects over video outlines.[06]

A novel connection between an online tracker and a trained detector was put out by **Gustav Häger (2016) et al.** to enable real-time people tracking via an unmanned aerial vehicle (UAV). UAVs flying normally are used to record an unrestricted dataset. With the use of the Kalman Filter, the tracker and detector combine the predicted location and size of the monitored individual for a more accurate assessment. [07]

On frontal view datasets from MOT15 and MOT16, Laura Leal-Taixé (2017) et al. examined several state-of-the-art trackers. The findings showed that most tracking systems have problems with speed and accuracy as the number of persons rises. The majority of trackers are having problems with varied colors and lighting situations, which causes false positives and false negatives, which is another problem that has been discovered. [08]

2.3 Research Summary

Each publication explores a few algorithms for comparison after examining a number of related studies. The summary of the related works we have studied is given below.

No	Author	Year	Algorithms	Best Performance
01	NS Raghava, Kushagra Gupta, Ishita Kedia and Anirudh Goyal	2020	TLD, Boosting, MIL, KCF, Median Flow	KCF
02	Akshat Mittal, Suryansh Pratap Singh, Manas Gupta	2021	KCF, CSRT, KCF+CSRT	CSRT
03	Kaleem Ullah, Imran Ahmed, Misbah Ahmad and Iqbal Khan	2019	BOOSTING, MIL, KCF, TLD, Median Flow, MOSSE	CSRT
04	Mukesh Tiwari, Dr.Rakesh Singhai	2017	Kalman filtering Algorithm, Recursive Bayes Filtering, MHT algorithm, Matching region of interest in video, Expression & location of object, optimal gradient decline, Positive & negative training values, Shape Representation using intensity, Gradient Descent Algorithm, Hough Transform	Hough Transform

No	Author	Year	Algorithms	Best Performance
01	NS Raghava, Kushagra Gupta, Ishita Kedia and Anirudh Goyal	2020	TLD, Boosting, MIL, KCF, Median Flow	KCF
02	Akshat Mittal, Suryansh Pratap Singh, Manas Gupta	2021	KCF, CSRT, KCF+CSRT	CSRT
03	Kaleem Ullah, Imran Ahmed, Misbah Ahmad and Iqbal Khan	2019	BOOSTING, MIL, KCF, TLD, Median Flow, MOSSE	CSRT
05	Peter Janku, Karel Koplik, Tomas Dulik and Istvan Szabo	2016	SURT, SIFT, ORB, MIL,BOOST,MF,TLD	BOOST,MIL

Table-2.1: Research Summary

2.4 Scope of the Problem

The literature review shows that above researchers compared object tracking algorithms by their performance based on challenges such as occlusion background and clutter attributes for visual imagery. And their results show promising results. We found that these algorithms can be tested for night images and their performance for night images needs to be evaluated. We will use IR image sequence for this comparison.

2.5 Challenges

In order to compare object tracking methods, we must first concentrate on a few key issues that occur in a variety of settings and circumstances. The following describes these issues.

- **Occlusion(OCC):** It can be challenging to monitor an object when it is partially or completely covered by another object.
- **Background and Clutter(BC):** In cluttered settings, it might be challenging to tell the object of attention apart from the background and other objects.
- **Scale Variability(SV):** In a video, objects may appear at various scales, making it challenging to maintain reliable tracking.
- **Motion Blur(MB):** Fast-moving objects may appear blurry in a video, making it challenging to precisely identify and track them.
- **Illumination Variations(IV):** It might be challenging to keep continuous track of an object when illumination conditions change.
- **Shape Deformation(DEF):** Consistent tracking might be challenging if the item of interest is flexible or able to alter shape.
- **Adaptation to novel circumstances:** Some object tracking techniques might not be very universal.

There are various issues we will encounter when implementing object tracking algorithms. They are.

- Acquiring a huge data collection.
- After it has been collected, manage the data.
- Our biggest issue with this study is the data collection process.
- Another issue is the disparity between experimental outcomes and outgrowth.

CHAPTER 3

PROPOSED SCHEME

3.1 Introduction

We will attempt to compare algorithms in this work from various angles and with various properties. Algorithms will be assessed using data sets that have been gathered from various sources. The assessment results will then be contrasted with one another, and the results of the comparison will be included in this report.

3.2 Research Subject and Instrumentation

We are comparing algorithms for single object tracking of infrared imaging.[17]

Device Configuration:

Title	Description
Processor	AMD Ryzen 7 3750H with Radeon Vega Mobile Gfx 2.30 GHz
Installed RAM	8.00 GB (7.44 GB usable)
System Type	64-bit operating system, x64-based processor
OS	Windows 11 Home Single Language
OS Version	22H2
Device Manufacturer	ASUSTeK COMPUTER INC

Table-3.1: Device Configuration

Python 3.9.6: The most recent stable version of the computer language, Python 3.9.6, is what we utilize. On October 5, 2021, it (Python 3.9.6) was released. What's more, it offers the package and library that we need to do the tests and experiments.

OpenCV 4.7.0.68: OpenCV 4.7.0.68 is a version of the OpenCV library for computer vision. This version was released on 2021-06-25. It includes various new features, improvements and bug fixes.

Opencv-contrib 4.7.0.68: A dedicated module called OpenCV contrib is available in the Python programming language, where OpenCV is mostly used for real-time computer vision. The package known as OpenCV-contrib contains extra modules not included in the main OpenCV package.

This package has a particular version, OpenCV-contrib 4.7.0.68, which was published on June 25, 2021. This version includes more modules including SIFT, SURF, and text detection on top of OpenCV 4.7.0. Additionally, it has performance enhancements, bug corrections, and new features.[09]

Algorithms Description:

- **MIL:** Multiple Instance Learning is referred to as MIL. Multiple instance learning research and interest have exploded as massive datasets are increasingly being utilized to tackle challenging problems nowadays. [10]. A classifier is taught for MIL object tracking using a set of positive and negative samples of the target item. A tracking method is then used to track the item as it moves across the frames after the classifier has been applied to each frame of the video to find instances of the object.
  It also permits the use of data with shoddy labeling, which is a benefit. As a result, it has been applied to many different tasks, including text classification, content-based picture retrieval, and pharmacological activity prediction. It is a supervised learning approach used when the training set of labels is incomplete. All training examples in supervised learning are each given a unique label. However, MIL[12] just labels a bag of

occurrences. A minimum of OpenCV 3.0.0 is needed in order to implement the MIL algorithm[13].

- **KCF**: Kernelized Correlation Filters are referred to as KCF. This approach tracks the object using a kernelized correlation filter and is based on the correlation filter framework. It is renowned for its quick response times and resistance to distortion and occlusion. BOOSTING and MIL, two algorithms, are combined to create it. The idea behind the approach is that a collection of photos pulled from a "bag" using the MIL method has a lot of overlap. Applying correlation filtering to these regions enables highly accurate tracking of an object's motion and future location prediction. [11]
  The Fourier transform's shift invariant attribute is used by the KCF algorithm[12].
  KCF is regarded as a real-time tracking algorithm because of its effectiveness and precision. A minimum of OpenCV 3.1.0 is needed to implement the KCF algorithm[13].

- **CSRT:** Discriminative Correlation Filter with Channel and Spatial Reliability is referred to as CSRT. A technique for tracking objects in video sequences is the CSRT object tracking algorithm. It is a variation of the well-known KCF (Kernelized Correlation Filter) algorithm that enhances the efficacy of the correlation filter by integrating channel and geographical reliability information. This makes it possible for the system to more effectively manage changes in illumination and object appearance, leading to more precise and reliable tracking. The CSRT algorithm is regarded as one of the most cutting-edge approaches to object tracking since it has been demonstrated to perform effectively in a variety of difficult tracking circumstances.[15] OpenCV 3.4.2 is the minimal version needed to implement the CSRT algorithm[13].

- **BOOSTING:** Using the machine learning ensemble approach of "boosting," object tracking systems can perform better. The fundamental principle underlying boosting is to merge a number of "poor" classifiers into a single, more reliable classifier.
  By integrating several theories about the position of the item in each frame of the video, boosting may be used to increase the accuracy of the tracking algorithm in the context of

object tracking. These hypotheses may be produced by a variety of techniques, including deep learning-based techniques, feature-based tracking, and template matching.
The bare minimum version of OpenCV 3.0.0 is needed to implement the BOOSTING algorithm[13].

- **MOSSE:** The mean-shift algorithm and the idea of adaptive correlation filters are combined in the well-known object tracking technique known as MOSSE (Mean-Shift based Online Object Tracking using Adaptive Correlation Filters).

  The initial bounding box for the item to be tracked in the first frame of the video is chosen by the MOSSE algorithm. With the help of the pixels included in this bounding box, it then develops an adaptive correlation filter for the object. The reaction of the item in each succeeding frame of the video is computed using the correlation filter. The mean-shift technique is then applied to the answer in order to update the bounding box for the item.

  Real-time tracking algorithms like the MOSSE algorithm have been proved to perform at the cutting edge in terms of computing efficiency. Additionally, it can withstand changes in lighting and size that can affect how an object looks. The MOSSE method may be utilized in a variety of applications, including security, robotics, and autonomous systems. It is also quite effective in terms of compute and memory utilization.[15] The MOSSE algorithm must be implemented using at least OpenCV 3.4.1[13].

- **MedianFlow:** Based on the idea of optical flow, the Median Flow method is a powerful object tracking technique. The algorithm's fundamental goal is to estimate an object's mobility from one frame to the next by comparing an object's characteristics across frames.
  The Lucas-Kanade optical flow technique is used by the Median Flow algorithm to first discover feature points in the first frame of the video and then estimate their mobility in following frames. The object's bounding box is then updated using the estimated motion.

An item is represented by a bounding box when utilizing the median flow technique, and the object's motion is then approximated between subsequent frames using sparse optical flow. [05]

OpenCV 3.0.0 at the very least is needed in order to implement the Median Flow algorithm[13].

- **TLD:** TLD is a framework for monitoring any item in a video for an extended period of time [18]. It is referred to as a framework rather than as a tracking technique since the three distinct subtasks of tracking, detection, and learning are carried out by various components of the same process. The method will become more resilient to errors as a result of the target item being monitored and learnt concurrently [19]. According to the frame of reference, the tracker calculates the target's motion and forecasts its location in the upcoming frame. To confine all prior appearances to a specific place, the Detector analyzes the image.

  To ensure that future errors are prevented, The learning element foresees the detector's mistakes and generates a large number of training samples [20]. The TLD method is renowned for its great accuracy, resilience, and capacity to deal with many objects and size changes. Its performance can be impacted by setup and parameter adjustment, and it can be computationally costly.

3.3 Data Collection Procedure

Occlusion, poor resolution, and other characteristics can all affect how well an object is tracked. [14] There are several requirements for these qualities. We make the decision to compare the algorithms using some of the important characteristics. For example, OCC (Occlusion), DEF (Deformation), MB (Motion Blur), FM (Fast Motion), IPR (In Plane Rotation), OPR (Out of Plane Rotation), LR (Low Resolution), BC (Background Clutters), OV (Out of View), etc. [4,14] We will compare algorithms with an appropriate data set that satisfies our needs based on these characteristics. We employ Science Data Bank and LLVIP datasets for infrared dim-small aircraft and visual-infrared image pair respectively. [16

3.4 Statistical Analysis

Each frame's ground truth for the image sequence (video) will be manually chosen from the dataset (image). The image sequence will be examined, and information on the bounding rectangle will be gathered and saved.

Evaluations of success, accuracy, and time requirement will be computed using the results of the experiment and test.

Success Evaluation :

We'll apply the algorithm below to every frame in each video: $C_{suc}(f) = |r_t \cap r_g|/|r_t \cup r_g|$ where $C_{suc}(f)$ is a success criterion function for frame f; r_t and r_g are the bounding rectangles supplied by the tracker and ground truth, respectively.[05]

Precision Evaluation:

We have chosen to utilize the scale of the rectangle derived by the tracker taking into account the ground truth as a measure of algorithm precision, whereas the optimal precision is equal to 1 if we apply this formula: Where $C_{prec}(f)$ is the precision criterion function for the presently being processed frame f, it has the formula $C_{prec}(f) =|((r_t - r_g)*100)|/r_g$

Time Demands Evaluation:

We suggest measuring time for each frame in order to impartially assess the time needed of each algorithm: The time it took to process the current frame f is given by the formula $C_{tim}(f) = t$ where $C_{tim}(f)$ is the algorithm's time demands criterion function. [05]

3.5 Research Methodology

We are showing our research methodology by a pictorial view.

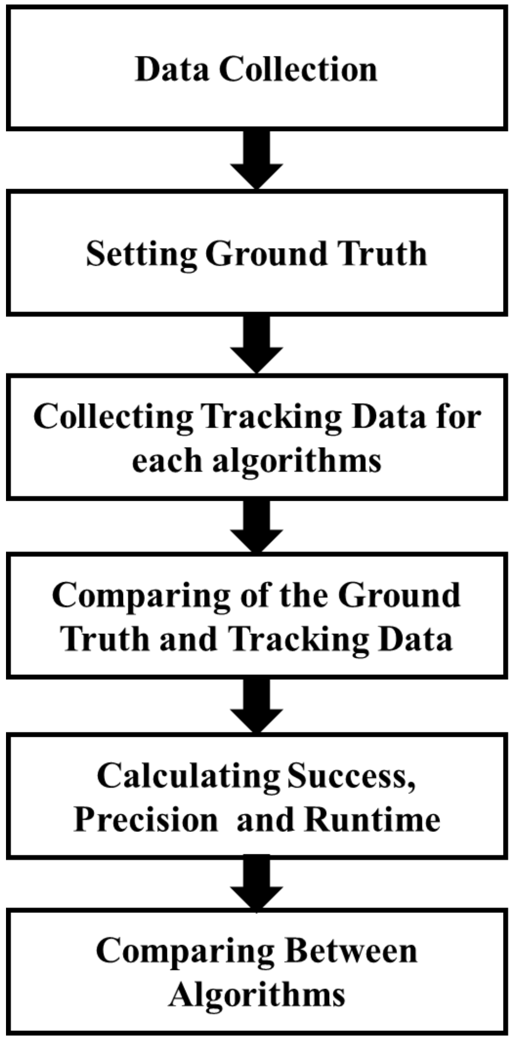

Fig-3.1: Workflow

CHAPTER 4

EXPERIMENTAL RESULT

4.1 Experimental Setup

We picked datasets of dim small infrared pictures based on the following issues encountered when tracking objects, which are subsequently used to assess the performance of dim small infrared images.

Abbr.	Problem Name	Descriptions
BC	Background Clutters	Similarities in color or texture in the background near the target and the target itself.
FM	Fast Motion	The motion of the ground truth is larger than the limit
IPR	In-Plane Rotation	The target rotates in the image plane
OV	Out-of-View	Some portion of the target leaves the view
LR	Low Resolution	The number of pixels inside the ground truth bounding box is below limit
OCC	Occlusion	The target is partially or fully occluded
SV	Scale Variation	The ratio of the bounding boxes of the first frame and the current frame is out of the range

Table 4.1: Problem Description

4.2 Result Analysis

Success, Precision and Runtime of different algorithms for In-Plane Rotation

Tracker	Success	Precision	Time (sec)
BOOSTING	0.329	0.354	12.87
MIL	0.215	0.598	50.52
KCF	0.223	0.598	60.34
TLD	0.268	0.741	82.29
MEDIANFLOW	0.321	0.764	88.52
CSRT	0.177	1.275	105.3
MOSSE	0.262	0.629	111.71
KCF + MIL	0.235	0.354	25.83

Table 4.2: Performance in case of IPR

Success, precision and runtime of different algorithm for Out of View

Tracker	Success	Precision	Runtime(Sec)
BOOSTING	0.268	1.077	3.57
MIL	0.129	1.077	9.54
KCF	0.466	2.224	10.16
TLD	0.203	0.358	15.08
MEDIANFLOW	0.353	1.943	16.44
CSRT	0.429	1.462	20.31

MOSSE	0.354	1.647	21.42
KCF + MIL	0.254	0.851	5.15

Table 4.3: Performance in case of OV

Success, precision and runtime of different algorithm for Scale Variation

Tracker	Success	Precision	Runtime(Sec)
BOOSTING	0.406	0.56	12.91
MIL	0.353	0.56	51.87
KCF	0.426	0.675	55.57
MEDIANFLOW	0.055	22.776	70.94
CSRT	0.265	0.968	86.91
MOSSE	0.452	0.598	92.47
TLD	0.056	19.899	64.75
KCF + MIL	0.054	0.168	24.42

Table 4.4: Performance in case of SV

Success, precision and runtime of different algorithm for Occlusion

Tracker	Success	Precision	Runtime(Sec)
BOOSTING	0.195	2.108	12.77
MIL	0.394	2.108	42.49
KCF	0.487	2.009	49.08
MEDIANFLOW	0.081	19.188	61.56

CSRT	0.414	2.651	74.31
MOSSE	0.457	2.255	80.55
TLD	0	0	0
KCF + MIL	0.425	1.773	27.21

Table 4.5: Performance in case of OCC

Success, precision and runtime of different algorithm for Fast Motion

Tracker	Success	Precision	Runtime(Sec)
BOOSTING	0.067	0.191	7.38
MIL	0.185	0.191	31.42
KCF	0.132	0.132	32.55
MEDIANFLOW	0.004	0.312	46.97
CSRT	0.056	0.266	63.07
MOSSE	0.151	0.153	64.17
TLD	0	0	0
KCF + MIL	0.007	0.087	24.04

Table 4.6: Performance in case of FM

Success, precision and runtime of different algorithm for Background Clutter

Tracker	Success	Precision	Runtime(Sec)
BOOSTING	0.048	0.087	6.67
MIL	0.054	0.087	27.81
KCF	0.019	0.08	28.1
MEDIANFLOW	0	27.137	68.28
CSRT	0	0.087	76
TLD	0.015	0.923	62.02
Mosse	0	0	0
KCF + MIL	0	0.087	21.54

Table 4.7: Performance in case of BC

Success Plots of different algorithms in case of different problems such as occlusion, fast motion, out of view, background clutter, in-plane rotation, scale variation :

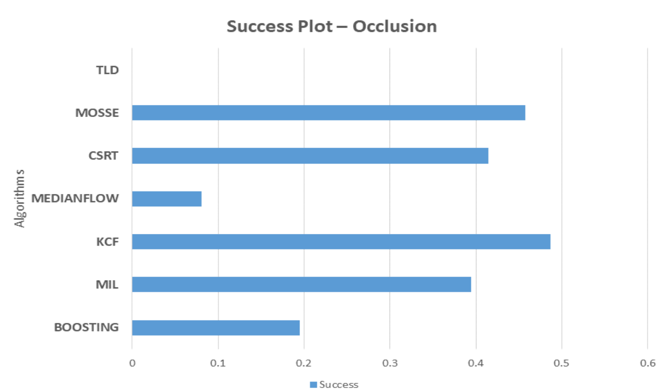

Fig-4.1: Success Plot - Occlusion

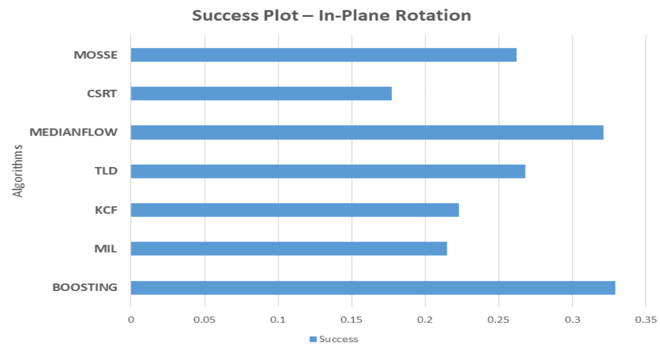

Fig-4.2: Success Plot - In-Plane Rotation

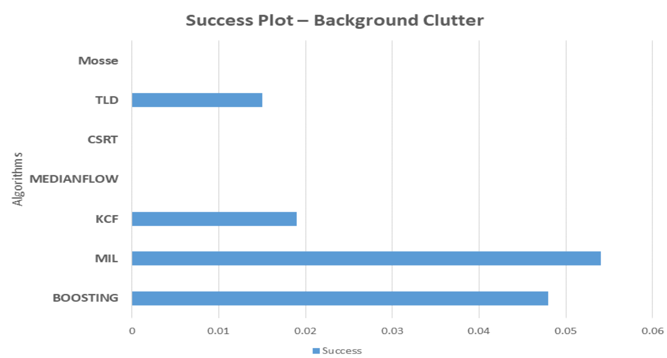

Fig-4.3: Success Plot - Background Clutter

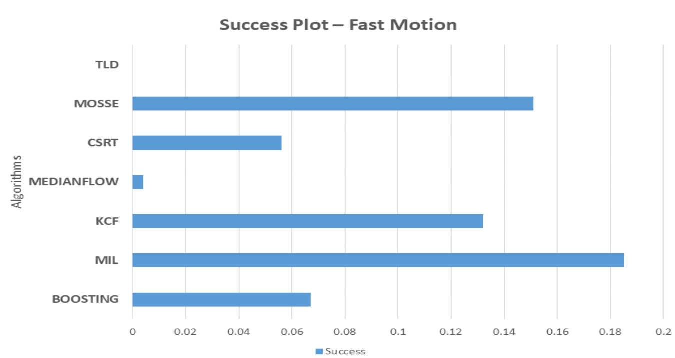

Fig-4.4: Success Plot - Fast Motion

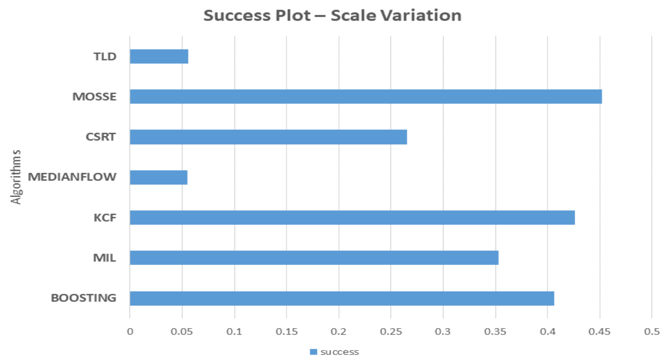

Fig-4.5: Success Plot - Scale Variation

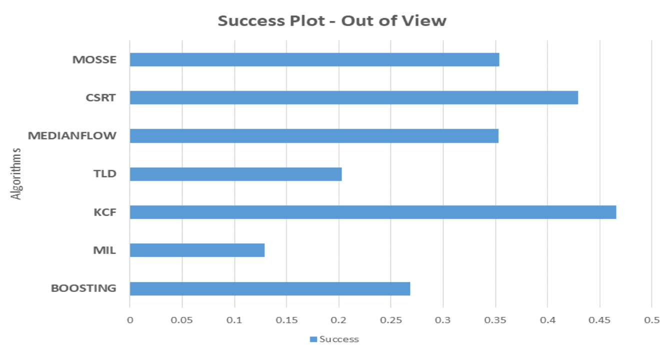

Fig-4.6: Success Plot - Out of View

Precision Plots of different algorithms in case of different problems such as occlusion, fast motion, out of view, background clutter, in-plane rotation, scale variation :

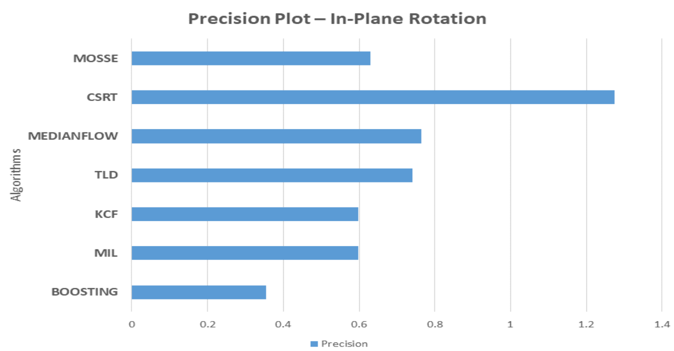

Fig-4.7: Precision Plot- In Plane Rotation

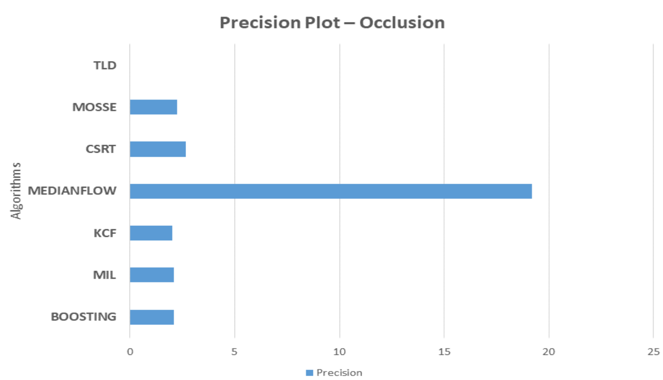

Fig-4.8: Precision Plot- Occlusion

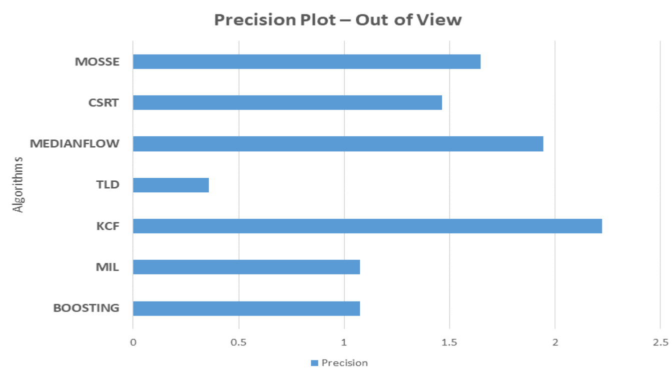

Fig-4.9: Precision Plot- Out of View

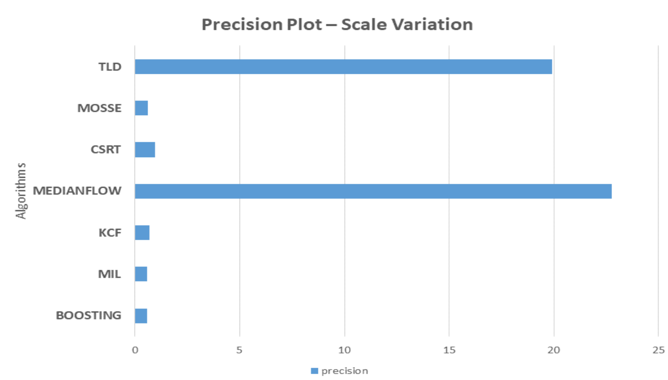

Fig-4.10: Precision Plot- Scale Variation

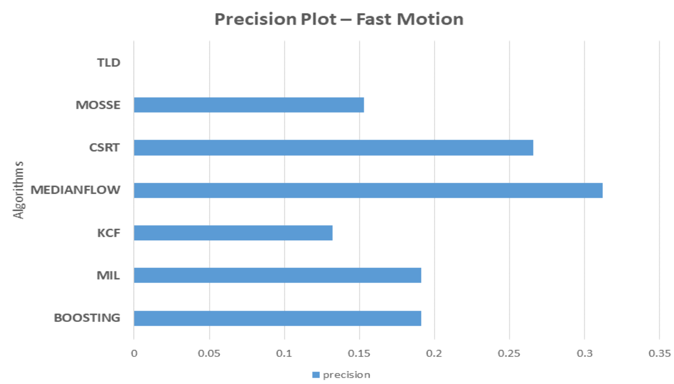

Fig-4.11: Precision Plot- Fast Motion

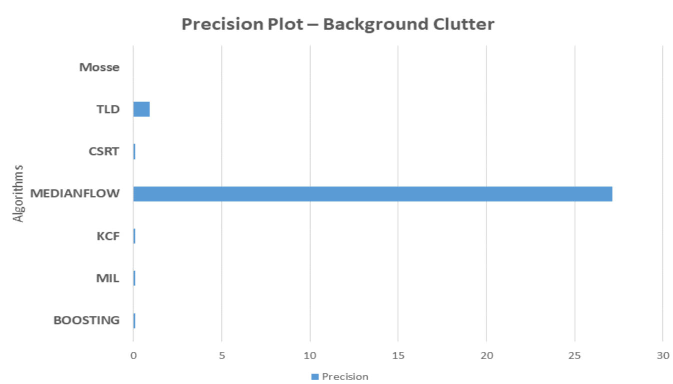

Fig-4.12: Precision Plot- Background Clutter

Runtime Plots of different algorithms in case of different problems such as occlusion, fast motion, out of view, background clutter, in-plane rotation, scale variation :

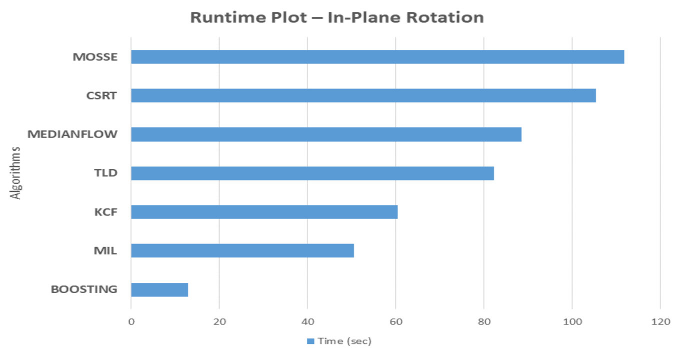

Fig-4.13: Runtime Plot- in plane rotation

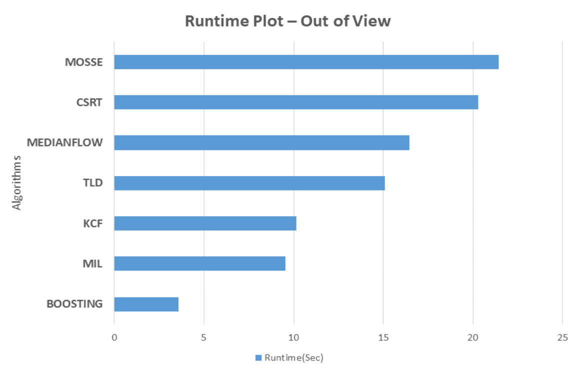

Fig-4.14: Runtime Plot- Out of View

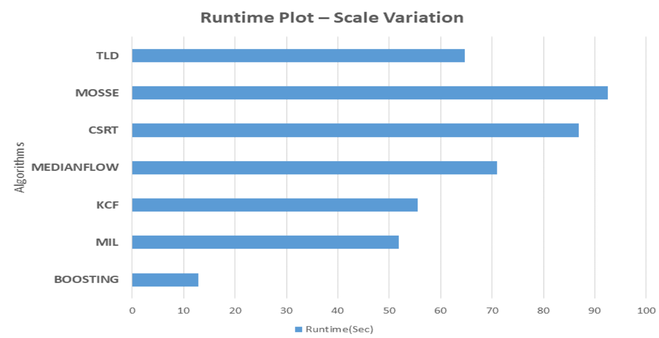

Fig-4.15: Runtime Plot- Scale Variation

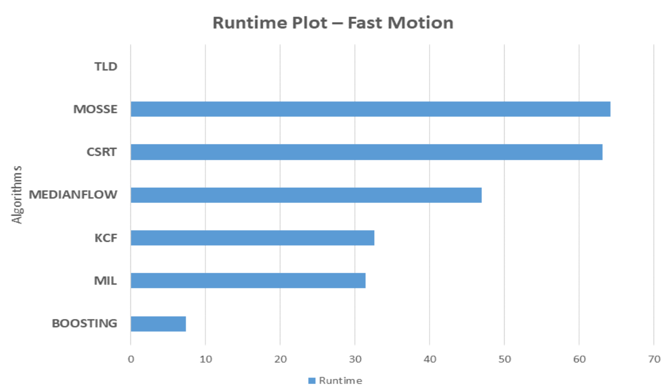

Fig-4.16: Runtime Plot- Fast Motion

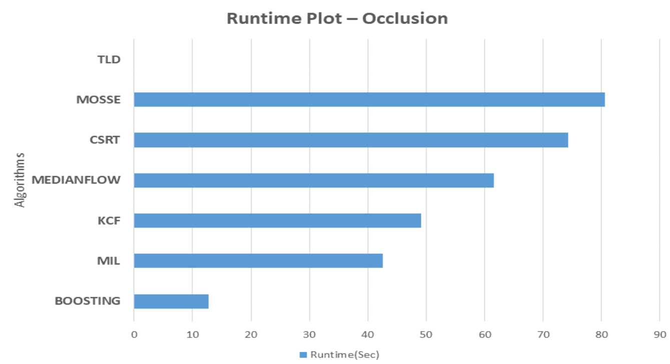

Fig-4.17: Runtime Plot-Occlusion

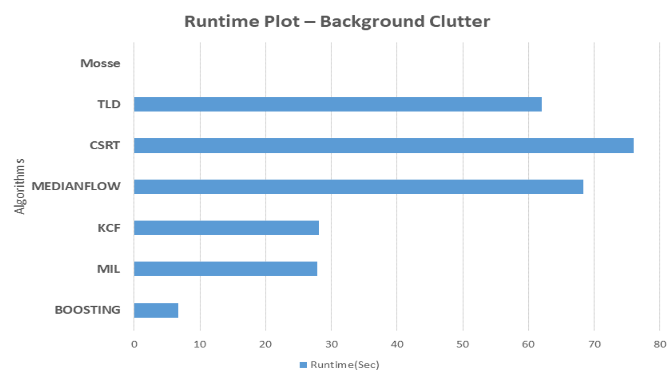

Fig-4.18: Runtime Plot- Background Clutter

4.3 Discussion and Recommendations

Boosting shows a better performance in the context of In plane rotation. In the case of Out of View, KSF and CSRT perform better in success evaluation, whereas Boosting and MIL do well in accuracy assessment and runtime. BOOSTING is quick to run for problems involving scale variation, but interestingly MOSSEE outperforms both KCF and BOOSTING in terms of success rates. The results of CSRT and KCF in the evaluation of precision are evident. KCF yields a respectable success and precision result for the occlusion problem, as expected, but it falls short of Boosting and MIL in terms of runtime performance. TLD was expected to perform well for Fast Motion, however the graph reveals that MIL, KCF, and Boosting perform better instead in terms of success point, accuracy, and runtime. In the context of Background Clutter, MIL and BOOSTING outperform in terms of success, whilst KCF and MIL outperform in terms of precision, and BOOSTING and MIL outperform in terms of runtime. Background Clutter, MIL and BOOSTING outperform in terms of success, whilst KCF and MIL dominate in terms of precision, and BOOSTING and MIL outperform in terms of runtime. When the findings of KCF and CSRT are combined, we observe that the precision evaluation for OV and OCC is pretty outstanding, and they perform better together.

The success chances for specific barriers differentiate for small, dim pictures. A lot of cases where the more recent more potent algorithms do well are situations where the preemptive algorithm Boosting does not. However, average performance for issues yields striking outcomes. The average success scores for the algorithms KCF, MOSSEE, MIL, CSRT, Boosting, and TLD are 0.292, 0.279, 0.238, 0.223, 0.219, and 0.09, respectively. Therefore, for dim tiny infrared pictures, KCF has a higher average success point at background clutter, occlusion, out of view, fast motion, in plane rotation, and scale change.

In order to increase accuracy and success rates, we attempt to combine many methods. In certain circumstances, it boosts tracking outcomes for algorithms with poor performance but lowers the success rate for algorithms with relatively excellent outcomes.

The key issue is that because the tracking item is small and the scenario is poor resolution, it is challenging to extract the data from the bounding region required to continually detect the object. Since we cannot increase the resolution for track objects, another approach must be used to find a solution.

Here is a possible approach to overcome the problems during object tracking. It takes a recognizing algorithm to identify the lost item by examining the predicted path that is achieved by numerical analysis of the object trajectory, then the relative frame difference of the object will be assessed. Thus the tracking algorithm will be tracked further.

CHAPTER 5

CONCLUSION AND FUTURE WORKS

5.1 Conclusion

We've carried out a thorough evaluation of the object tracking algorithms' performance in the demanding setting of tiny, dim infrared pictures. The goal was to assess these algorithms' precision and resilience in situations where poor lighting and low picture resolution present serious difficulties. The conclusions in this paper highlight the benefits and drawbacks of the examined algorithms and offer insightful guidance for further research and advancement in this field. Using a dataset that was especially selected for this work, these algorithms were carefully put into practice and assessed. In the assessment process, their performance was compared based on a number of measures, including tracking accuracy, speed, and robustness against occlusions and sudden changes in motion.

Our performance evaluation's findings showed that several object tracking algorithms work well in tiny, dim infrared pictures. Notably, tracking accuracy and resistance to difficult situations were significantly improved by algorithms that make use of resilient features, adaptive motion models, and probabilistic frameworks. The fact that no particular algorithm emerged as being universally superior in all cases highlights the necessity of choosing an algorithm based on the particular requirements and limitations of the application.

Although our results add to our understanding of object tracking in dark, tiny infrared pictures, it is important to recognize the limits of our investigation. The generalizability of the findings is constrained by the small sample size, potential subjectivity in the assessment measures, and lack of ground truth annotations. Additionally, the claimed performance of the algorithms may have been impacted by computing resource constraints and parameter adjustment difficulties.
Future research in a number of important areas is advised in order to get around these

restrictions. A more complete knowledge of algorithm performance will be feasible through expanding dataset size and variety, adding challenging circumstances, and upgrading assessment measures. To improve the precision and effectiveness of object tracking algorithms in practical applications, it should be investigated to integrate deep learning approaches, real-time implementation, and hardware optimization.

The discipline will also advance by tackling issues with long-term tracking, robustness analysis in the presence of occlusions and scale differences, and researching transfer learning and domain adaptation strategies. Furthermore, a more uniform assessment framework will be made possible by conducting comparison analysis using cutting-edge techniques and constructing benchmark algorithms particularly created for dim tiny infrared pictures.

In a nutshell, this thesis paper has presented a thorough evaluation of the effectiveness of numerous object tracking algorithms in the setting of tiny, dim infrared pictures. The outcomes have opened new possibilities for further study and progress while also highlighting positive aspects and flaws of various algorithms. Researchers can work toward implementing more precise and reliable object tracking systems for a variety of applications, including inspection, autonomous navigation, and target recognition in low-light small-infrared image environments by addressing the identified limitations and building on the findings presented in this report.

5.2 Future Works

The intent of this thesis report is to examine and assess how well different object tracking algorithms function when used with tiny, dim infrared pictures. There are a number of opportunities for further inquiry and refinement, even if the current research provides a thorough review of the current methods. This section describes prospective next projects that could broaden the study's reach and influence.

1. **The incorporation of Deep Learning Techniques:** Integrating deep learning methods into object tracking algorithms for tiny, dim infrared photographs is a relevant domain for future study. Recurrent neural networks (RNNs), convolutional neural networks (CNNs), and attention processes may all be used to increase detection and tracking accuracy, particularly in challenging low-light conditions. Look into and put into practice cutting-edge deep learning architectures created particularly for tracking and reviewing infrared photographs.
2. **Extension and variety of the Dataset:** The dataset's quality and variety are crucial variables in determining how well the algorithms function. Prospective studies should concentrate on growing the dataset by involving a broader range of tracking scenarios, item kinds, and ambient factors. To provide a more thorough evaluation framework, this involves taking infrared photographs under various lighting factors, at various distances, in occlusion settings, and against complicated backdrops.
3. **Modification of Evaluation Metrics:** Optimization of assessment metrics may facilitate a more thorough and precise evaluation of object tracking infrastructure. Currently, frequently used indicators such as accuracy, recall, and F1 score may not adequately represent the complexities and difficulties associated with dim tiny infrared pictures. Create innovative measures that take into account the special features of the infrared domain, such as thermal contrast, edge preservation, and noise resilience.
4. **Hardware optimization and execution in real time:** Prospective studies should examine how object tracking in dark tiny infrared pictures may be feasibly carried out, even though this thesis largely focuses on algorithmic performance. Optimize algorithms for rapid computation on specialist hardware platforms like GPUs or FPGAs to look into

real-time tracking capabilities. This will make it possible to use tracking systems in actual scenarios where speed and resource limitations are important aspects.

REFERENCES

[1] https://blog.roboflow.com/what-is-object-tracking-computer-vision/

[2] Mittal, A., Singh, S., Gupta, M., & Singh, S. P. (2021). EasyChair Preprint Comparing Various Tracking Algorithms in OpenCV COMPARING VARIOUS TRACKING ALGORITHMS IN OPENCV.

[3] Chakrabarti, S., Mukherjee, A., University of Engineering & Management, J., Institute of Electrical and Electronics Engineers. Delhi Section, & Institute of Electrical and Electronics Engineers. (n.d.). *IEMECON 2019 : the 9th Annual Information Technology, Electromechanical Engineering and Microelectronics : [13th-15th March, 2019]*.

[4] Janku, P., Koplik, K., Dulik, T., & Szabo, I. (n.d.). *Comparison of tracking algorithms implemented in OpenCV*. https://doi.org/10.1051/04

[5] Institute of Electrical and Electronics Engineers. Madras Section, & Institute of Electrical and Electronics Engineers. (n.d.). Proceedings of the 2020 IEEE International Conference on Communication and Signal Processing (ICCSP) : 28th - 30th July 2020, Melmaruvathur, India.

[6] Tiwari, M., & Singhai, R. (2017). A Review of Detection and Tracking of Objects from Image and Video Sequences. In International Journal of Computational Intelligence Research (Vol. 13, Issue 5). http://www.ripublication.com

[7] Häger, G., Bhat, G., Danelljan, M., Khan, F. S., Felsberg, M., Rudl, P., & Doherty, P. (2016). Combining visual tracking and person detection for long term tracking on a UAV. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture *Notes in Bioinformatics), 10072 LNCS, 557–568. https://doi.org/10.1007/978-3-319-50835-1_50*

[8] Leal-Taixé, L., Milan, A., Schindler, K., Cremers, D., Reid, I., & Roth, S. (2017). Tracking the Trackers: An Analysis of the State of the Art in Multiple Object Tracking. http://arxiv.org/abs/1704.02781

[9] https://www.educba.com/opencv-contrib/

[10] M.-A. Carbonneau, V. Cheplygina, E. Granger, and G. Gagnon,“Multiple instance learning: A survey of problem characteristics and applications,” Pattern Recognition, vol. 77, pp. 329–353, 2018

[11] https://broutonlab.com/blog/opencv-object-tracking

[12] V.lehtola, H.Huttunen, F.Christophe, and T.Mikkonen “Evaluation of Visual Tracking Algorithms for Embedded Devices,” Image Analysis Lecture Notes in Computer Science, pp. 88-97, 2017.

[13] https://pyimagesearch.com/2018/07/30/opencv-object-tracking/

[14] http://www.visual-tracking.net

[15] https://learnopencv.com/object-tracking-using-opencv-cpp-python/

[16] https://www.scidb.cn/en/detail?dataSetId=720626420933459968&dataSetType=journal

[17] Li, Y., Liang, S., Bai, B., & Feng, D. (2012). Detecting and tracking dim small targets in infrared image sequences under complex backgrounds. Multimedia Tools and Applications, 71(3), 1179–1199.

[18] Z. Kalal, K. Mikolajczyk and J. Matas, "Tracking-Learning-Detection," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 7, pp. 1409-1422, July 2012.

[19] J. Hu, S. Hu and Z. Sun, "A real time dual-camera surveillance system based on tracking-learning-detection algorithm," 2013 25th Chinese Control and Decision Conference (CCDC), Guiyang, 2013, pp. 886-891

[20] Jig et al., "A Tracking-Learning-Detection (TLD) method with local binary pattern improved," 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO), Zhuhai, 2015, pp. 1625-1630.

[21] Gundogdu, E., Ozkan, H., Demir, H. S., Ergezer, H., Akagündüz, E., & Pakin, S. K. (n.d.). *Comparison of Infrared and Visible Imagery for Object Tracking: Toward Trackers with Superior IR Performance*.