الجمهورية الجزائرية الديمقراطية الشعبية

People's Democratic Republic of Algeria

وزارة التعليم العالي والبحث العلمي

Ministry of Higher Education and Scientific Research

MOHAMED KHIDER UNIVERSITY – BISKRA
FACULTY OF SCIENCE AND TECHNOLOGY
DEPARTMENT OF ELECTRICAL ENGINEERING

جامعة محمد خيضر – بسكرة
كلية العلوم و التكنولوجيا
قسم الهندسة الكهربائية

Ref: ....

المرجع:.................

Thesis submitted to the department of electrical engineering in candidacy for the Degree of Doctor of Sciences in Electrical Engineering

Option: Electronics

Visual Object Tracking Approach Based on Wavelet Transforms

Presented by:

BOURENNANE Mohammed

Discussed publicly: 03/11/2022

In front the jury consists of:

President:	**Pr. Zine-Eddine BAARIR**	**Prof**	**University of Biskra**
Supervisor:	**Pr. Nadjiba TERKI**	**Prof**	**University of Biskra**
Examiner:	**Pr. Slami SAADI**	**Prof**	**University of Djelfa**
Examiner:	**Dr. Hilal NAIMI**	**MCA**	**University of Djelfa**

ACKNOWLEDGEMENTS

First of all, I thank ALLAH for helping me throughout my research and enabling me to finish my thesis.

I am deeply indebted to my supervisor **Pr. Nadjiba TERKI** for her support, stimulating suggestions and encouragement helped me in all the time of research and writing of this thesis.

I also want to say thank to all members of the jury. I am particularly grateful to **Pr. Slami SAADI**, and **Dr. Hilal NAIMI** for accepting to report my thesis and for their advice, comments and questions, and to **Pr. Zine-Eddine BAARIR** to participate as president of my Thesis Committee.

In general, I would like to thank all people who encouraged and supported me from both technical and human points of view during these years of intense work.

Dedication

I dedicate this modest work to:

All my family.

Abstract:

In this Thesis, a new visual object tracking (VOT) approach is proposed to overcome the main challenging problem encountered within the existing approaches known as the significant appearance changes which is due mainly to the heavy occlusion and illumination variations. Indeed, the proposed approach is based on combining the deep convolutional neural networks (CNN), the histograms of oriented gradients (HOG) features, and the discrete wavelet packet transform to ensure the implementation of three ideas. Firstly, solving the problem of illumination variation by incorporating the coefficients of the image discrete wavelet packet transform instead of the image template to handle the case of images with high saturation in the input of the used CNN, whereas the inverse discrete wavelet packet transform is used at the output for extracting the CNN features. Secondly, by combining four learned correlation filters with convolutional features, the target location is deduced using multichannel correlation maps at the CNNs output. On the other side, the maximum value of the resulting maps from correlation filters with convolutional features produced by HOG feature of the image template previously obtained are calculated and which are used as an updating parameter of the correlation filters extracted from CNN and from HOG where the major aim is to ensure long-term memory of target appearance so that the target item may be recovered if tracking fails. Thirdly, to increase the performance of HOG, the coefficients of the discrete packet wavelet transform are employed instead of the image template. Finally, for the validation and the evaluation of the proposed tracking approach performance based on specific performance metrics in comparison to the state-of-the-art counterparts, extensive simulation experiments on benchmark datasets have been conducted out, such as OTB50, OTB100 , TC128 ,and UAV20. The obtained results clearly prove the validity of the proposed approach in solving the encountered problems of visual object tracking in almost the experiment cases presented in this thesis compared to other existing tracking approaches.

Keywords: Visual tracking, deep convolution neural networks, wavelet transform, HOG features.

Résumé :

Dans cette thèse, une nouvelle approche de suivi visuel d'objets (VOT) est proposée pour surmonter le principal problème rencontré dans les approches existantes, connu sous le nom de changements d'apparence significatifs, principalement dus aux fortes variations d'occlusion et d'éclairage. En effet, l'approche proposée est basée sur la combinaison des réseaux de neurones convolutifs profonds (CNN), des caractéristiques des histogrammes de gradients orientés (HOG) et de la transformée discrète en paquets d'ondelettes pour assurer la mise en œuvre de trois idées. Tout d'abord, résoudre le problème de variation d'éclairage en incorporant les coefficients de la transformée en paquets d'ondelettes discrètes de l'image au lieu du modèle d'image pour gérer le cas d'images à forte saturation à l'entrée du CNN utilisé, alors que la transformée en paquets d'ondelettes discrète inverse est utilisée en sortie pour extraire les caractéristiques CNN. Deuxièmement, en combinant quatre filtres de corrélation appris avec des caractéristiques convolutives, l'emplacement cible est déduit à l'aide de cartes de corrélation multicanaux à la sortie des CNN. D'autre part, la valeur maximale des cartes résultantes des filtres de corrélation avec les caractéristiques convolutives produites par la caractéristique HOG du modèle d'image précédemment obtenu est calculée et qui est utilisée comme paramètre de mise à jour des filtres de corrélation extraits de CNN et de HOG où le L'objectif principal est d'assurer une mémoire à long terme de l'apparence de la cible afin que l'élément cible puisse être récupéré en cas d'échec du suivi. Troisièmement, pour augmenter les performances de HOG, les coefficients de la transformée discrète en ondelettes de paquets sont utilisés à la place du modèle d'image. Enfin, pour la validation et l'évaluation des performances de l'approche de suivi proposée sur la base de mesures de performance spécifiques par rapport aux homologues de pointe, des expériences de simulation approfondies sur des ensembles de données de référence ont été menées, telles que OTB50, OTB100 , TC128 , et UAV20. Les résultats obtenus prouvent clairement la validité de l'approche proposée pour résoudre les problèmes rencontrés de suivi visuel d'objets dans presque les cas expérimentaux présentés dans cet article par rapport aux autres approches de suivi existantes.

Mots clés: Suivi visuel, réseaux de neurones à convolution profonde, transformation en ondelettes, HOG.

الملخص:

في هذه الاطروحة ، تم اقتراح نهج جديد لتتبع الكائن المرئي (VOT) للتغلب على المشكلة الصعبة الرئيسية التي تمت مواجهتها في الأساليب الحالية المعروفة باسم التغييرات الكبيرة في المظهر والتي ترجع أساسًا إلى الاختلافات الشديدة في الانسداد والإضاءة. في الواقع ، يعتمد النهج المقترح على الجمع بين الشبكات العصبية التلافيفية العميقة (CNN) ، والرسوم البيانية لميزات التدرجات الموجهة (HOG) ، وتحويل الحزمة المويجة المنفصلة لضمان تنفيذ ثلاث أفكار. أولاً ، حل مشكلة اختلاف الإضاءة من خلال دمج معاملات تحويل حزمة المويجات المنفصلة للصورة بدلاً من قالب الصورة للتعامل مع حالة الصور ذات التشبع العالي في مدخلات CNN المستخدمة ، في حين يتم استخدام تحويل حزمة الموجة المنفصلة العكسية في الإخراج لاستخراج ميزات CNN. ثانيًا ، من خلال الجمع بين أربعة مرشحات ارتباط مكتسبة مع ميزات تلافيفية ، يتم استنتاج الموقع المستهدف باستخدام خرائط الارتباط متعدد القنوات في إخراج CNN. على الجانب الآخر ، يتم حساب القيمة القصوى للخرائط الناتجة من مرشحات الارتباط مع الميزات التلافيفية التي تنتجها ميزة HOG لقالب الصورة التي تم الحصول عليها مسبقًا والتي يتم استخدامها كمعامل تحديث لمرشحات الارتباط المستخرجة من CNN ومن HOG حيث الهدف الرئيسي هو ضمان وجود ذاكرة طويلة المدى لمظهر الهدف بحيث يمكن استعادة العنصر الهدف في حالة فشل التتبع. ثالثًا ، لزيادة أداء HOG ، يتم استخدام معاملات التحويل المويج للحزمة المنفصلة بدلاً من قالب الصورة. أخيرًا ، من أجل التحقق من صحة وتقييم أداء نهج التتبع المقترح استنادًا إلى مقاييس أداء محددة مقارنة بأحدث النظراء ، تم إجراء تجارب محاكاة واسعة النطاق على مجموعات البيانات المعيارية ، مثل OTB50 و OTB100 و TC128 و UAV20. تثبت النتائج التي تم الحصول عليها بوضوح صحة النهج المقترح في حل المشاكل المصادفة لتتبع الكائن المرئي في حالات التجربة المعروضة في هذه الورقة تقريبًا مقارنة بطرق التتبع الأخرى الحالية.

الكلمات المفتاحية: التتبع البصري ، الشبكات العصبية العميقة ، التحويل المويجي ، ميزات HOG.

TABLE OF CONTENTS

TABLE OF CONTENTS

List of Figures

List of Tables

List of Acronyms

2D-DWT	Two-Dimensional Discrete Wavelet Transform
adaDDCF	Adaptive Discriminative Deep Correlation Filter
ADVISOR	Annotated Digital Video for Surveillance and Optimized Retrieval
AI	Artificial Intelligence
ASEF	Average of Synthetic Exact Filters
ASLA	adaptive structural local sparse appearance model
AUC	Area Under Curve
BACF	Background-Aware Correlation Filter
BACF	Background-Aware Correlation Filters
CDVP	Cooperative Distributed Vision Project
CF	Correlation Filter
CLIP	Complementary Learners with Instance-specific Proposals
CNN	Convolutional Neural Networks
CSK	Circulant Structured Kernels
CT	Computed Tomography
CV	Computer Vision
CVPR	Computer Vision and Pattern Recognition
CWT	Continuous Wavelet Transform
DAMA	Distractor-Aware Learning and Multi-Anchor Detection
DARPA	Defense Advanced Research Projects Agency
DCF	Deep Collaborative Filtering
DCNNs	Deep Convolutional Neural Networks
DeepNCC	Deep Normalized Cross-Correlation
DFC	Deep Features and Correlation Filters
DFT	Distribution Fields for Tracking
DOG	Difference Of Two Gaussian
DP	Distance Precision
DRVT	Dual-regression model for visual tracking
DSST	Discriminative Scale Space Tracking
DWT	Discrete Wavelet Transform

FSNet	Feature selection accelerated convolutional neural networks
FWT	Fast Wavelet Transform
HCF	Hierarchical Convolutional Filters
HCFT	Hierarchical convolutional features for visual tracking
HCFTs	Robust Visual Tracking via Hierarchical Convolutional Features
HDT	Hedged Deep Tracker
HID	Human Identification at a Distance
HOG	Histograms of Oriented Gradients
HSI	Hue Saturation Intensity
HSV	Hue Saturation Value
ICCV	International Conference on Computer Vision
IDWT	Inverse Discrete Wavelet Transform
IJCAI	International Joint Conference on Artificial Intelligence
IVT	Incremental Learning for Robust Visual Tracking
KCF	Kernelized Correlation Filter
LCT	Long-term Correlation Tracker
MEEM	Multiple Experts Using Entropy Minimization
MemDTC	MemTrack with Distractor Template Canceling
MemTrack	Memory Networks for Object Tracking
MIL	Multiple Instance Learning
MOSSE	Minimum Output Sum of Squared Error
MST	Motion Saliency Guidance Tracking
MUSTER	MUlti-Store Tracker
NCC	Normalized Cross-Correlation
OPE	One-Pass Evaluation
OTB	Object Tracking Benchmark
RGB	Red, Green, and Blue
RSCF	Region Sparse Constraint Correlation Filter
SAMF	Scale Adaptive with Multiple Features tracker
siamfc3s	Fully-Convolutional Siamese Networks for Object Tracking
SRDCF	Spatially Regularized Discriminative Correlation Filters
STAPLE	Sum of Template And Pixel-wise Learners
Struck	Structured Output Tracking with Kernels Sam

SVM	Support Vector Machine
TC	Temple Color
TIP	Transactions on Image Processing
TLD	Track-Learn-Detect
TMCF	Target-Masked Correlation Filter
TMM	Transactions on Multimedia
TPAMI	Transactions on Pattern Analysis and Machine Intelligence
TripFC	Triplet network use a Fully-Convolutional
TSP	Transactions on Signal Processing
UAV	Unmanned Aerial Vehicles
VOT	Visual Object Tracking
VSAM	Surveillance And Monitoring
VStar	Visual Surveillance Star
YCBCr	Luminance, Chrominance

Chapter 1 :

INTRODUCTION

1.1 Context

Nowadays, the visual object tracking (VOT) is becoming a very interesting area of research which is attracting much attention due to its importance within numerous applications such as unmanned control systems, motion analysis, and video processing [1–4]. Indeed, the VOT is basically used to estimate an unknown visual target trajectory based on a known initial starting state of the considered target. However, the visual tracking remains a difficult problem to be solved accurately despite the substantial efforts made by many researchers in this area during the last decade, due to the new challenges that have been induced by new technology evolution and which make the target objects often experience important changes in their appearance Such as the scale variation, fast motion, in-plane rotation, deformation, motions blur, occlusion, illumination variation, out-of-plane rotation, background clutter, and out-of-view.

Recently, the convolutional neural networks (CNN) features have been put to use in a variety of computer vision applications, for example, object identification, image segmentation, and image classification[5].Whereas, the effective use of the rich hierarchical features of CNNs in visual tracking, has brought significant improvement . It has been proved that the convolutional layers have the ability of ensuring the presentation of the invariant features against the variation of the target appearance which can be very useful in visual object tracking applications. Unfortunately, it has been found that the CNNs has a major limitation resulting from the fact it was built based on principles of using other visual classification tasks[6].

The authors of [7] ,have proposed the exploitation of the rich hierarchical features of the deep convolutional neural networks (DCNNs) to ensure enhanced accuracy and robustness of the visual object tracking. Indeed, it has been proposed that to ensure invariant feature representation with respect to significant variations in the target appearance, the outputs of the last convolutional layers have been used to encode the semantic information of the target. This proposed technique has faced the problem of losing the precise localization of the target due large spatial resolution, even it has been resistant to the target significant appearance changes. In the same time, it has been observed in this technique that the features of the earlier convolutional layers can ensure precise localization of targets but in the same time they are less sensitive the target appearance changes.

Based on the concept that various layers in a CNN model give varying levels of information in describing an object[1, 2], some authors have attempted to address this issue by combining feature representations provided by different CNN layers with correlation filters to realize an efficient tracking performance [1–10].Despite the achieved advantages by the proposal of these techniques compared to those based on CNN, their proposal has presented some limitations where it has been based essentially on the learning and updating of the correlation filters in the frequency to overcome the problem of appearance variations, this has led to unwanted boundary effects and important degradation of the tracking model quality. Furthermore, it has been found that it cannot be effective for long time tracking and it cannot ensure the detection of the target position failures. In the same time it has faced a major problem against changes of illumination causing their extract limitation within specific color sequences [8].

1.2 Contributions

The main contributions presented in this thesis can be summarized in the following points:

- The wavelet decomposition based on different frequency sub-bands such as low-low (LL),low-high (LH),high-low (HL), and high-high (HH), have been used instead of RGB “Red Green Blue” image to resolve the problem of illumination variation in such cases when the saturation exceeds $\frac{2}{3} \times 100\%$.
- Based on the importance of combining feature representations from different CNN layers, such as in[2, 3, 9], a model of Hierarchical Convolutional Filters (HCF) has been proposed. The proposed model is composed of different convolutional layers (conv1-4,conv3-4, conv4-4, and conv5-4).
- A wavelet decomposition has been used to extracted histograms of oriented gradients features, which is made up of four layers [LL,LH,HL, and HH] instead of using the original image. The wavelet decomposition has been introduced to improve the performance of HOG features.
- In addition, an update control approach has been designed to allow the appearance changes identification while preventing model drift. This has been carried out by calculating the maximum value of the resulting maps from correlation filters with convolutional features products of HOG feature for image template that has been previously obtained, and which has been used as a parameter to the updating of the correlation filters.

For the evaluation of the proposed approach a large-scale benchmark dataset OTB50 with 50 challenging image sequences, OTB100 with 100 challenging image sequences have been used ,TC-128 and UAV20 with 128 and 20 sequences respectively .

1.3 Thesis Organization

This thesis work constitutes of seven chapters.

Chapter 2: In this chapter, we present the state-of-the-art of visual tracking, and we also give all performance terms used to judge such a visual tracking method.

Chapter 3: We give a detailed of image colors spaces, and imports methods of Image Features extraction, Convolution neural networks and Histograms of Oriented Gradients.

Chapter 4: This chapter provides the basic concept of wavelet transform types of wavelets, applications of wavelets, and the Discrete Wavelet Transforms.

Chapter 5: Presents in detail the main steps of our methods. Starting with Saturation condition, feature extraction then modeling correlation filters.

Chapter 6: We first introduce Benchmark Datasets, and the existing challenges in visual object tracking, including scale variation, background cluster, low resolution, etc. The performance evaluation of visual object tracking. Finally, the results and discussions in each database are given.

Chapter 7: The final chapter presents a conclusion on our work and a vision for future work.

Chapter 2 :

State-Of-The-Art

2.1 Introduction

In the field of computer vision, visual tracking is a rapidly developing area that has been getting more and more attention. One explanation is that the scientific problem of visual tracking presents many difficulties. Additionally, it is a component of numerous high-level computer vision issues, including activity comprehension, motion analysis, and event detection[11].

The most works related to the main proposal of this thesis, which were published during the last years, are described in this section. Indeed, the visual object tracking techniques are based mainly on to approaches such as the correlation filter (CF) and the convolutional neural networks (CNNs). It is worthy to note here that the proposed and developed technique in this thesis is based on these two approaches, which can be considered as the background for the presented proposal. Therefore, their descriptions are of great important for understanding the main features of the proposed visual tracker in this thesis compared to the existing approaches.

2.2 Motivation and Challenge

Humans have always yearned to figure out how to give computers the ability to see and analyze video data. Artificial intelligence (AI) has been gradually incorporated into several industries in recent years, along with advancements in machine learning and deep learning research [12], including automatic driving [13], speech recognition [14], face recognition [15], and virtual reality games [16].

One of the most important parts of Computer Vision is visual object tracking, which predicts the state of an object in a video based on its trajectory. This helps with behavior analysis. It has been used extensively in intelligent monitoring [17], human-computer interaction [18], automatic driving [19, 20], virtual reality [21, 22], crime projections[23], surgical navigation [24], aerospace[25, 26], and so on.

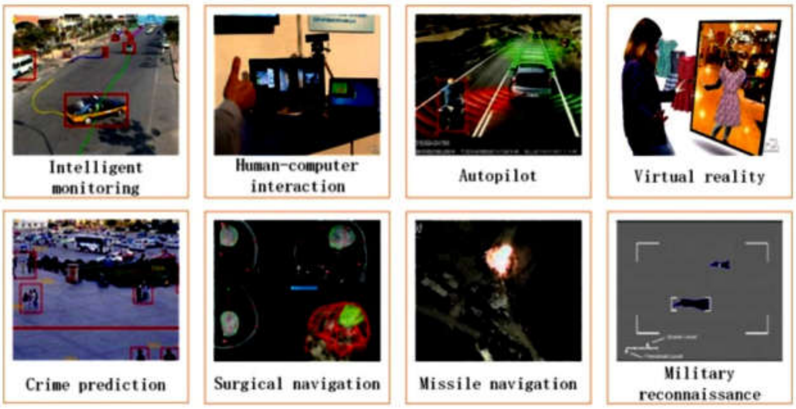


Figure 2. 1 : Some applications of visual object tracking

Figure 2.1 depicts some common visual object tracking application scenarios. For example, visual object tracking in smart traffic can determine whether there is a violation by monitoring the tracking of vehicles, such as illegal U-turns, speeding, and so on. In human-computer interaction, computers can track and calculate the states of human body parts such as hands, legs, head, eyes, and so on to determine the person's instructions and perform corresponding actions without the user pressing any buttons.

Visual object tracking in automatic driving can perceive the change and motion of objects around the vehicle to provide a specific reference for the vehicle computer. Visual object tracking combined with an object segmentation algorithm in virtual reality can calculate the location and shape of objects.

In the application of virtual changing, for instance, the shape of clothing can be automatically adjusted to fit the contours of the human body. Likewise, in crime prediction, monitoring and tracking the sudden aggregation and dispersion of people or other objects in the video could predict abnormal and possible emergencies, aid the police in locating illegal activities and enhance the social environment.

By tracking the position and posture of the scalpel and probe, the success rate of surgery can be increased using surgery navigation. Visual object tracking has important military applications as well.

In missile navigation and military reconnaissance, where the objects are frequently in motion and the cameras on the missiles are also jittery, visual object tracking can be used to determine the object's position and adjust the missile's attitude to improve the guidance accuracy.

Multiple nations and institutions have established major visual object tracking projects.

Carnegie Mellon University and the David Sarno Research Center co-founded Project Video Surveillance And Monitoring (VSAM) in the early 1997s with funding from the Defense Advanced Research Projects Agency (DARPA) [27]. This project intends to continuously track people and vehicles in complex environments using multiple video sensors and develops visual surveillance systems. In addition, DARPA funded the University of Maryland-led Human Identification at a Distance (HID) project in 2000 [28]. In 1999, Framework 5 on EU information technologies also established the Annotated Digital Video for Surveillance and Optimized Retrieval (ADVISOR) project, which was used to manage urban traffic systems and analyze pedestrian behaviour. During the same period, Hiroshima University in Japan hosted the Cooperative Distributed Vision Project (CDVP) intelligent monitoring project from 1996 to 1999 to develop community-oriented monitoring systems. The Chinese Academy of Science and Technology's Institute of Automation also oversaw the Visual Surveillance Star (VStar) project for urban traffic monitoring and management.

In addition to projects that use visual object tracking, many of the world's best magazines and conferences continue to support developing and improving visual object tracking algorithms. Visual object tracking is the main area of research in video processing, and every year, it takes up a certain amount of space in the top Computer Vision (CV) journals and conferences. The best conferences for visual object tracking direction are mostly the IEEE International Conference on Computer Vision (ICCV), the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), the European Conference on Computer Vision (ECCV), and the International Joint Conference on Artificial Intelligence (IJCAI), among others. In contrast, the best journals are mostly IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), IEEE Transactions on Signal Processing (TSP), IEEE Transactions on Image Processing (TIP), and IEEE Transactions on Multimedia (TMM), among others. Visual object tracking research has a lot of theoretical value and a wide range of practical applications [11]. People's demand for visual object tracking is growing in tandem with the advancement of science and technology. The objects to be tracked transition from rigid to non-grid. The video background ranges from simple to complex, including occlusion, motion blur, and other complex situations.

Furthermore, the demand for time tracking is increasing. The essence of visual object tracking is a problem of online learning with small non-annotated sample sizes: The main challenge in visual object tracking is constructing a robust representation model, a fast motion model, and an effective update model. Furthermore, tracking tasks are becoming more and more realistic. In one sequence, there are multiple tracking challenges. Thus, one of the key problems in visual object tracking research is designing a universal tracing algorithm that can deal with multiple challenges in a single sequence. Wu et al. [29] classified the difficulties in visual object tracking into 11 categories. The 11 challenges are described in detail in Chapter 5.

2.3 Basic Concepts and Features

Visual object tracking is a well-known research topic in computer vision. Man et al. [30] built the computer vision framework in 1982 and demonstrated that the Fourier transform of spatial frequency sensitive data can be used to derive the retinal receptive field geometry. We can detect edges and contours in image intensity gradients using the Laplace or second derivative method. The Difference Of Two Gaussian (DOG) functions can be used to calculate the excitatory and inhibitory receptive fields.

To optimize the bandwidth of the optical distribution, visual systems can use two-dimensional convolutions and Gaussian filters as operators. It lays the theoretical groundwork for computer vision and visual object tracking. Arnold W. M. Smeulders et al. [31] published a detailed definition of visual object tracking in TPAMI, one of the top journals in computer vision fields, in 2014. "Tracking is the analysis of video sequences to determine the target's location over a sequence of frames (time), beginning with the bounding box given in the first frame."

Arnold W. M. Smeulders et al. [31] not only defined visual object tracking but also summarized the principles and processes of some existing visual object tracking algorithms. Visual object tracking is divided into five parts: object area, appearance model, motion model, tracking algorithm, and update model. The object region that needs to be tracked should be selected at the start of tracking. The appearance model is then built based on the object image, and the motion model is built based on the relationship of the object states between two adjacent frames. Tracking algorithms, based on different principles such as similarity matching and optimization, calculate the state of the object in the current frame using appearance and motion models. Finally, the updated model is used to update the object's appearance model and motion model to adapt to changes in shape and appearance caused by the deformation and occlusion of the object.

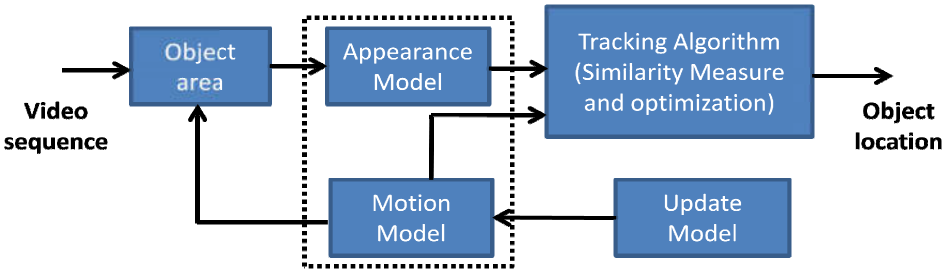


Figure 2. 2: Basic ideas of visual object tracking

Iteration could predict the object's position and state in each frame. The object area in Figure 2.2 refers to the image of the object that needs to be tracked. A bounding box is typically used to define the object area. NCC tracker, for example [32]. Some object areas, such as MST tracker [33], are built in an elliptical shape. The appearance model primarily extracts the feature representation of the object image. The appearance model can be divided into two dimensions image array [32–34], one dimension histogram [35–37], and a feature vector [38–40] based on the features extracted from objects. The motion model is used to simulate the object's motion pattern. Typically, it is assumed that the object in the video moves smoothly. That is, the object's centre in the following frame is located near the object's centre in the current frame. As a result, the motion model relies heavily on the Gaussian distribution [41, 42]. The detection methods and tracking algorithms are combined in some trackers, such as the TLD tracker [43], to construct the object's motion model.

Tracking algorithms use appearance and motion models to predict an object's position and state. Tracking algorithms are classified into matching-based [44] and classification-based [45], also known as generative and discriminative tracking algorithms. In addition, deep learning-based tracking algorithms have advanced rapidly in tandem with the advancement of deep learning and artificial intelligence [46, 47]. There are two methods for updating the appearance model. One method is to use the object template calculated in the current frame to partially update the entire object appearance model [48, 49], such as the weighted sum. The other option is to completely reconstruct the appearance model based on the object template computed in the current frame [50].

2.4 Evolution of Visual Object Tracking Technology

The evolution of the visual object tracking algorithm demonstrates a trend from conventional tracking methods [42, 51, 52] to deep learning-based tracking methods [53–55], as well as from generative methods [41, 56–58] to discriminative methods [33, 34], and [58]. Figure. 2.2 provides a concise overview of the development of visual object tracking technology.

From 2005 to 2010, generative methods such as the Bayesian framework, particle filter, and Kalman filter were used to track visual objects. Visual object tracking was primarily thought of as a template matching problem during this time. Some manually designed features are used to build the comparison appearance model. As a motion model, the Gaussian distribution is used to provide candidate objects, and the final state of the object can be calculated by finding the candidate with the highest similarity. Based on the Kalman filter and particle filter, visual object trackers were proposed by Abdel-Hadi et al. [56] and Han et al. [57]. Yang et al. [58] extracted the superpixel feature to build a comparison appearance model.

Researchers studied correlation filtering-based trackers with kernel methods, which belong to the discriminative methods, extensively between 2010 and 2014 [45, 59–62]. The goal of correlation filtering-based trackers is to train a correlation filter that will allow the object centre to be located at the response map's peak value after the correlation filtering operation. First, Bolme et al. [59] proposed the ASEF filter after using the correlation filter to determine the location of the eye. The ASEF filter was then improved further by Bolme et al. [60], who applied the correlation filter to visual object tracking and proposed the MOSSE tracker, which is also the first correlation filter-based tracker. Next, Henriques et al. [61] proposed the CSK tracker, which uses a linear classifier to solve the correlation filter. Tracking was viewed as a ridge regression problem in 2014 by Henriques et al. [45]. They used a circulant matrix to collect positive and negative samples around the object for training the correlation filter.

Danelljan et al. [62] used two correlation filters to address scale variation issues in the KCF tracker: translation filter and scale filter. The translation filter is used to detect the object's central location, while the scale filter is used to estimate the object's scale change.

The deep feature was merged into correlation filtering-based trackers in 2015 and 2017 due to its powerful representation [2, 63, 64]. As feature extractors, well-trained neural networks are used. To build the proposed HCF tracker, Ma et al. [2] used a pre-trained deep network to extract the deep feature of the object and combined multi-features from feature maps of different layers in the deep network. In addition, [2] discusses the characteristics of the feature map from different layers in the deep network. Hong et al. [63] proposed a CNN-based learnable saliency map and combined it with an SVM-based classifier to create the appearance model. To integrate multiple resolution feature maps and achieve accurate sub-pixel location, Danelljan et al. [64] proposed the continuous convolution operators.

Deep learning-based tracking methods, particularly the Siamese network, have seen rapid development from 2018 to 2020, alongside the development of deep networks [47, 65, 66]. Bertinetto et al. [65] proposed the Siamese FC tracker by combining the Siamese network and a correlation filter. Li et al. [47] integrated the region proposal network into the Siamese network to provide candidates for the object. The region proposal network in [47] can be considered a visual object tracking motion model. Wang et al. [66] proposed a unified solution combining visual object tracking and instance segmentation problems. Instance segmentation may improve tracking accuracy, whereas visual object tracking may increase instance segmentation speed.

Recently, some online update strategies and metal learning have been used in Siamese network-based trackers to improve the robustness of tracking performance [67–69], aiming at the online update and few-shot learning problems in deep learning-based trackers. Zhang et al. [67] treated the updated model as a function of ground truth from the first frame, the template from the previous frame, and the current frame's appearance model. Furthermore, the function was expressed as a deep network, and the UpdateNet was proposed for model updating. To improve the robustness of tracking performance, Huang et al. [68] and Wang et al. [69] introduced meta-learning into the siamese network-based tracking method as the initialization of the siamese network.

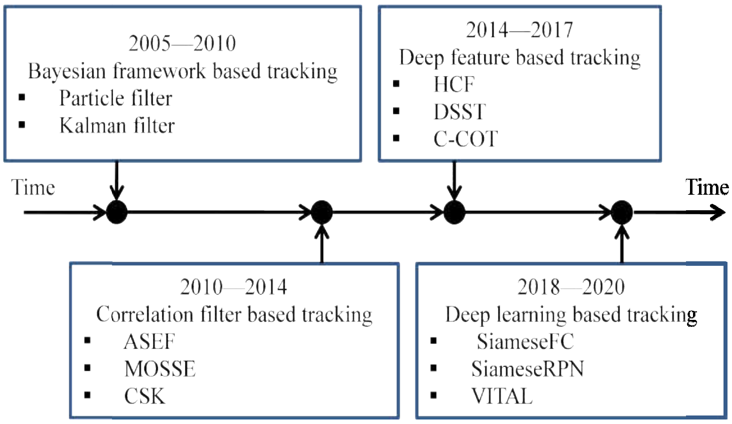


Figure 2. 3 : The evolution of visual object tracking

2.5 Correlation filter

Within the last years, the correlation filter (CF) has been considered to be one of the most widely used algorithms for optical tracking due to its high computational efficiency and robustness[70].This filter has been used later by Bolme et al, where they have proposed a new approach known as the minimum output sum of squared error (MOSSE) to ensure better performance of optical object tracking[60].The key innovation brought by the MOSSE is its ability to provide ongoing, adaptive training for the target object's appearance changes. As a result , the performance of the proposed MOSSE CF-based tracking techniques have received significant attention from the researchers working in the area of visual object tracking, and hence several updated variants of this tracker have tracker have been suggested[70]. The authors of [45] have proposed the KCF Kernelized Correlation Filter which is similar to the circulant structured kernels (CSK) technique[61],where in the proposed KCF the correlation filter has been used in kernel space. The authors of [70]have proposed a modified approach that has yielded to improved results compared to the conventional KCF in terms of tracking precision in the case of very important target movement within a real-time streaming condition. This approach has been presented as an improved kernelized correlation filter (KCF)-based tracking method in which three primordial functional modules have been incorporated such as the tracking failure detection, the re-tracking based on multiple search windows, and the motion vector analysis which is used to confirm the preferred search window to be adapted. Li et al. have integrated the HOG and color-naming to further boost the overall tracking performance taking into account the tracking accuracy and the robustness of the proposed approach [71]. Danelljan et al. have proposed discriminative scale space tracking (DSST) and have developed a robust scale estimation of tracking where the issue of target size shift has been successfully addressed by searching for the target at various scales using HOG features[62]. Furthermore, Danelljan et al, have presented in[72], the spatially regularized discriminative correlation filters (SRDCF) by adding a weight factor for space regularization during the training step, hence the boundary impact has been substantially reduced. Galoogahi et al have proposed the background-aware correlation filter (BACF) [73],where they have utilized a genuine background patch as well as the target patch in order to learn the tracker where they have used an online adaption approach for updating target and backgrounds over any time. Qi et al. have proposed the hedged deep tracker (HDT) to build a more powerful tracker based on merging several weak trackers[7]. Indeed, in this approach the weak trackers have been used to calculate the initial target positions, and the ultimate choice is made by the hedging algorithm which incorporates the trackers. Ma et al. have proposed long-term correlation tracker (LCT) which presents a resistive tracker to target translation and has the capability of the re-detection in the case of tracking failure[74]. Bertinetto et al. have presented the sum of template and pixel-wise learners method (STAPLE) where the target representation was achieved using color histograms and HOG approach [75]. All of the techniques discussed above have been proposed to enhance the performance of CNN-based trackers.

2.6 Convolution Neural Networks

Recently, several researches have been carried out on the convolution neural networks (CNNs), which have proved its application success, especially in computer vision tasks, such as object recognition[76],and classification[77].Thus, many researchers have paid more attention in their research to the strength of CNN and its practical application in visual object tracking where their obtained results have proved that the use of CNN has improved the performance of the visual object tracking. Furthermore, many proposed techniques have combined the deep collaborative filtering (DCF) framework with CNN to benefit in the same time from their both features[2, 7, 78]. In [2] the authors have used the features obtained from a deep convolutional neural networks, which have been extracted from the datasets of the object recognition training, where the main aim is the ensure the accuracy and the robustness of the visual object tracking. Whereas, Danelljan et al have used the shallow CNN features to identify the target of a moving object [79]. Song et al, have proposed a convolutional neural network that combines feature extraction, map creation and model updating, which significantly has improved the tracking resilience[80]. In[3], the authors added the scale estimation when the target tracking fails benefiting from the HOG functionality to ensure the reposition. As can be observed from the aforementioned proposed approaches, the HOG feature and CNN feature can be used to complement one another's inadequacies by utilizing their individual advantages, which can improve the expressive capacity of the fused feature where the weights can be adaptively distributed to enhance the tracking accuracy.

On the other side, due to the known capacity of wavelets to divide picture data into several scales and to create sub-bands, wavelet-based methods have been shown to be more effective where the fixed layer of convolution using wavelets was used in neural networks[81]. Bruna and Mallat have used a combination of wavelet transform convolution and a non-linear model to determine the invariant image characteristics that preserve high-frequency information for classification[82]. in [83]the wavelet transformation has been implemented in CNNs through a residual network which has performed well for super-resolution of single images. To retain detailed image edges, Deeba et al, have suggested the combination of the wavelet with multilayer convolution frameworks for computed tomography (CT) image restoration[84]. Whereas, in order to recover edge characteristics, the authors in [85]have developed an efficient correlated wavelets method.

Besides the reported efforts, and as a result of the sampling ambiguity, considerable efforts have been made in order to alleviate problems caused by model updating. In order to minimize tracking drifts, several methods that focus on how to update a discriminative classifier in a suitable manner have been proposed [3], such as multiple instance learning (MIL)[86], ensemble learning[87], among others. Kalal et al. have decomposed the tracking job into tracking, learning, and detection (TLD) modules rather than learning a single classifier, where in this approach the tracking and detection modules can be used to facilitate one another tasks, i.e., the tracker results give more training samples for the detector to be updated accurately [88].On the other side, when tracking fails, the tracker's online-learned detector may be used to re-initialize it. Other Similar methods have been used to retrieve target objects after tracking failures [89–91].

Hare et al. have demonstrated that the goal of label prediction through a classifier is not directly linked to the goal of tracking (precise location estimate), therefore they have reformulated the problem as a combined structured output prediction job[34]. Zhang et al. have integrated several classifiers with varying learning rates to ensured better performance tracking[92]. Other researchers have addressed the issue of sampling ambiguity more effectively based on proposed approaches, which have outperformed the benchmark research[93].

2.7 Wavelets

A “wave” is commonly defined as an oscillation function of time or space like sinusoid (complex exponential). It expands signals or functions on the basis of sinusoids which has proven to be extremely important in Mathematics, Science and Engineering specifically for periodic time invariant [94, 95] . The first and most basic wavelet transform was described by Haar [96]. Wavelet transform is able to furnish time and frequency information simultaneously. The word “wavelet” has been introduced by Morlet et al.,[97] . They used the French word “ondelette” , which means “small wave” originated from the study of time-frequency signal analysis, wave propagation and sampling theory [98]. Morlet first introduce the idea of wavelets as a family of functions constructed by using translation and dilation of single function, called mother wavelets, for analysis of nonstationary signals [99].

Wavelets are a mathematical tool and they can be used to extract information from many different kinds of data, including images, audio and video signals. The subject of wavelet analysis has recently drawn a great deal of attention from Mathematical Scientists in various disciplines. It is creating a common link between Mathematicians, Physicists, and Electrical Engineers with modern applications as diverse as wave propagation, data compression, image processing, image retrieval, image segmentation, edge detection, pattern recognition, computer graphics and other medical image technology [95, 99] . The wavelet transform decompose the signal with finite energy in the spatial domain into a set of function as a standard in the modular spatial domain of orthogonal. Then, we analyze the characteristics of the signal in the modular spatial domain. Compared with the traditional Fourier analysis, the wavelet transform can analyze the function in the modular spatial domain and timing domain which has better local capacity of the frequency and time. It is the development and sublimation of Fourier transform, which has a lot of advantages. Wavelets transform based method become visible more often in the early 1980s [100, 101]. One of the principal reasons for the evolution of wavelets and wavelet transforms is that the Fourier transform does not contain the confined information of signals.

With the rapid growth and advancement of technologies interconnected to communication and multimedia, information technology has penetrated its new span and generating innumerable digital image processing techniques. The implementation of multimedia needs some belongings like digital image processing and digital signal processing. An image can be defined as a matrix of pixel or intensity values and which can be analyzed using a digital computer [102]. Image processing has become an indispensible part of concurrent scientific and technological activity and it often used as a collection of techniques. Image processing generally applied in the transmission and analysis of two dimensional signals like traditional image processing images, satellite images, medical images and interpretation and analysis of time-frequency signals. The major objectives of image processing techniques are converting the images to discrete form, compressing the images to save storage space or channel capacity, enhancement of the degraded images for better representation, restoration of original images, reconstruction of original images from the set of projections, comparing and registering digital images to one another based on their properties, segmenting the images into different regions based on the requirements and comparing the performance analysis of the different models which define the classes of images. The latest growth of data throughout multimedia based internet applications has strengthen the necessity of much convenient ways to encode signals and images and compress such signals to store and communicate [103].

Segmentation or partition of an image fragments into its various meaningful regions. Image segmentation transforms an image in-between the objects to analyze further and the remaining objects, which are not required to analyze [102]. It improves the accuracy to analyze various image processing algorithms.

2.8 Conclusion

In this chapter, we have presented state-of-the-art methods in online visual tracking, including the motivations, practical algorithms, experimental evaluations, Correlation filter, Convolution Neural Networks, and Wavelets.

Chapter 3 :

Image analysis

3.1 Introduction

Color space is a mathematical approach for representing color information as three or four separate color components [104]. Color space shows how colors are represented and precisely specifies the components of color space to discover how each color spectrum appears.

Different color spaces are used for applications such as computer graphics, image processing, TV broadcasting, and computer vision. There are several color spaces accessible, including RGB-based color space (RGB, normalized RGB), Hue-based color space (HSI, HSV, and HSL), and Luminance-based color space (YCBCr, YIQ, and YUV). RGB and HSV are employed in this thesis.

On the other side, for many tasks in computer vision special kinds of features are used. These are elements which are extracted from the image. A feature is a vector calculated at a point in the input image using the local region. A feature map consists of these features sampled in a grid. Therefore feature extraction can be seen as a map from the input-image-space to $\square^{a,b,c}$ where a and b are the number of pixels in each row and column respectively; and c is the number of feature dimensions for that feature. Hence each image is transformed into another image, possibly with another amount of channels, and possibly with another resolution. There are various popular choices of features. Some examples are color names,Histogram of Oriented Gradients (HOG) , and features extracted from deep convolutional neural networks (CNN). In this thesis, HOG and features extracted from a CNN are used.

A wavelet transform is an excellent tool for image and signal processing. Many wavelet denoising algorithms in the literature demonstrate that the wavelet is particularly efficient for image denoising. Unlike Fourier transforms, the wavelet transform decomposes input data in terms of time and scale using a base wavelet function known as the mother wavelet. As a result, numerous types of wavelets with diverse properties might be constructed.

3.1 Image Color Spaces

3.1.1 Red, Green, and Blue (RGB) Color Model

RGB color space is extensively used and is typically utilized as the default color space for storing and expressing digital images. A linear or nonlinear RGB transformation can provide any other color space [104].

Computers, graphics cards, and displays or LCDs all employ the RGB color space. As seen in Figure.3.1, it is made up of three primary colors: red, green, and blue.

By combining the three base colors, any color can be created. Depending on how much of each base color is taken. Using this technique in reverse, a specific color can be broken down into its red, blue, and green components, as indicated in equations 3.1 to 3.3. These values can be used to identify similarly colored pixels in an image [105].

$$r = \frac{R}{R+G+B} \tag{3.1}$$

$$g = \frac{G}{R+G+B} \tag{3.2}$$

$$b = \frac{B}{R+G+B} \tag{3.3}$$

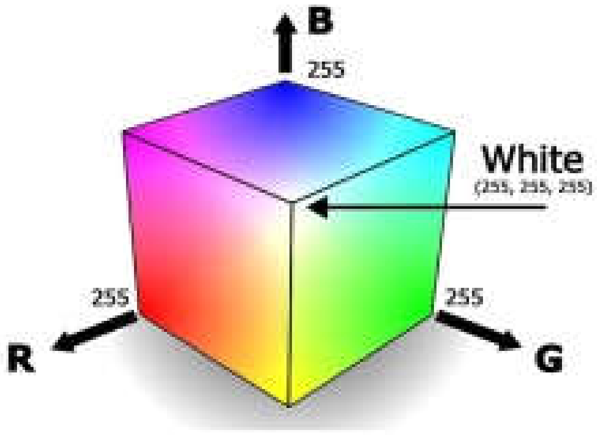


Figure 3. 1 : RGB Color Model

3.1.2 Hue Saturation Value (HSV) Color Model

HSV color system represents Hue, Saturation, and Value, and the three elements are independent. Hue is an angle ranging from 0 to 360 degrees. Normally, 0 degrees is red, 60 degrees is yellow, 120 degrees is green, 180 degrees is cyan, 240 degrees is blue, and 300 degrees is magenta. Hue indicates the type of color (such as red, blue, or yellow) or the hue of the color in the color spectrum. Colorful terms include red, yellow, and purple [106]. Saturation is the second component of the HSV color system. The saturation of a color is a measure of how pure the color is. Saturation is often measured on a scale of 0 to 1, with 0 representing gray and 1 representing pure primary color. The third component of HSV is value or intensity, which is a measure of how bright a color is or how much light emanates from it. Value might range from 0% to 100%. A color with a value of 100 percent will appear as brilliant as possible, while a color with a value of 0 will appear as dark as possible[107].

There are numerous benefits to extracting HSV color space from a picture. Other researchers have done some research on HSV color space in images. Reference [108] success in dehazing image using HSV color space in image. Image segmentation is more efficient when HSV color space is used [109]. HSV color space can also be utilized for categorization in color image data, and it produces better results [110].

There is a conversion method from RGB color system to HSV color space. Equations 3.1–3.3 are used to calculate the conversion [107].

$$V = max(r, g, b) \tag{3.4}$$

$$S = \begin{cases} 0\,, if\ V = 0 \\ 1 - \dfrac{min(r, g, b)}{V}\,, if\ V > 0 \end{cases} \tag{3.5}$$

$$H = \begin{cases} 0\,, if\ S = 0 \\ (60 * (g - b)/(S * V))\ \,, if\ V = r \\ (60 * [2 + (b - r)/(S * V))\ \,, if\ V = g \\ (60 * [4 + (r - g)/(S * V))\ \,, if\ V = b \end{cases} \tag{3.6}$$

$$H = H + 360\,, if\ H < 0 \tag{3.7}$$

where R, G, and B represents red, green, blue before normalization, r, g, b is red, green and blue after normalization, H is hue, S is saturation and V is value.

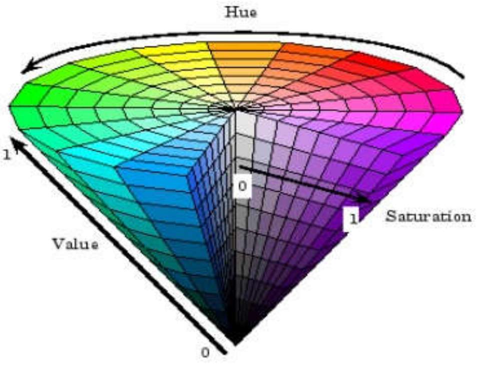


Figure 3. 2 : HSV Color Model

3.1.3 YCbCr (Luminance, Chrominance) Color Model

YCbCr is an encoded non-linear RGB signal that is extensively used in European television studios and for picture compression operations. As illustrated in Figure. 3.3, color is represented by luma (which is brightness computed from nonlinear RGB) formed as a weighted sum of RGB values [111]. In the digital video domain, YCbCr is a regularly used color space. Because the representation makes it simple to remove some extraneous color information, it is utilized in image and video compression formats such as JPEG, MPEG1, MPEG2, and MPEG4. YCbCr color space is distinguished by its ease of transformation and unambiguous separation of luminance and chrominance components [112]. In this format, luminance information is kept as a single component (Y), and chrominance information is encoded as two color-difference components (Cb and Cr). Cb is the difference between the blue component and the reference value. Cr denotes the difference between the red component and a reference value. YCbCr values can be calculated from RGB color space using equations 3.4 to 3.6.

$$Y = 0.299\,R + 0.287\,G + 0.11\,B \tag{3.8}$$

$$Cr = R - Y \tag{3.9}$$

$$Cb = B - Y \tag{3.10}$$

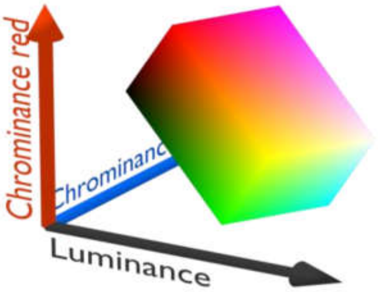


Figure 3. 3 : YCbCr Color Model

3.2 Image Features extraction

3.2.1. Convolutional neural networks

Convolutional neural networks (CNN) have produced impressive results in a variety of computer vision research areas in recent years. They demonstrated that CNN features are superior to other useful features in capturing the semantic and detailed features of a target object on a wide range of visual recognition tasks. A convolutional neural network (CNN) has an input layer, multiple hidden layers, and an output layer, just like a regular neural network (NN). Convolutional layers, ReLU layers, pooling layers, and fully-connected layers are common components of the hidden layers [113].

Neural networks take a single vector as input and transform it through a series of hidden layers. Each hidden layer consists of a set of neurons, each of which is fully connected to all neurons in the previous layer, and neurons in a single layer function completely independently and does not share any connections. The final fully-connected layer is known as the "output layer," and it represents the class scores in classification settings[113].

Regular Neural Nets do not scale well to full images. Because images in CIFAR-10 are only 32x32x3 (32 wide, 32 high, 3 color channels), a single fully-connected neuron in the first hidden layer of a regular Neural Network would have 32*32*3 = 3072 weights. This amount appears manageable, but this fully-connected structure clearly does not scale to larger images. For example, a more respectable image size, such as 200x200x3, would result in neurons with 200*200*3 = 120,000 weights. Furthermore, we would almost certainly want to have several such neurons, so the parameters would quickly add up! Clearly, full connectivity is wasteful, and the large number of parameters would quickly lead to over fitting.

Convolutional Neural Networks capitalize on the fact that the input consists of images by constraining the architecture in a more sensible manner. ConvNet layers, unlike regular Neural Network layers, have neurons arranged in three dimensions: width, height, and depth. (It is important to note that the term depth here refers to the third dimension of an activation volume, not the depth of a full Neural Network, which can refer to the total number of layers in a network.) For example, in CIFAR-10, the input images are an activation input volume with dimensions 32x32x3 (width, height, depth respectively). The neurons in a layer will only be connected to a small region of the layer preceding it, rather than all of the neurons in a fully connected manner. Furthermore, for CIFAR-10, the final output layer would have dimensions 1x1x10, because by the end of the ConvNet architecture, we will have reduced the entire image into a single vector of class scores arranged along the depth dimension. Here is an example:

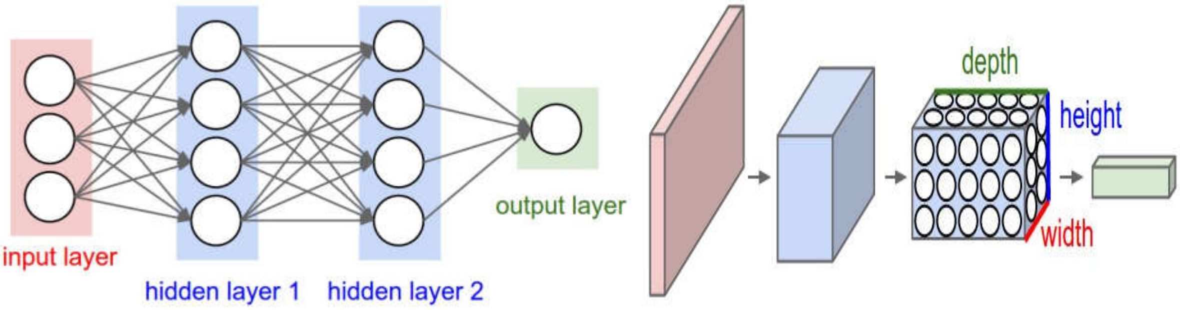


Figure 3. 4 : Left: A regular 3-layer Neural Network. Right: A ConvNet arranges its neurons in three dimensions (width, height, depth).

As visualized in one of the layers. Every layer of a ConvNet transforms the 3D input volume to a 3D output volume of neuron activations. In this example, the red input layer holds the image, so its width and height would be the dimensions of the image, and the depth would be 3 (Red, Green, Blue channels).

Layers used to build ConvNets

As previously stated, a simple ConvNet is a series of layers, and each layer of a ConvNet transforms one volume of activations to another using a differentiable function. ConvNet architectures are built using three types of layers: Convolutional Layer, Pooling Layer, and Fully-Connected Layer (exactly as seen in regular Neural Networks). These layers will be stacked to form a full ConvNet architecture.

Example Architecture: Overview. We will go into more details below, but a simple ConvNet for CIFAR-10 classification could have the architecture [INPUT - CONV - RELU - POOL - FC]. In more detail:

INPUT [32x32x3] will hold the raw pixel values of the image, in this case an image of width 32, height 32, and with three color channels R,G,B.

CONV layer will compute the output of neurons that are connected to local regions in the input, each computing a dot product between their weights and a small region they are connected to in the input volume. This may result in volume such as [32x32x12] if we decided to use 12 filters.

RELU layer will apply an elementwise activation function, such as the max(0,x) thresholding at zero. This leaves the size of the volume unchanged ([32x32x12]).

POOL layer will perform a downsampling operation along the spatial dimensions (width, height), resulting in volume such as [16x16x12].

FC (i.e. fully-connected) layer will compute the class scores, resulting in volume of size [1x1x10], where each of the 10 numbers correspond to a class score, such as among the 10 categories of CIFAR-10. As with ordinary Neural Networks and as the name implies, each neuron in this layer will be connected to all the numbers in the previous volume.

ConvNets do this by layering the original image from the original pixel values to the final class scores. It should be noted that some layers have parameters while others do not. The CONV/FC layers, in particular, perform transformations that are dependent not only on the activations in the input volume, but also on the parameters (the weights and biases of the neurons). The RELU/POOL layers, on the other hand, will implement a fixed function. Gradient descent will be used to train the parameters in the CONV/FC layers so that the class scores computed by the ConvNet are consistent with the labels in the training set for each image.

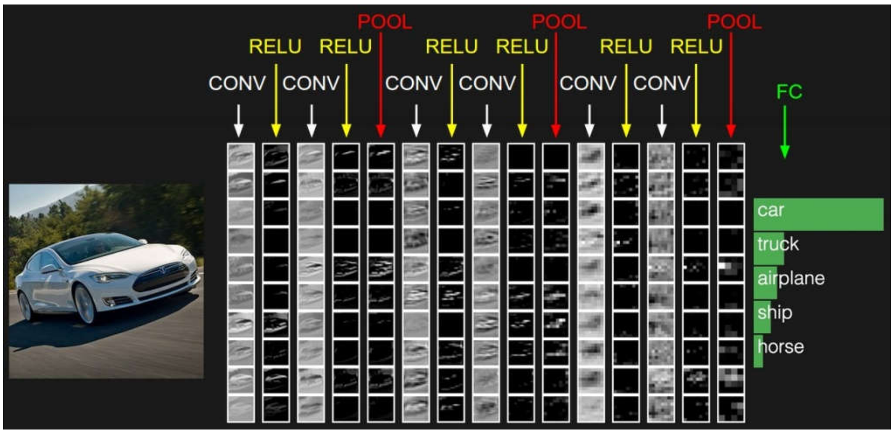


Figure 3. 5 : The activations of an example ConvNet architecture.

The first volume contains the raw image pixels (on the left), and the last volume contains the class scores (right). A column represents the volume of activations along the processing path. Because 3D volumes are difficult to visualize, we arrange the slices of each volume in rows. The last layer volume contains the scores for each class, but we only display the top 5 sorted scores and print the labels for each one here. The architecture depicted here is a miniature VGG Net, which we will go over later.

We now describe the individual layers, as well as the specifics of their hyper parameters and connectivity's.

3.2.1.1. Convolutional Layer :

A CNN's core building block is the convolution layer. It is made up of a series of filters. A filter can be thought of as a smaller window that convolves (slides) across the input image and computes dot products between the filter's values and the input image's pixel values. This action produces a 2-dimensional activation map that displays the filter's responses at each position. Each convolutional layer contains a set of filters, each of which generates a separate 2-dimensional map. The output of the convolutional layer is produced by stacking these separate activation maps. Figure 3.5 depicts the activation maps in a typical CNN architecture.

The filters' width and height are hyper-parameters, while their depth must be equal to the depth of the input image. Other hyper-parameters to consider, all of which will affect the size of the output image, are as follows:

- The number of filters. This will have an effect on the number of stacked activation maps in the layer's output.
- Stride. The stride specifies how many steps the filter will take while moving across the image. When the stride is set to two or more, the filter will jump two or more pixels at a time, resulting in an output with a smaller width and height than the input.
- Zero-padding surrounds the input image with zeros. Most commonly, zero-padding is used to preserve the size of the input to the output.

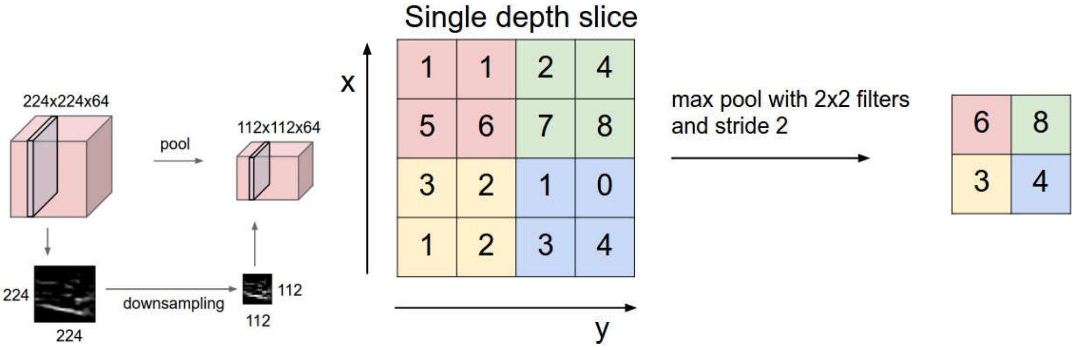
It is now possible to see how these hyper-parameters affect the size of the output. For example, if we have an input image of size 10 x 10 and a filter of size 3 x 3 with stride 1 and zero-padding = 0, we will get an output image of size 8 x 8. To keep the size, we would set zero-padding to 1, and the output would be 10 x 10. In this example, we wouldn't be able to set the stride to 2 with zero-padding = 0 because the filter would not fit evenly in the image. The stride must be set so that the filter can slide smoothly across the image.

3.2.1.2. Pooling Layer

The pooling layer's purpose is to reduce the number of parameters and computation in the network by reducing the spatial size of the image representations. The most common type of pooling is max-pooling. A max-pooling layer with a filter size of 2 x 2 and stride 2 moves across the input image, using the MAX operation at each step. The action removes 3/4 of the information because the filter takes the maximum of four numbers. The max-pooling layer has two commonly used settings: Size = 2 x 2 with stride = 2, and size = 3 x 3 with stride = 2.

The second setting is known as overlapping pooling, and it was one of the contributions of [114].

In addition to max pooling, the pooling units can also perform average pooling and L2-norm pooling. Average pooling was commonly used in the past but has recently fallen out of favor in favor of the max pooling operation, which has been shown to work better in practice.



1	1	2	4
5	6	7	8
3	2	1	0
1	2	3	4

6	8
3	4

Figure 3. 6 : Pooling layer

The pooling layer downsamples the volume spatially and independently in each depth slice of the input volume. In this example. Left: the input volume of size [224x224x64] is pooled with filter size 2, stride 2 into an output volume of size [112x112x64]. The volume depth is preserved. Right: The most common downsampling operation is max, which results in max pooling, as shown here with a stride of 2. That is, each maximum is divided by four numbers (little 2x2 square).

3.2.1.3. ReLU Layer

Convolutional layers are frequently followed by Rectified Linear Unit (ReLU) layers, which use the activation function.

$$f(x) = max(0, x) \qquad (3.11)$$

to the input x. This increases the non-linearity of the network while also removing negative values from the activation maps. Traditionally, other function were used such as $f(x) = tanh(x)$ or the sigmoid function $f(x) = (1 + e^{-1})^{-1}$, but the ReLU function has shown to be faster and is generally preferred [114, 115].

3.2.1.4. Fully-Connected Layer

Similar to conventional multi-layer perceptrons, the neurons in the fully connected layers are connected to all activations in the previous layer. Note that a convolutional layer can be viewed as a fully-connected layer with neurons acting as filters, with the exception that the neurons in the convolutional layers are connected to a local region in the input image and many neurons share parameters [113]. Using the output from the convolutional and pooling layers, it has been demonstrated that it is possible to omit the fully connected layers from a CNN and still achieve good performance[116].

3.2.1.5. Loss Layer (Softmax)

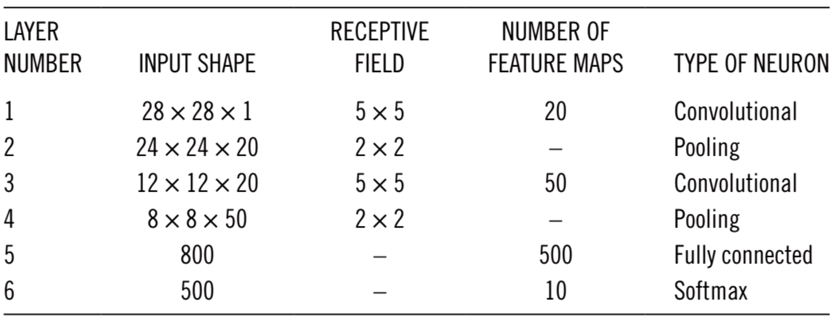
Typically, the loss layer is the final layer of a neural network. The objective of the loss layer during training is to specify a penalty between the prediction and the true label (target). For classification, the softmax algorithm is utilized. The softmax layer receives its input from the previous layer and outputs a probability distribution of K probabilities, where K is the number of classes during training. The output of the softmax layer is used to calculate a loss, which is then back-propagated throughout the network [117].

3.2.1.6. CNN architectures

There are several architectures with names in the field of Convolutional Networks. The most common are:

LeNet[118]**.** Prof. Yann LeCun's CNN for digit recognition is one of the earliest and most popular CNNs. This network is now commonly known as LeNet5, or simply LeNet. Table 4.1 shows the detailed network architecture of LeNet5, a CNN with two convolutional layers and one fully connected layer.

Table 3. 1 : A modern-Day Reincarnation of leNet for MNIST Classification



LAYER NUMBER	INPUT SHAPE	RECEPTIVE FIELD	NUMBER OF FEATURE MAPS	TYPE OF NEURON
1	28 × 28 × 1	5 × 5	20	Convolutional
2	24 × 24 × 20	2 × 2	–	Pooling
3	12 × 12 × 20	5 × 5	50	Convolutional
4	8 × 8 × 50	2 × 2	–	Pooling
5	800	–	500	Fully connected
6	500	–	10	Softmax

AlexNet[114]**.** Developed by Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton, was the first work that popularized Convolutional Networks in Computer Vision. In 2012, AlexNet was entered into the ImageNet ILSVRC challenge and significantly outperformed the second runner-up (top 5 error of 16 percent compared to runner-up with 26 percent error). The Network's architecture was very similar to LeNet's, but it was deeper, larger, and featured Convolutional Layers stacked on top of each other (previously it was common to only have a single CONV layer always immediately followed by a POOL layer).

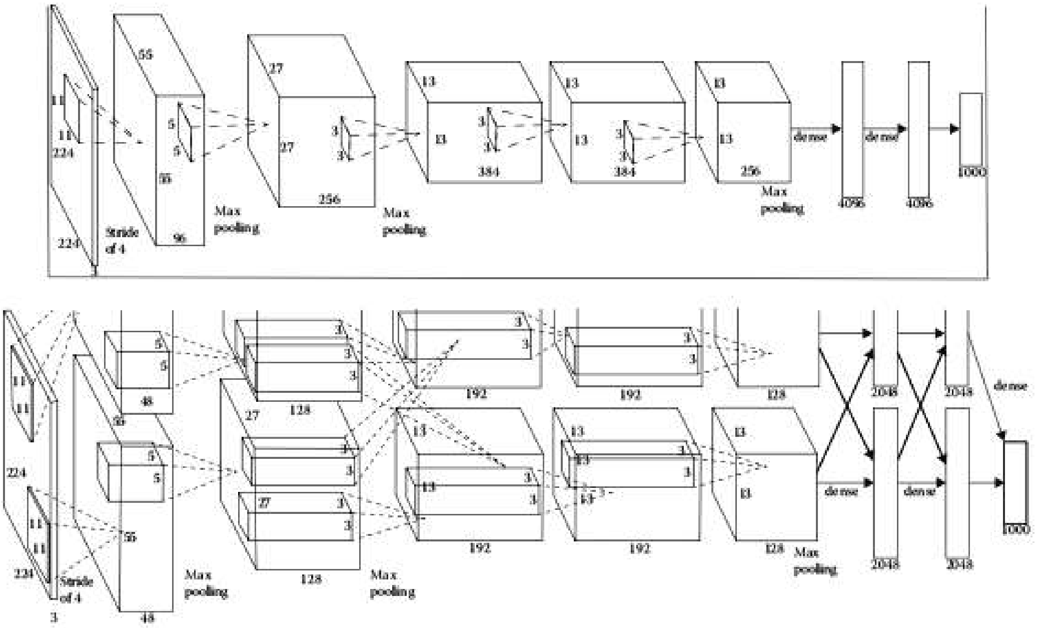
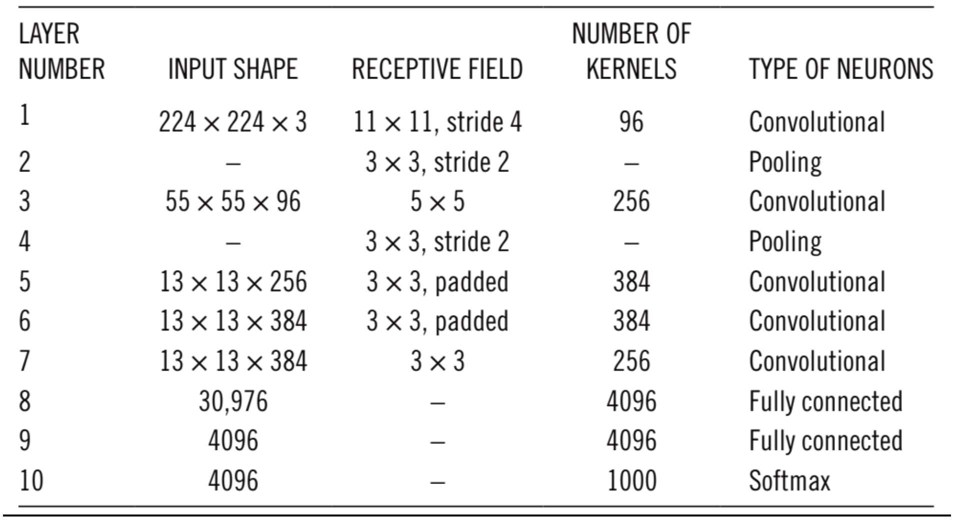


Figure 3. 7 : AlexNet Architecture .

Table 3. 2 : A modern-Day Reincarnation of leNet for MNIST Classification



LAYER NUMBER	INPUT SHAPE	RECEPTIVE FIELD	NUMBER OF KERNELS	TYPE OF NEURONS
1	224 × 224 × 3	11 × 11, stride 4	96	Convolutional
2	–	3 × 3, stride 2	–	Pooling
3	55 × 55 × 96	5 × 5	256	Convolutional
4	–	3 × 3, stride 2	–	Pooling
5	13 × 13 × 256	3 × 3, padded	384	Convolutional
6	13 × 13 × 384	3 × 3, padded	384	Convolutional
7	13 × 13 × 384	3 × 3	256	Convolutional
8	30,976	–	4096	Fully connected
9	4096	–	4096	Fully connected
10	4096	–	1000	Softmax

ZF Net[119]**.** Matthew Zeiler and Rob Fergus' Convolutional Network was the ILSVRC 2013 winner. It was dubbed the ZFNet. It improved on AlexNet by adjusting the architecture hyperparameters, specifically by increasing the size of the middle convolutional layers and decreasing the stride and filter size on the first layer.

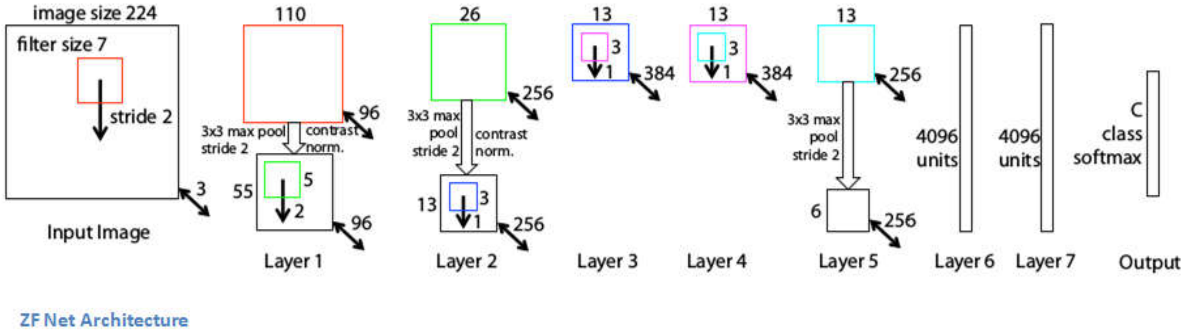


Figure 3. 8 : ZF Net Architecture .

GoogLeNet[120]**.** The ILSVRC 2014 winner was a Google Convolutional Network from Szegedy et al. Its main contribution was the creation of an Inception Module, which significantly reduced the number of parameters in the network (4M, compared to AlexNet with 60M). Furthermore, this paper employs Average Pooling rather than Fully Connected layers at the top of the ConvNet, removing a large number of parameters that do not appear to be important. There have also been several GoogLeNet followup versions, the most recent being Inception-v4. Figure 4.10 depicts an inception module.

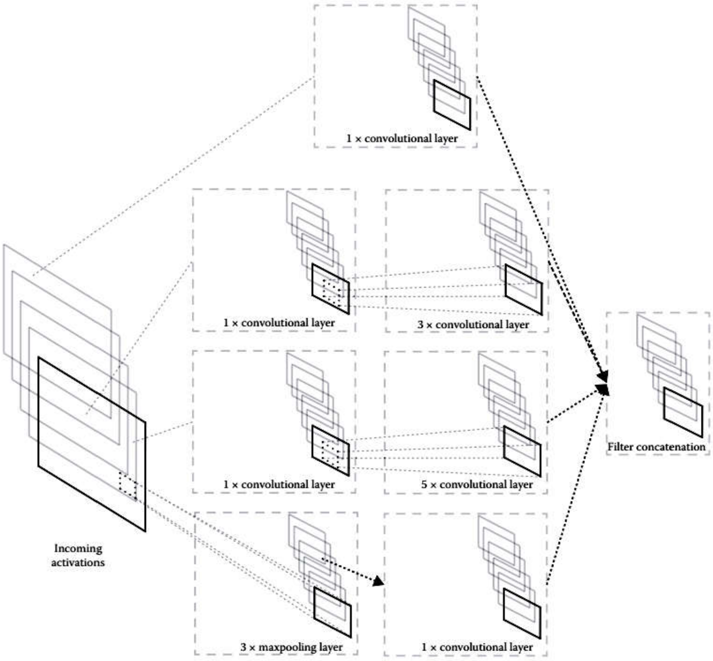
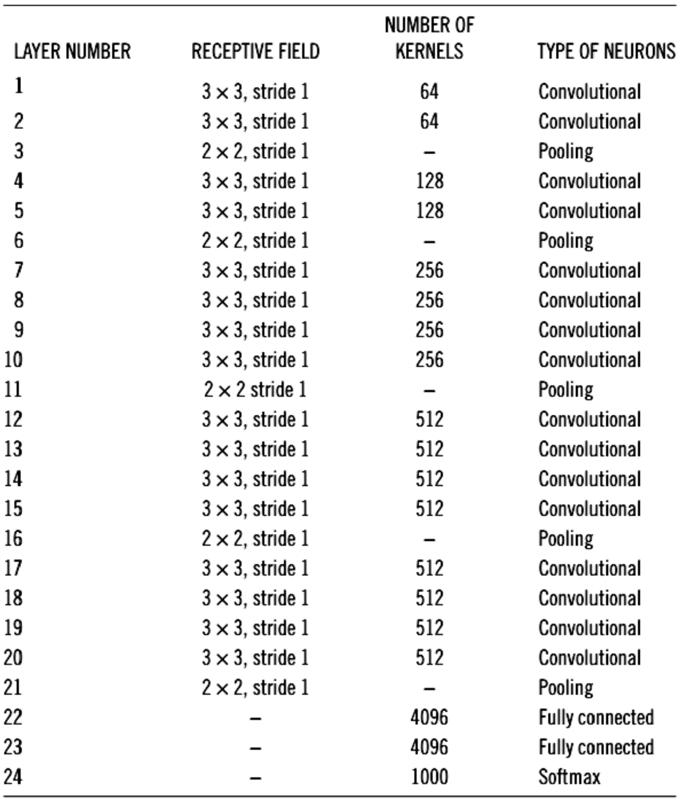


Figure 3. 9 : GoogLeNet's inception module.

VGGNet[121]. The VGGNet network, created by Karen Simonyan and Andrew Zisserman, finished second in the ILSVRC 2014. Its main contribution was demonstrating that network depth is a critical component for good performance. Their best network has 16 CONV/FC layers and an appealingly homogeneous architecture that only performs 3x3 convolutions and 2x2 pooling from start to finish. Their pretrained model is ready to use in Caffe. The VGGNet has the disadvantage of being more expensive to evaluate and requiring a lot more memory and parameters (140M). The majority of these parameters are in the first fully connected layer, and it has since been discovered that these FC layers can be removed with no performance loss, significantly reducing the number of required parameters. The architecture of interest is shown in Table 4.3.

Table 3. 3 : VGG Network



LAYER NUMBER	RECEPTIVE FIELD	NUMBER OF KERNELS	TYPE OF NEURONS
1	3 × 3, stride 1	64	Convolutional
2	3 × 3, stride 1	64	Convolutional
3	2 × 2, stride 1	–	Pooling
4	3 × 3, stride 1	128	Convolutional
5	3 × 3, stride 1	128	Convolutional
6	2 × 2, stride 1	–	Pooling
7	3 × 3, stride 1	256	Convolutional
8	3 × 3, stride 1	256	Convolutional
9	3 × 3, stride 1	256	Convolutional
10	3 × 3, stride 1	256	Convolutional
11	2 × 2 stride 1	–	Pooling
12	3 × 3, stride 1	512	Convolutional
13	3 × 3, stride 1	512	Convolutional
14	3 × 3, stride 1	512	Convolutional
15	3 × 3, stride 1	512	Convolutional
16	2 × 2, stride 1	–	Pooling
17	3 × 3, stride 1	512	Convolutional
18	3 × 3, stride 1	512	Convolutional
19	3 × 3, stride 1	512	Convolutional
20	3 × 3, stride 1	512	Convolutional
21	2 × 2, stride 1	–	Pooling
22	–	4096	Fully connected
23	–	4096	Fully connected
24	–	1000	Softmax

ResNet[77]. Kaiming He et al Residual .'s Network was the ILSVRC 2015 winner. It makes extensive use of batch normalization and has special skip connections. The architecture also lacks fully connected layers at the network's end. The reader is also directed to Kaiming's presentation (video, slides), as well as some recent Torch experiments that replicate these networks. ResNets are by far the most advanced Convolutional Neural Network models available today, and they are the default choice for using ConvNets in practice .

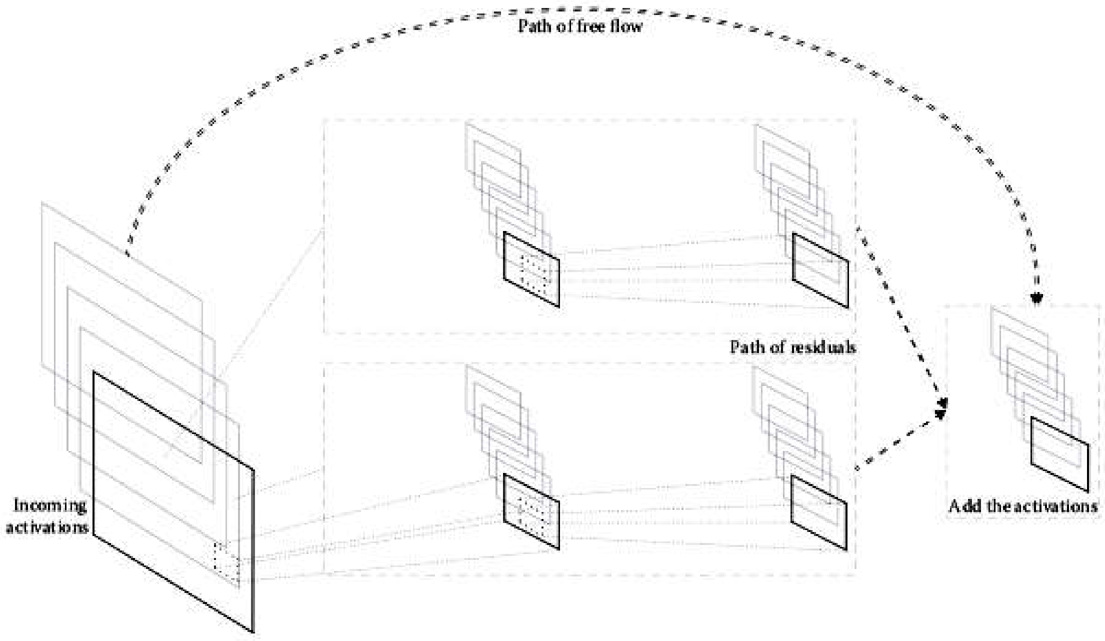


Figure 3. 10 : A typical residual layer architecture.

3.2.2 Histograms of Oriented Gradients

To extract shape features, we used the Histograms of Oriented Gradients (HOG) technique proposed by N. Dalal et al.[122] on each sub-band of a wavelet transformed image, which stores information about the shapes contained in the image, represented by histograms of the slopes of the object edges. Each bin in the histogram represents the number of edges with orientations within a specific angular range. The concatenation of computed histograms from all four sub-bands yields the HOG descriptor, which stores shape and texture information and can be used for content-based image retrieval. Because DBC and Haar wavelet transforms are used to enhance the edges and other high-frequency local features, the use of HOG yields more shape information of an image with enhanced edges than an unprocessed image. The steps below are used to compute gradient local histograms.

The first step is to compute the image's gradients, then build orientation histograms for each cell, and finally normalize the histograms within each block of cells.

Gradient Computation: The gradient of an image I is obtained by filtering it with a horizontal and vertical discrete derivative mask in one dimension [123].

$$D_X = [-1\ 0\ 1]\ \ and\ D_Y = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \tag{3.12}$$

where D_X and D_Y are horizontal and vertical masks respectively and obtain the X and Y derivatives using following convolution operation.

$$I_X = I\ * D_X \qquad and\ \ I_Y = I\ * D_Y \tag{3.13}$$

The magnitude of the gradient is

$$|G| = \sqrt{I_X^2 + I_Y^2} \tag{3.14}$$

The orientation of the gradient is given by

$$\theta = arctan\frac{I_X}{I_Y} \tag{3.15}$$

Orientation Binning: The second step is to create the cell histograms. Based on the values found in the gradient computation, each pixel calculates a weighted vote for an orientation-based histogram channel. The cells are rectangular, and the histogram channels are evenly spaced from 0* to 180* or 0* to 360*, depending on whether the gradient is unsigned or signed. N. Dalal and B. Triggs discovered that unsigned gradients combined with 9 histogram channels performed best in their experiments[123].

Descriptor Blocks: To account for changes in illumination and contrast, gradient strengths should be regionally normalized, which necessitates grouping the cells into larger spatially connected blocks. The HOG descriptor is then the vector of the elements of the normalized cell histograms from all block regions. These blocks generally overlap, which means that each cell contributes to the final descriptor more than once.

A normalization factor is then computed over the block, and all histograms within this block are normalized using this normalization factor. After this step, all of the histograms will be concatenated into a single feature vector. There are several approaches to block normalization. Let v be the non-normalized vector containing all histograms in a given block, $v - k$ be its k-norm for $k = 1,2$, and e be some small constant. These methods can be used to calculate the normalization factor f [123].

$$\mathrm{L1-norm} : f = \frac{v}{\|v\|_1 + e} \tag{3.16}$$

$$\text{L2} - \text{norm} : f = \frac{v}{\|v\|_2^2 + e^2} \tag{3.17}$$

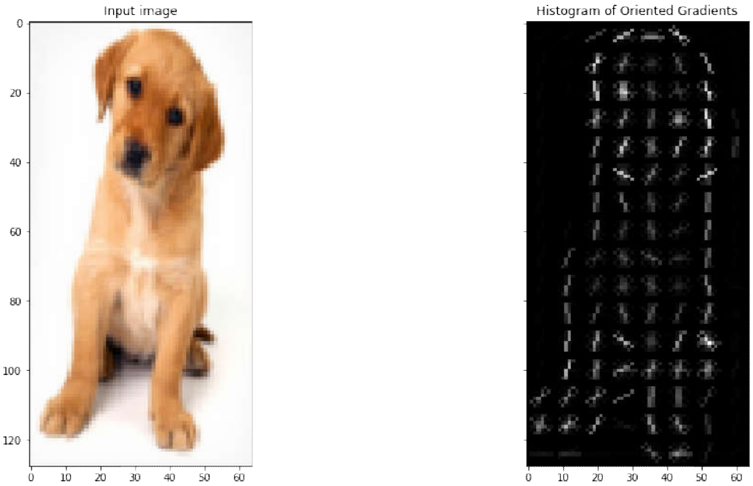


Figure 3. 11 : image and his features extraction with HOG**.**

3.3 Conclusion

In this chapter, from based on the main component of visual tracking is an image, we explained the most important things related to images in terms of color and extraction of features, and we also explained the wavelet transformations that are of great importance in processing the images that we are going to use in the next section.

Chapter 4 :

Wavelet transforms

4.1 Introduction

Wavelet transform is a great tool in image and signal processing. Many denoising approaches using wavelet in the literature show that the wavelet is very efficient for image denoising . Unlike the Fourier transforms, the wavelet transform decomposes input data in terms of time and scale by basis wavelet function, called mother wavelet .So various types of wavelet with different properties could be designed. In addition, researchers have developed different types of approaches to reinforce the theory by filling up some possible deficiencies.

In this chapter, the wavelets and its various aspects such as types of wavelets with their mathematical and graphical properties, applications, continuous wavelet transforms (CWT) , and discrete wavelet transforms (DWT) .To explain the system framework and formulation in order to conduct the present study, we have focused on different theoretical and graphical representations given by the inventors of wavelets.

4.2 Wavelets and Wavelet transforms

Wavelet transformations are built on small waves, called wavelets, of altering frequency with a finite duration[124]. Wavelet is a Mathematical tool applied for the hierarchical decomposition of an image and to transform an image from spatial domain to frequency domain. Wavelet permits certain functions regarding a rough comprehensive shape and details that range from wide to small for an image curve or a surface. In wavelet analysis the signal to be analyzed is accumulated with a wavelet function and then the transform is determined for each segment created. The Wavelet Transform, at high frequencies, comply significant time resolution and deficient frequency resolution, while at low frequencies; the Wavelet Transform complies significant frequency resolution and deficient time resolution. Wavelet analysis is a mechanism that has appeared recently from the mathematical community and has established various applications in areas like image processing, signal processing, numerical analysis, music synthesis, and computer graphics[125]. Wavelets have many appreciative properties such as vanishing moments, hierarchical and multiresolution structure decomposition, linear time and space complexity of the wavelet transformations, decorrelated coefficients, and a wide range of basis functions. The wavelet family is generated from a specific prototype function that is called a “mother wavelet”[94, 96, 126].

Given a real variable t, the function $\psi(\mathrm{t}_0)$ is called a “mother wavelet” assuming that it oscillates, averaging to zero $\int \psi(\mathrm{t}_0)\mathrm{dt} = 0$ and that is nicely localized (i.e., rapidly decreases to zero when $|\mathrm{x}|$ tends to infinity).The current use of the term "wavelet" is because of Grossmann's and Morlet's work on geophysical signal processing, which guided to the representation of the continuous wavelet transform[127].

In the growth of wavelets, the concepts from many different fields such as subband coding and computer vision have amalgamated. Magnificent works on this field are [128], and Daubechies paper "Where do the wavelets come from"[129]. In wavelet analysis, the scale can be portrayed as the inverse of frequency. STFT fragments the time-frequency level into identical blocks, the wavelet transform represents as a nanoscopic tool pointing on compact time occurrence as the scale reduces.

4.3 Mother Wavelet

Mother wavelets (ψ) are simply known as transformation functions or wavelet functions. The name wavelet came from the small waves or oscillatory functions[95]. The wavelet function is called mother wavelet (ψ) or analyzing wavelet if it is used as a basis function for all the transform based processing[94, 98, 130].

The wavelets are produced from a single basic wavelet $\psi(\mathrm{t}_0)$, called the mother wavelet function by the scaling and translation. Mathematically, the mother wavelet can be represented as[130]:

$$\psi_{\mathrm{S,T}}(\mathrm{t}_0) = \frac{1}{\sqrt{\mathrm{S}}}\psi\left(\frac{\mathrm{t}_0 - \mathrm{T}}{\mathrm{S}}\right) \tag{4.1}$$

Where S, is the scaling function and T is the translation property of mother wavelet.

4.4 Mathematical Analysis of Wavelet Transforms

A function $\psi \in \mathrm{L}^2(\mathrm{R})$ is known as orthonormal wavelet if it is used to define a Hilbert basis function called complete orthonormal system, for the Hilbert space $\mathrm{L}^2(\mathrm{R})$ of the square integrable functions [98].

The Hilbert basis function is constructed as a family of $\{\psi_{\mathrm{i,j}}: \mathrm{i}, \mathrm{j} \in \mathbb{Z}\}$ by means of dyadic translations and dilations of ψ

$$\psi_{\mathrm{i,j}}(\mathrm{x}) = 2^{\frac{\mathrm{i}}{2}}\,\psi(2^{\mathrm{i}}\mathrm{x} - \mathrm{j}) \tag{4.2}$$

for integers $\mathrm{i}, \mathrm{j} \in \mathbb{Z}$.

The integral wavelet transform is the integral transform defined as

$$\left[W_{\psi}f\right](A,B)=\frac{1}{\sqrt{A}}\int_{-\infty}^{+\infty}\psi\left(x-\frac{B}{A}\right)f(x)dx \tag{4.3}$$

The wavelet coefficients $C_{i,j}$ are then expressed as

$$C_{i,j}=\left[W_{\psi}f\right]\left(2^{-i},j2^{-i}\right) \tag{4.4}$$

Here, $A=2^{-i}$ is known as the binary dilation or dyadic dilation and $B=j2^{-i}$ is known as the binary or dyadic position.

4.5 Types of Wavelets

Different families of wavelets that have demonstrated to be especially very important are incorporated in Digital Image Processing.

4.5.1 Haar Wavelets

Wavelet starts with Haar wavelet. Haar wavelet is the first and simplest wavelet among the wavelet families. Discrete Haar wavelet can be defined as functions computed using sampling of the Haar functions at 2^n points [131]. The Haar functions can be appropriately depicted by using the matrix shape. In the Haar matrix, each row comprises of Haar functions. The Haar wavelet is discontinuous and its appearance is like a step function. It represents the same wavelet as Daubechies *db1* [132].

According to Haar wavelet each function is continuous on a particular interval $[0,1]$ which can be represented equally and convergent series with reference to the elements of the system. Mathematically, the original Haar definition can be presented as [131]:

$$\mathrm{Haar}(0,x)=1\,,\mathrm{for}\ x\in[0,1] \tag{4.5}$$

$$\mathrm{Haar}(1,x)=\left.\begin{matrix}1\,,\mathrm{for}\ x\in\left[0,\frac{1}{2}\right]\\ -1\,,\mathrm{for}\ x\in\left[\frac{1}{2},1\right]\end{matrix}\right\} \tag{4.6}$$

Where, x is the discrete point of the function calculation interval.

The Haar matrix of all ranges can be achieved by the following

$$M(n)=\left\{\begin{matrix}M(n-1) & K[1 & 1]\\ 2^{\frac{(n-1)}{2}}D(n-1) & K[1 & -1]\end{matrix}\right\}\quad M(0)=1 \tag{4.7}$$

And $M(n)\neq M(n)^t$ for $n>1$ and $M(n)^{-1}=2^{-n}.M(n)^t$

Where, $\mathrm{M(n)}$ indicates the matrix of the discrete Haar functions of degree 2^{n} , $\mathrm{D(n)}$ indicates the identity matrix of degree 2^{n} and K indicates the Kronecker (tensor) product.

The Haar wavelet functions are asymmetric and the elements of Haar functions are 1, -1 or 0, the product of these functions can be obtained by the powers of $\sqrt{2}$. Equation (3.6) is called the discrete and orthogonal Haar functions within an interval of [0, 1).

4.5.2 Daubechies Wavelets

Daubechies, one of the famous research scientists in the world of wavelet research, invented the compactly supported orthonormal wavelets and hence the construction of discrete wavelet transform becomes executable. The names of the Daubechies family wavelets are denoted as *dbn*, where *n* is the order, and *db* is the "family name" of the wavelet. *n* = 1 to 20. The *db1* wavelet is similar as Haar wavelet [126, 133].

Daubechies wavelet is a group of wavelet functions $\psi_{\mathrm{n,n}} \in \mathrm{N} \setminus \{0\}$, which satisfy some special conditions. The class $\psi_{\mathrm{n}}(\mathrm{x}-\mathrm{u})\,,\mathrm{u} \in \mathbb{Z}$, is an orthonormal function for a constant value of $\mathrm{n} \in \mathrm{N} \setminus \{0\}$. Each Daubechies wavelet function ψ_{n} is compactly supported. The index number *n* is also connected to the number of vanishing moments .

$$\int_{-\infty}^{+\infty} \mathrm{x}^{\mathrm{u}}\psi_{\mathrm{n}}(\mathrm{x})\,\mathrm{dx} = 0\,, \qquad 0 \leq \mathrm{u} \leq \mathrm{n} \tag{4.8}$$

Another important property of Daubechies wavelet is that the regularity of Daubechies wavelets can be extended linearly with its support width. The Daubechies wavelets are not either symmetric or asymmetric towards any axis, except *db1*, which is equivalent to Haar wavelet. Each Daubechies wavelets meet the admissibility condition and hence it assures a secure reconstruction [129].

4.5.3 Biorthogonal Wavelets

This family of wavelets represents the characteristics of linear phase, which is required for signal and image reconstruction. By applying two wavelets, one for decomposition (on the left side) and the other for reconstruction (on the right side) in place of the individual one, fascinating features are obtained .

According to orthogonal wavelets, $\psi(\mathrm{x})$ and $\varphi(\frac{\mathrm{x}}{2})$ are correlated by a scaling function that is a outcome of the addition of the resolution spaces from coarse level to fine level [134].

$$\frac{1}{\sqrt{2}}\psi\left(\frac{x}{2}\right) = \sum_{y=-\infty}^{\infty} g(y)\varphi(x-y) \tag{4.9}$$

Identical equation exists for the biorthogonal functions that compute the filters *l2* and *g2*. A biorthogonal wavelet has m vanishing moments in case of its dual scaling function produces polynomials up to degree *m*.

Hence there is an identical theorem in-between vanishing moments and the number of zeroes of the filter's displacement, situation should be like that the duality has to be taken into consideration. So, the following three properties are identical:

- The wavelet w has v vanishing moments.
- The dual scaling function $j2$ produces polynomials up to degree *v*.
- The displacement function of the dual filter $l2$ and it $v-1$ first derivatives vanish at $w = v$

and the dual outcome is also valid. Duality comes naturally, because the filters compute the degree of the polynomials that may be produced by the scaling function. This degree is equal to the number of vanishing moments of the *dual* wavelet.

4.5.3.1 Symmetrical Property

Except the orthogonal wavelet, it is feasible integrating the biorthogonal wavelets and scaling functions those are symmetric, asymmetric and compactly supported. It constructs feasibility to apply the folding technique to develop wavelets on an interval [135].

W, the filters *l* and *l*2 have and odd length and are symmetric with respect to 0, then the scaling functions will be an even length and are symmetric, and the wavelets are also symmetric. In case, the filters have an even length and are symmetric with reference to $n = \frac{1}{2}$, then the scaling functions are symmetric with reference to $n = \frac{1}{2}$, whilst the wavelets are asymmetric.

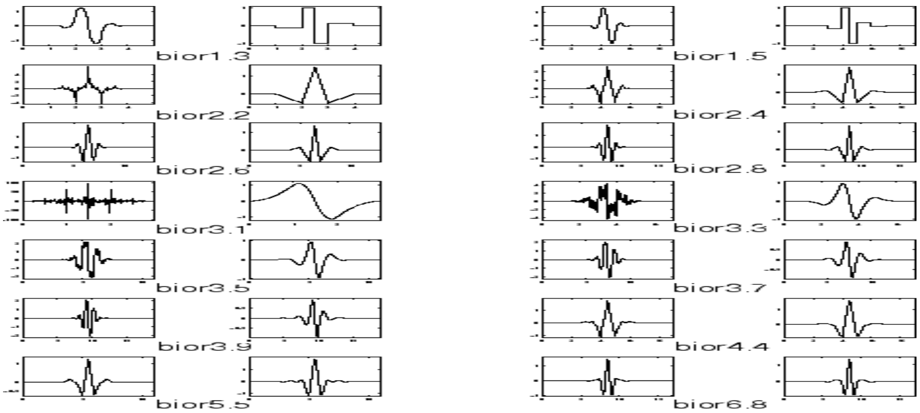


Figure 4. 1 : Biorthogonal wavelets

Figure 4.1 represents the boirthogonal scaling functions at different levels from biorthogonal coefficients *bior1.3* to *bior6.8* and differentiates them to the orthogonal scaling functions at the same levels. All the biorthogonal scaling functions holds the same size at each level other than the orthogonal bases which are layered.

4.5.4 Symlet Wavelets

The symlets are closely symmetrical wavelets proposed by Daubechies as the alterations to the *db* family. The features of the two wavelet families are similar. The *sym1* is equivalent to Haar wavelet. The wavelet function Ψ is represented below [133].

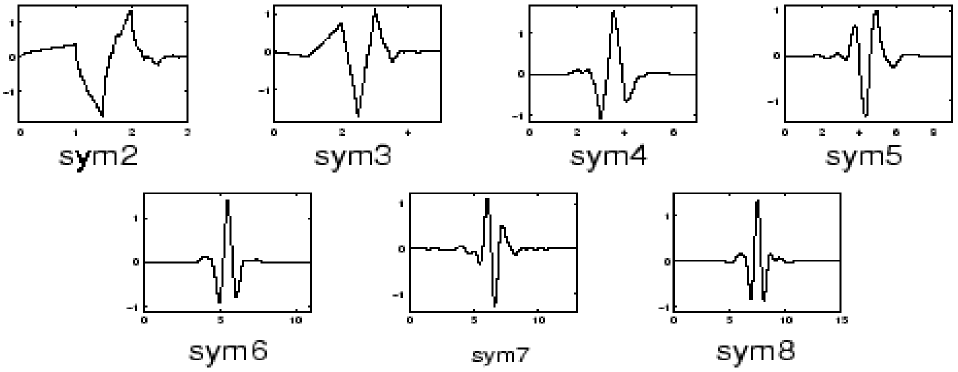


Figure 4. 2 : Symlet Wavelets

In figure 4.2, symlet scaling functions showing symlet coefficients from order 2 to order 8. The properties of symlet wavelets are similar as Daubechies wavelets.

Daubechies proposed alterations in her existing wavelets with increased symmetry are called symlet wavelets.

4.5.5 Morlet Wavelets (or Gabor Wavelet)

Morlet wavelet or Gabor wavelet has no scaling function, but it is straight-forward. The real-valued as well as complex-valued models of this wavelet survive [136].

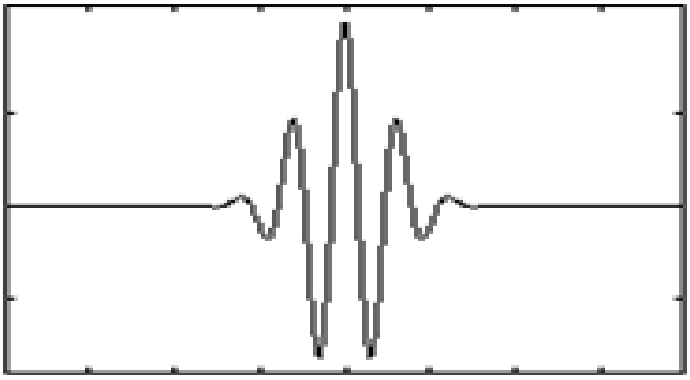


Figure 4. 3 : Morlet Wavelet

The above figure 4.3 depicts the Morlet Wavelet function at coefficient level 1 which is usually called as *Mor1*. The Morlet wavelet never satisfies any admissibility condition.

Based on Morlet wavelet, the mother wavelet may be defined as

$$\psi(t_0) = \left(e^{-jrt_0} - e^{\frac{-r^2}{2}}\right) e^{-\frac{t_0^2}{2}} \qquad (4.10)$$

Where, r is the reference frequency of mother wavelet, t_0 is the time, j is an imaginary variable and the term $e^{\frac{-r^2}{2}}$ is applied for correcting the non-zero mean of the complex sinusoid.

The common Morlet wavelet is constructed from the mother wavelet by scaling φ and displacement *d*

$$\psi_{\varphi d}(t_0) = \frac{1}{\sqrt{|\varphi|}}\left(e^{-jr\left(\frac{t_0-d}{\varphi}\right)} - e^{\frac{r^2}{2}}\right) e^{-\frac{\left(\frac{t_0-d}{\varphi}\right)^2}{2}} \qquad (4.11)$$

Both scale and displacement execute from $-\infty$ to ∞.

4.5.6 Mexican Hat Wavelets

Mexican Hat wavelet has no scaling function and it is derived from a function which is identical to the second derivative of the Gaussian probability density function. Mexican

Hat wavelet is also called the Ricker wavelet [137]. Mathematically, the Mexican Hat wavelet can be represented [138]

$$\psi(\mathrm{t}_0) = \frac{1}{\sqrt{3}} \pi^{\frac{1-\mathrm{t}_0{}^2}{4}} \mathrm{e}^{-\frac{\mathrm{t}_0{}^2}{2}} \qquad (4.12)$$

Where, t_0 indicates the time of the mother wavelet.

The complex category of the modified Mexican Hat can be built using the following equation.

$$\psi(\mathrm{t}_0) = \pi^{-\frac{\mathrm{e}^{-\mathrm{t}_0{}^{\frac{2}{8}}}}{4}} \cos\left(r\frac{\mathrm{t}_0}{2}\right) \qquad (4.13)$$

Where, t_0 indicates the time of the mother wavelet and r indicates the reference frequency of the mother wavelet.

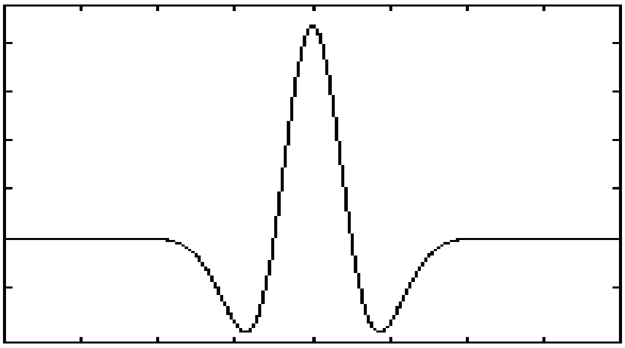


Figure 4. 4 : Mexican Hat Wavelet

The above figure 4.4 shows the Mexican Hat Wavelet function which is usually denoted as *mexh*. The Mexican Hat wavelet function is equivalent to the second derivative of the Gaussian probability of density function.

4.5.7 Meyer Wavelets

The Meyer wavelet and scaling function are defined in the frequency domain. The Meyer wavelet is not compactly supported but there exists a good approximation leading to FIR (Finite Impulse Response) filters that can be applied in the .

The figure 4.5 represents the basic Meyer wavelet function which is usually denoted by *Meyr*. Meyer wavelet function is not a compactly supported wavelet function but it holds good approximation coefficients with respect to DWT [137].

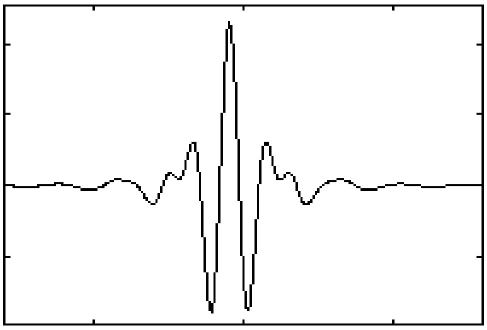


Figure 4. 5 : Meyer Wavelet

The Meyer wavelet is applied to achieve better localization features in the frequency domain. In general, the Meyer wavelet function may be achieved from the scaling function is as follows [139].

$$\psi(t_0) = \frac{1}{\sqrt{2\pi}} \sin\left(\frac{\pi}{2} y\left(\frac{3|t_0|}{2\pi} - 1\right)\right) e^{J\frac{t_0}{2}}, \text{ if } \frac{2\pi}{3} < |t_0| < \frac{4\pi}{3} \tag{4.14}$$

$$\psi(t_0) = \frac{1}{\sqrt{2\pi}} \cos\left(\frac{\pi}{2} y\left(\frac{3|t_0|}{2\pi} - 1\right)\right) e^{J\frac{t_0}{2}}, \text{ if } \frac{4\pi}{3} < |t_0| < \frac{8\pi}{3} \tag{4.15}$$

Otherwise,

$$y(t_0) = \begin{cases} 0 & \text{if } y < 0 \\ y & \text{if } 0 < y < 1 \\ 1 & \text{if } y > 1 \end{cases} \tag{4.16}$$

4.5.8 Lifting Wavelet Scheme

The lifting wavelet scheme works as an easy interpreter between each multiresolution analysis with a common scaling function. The lifting wavelet based scheme is used for construction and reconstruction of a discrete wavelet transform based images. The lifting scheme separates the level of liberty left out followed by the biorthogonal relations. The lifting based wavelet scheme is fast, easy to compute and the cost to perform any of the lifting based wavelet schemes is low. It can be used to speedup the fast wavelet transform method. The wavelets those which are based on Mallat's algorithm can be used through lifting wavelet scheme [140].

4.6 Applications of Wavelet Transforms

The wavelet transform is a mathematical tool which decomposes a signal into a representation and shows signal details as well as drifts as a function of time. Wavelet transform can be applied to represent short-term occurrences, minimize noise, compress image and data, and execute numerous activities.

The following are applications of wavelet transform :

- Data and image compression
- Image Denoising
- Image Segmentation
- Image Edge Detection
- Image Restoration
- Image Fusion
- Image Enhancement
- Pattern recognition
- Texture analysis

The following are the types of wavelet transforms:

- continuous wavelet transform (CWT)
- discrete wavelet transform (DWT)
- fast wavelet transform (FWT)
- wavelet packets
- complex wavelet transform

4.7 Discret Wavelet Transform

The two-dimensional discrete wavelet transform (2D-DWT) can be used to decompose an image into sub-signals, which present the original image components (I) under different frequency ranges. It means that 2D-DWT is used in the input side to ensure the process of splitting the original image (I) into four sub images such as (ILL, IHL, ILH and IHH). Firstly two down-sampling filters(noted as ↓2)of low (L) and high (H) bands are used yielding to two rows (IL and IH), then each obtained images in bothrowswill be passed through two filters of low (L) and high (H) down-sampling bands which means four filters are used in this phase to obtain four sub-images as two columns such as the first column is (ILL, IHL) and the second column is (ILH, IHH). Indeed, ILL presents the approximation coefficients matrix which is obtained from the passage through two simultaneous low-pass filters and the other three present the detail coefficients matrices IHL(horizontal features), ILH(vertical features), and IHH(diagonal features), as shown in figure 4.6 (left) .Moreover, the 2-D DWT has a separable characteristic with the scaling function $\Phi_{LL}(x, y)$, and three 2D-wavelets,$\Psi_{HL}(x, y)$,$\Psi_{LH}(x, y)$,and Ψ_{HH}, , which can be expressed as follows[141]:

$$\Phi_{LL}(x, y) = \Phi(x)\Phi(y) \tag{4.17}$$

$$\Psi_{HL}(x, y) = \Psi(x)\Phi(y) \tag{4.18}$$

$$\Psi_{LH}(x, y) = \Phi(x)\Psi(y) \tag{4.19}$$

$$\Psi_{HH}(x, y) = \Psi(x)\Psi(y) \tag{4.20}$$

Where $\Phi(x)$ and $\Phi(y)$ are the wavelet function following the x-axis (horizontal) and the y-axis (vertical), $\Psi(x)$ and $\Psi(y)$ are the horizontal and vertical 1D scaling functions.

In contrast, the inverse DWT (IDWT) is used in the output for adding four sub-images to the original one using up-sampling filters (notes ↑2) with the same concept as the DWT but with the inverse operation as shown in figure 4.6 (right). It is clear that the inputs are the four sub-images (ILL, IHL, ILH and IHH) and the output is the filtered original image (I).

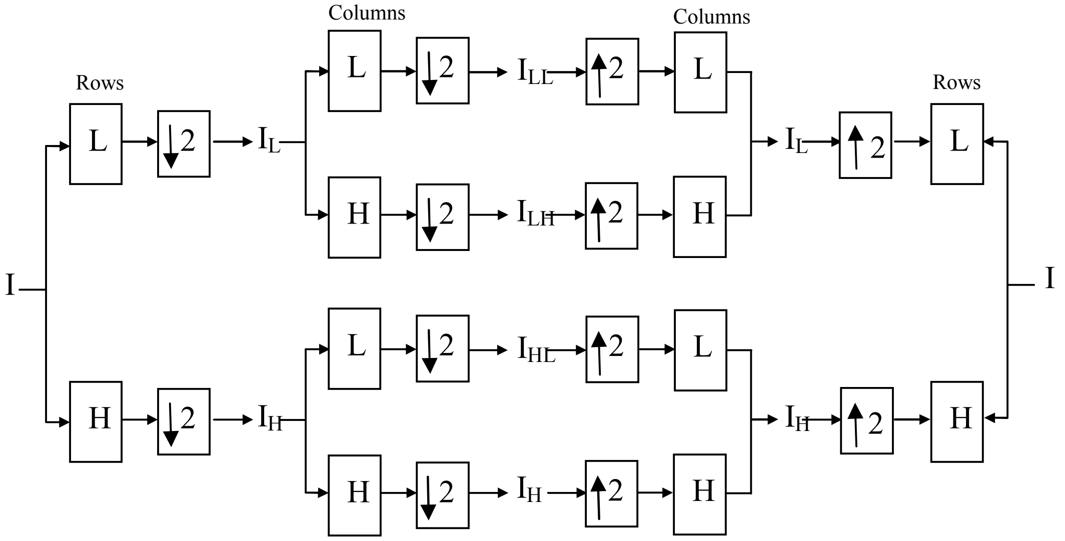


Figure 4. 6 : Downsample and Upsample comparison of DWT and IDWT.

4.8 Conclusion

This chapter provides a brief overview on wavelet transformations and the mathematical analysis of wavelet transformations. Different types of wavelets with their mathematical and graphical analysis, applications of wavelet transformations, the two-dimensional discrete wavelet transform are also discussed.

Chapter 5 :

Proposed Method

5.1 Introduction

This chapter introduces our visual object tracking method.

The main aim of the algorithm proposed in this chapter is to present a new contribution that can overcome the main difficulties encountered in visual tracking with most of the previously proposed approaches under target appearance changes such as severe occlusion and illumination variation. In this chapter, the proposed algorithm is described in detail where the different stages of the proposed tracking algorithm are shown in Figure 4.1.

Firstly, based on the works presented in [2, 3], the target location is estimated by learning four two-dimensional correlation filters with CNN features. Secondly, according to the properties of the input image, the selection of the use of the RGB or GRAY with the wavelet decomposition is carried out based on a proposed approche. Thirdly,the maximum value of the resulting maps is calculated from the correlation filters with convolutional features products of HOG feature based on the image template previously obtained. This calculated value is used as a parameter to update the correlation filter

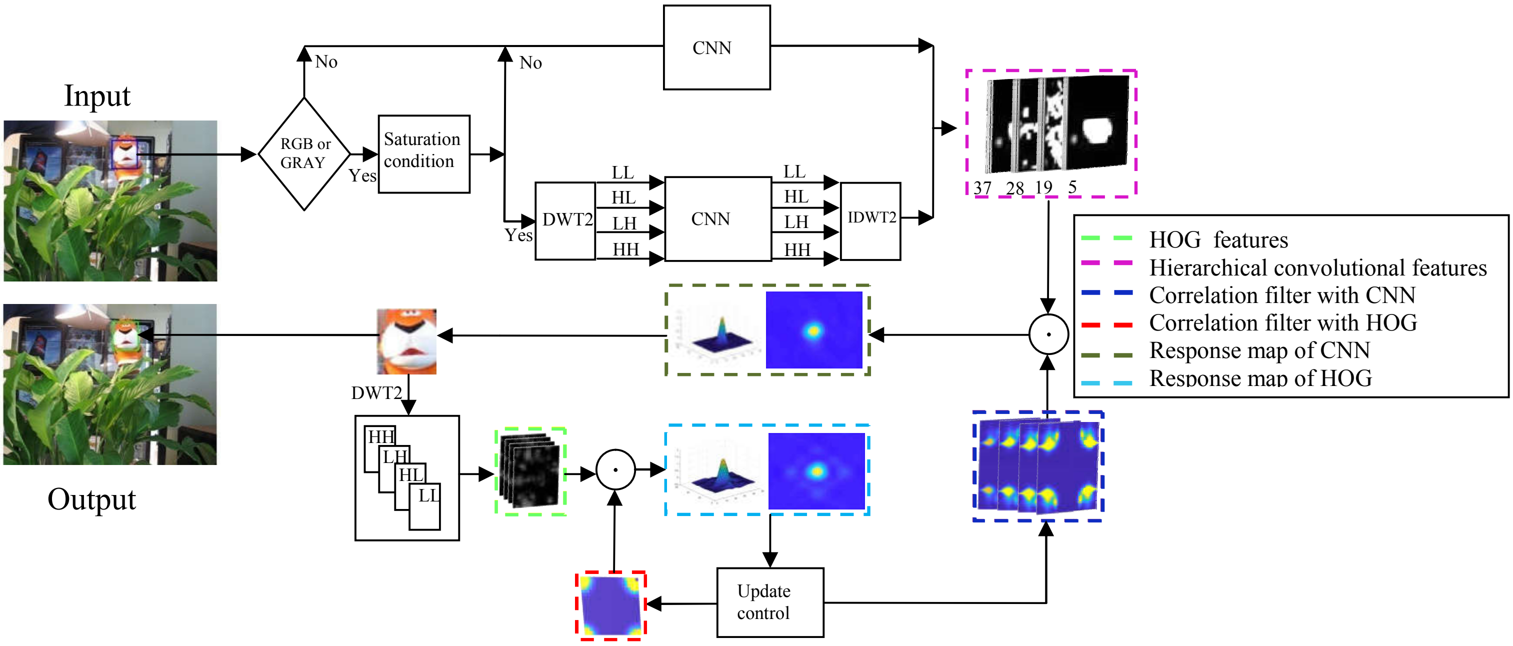


Figure 5. 1 : Main stages of the proposed algorithm.

5.2 Saturation condition

In the proposed approach the illumination variation has been handled in a reliable way based on a new proposed concept. The main idea of this proposed concept is based on integrating the wavelet decomposition obtained from the DWT [ILL,ILH,IHL,IHH] when the saturation of the image is very high instead of using the image components (RGB) directly in the network.

The saturation power state can be computed for each input frame according to the following three steps:

First step: the conversion of the red, green, and blue values of an RGB image to hue (H), saturation (S), and value (V) values of an HSV image.

Second step: the calculation of the saturation energy according to the following equations [8] :

$$E_S = 100 \times \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}(S_{ij})^2}{E_T} \tag{5.1}$$

$$E_T = \sum_{i=1}^{m}\sum_{j=1}^{n}(H_{ij})^2 + \sum_{i=1}^{m}\sum_{j=1}^{n}(S_{ij})^2 + \sum_{i=1}^{m}\sum_{j=1}^{n}(V_{ij})^2 \tag{5.2}$$

Where, E_S refers the saturation energy and E_T refers to the total energy of the input frame.

Third step: if $E_S > \frac{2}{3} \times 100\%$, then the illumination is very weak, In this case, the wavelet decomposition [LL,LH,HL,HH] is used in the input of CNN .In the reverse case , the decomposition RGB of the input image is used. This step is carried out following the proposed approach in this thesis.

The combined DWT and CNN method is found to be robust and therefore it is capable of alleviating the problem of illumination variation. As an example, Table 1 presents the percentage of energy saturation of the Singer2 sequence calculated along six frames following equation (5.1).

Table 5. 1 : The percentage of energy saturation in the Singer2 sequence.

Frame	**6**	8	10	**12**	14	16	**18**
Energy saturation (%)	**76.58**	76.30	76.82	**25.16**	71.62	73.39	**72.03**

It is clear from Table 1 that the energy saturation varies from a frame to another between the minimum value of 25.16% corresponding to frame 12 and the maximum value of 76.82% corresponding to frame 10.Based on the condition mentioned in the third step, it can be observed that only in the case of frame 12, energy saturation is low and the required condition is not satisfied, in this case the RGB approach is used. It is obvious that for the other frames that required condition is satisfied, hence the wavelet decomposition [LL,LH,HL,HH] is used in the input of CNNs in all these frames. It can be concluded that under the application of the proposed approach in this thesis the case of illumination variation can be handled more accurately based on the beneficial features of 2-DWT.Figure.5.2 illustrates the accurate placement of the target within the six chosen frames of the Singer2 sequence. The bleu frame is corresponding to the initial position of the tracked object and the red frame is corresponding to the tracking (used tracker) based on the proposed approach combining the wavelet and the CNNs. It is obviously noted that the proposed approach allows robust tracking of the moving object under illumination variations and in the same time it maintains the long-term memory of target appearance which ensures a high degree of accuracy in locating the target in the majority of the frames of the singer 2 sequence. On the other side, for the validation of the proposed approach based on the proposed saturation condition, two tests have been carried out based on the calculation of the error tracking in the both cases of using the saturation condition (in red) and without taking into account the saturation condition (in blue). It is clearly observed in Figure. 5.2 (in the middle) that the tracking error obtained under the proposed approach is minimized to a very low value compared to the standard case where the saturation condition variation is not taken into account.

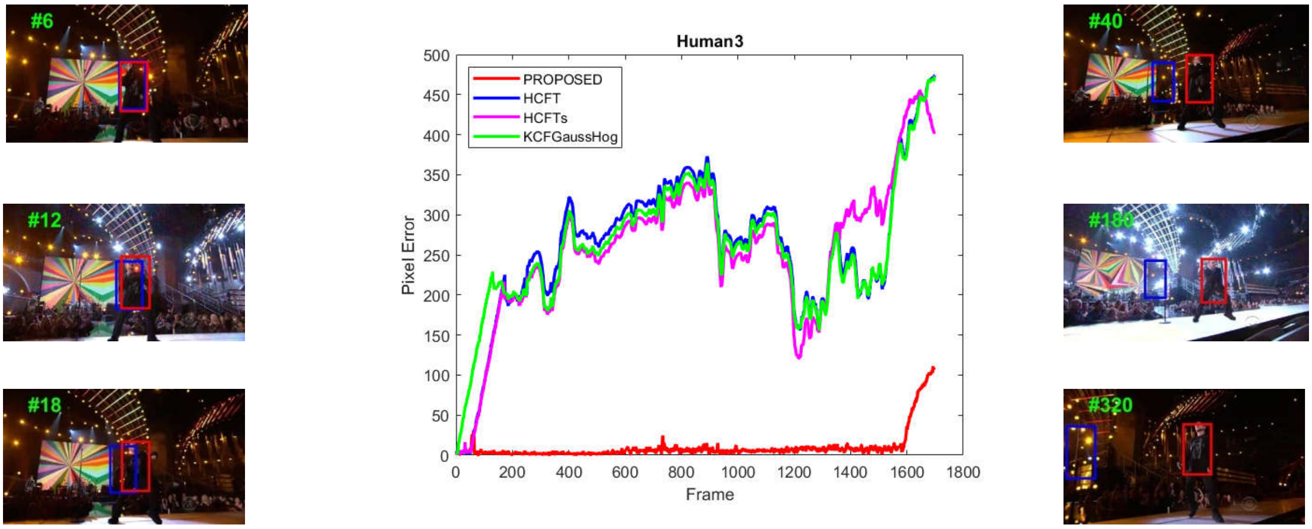


Figure 5. 2 : A frame-by-frame display of the results of the Singer2 sequence tracking, with and without the saturation condition (in pixel).

5.3 Convolution Features

Due to the very interesting properties of CNNs in ensuring accurate separation between the object and its background, it has been used largely to improve many dimensions of computer vision. We present translation estimation with creation of a translation model by form extraction using a CNN model.

VGGNet-19 feature extractor trained on the ImageNet dataset [56-57] is used to encode the target appearance, while characteristics propagate to deeper layers spatial resolution progressively decreases but Semantic discrimination between objects belonging to various categories is enhanced. The determination of the exact target item position in visual object tracking is more relevant than semantic category, bilinear interpolation [2] is used to resize each input frame to size 224×224.

Firstly, the fully connected layers are removed and the outputs of the convolution layers conv1-4, conv3-4, conv4-4, and conv5-4 are used as deep features. In addition, a cosine window to weight each feature channel is used to eliminate the boundary discontinuities[72, 143].

When the CNN depth increases the spatial resolution of a target object decreases progressively because of the pooling processes.

By using bilinear interpolation given in Equation. (5.3) each feature map is also rescaled to size $\frac{M}{4} \times \frac{N}{4}$, where M and N are the dimensions of the feature vector x, to correct the spatial resolution across the pooling layers.

$$x_i = \sum_k \alpha_{ik} h_k \qquad (5.3)$$

5.4 Correlation filters

Usually, a correlation tracker search for the maximum value on the response in a discriminative classifier to locates target objects [63, 64, 144–146]. In this research, each convolutional layer outputs are used as multi-channel features[45, 144] . we assume that $\boldsymbol{x}$ isthe $l-$th layer of feature vector of size $M \times N \times D$, with M, N, and D are the width, height, and the number of channels, respectively. we ignore the dependence of M, N, and D on the layer index l and note $\boldsymbol{x}^{(l)}$directly as $\boldsymbol{x}$.all the circular shifts of $\boldsymbol{x}$ along the M and N dimensions are taken as training samples. a Gaussian function label $y(m,n) = e^{-\frac{(m-M/_2)^2+(n-N/_2)^2}{2\sigma^2}}$,where σ is the kernel width , is attributed to Each shifted sample $\boldsymbol{x}_{(m,n)}(m,n) \in \{0,1,\dots,M-1\} \times \{0,1,\dots,N-1\}$. By solving the minimization problem (8), a correlation filter $\mathbf{W}$with the same size of $\boldsymbol{x}$ is trained.

$$\mathbf{W}^* = \mathrm{argmin}_{\mathbf{W}} \sum_{m,n} \left\| \mathbf{W} \cdot \boldsymbol{x}_{m,n} - y(m,n) \right\|^2 + \lambda \|\mathbf{W}\|^2 \tag{5.4}$$

λ is a regularization parameter ($\lambda \geq 0$).

linear kernel in a Hilbert space is used to induce the inner product in Equation .(5.4) e.g.,$\mathbf{W} \cdot \boldsymbol{x}_{m,n} = \sum_{d=1}^{D} \mathrm{W}_{\mathrm{m,n,d}}^{\mathrm{T}} \boldsymbol{x}_{m,n,d}$.As the label $y(m,n)$ is soft (not binary), so no hard-threshold sample is required.

the minimization problem in (5.4) could be solved in each individual feature channel using fast Fourier transformation (FFT) since it's similar to training the vector correlation filters in[146].capital letters denote Fourier transformed signals.

In the frequency domain; The learned filter on the $d-\mathrm{th}(d \in \{1,\dots,D\})$channel is given in (5.5)

$$\mathbf{W}^d = \frac{\mathbf{Y} \odot \bar{\mathbf{X}}^d}{\sum_{i=1}^{D} \mathbf{X}^i \odot \bar{\mathbf{X}}^i + \lambda} \tag{5.5}$$

where $\mathbf{Y}$ is the Fourier transformation form of $\mathbf{y} = \{y(m,n)|(m,n)\{0,1,\dots,M-1\} \times \{0,1,\dots,N-1\}\}$and the bar refer to the complex conjugation. The operator $\odot$ is the Hadamard product. For each image patch in the next frame the $l-\mathrm{th}$ layer feature vector is noted z with size $M \times N \times D$. The $l-\mathrm{th}$ correlation response map is given by

$$f_l = \boldsymbol{\mathcal{F}}^{-1}\left(\sum_{d=1}^{D} \mathbf{W}^d \odot \bar{\mathbf{Z}}^d\right) \tag{5.6}$$

The operator $\boldsymbol{\mathcal{F}}^{-1}$ refer to the inverse FFT transform. the position of maximum value of the correlation response map f_l of size $M \times N$ refer to the target location on the $l-\mathrm{th}$ convolution layer.

5.5 Estimation of Coarse-to-Fine Translation

For each set of correlation response maps $\{f_l\}$, we could deduce the target translation of each layer i.e. To search for the maximum value of the earlier layer$(l-1)-\mathrm{th}$, the location of maximum value in last layer $l-\mathrm{th}$ is taken as regularization. if $(\hat{m},\hat{n}) = \underset{m,n}{\mathrm{argmax}}\, f_l(m,n)$refer to the location of the maximum values on the $l-\mathrm{th}$ layer then the optimal location of target in the $(l-1)-\mathrm{th}$ layer is given by:

$$\underset{m,n}{\mathrm{argmax}}\, f_{l-1}(m,n) + \gamma f_l(m,n) \quad ; \quad |m-\hat{m}| + |n-\hat{n}| \leq r \tag{5.7}$$

Constraint imposed on m and n limit the searched area in the $(l-1)-$th correlation response map to the $r \times r$ neighboring regions of$(\hat{m}, \hat{n})$.

From the last to the inner layers each response value is weighted by a regularization term γ and propagated back tothe response maps of early layers[2, 3]. Finally, by maximizing the result of (5.7) on the layer with the best spatial resolution the target location is estimated.

On the other hand, using the equations (5.4),(5.5), and (5.7), we can calculate the maximum response of the correlation filter of HOG considering l =1, et γ =1.

5.6 Model Update

In this work, we update the correlations filters as proposed in [63]. Initially, we update the numerator $\mathrm{A^d}$ and denominator $\mathrm{B^d}$ of the correlation filter $\mathrm{W^d}$ in (5.5), separately using a moving average:

$$\mathrm{A_t^d} = (1-\eta)\mathrm{A_{t-1}^d} + \eta\, \mathrm{Y} \odot \overline{\mathrm{X}}_\mathrm{t}^\mathrm{d}; \tag{5.8}$$

$$\mathrm{B_t^d} = (1-\eta)\mathrm{B_{t-1}^d} + \eta\, \textstyle\sum_{\mathrm{i=1}}^{\mathrm{D}} \mathrm{X_t^i} \odot \overline{\mathrm{X}}_\mathrm{t}^\mathrm{i}; \tag{5.9}$$

$$\mathrm{W_t^d} = \frac{\mathrm{A_t^d}}{\mathrm{B_t^d}+\lambda} \tag{5.10}$$

We update the correlations filters extracted from CNN and HOG features conservatively because the conservatively learned filter is robust to noisy updates and succeed in estimating the confidence of every tracked result. To see if tracking failures occur, we establish a $\mathrm{T_0}$ threshold, if the maximum filter response of the correlation filter of HOG is greater than $\mathrm{T_0}$, this means that the tracked result z has a very high degree of confidence in that case we update correlations filters. On the other hand, when the confidence score is below $\mathrm{T_0}$, we don't update.

5.7 Conclusion

In this chapter, we proposed our method for visual object tracking. This method is based on image preprocessing, and feature extraction.

We solve the problem of illumination variation by using discrete wavelet transforms. And we have used correlation filters and convolutional features of CNN for getting the target location. On the other side, we have used the HOG feature of the image to ensure long-term memory of the target appearance so that the target item may be recovered if tracking fails. The coefficients of the discrete wavelet transform are employed instead of the image template to increase the performance of HOG.

Chapter 6 :

Experiment Results

6.1 Introduction

The applications of visual tracking are many and diverse, with each application having its own set of requirements. These can be real-time speed constraints, or performance requirements for a special class of videos. In practice, picking the best tracker is application specific and will imply some kind of trade-off.

However, to be able to properly and in a fair manner, test and compare generic visual object trackers, the community has developed datasets and evaluation metrics. Any publication within the field is expected to evaluate their method on these collective datasets. The method proposed in this thesis is evaluated on three such datasets: OTB-2015 [48] consisting of 100 videos, TempleColor [30] consisting of 128 videos, and UAV20 [27] consisting of 20 videos. The datasets are diverse and challenging, containing a large set of scenes and targets. Some experiments of this thesis are performed using OTB-2015 or TempleColor. Evaluation of the final tracker is done across all three datasets however.

6.2 Benchmark Datasets (OTB,TC128,UAV123)

This section introduces some datasets used in visual object tracking, 3 video tracking datasets including the OTB dataset, TC128, and UAV20 datasets. These datasets provide a large number of manually annotated video sequences. Moreover, evaluation standards and evaluation toolboxes are provided to facilitate the comparison between various tracking methods.

Among them, the OTB dataset first standardizes the standards of video tracking methods, and provides a comparison environment for tracking methods. It contains many types of tracking challenges and is widely used by tracking method.

6.2.1 OTB:

The OTB dataset (Object Tracking Benchmark, OTB) mainly includes three datasets, namely: OTB-2013 [30], OTB100 (OTB-2015) [31] and OTB50. The OTB dataset was first proposed by Y. Wu et al. [30] in CVPR in 2013 and named OTB- 013. OTB-2013 contains 51 video sequences and has more than 2900 frames with artificially labelled target boxes. The Skating video sequence can be regarded as two different video sequences because of the different labeling objects. The OTB-2013 dataset. Also divides visual object tracking into 11 types of challenges, such as: scale change, illumination change, occlusion, etc., and annotates the tracking challenges corresponding to each video sequence in the dataset to facilitate the analysis of tracking methods to deal with different challenges. Part of the video sequence in the OTB dataset is shown in Figure. 6.1.

It is worth noting that a video sequence may correspond to multiple tracking challenges. The OTB-2015 dataset is Y. Wu et al. [31] based on OTB-2013 to expand the video sequence, the number of video sequences expands to 100, so the OTB-2015 dataset is also called OTB100. With the wide application of OTB-2013 and OTB-2015 datasets, many tracking methods have achieved good tracking results on these two datasets. In order to increase the difficulty of the OTB tracking dataset, another 50 complex video sequences were extracted from the OTB-2015 dataset to form a new dataset called OTB50.

The OTB dataset also proposes an evaluation tool, which has good compatibility and is also suitable for some other datasets, such as: TC128, UAV123, etc. Therefore, the OTB dataset is currently the most widely used dataset in visual object tracking.

Figure 6. 1 : samples of some sequences in the OTB benchmark

6.2.2 TC128:

The TC128 dataset (Temple Color 128, TC128) was proposed by Liang et al. [39] of Temple University in the United States in 2015 in the IEEE Transactions on Image Processing journal. It contains 128 video sequences with manual annotations. The source of these 128 video sequences is mainly divided into two parts, one is the 50 video sequences commonly used in other video datasets, and the other is the 78 manually labelled video sequences. The video tracking dataset is mainly to explore the influence of color information on the video target tracking algorithm, so the video sequences in the video dataset are all color pictures. Part of the video sequence in the TC128 dataset is shown in Figure. 6.2.

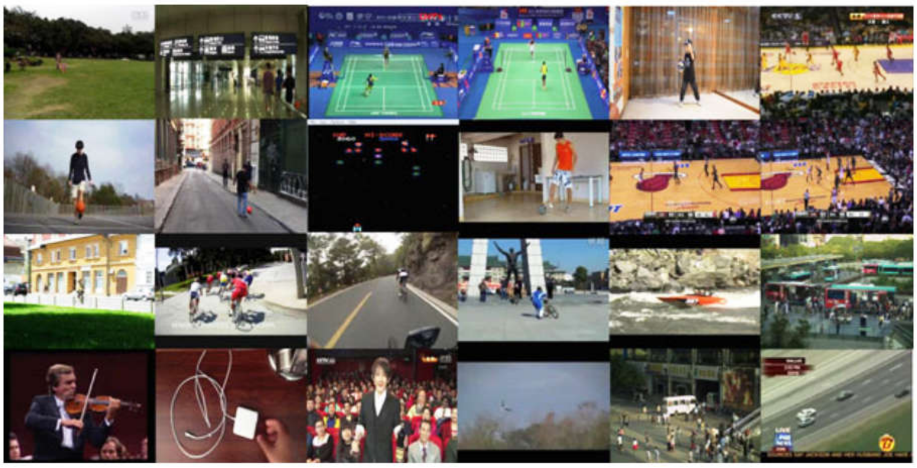


Figure 6. 2 : samples of some sequences in the TC128 benchmark

6.2.3 UAV123:

The UAV123 dataset (Unmanned Aerial Vehicles, UAV) was proposed by Mueller et al. [40] of King Abdullah University of Science and Technology at the 2016 ECCV conference. The video sequences in this dataset are all taken from overhead and most of these videos were shot and produced by unmanned aerial vehicles. There are also some video sequences that are synthesized by computer. The UAV123 dataset has more specific tasks and application scenarios, which includes 123 video sequences with manual annotations. In addition, the dataset also provides 20 ultra-long video sequences to test the tracking ability of the visual object tracking methods in long period videos and is named UAV20L. Part of the video sequence of the UAV123 dataset is shown in Figure. 6.3.

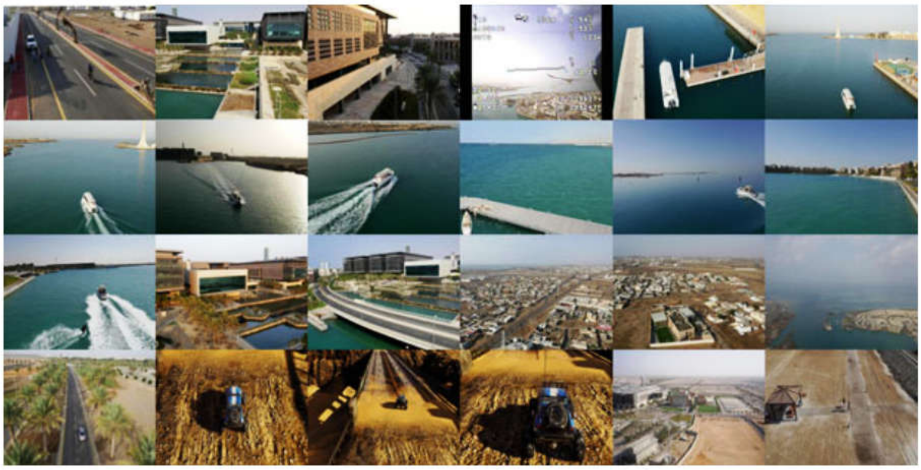


Figure 6. 3 : samples of some sequences in the UAV123 benchmark

6.3 Attributes

To be able to analyze a tracker it is necessary to understand why a tracker may fail. The datasets are rich with instances where the target to be tracked undergoes large changes in appearance or when the scene contains difficult background. This section briefly describes several types of such tracking challenges. In the literature, these are often referred to as *attributes*.

Background Clutter: In many cases the features of a target may look much like the background. Consider tracking a face when there is a crowd in the background. Small changes in target appearance may result in a part of the background looking more similar to the target than the target itself. If the tracker sees and is trained on a face for several frames a simple out-of-plane rotation of the target may yield a situation where a face in the crowd looks more like the target than the target itself. Any situation where the background contains similar features to that of the target can result in a tracker failure.

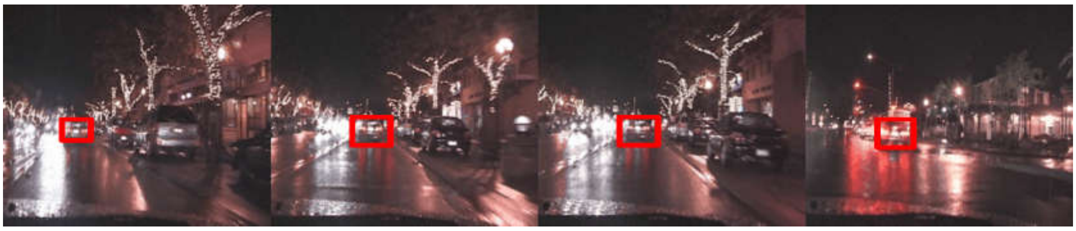


Figure 6. 4 : example of Background Clutter

Deformations: Non-rigid deformations forms a large set of transformations which a target can undergo. One of the most common examples is a target consisting of fairly rigid parts which move and rotate with respect to each other, such as a human. Humans are deformable but under most circumstances they are piecewise rigid.

Another example of piecewise rigidity are flying birds where each wing, and the center part, forms three fairly rigid parts. An example where the target contains little to no rigidity is a swimming octopus.

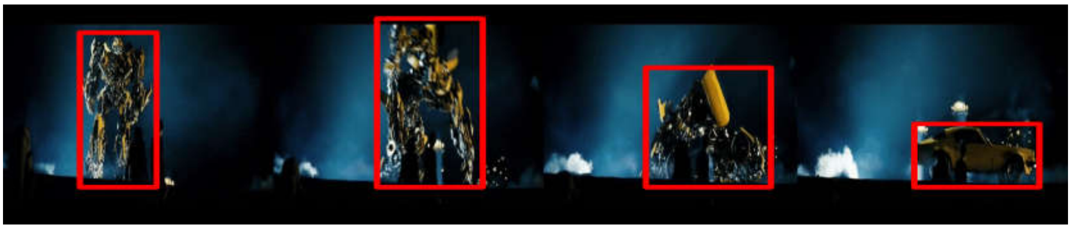


Figure 6. 5 : example of Deformations

Fast Motion: Several videos depict targets which moves very quickly with respect to their size. An oftentimes utilized prior in the visual tracking problem is that a target rarely moves very far between two subsequent frames. This prior information can stop atracker from switching between two similar targets, which can otherwise be the case for instance in a video showing competing 100m runners. However, utilizing this information in videos showing fast motion, such as several scenes of "The Matrix", may cause a tracker to lose its target.

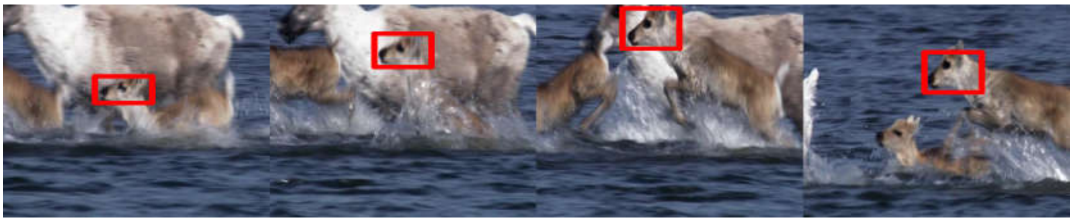


Figure 6. 6 : example of Fast Motion

Illumination Variations: The tracked target may be subject to spatial and temporal illumination variations. The scene may contain different lighting conditions in different places and the target may suffer from events such as lightning flashes or moving lights. The impact of such effects can be attenuated by utilizing features invariant to changes inillumination, such as HOG [9].

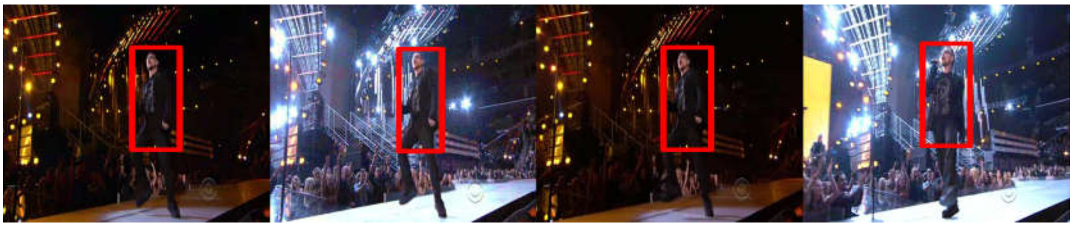


Figure 6. 7 : example of Illumination Variations

In-plane Rotations: In-plane rotations are rotations occurring in the 2D image plane.An example of an in-plane rotation is a motorcyclist performing a back flip, seen from the side. There is prior information available for targets undergoing such transformations, namely that the appearance remains the same, just rotated. This is however rarely utilized, and in-plane rotations proves challenging for many trackers. The DCF-based trackers contain an assumption that targets do not rotate, and other trackers such as LOT [39] view all changes in appearance as noise.

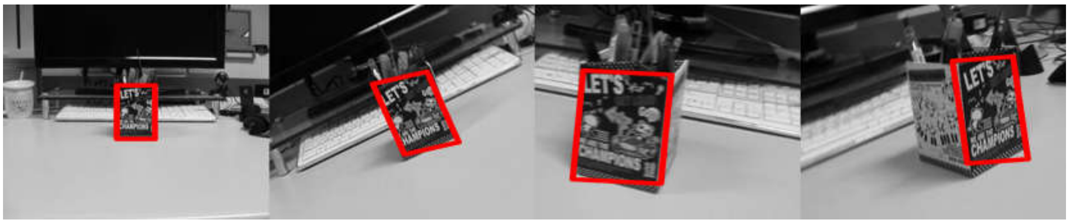


Figure 6. 8 : example of In-plane Rotations

Low Resolution: Another challenging attribute is low resolution. Low resolution reduces the available information of the target which may reduce the ability to discriminate between two targets. It is however common, both due to cheap cameras and due to large distances to targets.

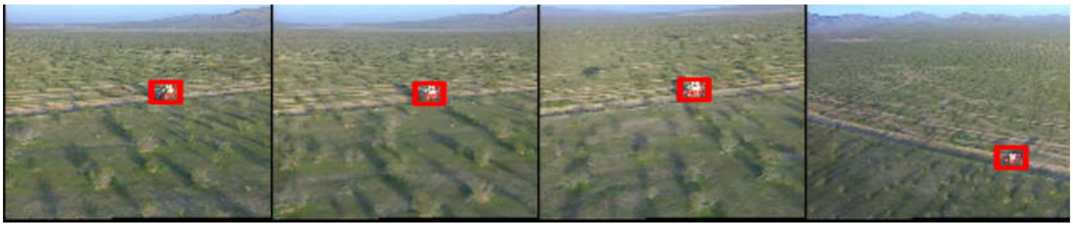


Figure 6. 9 : example of Low Resolution

Motion Blur: Motion blur occurs when a target moves quickly, drastically changing the target appearance into a smudged mess.

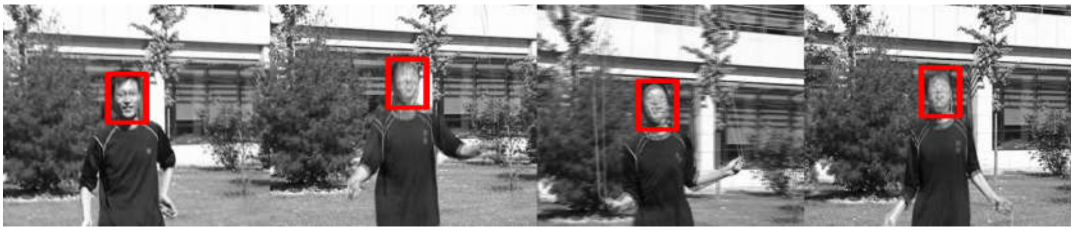


Figure 6. 10 : example of Motion Blur

Occlusions: Targets are often covered by something else. This can be seen in sequences showing a person walking behind a car, or a tiger moving behind some vegetation. This attribute is referred to as occlusions and they may be partial in the sense that only a part of the target is occluded, or full.

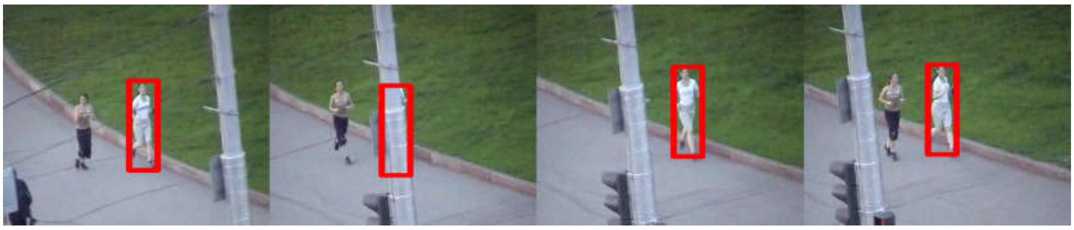


Figure 6. 11 : example of Occlusions

Out-of-Plane Rotations: Contrary to in-plane rotations, out-of-plane rotations are not in the image plane, that is, the rotation vector is not perpendicular to the image plane. This results in some parts of the target disappearing, and others appearing.

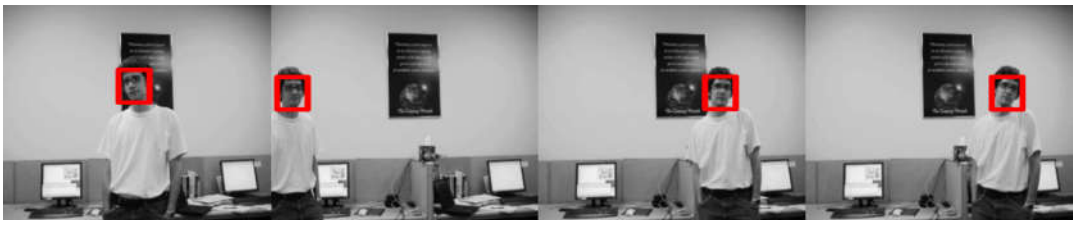


Figure 6. 12 : example of Out-of-Plane Rotations

Out-of-View: Videos may show a target moving beyond the image border, disappearing for a set of frames, displaying the out-of-view attribute. Many trackers become unstable when the target disappears and move around the video looking for the target, but are unable to locate the target when it reappears.

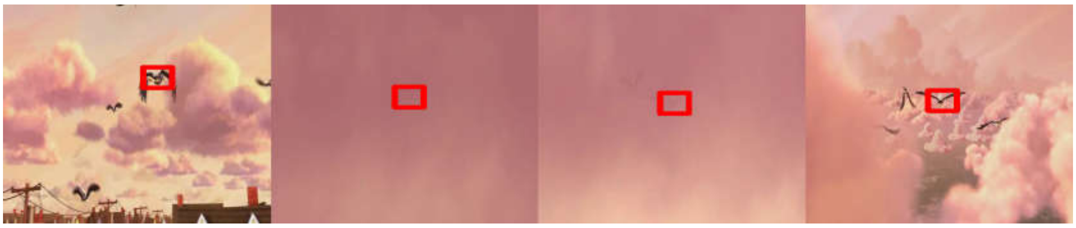


Figure 6. 13 : example of Out-of-View

Scale Variations: Often the target moves closer or further away from the camera, changing in scale. This problem is commonly remedied by an attempt to estimate the scale and then rescaling the input image.

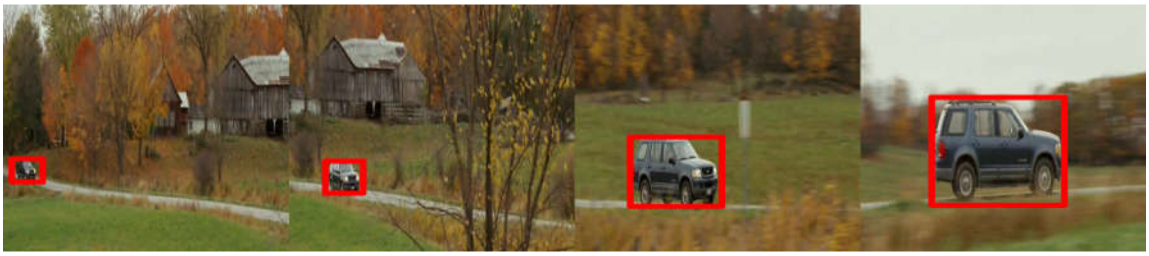


Figure 6. 14 : example of Scale Variations

6.4 Evaluation Methodology

Tracking proposed method are evaluated on the OTB,TC-128,and UAV20L datasets using one-pass evaluation (OPE) protocol with either distance precision or overlap success rates .

Robustness Evaluation: The conventional way to evaluate trackers is to run them throughout a test sequence with initialization from the ground truth position in the first frame and report the average precision or success rate. We refer this as one-pass evaluation (OPE).

Precision Plot: One widely used evaluation metric on tracking precision is the center location error, which is defined as the average Euclidean distance between the center locations of the tracked targets and the manually labeled ground truths. Then the average center location error over all the frames of one sequence is used to summarize the overall performance for that sequence. However, when the tracker loses the target, the output location can be random and the average error value may not measure the tracking performance correctly [6]. Recently the precision plot [6, 27] has been adopted to measure the overall tracking performance.

It shows the percentage of frames whose estimated location is within the given threshold distance of the ground truth.

As the representative precision score for each tracker we use the score for the threshold = 20 pixels [6].

Success Plot: Another evaluation metric is the bounding box overlap. Given the tracked bounding box r_t and the ground truth bounding box r_a, the overlap score is defined as $S = \frac{|r_t \cap r_a|}{|r_t \cup r_a|}$, where ∩ and ∪ represent the intersection and union of two regions, respectively, and $|\cdot|$ denotes the number of pixels in the region. To measure the performance on a sequence of frames, we count the number of successful frames whose overlap S is larger than the given threshold t_0. The success plot shows the ratios of successful frames at the thresholds varied from 0 to 1. Using one success rate value at a specific threshold (e.g. $t_0 = 0.5$) for tracker evaluation may not be fair or representative. Instead we use the area under curve (AUC) of each success plot to rank the tracking algorithms.

6.5 Results and discussion

The proposed algorithm has been validated and evaluated on two benchmark datasets such as OTB50 [145]which includes 50 videos and OTB100 [145]which includes 100 videos. The tracking algorithm has been implemented in MATLAB on an Intel I7-8750H 2.20 GHz CPU with 16 GB RAM and the MatConvNet toolbox[147], while the feature extraction using CNN forward propagation has been carried out on a GeForce GTX1060 GPU.

The convolutional neural network based VGG-Net-19 proposed in 2012 by Visual Geometry Group and which is consisted of 19 layers (16 convolution layers, 3 Fully connected layer, 5 MaxPool layers and 1 SoftMax layer) [56], has been trained on the free large-scale hierarchical image database ImageNet[142] and adopted for feature extraction.

The features are used only from the outputs of pool 1, pool 3, pool 4, and pool 5. We fix the size of the search window to 1.8 times the target size. The regularization parameter of (2) is set to λ = 10-4. and kernel width of taken as 0.1 for the generation of the Gaussian function labels, the learning rate η in (6) is set to 0.01 and for control updating T_0 is taken equal to 0.3.The value of γ is set as 1, 0.5,0.25 and 0.15 for the conv5-4, conv4-4, conv3-4, conv1-4 layers, respectively.

6.5.1 Results of OTB50 and OTB100

The performance of the proposed trackers has been evaluated based on two performance metrics such as the area-under-the-curve (AUC) and the distance precision (DP). On the other side, for the validation of the proposed tracker performance, a comparison has been carried out based on a 25 trackers, which were presented in several previous works such as ASLA[148], CSK[61], DSST[62], MEEM[92], MUSTER[149], SAMF[71], SRDCF[72], Struck[34], siamfc3s[65], HCFTs[3], HDT[7], Staple[75], CNN-SVM[63], CF2[2], LCT[74], KCF[45], TLD[88], KCF_GaussHog [45],KCF_LinearHog [45], BACF[73], DeepSRDCF[79], DRVT[150], MemDTC[151],MemTrack[151], SRDCFdecon[152].

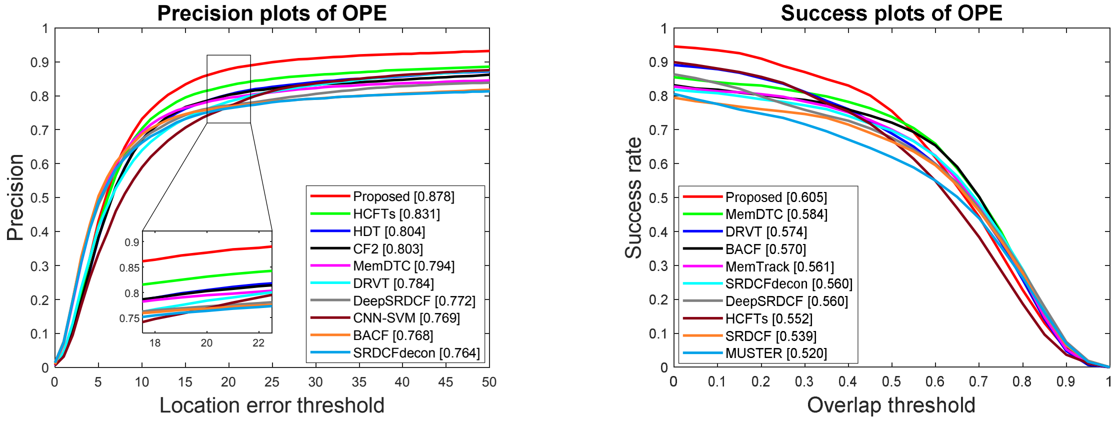


Figure 6. 15: One-Pass-Evaluation (OPE) curves on the OTB50 dataset. Left: **Overlap precision (OP)**, and right**: center localization error (CLE).**

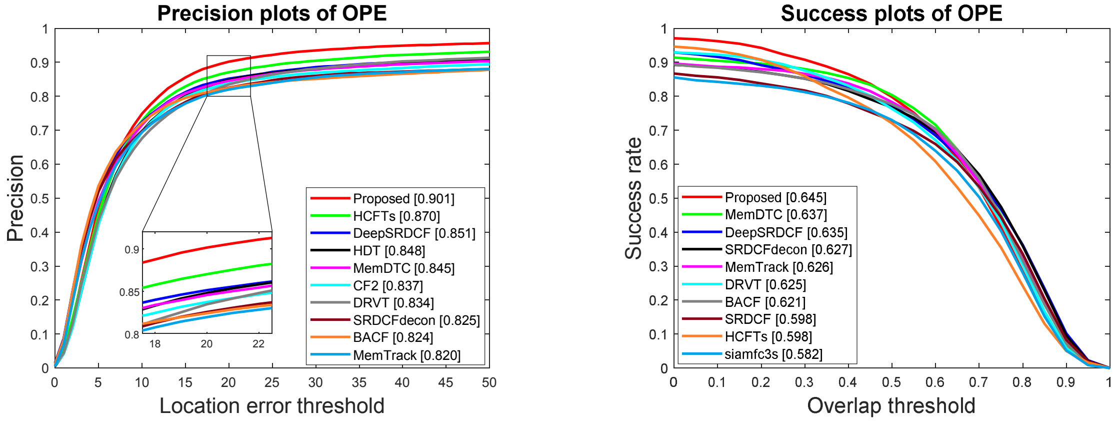
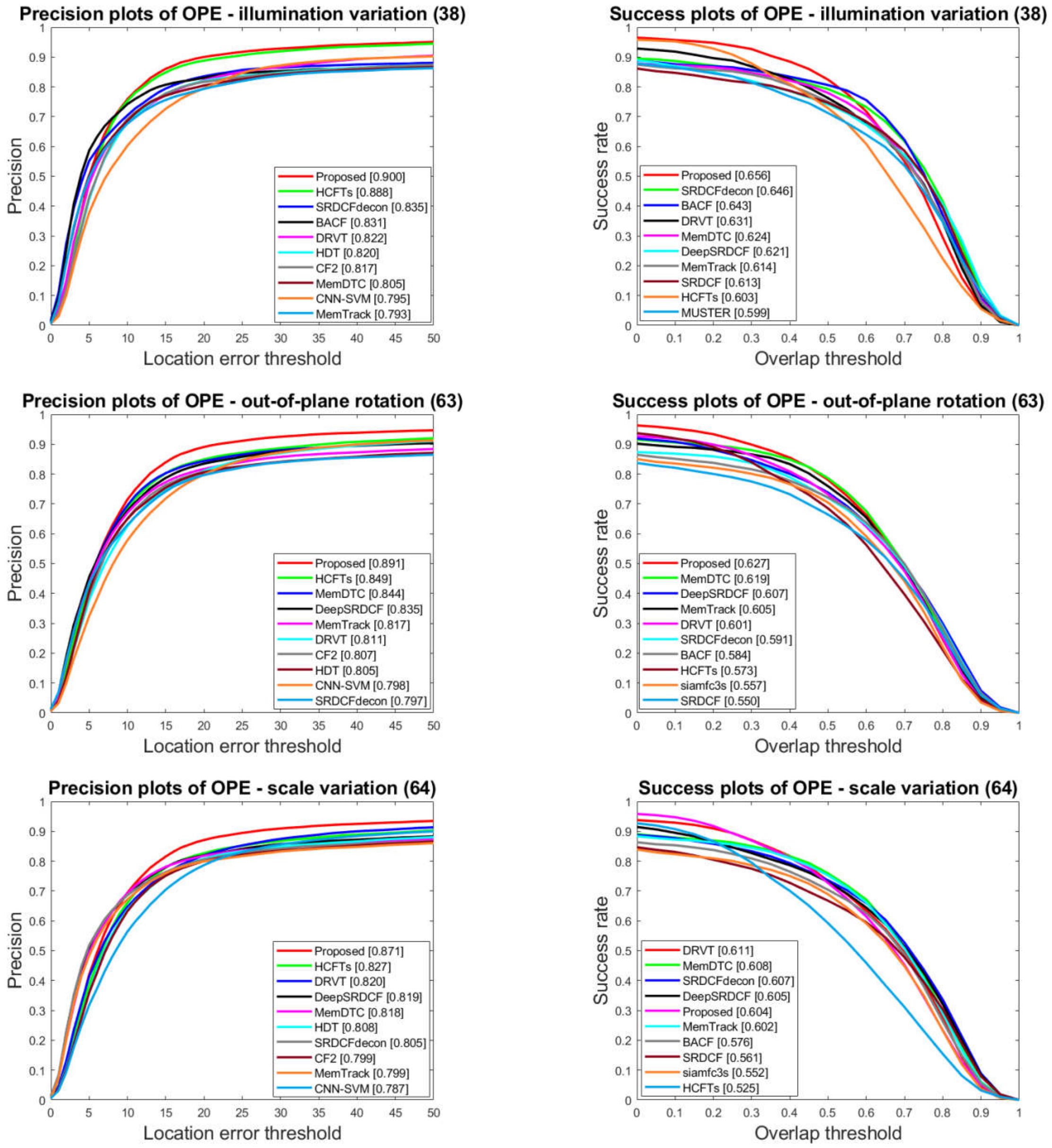
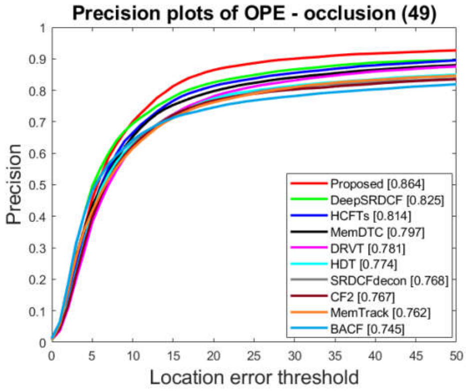
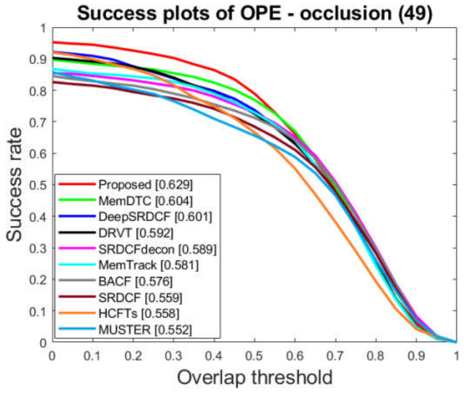


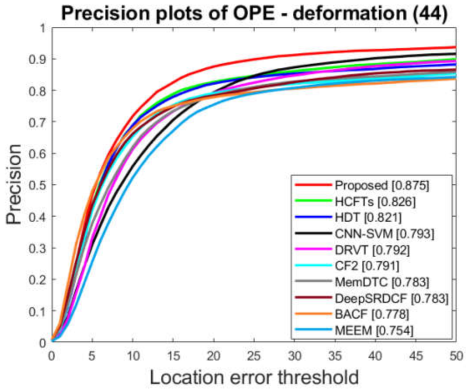
Figure 6. 16: One-Pass-Evaluation (OPE) curves on the OTB100 dataset. Left: **Overlap precision (OP)**, and right**: center localization error (CLE).**

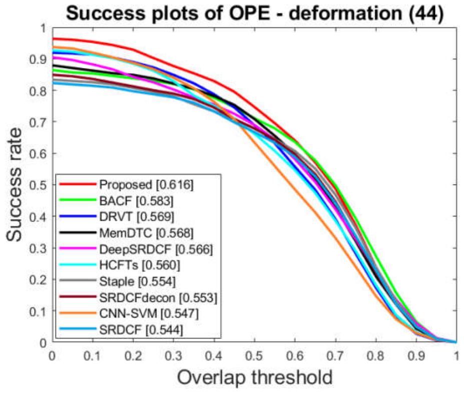
Indeed, for the clarity of results presentation, only the top 10 ranked trackers are taken into account as shown in Figures 5 and 6. Based on the results of all the trackers, it is clear that the proposed tracker performs favorably with an AUC of (64.5%, 60.5%) and a DP of (90.1%, 87.8%) on OTB-100 and OTB-50, respectively. These obtained results prove clearly that the proposed tracker is improved with a gain of 3.1% in the average distance precision compared to the second best tracker among other trackers HCFTs, and 0.8% in the average overlap precision compared to the second best tracker among other trackers MemDTC in the case of OTB-100. For the OTB-50, the proposed tracker has an improved gain of 4.7% and 2.1% for the average distance precision and average overlap precision respectively compared to the second best tracker which is the same as the case of OTB-100.

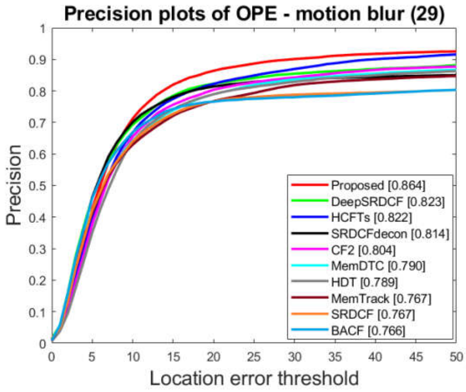


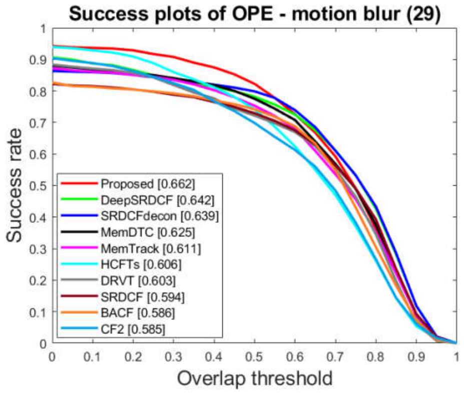

Precision plots of OPE - occlusion (49)
Precision
Location error threshold
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
5
10
15
20
25
30
35
40
45
50
Proposed [0.864]
DeepSRDCF [0.825]
HCFTs [0.814]
MemDTC [0.797]
DRVT [0.781]
HDT [0.774]
SRDCFdecon [0.768]
CF2 [0.767]
MemTrack [0.762]
BACF [0.745]

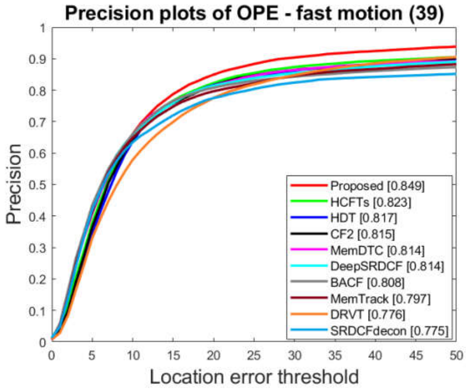

Success plots of OPE - occlusion (49)
Success rate
Overlap threshold
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Proposed [0.629]
MemDTC [0.604]
DeepSRDCF [0.601]
DRVT [0.592]
SRDCFdecon [0.589]
MemTrack [0.581]
BACF [0.576]
SRDCF [0.559]
HCFTs [0.558]
MUSTER [0.552]

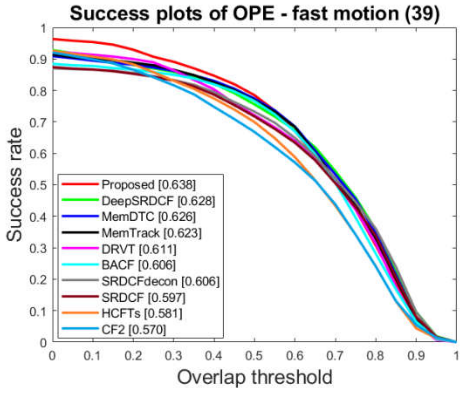

Precision plots of OPE - deformation (44)
Precision
Location error threshold
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
5
10
15
20
25
30
35
40
45
50
Proposed [0.875]
HCFTs [0.826]
HDT [0.821]
CNN-SVM [0.793]
DRVT [0.792]
CF2 [0.791]
MemDTC [0.783]
DeepSRDCF [0.783]
BACF [0.778]
MEEM [0.754]


Success plots of OPE - deformation (44)
Success rate
Overlap threshold
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Proposed [0.616]
BACF [0.583]
DRVT [0.569]
MemDTC [0.568]
DeepSRDCF [0.566]
HCFTs [0.560]
Staple [0.554]
SRDCFdecon [0.553]
CNN-SVM [0.547]
SRDCF [0.544]


Precision plots of OPE - motion blur (29)
Precision
Location error threshold
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
5
10
15
20
25
30
35
40
45
50
Proposed [0.864]
DeepSRDCF [0.823]
HCFTs [0.822]
SRDCFdecon [0.814]
CF2 [0.804]
MemDTC [0.790]
HDT [0.789]
MemTrack [0.767]
SRDCF [0.767]
BACF [0.766]


Success plots of OPE - motion blur (29)
Success rate
Overlap threshold
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Proposed [0.662]
DeepSRDCF [0.642]
SRDCFdecon [0.639]
MemDTC [0.625]
MemTrack [0.611]
HCFTs [0.606]
DRVT [0.603]
SRDCF [0.594]
BACF [0.586]
CF2 [0.585]


Precision plots of OPE - fast motion (39)
Precision
Location error threshold
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
0
5
10
15
20
25
30
35
40
45
50
Proposed [0.849]
HCFTs [0.823]
HDT [0.817]
CF2 [0.815]
MemDTC [0.814]
DeepSRDCF [0.814]
BACF [0.808]
MemTrack [0.797]
DRVT [0.776]
SRDCFdecon [0.775]


Success plots of OPE - fast motion (39)
Success rate
Overlap threshold
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Proposed [0.638]
DeepSRDCF [0.628]
MemDTC [0.626]
MemTrack [0.623]
DRVT [0.611]
BACF [0.606]
SRDCFdecon [0.606]
SRDCF [0.597]
HCFTs [0.581]
CF2 [0.570]

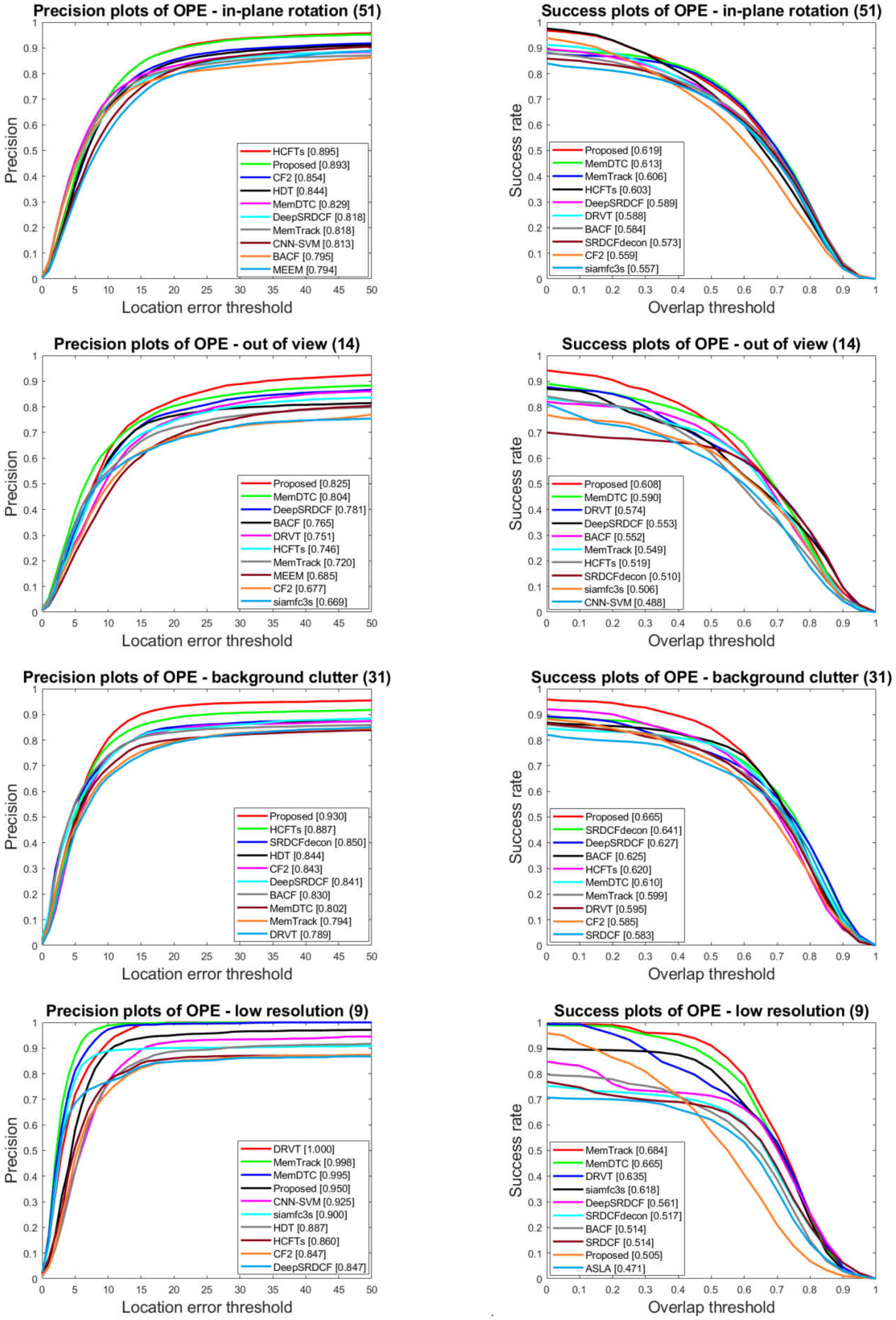


Figure 6. 17 : Overlap success plots and distance precision plots in 11 tracking challenge situations.

Figure 6.17 illustrates the overlap success rate plots and the distance precision plots obtained on OTB100 dataset for 11 challenging instances such as the scale variation, fast motion, in-plane rotation, deformation, motions blur, occlusion, illumination variation, out-of-plane rotation, background clutter, out-of-view, and low resolution. It can be clearly notes within all the sub-figures of Figure.6.17 that the proposed tracker outperforms its aforementioned state-of-the-art counterparts, except low resolution.

Table 6. 1 : **The distance precision results achieved by the proposed tracker and other 10 trackers on 11 different attributes on the OTB-2015 benchmark. The best, second best, and third best values are highlighted in red, green, and blue, respectively**

	Proposed	MemDTC	DeepSRDCF	SRDCFdecon	MemTrack	DRVT	BACF	SRDCF	HCFTs	CF2	siamfc3s
ALL	**0.901**	0.845	**0.851**	0.825	0.820	0.834	0.824	0.789	**0.870**	0.837	0.771
IV	**0.900**	0.805	0.791	**0.835**	0.793	0.822	0.831	0.792	**0.888**	0.817	0.736
SV	**0.871**	0.818	0.819	0.805	0.799	**0.820**	0.774	0.745	**0.827**	0.799	0.735
OCC	**0.864**	0.797	**0.825**	0.768	0.762	0.781	0.745	0.735	**0.814**	0.767	0.722
DEF	**0.875**	0.783	0.783	0.753	0.718	**0.792**	0.778	0.734	**0.826**	0.791	0.690
MB	**0.864**	0.790	**0.823**	0.814	0.767	0.742	0.766	0.767	**0.822**	0.804	0.705
FM	**0.849**	0.814	0.814	0.775	0.797	0.776	0.808	0.769	**0.823**	**0.815**	0.743
IPR	**0.893**	0.829	0.818	0.776	0.818	0.792	0.795	0.745	**0.895**	**0.854**	0.742
OPR	**0.891**	**0.844**	0.835	0.797	0.817	0.811	0.787	0.742	**0.849**	0.807	0.756
OV	**0.825**	**0.804**	**0.781**	0.641	0.720	0.751	0.765	0.597	0.746	0.677	0.669
BC	**0.930**	0.802	0.841	**0.850**	0.794	0.789	0.830	0.775	**0.887**	0.843	0.775
LR	0.950	**0.995**	0.847	0.747	**0.998**	**1.000**	0.795	0.765	0.860	0.847	0.900

Table 6. 2 : **The overlap success rate results achieved by the proposed tracker and other 10 trackers on 11 different attributes on the OTB-2015 benchmark. The best, second best, and third best values are highlighted in red, green, and blue, respectively**

	Proposed	MemDTC	DeepSRDCF	SRDCFdecon	MemTrack	DRVT	BACF	SRDCF	HCFTs	CF2	siamfc3s
ALL	**0.645**	**0.637**	**0.635**	0.627	0.626	0.625	0.621	0.598	0.598	0.562	0.582
IV	**0.656**	0.624	0.621	**0.646**	0.614	0.631	**0.634**	0.613	0.603	0.540	0.568
SV	0.604	**0.608**	0.605	**0.607**	0.602	**0.611**	0.576	0.561	0.525	0.485	0.552
OCC	**0.629**	**0.604**	**0.601**	0.589	0.581	0.592	0.576	0.559	0.558	0.525	0.543
DEF	**0.616**	0.568	0.566	0.553	0.539	**0.569**	**0.583**	0.544	0.560	0.530	0.506
MB	**0.662**	0.625	**0.642**	**0.639**	0.611	0.603	0.586	0.594	0.606	0.585	0.550
FM	**0.638**	**0.626**	**0.628**	0.606	0.623	0.611	0.606	0.597	0.581	0.570	0.568
IPR	**0.619**	**0.613**	0.589	0.573	**0.606**	0.588	0.584	0.544	0.603	0.559	0.557
OPR	**0.627**	**0.619**	**0.607**	0.591	0.605	0.601	0.584	0.550	0.573	0.534	0.557
OV	**0.608**	**0.590**	0.553	0.510	0.549	**0.574**	0.552	0.460	0.519	0.474	0.506
BC	**0.665**	0.610	**0.627**	**0.641**	0.599	0.595	0.625	0.583	0.620	0.585	0.523
LR	0.505	**0.665**	0.561	0.517	**0.684**	**0.635**	0.514	0.514	0.435	0.388	0.618

We use OTB2015 [15] to further evaluate the Proposed tracker. The distance precision (DP) and The overlap success rate result under 11 different attributes achieved by the proposed tracker and other 10 trackers are listed in Tables II and III, respectively. It can be seen that our tracker performs best 9 out of 11 attributes in terms of DP, including the scale variation, fast motion, deformation, motions blur, occlusion, illumination variation, out-of-plane rotation, background clutter, out-of-view. Furthermore, it also achieves the best overlap success rate performance under nine attributes. On the other hand, the Proposed tracker does not perform well in scenes with low resolution. This is because the features are difficult to capture enough information in low-resolution scenes. Overall, that the proposed tracker outperforms its aforementioned state-of-the-art counterparts.

Table 6. 3 : **Comparison between the proposed trackers and the state -of-the-art trackers based on the tests carried out on OTB2015 benchmark**

METHOD	Proposed	DFC	DeepNCC	BACF_M	RSCF	TripFC	MST	FSNet	TMCF	adaDDCF	CLIP	DAMA
Reference	[153]	[4]	[154]	[155]	[156]	[157]	[158]	[159]	[160]	[161]	[90]	[162]
Year	**2022**	**2020**	**2020**	**2021**	**2021**	**2021**	**2020**	**2021**	**2020**	**2020**	**2020**	**2020**
Precision rate	**0.901**	**0.816**	**0.847**	**0.838**	**0.838**	**0.808**	**0.822**	**0.829**	**0.795**	**0.872**	**0.845**	**0.834**
Success rate	**0.645**	**0.624**	**0.620**	**0.640**	**0.617**	**0.597**	**0.625**	**0.596**	**0.593**	**0.612**	**0.628**	**0.623**

The obtained results of the precision rate and the success rate on the OTB2015 dataset benchmark of the proposed tracker and the other aforementioned trackers are presented in Table 3. Based on these results, it can be concluded that the proposed tracker has ensured superior values compared to the other trackers with values of 90.1% and 64.5% for the two performance metrics such as the distance precision (DP) score and the AUC score, respectively.

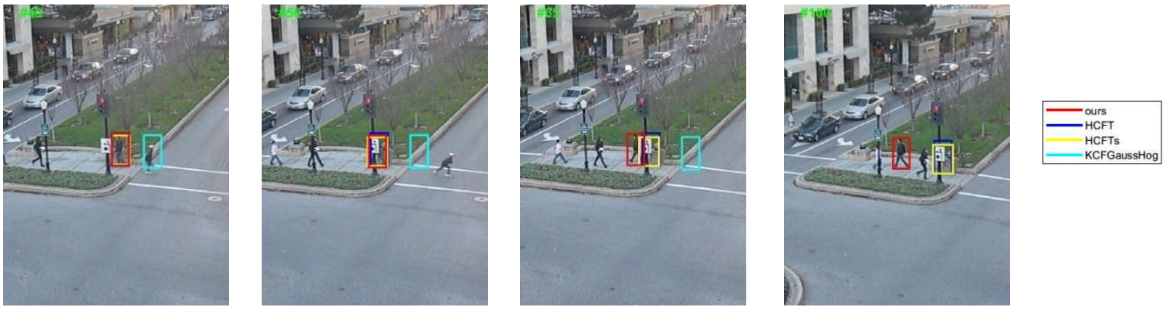


Figure 6. 18 : Comparisons of our proposed approach with HCFT,HCFTs,and KCF_GaussHog trackers in the challenging Human3 sequence.

To evaluate the proposed tracker face to occlusion problem, the proposed tracker has been implemented on the Human3 sequence and compared with three count parts trackers such as HCF [2],HCFTs [3],and KCF_GaussHog [45], which have been implemented on the same benchmark of Human 3.

It is obvious that during the tracking process in real time, each new detection generates a new picture patch that may be utilized to update the model tracking as explain in [45], although the model tracking update mechanism which has been used in HCFT[2],HCFTs[3],and KCF_GaussHog[45] based on Human3, have been insufficient in the case of occlusion. Therefore, mixes of approaches have been proposed in this thesis to develop a system that can solve the occlusion problem. Indeed, four sequences of tracking resulting from the implementation of the proposed tracker based on Human3 are shown in Figure. 8 starting from the right to the left. It can be noted that the physical visual obstacle which has occluded the tracked object has not affected the tracking process, which means that the proposed tracker can effectively and more accurately deal with the physical structures and background structures occluded along the tracking trajectory without losing the target object.

Furthermore, a frame-by-frame curve based on the obtained tracking results using the Human3 sequence tracking for the proposed tracker and the other three counterparts' trackers as shown in Figure.6.19. It can be clearly observed that the proposed trackers (red curve) deal effectively with the occlusion issue in comparison to the other trackers who have shown nearly similar dynamics. It is worthy here to note that the difference is important which means the net superiority of the proposed tracker in dealing with such a problem.

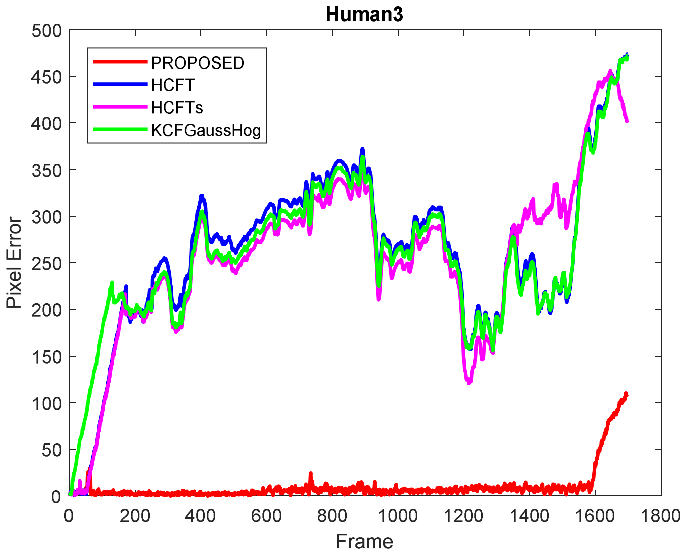


Figure 6. 19 : A frame-by-frame display of the results of the Human3 sequence tracking (in pixel).

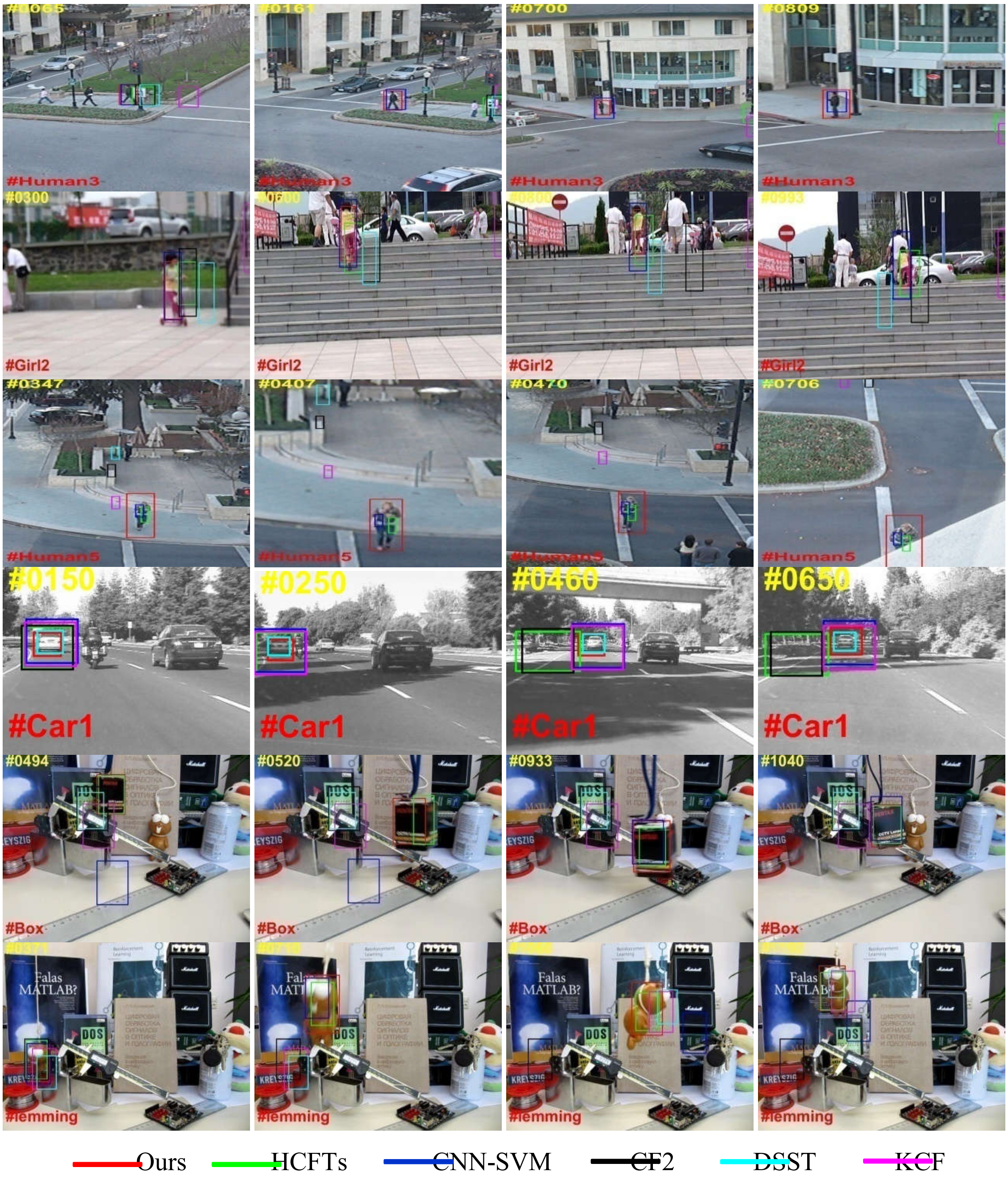


Figure 6. 20 : Qualitative results of our proposed method, along with HCFTs,CNN-SVM,CF2,DSST,KCF, on six challenge sequences.

Figure 6.20 shows the tracking performance of several representative trackers such as HCFTs[3] ,CNN-SVM[63] , CF2[2], DSST[62] ,KCF[45] ,and the proposed tracker on six challenging sequences. From top to down, the sequences are Human3, Girl2, Human5, Box, Car1, and Lemming, respectively. Human3 has the challenge of scale variation, occlusion, deformation, background clutters, and out-of-plane rotation. Girl2 contains scale variation, occlusion, deformation, out-of-plane rotation, and motion blur. Human5 includes scale variation, occlusion and deformation. Car1 compromises illumination variation, motion blur , scale variation, fast motion and background clutters. Box holds illumination variation, motion blur, occlusion, out-of-plane rotation, in-plane rotation, background clutters, scale variation, and out-of-view. Lemming gathers illumination variation, fast motion, occlusion, scale variation, out-of-view, and out-of-plane rotation. It is obvious that the proposed tracker handles all these complicated scenarios favorably for the aforementioned sequences compared to the other trackers.

In the same time, it can be observed clearly that the other trackers are unable to handle some of them. CNN-SVM with deep features performs well when scale variation, out-of-plane rotation, and occlusion are present. (Human3, and Girl2) but it is less effective in handling with drastic variations (Human5,Box,Lemming). CF, DSST and KCF are less effective in dealing with occlusion and deformation(Human3,Girl2, Human5, Box, and Lemming).HCFTs performs well in presence of scale variation, occlusion, motion blur and background clutter (Box, Lemming) but it is less effective in dealing with deformation (Human3,Girl2 and Human5). Therefore the proposed tracker operates adaptively and robustly when confronted with a variety of challenging factors.

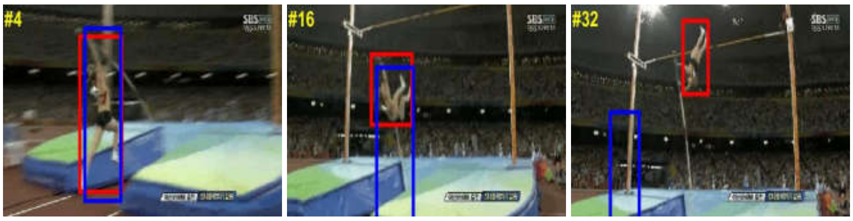


Figure 6. 21 : The proposed tracker failed on the sequence Jump from OTB-100. The red and blue bounding boxes indicate the ground truths and the proposed tracker results, respectively.

It is worthy to clarify a minor drawback which has been faced while the application of the proposed tracker, Indeed, it has been remarked that the significant change in the aspect ratio in the case of Jump sequence has caused the target missing, which implies that a more robust design is required for scale variation, and aspect ratio adjustment strategy for the proposed tracker to overcome completely these kinds of deficiencies.

To check the effectiveness of the proposed tracker, it has been implemented based on two different proposed methods such as the HOG with wavelet (HOG-DWT) and HOG without wavelet, which have been combined with hierarchical convolutional features (HCF) with and without wavelet, then it has been evaluated using OTB-100.Indeed, the obtained results of the proposed tracker based on the two proposed methods are shown in Figure 6.22 taking into account the different combinations such as:

- (Proposed): the tracker is based on HOG with DWT and hierarchical convolutional features with DWT.
- (Proposed_No_DWTinCNN): the tracker is based on HOG with DWT and hierarchical convolutional features without DWT.
- (Proposed_No_DWTinHOG): the tracker is based on HOG without DWT and hierarchical convolutional features with DWT.
- (Proposed_No_DWTinHOG&CNN): the tracker is based on HOG without DWT and hierarchical convolutional features without DWT.
- (Proposed_No_HOG): the tracker is based only on hierarchical convolutional features with DWT.
- (Proposed_No_HOG_No_DWTinCNN): the tracker is based on hierarchical convolutional features without DWT.

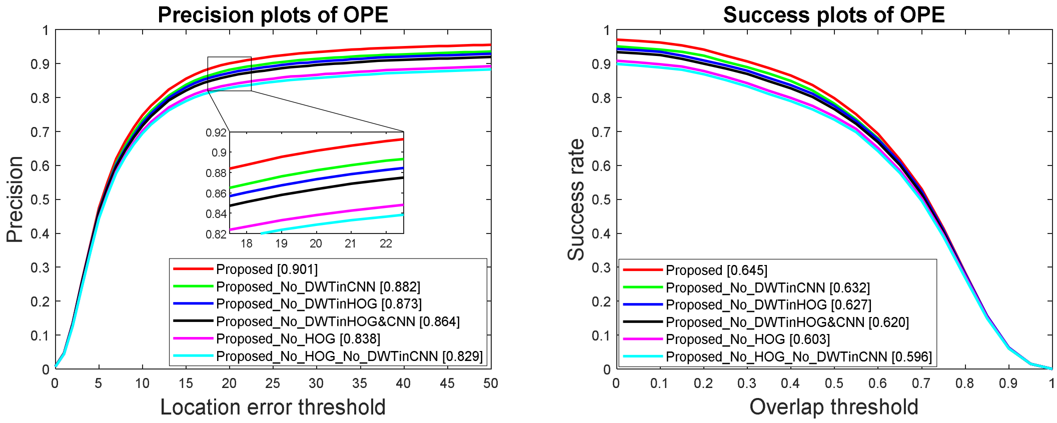


Figure 6. 22 : Performance evaluation of the proposed tracker at each stage using the OTB-100 benchmark.

It is clear that combining HCF, HOG and the wavelet has ensured optimal results. Although, HOG affects the proposed method in tracking object in under the occurrence of out-of-view, occlusion, out-of-plane rotation, motion blur , scale variation, deformation and illumination variation. From the obtained results, it can be said that the exploitation of wavelet in HOG has improved the proposed tracker in handling the occlusion, scale variation and out-of-plane rotation.

Furthermore, the use of wavelet in HCF improves the proposed tracker behavior in handling only the illumination variation occurrence. Through the above analysis, it can be concluded that the combination of hierarchical convolution features, HOG and DWT can greatly improve the robustness and accuracy of the proposed tracker.

Indeed, in the present analysis the use of different types of wavelets has been investigated on the proposed tracker's performance. Figure 6.23 shows the calculated precision for a series of distance thresholds (percentage of frames where the distance to the groundtruth is within the threshold) of Singer2 sequence. It can be clearly noticed that the type of wavelet 'bior2.4' lead to obtaining the best result of 98.4% compared to the other types, which further justifies the validity of the proposed approach.

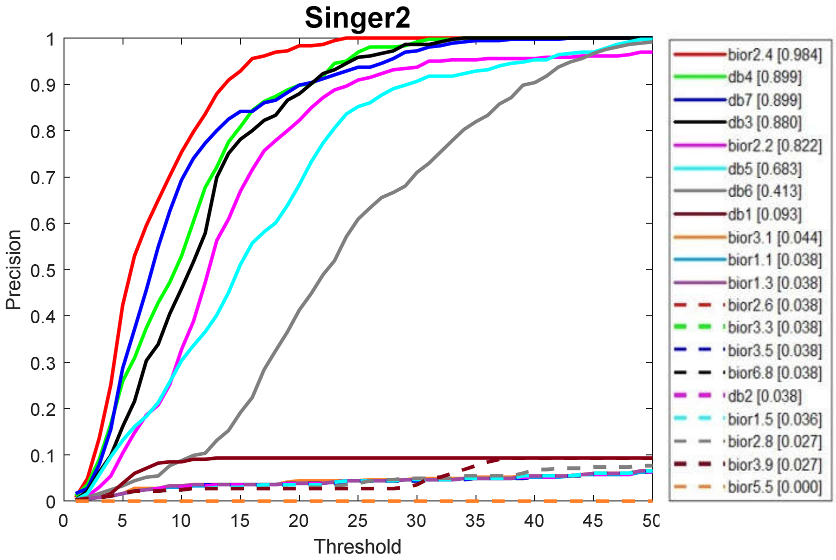


Figure 6. 23 : Precision of sequence -Singer2- with different types of wavelet.

6.5.2 Results of TC128

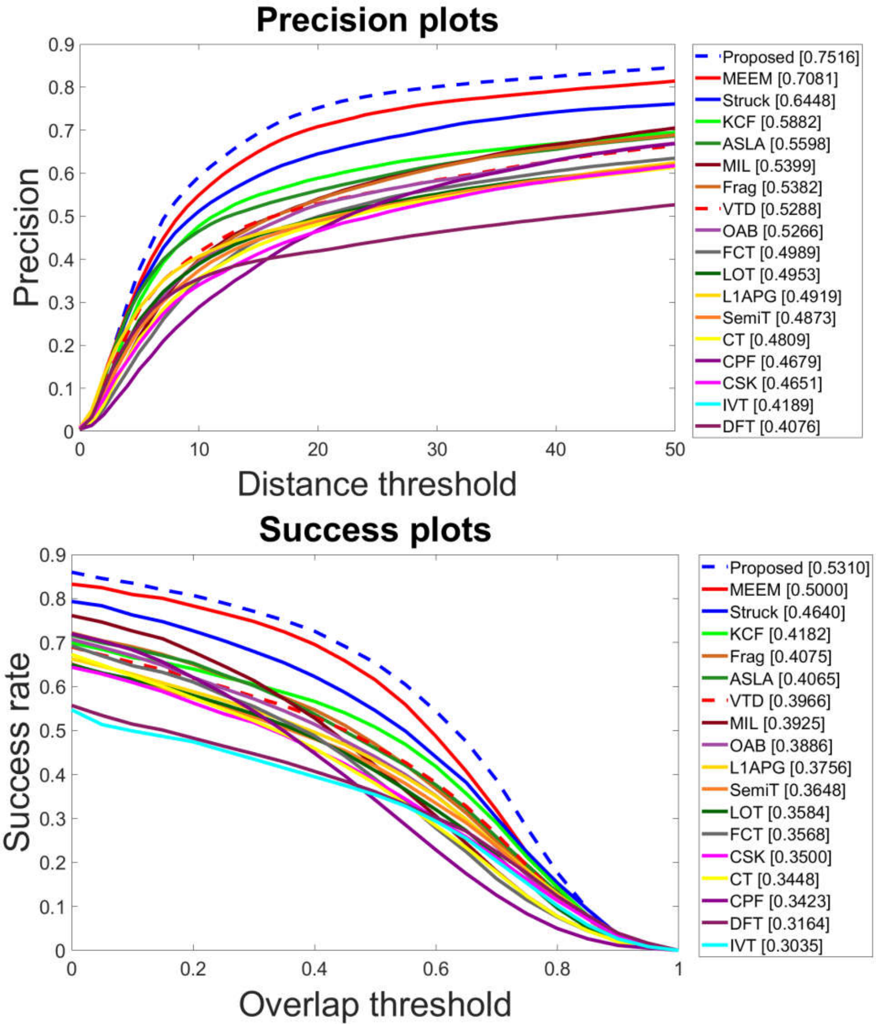


Figure 6. 24 : Success and precision plots for all color-enhanced trackers and proposed trackers on TColor-128

Figure. 6.24 shows the precision and success plots of visual tracking methods on TC-128 datasets. These results indicate that the proposed method has improved the average precision rate up to at least 4.35%, and at most 34.4% compared to the MEEM and DFT, and the average success rate at least 3.1%, and at most 22.75% compared to the MEEM and IVT on the TC-128 dataset, respectively.

6.5.3 Results of UAV20

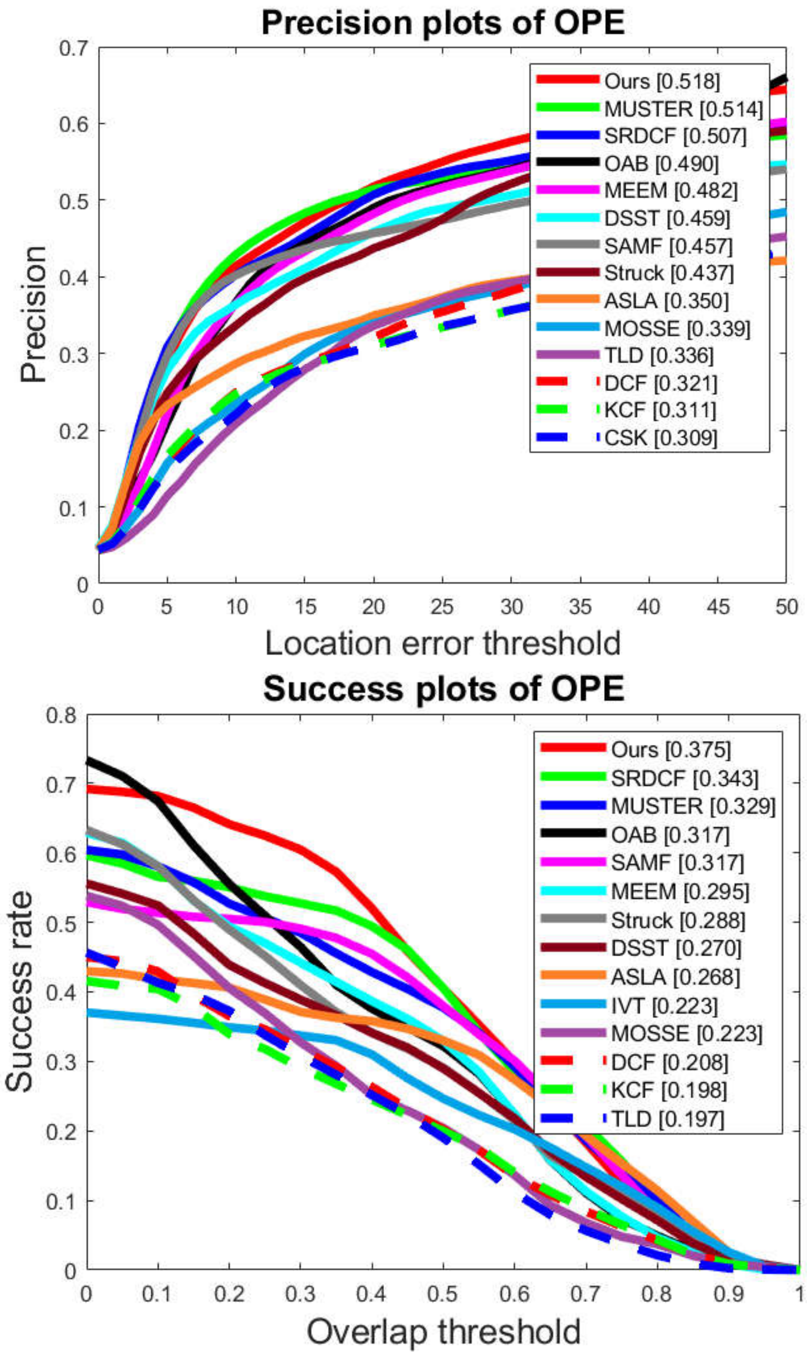


Figure 6. 25 : Comparison results with eight reference trackers on the UAV20L database [2] using distance precision (DP) and overlap success (OS) metrics.

Finally, we evaluate our method on the UAV20L database [2]. As shown in figure 6.25, the proposed tracker outperforms the other trackers. It is clear, these results indicate that the proposed method has improved the average precision rate up to at least 0.4%, and at most 20.9% compared to the MUSTER and CSK, and the average success rate at least 3.2%, and at most 17.8% compared to the SRDCF and TLD, respectively.

6.6 Conclusion

In this chapter, we compared the proposed tracker with different databases such as OTB50, OTB100, TC-128, and UAV20L. The performance of the proposed trackers has been evaluated based on two performance metrics such as the area-under-the-curve (AUC) and the distance precision (DP). On the other side, for the validation of the proposed tracker performance, a comparison has been carried out based on different trackers. The obtained results clearly prove the validity of the proposed approach in solving the encountered many problems of visual object tracking compared to other existing tracking approaches.

Chapter 7 :

CONCLUSION AND FUTURE WORK

CONCLUSION AND FUTURE WORK

Visual object tracking is an important part of video analysis, and it is widely used in real life. The complexity of video sequences has increased in tandem with the diversification of tracking objects. Visual object tracking algorithm research is also fraught with difficulties.

Even today, there is no tracker capable of overcoming all the difficult situations that can appear such as: such as illumination variation, occlusion, deformation, background clutter, etc.

In this thesis, we have given a detailed introduction to visual tracking. And we also have given all performance terms used to judge such a visual tracking method.

All algorithms for object tracking are based on images, so we have given detail of image color spaces, and imported methods of image features extraction, convolutional neural networks, and histograms of oriented gradients.

One of the most important fields of image analysis is the wavelet transforms which was presented in this thesis.

In this thesis, an effective combination among CNN layers features, the Hog features, and the discrete wavelet packet transform DWT image wavelet transforms has been proposed based on the exploitation of the hierarchical CNN feature which has been trained on a large-scale database. Therefore, the output layers of CNNs are used to preserve the semantics of the target objects, which are robust to significant appearance changes. Whereas, the input layers of the CNNs are exploited for encoding the more precise spatial details, which are useful for precise localization. Indeed, both features with precise details are used at the same time for visual object tracking. Whereas, a linear correlation filter has been trained on each CNN layer for the deduction of the targeted location based on hierarchical correlation maps in a coarse-to-fine manner. At the same time, to enhance the accuracy of the proposed tracker and to overcome the problem of drifting encountered during the update process of the correlation filter, it has been proposed in this thesis an approach for ensuring such process in real-time along each step. This approach is based on training the correlation filter on HOG features to make it as a tool to update the filters produced by CNN and HOG features. Furthermore, to improve the performance of the proposed tracker, the DWT has been proposed to achieve two main goals. Firstly, the calculation of the HOG features instead of using RGB, and secondly the calculation of CNN features in the case of images with high saturation to improve their performance.

The obtained results from the extensive simulations experiments on benchmark datasets have been conducted, such as OTB50, OTB100, TC-128, and UAV20, which have been carried out in this thesis show that the proposed tracker outperforms many modern trackers. However, despite the proven effectiveness of the proposed tracker, there is a need to further robustness improvement of the proposed tracker in the future with the aim that it can be a promising tracker to be applied within a wide range of cases without major deficiencies compared to the existing counterparts.

Bibliography

BIBLIOGRAPHY

1. Abbass, M.Y., Kwon, K.C., Kim, N., Abdelwahab, S.A., El-Samie, F.E.A., Khalaf, A.A.M.: Efficient object tracking using hierarchical convolutional features model and correlation filters. Vis. Comput. 37, 831–842 (2021). https://doi.org/10.1007/s00371-020-01833-5

2. Yang, X., Ma, C., Huang, J.-B., Yang, M.-H.: Hierarchical Convolutional Features for Visual Tracking. Proc. IEEE Int. Conf. Comput. Vis. 3074–3082 (2015). https://doi.org/10.1109/ICCV.2015.352

3. Ma, C., Huang, J. Bin, Yang, X., Yang, M.H.: Robust Visual Tracking via Hierarchical Convolutional Features. IEEE Trans. Pattern Anal. Mach. Intell. 41, 2709–2723 (2019). https://doi.org/10.1109/TPAMI.2018.2865311

4. Zgaren, A., Bouachir, W., Ksantini, R.: Coarse-to-Fine Object Tracking Using Deep Features and Correlation Filters. Lect. Notes Comput. Sci. 12509 LNCS, 517–529 (2020). https://doi.org/10.1007/978-3-030-64556-4_40

5. Zhang, J., Sun, J., Wang, J., Yue, X.G.: Visual object tracking based on residual network and cascaded correlation filters. J. Ambient Intell. Humaniz. Comput. (2020). https://doi.org/10.1007/s12652-020-02572-0

6. Bai, Y., Xu, T., Huang, B., Yang, R.: Deep Deblurring Correlation Filter for Object Tracking. IEEE Access. 8, 68623–68637 (2020). https://doi.org/10.1109/ACCESS.2020.2986311

7. Qi, Y., Zhang, S., Qin, L., Yao, H., Huang, Q., Lim, J., Yang, M.H.: Hedged Deep Tracking. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2016-Decem, 4303–4311 (2016). https://doi.org/10.1109/CVPR.2016.466

8. Touil, D.E., Terki, N., Medouakh, S.: Hierarchical convolutional features for visual tracking via two combined color spaces with SVM classifier. Signal, Image Video Process. 13, 359–368 (2019). https://doi.org/10.1007/s11760-018-1364-z

9. Touil, D.E., Terki, N., Medouakh, S.: Learning spatially correlation filters based on convolutional features via PSO algorithm and two combined color spaces for visual tracking. Appl. Intell. 48, 2837–2846 (2018). https://doi.org/10.1007/s10489-017-1120-z

10. Ma, C., Xu, Y., Ni, B., Yang, X.: When Correlation Filters Meet Convolutional Neural Networks for Visual Tracking. IEEE Signal Process. Lett. 23, 1454–1458 (2016). https://doi.org/10.1109/LSP.2016.2601691

11. Xing, W., Liu, W., Wang, J., Zhang, S., Wang, L., Yang, Y., Song, B.: Visual Object Tracking from Correlation Filter to Deep Learning. Vis. Object Track. from Correl. Filter to Deep Learn. (2021). https://doi.org/10.1007/978-981-16-6242-3

12. Norvig, P.R., Intelligence, S.A.: A modern approach. Prentice Hall Upper Saddle River, NJ, USA: (2002)

13. Zhang, X., Gao, H., Guo, M., Li, G., Liu, Y., Li, D.: A study on key technologies of unmanned driving. CAAI Trans. Intell. Technol. 1, 4–13 (2016). https://doi.org/10.1016/J.TRIT.2016.03.003

14. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. J. Mach. Learn. Res. 12, 2493–2537 (2011)

15. Parkhi, O.M., Vedaldi, A., Zisserman, A.: Deep Face Recognition. Br. Mach. Vis. Conf. 41.1-41.12 (2015). https://doi.org/10.5244/c.29.41

16. Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M.: Mastering the game of Go with deep neural networks and tree search. Nature. 529, 484–489 (2016)

17. Tai, J.-C., Tseng, S.-T., Lin, C.-P., Song, K.-T.: Real-time image tracking for automatic traffic monitoring and enforcement applications. Image Vis. Comput. 22, 485–501 (2004)

18. Rautaray, S.S., Agrawal, A.: Vision based hand gesture recognition for human computer interaction: a survey. Artif. Intell. Rev. 43, 1–54 (2015)

19. Coifman, B., Beymer, D., McLauchlan, P., Malik, J.: A real-time computer vision system for vehicle tracking and traffic surveillance. Transp. Res. Part C Emerg. Technol. 6, 271–288 (1998)

20. Tang, Z., Naphade, M., Liu, M.-Y., Yang, X., Birchfield, S., Wang, S., Kumar, R., Anastasiu, D., Hwang, J.-N.: Cityflow: A city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8797–8806 (2019)

21. Pavlovic, V.I., Sharma, R., Huang, T.S.: Visual interpretation of hand gestures for human-computer interaction: A review. IEEE Trans. Pattern Anal. Mach. Intell. 19, 677–695 (1997)

22. Masi, I., Chang, F.-J., Choi, J., Harel, S., Kim, J., Kim, K., Leksut, J., Rawls, S., Wu, Y., Hassner, T.: Learning pose-aware models for pose-invariant face recognition in the wild. IEEE Trans. Pattern Anal. Mach. Intell. 41, 379–393 (2018)

23. Haritaoglu, I., Harwood, D., Davis, L.S.: W/sup 4: real-time surveillance of people and their activities. IEEE Trans. Pattern Anal. Mach. Intell. 22, 809–830 (2000)

24. Ko, S.-Y., Kwon, D.-S.: A surgical knowledge based interaction method for a laparoscopic assistant robot. In: RO-MAN 2004. 13th IEEE International Workshop on Robot and Human Interactive Communication (IEEE Catalog No. 04TH8759). pp. 313–318. IEEE (2004)

25. Wu, B., Ji, D., Guo, Z., Shen, H., Xiao, Z.: A method for plane-symmetrical vehicle trajectory tracking in maneuver flight. In: 2016 35th Chinese Control Conference (CCC). pp. 5743–5746. IEEE (2016)

26. Lei, Q., Di, Z., Jun-long, L.: Tracking for near space nonballistic target based on

several filter algorithms. In: 2015 34th Chinese Control Conference (CCC). pp. 4997–5002. IEEE (2015)

27. Kanade, T., Collins, R.T., Lipton, A.J., Fujiyoshi, H., Duggins, D.: A System for Video Surveillance and Monitoring CMU VSAM Final Report. CARNEGIE-MELLON UNIV PITTSBURGH PA ROBOTICS INST (1999)

28. Tanawongsuwan, R., Bobick, A.F.: Characteristics of time-distance gait parameters across speeds. Georgia Institute of Technology (2003)

29. Wu, Y., Lim, J., Yang, M.H.: Online object tracking: A benchmark. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2411–2418 (2013). https://doi.org/10.1109/CVPR.2013.312

30. Vision, M.D.: A Computational Investigation into the Human Representation. Process. Vis. Information, Free. WH Company. San Fr. (1982)

31. Smeulders, A.W.M., Chu, D.M., Cucchiara, R., Calderara, S., Dehghan, A., Shah, M.: Visual tracking: An experimental survey. IEEE Trans. Pattern Anal. Mach. Intell. 36, 1442–1468 (2013)

32. Briechle, K., Hanebeck, U.D.: Template matching using fast normalized cross correlation. In: Optical Pattern Recognition XII. pp. 95–102. International Society for Optics and Photonics (2001)

33. Comaniciu, D., Ramesh, V., Meer, P.: Real-time tracking of non-rigid objects using mean shift. In: Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662). pp. 142–149. IEEE (2000)

34. Hare, S., Golodetz, S., Saffari, A., Vineet, V., Cheng, M.M., Hicks, S.L., Torr, P.H.S.: Struck: Structured Output Tracking with Kernels. IEEE Trans. Pattern Anal. Mach. Intell. 38, 2096–2109 (2011). https://doi.org/10.1109/TPAMI.2015.2509974

35. Godec, M., Roth, P.M., Bischof, H.: Hough-based tracking of non-rigid objects. Comput. Vis. Image Underst. 117, 1245–1256 (2013)

36. Kwon, J., Lee, K.M.: Tracking by sampling trackers. In: 2011 International Conference on Computer Vision. pp. 1195–1202. IEEE (2011)

37. Nilski, A.: An evaluation metric for multiple camera tracking systems: The i-LIDS 5th scenario. In: Optics and Photonics for Counterterrorism and Crime Fighting IV. pp. 55–62. SPIE (2008)

38. Babenko, B., Yang, M.-H., Belongie, S.: Visual tracking with online multiple instance learning. In: 2009 IEEE Conference on computer vision and Pattern Recognition. pp. 983–990. IEEE (2009)

39. Chu, D.M., Smeulders, A.W.M.: Color invariant surf in discriminative object tracking. In: European Conference on Computer Vision. pp. 62–75. Springer (2010)

40. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (SURF). Comput. Vis. image Underst. 110, 346–359 (2008)

41. Ross, D.A., Lim, J., Lin, R.-S., Yang, M.-H.: Incremental learning for robust visual tracking. Int. J. Comput. Vis. 77, 125–141 (2008)

42. Mei, X., Ling, H.: Robust visual tracking using ℓ 1 minimization. In: 2009 IEEE 12th international conference on computer vision. pp. 1436–1443. IEEE (2009)

43. Kalal, Z., Matas, J., Mikolajczyk, K.: Pn learning: Bootstrapping binary classifiers by structural constraints. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 49–56. IEEE (2010)

44. Baker, S., Matthews, I.: Lucas-kanade 20 years on: A unifying framework. Int. J. Comput. Vis. 56, 221–255 (2004)

45. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: High-speed tracking with kernelized correlation filters. IEEE Trans. Pattern Anal. Mach. Intell. 37, 583–596 (2015). https://doi.org/10.1109/TPAMI.2014.2345390

46. Zhang, Z., Peng, H.: Deeper and wider siamese networks for real-time visual tracking. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2019-June, 4586–4595 (2019). https://doi.org/10.1109/CVPR.2019.00472

47. Li, B., Yan, J., Wu, W., Zhu, Z., Hu, X.: High performance visual tracking with siamese region proposal network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8971–8980 (2018)

48. Porikli, F., Tuzel, O., Meer, P.: Covariance tracking using model update based on lie algebra. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). pp. 728–735. IEEE (2006)

49. Wu, Y., Cheng, J., Wang, J., Lu, H.: Real-time visual tracking via incremental covariance tensor learning. In: 2009 IEEE 12th International Conference on Computer Vision. pp. 1631–1638. IEEE (2009)

50. Avidan, S.: Support vector tracking. IEEE Trans. Pattern Anal. Mach. Intell. 26, 1064–1072 (2004)

51. Matthies, L., Kanade, T., Szeliski, R.: Kalman filter-based algorithms for estimating depth from image sequences. Int. J. Comput. Vis. 3, 209–238 (1989)

52. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. Int. J. Comput. Vis. 1, 321–331 (1988)

53. Nam, H., Han, B.: Learning multi-domain convolutional neural networks for visual tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4293–4302 (2016)

54. Fan, H., Ling, H.: Sanet: Structure-aware network for visual tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 42–49 (2017)

55. Guo, Q., Feng, W., Zhou, C., Huang, R., Wan, L., Wang, S.: Learning dynamic siamese network for visual object tracking. In: Proceedings of the IEEE international

conference on computer vision. pp. 1763–1771 (2017)

56. Abdel-Hadi, A.: Real-time object tracking using color-based Kalman particle filter. In: The 2010 International Conference on Computer Engineering & Systems. pp. 337–341. IEEE (2010)

57. Han, Z., Xu, T., Chen, Z.: An improved color-based tracking by particle filter. In: Proceedings 2011 International Conference on Transportation, Mechanical, and Electrical Engineering (TMEE). pp. 2512–2515. IEEE (2011)

58. Yang, F., Lu, H., Yang, M.-H.: Robust superpixel tracking. IEEE Trans. Image Process. 23, 1639–1651 (2014)

59. Bolme, D.S., Draper, B.A., Beveridge, J.R.: Average of synthetic exact filters. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 2105–2112. IEEE (2009)

60. Bolme, D.S., Beveridge, J.R., Draper, B.A., Lui, Y.M.: Visual object tracking using adaptive correlation filters. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2544–2550 (2010). https://doi.org/10.1109/CVPR.2010.5539960

61. Henriques, J.F., Caseiro, R., Martins, P., Batista, J.: Exploiting the circulant structure of tracking-by-detection with kernels. Springer Berlin Heidelb. Computer V, 702–715 (2012). https://doi.org/10.1007/978-3-642-33765-9_50

62. Danelljan, M., Häger, G., Khan, F.S., Felsberg, M.: Accurate scale estimation for robust visual tracking. BMVC 2014 - Proc. Br. Mach. Vis. Conf. 2014. 7584 (2014). https://doi.org/10.5244/c.28.65

63. Hong, S., You, T., Kwak, S., Han, B.: Online tracking by learning discriminative saliency map with convolutional neural network. 32nd Int. Conf. Mach. Learn. ICML 2015. 1, 597–606 (2015). https://doi.org/10.48550/arXiv.1502.06796

64. Danelljan, M., Robinson, A., Shahbaz Khan, F., Felsberg, M.: Beyond correlation filters: Learning continuous convolution operators for visual tracking. In: European conference on computer vision. pp. 472–488. Springer (2016)

65. Bertinetto, L., Valmadre, J., Henriques, J.F., Vedaldi, A., Torr, P.H.S.: Fully-convolutional siamese networks for object tracking. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 9914 LNCS, 850–865 (2016). https://doi.org/10.1007/978-3-319-48881-3_56

66. Wang, Q., Zhang, L., Bertinetto, L., Hu, W., Torr, P.H.S.: Fast online object tracking and segmentation: A unifying approach. In: Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition. pp. 1328–1338 (2019)

67. Zhang, L., Gonzalez-Garcia, A., Weijer, J. van de, Danelljan, M., Khan, F.S.: Learning the model update for siamese trackers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4010–4019 (2019)

68. Huang, L., Zhao, X., Huang, K.: Bridging the gap between detection and tracking: A

unified approach. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3999–4009 (2019)

69. Wang, G., Luo, C., Sun, X., Xiong, Z., Zeng, W.: Tracking by instance detection: A meta-learning approach. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6288–6297 (2020)

70. Shin, J., Kim, H., Kim, D., Paik, J.: Fast and robust object tracking using tracking failure detection in kernelized correlation filter. Appl. Sci. 10, (2020). https://doi.org/10.3390/app10020713

71. Li, Y., Zhu, J.: A scale adaptive kernel correlation filter tracker with feature integration. Springer Int. Publ. 254–265 (2015). https://doi.org/10.1007/978-3-319-16181-5_18

72. Danelljan, M., Hager, G., Khan, F.S., Felsberg, M.: Learning spatially regularized correlation filters for visual tracking. Proc. IEEE Int. Conf. Comput. Vis. 2015 Inter, 4310–4318 (2015). https://doi.org/10.1109/ICCV.2015.490

73. Galoogahi, H.K., Fagg, A., Lucey, S.: Learning Background-Aware Correlation Filters for Visual Tracking. Proc. IEEE Int. Conf. Comput. Vis. 2017-Octob, 1144–1152 (2017). https://doi.org/10.1109/ICCV.2017.129

74. Ma, C., Yang, X., Zhang, C., Yang, M.H.: Long-term correlation tracking. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 07-12-June, 5388–5396 (2015). https://doi.org/10.1109/CVPR.2015.7299177

75. Bertinetto, L., Valmadre, J., Golodetz, S., Miksik, O., Torr, P.H.S.: Staple: Complementary learners for real-time tracking. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2016-Decem, 1401–1409 (2016). https://doi.org/10.1109/CVPR.2016.156

76. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. IEEE Trans. Pattern Anal. Mach. Intell. 39, 1137–1149 (2017). https://doi.org/10.1109/TPAMI.2016.2577031

77. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2016-Decem, 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90

78. Wu, Y., Wei, J., Yin, J., Liu, X., Zhang, J.: Deep Collaborative Filtering Based on Outer Product. IEEE Access. 8, 85567–85574 (2020). https://doi.org/10.1109/ACCESS.2020.2992519

79. Danelljan, M., Hager, G., Khan, F.S., Felsberg, M.: Convolutional Features for Correlation Filter Based Visual Tracking. Proc. IEEE Int. Conf. Comput. Vis. 2016-Febru, 621–629 (2016). https://doi.org/10.1109/ICCVW.2015.84

80. Song, Y., Ma, C., Gong, L., Zhang, J., Lau, R.W.H., Yang, M.H.: CREST: Convolutional Residual Learning for Visual Tracking. Proc. IEEE Int. Conf. Comput. Vis. 2017-Octob, 2574–2583 (2017). https://doi.org/10.1109/ICCV.2017.279

81. Sturm, B.L.: Stéphane Mallat: A Wavelet Tour of Signal Processing, 2nd Edition. Acad. Press. (1999). https://doi.org/https://doi.org/10.1016/B978-0-12-466606-1.X5000-4

82. Bruna, J., Mallat, S.: Invariant scattering convolution networks. IEEE Trans. Pattern Anal. Mach. Intell. 35, 1872–1886 (2013). https://doi.org/10.1109/TPAMI.2012.230

83. Bae, W., Yoo, J., Ye, J.C.: Beyond Deep Residual Learning for Image Restoration: Persistent Homology-Guided Manifold Simplification. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work. 2017-July, 1141–1149 (2017). https://doi.org/10.1109/CVPRW.2017.152

84. Deeba, F., Kun, S., Ali Dharejo, F., Zhou, Y.: Wavelet-based enhanced medical image super resolution. IEEE Access. 8, 37035–37044 (2020). https://doi.org/10.1109/ACCESS.2020.2974278

85. Liu, X., Zhang, H., Cheung, Y. ming, You, X., Tang, Y.Y.: Efficient single image dehazing and denoising: An efficient multi-scale correlated wavelet approach. Comput. Vis. Image Underst. 162, 23–33 (2017). https://doi.org/10.1016/J.CVIU.2017.08.002

86. Babenko, B., Yang, M.H., Belongie, S.: Robust object tracking with online multiple instance learning. IEEE Trans. Pattern Anal. Mach. Intell. 33, 1619–1632 (2011). https://doi.org/10.1109/TPAMI.2010.226

87. Bai, Q., Wu, Z., Sclaroff, S., Betke, M., Monnier, C.: Randomized ensemble tracking. Proc. IEEE Int. Conf. Comput. Vis. 2040–2047 (2013). https://doi.org/10.1109/ICCV.2013.255

88. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-Learning-Detection. IEEE Trans. Pattern Anal. Mach. Intell. 34, 1409–1422 (2012). https://doi.org/10.1109/TPAMI.2011.239

89. Wang, J., Yang, H., Xu, N., Wu, C., Zhao, Z., Zhang, J., Wu, D.O.: Long-term target tracking combined with re-detection. EURASIP J. Adv. Signal Process. 2021, (2021). https://doi.org/10.1186/s13634-020-00713-3

90. Proposals, I., Liu, H., Hu, Q., Li, B., Guo, Y.: Robust long-term tracking via instance-specific proposals. IEEE Trans. Instrum. Meas. 69, 950–962 (2020). https://doi.org/10.1109/TIM.2019.2908715

91. Ma, C., Huang, J. Bin, Yang, X., Yang, M.H.: Adaptive Correlation Filters with Long-Term and Short-Term Memory for Object Tracking. Int. J. Comput. Vis. 126, 771–796 (2018). https://doi.org/10.1007/s11263-018-1076-4

92. Zhang, J., Ma, S., Sclaroff, S.: MEEM: Robust tracking via multiple experts using entropy minimization. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 8694 LNCS, 188–203 (2014). https://doi.org/10.1007/978-3-319-10599-4_13

93. Wu, Y., Lim, J., Yang, M.-H.: Online Object Tracking: A Benchmark Supplemental Material. 2013 IEEE Conf. Comput. Vis. Pattern Recognit. 1–13 (2013)

94. Sidney, B.C.: Introduction to wavelets and wavelet transforms: a primer, (1998)

95. Debnath, L., Shah, F.A.: Lecture notes on wavelet transforms. Springer (2017)

96. Hair, A.: On the theory of orthogonal function systems. Math. Ann. 69, 331–371 (1910)

97. Morlet, J., Arens, G., Fourgeau, E., Glard, D.: Wave propagation and sampling theory—Part I: Complex signal and scattering in multilayered media. Geophysics. 47, 203–221 (1982)

98. Debnath, L., Shah, F.A.: Wavelet transforms and their applications. Springer (2002)

99. Sifuzzaman, M., Islam, M.R., Ali, M.Z.: Application of wavelet transform and its advantages compared to Fourier transform. (2009)

100. Morlet, J., Arens, G., Fourgeau, E., Giard, D.: Wave propagation and sampling theory—Part II: Sampling theory and complex waves. Geophysics. 47, 222–236 (1982)

101. Grossmann, A., Morlet, J.: Decomposition of Hardy functions into square integrable wavelets of constant shape. SIAM J. Math. Anal. 15, 723–736 (1984)

102. Gonzalez, R.C., Woods, R.E., Eddins, S.L.: Digital Image Processing Using MATLAB. Pearson Prentice Hall (2004)

103. Kumar, A., Shaik, F.: Image processing in diabetic related causes. Springer (2015)

104. Kolkur, S., Kalbande, D., Shimpi, P., Bapat, C., Jatakia, J.: Human Skin Detection Using RGB, HSV and YCbCr Color Models. In: Proceedings of the International Conference on Communication and Signal Processing 2016 (ICCASP 2016). pp. 324–332. Atlantis Press (2016)

105. Prajakta M.Patil, Y.M.P.: Robust Skin Colour Detection And Tracking Algorithm. Int. J. Eng. Res. Technol. 1, 1–6 (2012)

106. Kurniastuti, I., Wulan, T.D., Andini, A.: Color Feature Extraction of Fingernail Image based on HSV Color Space as Early Detection Risk of Diabetes Mellitus. 2021 Int. Conf. Comput. Sci. Inf. Technol. Electr. Eng. ICOMITEE 2021. 51–55 (2021). https://doi.org/10.1109/ICOMITEE53461.2021.9650161

107. Kurniastuti, I., Yuliati, E.N.I., Yudianto, F., Wulan, T.D.: Determination of Hue Saturation Value (HSV) color feature in kidney histology image. J. Phys. Conf. Ser. 2157, 012020 (2022). https://doi.org/10.1088/1742-6596/2157/1/012020

108. Wan, Y., Chen, Q.: Joint image dehazing and contrast enhancement using the HSV color space. 2015 Vis. Commun. Image Process. VCIP 2015. (2015). https://doi.org/10.1109/VCIP.2015.7457892

109. Ganesan, P., Rajini, V., Sathish, B.S., Shaik, K.B.: HSV color space based segmentation of region of interest in satellite images. 2014 Int. Conf. Control. Instrumentation, Commun. Comput. Technol. ICCICCT 2014. 101–105 (2014). https://doi.org/10.1109/ICCICCT.2014.6992938

110. Charoensawan, P., Phongsuphap, S., Shimizu, I.: Comparison of Fabric Color Naming Using RGB and HSV Color Models. Proceeding 2018 15th Int. Jt. Conf. Comput. Sci. Softw. Eng. JCSSE 2018. (2018). https://doi.org/10.1109/JCSSE.2018.8457329

111. Zarit, B.D., Super, B.J., Quek, F.K.H.: Comparison of five color models in skin pixel classification. In: Proceedings International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems. In Conjunction with ICCV'99 (Cat. No.PR00378). pp. 58–63 (1999)

112. Poorani M,Prathiba T, R.G.: Integrated Feature Extraction for Image Retrieval. Int. J. Comput. Sci. Mob. Comput. 2, 28–35 (2013)

113. Karpathy, A., Li, F.F., Johnson, J.: Cs231n convolutional neural networks for visual recognition. 2016. URL http//cs231n. github. io. 50, (2017)

114. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. 25, (2012)

115. Romanuke, V.: Appropriate number and allocation of ReLUs in convolutional neural networks. Res. Bull. Natl. Tech. Univ. Ukr. Kyiv Politech. Institute". 69–78 (2017)

116. Babenko, A., Slesarev, A., Chigorin, A., Lempitsky, V.: Neural Codes for Image Retrieval. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics). 8689 LNCS, 584–599 (2014). https://doi.org/10.48550/arxiv.1404.1777

117. Bishop, C.M.: Pattern Recoginiton and Machine Learning. Inf. Sci. Stat. 738 (2006)

118. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE. 86, 2278–2323 (1998). https://doi.org/10.1109/5.726791

119. Zeiler, M.D., Fergus, R.: Visualizing and Understanding Convolutional Networks. Comput. Vision–ECCV 2014. 8689, 818–833 (2014). https://doi.org/10.1007/978-3-319-10590-1_53

120. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 07-12-June-2015, 1–9 (2015). https://doi.org/10.1109/CVPR.2015.7298594

121. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. In: Bengio, Y. and LeCun, Y. (eds.) 3rd International Conference on Learning Representations, {ICLR} 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (2015)

122. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. Proc. - 2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition, CVPR 2005. I, 886–893 (2005). https://doi.org/10.1109/CVPR.2005.177

123. S, N., C.J, P.: Low-Level Features for Image Retrieval Based on Extraction of

Directional Binary Patterns And its Oriented Gradients Histogram. Comput. Appl. An Int. J. 2, 13–28 (2015). https://doi.org/10.5121/caij.2015.2102

124. Gonzalez, R.C.: Digital image processing. Pearson education india (2009)

125. Stollnitz, E.J., DeRose, T.D., DeRose, A.D., Salesin, D.H.: Wavelets for computer graphics: theory and applications. Morgan Kaufmann (1996)

126. Daubechies, I.: Ten lectures on wavelets. SIAM (1992)

127. Goupillaud, P., Grossmann, A., Morlet, J.: Cycle-octave and related transforms in seismic signal analysis. Geoexploration. 23, 85–102 (1984)

128. Vetterli, M.: Fast 2-D discrete cosine transform. In: ICASSP'85. IEEE International Conference on Acoustics, Speech, and Signal Processing. pp. 1538–1541 (1985)

129. Daubechies, I.: Where do wavelets come from? A personal point of view. Proc. IEEE. 84, 510–513 (1996)

130. Chun-Lin, L.: A tutorial of the wavelet transform. NTUEE, Taiwan. 21, 22 (2010)

131. Porwik, P., Lisowska, A.: The Haar-wavelet transform in digital image processing: its status and achievements. Mach. Graph. Vis. 13, 79–98 (2004)

132. Kessler, B.M., Payne, G.L., Polyzou, W.N.: Wavelet notes. arXiv Prepr. nucl-th/0305025. (2003)

133. Sridhar, S., Kumar, P.R., Ramanaiah, K. V: Wavelet transform techniques for image compression-an evaluation. Int. J. image, Graph. signal Process. 6, 54 (2014)

134. Rout, S.: Orthogonal vs. biorthogonal wavelets for image compression, (2003)

135. Maggioni, M., Bremer Jr, J.C., Coifman, R.R., Szlam, A.D.: Biorthogonal diffusion wavelets for multiscale representation on manifolds and graphs. In: Wavelets XI. pp. 543–555 (2005)

136. Ashmead, J.: Morlet wavelets in quantum mechanics. arXiv Prepr. arXiv1001.0250. (2010)

137. Torrence, C., Compo, G.P.: A practical guide to wavelet analysis. Bull. Am. Meteorol. Soc. 79, 61–78 (1998)

138. Mi, X., Ren, H., Ouyang, Z., Wei, W., Ma, K.: The use of the Mexican Hat and the Morlet wavelets for detection of ecological patterns. Plant Ecol. 179, 1–19 (2005)

139. Xudong, T., Yiqing, D., Xinyuan, L., Jianru, L.: Design of orthonormal filter banks based on meyer wavelet. Int. J. Adv. Comput. Sci. Appl. 6, 109–112 (2015)

140. Singha, M., Hemachandran, K., Paul, A.: Content-based image retrieval using the combination of the fast wavelet transformation and the colour histogram. IET Image Process. 6, 1221–1226 (2012)

141. Dharejo, F.A., Zhou, Y., Deeba, F., Jatoi, M.A., Khan, M.A., Mallah, G.A., Ghaffar, A., Chhattal, M., Du, Y., Wang, X.: A deep hybrid neural network for single image

dehazing via wavelet transform. Optik (Stuttg). 231, 1–13 (2021). https://doi.org/10.1016/j.ijleo.2021.166462

142. Deng, J., Dong, W., Socher, R., Li, L.-J., Kai Li, Li Fei-Fei: ImageNet: A large-scale hierarchical image database. 2009 IEEE Conf. Comput. Vis. Pattern Recognit. 248–255 (2010). https://doi.org/10.1109/CVPR.2009.5206848

143. Fujieda, S., Takayama, K., Hachisuka, T.: Wavelet Convolutional Neural Networks. arXiv Prepr. arXiv1805.08620. (2018). https://doi.org/10.48550/arXiv.1805.08620

144. Galoogahi, H.K., Sim, T., Lucey, S.: Multi-channel correlation filters. Proc. IEEE Int. Conf. Comput. Vis. 3072–3079 (2013). https://doi.org/10.1109/ICCV.2013.381

145. Wu, Y., Lim, J., Yang, M.H.: Object tracking benchmark. IEEE Trans. Pattern Anal. Mach. Intell. 37, 1834–1848 (2015). https://doi.org/10.1109/TPAMI.2014.2388226

146. Boddeti, V.N., Kanade, T., Kumar, B.V.K.V.: Correlation filters for object alignment. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2291–2298 (2013). https://doi.org/10.1109/CVPR.2013.297

147. Vedaldi, A., Lenc, K.: MatConvNet - Convolutional Neural Networks for MATLAB. MM 2015 - Proc. 2015 ACM Multimed. Conf. 689–692 (2014). https://doi.org/10.48550/arXiv.1412.4564

148. Lu, H., Jia, X., Yang, M.-H.: Visual tracking via adaptive structural local sparse appearance model. 2012 IEEE Conf. Comput. Vis. Pattern Recognit. 1822–1829 (2012). https://doi.org/10.1109/CVPR.2012.6247880

149. Zhibin, H., Chen, Z., Wang, C., Mei, X., Prokhorov, D., Tao, D.: MUlti-Store Tracker (MUSTer): a Cognitive Psychology Inspired Approach to Object Tracking. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 07-12-June, 749–758 (2015). https://doi.org/10.1109/CVPR.2015.7298675

150. Li, X., Liu, Q., Fan, N., Zhou, Z., He, Z., Jing, X. yuan: Dual-regression model for visual tracking. Neural Networks. 132, 364–374 (2020). https://doi.org/10.1016/j.neunet.2020.09.011

151. Yang, T., Chan, A.B.: Visual Tracking via Dynamic Memory Networks. IEEE Trans. Pattern Anal. Mach. Intell. 43, 360–374 (2021). https://doi.org/10.1109/TPAMI.2019.2929034

152. Danelljan, M., Gustav, H., Khan, F.S., Felsberg, M.: Adaptive Decontamination of the Training Set : A Unified Formulation for Discriminative Visual Tracking. IEEE Conf. Comput. Vis. Pattern Recognit. 1430–1438 (2016). https://doi.org/10.1109/CVPR.2016.159

153. Bourennane, M., Terki, N., Hamiane, M., Kouzou, A.: An Enhanced Visual Object Tracking Approach based on Combined Features of Neural Networks, Wavelet Transforms, and Histogram of Oriented Gradients. Eng. Technol. Appl. Sci. Res. 12, 8745–8754 (2022). https://doi.org/10.48084/etasr.5026

154. Dai, K., Wang, Y.: End-to-end DeepNCC framework for robust visual tracking. J. Vis. Commun. Image Represent. 70, 102800 (2020). https://doi.org/10.1016/j.jvcir.2020.102800

155. Liu, S., Liu, D., Muhammad, K., Ding, W.: Effective template update mechanism in visual tracking with background clutter. Neurocomputing. 458, 615–625 (2021). https://doi.org/https://doi.org/10.1016/j.neucom.2019.12.143

156. Peng, Z., Lu, X.J.: Learning region sparse constraint correlation filter for tracking. Signal Process. Image Commun. 90, 116042 (2021). https://doi.org/10.1016/j.image.2020.116042

157. Shi, T.A.O., Wang, D., Ren, H.: Triplet Network Template for Siamese Trackers. 9, (2021). https://doi.org/10.1109/ACCESS.2021.3066294

158. Zhang, Y., Liu, K., Wang, T.: End-to-end Visual Object Tracking with Motion Saliency Guidance. Chinese Control Conf. CCC. 2020-July, 6566–6571 (2020). https://doi.org/10.23919/CCC50068.2020.9188450

159. Cui, Z., Lu, N.: Feature selection accelerated convolutional neural networks for visual tracking. Appl. Intell. (2021). https://doi.org/10.1007/s10489-021-02234-4

160. Zhang, Y., Liu, G., Gao, J., Zhang, H.: Robust visual tracker integrating adaptively foreground segmentation into multi-feature fusion framework. Multimed. Tools Appl. 79, 31865–31888 (2020). https://doi.org/10.1007/s11042-020-09443-y

161. Han, Z., Wang, P., Ye, Q.: Adaptive Discriminative Deep Correlation Filter for Visual Object Tracking. IEEE Trans. Circuits Syst. Video Technol. 30, 155–166 (2020). https://doi.org/10.1109/TCSVT.2018.2888492

162. Chen, G., Pan, G., Zhou, Y., Kang, W., Hou, J., Deng, F.: Correlation Filter Tracking via Distractor-Aware Learning and Multi-Anchor Detection. IEEE Trans. Circuits Syst. Video Technol. 30, 4810–4822 (2020). https://doi.org/10.1109/TCSVT.2019.2961999

Publication in journal

- **M. Bourennane,** N. Terki, M. Hamiane, and A. Kouzou, “An Enhanced Visual Object Tracking Approach based on Combined Features of Neural Networks, Wavelet Transforms, and Histogram of Oriented Gradients”, Eng. Technol. Appl. Sci. Res., vol. 12, no. 3, pp. 8745–8754, Jun. 2022. **DOI : https://doi.org/10.48084/etasr.5026**