REAL TIME TRACKING USING NATURE-INSPIRED ALGORITHMS.

BY

INTEKHAB ALAM

A thesis submitted to the University of Birmingham for the degree of

DOCTOR OF PHILOSOPHY

School of Electronics, Electrical and Systems Engineering
College of Engineering & Physical Sciences
University of Birmingham
March 2017

Abstract

This thesis investigates the core difficulties in the tracking field of computer vision. The aim is to develop a suitable tuning free optimisation strategy so that a real time tracking could be achieved. The population and multi-solution based approaches have been applied first to analyse the convergence behaviours in the evolutionary test cases. The aim is to identify the core misconceptions in the manner the search characteristics of particles are defined in the literature. A general perception in the scientific community is that the particle based methods are not suitable for the real time applications. This thesis improves the convergence properties of particles by a novel scale free correlation approach. By altering the fundamental definition of a particle and by avoiding the nostalgic operations the tracking was expedited to a rate of 250 FPS.

There is a reasonable amount of similarity between the tracking landscapes and the ones generated by three dimensional evolutionary test cases. Several experimental studies are conducted that compares the performances of the novel optimisation to the ones observed with the swarming methods. It is therefore concluded that the modified particle behaviour outclassed the traditional approaches by huge margins in almost every test scenario.

This study is dedicated to

The **CENTRE POINT** *and the* **WORLD’S HOMELESS**

Acknowledgment

I acknowledge here the encouragement from some well wishers to write this report.

I am very grateful for the kind help from the following university staff

- *Professor Clive Roberts*
- *Dr Jeff Bale*
- *Dr Phillipa Semper*
- *Dr Rosie Day*
- *Professor Ganesh Gupta*
- *Dr Kieran McGovern*

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

CAD-Computer aided design

CNC-Computerised numerical control

CCD-Charged coupled device

CMOS-Complementary metal oxide semiconductor

CIS-Contact image sensor

FPS-Frames per second

PSO-Particle swarm optimization

DE-Differential evolution

SDS-Stochastic differential search

ROI-Region of interest

OOI-Object of Interest

KF-Kalman filter

NSS-Nest site Selection

RSO-Radical search optimisation/Random search optimisation

IF-Information fusion

CAC-Competition and collaboration

SOS-Save our souls

VA-Virtual assisted

VGS-Virtual guided search

RGB-Red Green Blue

AM-Amplitude modulation

EAE-Exploration and Exploitation

AMD-Automated mechanism design

ICS-Inertial collaborative social

VP-Virtual particle

CO-Code optimisation

FF-Firefly

DOP-Dynamic optimisation problems

Chapter 1

Introduction

The post millennium era has witnessed some of the most remarkable inventions in the modern human history. We are now living in an age that has transformed this world into a global village. The digital revolution is predominantly due to the advances in the integrated circuit technology, and this ongoing technological research has resulted in the mass production of economical memory devices, compact ultra high resolution photometric sensors, and multi-core embedded processing capabilities that have since targeted the demands of diverse consumers. Vision based applications are therefore becoming much more common in the domestic consumer market due to the reduced manufacturing costs.

“A picture is worth a thousand words”, perhaps when Fredrick. F. Bernard had written this famous quote [1], he might not have fully contemplated the technological unpredictability being experienced by the people of all ages in the 21st century. After the internet revolution, the human race is now entering into a phase of distributed intelligence, and imaging based automation. This thesis identifies the fundamental problems encountered during the object tracking phases in the digital moving imagery, and also proposes several remedies.

The system on chip (SoC) [2] and mobile processing technology interconnected with the high speed information gateways could be utilized to recollect diverse experiences, much needed in the safety critical processes and applications (e.g., driverless technology). Similar to the data mining processes that are applied in the computer science to understand data relevancies and dependencies, the iterative learning of the semantic knowledge (retrieved from the time evolving images) grants a robotic system with an alternative to the natural vision (which is normally taken as granted by the majority of us). But a compelling question in this regard is that why someone would program and train a computer to perceive surroundings by using visual influxes, and to act like a biological entity.

The answer to the above question could spark an ethical debate (which is not the subject matter of this thesis [3]). With the depleting of world resources, and rising inflation and living costs, it would be rather irrational for us to designate human jobs to the machines. However, computers could prove much more efficient in performing certain repetitive tasks without suffering from the boredom and fatigue frequently observed in humans. The field of computer vision is an evolving science that is attempting to pursue answers to the common paradoxes being experienced in the robotic vision (and particularly noticed during the diverse task assignments). However, it is usually a very rare occurrence for an artificial vision to work as effectively as a biological creation in the natural cluttered environments.

The nature of the tasks in the computer vision field is manifold; it could range from accurate and timely detection of various objects constituting an imaging frame to the recognition phases needed in the automatic resource allocations. A digital image is constituted of a gigantic arrangement of data values which are commonly referred to as pixels, the colour intensity of these minute pieces of information are therefore used to identify various objects in a scene. Hence, a region of interest (ROI) is a collection of pixels with varying colours and intensity values. Furthermore, the shape versatilities are also applied sometimes to establish identities of objects in a video frame. In computer vision, object tracking is referred to as a higher level task that identifies viable dynamic and moving clusters in a digital video. In practical terms, the object tracking is more than a simple pixel to pixel exhaustive search process. Therefore, some kind of hierarchy is generally needed to predict the object movements, so that a specific region of an image could be designated in time to conduct area confined searches, and to divert resources towards other lower level processes.

The main objective of developing an artificial eye for the scientific purposes is to deduce visual inferences that could be applied to prescribe accurate and meaningful system responses and behaviours in time. A navigating robot utilises tracking modules to plan its trajectory, and to avoid collisions in a cluttered environment. In medical sciences, both detection and tracking algorithms are executed (along with image processing routines) to outline shapes of diseased areas, and to trace the irregular movements of vital organs to aid physicians. Despite of the presence of some pessimistic views, an artificial vision is generally a cost effective and environmentally friendly way to monitor remote applications.

The natural world (in most cases) is much more dynamic than the majority of man made process controllers; this is mainly due to the fact that the operating environment and the configuration space of the system could deviate (in the course of a few frames) in a non-linear and unpredictable fashion. The observed volatility could be due to the changes in the lighting, reflections and shadows from other dynamic objects, and partial occlusions that usually invalidate structural information content of an object [4]. Furthermore, complications in the pattern matching algorithms usually make the simplistic detections much more cumbersome to be handled in a real time frame [5]; this non-differentiability gives rise to the false detections and causes instability. The core theme of this thesis is that the surge in improbable identifications could be contemplated by using non-deterministic and evolutionary mathematical routines. In contrast, the precision modelling within an analytical framework is generally much more prone to errors [6].

Figure 1.1: The Quantum Cloud by Antoney Gromley near Millennium Dome source [7].

An alternative to the common deterministic dynamics is portrayed in Figure 1.1; it delineates the importance of an organised random walk, and is a valid demonstration of the fact that many aesthetically impressive structures could be created by alternating roles, and through undergoing designated movements along various subsections of a solution space. The information content of the scene (Figure 1.1) has been perfectly preserved by selecting versatile step lengths and turning angles, and at the same time a range of organised random walk is also carried out to render the important background information. Similarly, it would be fascinating to study the consequences of implementing general evolutionary behaviours (in which transitions are composed of both social and randomised search velocities) in a visual tracking environment. This chapter emphasizes the need for adapting evolutionary trends in the object tracking algorithms, the aims and objectives of this research are identified along with the contributions and structure of this thesis. The key theoretical concepts are also introduced that contributed in the reduction of the algorithmic complexity in this thesis.

1.1 Aims & Objectives

The aim of this thesis is to develop self reliant, portable and light weight real time tracking and detection algorithms inspired by the natural expositions. A self reliant tracking/detection facilitates timely recovery (e.g., if an object track is temporarily lost in a sequence of frames) without elaborative initialization steps. The reinstatement of a tracker is of paramount importance for remedial actions required in the safety critical applications and in surveillance tasks. In order to addresses the video tracking problem, a bifold strategy has been adapted in this thesis. First the major disparities in the evolutionary literature are highlighted and explored with experiments, and a novel tuning free optimisation strategy is therefore proposed. Later on, the newly developed algorithms are applied to both natural and indoor scenes to expedite the tracking process. Hence after methodological testing, it is strongly/passionately concluded that some loose ends (in the evolutionary science) are accurately resolved in this report.

1.2 Background.

Image denoising, segmentation and tracking are all classed as inverse problems [8]. In inverse problems, only corrupted data (due to the sensor and medium imperfections) is available to develop an accurate model of the underline dynamics, and to get rid of the unwanted interferences. To define spatial transitions between frames, the observed velocity, acceleration and positional coordinates are all commonly used as state variables. The hidden information that we may need to retrieve could be composed of any other electronic signal (e.g., utterances at a microphone). Removing interferences from an image that has travelled through the communication technology is a further example of an inverse problem.

Non-linear transformations and related abnormalities (e.g., the changing position of light sources) further complicate the process of developing an accurate model of the dynamics, and therefore object recovery within two subsequent imaging frames becomes more problematic [9]. To address an inverse problem, and in order to discover a promising solution one has to deal with high data sensitivity causing illpossdness [10], and have to perform relative verifications that the solution is indeed unique in all sense [11]. Direct inversion has been studied in classical mathematics, but its applicability is limited to only a handful of applications [12], indirect inversion on the other hand is more costly and subtle procedure.

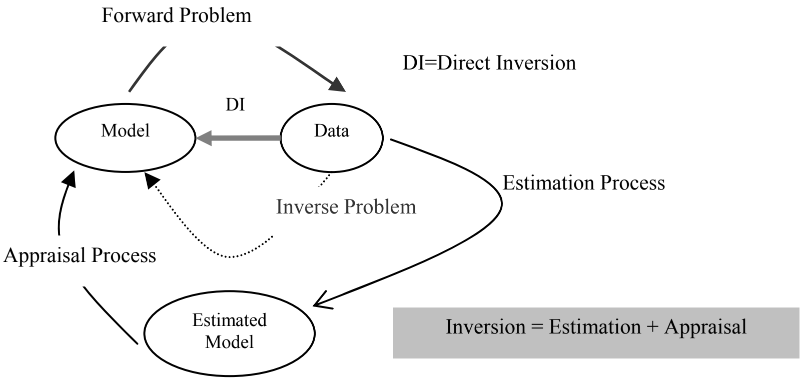


Figure 1.2: Graphical depiction and division into forward and inverse problems.

One rectification (to tackle the mentioned problems) is to smooth observed datasets by applying regularization techniques [13]. The unnecessary variations and inconsistencies in a digital image (and in other electronic signals) are significantly reduced during a regularization stage, and after performing direct inversions (Figure 1.2) true information is finally retrieved. The direct inversion is generally impractical in the computer vision algorithms because of the large datasets and due to the large variety of information.

The irrationality involved with direct inversion is dealt with by bringing in a notion of an appraisal mechanism [14], and therefore inversions are carried out in a more implicit manner. The inverse problems are therefore systematically tackled by a collaborative mechanism which breaks down the problem into two stages as represented in Figure 1.2. The estimation stage predicts a possible solution to the problem (using a plant model or a system dynamics), whereas the appraisal stage investigates this problem further and much deeper by evaluating the likelihood of the observed/measured densities against the vague projections made at an estimation stage. The underline aim is to establish the missing link by anticipating the discrepancy margins between the projected hypothesis or predictions and the objective conditions of optimality. Estimation-appraisal processes are more commonly recalled as data fusion filters (Kalman filter is a major example of this approach [28]). Such highly analytical schemes have celebrated huge successes in the industrial plant modelling, and abundantly applied in the manufacturing industry. However an amicable research question (remain unanswered) is to analyse with experimentation that whether precision controllers are in fact the optimal methodologies to handle visual disparity within images (in the first place), or are there better alternatives to perform these vision tasks outside the control science.

1.3 Environment mapping and computational intelligence.

Estimation process (as mentioned in Section 1.2) performs linear projections in the solution space. This modified shape of the search space is therefore expected to be much more confined and regular, so that appropriate local searches (e.g., Newton Method [15]) or gradient techniques (like the steepest descent and ascend [16]) could be applied for an optimal convergence. This iterative minimization or maximization process theoretically seems reasonably straightforward, but practical imaging scenarios are contaminated with non differentiability issues as shown in Figure 1.3.

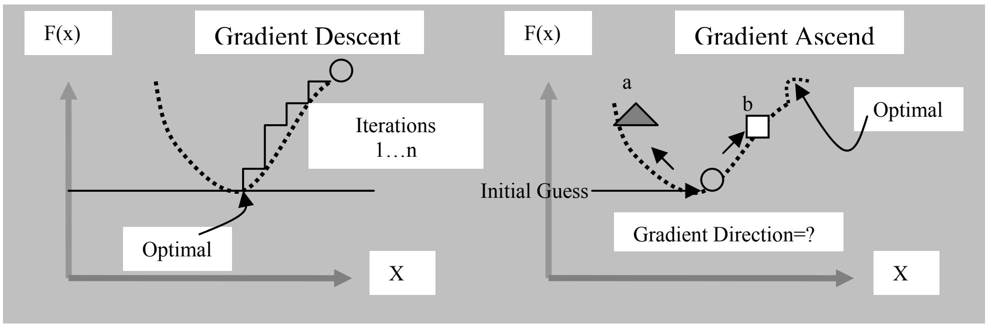


Figure 1.3: The importance of proper initialization in iterative solution searches.

Such problems have been reported in the leading computer vision literature, e.g., in the Level Set curve tracking [17], Meanshift and Camshift approaches [18] [19], sequential Monte-Carlo and particle filters [20], and also have been noticed in the recursive Bayesian Estimations [21]. The instability issue has been identified in Figure 1.3. Seeking optimal in this simple one dimensional problem appears totally circumstantial as distinct initializations (at points a, b) in the search space could lead to the discovery of only local best values. The convergence timing (to a feasible solution) is also very sensitive to this initialization and varies accordingly. In the gradient ascend scenario, the solution has minimal chance of escaping the valley (where the direction of the gradient could not be established), unless some kind of meta-heuristic is used to guide the solution into a feasible region.

Through employing more meaningful initialization stages, the number of steps required in the gradient descent scenario (Figure 1.3) could also be significantly reduced. By providing more realistic and multiple initial guesses, the prospects of finding an acceptable solution are generally increased. As diversity is an inherent part of the evolutionary trends, it therefore could also improve stability in the visual tracking procedures. The computational techniques address inverse problems by iteratively seeking better solutions; an alternative option to the traditional deterministic methodologies. The essence of which is in the underlying ability to conduct global optimisation, it therefore has much better potential to penalise the unwanted local traps (e.g., by using guided searches to escape local minima in a multi-modal cluttered environment). This comprehensive environment mapping is in principle very similar to the other well known scanning methods (e.g., conducted by the Doppler weather radars, laser scanners and in the modelling of submerged terrains using sonar).

The individual photons of electromagnetic pulses (in those cases) are replaced by the discrete elements which are known as particles. The relationship between the strength of population and convergence timings are important factors to solve computational problems; therefore, a significant portion of this thesis is dedicated to answer these questions. The movement of particles in the search space could be completely autonomous, or the trajectories may be imposed by the gravitational forces (a usual form of exertion by the peer particles). In the particle swarm optimisation (PSO) [22], the particle positions are therefore calculated using a linear combination of random and social perturbations, and the magnitude of velocity vectors are induced by employing tuning variables.

Some recent optimisation methods work on the principle of hierarchical division using elitism, which separates population into foragers and workers [23]. It is the role of the foragers to predict initial solutions, which are then iteratively refined by collating opinions of the rest of the population. In an ant colony optimisation (ACO) [24], the best solutions compete against each other by calculating strengths of the pheromone trails [25]. In bee hives, waggle dances are performed by the foraging insects to solicit, and the sole perspective of this act is to gather wider audiences and recruits [26], this environment is commonly referred to as combinatorial optimisation schematics in the leading scientific literature.

1.4 Thesis contributions

I. The deep-seated problems in the vision applications are discussed in this thesis. It is explained (by investigating 3 prominent tracking methods) that why tracking is an entirely different class of problem than the control theory (Section 2.5). This novel understanding of the vision problems could help to devise more feasible methodologies in the science of computer vision.

II. The partial function evaluation has proven a superior approach (in terms of both convergence accuracy and timing) than the traditional Bhattacharyya measure between two discrete colour probability densities (Section 4.2 and Figures 5.1-5.14) in tracking applications. By avoiding complicated data structures (matrices and associated linear algebraic operations), and portraying knowledge using simplistic constructs addresses the curse of dimensionality problem. Therefore, simpler and effective binary flags are used in the particle detections, and to associate measured data with the predictions.

III. Several weaknesses of the evolutionary and swarming methods are identified in this thesis (Sections 3.3-3.4). There is a significant amount of flaw and misconceptions when the role of an elementary search agent (particle) is defined in the evolutionary literature. In the view of this thesis, particles must be free in their searches and spatial correlations in order to compete efficiently against each other in a solution space (like real life hunters, and foragers). When the element of immortality is removed from the particle definitions, a much faster task oriented convergence is experienced. Several new particle paradigms were introduced in order to rectify the common misconceptions in the evolutionary literature (Sections 3.1-3.4).

IV. A novel stochastic particle swarm optimization (PSO) algorithm was implemented, and analytical comparisons are performed between the nostalgic and concurrent variants of PSO using evolutionary test cases. From these experiments, it was learnt that the best technique to solve the optimal trajectory problem is definitely a memory free approach (Figures 3.42-3.66).

V. A novel scale free method (radical search optimisation-RSO) was introduced in this thesis that efficiently solved a lot of complex evolutionary test cases with ease. Better convergence timing (compared to both PSO and Bat algorithm [27]) are witnessed using RSO, and in some cases the convergence time was decreased to $0.25 \circ t_{pso}$ (where t_{PSO} is the time the PSO algorithm has taken to converge with similar accumulation of errors), therefore the prospects of real time tracking are significantly increased (Figures 3.45-3.51).

VI. Heuristic and guided searches are applied in both detections and tracking of the objects of interest. By rectifying the unnecessary memory operations (a new technique was implemented that does not store the intermittent particle positions (Section 6.3 and Section 3.4) in the search space, a significant performance boost was observed in both particle swarm optimisation and Bat algorithm when the searches were motivated by the RSO. The evolutionary based tracking also worked much better than the Kalman filtering based approaches [28]

VII. Artificial BAT is a newly developed meta-heuristic optimisation technique, which in many cases has performed the standard PSO [29]. Both BAT and PSO methods are implemented to track a range of objects in Chapter 6. Later on, radical searches are applied to direct BAT and PSO algorithms towards an optimal region of interest (with an objective to reduce the algorithmic complexity). By applying the novel variants of PSO, RSO-VGS, and the scale free correlations, a frame rate of as high as 250FPs are observed in some of the most difficult test conditions (Sections 6.5-6.10).

1.5 Thesis Structure

Chapter 1

Readers are introduced to the significance of the computer vision algorithms, and the aims and objectives of this research are explored. After establishing a research background, the novelty of the adapted framework is presented. The major contributions and the structure of this thesis are also discussed.

Chapter 2

The research background is presented in this chapter. The tracking problem is formulated in an optimisation framework, and relevant discussions are presented for adapting a nature inspired methodology. Three historical tracking schemes are discussed, and the ramifications of using analytical mathematics to solve frame tracking problems (in the past) are highlighted.

Chapter 3

This is the methodology section, and constitutes an important section of this thesis due to the scientific and comparative analysis performed that also lead us to the development of a novel methodology The limitations of the gradient based methods are experimented using test cases, and later on strategic initializations have been strongly emphasised. The optimisation redundancies (e.g., in both PSO and evolutionary strategies (ES)) are mentioned, and computational bottlenecks (e.g., social calling) are highlighted for developing an expedited tracking scheme.

Chapter 4

The procedural flowcharts of various algorithms are developed and presented in this chapter. Detailed discussions are presented that establishes tracking within a dynamic optimisation framework.

Chapter 5

Several test benches are introduced in this chapter. The diversity of test cases is established by drawing movement graphs and corresponding patterns in the feature space.

Chapter 6

This chapter implements the previously developed novel ideas in Chapters 3 and 4, and experiments are performed using a wide variety of detection and tracking scenarios. A detailed analysis is conducted which first modifies the optimisation search basin using a nested radical approach, and later on, the penalising guided searches are applied to distinguish an object from the background clutter. A range of meta-heuristic algorithms (BAT, PSO and RSO-VGS) are programmed in Matlab, and applied to frame tracking problems, and the convergence characteristics are plotted in the graphical formats.

Chapter 7

This is the concluding chapter, and the results of our research are presented and discussed. Moreover some conspicuous research directions have also been proposed for a keen reader.

Chapter 2

Principles of Video Tracking

Kinematics is a branch in mathematics that explicitly deals with the geometry of motion [30]. In the historical computer vision applications, the three leading principles of kinematics have been extensively applied in the motion modelling of various objects of interest [31]. In contrast to the general mechanics, the applied forces causing spatial displacements are not explicitly calculated in the vision applications. Several branches of classical physics (e.g., kinetics and analytical dynamics) shed more light and focus on the evaluation of applied forces to prescribe desired movements. In engineering environment kinematics is applied in the precision modelling of electromechanical systems, which basically rely on a number of interconnected modules to perform a designated task. The coordinated and precision movements required in the manoeuvring and navigating robots [32], calculating the efficiency of high speed turbines and modern internal combustion engines are all prime applications of kinematics [33].

On the other hand, the Newton laws of motion are classical examples of the analytical dynamics [35], and are the main building blocks of many reconnaissance machines ranging from sea and land to the enduring designs needed in the space exploratory capsules [34]. Choosing a suitable coordinate system is a crucial first step in the subject of motion analysis. The newly formulated Hamiltonian and Langrangian frameworks are prominent attempts that try to reduce the dimensionality of problems for more robust designs [36] [37]. Whereby, many branches of classical physics explicitly apply Newtonian dynamics to define the trajectories of particles and matter undergoing transitions, a video tracking application is somewhat different class of motion problem. A change in dimensionality takes place when motion is captured onto a camera plane, regardless of whether it is mounted on a static or a moving platform. The transformation from a 3D coordinate system of the world to an imaging plane is a complicated and a delicate process, that generally results in the loss of information embedded in the real dynamical objects (Figure 2.1). Therefore in applications where reverse projections (2D-3D) are needed (e.g., to generate motion using electrical motors in robotics), the study of structural variations within a camera plane are mandatory. The vertical motion of an object moving in world coordinates using a static camera platform could only be analysed by studying the size of a moving object (shown in Figure 2.1). Real life objects often exhibit versatility in their motion trajectories, the movements could be confined to a specific plane in a Euclidean search space (e.g., traffic on an open stretch of road), or the patterns of motions could be completely randomised as observed in sports, and also when the behaviour of wildlife is studied under lab conditions. Therefore, a truly portable tracking algorithm is not context sensitive, and must be applicable in all scenarios.

.

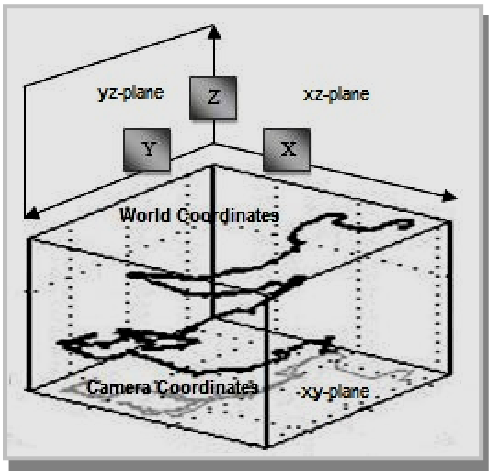


Figure 2.1: Projection of a 3d motion from the world onto (XY) camera plane.

Once a suitable image is captured onto an imaging plane and photometric sensor, the computer vision algorithms are then applied to pursue knowledge needed to facilitate a reverse projection (to generate the required effects as mentioned earlier). The timely selection of a closed and confined solution space is important for a tracking algorithm to produce a real time response. For a human observer, the video has to be presented at around 25 frames per second (FPS) in order to appear continuous and smooth. The real time convergences (to process frames at more than 25FPS) could be facilitated if the search variables (a design choice generally made by a computer analyst) are effectively reduced to address the curse of dimensionality problem [38]. Both video tracking in computer vision and classical Newtonian dynamics use vector formulations to represent motions.

In the evolutionary branch of mathematics, the search characteristics of agents are also defined by manipulating vectors in an algebraic formulation [40]. In cases where the capturing device (camera) is dynamic and jittering defining motions becomes much more cumbersome, as it is not normally possible to build a generic model addressing all possible displacements. The applications of video tracking are versatile and ranges from the trajectory planning in robotic application to predicting motions in the collision avoidance systems. The Mars rover is a mobile platform, and applies both binocularity and visual disparity to take advantage of the mounted robotic arm, and to plan its motion in the rugged terrain (see Figure 2.2). The rover therefore exhibits classical transformations in both forward (3D-2D) and in the reverse (2D-3D) directions.

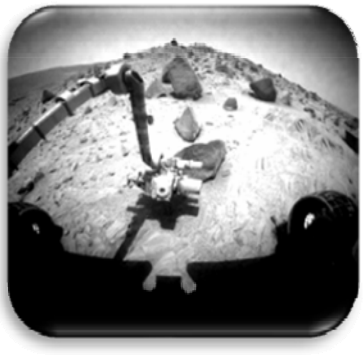


Figure 2.2: Disparity in trajectory planning through Boulders. Mars Rover Missions [39]

The real time response of a computer vision task is of paramount importance when the safety is a critical issue. An automatic driving assistance applies tracking to detect the symptoms of fatigue in drivers, and takes remedial actions to deal with the developing hazards [41]. The artificial learning environments apply both eye and body language tracking to develop an optimised lesson plan [42]. The Tesla automotive self driving technology utilizes radar and camera inputs to detect stationary and dynamic objects, and for mapping their surrounding landscape [43]. The tracking of microscopic organisms provides the biologists with vital visual clues to understand complex behaviours [44]. Tracking the movements of insects and reptiles in scientific labs is a useful technique that helps to reduce the harmful impacts of parasites to boost crop productions [45]. The flow of traffic could be automatically controlled if blobs of vehicles are detected and analysed using computer vision technology. Modern remote surveillance systems also apply visual detectors to identify intrusions [46].

2.1. Aims and Objectives.

The aim of Chapter 2 is to introduce to the readers the fundamental building blocks of visual tracking systems. The three most prominent principles of tracking are discussed in Section 2.5, and the prospects for further improvements will also be discussed. The scope of research will be briefly discussed, and the tracking problem is established within an optimization framework. The discussions presented in this chapter also pave the way towards developing a new evolutionary computational tracking methodology.

2.2. Scope of research in imaging.

The scientific research in the context of image science is perpetually vast, and the scope of this horizon span from the clean fabrication labs that help to develop the latest sensor technology (CCD, CMOS and CIS etc. [47]) to the understanding of semantics and structural information contents of a digital image. The storage and transportation using information retaining compression, digital filtering and the identification of dynamical structures are some other related perspectives of the image based science [48]. The removal of interferences using structure preserving denoising and image editing/enhancement routines have been the main focal points for many researches in this subject. Images and three dimensional graphics are extensively applied in prototype developments in manufacturing engineering using CAD software, and imaging information is also applied to guide modern CNC machines [49].

The image enhancing routines facilitate retrieval of the structural information content needed in the forensic labs and to allocate resources. The optimal segmentation process is scientifically learnt in medical sciences by examining a large database of images to facilitate radiologists [50]. The expert medical imaging systems use advanced data mining techniques to analyse hundreds of thousands of patient records and images, and therefore assist physicians in their diagnostics. The tracking or sequential object segmentation is a higher level task, and the aim therefore is to identify feasible regions in an image to conduct further lower level analysis. The dynamical structures are used in the broadcast technology to save the bandwidth of a communication medium (e.g., by broadcasting only the dynamical areas of an image) [51]. In this age of cybernetic intelligence, the human race is entering into a new era of distributed and biomimetic robotics which apply visual inferences in a collaborative fashion to perceive the surrounding environments [52].

2.3. Tracking as a both space and time problem.

Tracking an object of interest in moving imagery is a spatio temporal problem, hence; suitable measures in both space and time are needed to increase the algorithmic efficiency. The observed displacements are then further analysed along the discrete time dimensions t_n and $t_{n+\Delta}$ in order to perceive motion patterns. The video refresh rate of the capturing device is a relevant issue in determining motions, as intuitively larger displacements $\Delta d = s_{t+\Delta} - s_t$ are predicted in low speed cameras (where $s_t, s_{t+\Delta}$ are positional coordinates of an object of interest (OOI) in a R^n dimension space at times t and $t+\Delta$) between frames. High speed cameras are rarely deployed in general tracking applications, and are usually reserved for the precision industrial and military applications.

An important step in developing an automated detection and tracking system is to create an object identifier in a feature space (e.g., RGB colour space). The interconnected object regions $\Omega_{1..n}$ are therefore identified using stored patterns expressed in a mathematical format (Figure 2.3). It is possible for the region based features to be described as parametric Gaussian or through non-parametric histograms. The illumination impacts are generally reduced by allocating fewer weights to the boundary pixels of an object. The object clusters in an object may occupy distinct positions in the RGB feature space (as shown in the middle section of Figure 2.3), but as they are associated with the similar object, they are all encoded mathematically as parts of an objective pattern.

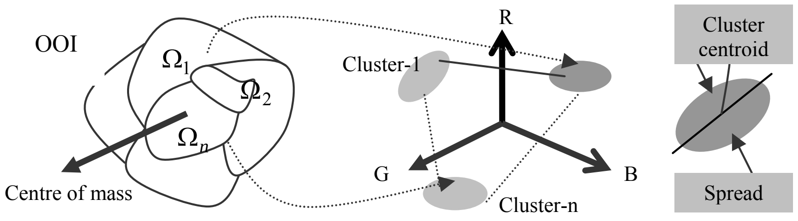


Figure 2.3: Various connected sub regions Ω_n of OOI with distinctive clusters in the feature space.

A non-parametric colour density also stores cluster centroid and relative spread of measurements using data histograms. In Figure 2.4, a general tracking and detection scenario is presented. Detection and tracking often work in collaboration, and are associated processes in vision technology. In some literature detection and tracking phases are also referred to as unique and distinct phases. In video tracking detection is usually carried out during an initialization stage.

Tracking is performed on a confined and narrowed down space (with an objective to reduce the associated computational costs). Whereas, detection explores all relevant areas of an image and conduct much broader attempts to match stored templates with the candidate model. At a tracking stage where the required condition of optimality (e.g., colour pattern matching within a tracking window) has not been observed, a detection phase normally takes over to reinstate tracking algorithm back into a feasible search space (shown in Figure 2.4.). The anticipated line of research in this thesis is based on the fact that if detections are robust and timely, there is no particular need to confine tracking (also known as limiting the search basin), and could supplement difficult tracking phases by dynamically adjusting the current basin for a robust recovery. The detections performed by particle centred approaches are multiple solution techniques that are inherently designed to conduct diversified detections.

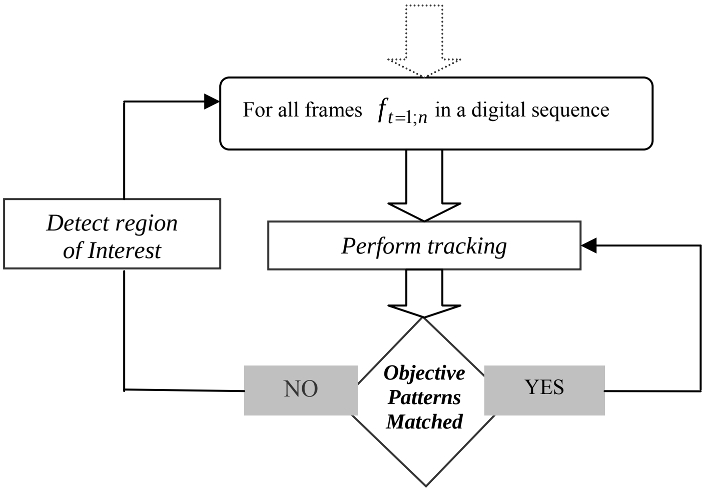

For all frames $f_{t=1;n}$ in a digital sequence
Detect region of Interest
Perform tracking
NO
Objective Patterns Matched
YES

Figure 2.4: A generic object tracking and detection scenario used in the computer vision.

2.4. Tracking as an optimization process.

The objective of video tracking is to identify a dynamical object, and to consequently reveal its positional coordinates as it drifts away from an initial known position in an image. Therefore various dynamical objects are also identified sometimes through trajectory classifications in a search space. At any moment in time, the state vector of an object could comprise of its movement components, that usually consists of previous known position (p), velocity ($\dot{p}$) and acceleration ($\ddot{p}$). The future positions of objects in the Euclidean space are therefore represented by the positional derivatives with respect to time as shown in Equations (2.1) and (2.2).

$$P = (x_p, y_p, z_p) = x_p \hat{i} + y_p \hat{j} + z_p \hat{k} \tag{2.1}$$

$$V = \lim_{\Delta t \to 0} \frac{\Delta p}{\Delta t} = \frac{dp}{dt} = \dot{p} = \dot{x}_p . \hat{i} + \dot{y}_p . \hat{j} + \dot{z}_p . \hat{k} \tag{2.2}$$

Here (x_p, y_p, z_p) are the associated Cartesian coordinates, and i, j, k are the unit vectors of this coordinate system. The components of velocities $\begin{bmatrix} v_y \\ v_x \end{bmatrix}$ are then calculated, and are applied to refresh the object coordinates along time using Equations (2.1)-(2.2).

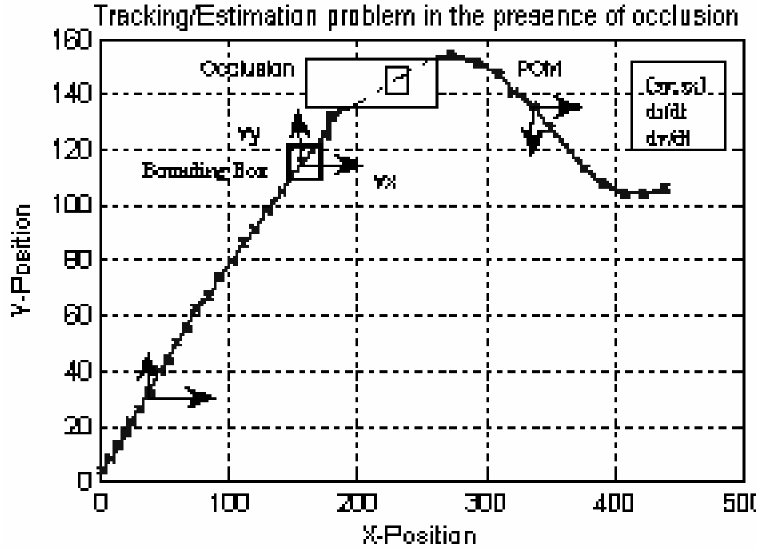


Figure 2.5: Tracking problem in the presence of missing data (occlusion) in a video sequence.

The path of one translating object is represented in Figure 2.5. The trajectory comprises of various movement phases, including linear displacements, and also some curvilinear motion phases are present in this particular case. At points of manoeuvrability (POM), it suddenly accelerates away, and changes its direction of travel. Therefore, no patterns are immediately observable within the bounding box (representing an object of interest), hence the tracker diverges away from the true solution. It could be seen in Figure 2.5 that the tracked object may undergo complete occlusion (where the trajectory information is missing); the dimensions of the bounding box define the basin of searches in this particular experiment.

During non-linear phases, the tracking problem is more challenging, as it becomes very difficult to mathematically describe all associated motions using a linear Gaussian model. Other problems include partial and complete occlusions, and estimating the next point in their trajectories essentially becomes a black-box problem (due to the missing information for a length of time), these scenarios have been identified in the relevant research literature [53], and broadly speaking, different strategic variations in tracking are attempts to address these issues.

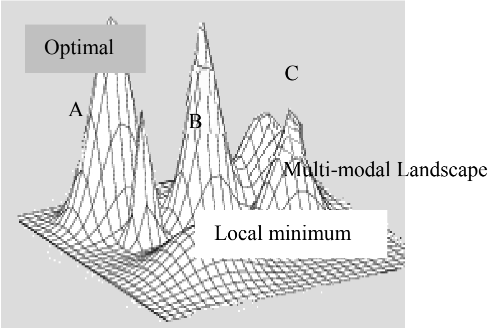


Figure 2.6: Landscape of an optimisation problem [54].

Figure 2.6 is a multi-modal landscape that constitutes of many local minimisers, and the main optimization objective is to identify the optimal peak, and to facilitate conditions so that the solution is guided towards the optimal region of interest. The underline task of an optimisation algorithm could therefore also constitute of timely identification and ranking of all admissible local solutions as well along with revealing the global optimiser.

Evolutionary optimization is an emerging branch of applied mathematics that is proving much more effective in addressing black-box problems [55]. Mathematical optimization is the science of minimizing or maximizing a function value over a finite space in which the problem is defined. Optimization problems could be befitting application of the proverb ‘courses for horses’, therefore, the optimisation techniques are frequently varied in both space and time dimensions, and an optimal strategy depends on the shape and terrain of the problematic landscape. The landscapes of optimization problem are the functional variations over the domain of operation. Therefore, the aim of an optimization run is to select a particular set of independent variables (coordinate selections in the tracking applications), that yields the maximum objective values (a maximization process).

The objective task in the computer vision applications could range from segmenting contours to shape alignment routines required to establish identities. The ‘minimizing or maximizing’ of an energy function refers to an optimisation stage in which the detected features are matched to the stored values in a spatio-temporal domain. In minimization problems (for example), it refers to a situation when an evolving curve encapsulates an area of an image, so that all pixels falling within are probabilistically more relevant to the region under observation. In Figure 2.7, the minimum energy state is achieved when evolving curve breaks and merges and formulates a closed boundary around the bacterial organisms, hence the total length of the segmenting curve is reduced in the optimal state.

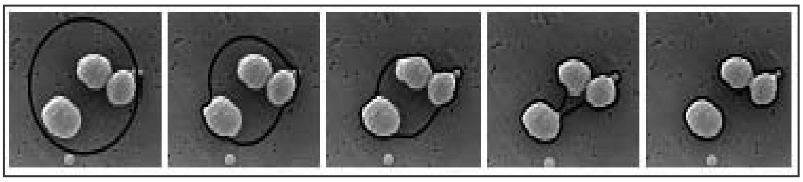


Figure 2.7: Energy minimization process used in curve segmentation [56].

Whilst segmentation is the process that differentiates an object from the background in static imagery, tracking is the prolongation of this knowledge into the temporal domain. Therefore, the strategic know-how of the landscape is seemingly crucial for domain reductions in order to converge in a real time. The determination of a pixel neighbourhood is also a domain related problem (as shown in Figure 2.8), many tracking and segmentation implementations use a variety of pixel connection schemes (e.g., 4 or 8 connected pixels to identify the intensity variations) to analyse changes at an observation position (x, y) as shown in Figure 2.8. In image denoising, the intensity value of a pixel at position $\begin{bmatrix} x \\ y \end{bmatrix}$ could be selected as an average of the local neighbourhood.

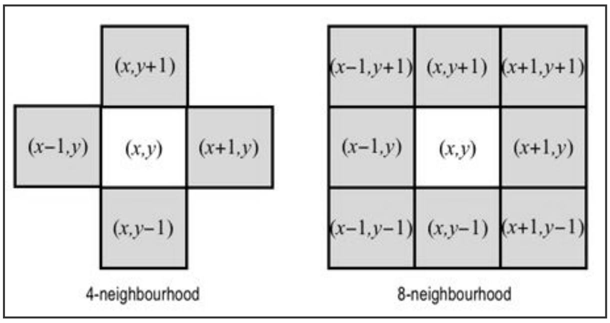


Figure 2.8: Two connected pixel schemes commonly used in the imaging literature [57].

Evolutionary mathematics addresses the issues discussed earlier by iterative selections of the neighbourhood regions. The variable searches could prove particularly effective during the occlusion phases (Figure 2.5), where classical methods usually fail to converge. In general analytical optimization techniques, when no valid measurements are observed the algorithm predominantly relies on the deterministic models (in a hope) to find an optimal. The analytical framework sometimes generates an unwanted complexity as the stored plant dynamics may not be applicable when sudden velocity changes are observed. To compensate in the difficult tracking scenarios, this thesis applies parallel tracking hypothesis using multiple solution techniques (also known as combinatorial optimization scheme).

2.4.1. Optimisation-An act of steering.

Norbert Weiner was a well known cyberneticist and a renowned scientist for his work in the fields of control and communication technologies, his publications and quotes (‘optimisation is an act of steering’) provide a common platform for both neurophysiologists and engineers to understand generic automatic control problems [58]. The goal of Norbert’s research was to gather natural inferences from the complex motor solutions in a human body, and to apply those later on in other complicated engineering scenarios. Following Norbert’s guidelines, Nikolan Bernstein also concluded in his experimental studies that the artistic motor perfection and control (in the biological systems) is not entirely based on the precision and tighter control laws (as previously thought), but the key is in the precise application of the motor variability that applies diversity to perform the same task in many different ways [59].

This diverse control (for similar end results) is due to the multiple moto-neurons synopsing on the same muscle, a conceptual portrayal is shown in Figure 2.9. The control policy consists of free-scale motions (isotropic regions) as well as correlated and tighter stages of control policy. A human body performs similar sequence of precision-tighter-loose-tighter-precision controls by flexing and tightening of muscles.

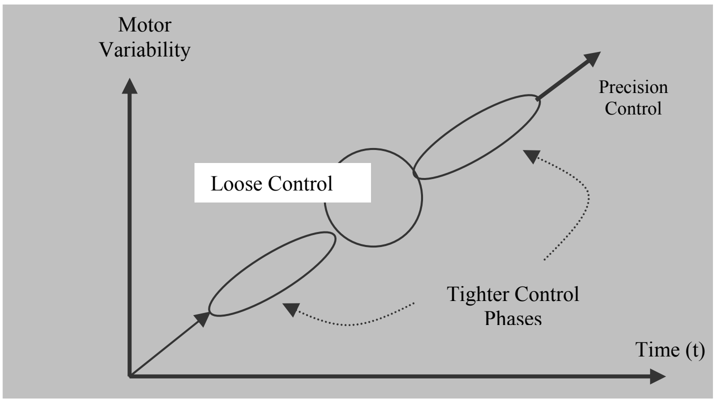


Figure 2.9: Control variability in repetitive assignments of a task in a human body.

The Bersteinian theory in general and the degree of freedom (DOF) problem [60] in particular are the foundation stones for the strategic particle behaviour in this thesis.

According to the DOF problem, there are multiple ways in which humans and animals conduct their bodily movements to reach to a similar goal and objective (i.e., an element of repetition without repetitions is actually present in these situations). Thus, by implying a loose optimization stage with diversified particle motions, the search space could be more effectively sampled than the frequent application of much tighter deterministic drifts (e.g., used in the historical particle filtering methods).

There could be all sorts of related reasons/justifications to adapt a nature inspired approach in this thesis. Dr Wayne Dyer *(1942-2015),* a renowned spiritualist, naturalist and a modern day philosopher has mentioned in his book “Inspiration: Your Ultimate Calling” [61] the mysterious and magical journeys of the monarch butterflies, and their prevailing navigational intelligence during migration trips are remarkable events. The round journey from Brazil to Nova Scotia (Canada) of the fragile monarch butterfly (through varying gusts of wind and atmospheric fluctuations) is fascinating for both evolutionary mathematicians and engineers alike.

Despite the volatile physical capacity, and the density of brain not much wider than a normal pin head, the Monarch’s journey is an ultimate call from the nature (in the view of Dr Dyer’s book) and an explicit demonstration of a natural distributed intelligence. Many evolutionary optimisation methods also rely on the distributed intelligences in order to minimise or maximise rewards [62]. The collaboration among social insect boost their chances of survival in the natural world. A lot more caution however is needed when social gestures are applied in the computational grids. It would be interesting to understand the role of social calling in this thesis, and to analyse the corresponding impacts (in terms of convergence timing/accuracy) for several test functions. One main reason for dealing optimisation in slightly differently context is due to the fact that the natural conditions are generally more detrimental than the artificially simulated environments.

2.4.2. Essential properties and categorization of tracking algorithms.

Sections 2.3 and 2.4 focused on some fundamental characteristics of a video tracking problem, and one possible resolution is to address tracking issue within a combinatorial optimization framework. Therefore, the most relevant path of objects is chosen from a list of discrete choices, and this selection is done by establishing relationships using strategically spread particles in a solution space. The aim of tracking is to match the dynamics of a bounding box (an operational basin, or a contour) to the kinematics of an object undergoing transitions in a world coordinate system. The motion of objects could be of predictable nature and type, as some movements are cyclic and more recurrent in nature. On the other hand, many observed movements are relatively randomised in nature, and thus are rather difficult to be described in an appropriate mathematical format. The moving objects also have varying degrees of freedom impacting the efficacies of the tracker.

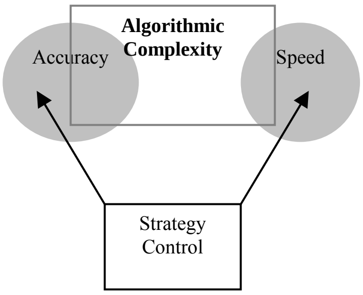


Figure 2.10: The strategic controller’s responsibility in order to assign a balanced tracking policy.

In order to achieve a real time solution, a basic necessity however for any tracking algorithm is to possess the inherent ability to alter the algorithmic complexity (as shown in Figure 2.10). The selection of a right balance between accuracy and the speed of convergence is pivotal for many tracking problems. The scheme shown in Figure 2.10 is a centralised control method, and a strategy control module is responsible for declaring the desired control policy. During stages of manoeuvrability, it may therefore be more efficient technique to put more emphasis on the speed of convergence rather than calculating precise boundaries of an object of interest.

The background subtraction formulates clusters of knowledge [63], and performing analysis using scale space is relevantly less complicated localization method, and could also help to escape local traps [64]. As a matter of fact, in most type of linear motions with limited degree of freedom such methods could generate a real time response. In curve based tracking, the aim is to adapt the shape and length of an evolving contour to address the translational movements and relevant scale changes of a region [65]. The shape of an object is also matched sometimes with the stored priors for an identification purpose [66].

In many tracking situations, the mean and spread of the object like clusters are deemed sufficient to analyse a basic imaging frame. An ideal tracking algorithm must be well aware of the degree of freedom of an object, and must have recovery techniques in place to adapt to a changing environment. The reactivity of tracker incorporates both spatial and feature based changes. Therefore, to attain the desired level of stability in complex search domains, a pattern prediction and updating algorithm could rectify clutter. Some brief definitions of essential properties of a tracking algorithm have been presented here, and would be explored further in Chapters 4 and 6.

A. Predictability: To incorporate balance between predictions and measurements.

B. Cyclicity: To save computational costs by learning underlying motions.

C. Reactivity: Alters the models in cases of changes due to reflection and refraction.

D. Accuracy: Dynamically adjust precision and computational complexity (Fig 2.10).

E. Stability: To handle varying conditions, e.g., occlusion handling.

F. Recovery: In cases of lost frames, the ability to successfully detect the region.

There are several distinguishing merits that could be used to classify the object tracking algorithms. It is important to establish the core differences among various tracking approaches so that a better understanding of the problem could be developed. Several authors have simply used nomenclature to categorise (without establishing the similarity/uniqueness observed at the operational level).

Colour is an excellent tracking feature due to the underlying properties of invariance to both rotation and scaling [67]. Object represented through primitive shapes has also been studied extensively, but are considered as stable features only if the structure is rigid [66]. Categorizing shapes in an image involves studying the inflection points, e.g., the calculation of corners and gradients. Therefore, in subsequent frames, the tracking simply becomes a frame correspondence problem. Relating a set of point $S(p)_{t-1}$ to $S(p)_t$ can be an extremely difficult task to be handled in a real time, e.g., in the optical flow method the proximity information is applied to allocate weights using the Euclidean distances calculated at various points in the search space [68].

Several frame correspondences (along the time dimensions) are used in the multiple hypothesis trackers [69]. The aim of the multiple region tracking is to generate a connected graph of movements; this is achieved by associating various trajectories to the corresponding objects in video frames. To rectify the correspondence relating issues, smaller structure like edgelets and primitive shapes are used to analyse motion in related configuration space [70]. Shapes could also be represented through silhouettes and closed contours; the level set representations of evolving shapes are higher dimensional surfaces used to make the curve tracking (merging and splitting processes) processes much easier to handle [71].

The process of shape matching could be very extensive due to the affine transformational routines needed to match with the stored templates [72]. These enhanced shapes are then used in the automatic recognition of the objects (e.g., face recognition). One possible appearance based categorization will be briefly mentioned in this section. Rather than based on the principle or a particular methodology, the appearance based classification uses the colour density and shape orientations as parameters for the tracker classifications, such distinguishing merits are portrayed in Figure 2.11. To distinguish among different objects, both shape and colour densities are therefore encoded in a feature space. In contrast to the global density models (e.g., in an optical flow technique) a local method only formulates and stores the colour variation of a region of interest. Similarly, the shape of an object could be represented by higher dimension techniques using level sets or by storing the corners and edges based information using gradient vector field.

Furthermore, as the shape of an object is variant to different affine transformations (in contrast to the colour representations); it is also a common practise to use both colour and shape of an object as a penalising force. However, as mentioned earlier in this section, there are many possibilities in which a distinguishing criterion could be established. Any keen reader is strongly advised to review these literatures to gain further insight into this categorization problem [73] [74].

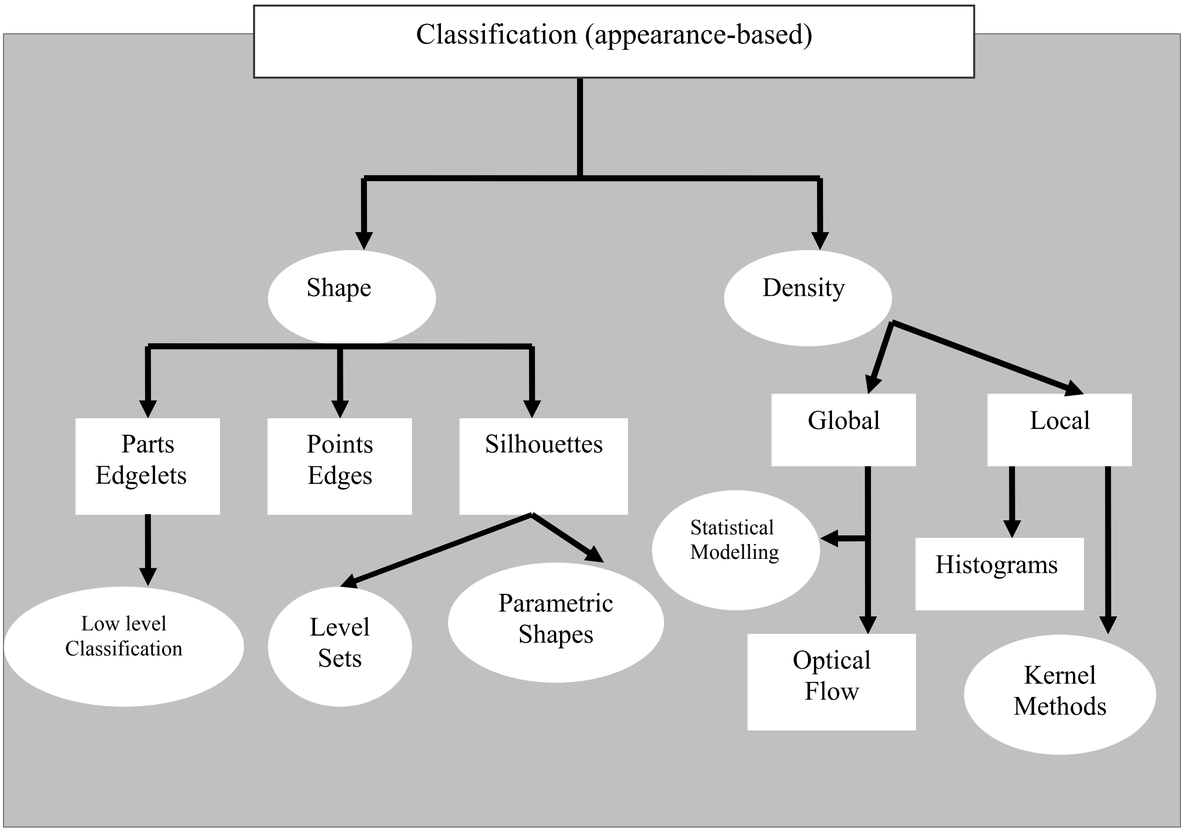


Figure 2.11: Classification of tracking algorithms based on appropriate feature selections.

2.5 Principles of video tracking

In this section three main tracking methodologies will be discussed. The first two approaches apply temporal propagation of conditional densities to identify the most probabilistic object position in a digital image. The third technique exploits gradient information to climb to the peak of the density.

2.5.1 Recursive Bayesian estimation (RBE) in tracking applications

Life is full of uncertainty and unpredictable happenings. According to the frequentists, the probability or likelihood of an event *'E'* taking place could be established using the past observations and history. As an example, let's imagine a group of mountaineers are planning to climb a tropical peak; they want to depart with the appropriate climbing gear for the dry conditions. A frequentist usually prefers to utilise historical data to deduce that whether a particular day would be a rainy day. The frequentist's approach is a good prediction tool, but in not sufficient on its own to accurately solve this problem at hand. Using advanced measurement tool like altimeter, atmospheric pressure gauges and isobars the problem could be more precisely addressed. This is the core concept of the recursive Bayesian estimation (RBE) [75]; the belief is updated on the arrival of latest information. In Bayesian inferences, the likelihood of an event (e.g., a rainy day) is represented by $P(e|k)$, where P is the probability of an event taking place in the light of a specific body of knowledge and observations *k*.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{2.3}$$

The Bayes's theorem [76] in Equation (2.3) was proposed by the *Rev.* Thomas Bayes (1701-1761). It is used to calculate the conditional probabilities without explicitly knowing the joint probability distributions $P(A,B)$. In Equation (2.3), $P(A)$ and $P(B)$ are the independent probabilities of the events A,B, whereas, $P(A|B)$ and $P(B|A)$ are the conditional probability density functions. We are interested in calculating the probability of an event A taking place when B is also true using the reverse conditional probability $P(B \mid A)$ (which relates past observations of B taking place when A is found to be true as well). With slight modification, a more practical format (suitable using a measurement context) could be written as in Equation (2.4).

$$P(X \mid Z) = \frac{P(Z \mid X)P(X)}{P(Z)} \tag{2.4}$$

Equation (2.4) reorganises the Bayes theorem in terms of a hypothesis $X = \{x_1, x_2, ... x_n) \in R^n$ in a multi-dimensional space R^n. In tracking, $'X'$ (a state vector) is composed of n dimensional positional coordinates, velocity and acceleration of a moving object at time t, $P(Z)$ is normalization constant (the probability of an observation in an image space which is a constant), and $P(X)$ is the prior probability calculated by applying Newtonian dynamics using previously assigned state vector and the known position at time $(t-1)$, $P(Z \mid X)$ is the measurement density which is used to link observations with a particular event.

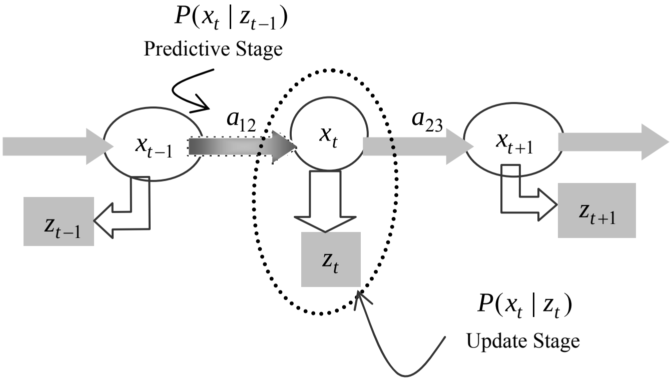


Figure 2.12: Graphical depiction of a recursive Bayesian estimation process.

If Equation (2.4) is recursively applied (as in Figure 2.12) over a discrete time period (assuming that all future and past states are drawn from a normal distribution), then such RBE is equivalent to the standard Kalman filter [77]. Thus, posterior or conditional probability distribution of a future state $P(B \mid A)$ largely depends on the current state, the state transition and the observation model, and is usually independent of any previous state. In Figure 2.12, a state x_t at time instant t is therefore dependent only on the state x_{t-1}.

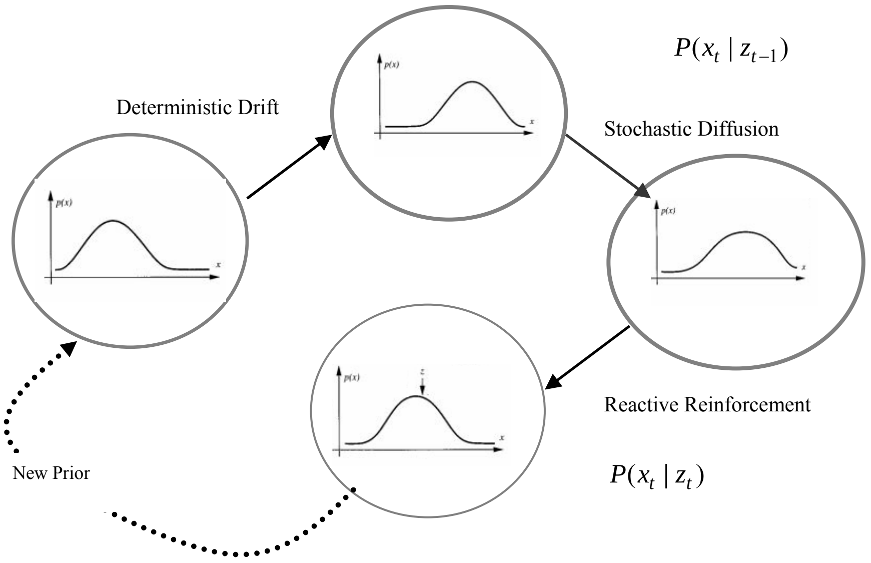


Figure 2.13: Reshaping of a Gaussian pulse in recursive linear Bayesian estimation processes.

Under uncertainty, the covariance of posterior $P(x_t \mid z_t)$ is enlarged, this is the theoretical reasoning of the introduction of the Kalman gain, which uses the residual covariance (error/distance between predicted and measurement state vectors) in order to update the posterior covariance model. Generally, any recursive linear estimation filter could be represented by the stages shown in Figure 2.13. On the arrival of a new video frame f_{t+1}, the previous posterior density becomes a prior.

A recursive linear estimation filter is generally comprised of three distinct processes as shown in Figure 2.13. During the first stage, the density x_t drifts bodily, and translates under the effects of a deterministic component composed of its own dynamics. Later on, the stochastic component of this transition spreads the Gaussian pulse increasing its uncertainty, and finally the density goes through a fine tuning stage known as a reactive reinforcement (where the distribution is convolved with another obtained through the measurement process, and generates a new density. Mathematically, the process is normally written as [see 78].

$$P(X_t \mid Z_{t-1}) = \int P(X_t \mid X_{t-1}) \cdot P(X_{t-1} \mid Z_{t-1}) \cdot dx_{t-1} \tag{2.5}$$

$$P(X_t \mid Z_t) = \frac{P(Z_t \mid X_t) \cdot P(X_t \mid Z_{t-1})}{P(Z_t)} = \alpha P(Z_t \mid X_t) \cdot P(X_t \mid Z_{t-1}) \tag{2.6}$$

The above Equations (2.5)-(2.6) represent the two step estimation process. Equation (2.5) states that the prediction density is the integral sum of the products of state transition models and the prior posterior calculated during the time step $(t-1)$. During the next stage, the posterior density is calculated by normalizing the product of the current likelihood and the prediction density as expressed in Equation (2.6).

2.5.2 Critical Analysis of RBE.

RBE is intensively used in many video tracking applications, and suffers from both stability and convergence timing problems. RBE could also be anticipated from the viewpoint of the Bernsteinian theory (Section 2.4.1). The control sequence in RBE is rather confined, and relies heavily on the plant dynamics which is difficult to be validated in the tracking applications. Similarly, the stochastic diffusive component has its own limitations (due to the reason that a Gaussian pulse has to remain a Gaussian) during the state transition in the standard Kalman Filter. One interesting thought in the course of this analysis is that the tracking applications do not require stringent control of dynamics as no safety fears are generally involved (e.g., required in dynamical control of an aircraft using ailerons or tail fins). Hence, in the opinion of this report, the deterministic components, and therefore the motion modelling is less crucial in tracking applications.

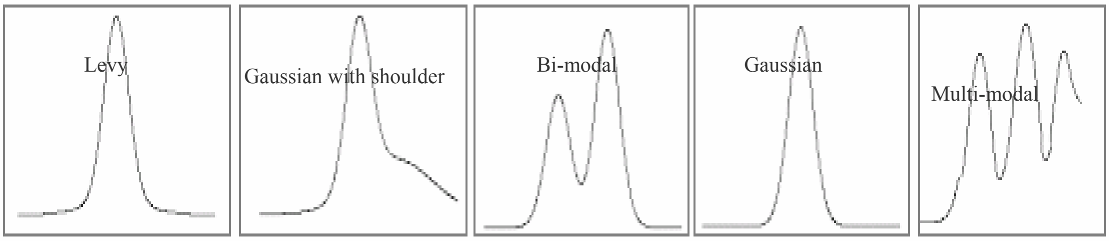


Figure 2.14: Various diversity models which could be employed in tracking scenarios.

Therefore, a more feasible approach is to use a whole variety of bodily drifts to sample the search space. As a single solution based approach, the standard Kalman filter/ RBE is limited in its applications. Alternatively, as shown in Figure 2.14, the multiple solution approach is much more recommended methodology to address unpredictable and larger variety of motions (of a region of interest). In Section 2.5.3, the limitations of recursive Bayesian estimations are overcome by initiating a multiple solution approach, known as the Monte Carlo sampling.

2.5.3 Monte Carlo Sampling (MCS).

According to historians, the name Monte Carlo initially emerged from a course of leisure testing conducted by a statistician at a famous casino in Monte-Carlo/Monaco [79]. It is believed that he took a series of repetitive measurements on several roulette machines to discover any possible hidden bias and a concealed probabilistic selective mechanism. During a fair trial, chances of winning and loosing games are equivalent, and therefore gathering repetitive samples for a reasonably longer period of time could reveal the shape of its underlying distribution (e.g., if it has a Gaussian hump, or is uniform etc). Alternatively, if the shape of a PDF is known, then the weighted samples drawn from it could be used as its representatives, and underline changes in distribution are reflected by using such samples in both space and time.

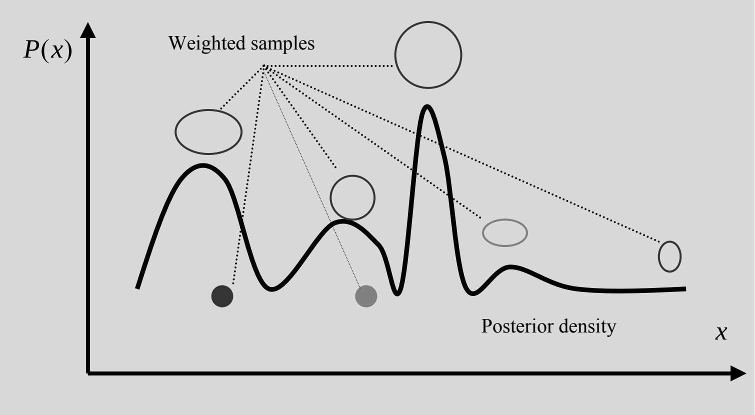


Figure 2.15: A set of sample points are used to represent a complex multimodal distribution.

In Figure 2.15, the blob sizes are used to allocate the importance and weights to various samples π_j, which are then used to construct the observation density $p(z \mid x)$ by integrating the measurements of several individual elements $s^{(j)}$. In the factored sampling, an unknown probability density function is approximated by mathematical convoluting the two known distributions $g_1(x)$ and $g_2(x)$ as shown in Equation (2.9), and the process could be represented graphically as in Figure 2.16.

$$f(x) = g_1(x).g_2(x) \tag{2.9}$$

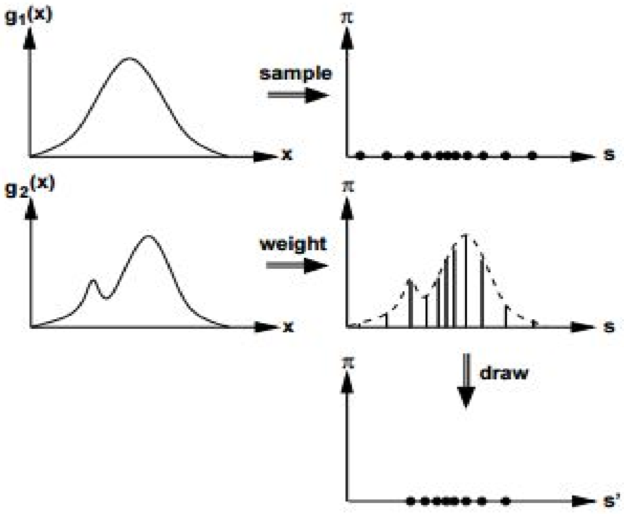


Figure 2.16: Factored sampling using two separate density functions [80]

If 'X' are the space parameters where a sample distribution x is defined, then in the first instance, a set $s^j \in \mathrm{X}$ is sampled randomly ($'N'$ times) from the density shown on the top left of Figure 2.16. During the second phase the weights of the samples are calculated according to the following formula [81].

$$\pi^{(j)} = \frac{g_2(s^{(j)})}{\sum_{i=1}^{n} g_2(s^{(j)})} \tag{2.10}$$

Equation (2.10) states that the weight of any particle $\pi^{(j)}$ is determined after normalizing its fitness likelihood measurement (shown on the top) with the overall observation density $g_2(x)$ of n particles in the population. Generally, particles are represented by multi-dimensional vectors, which is also used to stipulate their individual state vectors (positional coordinates, velocities and accelerations in a coordinate space), and associated importance weight i.e. $\{s_k^n, \pi_k^n\}$. Here the subscript $'k'$ is used to indicate the particle's own state in relevance to the state of the system.

$$S_k^n = A[S_{k-1}^n] + B\omega_{k-1} \qquad (2.11)$$

In Equation (2.11), a particle undergoes a state transition based on a known deterministic component $'A'$ representing the system/plant dynamical model, and $'B'$ is a multiplicative factor that is used to adjust the noise covariance.

The effects of this dynamics have been portrayed in Figure 2.17. The blob sizes are used to represent the relativity of the individual measurements. A state vector of any particle is composed of its position in the search space, and the associate weight matrices $\begin{bmatrix} S_k \\ \pi_k \end{bmatrix}$. The bodily shift of the particles is shown in the middle section (with a broken line), where the particles are translated in the search space by implementing Equation (2.11). Finally, the re-sampling process is applied in which less important particles are pruned out of the system and new ones are generated in the feasible areas (the bottom row in Figure 2.17).

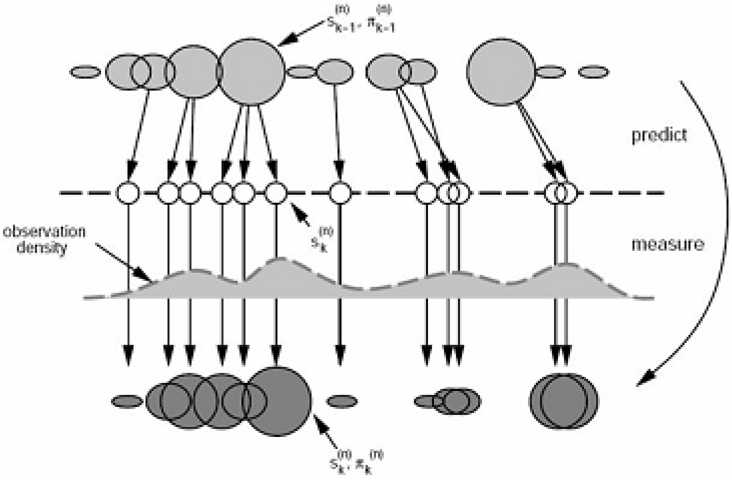


Figure 2.17: The sequential importance sampling in the prediction based systems [82].

In order to improve particles effective rate (η_e), only those particles with appropriate weights progress into the next prediction stages. In the spatial-temporal domains, particles need to be deployed at strategic location in both space and time with a prospect for a rapid convergence to a global optimal solution. During the transitional or bodily shift, some particles diverge further away from the feasible areas, and therefore become computationally ineffective. These particles are also phased out during the next stage of simulations as shown in Figure 2.17.

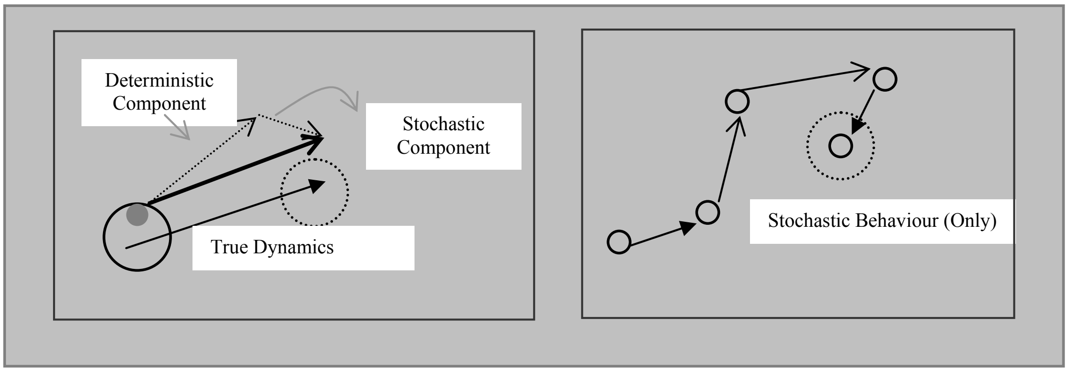


Figure 2.18: An initial concept of the virtual/sleeping particles (which do not need memory).

The main concerns (flaw in the view of author) in MCS are the associated computational overheads in determining the transitional energies of the particles. This idea can be conveyed as in Figure 2.18, where accumulating errors drift a solution away from the true dynamics, and even applying a stochastic velocity component (shown on the left of Figure 2.18) has failed to locate the true coordinates of an object. Instead of selective or rejection samplings, which are contradictory phases in the view of the author, a more indulging scheme could be based on the autonomous particle behaviours. The free-scale behaviour of particles would be studied in Chapters 3-6.

The swarming behaviour is an alternative nature inspired particle characteristic, which has been gaining popularity in recent years. There is a factor of emotional intelligence in the swarming particle characteristics, which makes them to alter their trajectories without any re-sampling requisites. Moreover, the nature based particle formulations do not generally require complicated data structures to store dynamical models in the system memory, and instead of being phased out, they evolve their trajectories towards an optimal region.

2.5.4 Mode seeking algorithms.

In contrast to the projective transformations explained in Section 2.5.1 and Section 2.5.3, a mode seeking (MS) algorithm does not utilise predictive tools as such, but instead exploits the differentiability of an objective function to gain an insight into the possible locations of an optimal region [83]. Therefore, a prime focus in the MS is to develop a sense of direction that guides the solution towards convergence. The term 'mode' explicitly refers to the highest possible altitude (peak) of the density, usually sought in an iterative manner. Similar to experienced mountaineers, who have to reach many rational decisions along various points in a summit, a mode seeking algorithm analyses local information to anticipate the best direction of ascend (where the objective function seems to be changing fastest towards a possible solution). Some simplistic but effective mode seekers are standard hill climbing [84] and the steepest descent minimization [85].

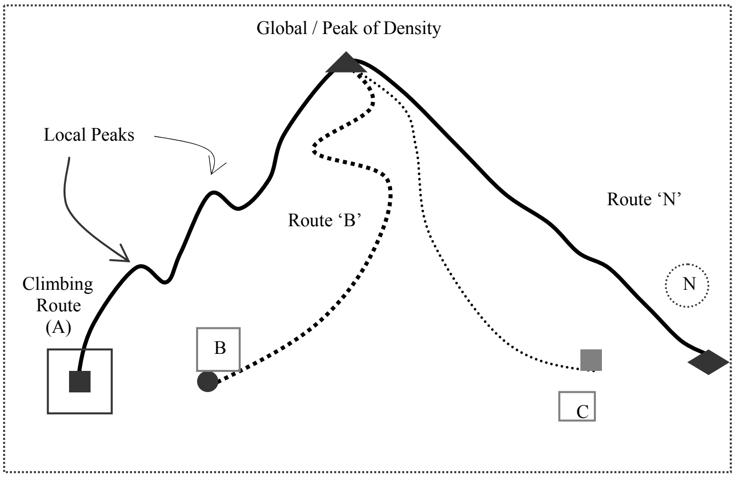


Figure 2.19: Iterative climbing process encountered in mode seeking algorithms.

To develop further insight into these techniques, a methodological example is portrayed in Figure 2.19; it shows four possible climbing routes, and all potentially leads to the global solution. The climbers would experience slope variability, encounter diverse hardships demanding mental and physical consistencies and agilities during any such ascending journey. Particular, in the presence of a poor visibility, climber 'A' may misinterpret a local peak as the global or highest altitude point, especially if they are not equipped with the appropriate instruments (e.g., altitude meters), and are oblivious of the peak due to poor visibility at this time. In the analytical terms, at each step during this ascend the algorithm calculates the gradient $\nabla f(x)$ of function $f(x)$. A condition of optimality is that at the peak of density, the gradient vanishes to zero i.e. $\nabla f(x^*)=0$. Here, $x^* \in R^n$ is a point in space where the vector $\pm(x_n^* - x_{n-1})$ had yielded the best change in the function value, and no further incremental modifications could be observed in any close neighbourhood or closer vicinities of x^*. The angle of this optimising vector $\vec{V}$ is called the gradient ascent/descent direction (in case of minimization). On the other hand, the determination of the magnitude of the gradient vector is relatively complicated, and a great deal of research has been particularly dedicated (e.g., exact line search, variable line search and the conjugate search direction [86]) to expedite the convergence. For the gradient descent scenario we can write the expression as

$$x^{(k+1)} = x^{(k)} - \eta^{(k)}\nabla f(x^{(k)}) \text{ and } f(x^{k+1}) < f(x^k) \tag{2.12}$$

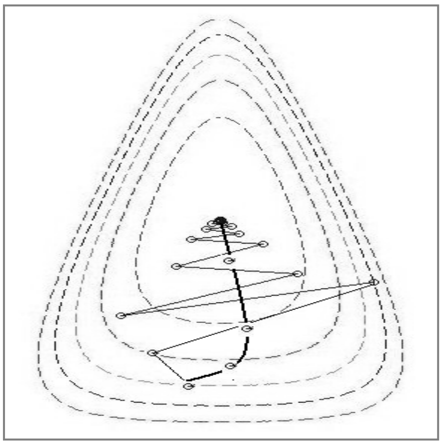


Figure 2.20: Contour plot of variable step gradient method to reach an optimum.

The descent process is drawn in Figure 2.20, but the convergences in the gradient based cases does not always yield the optimal solutions, as the solution is not able to differentiate between a global and a relative best solution. In Equation (2.12) η^k is the step size mentioned earlier, if it is selected too large then the solution may overshoot and fall into the non feasible area. The calculations to determine these step lengths play a significant role in the tracking applications, as very small convergent steps mean greater computational complexities (which could deter a real time approach). A real time solution needs optimal selection of jumps in the search space, and usually it is not possible for a single solution based approach to deal with such issues. There are two main aspects of standard mean shift tracking algorithm as mentioned underneath [87].

(A)-Operational basin (OB)

In MS tracking, the operational basin refers to the depth of the measurements in the search space, it refers to an area in space, in which various competing clusters are analysed, matched using density comparison tools [88], and therefore a gradient vector (mean shift vector) is calculated which sees the whole operational basin shifting to a new location. A selection mechanism is also programmed into the process, and more weights are allocated to the pixels lying at the centre of the regions.

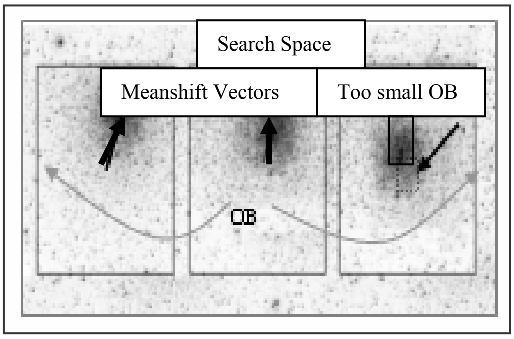


Figure 2.21: Iterative climbing to the peak of density using Mean Shift vectors.

This process is shown in Figure 2.21. At the far right, the operational basin is relatively small compared to the spread of the cluster under analysis, therefore calculating MS vector is based on local information only and hence will cause the tracking window to slowly diverge away from the true solution (in the subsequent frames). Alternatively if OB is too large and elaborative, the convergence would restrict a real time tracking application as usually an exhaustive search is conducted within the basin. Therefore, in order to find a best compromise between the speed of convergence and algorithmic accuracy (see Figure 2.10), the window size and relevant operational basins are dynamically allocated (in an ideal environment) as an object moves towards or away from the camera.

(B)-Kernel weighting.

To formulate a confidence map within an operational basin, the mean shift algorithm uses principles of kernel weighting to identify the modes of density. The most popular kernel in this respect is the parametric multivariate isotropic Gaussian [89]. The main idea of allocating preferences in this manner is to rectify noisy observations and to eliminate the effects of reflections from other static or moving objects in a scene.

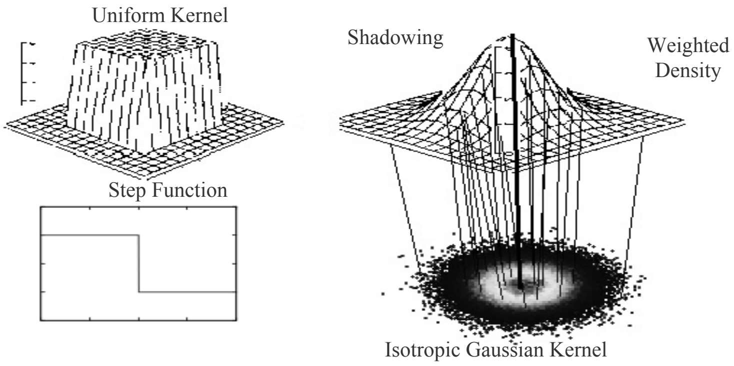


Figure 2.22: Effects of convolution of a discrete probability density function with Gaussian Kernel.

This confidence mapping process can be shown as in Figure 2.22. In contrast to the uniform kernel (a step function switching between two states), the weighting of the pixel values using Gaussian Kernels is more discriminatory, and for a clear understanding, it could be seen as a shadowing process displayed in Figure 2.22. To build a reliable map, some training could also be used to assign more accurate labels to the pixels under question, as shown in Figure 2.23. The overall accuracy of the confidence map on the left side is much higher than the one on the right. The fundamental objective of this process is to distinguish the background clutter. A detailed study of the background subtraction methods are presented in this publication [90].

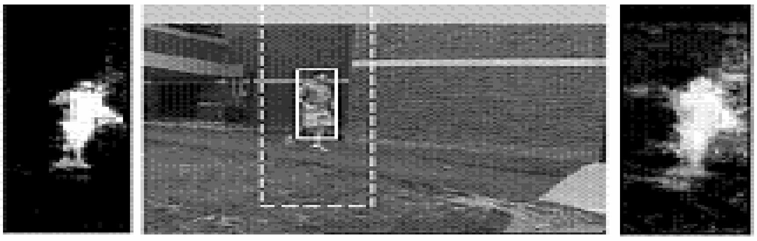


Figure 2.23: Confidence mapping in order to track a pedestrian [91].

Particularly, in the outdoor environments, addressing the ramifications of the changing light and other weather related conditions poses treacherous tracking conditions; hence, timely updates of models (background or foreground) may become obligatory. On the other hand, it is also possible to model a linear deterministic drift (in the feature space) as well after conducting a detailed examination of the lighting conditions overtime.

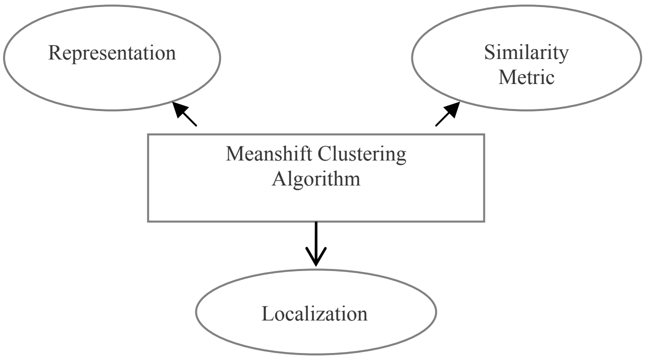


Figure 2.24: Three main parts of the Mean-shift tracking algorithm.

The Mean shift clustering algorithm is composed of three main stages as shown in Figure 2.24. The mathematical background in the context of these three stages will be described in this section. In Equation (2.13), the object model/representation is constructed in terms of its discrete density estimate 'q_u', this is accomplished by first calculating corresponding bin indexes of individual pixels $b(x_i^*)$, each contributing one delta to the feature histogram. All delta functionalities are then summed up, and weighted in accordance to a Gaussian kernel definition $k(.)$.

$$\hat{q_u} = C\sum_{i=1}^{n} k(\| x_i \|^2)\delta[b(x_i^*) - u] \qquad (2.13)$$

This process is repeated for all pixels $(i = 1 : n)$, and the weights are then normalised as shown in Equation (2.13) by the coefficient $'C'$. These object models are then stored in forms of arrays, and in upcoming frames are statistically matched to the candidate models constructed during live tracking phases (also discrete distributions). The most common metric used to determine the overlap of densities is the Bhattacharyya similarity measure [92] as shown in Equation (2.14). Therefore the distance between two densities is calculated by using Equation (2.15).

$$D_B(p,q) = \int \sqrt{p(x).q(x)}dx \qquad (2.14)$$

$$d_H = \{1 - D_b(p,q)\}^{\frac{1}{2}} \qquad (2.15)$$

If $p = (p_1, p_2,...p_n)$ and $q = (q_1, q_2,....q_n)$ are any two such vectors in an n-dimensional Euclidean space, then Equation (2.14) could also be interpreted as the dot (scalar) product between these two vectors, and the resultant measure is a real number $R \in 0:1$. In the context of MS tracking, the Bhattacharyya coefficient is the objective criterion calculated during each frame, and acts as a surface mapping where the tracking window is drifted towards the more dominant mode of the density (a spatial location where the distance in Equation (2.15) is minimised).

The landscape of this objective function (BC) could be smooth and subtle or highly rippled, and therefore, seeking modes of density using single solution based approaches is somewhat difficult to achieve in the practical real life applications. Hence, MS works perfectly well in the linear and uni-modal environments, as it does not possess an inherent capacity (in the original format) to address the multiple modes in the local landscape (as shown in Figure 2.25). In the simplistic one dominant modal case, the density climbing is much smoother and accurate compared to the one on the right hand side of Figure 2.25.

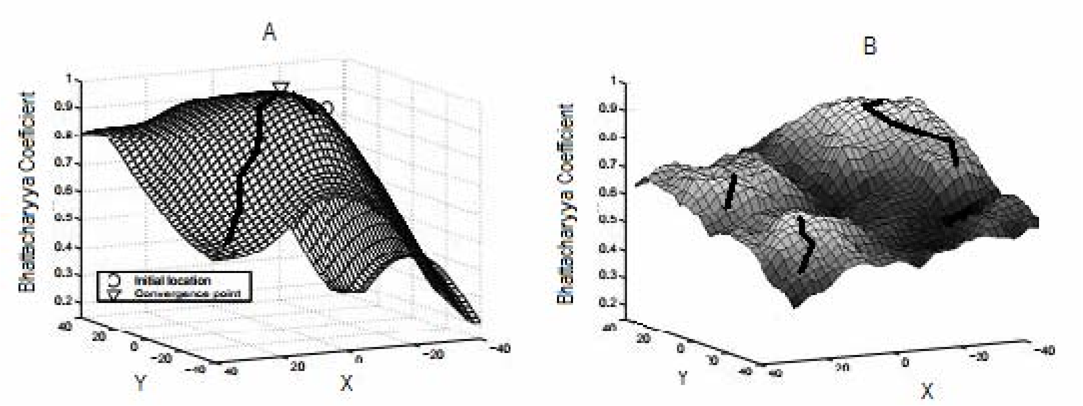


Figure 2.25: Landscape generated by Bhattacharyya coefficients in tracking [93].

The initial location of the solution is represented by a small circle along with the convergence point which is marked using a triangle in Figure 2.25a. The standard meanshift tracking is a single solution based approach, and in its original format is unable to address multi-modal environments as represented in Figure 2.25b. The iterative localization procedure in Figure 2.25 is accomplished by calculating the gradient (represented by the mean shift vector), and then it shifts the kernel towards this newly determined position in the search space. For an n data points in a d-dimensional space R^d, we can write this multivariate kernel density estimate as in Equation (2.16).

$$f(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K(\frac{x - x_i}{h}) \qquad (2.16)$$

where $x_{i,}, i = 1,...,n$ are n data points, nh^d is a normalisation constant to take into account the effects of window sizes h.Thus the convergence steps using a circular symmetric kernel profile(isotropic Gaussian) could be written in the format shown in Equations (2.17) and

(2.18) [93]. The first term in Equation (2.18) is the position of the mode in newly perceived likelihood map obtained through matching the densities, and x_0 refers to the previously calculated position of the region of interest.

Therefore, the new window position x^{t+1} is calculated through an iterative shifting of the previously known coordinates x^t of an object by applying the mean shift vector translations as calculated by the expression shown in Equation (2.18). The meanshift is a very efficient algorithm, and only the object based features and characteristics are required to track an object of interest. In Equation (2.17), 'C' is the normalization constant, so that the kernel weights add up to unity. All observations in time are weighted by the kernel k, and the distance metric $\| x \|$ stipulates the fact that the kernel preferences depend on the standard Euclidean l^2 norm [94]. Finally, the mean shift vector is calculated using Equations (2.18) and (2.19).

$$K(x) = C_{k,d} k(\| x \|^2) \qquad (2.17)$$

$$m_h(x) = \left[\frac{\left[\sum_{i=1}^{n} x_i g\left(\| \frac{x - x_i}{h} \|^2 \right) \right]}{\left[\sum_{i=1}^{n} g\left(\| \frac{x - x_i}{h} \|^2 \right) \right]} - x_0 \right] \qquad (2.18)$$

$$x^{t+1} = x^t + m_h(x^t) \qquad (2.19)$$

2.6 Critical analysis of the tracking principles.

In Section 2.5 three dominant techniques are presented that almost constitute the bulk of modern tracking methodologies in computer vision applications. However, the main focus of this thesis is to adapt an objective orientated method that manifests simplistic and dynamically altering solutions to the tracking problem that could be implemented in an embedded processing environment. This could mean finding and discovering solutions using a scale-space of problem (e.g., using Gaussian pyramids [95], and then clustering using frame differences [96]). Restricting the domain of measurements analytically might not be the most effective or feasible approach in this regard.

Alternatively, rather than viewing the tracking problem through an analytical eye, it might also help to overcome some of the most common misconceptions, and to avoid configuring frame tracking problems in the light of the general control theory devised particularly for process design applications. Therefore, it would be much more interesting to study the tracking problem in a hybrid framework (evolutionary factored sampling/MCS and mode seeking algorithms). This problem is also identified in Figure 2.25, where the landscape proposes a multiple solution approach, and due to the reason that such kind of problems are highly sensitive to an accurate declaration of initial conditions.

2.7 Conclusions.

Some of the most prominent tracking methodologies are discussed in Chapter 2. The fundamental flaws of the recursive Bayesian estimation (RBE) are discussed in Section 2.5.2. The Monte-Carlo sampling (MCS) in Section 2.5.3 addresses some of the inherent weaknesses/flaws of the RBE technique but even the most popular MCS method (a particle filter [97]) is self contradictory in the view of this report. The fundamental reason of the contradictory behaviour is due to the possibility of accumulation of errors in the state transition models; a re-sampling stage therefore is usually required to compensate the detrimental effects of the discrepancies in the plant models, which usually restrict a real time and fast convergence.

Chapter 3

Natural Experiential Learning

The evolutionary branch of mathematics is a fast developing science primarily based on the theories of learning and swarming among social insects, birds flocking behaviour, and is deeply inspired by both microscopic and macroscopic world. In order to achieve collective goals, a two phase inter species phenomenon of competition and collaboration has been frequently observed in the biological life forms, and is also extensively studied in the scientific literature [98]. Our planet earth is a typical example of a multiple agent system, where members of certain species not only race against each other to gain peer attention, but at the same time provide navigational aids to the colony that helps to reach significant food reserves. The competitive and collaborative mechanisms in natural colonies are therefore leading research directions in the modern optimisation literature.

Figure 3.1: A sensational murmuring phase observed in a group of starlings [101].

One specific example worth mentioning in this context are the research findings of a group of mathematicians working on the Starflag project, its research goal is to understand the flocking and murmuring phases observed during starling flypasts (within the boundaries of the cosmopolitan city of Rome [99]), an intriguing murmuring phase of starlings is shown in Figure 3.1. The fascinating formations of starlings are due to the marvellous achievements at both individual and collective levels, and are not only aesthetically admirable but are also remarkable displays of strength [100]. During murmuring phases, starlings tend to split and merge into intriguing group formats and at a blink of an eye. The effective scanning of environment at both individual and combined level therefore caught the attention of scientists and engineers alike to solve complex processes.

Another profound scenario of the natural optimisation sequence (a chain of events) has been noticed in the ant colonies, where foragers compete among each other to reach most admissible food resources [102]. The waggling of a forager bee at the hive dance floor is another fascinating phenomenon which has baffled researches for many decades. As a matter of fact, it was later learnt that the waggle dances of honey bees are social techniques that convince other members of the hive to investigate prominent food resources [103]. The second most important issue to be addressed in this thesis is to introduce the radical particle behaviours that could help to resolve the fundamental flaw in standard optimisation algorithms. Modern evolutionary algorithms, e.g., ant colony (ACO) [104] and particle swarm optimisation (PSO) rely on one major assumption that all agents of population have to be physically transported and shifted [105] (e.g., from a nest position A to the newly discovered location B).

The aim of this chapter is to explore the applicability by programming foraging characteristics within particles to solve complex mathematical test functions. After studying the inherent weaknesses of general gradient based methods, a new optimisation scheme is proposed that grants a lot better vision to the agents. The underline freedom in choosing motion trajectories in exploring complicated multi-modal state space relies on a novel revolutionary dynamics described in Section 3.4.

3.1 David Kolb's Learning Model (KLM) in the tracking applications.

During 1984, David Kolb presented his profoundly famous learning model [106] that ever since has been applied to a wide variety of educational programs to facilitate learning. The fundamental aim of the Kolb model is to deliver tailor made lesson plans for both individuals and groups of learners alike. The possible domain of KLM could range from artificial computation to devising effective lesson plans in both primary and higher level educational and training institutes. According to Kolb, "learning is a process, whereby knowledge is a transformation of experience gained through phases of active experimentations [107]". The Kolb's theory initially tries to address the learner's internal cognitive processes through experiential observations, and once such information is suitably inferred, it is reflected back into delivering more appropriate sequences of information, hence maximising the chances of gaining a far reaching body of knowledge (as mentioned above).

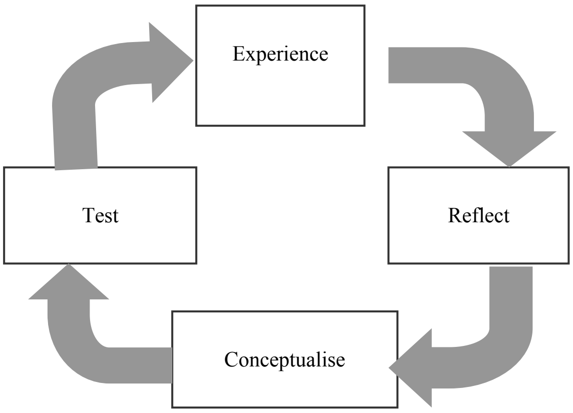


Figure 3.2: David Kolb learning model.

In Figure 3.2, the four typical stages of KLM are presented, the iterative application of self reflections and conceptualisation help in devising more meaningful experiences. The title of this chapter (experiential learning) also implies this core idea (presented in the Kolb's theory of learning), and the main differentiation of this technique revolves around embedding broader experimental variations in tests in order to acquire knowledge and expertise to perform a task in the best possible manner. In short, an experiential learning is a process in which a mass or body of knowledge is gained through active experimentations rather than solely based on the theoretical expositions and models. This experimental form of learning is also very close in nature to the naturally occurring processes seen in early year children and infants, and is witnessed in both human and natural populations. Based on such considerations, the tracking problem in computer vision could be effectively solved by granting searching particles the experiential freedom to plan their personal expeditions.

3.2 Towards developing a faster Global convergence methodology.

This section prepares the readers to gain further insight into various mathematical optimization processes. The Fermat theorem (FT) [108] is a basic optimisation strategy that proposes an initial search direction (e.g., along the gradient) to find a solution. The first order condition in FT imposes a necessary condition (to be satisfied) before a point in a search domain could be further tested to establish if it is indeed a local extremum.

If S is a feasible subset of a Euclidean search space of dimension R^n, and $f \in R$ ($'R'$ is the real line) are the function values over the defined set $'S'$, then according to FT, at a relative minimum or maximum point x^*, the gradient of the objective function vanishes and becomes zero (i.e., $\nabla f(x^*) = 0$). In order to distinguish other possible stationary points (e.g., inflection points [109]) from the local optimal solutions, it is a common practice to carry out a second order derivative test. The second derivative test imposes sufficient conditions, and a point satisfying both first and second order derivative tests is finally classified as a relative critical point [110].

In the simplistic words, sufficient conditions theorem (second derivative test) implies that if the function's first derivative vanishes at some point, and its second order derivative is greater than zero at that point $f''(x \ ) > 0$, then the point x is a local maximizer of the function f . On the other hand, a point x^* is a local minimum if the value of the function second derivative at this point is less than zero $f''(x^*) < 0$. The third interesting but rather confusing scenario takes place when the second derivative tests prove inconclusive, and in those circumstances, further higher order derivative tests are imposed (e.g., Taylor series expansions [111]) in order to locate further critical points, and to investigate the general trend of changes in its domain.

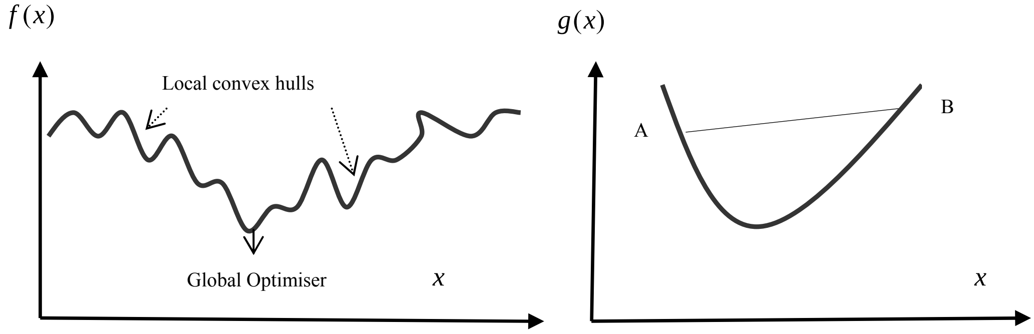


Figure 3.3: Examples of a non-convex and a strictly convex function on the right.

In Figure 3.3, two function plots are presented. In the monotonic case (on the right), the functions values either increase or decrease over its domain (and gradient information could be used as a guidance), and therefore is simpler to analyse and investigate in comparison to the non-monotonic case. Furthermore, the function on the right in Figure 3.3 is a uni-modal test case, because it is monotonically decreasing to a certain point, and then function value starts increasing in its domain.

In contrast $f(x)$ in Figure 3.3 is multi-modal, and the monotonic properties (changes in function values) could only guide to a local solution. There is a wide range of competing regions and convex hulls in this test case, but only one leads to the global best or an optimal solution. If the solution gets trapped into a local ridge and valley, there is no guarantee that the optimal solution will be discovered, such problematic situation has been discussed in literature previously and any interested reader is referred to the following literature to gain further insight into this mathematical problem [112] [113].

3.2.1 Greedy optimization strategies.

The fundamental weakness encountered in the monotonic object tracking techniques is due the factual possibility that these methods could get short sighted very quickly, and may converge only to maximise the immediately available rewards at any moment in time. This is the main reason of failure of mode seeking algorithms [114], the contour methods also suffer from similar setbacks due to the non differentiable stationary points developing on the surface of a contour [115]. Many high curvature points generated on the curve surfaces are the consequence of a reduced visibility after following the steepest descent directions for a number of iterative steps.

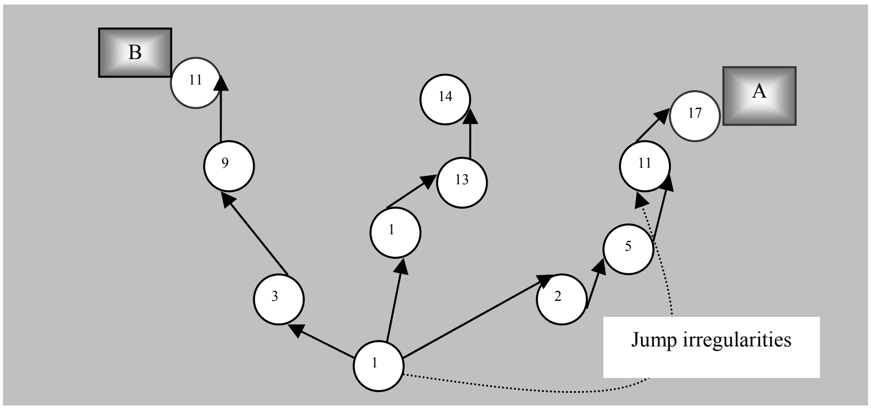


Figure 3.4: A short sighted solution tends to converge towards local mode (node B).

The diagram presented in Figure 3.4 shows how a short sighted and greedy algorithm tend to converge to a locally optimal solution (node-B), and generally would remain oblivious of the position of the global best solution (node-A). We will conduct a range of tests to clarify the effects of the above mentioned adversities in the evolutionary test cases in this chapter. In order to expedite the convergence timing and to provide a real time response, a correct locus to the optimal solution is needed within a allowable time limit.

To increase the prospects of locating the correct node, one possible resolution is to introduce the jump irregularities as shown in Figure 3.4. An intuitional heuristic shortcut and jump could also result in the reduction of the convergence timing (however, the optimality is not always guaranteed), which becomes more apparent if the depth and breadth of the searches are vast (e.g., there are $307k$ data points in an image of resolution $640x480$. On the other hand, if all the data nodes are revisited in order to understand their relevance in the feasibility space, then such a search resembles a brute force search, and is a gigantic computational overhead. A one dimensional graph is plotted in Figure 3.5a to elaborate this further using a simplistic test polynomial.

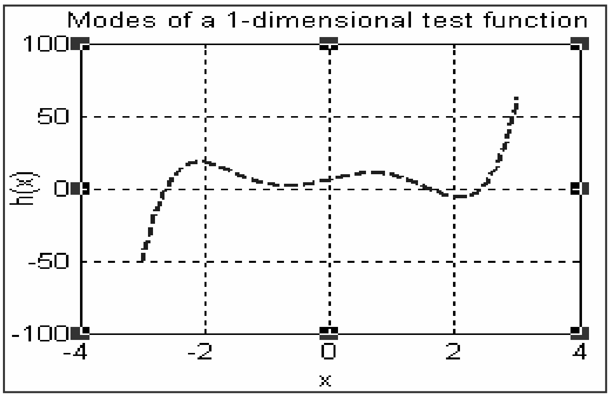


Figure 3.5a: Graph of polynomial $h(x) = x^5 - 8x^3 + 10x + 6$.

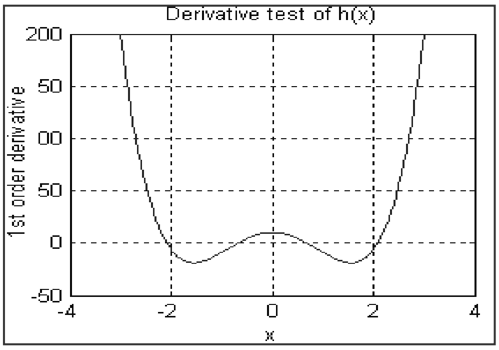


Figure 3.5b: First derivative test of function $h(x) = x^5 - 8x^3 + 10x + 6$.

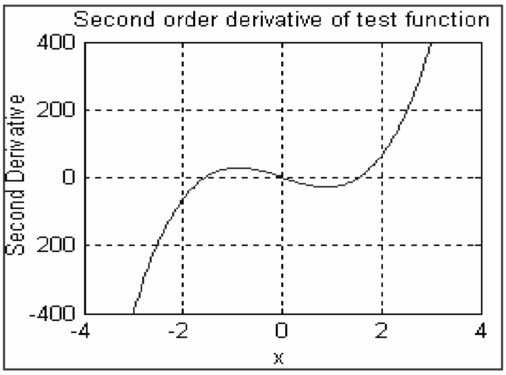


Figure 3.5c: Second order derivative test of function $h(x)$.

Suppose the optimisation task is to find the absolute minimum (x^*) of the function $h(x)$ which is defined in the intervals $x=-2$ to $x=3$. But in reality, many practical optimisation problems have no detailed knowledge of the graph of a function. After analysing the graphs of the 1st order derivative test (necessary conditions) in Figure 3.5b, it becomes apparent that the relative minimums are located in the vicinity $(x=-0.68)/(x=2.083)$.

The second derivative further tests the projected hypothesis by implying a sufficient condition test and the position of the absolute minimum of $h(x)$ are analysed using higher order tests (Figure 3.5c). It is evident from the function plots, that, in order to use the local convexity to reach a solution, we have to initiate the searches at diverse points in the search space. There are two competing troughs in the domain of this function (Figure 3.5), and when the minimization process was initiated at $(x=0)$, and the function monotonic characteristics are applied, it resulted in reaching to a local solution $(x=-0.68 \ \& \ h(x)=1.5701)$. However when two competing solutions were initialized in the search regions, it enabled the same algorithm to detect the absolute and the global optimiser $h(x)=-6.2586$ is observed at ($x=2.083$).

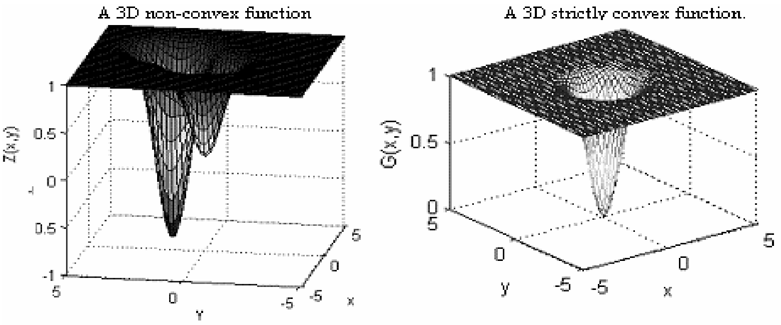


Figure 3.6: The landscape of a non-convex and convex three dimensional problems.

The test scenario discussed above is a clear demonstration of the importance of initialization in the multi-modal landscapes, and is one of the most rigorous issues to be tackled in the global optimization branch of mathematics. The minimization procedure in higher dimensional problems is similar to the 1-dimensional case presented earlier; the only exception is that search directions are established using negative gradients. The landscapes of both convex and non-convex functions are plotted in Figure 3.6. These $3D$ cases resemble camera generated images (as both are functions of two variables); therefore, it seems reasonable to study convergences using such artificial landscapes. The decaying graphs in Figure 3.6 are plotted using Equations (3.1) and (3.2) respectively. The exponential function defined in Equation (3.1) is strictly convex and a trough at $(x = 0, y = 0)$ is prominent to notice, whereas in Equation (3.2), two competing convex regions are present with a non-convex overall response.

$$G(x, y) = 1 - e^{(-x^2 - y^2)} \tag*{3.1}$$

$$Z(x, y) = 1 - (e^{-x^2 - y^2} + 2e^{-(x-1.7)^2 - (y-1.7)^2}) \tag*{3.2}$$

```
Algorithm 2.1

1. Compute $-\nabla_R f(x^{(k)})$
2. Choose $\eta^k = \arg\min_{\eta}\{f(x^k - \eta\nabla_R f(x^k))\}$
3. Update $x^{k+1} = x^k - \eta^k\nabla_R f(x^k)$
4. Repeat (Go to 1) until $\left\|\nabla_R f(x^k)\right\| < \varepsilon$
```

Figure 3.7: A generalised pseudo code of the gradient descent (GD) convergence algorithm.

In this section we will try to understand the convergence properties using gradient descent optimisation (gradient testing is also common in contour tracking methods [116] [117]). The pseudo code for a generic minimization problem is portrayed in Figure 3.7. In Algorithm 2.1, the fastest change in the function values is calculated using a negative gradient first (step 1), the selection of an appropriate scale of measurements ∇_R ($'R'$ is the bandwidth in which changes are observed) is an important aspect in accomplishing this step. The second step/stage establishes how long it would be feasible to travel in the direction of the gradient to minimise the given function $\eta\nabla_g f(x^k)$ within a neighbourhood, hence a vector/ray is projected in that direction to answer this question. The factor 'η' assigns a suitable magnitude to this vector/line search to calculate an optimal value of $f(x^k)$ (step 2), and finally the solution is updated by calculating x^{k+1} in step 3. The iterative process shown in Figure 3.8 is repeated until the magnitudes of the changes fall below a predefined threshold $\left\|\nabla_R f(x^k)\right\| < \varepsilon$, gradient ascend works in a similar way with the exception that the optimum points are yielded using $+\nabla_R f(x^k)$ direction.

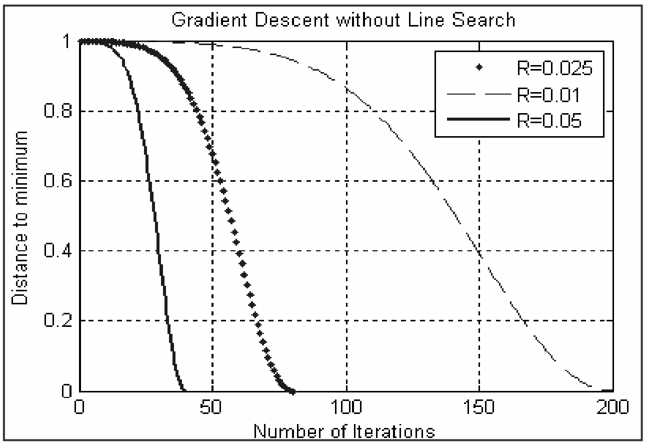


Figure 3.8: The gradient descent convergence of $1-e^{(-x^2-y^2)}$ using different bandwidth values.

In Figure 3.8, the number of iterations have been significantly reduced (from '200' steps at $R=0.01$ to '40' when $R=0.05$) by choosing more appropriate scale. Similarly, some built in ray casting procedure and automatic selections of resolution may prove beneficial to detect intensity variations in a digital image, and the computational complexity could also be reduce as such technique requires less memory operations.

The convergence timing graph (Figure 3.9) was plotted using a resolution sweep spanning from 0.005 to 0.1050. Figure 3.9 emphasizes that the convergence timing could be significantly reduced by systematically varying the solution starting points in the search space. In all 3 cases key changes in characteristic are observed around 0.04, and further decrementing R may prove detrimental in terms of converge timing, a threshold error of $\varepsilon = 0.01$ was used in this experiment.

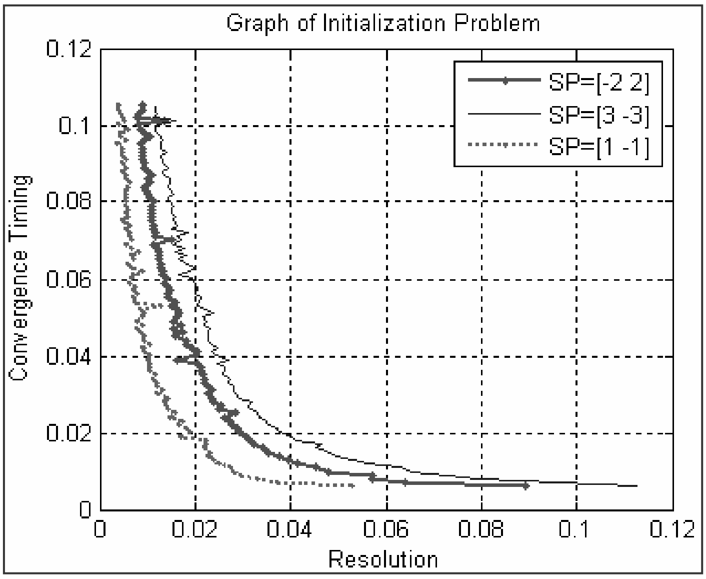


Figure 3.9: Timing graph of a test function (defined in Equation 3.1 with distinctive starts).

The experiment conducted in Figure 3.8 was repeated using fixed line searches (FLS) this time, and Figure 3.10 provides a graphical comparison when the gradient descent was applied without inbuilt line searches (NLS). The advantage of an imminent line search is clearly evident in Figure 3.10 as the solution converges in just 10 steps.

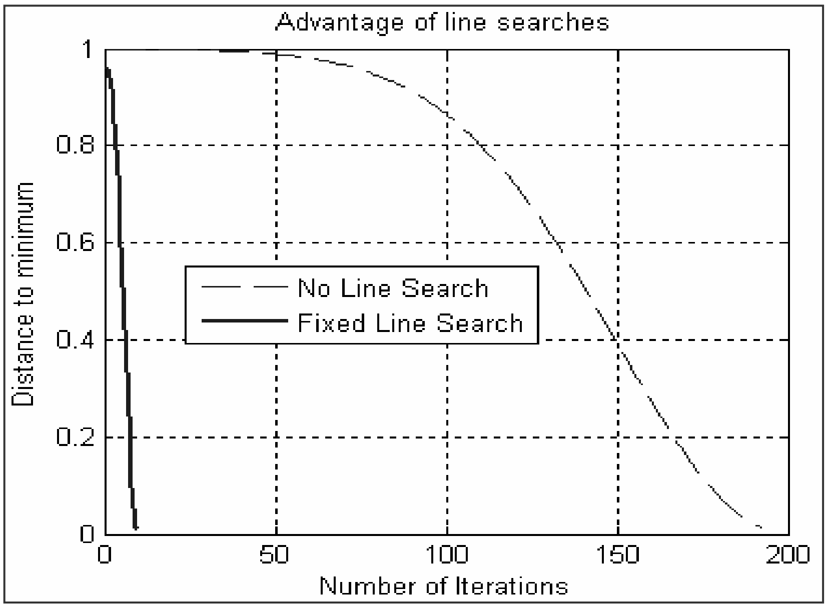


Figure 3.10: The convergence of a test function using a fixed line search operation (FLS)

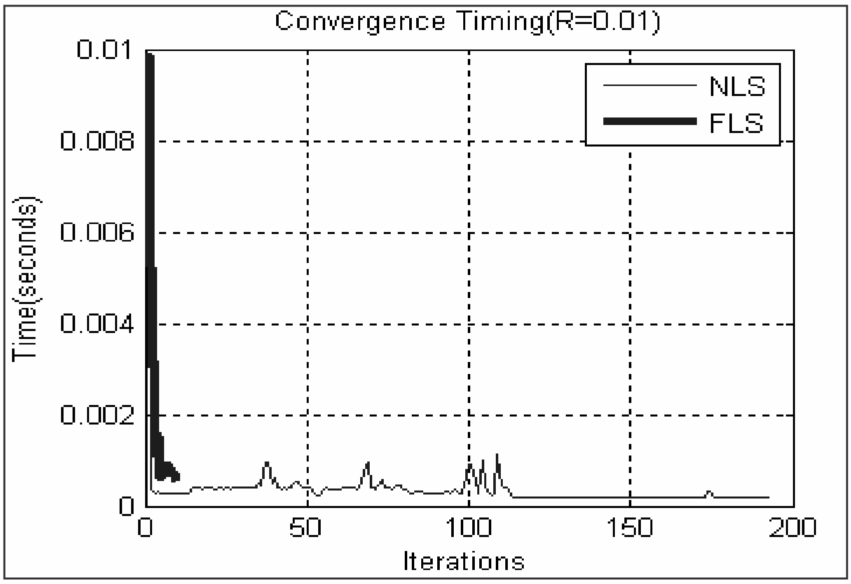


Figure 3.11: The comparison of the convergence timings (NLS vs. FLS)

The convergence timing graph is presented in Figure 3.11. Although the time needed for the NLS to converge at each iterative step is far less than the FLS scenario ($3.48e^{-004}$ and 0.0022 seconds respectively), the overall descent time in the FLS case was recorded to be just 0.022 seconds, which is significantly lower than NLS, and has taken 0.0674 seconds to converge to the global solution. The reasons for the individual FLS iterations to be computationally expensive than NLS is simply due to the fact that, a significant proportion of time is spent in determining the slope (ray casting process as shown in Equation 3.3) and in evaluating function values at new data points.

$$y = (\frac{y_2 - y_1}{x_2 - x_1})x + C \qquad 3.3$$

The general line equation is presented in Equation (3.3), where the values of x are stipulated using the history of movements (gradient directions), whereas, the value of the second independent variable $'y'$ is determined using the slope $(\frac{y_2 - y_1}{x_2 - x_1})$ (calculated by using the previous point of convergence to the newly calculated direction of the gradient i.e. $\nabla_R f(X)$), where $X = \begin{bmatrix} x \\ y \end{bmatrix}$ is a vector in this two dimensional Euclidean space and $'C'$ is a constant that refers to the initial location of the data points in a solution space. In object tracking, we often experience similar situations (mentioned in the previous paragraphs) especially when deterministic techniques (e.g., Kalman Filter [118]) are applied in tracking, the historical movements are generally ignored in predicting better solutions (but could be utilised due to economical memory devices at this technological age). Another relevant technique is Hill climbing [119] which is also a single solution based approach.

The contour plot of the function $Z(x,y)$ (Equation 3.2) is shown in Figure 3.12. The plot of the monotonic characteristics of $Z(x,y)$ sheds a focus on the inherent weaknesses of the gradient oriented solutions. Particularly, in the multi-modal imaging landscapes the monotonic feature is not a stable approach due to the reasons explained earlier in this section. The arrows in Figure 3.12 indicate the direction of $+\nabla_R f(X)$ which are highly inconsistent at various point in the graph of function.

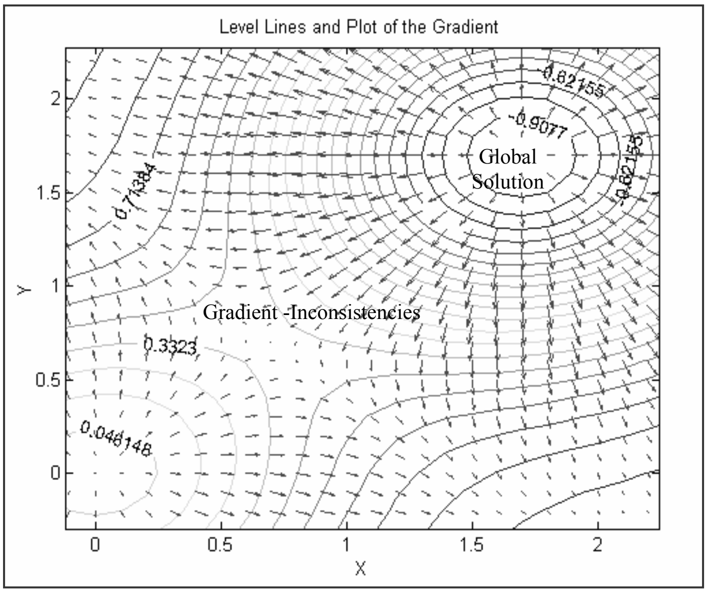


Figure 3.12: The combined (Level lines and direction of the gradients) plot of $Z(x,y)$.

Figure 3.13 is a rotated graph of function $Z(x,y)$ that was presented earlier in Figure 3.6, and only 90 data points are used to sample its unique camel hump characteristics using Matlab. The gradient inconsistencies in relation to both Figures 3.12 and 3.13 clearly demonstrate the need for an experiential variation type of approach as discussed earlier in Section 3.1. The prospects of converging to a global solution are therefore more circumstantial in the multi-modal landscape. If the GD optimization is initiated in the valleys in Figure 3.12, there are minimal chances of recovery.

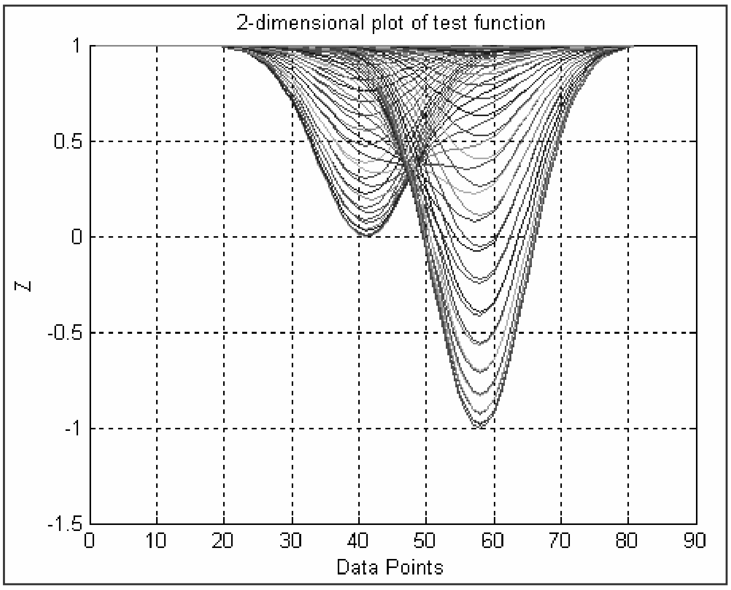


Figure 3.13: 2D plot of the local and the global optimal solutions of $Z(x, y)$.

Figure 3.13 indicates a relative minimum $Z(X) = -0064$ at $X = \begin{bmatrix} 0.01 \\ 0.01 \end{bmatrix}$ and a global optimum $Z(X) = -1.0031$ at $X = \begin{bmatrix} 1.7 \\ 1.7 \end{bmatrix}$. Having mentioned the importance of the initialization stages previously (Figure 3.4), we will now try to study the initialisation problem in more detail by performing a gradient descent test on a multi-modal test case. In this section 500 test runs are conducted in order to understand the convergence issues, and therefore the accumulated errors are analysed using pseudo random sampling from two Normal distributions in Figure 3.14.

In this experiment, the aim is to guide the solution into the correct convex hull (as was emphasized in Figure 3.3) by perturbing searches using two unique Gaussian seeds. In comparison to the function in Equation 3.1 (where optimality is guaranteed), the aim of these tests (using highly rippled landscape with a lot more local distracters) is to analyse if randomization could facilitate in discovering dominant modes (Section 2.5.4).

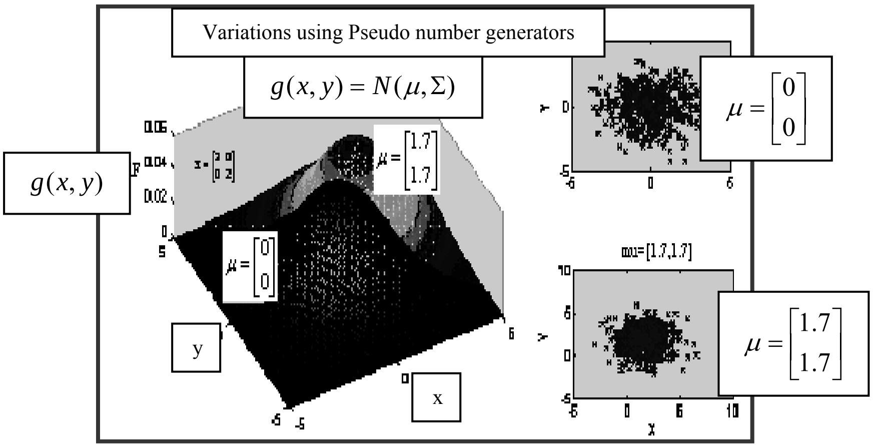


Figure 3.14: Random sampling from 2 Gaussian distributions (with distinct means).

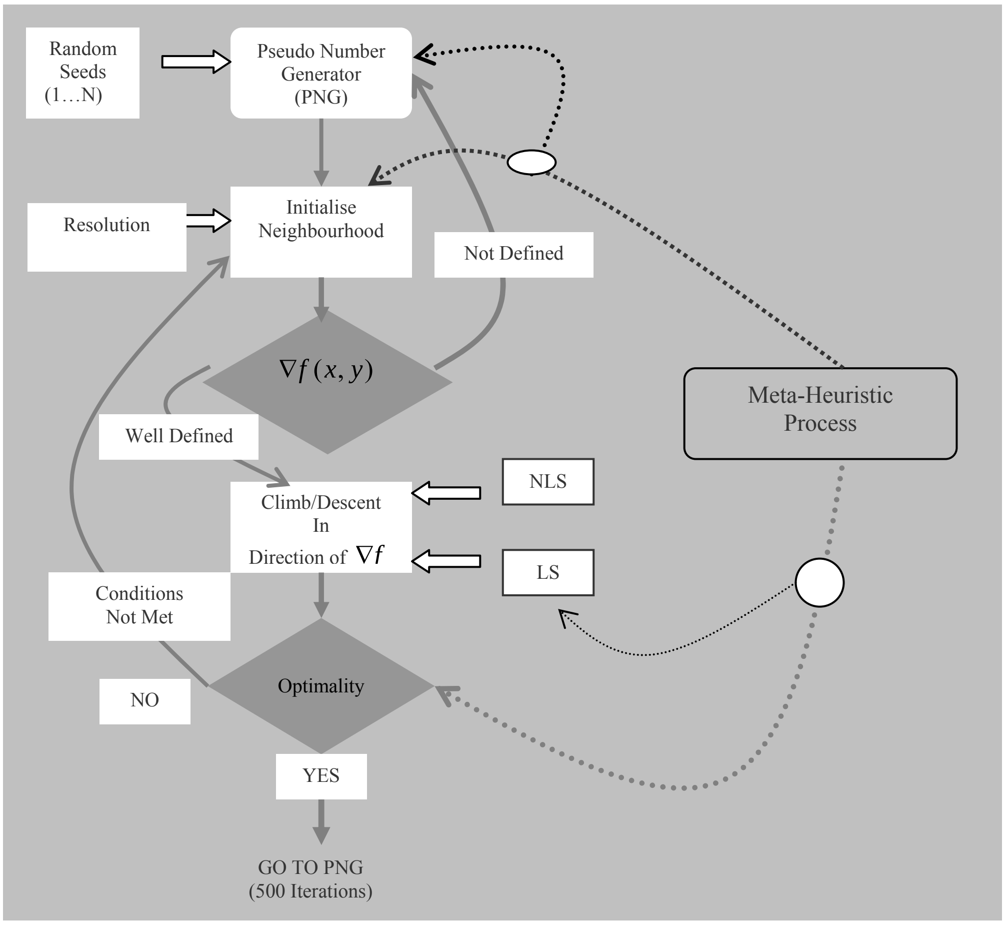


Figure 3.15: A stochastic meta-heuristic based gradient descent algorithm.

The flow diagram of the tests conducted in Figure 3.16 is drawn in Figure 3.15; the main focal point of this novel stochastic gradient based descent test (beside other key points mentioned earlier) is the random seed generator which diversifies the searches in order to develop several parallel hypotheses in the search domain.

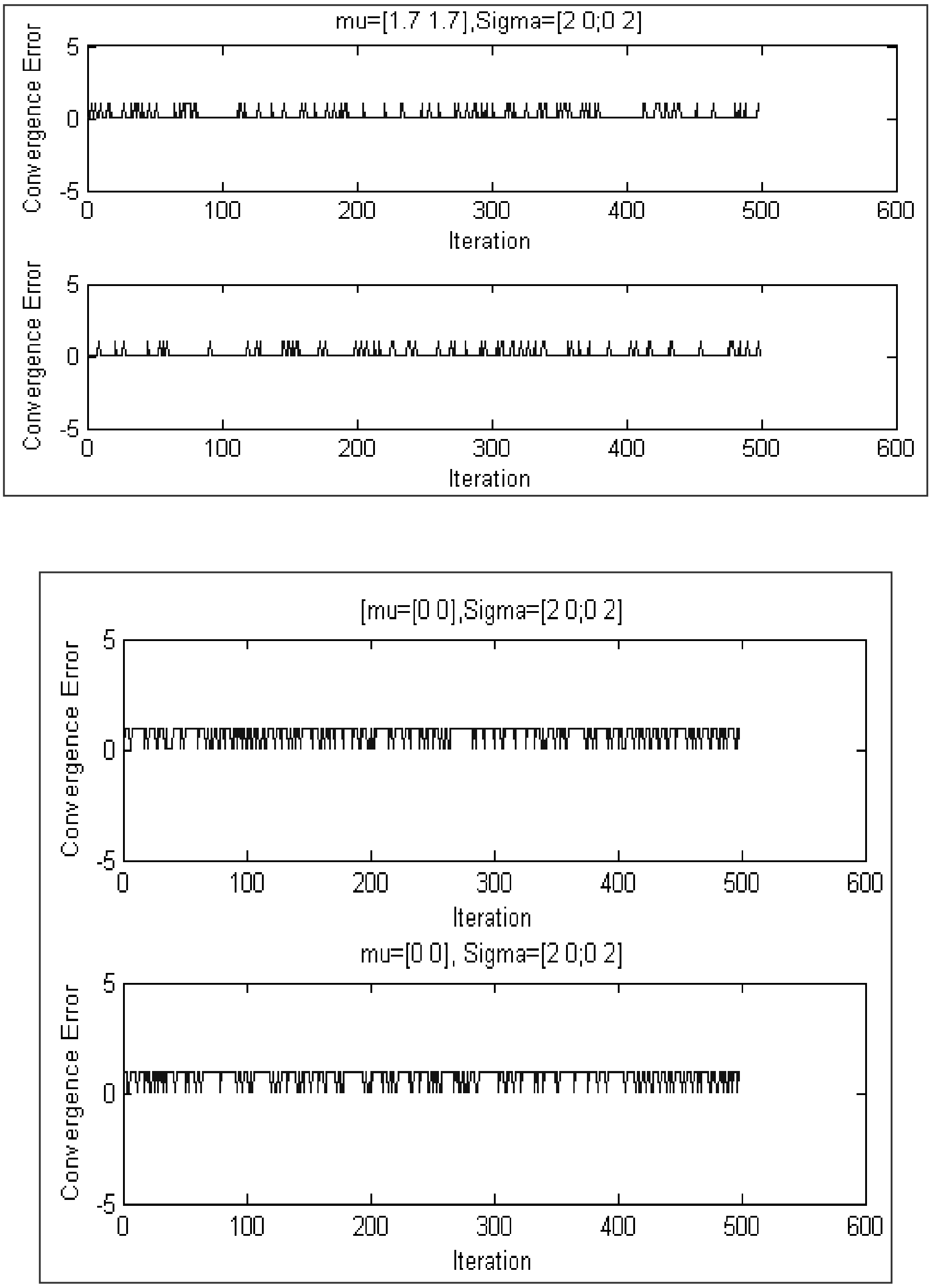


Figure 3.16: Error graphs generated by stochastic process defined in Figure 3.15.

In comparison to the gradient descent process in Algorithm 2.1, the downhill walk in the direction of the negative gradient is repeated here until the convergence conditions are met for a variety of starting points (500 iterations/seed). Along with the NLS/LS processes (explained earlier in Figure 3.10), it is generally assumed that an additional meta-heuristic stage (if inserted onto the top of tracking and optimization algorithms) could introduce considerable improvements in all key stages during the descent walks. However at this stage of Chapter 3, only the results of the randomization processes are shown in Figure 3.16.

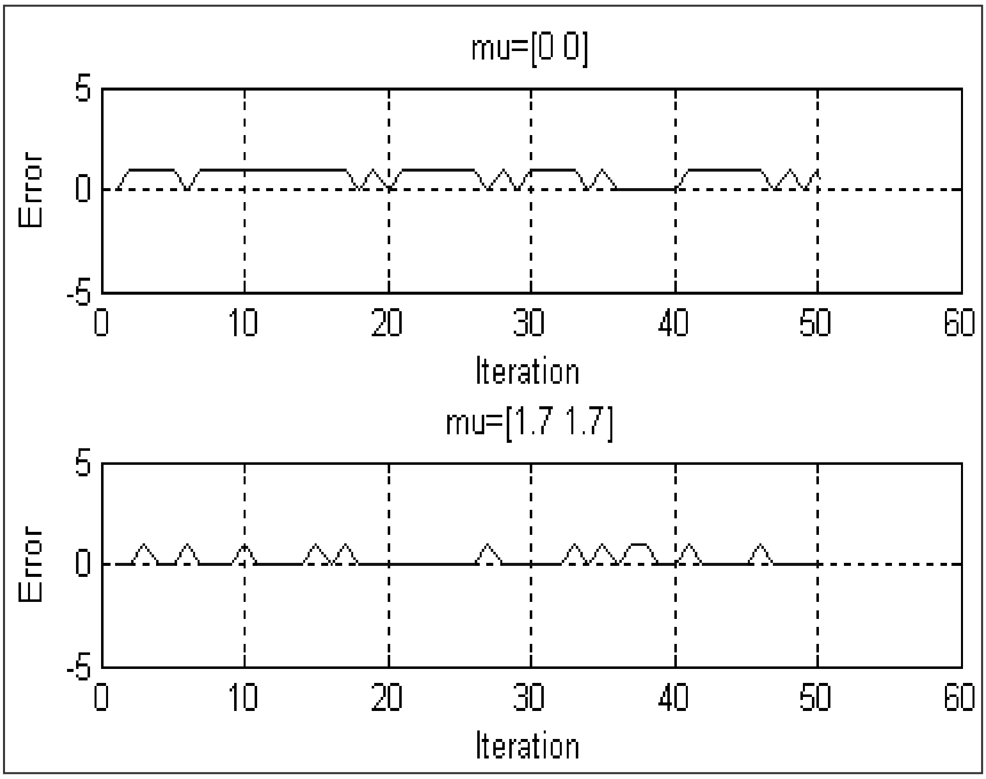


Figure 3.17: The zoomed in version of the graphs drawn in Figure 3.16.

From the graphs of the convergence errors (Figure 3.16), it is clear that the convergence errors are significantly reduced when the perturbations are generated using a Gaussian with a mean $\mu = \begin{bmatrix} 1.7 \\ 1.7 \end{bmatrix}$. Figure 3.17 is a zoomed version of the graphs presented in Figure 3.16, which also confirms the suggested proposition that suitable initialization is mandatory for accuracy.

In Figure 3.17 the ratio of accumulated error was observed to be 39:15 which is a significant improvement compared to when $\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$. The probability of iterative random solutions correctly identifying the global best were observed to be 0.82 when $\mu = \begin{bmatrix} 1.7 \\ 1.7 \end{bmatrix}$ and in the order of 0.3034 when $\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$. Therefore, it could be safely concluded that by increasing the level of diversity (a kind of pseudo number seeding used earlier) would also facilitate significant improvements in exploring the dominant modes in the detection algorithms in computer vision (due to close resemblances of landscapes).

3.2.2 The simulated annealing as an optimization process.

A controlled annealing is a technique in the Metallurgical engineering that removes the molecular level defects in machine parts constructed from versatile alloys. In the first stage of the annealing process, metallic components are subjected to very high temperatures which facilitate the removal of existing bonds between atoms and alter the molecular properties of a material. During the next phases, a temperature schedule is maintained by a slow cooling mechanism resulting in re-crystallization, hence the desired characteristics (e.g., better stress/strain capacities) are forged which are required for a particular design scenario. The heart of the annealing process is a molecular diffusion process. The simulated annealing (SA) algorithm is a relatively new state space exploration methodology (compared to the deterministic analytical methods) in which a solution is perturbed to escape the local traps using controlled cooling schedules [120].

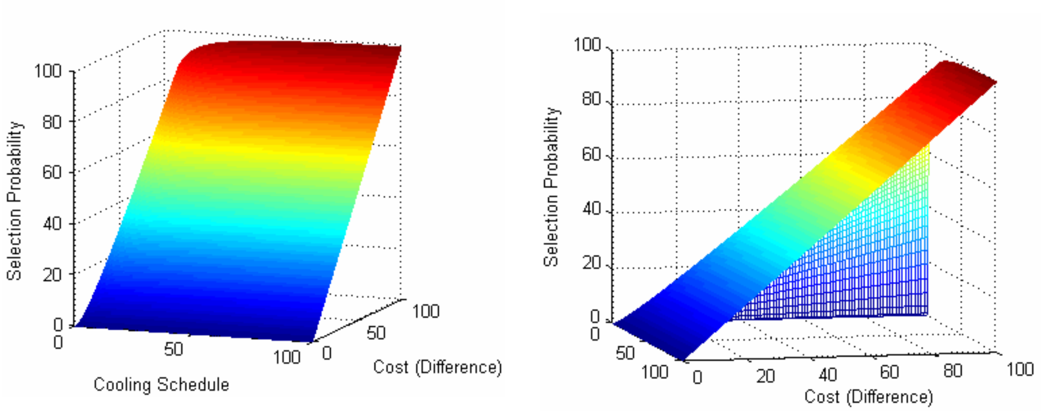


Figure 3.18: The temperature controlled selection of neighbourhood nodes in SA.

In contrast to the solution ascending/descending towards the immediately best solutions in the gradient based methods, in SA algorithms some worse solutions are also selected depending on the current temperature schedules. In Figure 3.18, the temperature changes were simulated from $(100-0)C^{O}$ in order to study the selection probability of non optimum nodes. The exponentially decaying temperature schedule is usually implemented [121] in accordance with Equation (3.4).

$$Ps_{(n)} = e^{\frac{(-1) * \max([0, R(n) - R(m)])}{T(t)}} \quad (3.4)$$

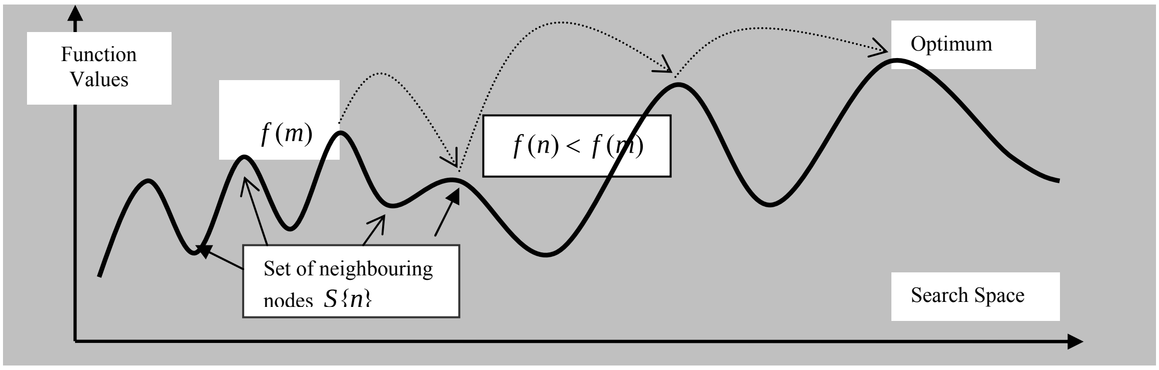


Figure 3.19: The function gets out of local minimum by selecting node with $f(n) < f(m)$. .

Suppose we are stuck at a particular node $'m'$ (Figure 3.19), and have relevant information that the current point in space is not a local optimal solution, also the function values within the local neighbourhood set of nodes $S(m)$ are all worse than the current position in the search space i.e., $f(m) > f(S_m)$. To incorporate diversity, the selection of a new evaluation point in the search space is made based on a decaying function $e^{\frac{-1}{T(t)}}$ (as defined in Equation 3.4) and a multiplicative factor $\max[0, R(n) - R(m)]$ governing the temperature decay (where $'R'$ are the available rewards at nodes) and the cooling schedule. The graph of temperature and selection based probability is shown in Figure 3.18 which reduces exponentially as the temperature is decreased (interpreting the graph from right to left). The chances of finding the optimum therefore increases significantly in this temperature controlled selection mechanism (shown in Figure 3.19).

A further graphical interpretation of implementing the SA meta-heuristic in optimization applications could be established in the context of Figure 3.20. The intensity of the temperature and cooling schedule alters two relative simultaneous processes at any given moment in time. The first of the two properties of any SA converging solutions produces a similar effect as a variable amplitude modulated signal (AM) [122], where the modulating waveform is produced by the devised heating and cooling mechanisms (e.g., Equation 3.4). The second factor of interest controls the frequency of observations carried out during a specific length of time, a period in which SA is being run to find a global optimal solution. At higher frequency a search particle could undergo many thousands of scheduled jumps in the relevant search space as shown by the zigzag lines in Figure 3.20.

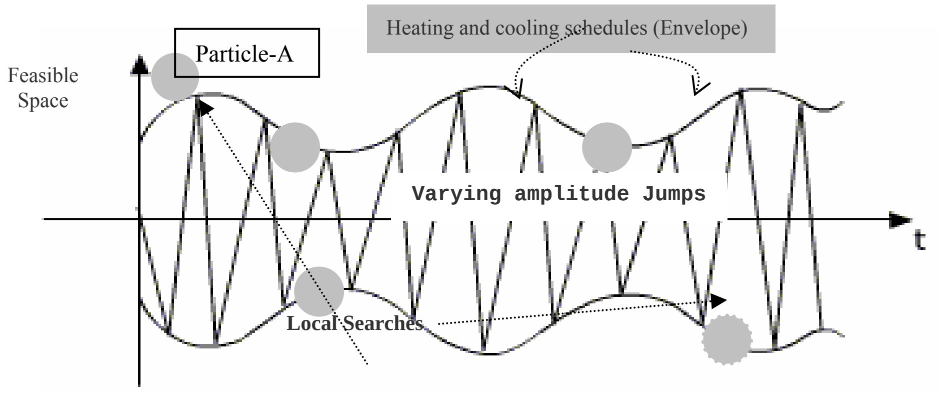


Figure 3.20: A generic representation of an amplitude modulated (AM) signal.

Figure 3.20 also demonstrates graphically how the movement of a particle/agent is governed by a number of simulated heating and cooling schedules along time dimension. The small circles represent local searches carried out by a particle after undergoing jump irregularities (represented by zigzag lines). As at high temperature, the dispersion and diffusion of molecules is much more intensive because the matter is in a more agitated state, therefore, at higher temperatures a search particle undergoes higher intensity jumps in the search space. Therefore at intensified temperature stages, the SA algorithm prefers frequent high amplitude jumps (Figure 3.20 shows several heating and cooling schedules) rather than conducting more intensified local searches (compared to a gradient method). The whole scenario involving both local and global searches therefore mimics a meta-heuristic environment. The meta-heuristic search processes would be further clarified in the upcoming sections of this report.

Despite the variations introduced by the annealing process, the SA algorithm suffers from a variety of complexities. Generally, the switching of states between the local and global searches in the simulated annealing processes require considerable planning stages and are strongly dependent on the landscape of the optimization problem. We can also relate the SA process as a variable line search technique $x^{k+1} = x^k - \eta^k \nabla_R f(x^k)$. A similar ray casting approach was applied in one of the author publication to track objects of interest [123].

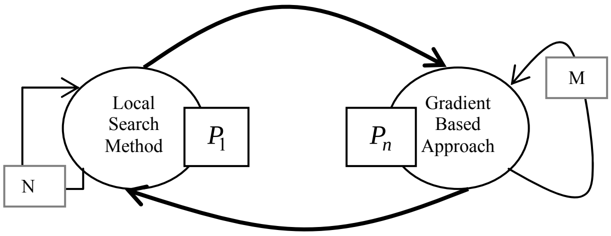


Figure 3.21: The state machine adaptation in a general simulated annealing algorithm.

Figure 3.21 is a state machine representation of the SA algorithm. Along with utilising a variety of combinations of local search methods, it must also provide parametric control (e.g., N, M) and introduce mechanism to define jumps (bold curves).

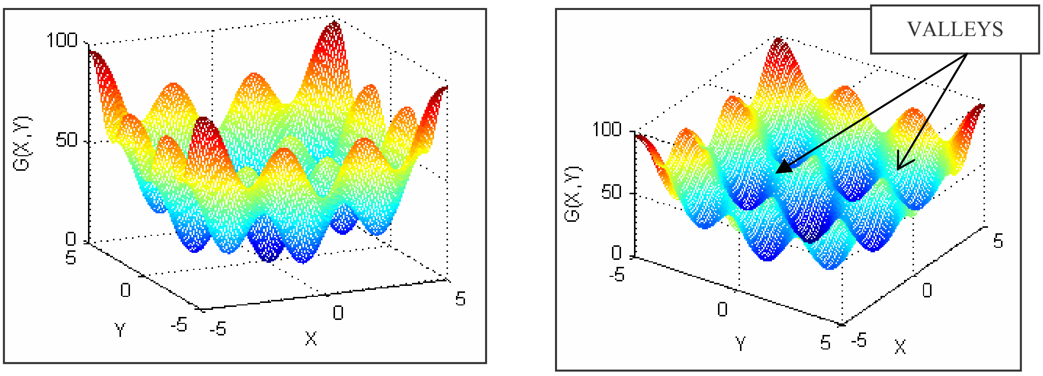


Figure 3.22: A multi-modal landscape generated by the Egg-Crate function (Matlab).

In the presence of adversities, normally the solution strategically traverses between points P_1 and P_n (two such points in the search space are shown in Figure 3.21) in an optimistic attempt to find the global solution. Furthermore, a badly tuned algorithm (where the policies behind performing larger intuitive shifts or alternatively the continuation of the current local search modes (N/M) are not comprehensively defined) therefore has larger tendencies to drift towards non optimal regions. Generally, along the course of optimisation journey, if the solution diverges far from the locus (towards the minimum), then in reality, the prospects of finding optimal solutions vanish leading to the tracking window roaming around in non feasibility space. Figure 3.22 shows the landscape of a 2 dimensional Egg-crate function [124]. There are many relative/local solutions (in comparison to Equation 3.2) to this problem but one global best, the optimal solution is located at $X = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ where the function attains $G_{(x,y)} = 0$. The contour plot in Figure 3.23 reveals the extent of the problem and maximization/minimization of the function is a difficult task due to the local ridges/valleys.

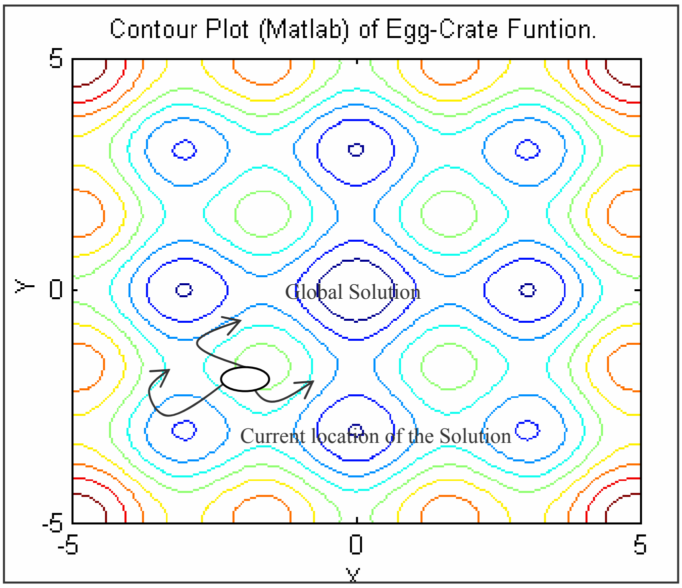


Figure 3.23: Contour plot of the Egg crate function i.e. $X^2 + Y^2 + 25(Sin^2 X + Sin^2 Y)$.

Figure 3.23 postulates an artefact which is created by the uncertainties. At the beginning, the chances to converge to any of the local optimal solution are equally probable, however, once the solution has descended or ascended much deeper into the regions (Figure 3.24 with a decision point $'D'$), there are minimal chances of recovery as the landscape (on its own) could not direct the solution towards the correct locus (path to the optimal $'R'$ in contrast to relatively incorrect $'W'$).

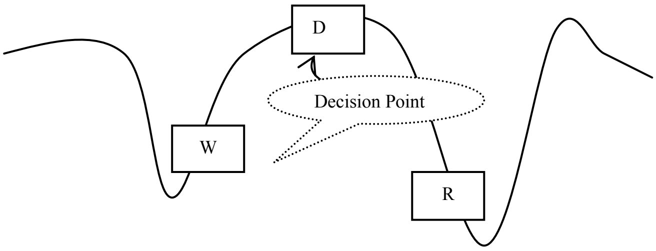


Figure 3.24: A single solution based approach traps the solution into wrong regions.

3.2.3 Case Study-Honey bird in a Meta-heuristic environment.

Before concluding this section, a case study would be presented here that which would help to rectify some of the misconceptions in relation to the higher level guidance system, needed desperately in the object tracking applications. In Section 3.2.2, it was mentioned that the embedded Meta-heuristics in simulated annealing is generally weak, and therefore has an inability to scan the environment in an effective manner. The cooperative mutual recruitment of human hunters and honey-guide (Figure 3.25) is an excellent example of both a strong Meta-heuristics and the much required control variability (as was studied in relation to the Bersteinian theory and the DOF problem (Section 2.4.1).

Figure 3.25: An artistic impression of the cooperative recruitment in honey guides [125].

The honey-bird guided hunting facilitates the much needed inclusiveness which could explicitly solve the terrain related problems. Due to the airborne characteristics of the honey bird (compared to the human hunters who get themselves trapped in the local landscape), it has better anticipation of the environment but due to its structural limits has an inability to subdue the stinging bees to reach the reward. David Attenborough (renowned for the natural world programs) exclusively presented an episode ('talking to strangers') in which the honeyguide calls to solicit human partners [126]. An extra dimension to this already supercharged drama was that the birds also responded to the humans save our soul (SOS) calls. According to Dr Claire Spoottiswoode (behavioural Ecologist at the University of Cambridge), the most remarkable fact about this human-honey guide relationship is the cooperative evolution of free-living wild animals and humans which might have spanned over the course of thousand of years [127].

To relate the honeyguide scenario to optimisation problems, we can see the experiments conducted using gradient based methods (Section 3.2.1) as the ability of human hunters, which therefore have higher sensitivity to the terrain and landscapes. The Meanshift algorithm (Section 2.54) is one further example where the absence of a meta-heuristics causes seizures of tracking windows. Therefore if a particle based guidance system is imposed over the mean shift tracking algorithm, the global convergence characteristics could be significantly improved.

The answer to the adversities in the above paragraph are somewhat contained within the problem definitions itself. Sections 2.5.1 to 2.5.3 focused around these explicit issues, that preconceived deterministic drifts could possibly deteriorate both stability as well as convergence timing in many tracking applications. What we aim to portray in this thesis is to introduce much wider search experience based on experimental variations and broader learning experiences (Section 3.1). One way to incorporate the necessary meta-heuristics is through the information fusion techniques that are common in computer science [128]. The division of landscape into sub-regions by deployment of static and roaming particles is one possibility that mimics the SOS calls in a human-honey bird relationship. By dynamically altering the complexity level, and bringing in the norms of scale-space methods (sparse to denser datasets, as was implemented in another author's publication [129]) has proven more effective tracking schemes in a variety of situations (in contrast to the deterministic methods).

3.3 Population based nature inspired algorithms.

Since the start of the millennium, scientists and engineers have been focusing onto the possibilities to enhance the capabilities of nature inspired algorithms especially in the context of non-linear mathematical problems. In reality, some of the most pragmatic aspirations to tackle complex mathematical problems come from the very simplistic of the naturally occurring phenomenon. One of the most common urge for the people of almost every generation and cultures is to witness a snowfall, especially during the festive season. Furthermore, the formation of snowflakes introduces a superficial element to the otherwise unnoticeable tiny frozen water molecules, which when ride on the horizontal/ vertical gusts of winds and thermals, create a delightful weather extravaganza.

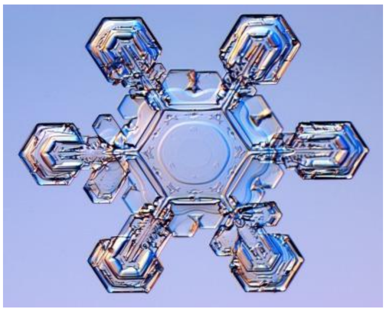


Figure 3.26: A magnified depiction of a snowflake/ ice crystal. [130].

In this whole rather artistic masterpieces of nature, the most interesting factor for scientists is to research/explore how smaller entities integrate in order to formulate larger influential structures and groups that assign them power to override the undesirable operational and atmospheric conditions. Although, the snowflakes or ice crystals are lifeless creations (the six sided snowflake/crystal is shown in Figure 3.26), but when sufficiently populated could whitewash the landscape in a matter of a few hours. There is no social element among non-living objects like ice crystals, therefore, these are incapacitated to an extent and unable to change the course of their destination (falling and ultimately to meltdown), and hence are solely dependent on the implication of external projection. The aim of population based nature inspired algorithms is however to bring in the element of a social life into an otherwise segregated particle. The snowflakes also create an emergent behaviour but without a much needed social element, which when is explicitly introduced into the landscape gives birth to a complete different dimension and search expeditions. Utilising the gifted imaginations and following the hierarchical discipline birds could create larger influential groups and versatile flocking behaviours (Figure 3.27), that could even baffle local authorities into spending treacherous resources and research time to deal with the problem [131].

One example of the research of emergence phenomenon is the Strarflag project [132], which primarily aimed to deal with the nuisance (due to the ever growing population size to what Rome could dwell) and antisocial behaviour (human perspective) of starlings that could change the landscape of this modern city within a matter of seconds. There are other uncountable examples in terms of the swarming behaviours observed in the social insects including the travel journeys of Monarch butterflies, schooling fish, bee hives and ant colonies.

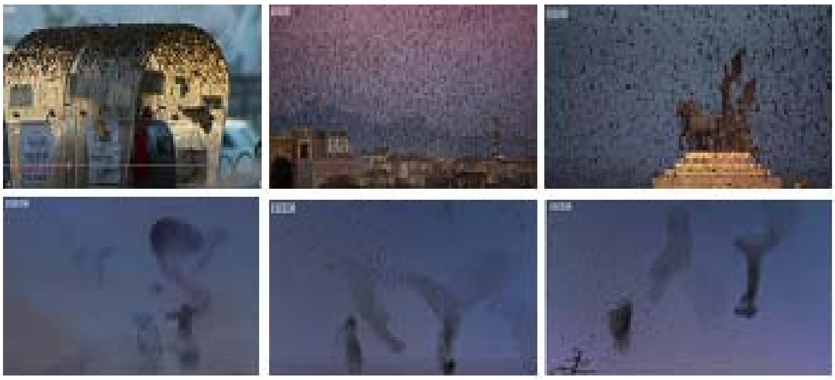


Figure 3.27: The display of nuisance and spectacular starling formations and acrobatics [132].

Figure 3.27 shows some of the excerpts from one of the latest documentary presented by Sir David Attenborough (‘cities, planet earth 2’ [133]) filmed at Rome in Italy. The top row shows some of the landmarks and monuments that are being damaged by the growing population of starling, and the bottom exhibits the extraordinary features involving the group disciplines which also vary in the population sizes and underlying structures. Each starling formations seem to be competing with rivals, and usually take place around the sunset, whereas, during day times most of the starlings fly around randomly producing an effect of a noise contaminated images. In the last few decades, many similar population based optimisation techniques have emerged that also aim to address these core issues [134] [135].

3.3.1 The evolutionary and swarm based optimization methodologies.

Since the start of the millennium, there has been an endeavour by scientific community to nurture the evolving branch of evolutionary mathematics [136]. The lack of coordination and contemptuous attempts have resulted in a paradox promoting a thinking that evolutionary [137] and swarm based optimization methods [138] are fundamentally distinct branches of evolutionary computation. An evolutionist (Godless thinking) believes that all living beings are constantly changing in order to adapt the trait that revamps them to compete against the threats imposed by the potent environment. Whereas, the elemental belief of a creative thinker comes from an idea that all creations are purposeful inventions, and are contraption free products of a supernatural force with no past genetic linkage whatsoever [139]. Nevertheless, as this thesis is not about the ethical behaviours, in engineering the concept of evolution must not be confined to its natural counterpart.

Moreover, an evolutionary algorithm is a progressive methodology, which when is least wilful, provides an environment that promotes to collate opinions from versatile school of thoughts. The closest association of coordinated behaviours in lifeless objects has been found in ferromagnetic materials [140], where particles (spinning electrons) align themselves (at critical temperatures) and form strong interconnections in order to enhance the fundamental properties of the matter [141]. One of the most mesmerizing applications of self learning (prompting collective response) could be seen in the twisting and morphing cloud of flocking/swarming starlings.

The swirling masses of clouds created by murmuring starlings is a partial revelation (similar to the automated mechanism design (AMD) [142]) integrating both distributed and centralised intelligence, and delineates that the strength is in the unification and disciplined rendering (see Figure 1.1). According to the findings of the Starflag research project (partly supervised by Giorgio Parisi, a physicist from the University of Rome), the abilities of starlings to form unique structures, that sometimes evolve from within an already existing cloud is the result of the scale free correlations, the possible physics and the mathematical findings of this group have also been published [143]. The murmuration of starling and the resultant intelligent cloud is certainly more than a freak show, and a critical thinker may infer that this marvel of collective genius could be in fact a deterrent to scare and to keep at bay powerful opponents like the Peregrine Falcons.

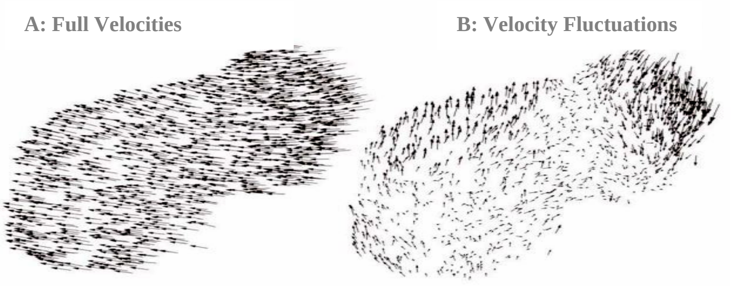


Figure 3.28: The two dimensional projection (velocity vectors) of the starling flypasts [144].

The possibility of territorial warfare among rival starling groups (resulting in such exhibitions of highly coordinated flight patterns) may not be completely ruled out, and one line of thought is that, it is merely a recruitment strategy in order to further grow in size. Through analysing hours of recording of swarming and flocking behaviours, we came to a similar conclusion that murmuration is just one of the modes of starlings in flight. One of several different occasions, the flying birds mimic swarming behaviour without any higher order thoughts and need to create a disciplined flypast. In Figure 3.28, the two dimensional projection of 3D velocity vectors identifies two non-identical flight patterns, in the first case, the velocity vectors of the whole flock seems to be highly correlated and form a simple swarming effect, whereas scenario B shows the presence of a significant level of diversity and the birds appear to be in murmuring phases.

Whether the aesthetic formations of flocking birds, schooling fish or even swarming Monarch butterflies are the products of scale free correlations, or of neighbourhood velocity adjustments [145], the computational charm lies in answering the question that how inferences from these natural agents could alleviate convergence to the trivial search areas in an optimization problem. It would be impartial to write that the most prudent and problem solving technique is still the original (1995) work of Kennedy and Eberhart in the context of particle swarm optimization (PSO) [146]. Apparently, the original authors were quite aware of a possible deluge, and therefore in order to set the sequel and scope of the problem have discussed several hypotheses and some were strengthened through benchmark testing.

The original PSO method is highly intrigued by the development in human cognition by means of social interaction [147]. The work of Kennedy-Eberhart (KE) could also be seen as the continuation of the findings of sociobiologist E.O.Wilson [148]. According to Wilson, the collaboration of individual knowledge (in a school of fish) to discover new sparsely distributed food resources far outweighs the disadvantages of the competition in order to gain a fairer share from the hunt. A primitive conclusion from the work of KE is a short excerpt [149] "*social sharing of information among conspeciates offers an evolutionary advantage",* which also suggests some grass-root level relationships among swarm and evolutionary approaches as predicted earlier in this section.

Furthermore, KE explicitly mentioned in their paper, that problem solving abilities in physical space in human beings (e.g., the collision avoidance techniques) are learned at a very early age but form only trivial component of the overall psychological experiences, and PSO is thus only a simplistic coding of the social milieu of the flocking birds. The history of social development in humans goes much further than the unpredictable choreography of the murmuring birds [150]. On one side, the work of KE explored the repercussions of using the Cornfield vectors alone (when all the wandering birds are well informed about the location of a cornfield) introduced in the flocking simulations by Heppner [151], and then studies the aftermaths of introducing an element of craziness/madness which prohibits the particles to collapse in a short space of time (with a primary aim to enhance exploration). The insertion of the charged particles (magnetic fields to discriminate searches [152]) giving the exploring agents tendencies to be attracted by the opposite genders (e.g., negative charges) are (as a matter of fact) the same cornfield influences used by Heppner to formulate a tightly knitted group with genetic orders. Mathematically, we can write the PSO algorithm as shown in Equation (3.5) [146].

$$v_i(t+1) = av_i(t) + \Phi_1 r_1[\hat{x}_i(t) - x_i(t)] + \Phi_2 r_2[g_b(t) - x_i(t)] \qquad 3.5$$

It is evident from Equation (3.5), that the velocity control of particles (represented by the index i at time instant$t+1$) is provided through the linear combination of the inertial, cognitive and social elements of the trajectories. The first term in Equation (3.5) is an inertial factor and when scaled by the variable$'a'$, generates a variety of preferences in terms of how further any particle should travel along its present search direction. The second and third components of movement (cognitive and social factors) influence the particles to exhibit a tendency to converge towards their own personal best findings and an assimilative factor respectively, which influences the particles to exhibit the desired level of group empathy. The velocities could be amplified or dampened by $\Phi_1$and Φ_2(which are known as tuning parameters and usually range between$0-2$) and $r_1/r_2$are the stochastic components to introduce variability.

The phenomenon of emergence and murmuration (unification of smaller entities to create larger structures) is inherently absent from the PSO (of course as the name implies) and is merely a swarm of particles. Furthermore, the dynamic environmental conditions impose a demand for versatility and are much harder to be controlled using static means (Equation 3.5 and using Φ_1 and Φ_2). The changing global best assignments $g_b(t)$ further deteriorate the problem and a large population is needed to avoid convergence to sub-optimal regions. Moreover, the nostalgia in PSO (personal best position $\hat{x}_i(t)$) is memory intensive task and demand amendments.

Therefore in order to reduce convergence timing, a strategic control of tuning parameters is mandatory, otherwise the solution produces an oscillatory response (in trivial regions) prohibiting the particles to ever settle down. Along with definitions of allowed maximum velocities $V_{\max}$ (so that the particles do not jump out of the search space boundaries/limitations [153]), the strategic dampening of speed (and velocities) is crucial so that particles could evolve into the next phases (in an evolutionary sense). The next section is aimed to rectify the inherent flaws in general swarm based methodologies.

3.3.2 The role of mortality in the ecosystem of particles.

Researchers have found evidences that many natural processes are designed to counterpoise detrimental effects (of one another) and generate timely medicaments to neutralise, therefore creating an atmosphere of harmony in which life flourishes [154]. Many man made machinery has also been influenced by these natural manifestations (e.g., the rear rotor of a flying helicopter counterbalances the rotational forces exerted by the main rotor [155]). One of the key demonstrations of this natural balancing lies deep down under our feet and into the molten outer core of our planet (Figure 3.29). The change in magnetic fields created by the structural variations of molten core and super-rotations (rotational speed changes) of solid core (which rotates in an opposite direction to earth layers) formulates a geodynamo, inducing billions of amperes of electrical current. The resultant magnetic field created by earth's internal processes helps to keep earth's orbit, and therefore preventing our thin atmosphere to dissolve into the space [156].

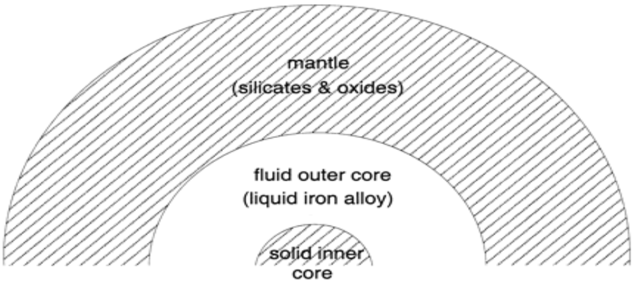


Figure 3.29: The antidote created by the super-rotation of the earth solid inner core [157].

The advantages of the introduction of an antidote in a solution space could be observed in many other circumstances as well [158], the eradication of wolves from the Yellowstone national park resulted in an unprecedented escalation of the Elk population which wreaked havocs on the surrounding ranches. Therefore, since 1995, eco-scientists and biologists are trying to reinforce the balance (in this particular ecosystem) by manual injections and monitoring of the wolf populations [159].

The essence of the earlier discussion (both natural and geological) is to understand the role of the convergence phenomenon and to investigate and develop a possible antidote in order to balance the ecosystem of particles. As any particle is the prevalence of a hypothesis/belief in a search space, hence the converging particles could result in a diversity loss, therefore without devising suitable techniques to address these ramifications would consequently lead the population into very confined search regions. With the beliefs collapsing onto narrow hypothetical areas in space, the universal impact of this crowding is similar to the gradient/analytical based methods (Section 2.5.1).

The utilization of the particle based methodologies (inspired by natural entities) to solve complex mathematical problem requires a meticulous know-how of both the morphology (structure) and physiology (behaviour) of living entities. All living organisms have individualized genetic encodings (genome) which is the product of two complicated processes known as the crossovers and mutations. The genetic crossover is the transition of genetic material into the offspring [160] and culminates the morphological as well as physiological features of the child population. The appearances of certain characteristics also depend on the dominant and recessive alleles (e.g., blue eyes have a dominant allele, whereas, blue eyes need two signatures in order to appear [161]) and despite of the presence of a specific element in their genetic material might not prevail morphologically in the next generation.

The Heredity or passing of genetic information into the genome of a child is an error-prone procedure, the crossover defects (also known as mutations) are one of the consequences of this transformation. Some evolutionary computational techniques (e.g., genetic algorithm (GA) [162]) therefore introduce some forms of random mutations to reflect this natural process. Along with the microbiological imperfections that arise during crossovers, some elements in the genome are environment perturbations which after going through micro and macro stages of evolution create the phenotype traits (behavioural and structural) of a species [163]. Along the course of the lives of living organisms, the interactions of individuals with their surroundings also affect their genotype (set of genes) in such a way that enhances their natural fitness abilities (similarly to an antidote). The depositional effects of mutation and the retentions of a simple crossover of genetic material are shown in Figure 3.30. Although the child inherited and retained some of the genes from both of its parents but the effects of mutation have created a different genetic structure.

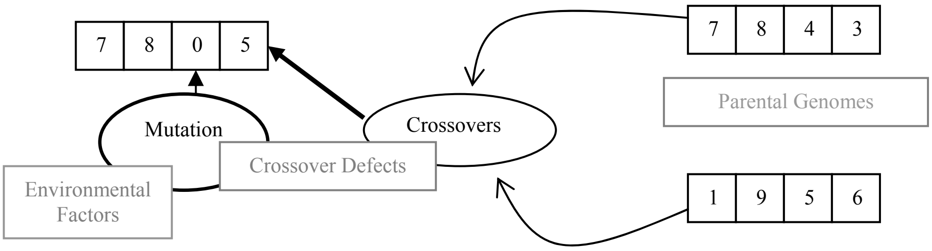


Figure 3.30: The effects of crossovers and mutations in a child population

Most particle based methods (e.g., PSO, Evolutionary Strategies (ES) [164]) address only limited physiological trends and starkly overlook the important morphological features commonly observed at both individual and social levels in both microscopic and macroscopic life forms. The changes in swarming starlings into murmuring also emphasize a transition in morphological structures, and this collective evolution could be the result of the radical roles within the population. It is pivotal to seemingly program and to implement the phenotype changes (in particle generations) in real vector space (e.g., camera plane and multivariate functions) and to diversify in a compelling but timely fashion after analysing all the tell-tale signs (e.g., data fusion [165] and applying scale space changes [166].

One of the most fundamental computational techniques that apply the rules of natural variation and recombination is the Evolutionary Strategies (ES) [167]. The ES method does not explicitly apply the natural selection but relies heavily on the random Gaussian processes [168]. Therefore, all particles in ES have a possibility to be selected as parents. In contrast, in EA, parents are chosen from a genetic pool based on their fitness levels (an example of fitness proportionate selection is roulette wheel selection (RWS) [169])

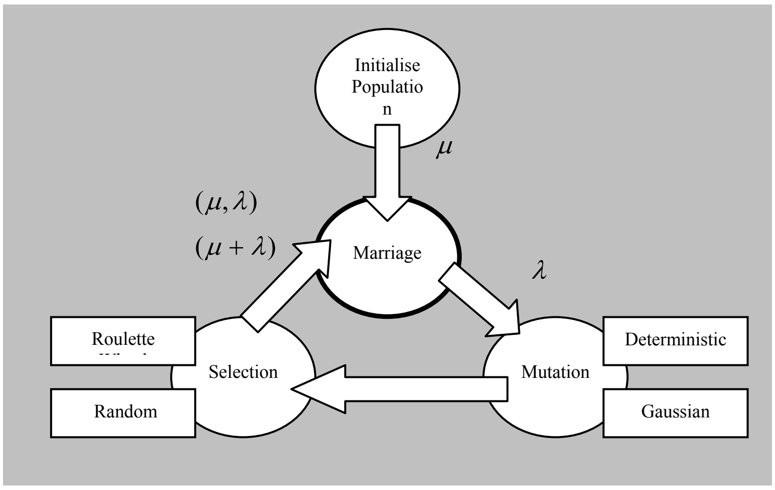


Figure 3.31: The cyclic reproductive processes in the evolutionary computational algorithms.

Figure 3.31 is an umbrella representation of the class of computational algorithms inspired by the natural evolution, and usually consists of three main processes commonly referenced in literature as the marriage, mutation and selection. Proscribing the initialization debate here (see Figure 3.16), a generation run of EA starts from the μ parental population, the marriage operator reproduces λ offspring which after going through diversification and enhancements become mutated individuals (MI). The fitness of all MIs is retested and some are chosen to be married in order to generate next child generation. In ES all individuals (whether children or parents) are represented in the form of real vectors and controlled by the endogenous and exogenous processes [170].

The endogenous parameters expedite mutations to increase the fitness level of an individual. The endogens also explicate the means, variances and correlations in Gaussian mutation (to grant the evolution a sense of direction), whereby, the exogens tend to proclaim the predominant reproductive characteristics (e.g., birth rates and selection techniques). One key exogenous control is to define the genetic pool from which future parents are selected, the (μ,λ) type of selection restricts the gene pool to individuals from the offspring population λ only and the parents μ are forgotten regardless of their fitness levels, whereas in the $(\mu+\lambda)$ reproduction both the parents and children have an equal probability to be chosen in virtual marriages.

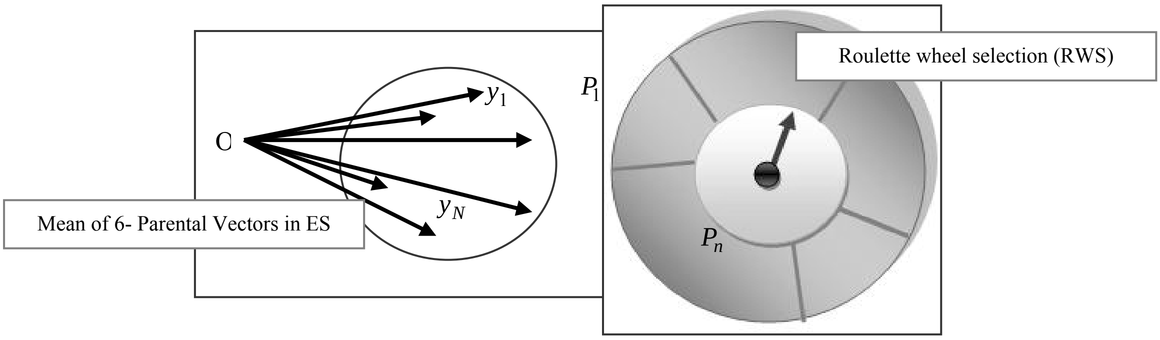

Roulette wheel selection (RWS)
P_1
y_1
C
y_N
Mean of 6- Parental Vectors in ES
P_n

Figure 3.32: Two commonly used parental selection techniques in ES and EA.

On the left of Figure 3.32, $y_{k=1:N}$ are N vectors in the R^n dimensional search space and all have an associated fitness $F(y_k)$. All vectors are potential solutions to the optimisation problem and are stored in the memory in format $c_k := (y_k, s_k, F(y_k))$, where s_k are the endogenous strategy parameters associated with each solution in the search space.

Figure 3.33 display only four such Gaussian mutations (however in practice there could be a large variety of s_k) which are asserted using the endogenous strategic definitions s_k affiliated with each vector c_k. The relationship (to retain a Gaussian like hump) between standard deviation and the strength of mutations are presented in the graphs (Figure 3.33) using four unique mutation operators (a-d). To incorporate wider genetic diversity in tracking/vision algorithms, the mutation vector for particles could be randomly sampled from a variety of such normal shaped distributions.

Furthermore, in order to diversify the searches for any specific agent/particle, each vector y_k might also be prescribed with time contingent features. Therefore, the endogenous strategy parameters are also varied in time (mutates the mutation strengths) to introduce the relevant transitions in space for each and every particle in population. Especially, for converging behaviours a purposeful and diminishing mutation becomes statutory, or else the particles would hop out of the feasible areas of the search space.

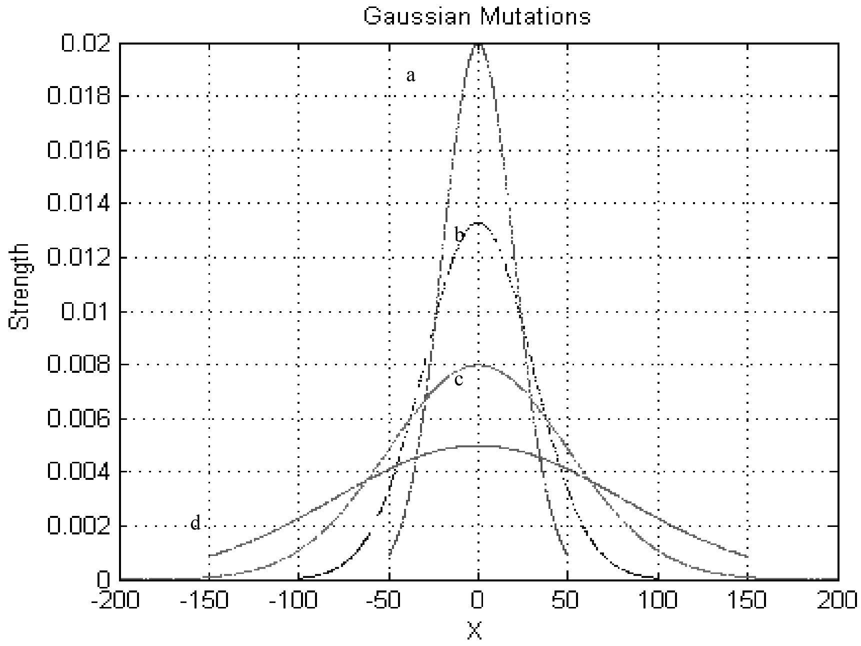


Figure 3.33: Defining four endogenous strategy parameters $N_1(\mu,\Sigma)$ to $N_4(\mu,\Sigma)$ $y_{k=4}$ vectors.

$$c_k = \frac{1}{N}\sum_{i=1}^{N}(V_i)_k \qquad 3.6$$

The graphical depiction (on the left) in Figure 3.32 is expressed mathematically in Equation (3.6). The strength of a specific mutation is chosen by calculating the mean of $'N'$ random mutation vectors V_i. The Σ operator is used here to represent such linear combinations, and therefore the magnitude/strength of this newly created mutation is the average of all relevant vectors $v_{i=1..N}$ in Equation (3.6). Similarly, in multi-parent recombination (to generate a genetic crossover effect in the ES), a child vector is reproduced by calculating the mean (Equation 3.6) of all the parental vectors. After the addition of a random mutation, the mean of vectors (a child centroid c_k) would be shifted again in order to further enhance the exploration of the search space.

The RWS on the other hand exerts in a more disparate fashion as the parental fitness levels are scaled on a virtual roulette wheel, hence the fitter individuals occupy a larger proportion of the space, and therefore have more chances of being selected in a spin. In Equation (3.7), the probability of selection of a parent p_i is determined by dividing the fitness level of the individual f_i with the compound fitness of the generation $\sum_{j=1}^{N} f_j$.

$$P_i = \frac{f_i}{\sum_{j=1}^{N} f_j} \qquad 3.7$$

There are untold variants in literature and largely differ in a manner in which the parental features are transplanted onto the future generations. In differential evolution (DE) of Storn and Price [171], the binary crossover of a particle X (current solution) with intermediate agents (calculated by linear projections of 3 randomly selected agents) is used to scrutinize the search space. Whereas in stochastic diffusion searches (SDS), the random experiences of particles/delegates are used to congregate around the best regions in the space and future trajectories are just opinionated decisions [172].

We have reached a climactic stage here postulating that a maverick particle physiology is possible using wider learning experiences. Furthermore, a detailed analysis is needed to determine whether forming particle log offer any optimization advantages at all. Most of the particle based methodologies (e.g., PSO [173], DE [174]) rely on registration of searches. The idea of non-nostalgic scale free explorations could be augmented with human experiences. Generally, as the confidence in acquisition of new skill increases we utilise fewer memory which also eases the cognitive pressure/overload. Similarly, it might be possible to reduce the complexity of particle methods using some radical, novel and record free phases.

3.4 Radical Search Optimisation (RSO)

In the earlier sections some key challenges in the optimisation of multi-modal problems are discussed. It was learnt through experiments that a higher order guidance sought through the monotonic information (embedded in the function domain) might only lead to the local optimal solutions. Hence, in highly rippled cases (e.g., imaging frames) a suitable emphasis on forming top order meta-heuristics is needed to resolve short sightedness problems (as in Figures 3.3 and 3.4). The adversity introduced by assuming that particles are immortal agents (similar to a natural colony) is previously discussed in reasonable details. Therefore, a more balanced ecosystem of particles could be created when particle rebirths (and a revolutionary dynamics) are introduced to overcome the crawling natural evolution. By building upon the already set foundations (established in Sections 3.1-3.3), we are in a better position to formally discuss the novel radical search optimisation (RSO) discovered in this research. The vital properties and key characteristics will be discussed (in this section) that could accomplish fastest and more efficient solutions than the conventional methods.

Before going into further detail, it might be useful to gain some acquaintance with the word *'radical'*, which is frequently mentioned in psychology and behaviourism science. The market based research carried out by Norman [175] associated the radical phenomenon with the introduction of more innovative approach in the product development (rather than the conventional incremental changes in an existing design). The radical social reforms are also more unconventional and thorough in nature, and exhibit far reaching effects in diversified human societies [176]. In the political arena, radicals and reformists are two widely debated groups (and political sympathizers) that try to improve governance styles for the benefits of general public [177]. In Cambridge English Dictionary the term radical is defined as a trait that seeks greater, extreme and wide ranging changes in a variety of disciplines [178].

In the context of the optimisation procedures to track an object of interest, the radical search optimisation (RSO) refers to the strategic grouping and deployments of radical particles. A radical particle is an autonomous computational agent, and it does not believe in social captivity induced within colonised group of particles, and generally follows its personal intuition and judgements to explore a search space. Moreover, unlike conventional particles in PSO; the radical particles are not governed by any hard-wired logic, and hence, all radical particles perform scale free searches without any neighbourhood confinements shown in Figure 2.8. Any keen reader is referred back to Figure 3.28, and to the discussion (presented on pages 79-80) using Giorgio Parisi research work regarding spatial correlations of murmuring starlings. A RSO is therefore a collective characteristic of a population of radical particles undergoing broader and innovative movements in a solution space.

We have reached a pinnacle stage, and the optimisation benefits using RSO will be methodologically tested using evolutionary test bench problems. To achieve a real time goal set in this research, and in order to process visual data at 25+ FPS, we need to clampdown on the computational overheads in the evolutionary approaches, and therefore a more functional contingency (forward looking) plan is needed in the computer vision algorithm.

The selection-variation antagonist (applied in the computational evolution) although drafts in some of the leading features present in Darwin’s theory, but an impeccable question is that whether these processes are the real and factual antidotes we witness in the naturally engineered projects seen in nature. In a visual dimension the bulk of computational time should be focused towards discovering macro changes observed in the manoeuvring and translating objects, and therefore biological evolution in particles has limited applicability (e.g., to analyse a linear dynamics only) and may not be a goal oriented and practical approach. A case study is presented in Section 3.4.1 that could enable us to understand scale free searches (using RSO) in a practical computer vision environment, and to understand the problem within an optimisation framework.

3.4.1 A case study involving radical and scale-free searches.

A common approach in both Lagrangian [179] and Eulerian [180] formulations in contour based tracking is to assemble data points in such a manner that tracking flaunts the distinctive properties of a moving front (e.g., the flame propagation in thermodynamics [181] as shown in Figure 3.34). A fundamental question that may surface (within the natural comprehensions of a critical thinker and scientist) is to challenge the restrictive calculations enumerated as narrowband along the curve's surface.

Whereas, we can facilitate scale free searches using computational agents to immediately reveal the boundaries of an object (shown through arrows and RSO particles in Figure 3.34b). One key focal point in this discussion is that the curvature reduction measures [182], and the computational lethargy introduced by generating the coordinated movement styles are not mandatory during intermittent searches (of an evolving contour). Therefore in order to segment a moving object in time, such denser datasets should only be introduced in the closer vicinities, and only during the final evolution stages (see Figure 2.10).

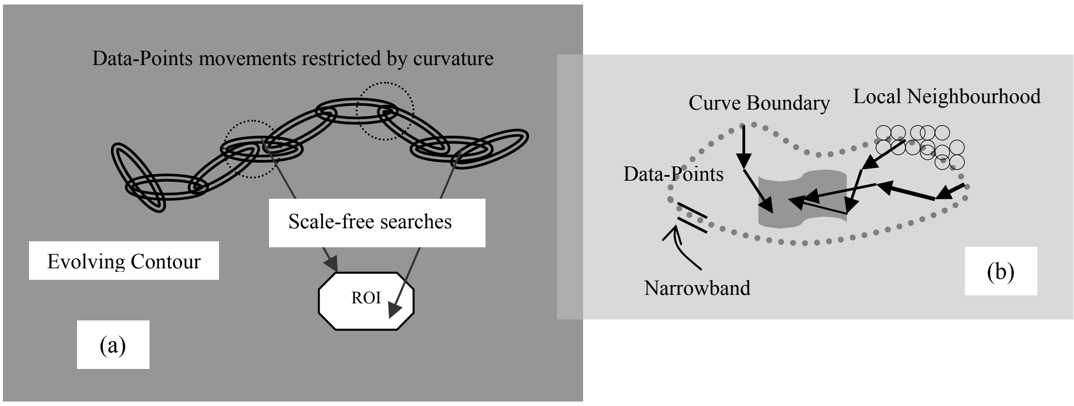


Figure 3.34: Scale free searches in contour tracking to reduce evolution time.

In Figure 3.34a, the region of interest is quite distant and further away from an evolving contour (shown by using an analogy of the metallic rings). The ring structures (created by imposing conditions in terms of curvature of the curve) generally prohibit the contour to evolve (as a whole) in the direction of the object. A scale free search (on the other hand) is not bounded in a confined domain, as there is no real benefit in selecting a neighbourhood in the first place (shown by a number of small circles and dotted lines). The scale free searches in this detection mode do not restrict particles using narrowband (similar to foraging bees departing from a hive in search of more suitable sites), and this concept is portrayed in Figures 3.34a and 3.34b. Hence, the domain of applying precision oriented segmentation within a narrowband is limited (e.g., in medical imaging), and in the view of the author is a major misconception requiring calculations of explicit and contradictive energy terms (e.g., smoothing movements to reduce curvature and curve length, and feature based energies that drive a contour towards an object of interest), and this issue has been starkly overlooked in the past. One fascinating idea is to incorporate visual diversity using RSO as a (separate) meta-heuristics over an evolving contour to attract it (in a more meaningful fashion) towards an object boundary.

3.4.2 Basic characteristics of a radical particle (RP).

As the title of this thesis suggests, the natural inspirations are at the core of the tracking methodology in this report. One of the main focal points therefore is to understand and investigate the behaviour of many natural world foragers (bees, ants, fireflies, monarch butterflies etc) from everyday occurrences. All natural foragers and hunters possess an extraordinary ability to integrate measurements (e.g., using visual and olfactory senses) and commonly apply those in trajectory planning, these paths may be composed of completely different walks and flight patterns than their peers.

Figure 3.35 portrays graphically the main features of an elementary radical particle proposed in this thesis. Similar to the dances performed by foragers at hive dance floor after conducting independent walks and flights, a radical particle only registers its searches after a significant breakthrough in their hunts. Therefore majority of the searches performed by radical particles are in fact processed in virtual modes, and therefore these agents are termed as virtual particles (VP) in this thesis. Some promising search strategies in this context could be uniformly distributed random walks or flights sampled from parametric Gaussian (or Levy distributions mentioned earlier). Several anthropological studies also concluded that 21st century hunter-gatherers exhibit levy characteristics in their search patterns for bush-meat (like their natural counterparts [183] [184]).

On several occasions, the searches appear to be signal modulated meaning that the search parameters are altered using an information fusion process (e.g., honey bird scenario). Hence, versatility in searches during hunting phases is the backbone of any successful run as depicted in Figure 3.35. The penalising ability is the prime characteristic of a RP; it integrates a wide variety of opinions in order to accept/reject object detections. Such hypothesis pruning is fundamentally similar to the nest site selections (NSS) in medium sized ant colonies [185]. All radical particles can glance beyond their physical position by taking virtual measurements, and could undergo spatial transitions based on a variety of sampling techniques. The nested behaviour is based on a rebirth phenomenon where inner searches are guided by an external process.

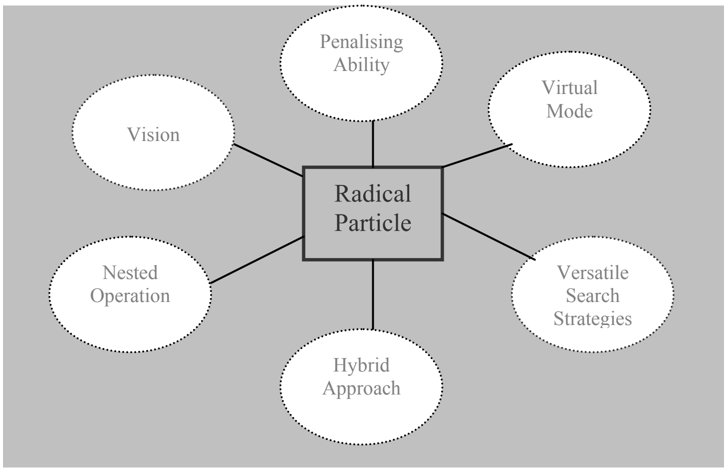


Figure 3.35: The characteristics of a basic radical particle.

The guided search [186] is a novel technique that has never been applied in the computer vision algorithms before (and further explored in Chapters 4-6), it formulates several discriminatory criterions to accept a hypothesis that a region belongs to an object. Only those regions that pass the penalising test undergo further investigations in order to establish them as legitimate targets. A radical particle is a hybrid technique in the sense that it applies both registered and undocumented phases to attract peer attention.

3.4.3 Nested operations in RSO.

A brief synopsis of the developments presented in the earlier sections of this thesis is presented here (for the benefit of the readers). The radical search optimization is developed on the particle sovereignty, and neither is controlled by the velocity models (which demand extensive memory operations), nor any deterministic drifts are used to scrutinize the search space (Sections 2.5.1. 2.5.2). The RSO does not require parametric tuning; therefore it has an overwhelming advantage over the PSO due to their higher algorithmic efficiency and stability in unknown test cases (increasing portability of a solution), as usually minor perturbations in the values of tuning parameters Φ_1 / Φ_2 in Equation (3.5) could forge substandard convergences.

Furthermore, there are no social elements in the RSO, and it does not rely on the empathetic social calling of agents, where all individualised solutions are mortal in nature (meaning that all particles are eliminated while testing a projected hypothesis). Due to the higher search motivations and capacities, RSO could also be formulated as a meta-heuristics (see Section 3.2.3) over the traditional swarm methodologies (this axiom would be explored in Chapter 6). One key property of RSO is that the particles do not converge themselves, but in practice, the search space narrows down itself using successive approximations provided by the scale-free radicals. These successive approximations and discoveries are nested explorations (we called scanning phases) which help to guide the inner solutions towards a more feasible space (Figure 3.36). The outer loop generally formulates a meta-heuristics to guide the inner loops to reduce the convergence timings.

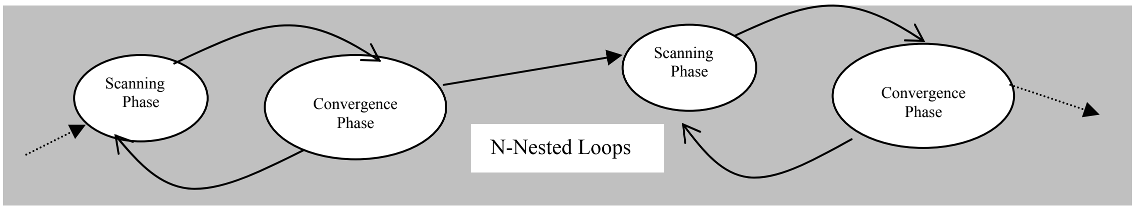


Figure 3:36: Two nested RSO phases used to detect a global optimal solution.

Table 3.1

	PSO	RSO
Particle Vision	None	Yes
Social	At all times	No
Mortality	No	Yes
Rebirths	Never	After each iteration
Nested	No	Yes

Therefore, RSO only require the declarations of the counter variables rather than extensive tuning mechanism applied in PSO. In complex multi-modal mathematical problems, a higher counter value generally enables more rigorous and thorough outer searches. The successive particle placements and rebirths after rigorous scanning phases therefore tend to solve the usual premature stagnation problem (witnessed in the traditional particle based methodologies). Moreover, at an instance in time, the counter variables and identification of a feasible space (using mean and variance) are generally adequate to find a global optimal solution.

Based on the discussions presented in this section, we are in a position to present the feature based comparisons between PSO and RSO (as shown in Table 3.1). All particles in RSO have a visionary radius and therefore unlike PSO could glance much further than their current positions in the search space. Each particle is subjected to mortality in RSO, and further exploitation of a projected hypothesis is performed through intelligent placements and rebirths (to avoid the costly translations in PSO). To find out whether there is any social advantage in PSO is the goal of the experimentation part (Section 3.5) in this chapter. The nested convergence process is discussed earlier with the aid of Figure 3.36, and could facilitate in guiding the solutions out of local traps. Further tests would be conducted in Chapter 6 where radical particles are deployed in video frames to guide PSO towards an object of interest.

3.4.4 Assigning search policies in RSO.

The vision characteristic in RSO enables particles to work in non tactile modes; therefore particles are well aware of the surrounding landscape. As this maverick particle psychology (in RSO) is based on the natural hunting patterns (observed in the biological life) it is much simplistic and a common sense approach (e.g., in comparison to the particle filters). The particles are strategically placed in the key areas exerted by a meta-heuristic process, and after going through several diffusion phases nominate particular areas in solution space for further testing as shown in Figure 3.37, and after several independent diffusion phases, particles are directed immediately within these regeneration areas (rather than evolving the entire population in the search space).

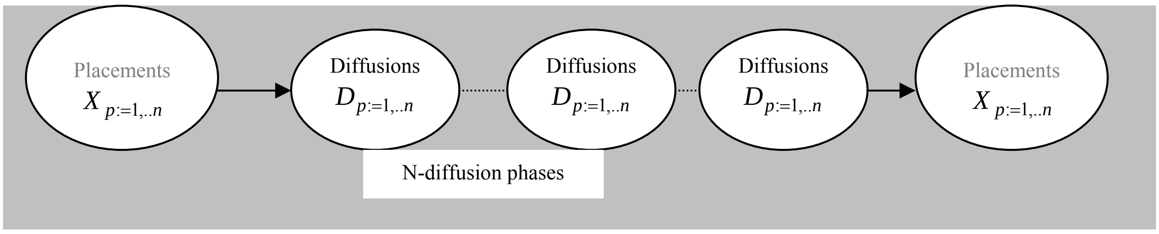

Placements $X_{p:=1,..n}$
Diffusions $D_{p:=1,..n}$
Diffusions $D_{p:=1,..n}$
Diffusions $D_{p:=1,..n}$
Placements $X_{p:=1,..n}$
N-diffusion phases

Figure 3:37: One specific schematics/arrangement of placements and diffusion phases in RSO.

Algorithm 3.1: RSO

```
1: [Placements] - $x_{p=1..n}$ Distribute n-particles in the search space using meta-heuristics.
2: [Olfactory State (a)]-Assign each particle with a local/visionary search radius $R_n$.
3: for each particle in the Search Space ‘S’
4: [Designation]-Allocate $P_{1..n}$ their unique search strategies $P_{s=1..n}$
5: [Local Searches]-Conduct local searches in visual proximities defined in lines 3-4.
6: [Current Best Solution] - Choose the current best solution $C_b$.
7: [Local Perturbations]-Apply k perturbation $L_k$ to discover the best solution.
8: [Hypothesis Selection] - If matching condition met then deactivate rest of particles.
9: [Inner Searches] - Modify search space parameters to define new feasible space.
10: [Regeneration] - Apply particle rebirths in the regeneration region found in line-9.
11: [Olfactory State (b)] - Modify $R_n$ in order to apply inner searches.
12: [Convergence] - if $|g_b - c_b| < T_g$ then
13: [Break] - Export the best solution as the global optimal solution and terminate
13: [Re-Selections] – else if convergence conditions failed at line 12 then
14: [Reinitialization] - GOTO line-1, re-organise particles using higher level heuristics.
```

The RSO algorithm (in a pseudo-code format) is shown in Algorithm 3.1. The prime focal point that makes this approach quite unique is the designation of local search radiuses (Line-2), and usually undocumented searches (like natural foragers) are carried out in a virtual mode. In the tracking algorithms, particles are particularly placed using intelligent meta-heuristics (using frame subtractions and scale space methods). Instead of an inherent/hardwired social structure as seen in PSO, all computational agents in RSO hop around (in feasible space) using pre-allocated strategies, e.g., levy flights and using a mixture of Gaussians and random perturbations (Lines 4-7). A list of best hypothesis (usually<5) are meticulously analysed (Steps 8-13), and the search is terminated when a specified threshold condition is met (Line-12). Another differentiating aspect (of RSO) is that instead of particle convergences (in PSO), a rebirth phenomenon is applied here (Lines 9-10). Moreover, only activated particles are used within inner searches (Line-8) which also saves a lot of computational complexity.

3.4.5 Individual and global properties of RSO.

In this section the three main properties of novel RSO algorithm/technique would be discussed. The idea of a revolutionary dynamics (instead of slowly incurring changes through an evolutionary process) is of pivotal importance in radical behaviours explained in the earlier section. The overall approach is primarily deduced from the murmuring phases observed in a starling population (and explained earlier in the context of Figures 3.27 and 3.28).

A-Zero transitional energy during converging/dispersing phases

In many optimisation scenarios, the painstaking exposition (requirement of a strategic control by the parametric tuning) of converging particles towards a potential solution is purely a simulative gesture (in the view of this thesis is a redundant optimisation feature), and is based on one weak assumption that computational particles are indispensable family members of a natural colony (e.g., bee ant ant). On the other hand (although less empathetic towards the collective well being of any natural colony) if particles are treated as mortal elements, the relocation expenditures of thousands and thousands of particles could be eradicated which could expedite tracking. Therefore, the decimation of such tenacious features mainly arising from the presumption of the permanent and generational particles (in both evolutionary and swarm based methods), and employing the provisional or task oriented computational agents instead is a more resolute option in the view of this thesis.

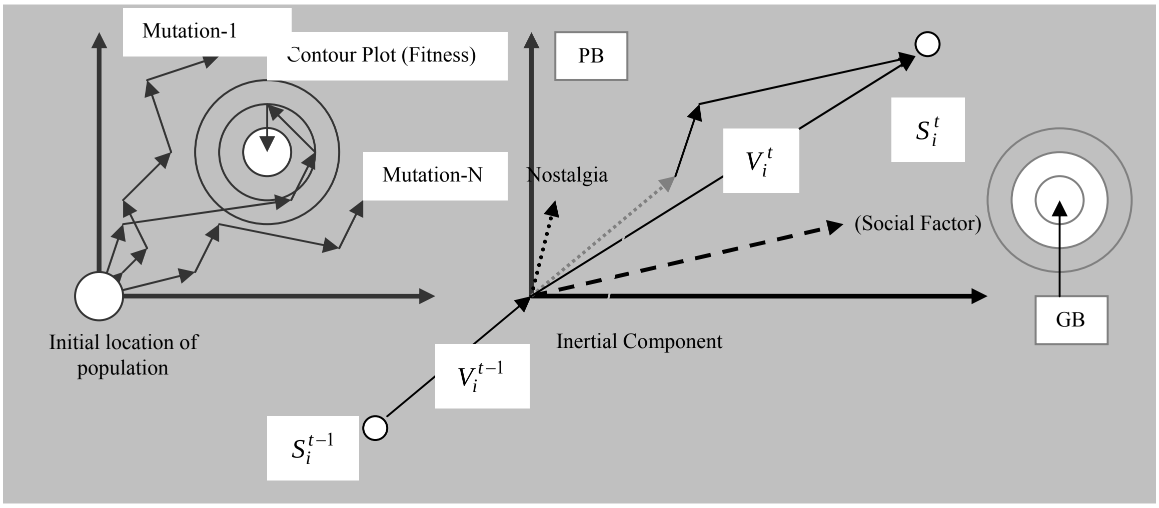


Figure 3.38: Dynamics of immortal particles in both ES and PSO.

The time lapsing trajectories of some of the particles are drawn in Figure 3.38. On the left is the typical particle dynamics scenario inspired by ES, it illustrates the fact that out of three particles only one has ever managed to find the optimal solution, and this has been possible only in their 5th generation (after undergoing extensive transitional phases involving mutation and selection processes). Whereas in the PSO case, the position of particle (after consolidating the effects of the inertial, nostalgic and social components) is still much further away, and is in a transitional state (at position S_i^t from the current global best solution and moving with a net velocity of V_i^t).

The integrated aftermaths of these behaviours of all particles $(P_{i=1:N})$ in the whole population could be very time demanding and deteriorates further with changing global best assignments. What we are trying to emphasize here is to utilise particles in an expeditionary manner, and save the transitional energies by halting the unsuccessful explorations and missions. The dormant particles could be aroused (although rebirth is also a suitable option, as described in the context of Algorithm 3.1) once the potential regions are comprehensively searched.

A virtual particle (VP) using radical search approach does not have to divert attention towards the collision avoidance, and intentionally avoids the formalities of calculating the magnetic attractions and repulsive forces. Therefore due to the very nature of VP, it could be anywhere at any time and has better prospects to detect changes due to its foraging behaviour. One selection of such a direct path is shown in Figure 3.39. The potential region is scrutinized immediately by flying the particles directly (which is not possible in real life environment) into the regeneration areas (around potential global best). In contrast, the computational strategies used in standard swarm based methods are elaborative and complex as shown by the dotted line.

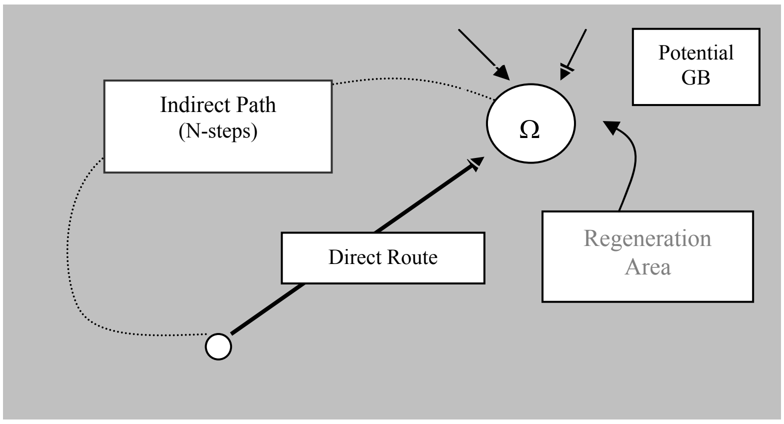


Figure 3.39: Direct path VS indirect trajectories calculated in swarm based methods.

B-Collective Vision of a moving Swarm.

The intrinsic objective of a population of particles is predominantly to take ample measurements in a solution space. The ultimate goal of tens of thousands of particles (when evenly spread in the space of function variables) is to discover data correlations, and if befitting conditions (imposed by the objective function) are detected then those are instantly broadcasted (e.g., GB) using social networking of particles (any keen reader is referred to the Firefly algorithm [187], where a much clearer networking criterion has been developed). Therefore, in contrast to general analytical models (Section 2.5.1, where much rigid control is impelled), this intended spread and collation of measurements captivate the recoverability of the algorithm (especially when a large discrepancy among true and assumed dynamics is present). As discussed in Section 2.6, object tracking in a virtual space does not demand such compacted control and provisions (e.g., imposed through the gain in the standard Kalman filter [28]) as there are no rigorous safety implications as such (e.g., to reduce current transients, copper losses and heat generation).

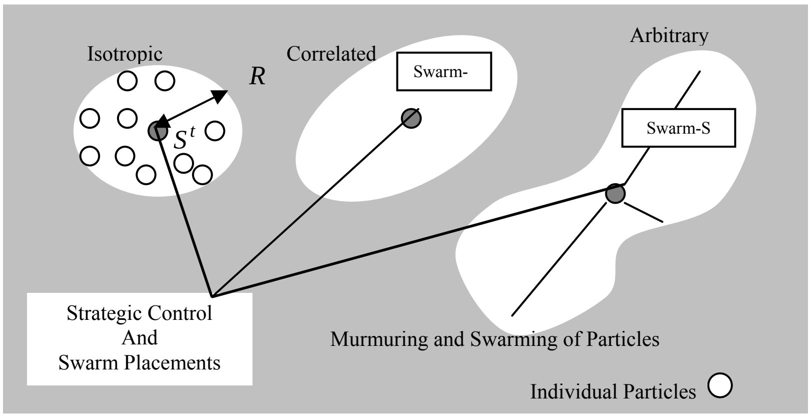


Figure 3.40: A new form of evolution which is based on the revolutionary dynamics.

The virtual swarms are strategically arranged in key areas discerned by the higher-order cognitive processes (e.g., scale space methods and using motion detectors and frame differencing). The mean position of swarm at time instant t is represented by s^t and shape and covariance are controlled by endogenous factors (e.g., the spread $'R'$) as shown in Figure 3.40. To reduce the deployment of a large number of computational agents, we instigate two novel properties in our method i.e. a particle's personal vision (PV) and a group quality imputed by the swarm vision (SV) which is a collective PV characteristic.

In Figure 3.40, instead of deploying evolutionary trends in individual particles, the specific portions of the swarms are displaced using a mean velocity component. Figure 3.40 also reciprocates that the resulting swarm format could acquire any shape and form like murmuring starlings using a strategic control element.

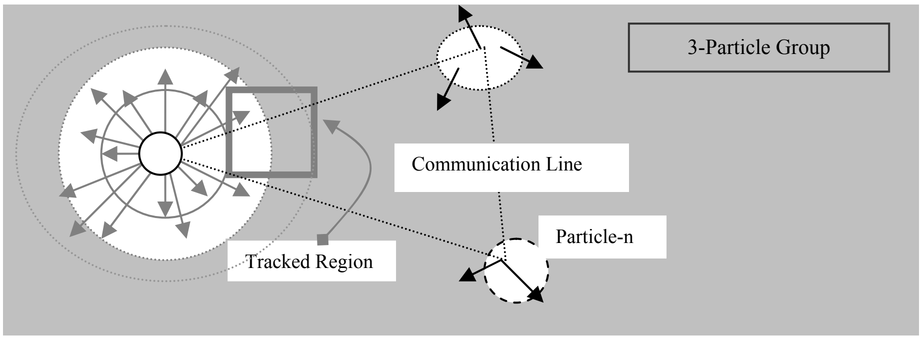


Figure 3.41: The antagonistic processes in controlling the particle visionary radiuses.

This scenario portrayed in Figure 3.41 uses both personal particle vision and also of the surrounding particles forming a network like structure. However, unlike real time foragers they do not have to conduct an expensive journey (in computational terms) to the origin. Generally, raising a flag (could be a binary number) would suffice in such artificial environments. In virtual reality mode (zero transitional energy assumption), we therefore require fewer particles to conduct a wide-ranging analysis of the search space, an idea that fixates on the illusion of swarms of virtual particles rather than pledging for a physical one.

To eliminate/rectify the problems discussed in the previous discussion, in our novel methodology every virtual particle has substantial circle of influence usually orchestrated by the information integration methodologies (as mentioned earlier), this culminates the particles to glance much further into their search space and without explicit data calculations and maintenance. Using the concept of the PV, the tracked region (shown as a rectangular block) is readily recovered (Figure 3.41) despite it was initially beyond the particle's tactile capacity (and normally it is only a pixel wide if standard PSO is used in tracking) but using visionary/olfactory radiuses the problem is rectified without incurring expensive translational costs.

Taking aspirations from the biological life forms, each individual member of a colony (e.g., a forager bee) has its personal visual radius and forecasts its own behaviour by integrating olfactory senses with vibrations or social alerts. In standard PSO, the visions of particles are cramped in the search space (size of a pixel in computer vision) therefore, despite being in close proximity to an optimal solution at times could still remain oblivious of its location.

To remedy this, in our novel methodology every virtual particle has substantial circle of influence usually orchestrated by the information integration methodologies as mentioned earlier, this culminates the particles to glance much further into their search space and without explicit data calculations and maintenance. Using the concept of the PV, the tracked region is readily recovered (Figure 3.38) despite it was initially beyond the particle's visionary/olfactory radius.

The dynamical allocations of the complexity levels and covariance assignments are exerted through an iterative search algorithm (ILS) [188] as shown in Figure 3.41. By establishing the comprehensive swarm visions, the tracking in subsequent frames (like murmuring starlings) demands only a mean velocity control (Figure 3.40). This fractal like movement of swarms is based on a faster revolutionary dynamics (in contrast to the crawling evolutionary changes) and has been found to be much more functional in achieving real time tracking, and further tests would be carried out at later stages of Chapter 3 and in Chapter 6.

The revolutionary dynamics and sweeping movements of swarming structures (in feasibility space) could question the legitimacy of the traditional swarm based models (in the first place, as was discussed critically in reference to Figures 3.38, 3.39, 3.40 and 3.41). Instead of plying with the concentrated converging and dispersion mechanisms, these swarming delusions are virtual manifestations where faster movements are fabricated using coordinated (fixated) particle clouds. There are ample supplements available in the nature inspired literature (e.g., heuristic searches by the Levy walks/flights [189], Brownian motion [190], organised random walks etc.) that could wield such autonomous expeditions (within and beyond the olfactory radiuses).

The virtual murmuring could be simulated by only tens of particles, whereby in PSO we need hundreds and thousands of particles in order to facilitate the same level of search. According to the no free lunch theorem (NFLT) of Wolpert and Macready [191], a particular optimization strategy is likely to be biased in a particular scenario, and when algorithmic instances are averaged out they appear to be almost equivalent. The bias within the PSO and Bat algorithm could also be analysed in the context of the NFLT as described in the following paragraph, and in terms of the elevated data processing requirements.

A particle conducing autonomous scale-free searches must acclimatize with the changing conditions imposed by the objective function. A significant improvement in the speed of convergence could also be achieved by allocating computational agents with simple binary objectives (e.g., by using only relevant bin numbers $b(x_i)$ in Equation 2.13). The evolutionary literature is abundant in such penalising methods and two relevant techniques are guided search (GS) [192] and Tabu search (TS) [193].

C- Locus to the optimal solution and a new reproduction paradigm.

Finally, the most important axiom would be developed in this section which comprises the bulk of the evolutionary literature in mathematics, and in its entirety is dedicated to granting an evolution a ‘sense of direction’. The selection of the evolution path and prescribing the appropriate dosage of the mutation strengths in time are two admissible characteristics in this regard. In general, what we are aiming is to mutate the mutation strength itself in order to acquire the conditions impelled by the objective function.

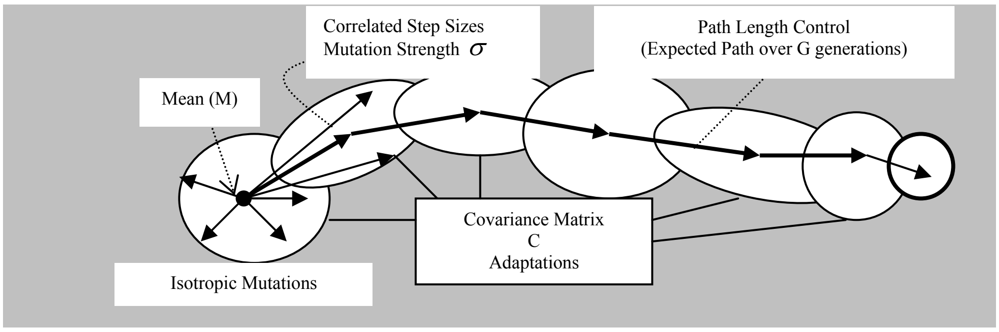


Figure 3.42: The affects of the path length control and correlated mutations over G generations.

The convergence and its antagonist mechanism (dispersions of solutions) are usually implemented in ES by partially inducing the future genetic perturbations in the direction of the historical transitions (spanning over the course of hundreds of generations) $g_{1:n}$. The evolutionary path for seven such generations is shown in Figure 3.42. The core idea of keeping an essential control through computationally cumbersome covariance matrices and adaptations is mainly to define a locus towards a potential optimal solution (Figure 3.42).

In cumulative matrix adaptation evolutionary strategy (CMA-ES [194]), these resolute indulgences (weighted sum) of both anisotropic and isotropic factors facilitate a futuristic but task oriented evolution. The prime aim of integrating the evolutionary paths is to impulse the productivity of any generation and to cut algorithmic complexity. Assigning the mean $m \in R^n$ (after testing the fitness levels of a population) is only the start to embark on this relentless journey, and future mutations are therefore carefully dispersed using the favourite solutions (as shown in Equation 3.8).

However, the aim of the iterative macro-evolutionary stage is to divert the solution towards the optimal regions in the landscape. Sometimes, additional enterprising steps are enumerated by impeding the isotropic quality, and therefore mutation preferences $(\sigma \in R^{+})$ are incorporated using extra measures. Such penalising ellipsoids (mutation directions) are projected by manipulating the $C \in R^{nxn}$ covariance matrices along time as shown in Figure 3.39. This process is mathematically depicted underneath [194].

$$y_i \approx m + \sigma^g N_i(0, C) \tag{3.8}$$

$$z^g = \frac{1}{\mu} \sum_{i=1}^{\mu} w_{i:\lambda}{}^{(g)} \tag{3.9}$$

In Equations (3.8) and (3.9) could be generally interpreted as a mean update procedure (shown in Figure 3.39). Furthermore, in Equation (3.8), i random mutations (y_i) are sampled from a zero mean Gaussian distribution $N(0,C)$ specified by its covariance matrix C (if C=I or identity matrix, then all genetic variations are isotropic Gaussian in nature). During the next phase μ parents are chosen from the λ offspring population (using fitness levels as a selection criterion), and the mean is updated as shown in Equation (3.9).

In more sophisticated evolutionary approaches (e.g., cumulative step-size adaptation evolutionary strategies (CSA-ES]) additional control is emphasized by a finely tuned mechanism which reactively diffuses future mutations σ^{g+1} by taking the historical changes into consideration as well (normally implied by $\sigma^g N_i(0,C)$ distribution during each generational step). Therefore the covariance matrix C in generation g and the updated parental centroid z^g are both taken into considerations to determine the new genetic correlations (Equation 3.10). If v^{g+1} is one such adapted path in current generation, then Beyer [195] imposes evolutionary control as shown in Equation (3.10).

$$v^{g+1} := (1-c)v^g + \sqrt{c(2-c)}\frac{\sqrt{\mu}}{\sigma^g}z^g \tag{3.10}$$

The cumulative control of evolution is therefore subjected by determining the weighted sum of v^g (using cumulative time parameter $(0 \le c \le 1)$) and the newly calculated vector specified by $\frac{\sqrt{\mu}}{\sigma^g}z^g$ (for detailed proof please consult [195]). The exogenous control is therefore introduced by mutating the mutation strength σ^g and inducing the path control using $\sqrt{\mu}$ parental importance towards z^g direction (as shown in Equation (3.10) [195]).

$$\sigma^{g+1} = \sigma^g \exp\left[\frac{\left\|v^{g+1}\right\| - \underset{n}{\bar{X}}}{D\underset{n}{\bar{X}}}\right] \tag{3.11}$$

Finally the mutation strength is updated by comparing the intensity of $\left\|v^{g+1}\right\|$ with the expected path $\underset{n}{\bar{X}}$ as shown in Equation (3.11), where $D \propto \sqrt{N}$ is a damping factor and N is the dimension of the search space.

One reason for this eccentric control (of the shapes of distributions) is to increase the efficiency of the mutations (in relevance to the fitness landscape), and to discourage the proliferation of particles in non optimal regions. In essence, we need to entail viable path control strategy without complicated matrix manipulations and storage requirements (suitable for platforms with limited computational powers). In tracking such correlations (of solutions) could be determined by studying the history of movements, where the mean of the converged particles specifies the positional coordinates of any dynamic object.

The exogenous and endogenous controls are usually directed to entertain the peculiarities introduced by the immortality assumption (of particles) as explained earlier in this chapter. Alternatively, the path control could be nurtured by the strategic deployment of particles with varying olfactory circles as shown in Figures 3.41 and 3.43. Eventually, as mortal entities, there are no strict requisites for documenting the parental explorations (shown on the bottom left of Figure 3.43).

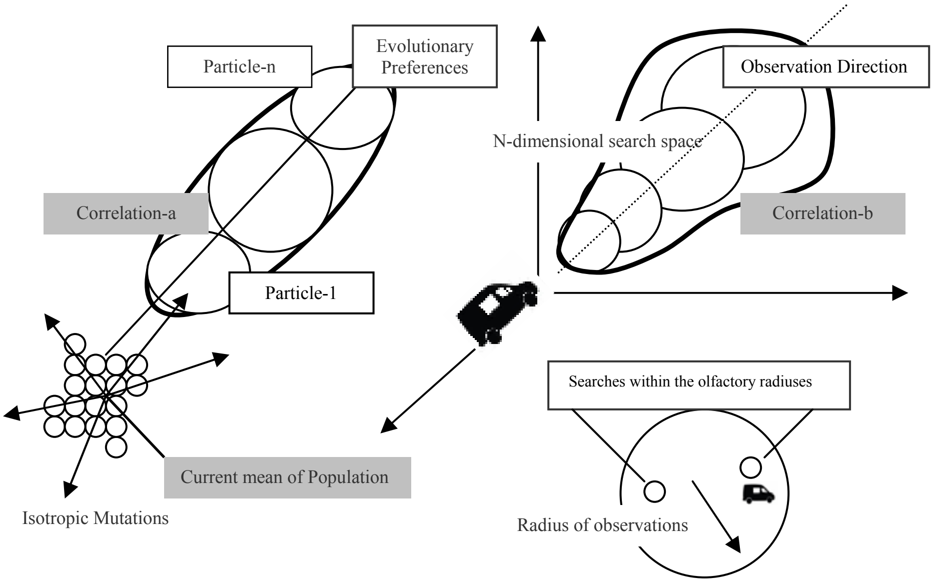


Figure 3.43: The correlation effects could be achieved using strategic placement of particles.

To detect and track a dynamic object (e.g., a moving automobile) we only create observational dispositions by cleverly deploying search particles with diversified scope of measurements. A typical object detection scenario has been presented at the bottom right (Figure 3.40) where an otherwise foist exogenous strategy is exerted simplistically using the observation radiuses $'R'$ and mean position of the particles $\begin{bmatrix} s_y \\ s_x \end{bmatrix}$.

In Figure 3.44, two such searches are explored, on the left the searches appear to be familiar to a Levy distribution, whereby the coordinated movement styles are used on the right to detect an object. The main objective of the next section is to conduct detailed experimental analysis (using a variety of cases) to test the convergence features of RSO. This would enable us to find out that whether radical searches offer any computational advantages over the inherently complex social systems.

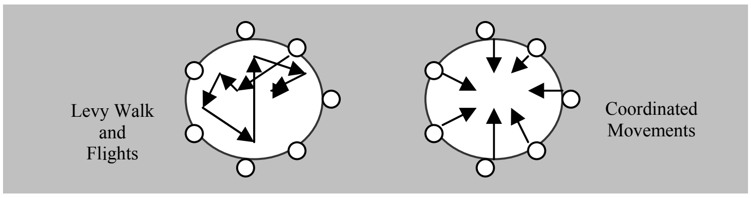


Figure 3.44: Utilizing two established techniques in searching within olfactory radiuses.

In Chapter 6 we will perform further experimentation in order to research into the most practical autonomous search techniques. The virtual measurements taken within olfactory circles does not require endogenous controls as such (e.g., mutation strengths and path length control) but conducive responses of self-centred heuristic searches may suffice (e.g., Levy flights and walks which are still utilized in the 21st century by the African hunter-gatherers in search of bush meat, and the foraging patterns of honey bees during nest site selections.

3.5 Experimental Analysis.

The principal goal of the fact-finding analysis performed earlier in Chapter 3 (and graphically portrayed in Figures 3.5-3.17) was to identify some possible flaws in commonly used gradient based techniques in the context of computer vision applications. From these experiments we reached to the conclusion that the optimization success probabilities are acutely conditioned by the problematic landscapes (Figure 3.12). Therefore in order to attain an optimal solution in space-time dimensions and to increase the success likelihood, we have to commence searches at multiple locations (feasible portions) in the landscape. A variety of experiments were conducted in this regard and some were recollected in Figure 3.15. The novel stochastic perturbations (in the bi-modal test problem) facilitated reductions in the convergence error.

In this second episode of demonstrations, we aim to minimise the test functions described by Equations (3.1) and (3.2), the focal point is to identify several algorithmic peculiarities in regards to the population based formulations. The comparative analysis among three distinct swarm based methodologies and RSO variants (Figure 3.45) would help the readers to explicate the ideas developed in Chapter-3.

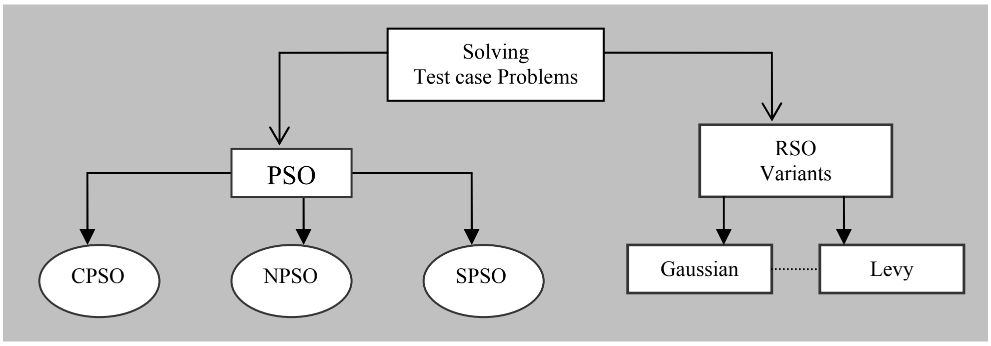


Figure 3.45: The comparative test structure of the forthcoming experiments.

One of the most prestigious particle swarm technique relies heavily on the nostalgic and historical trajectories (stored in memory) in order to select a current global best solution.

Alternatively, it is also possible to assign GB during each generational run by only using the current best solutions. The stochastic particle swarm optimization (SPSO) is a novel algorithm introduced in this context to increase the exploration of agents. Instead of the nostalgic (PB) tendencies, stochastically defined velocities are inaugurated as shown in the mathematical expressions underneath.

$$v_i^t = \Phi_1(GB - S_i^{t-1}) + \Phi_2.(N(\mu,\Sigma) - S_i^{t-1}) \qquad 3.12$$

$$S_i^t = V_i^t + S_i^{t-1} \qquad 3.13$$

In Equation (3.12), Φ_1 and Φ_2 are the tuning parameters which tend to modulate the velocities in accordance to the problem landscapes (so that particle do not jump out of feasible space), $N(\mu,\Sigma)$ are the multivariate Gaussian distributions defined by their means μ and variances Σ, whereas S_i^{t-1} represent the coordinates of any particle i during the time step $t-1$. The positions of all particles in the population are therefore updated as shown in Equation (3.13). During the next stage, the fitness of a particle is tested and both personal best and global best vectors are promptly updated.

The flowdiagram in Figure 3.46 clearly indicates the generic nostalgic phenomenon (for a particular iteration) that takes place in the standard PSO. If the newly anticipated objective function values (at a recent position of a computational agent/particle) are not improved from the historical searches, then the future trajectories (of particles) are calculated through a vector summation which takes into consideration the previously best known position P_b. The nostalgic velocity components are shown in Equation (3.5) using $\Phi_1 r_1[\hat{x}_i(t) - x_i(t)]$ vector, $\Phi_1 r$ is a dampening factor that allocates preferences/weight we allocate to the nostalgic features in order to update particles positions S_i^t (Figure 3.46).

Therefore, NPSO is a form of PSO which considers nostalgic/historical velocities at each and every transitional stage. In contrast, concurrent or CPSO only depends on the current iterations without recording previously best achievements of all particles in the search space.

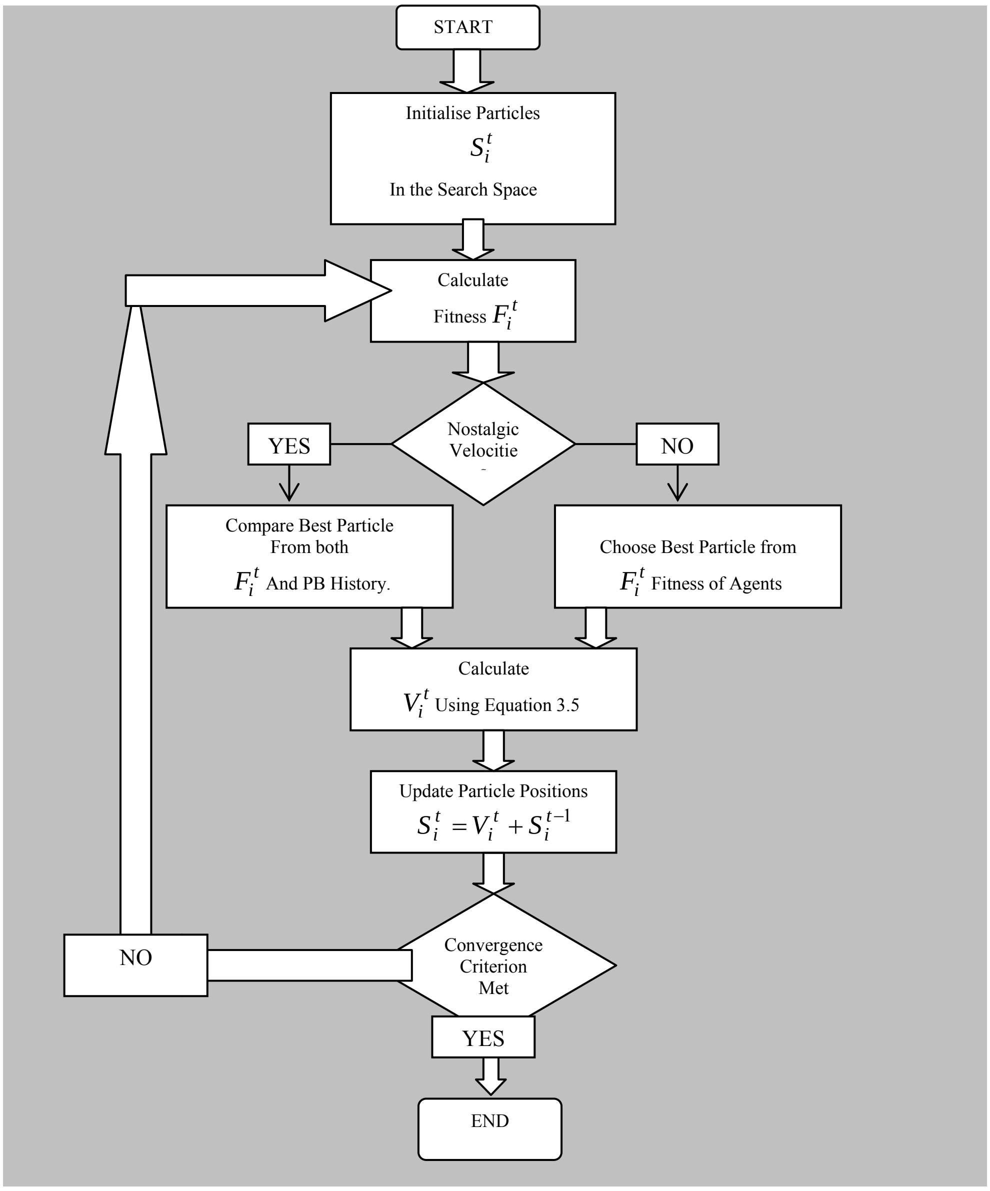


Figure 3.46: A generic PSO flow diagram used to solve test problems in Chapter-3.

3.5.1 Solving uni-modal test case.

All three variants of PSO are applied to solve the convex uni-modal test case introduced in Figure 3.6. The results of this analysis are presented in Figures 3.47-3.49.

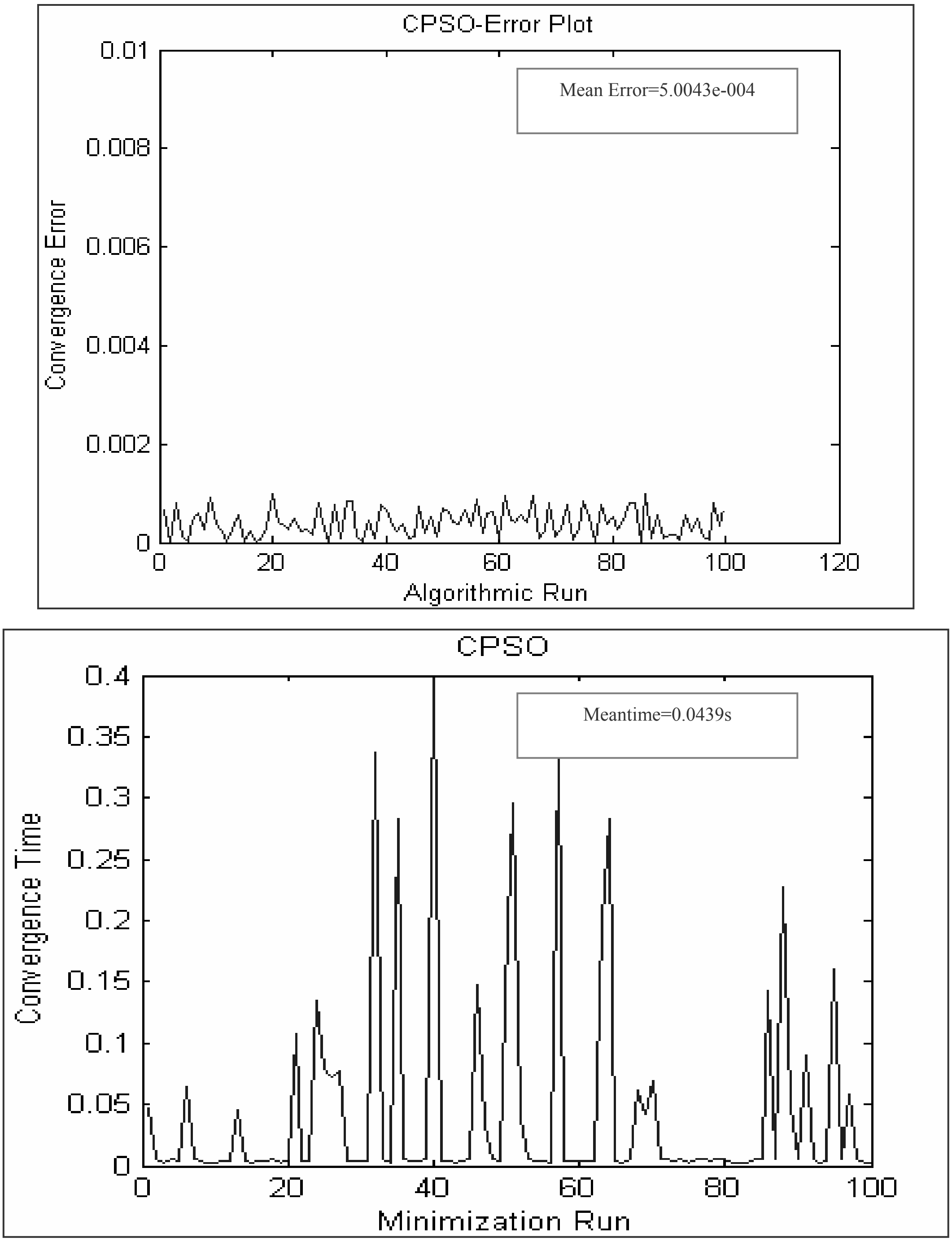
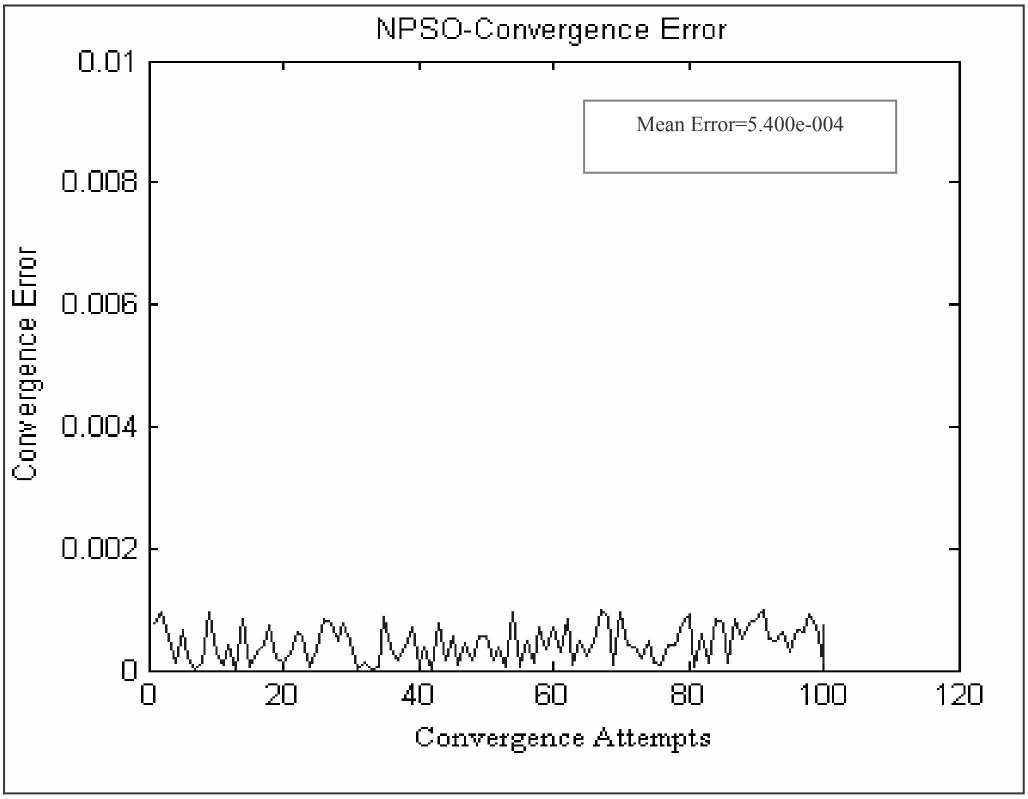


Figure 3.47: Behaviour of CPSO to solve the uni-modal test problem defined in Figure 3.6



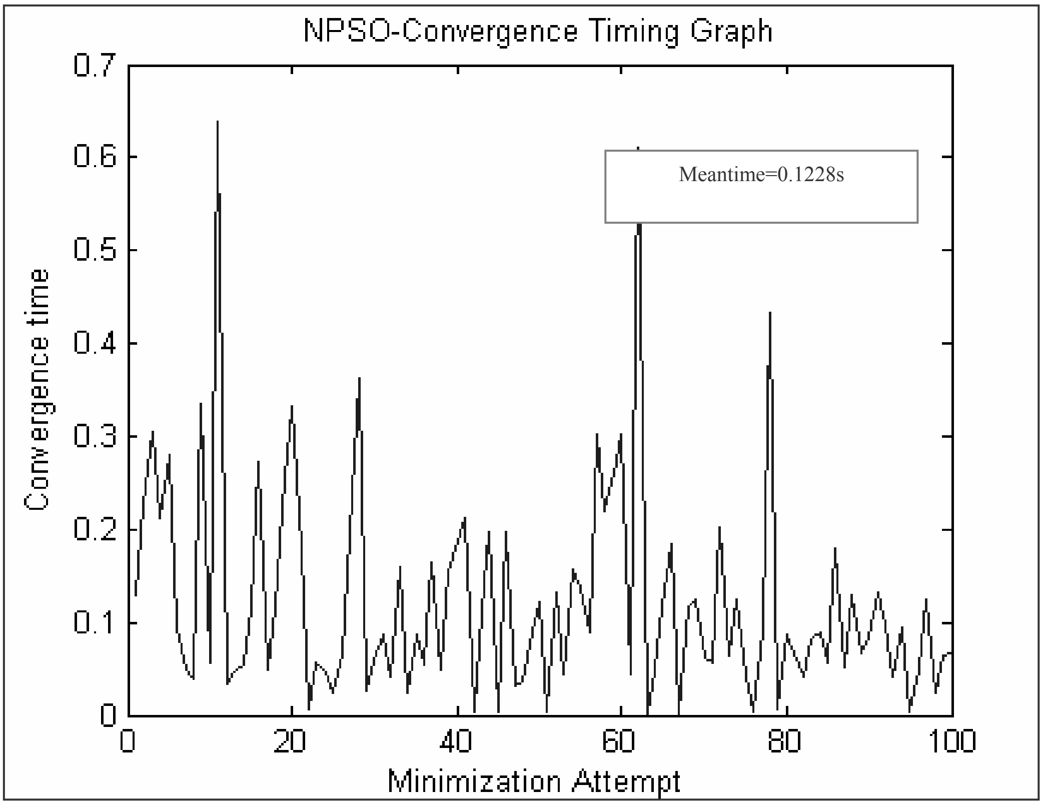
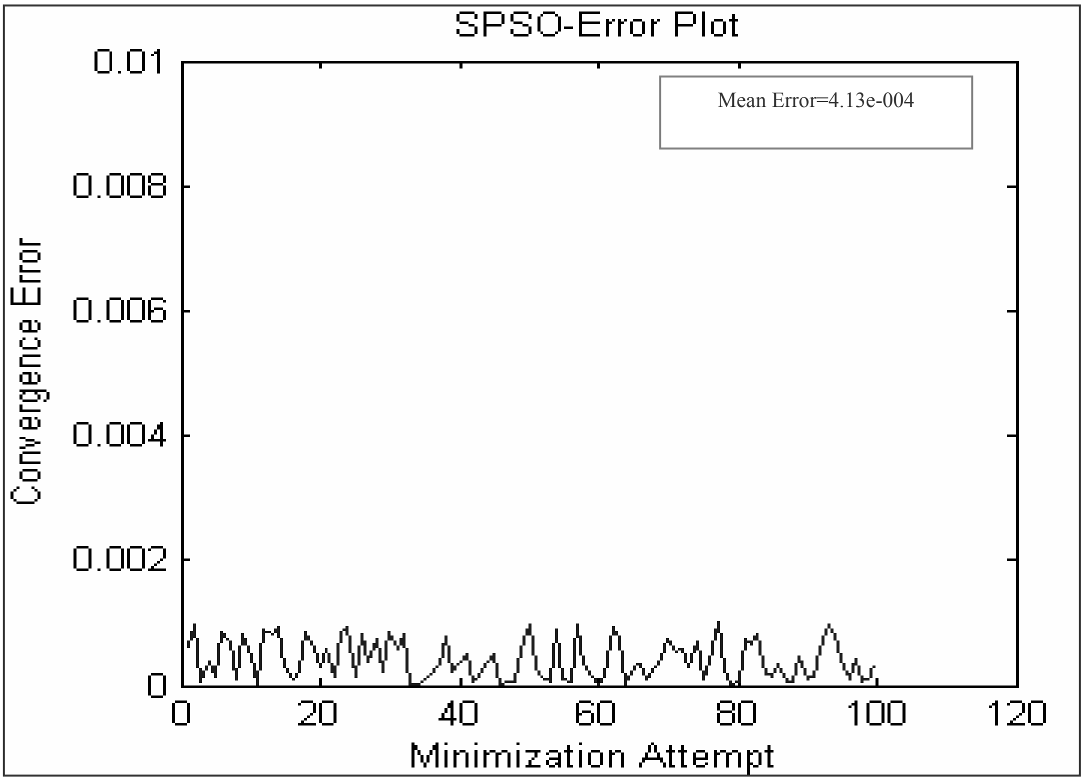


Figure 3.48: Applying NPSO to solve the uni-modal test problem.



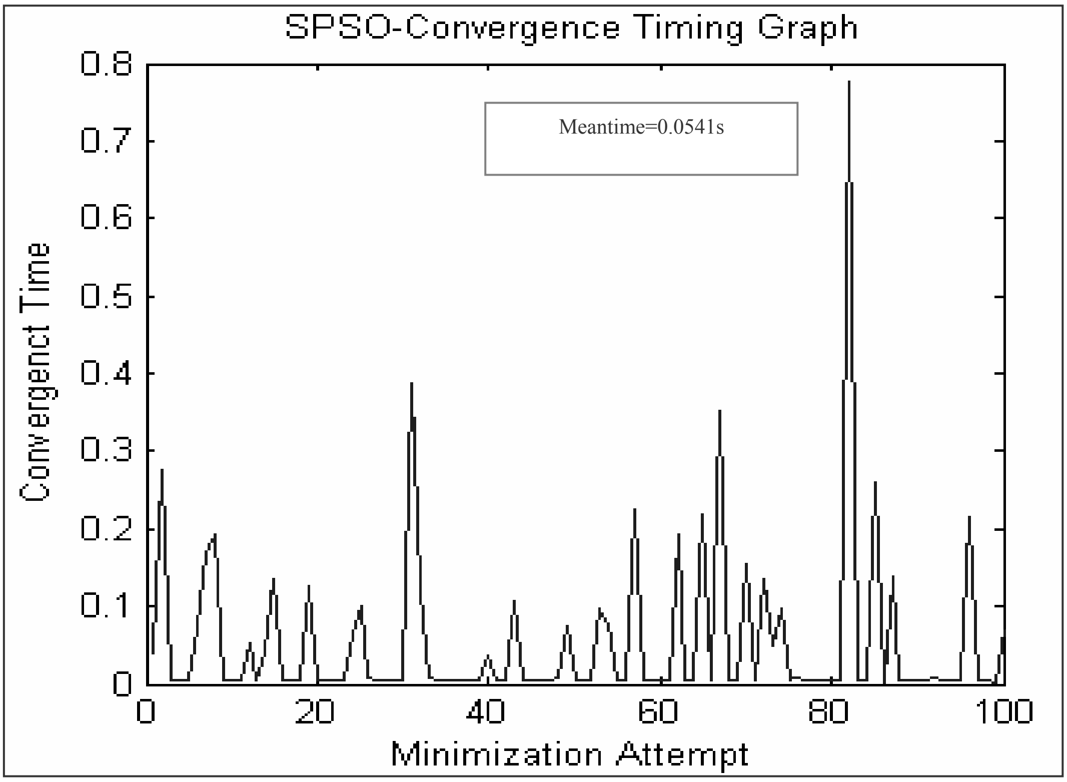


Figure 3.49: Applying SPSO in minimization case defined in Equation (3.1).

It is evident from Figures 3.47-3.49 that in terms of convergence timings, NPSO (nostalgic variant of PSO) behaved much worse than the CPSO (concurrent-PSO) and SPSO (stochastic-PSO) implementations of the algorithm. The mean convergence time for 100 serial runs of this minimization algorithm has been recorded to be around 0.1228 seconds, which is considerably higher than both CPSO and SPSO. The errors plots also confirmed that all the variants of PSO descended to an acceptable solution in their search space. The algorithmic parameters used in this analysis have been selected to be $n = population\_size = 50 \ \Phi_i = 0.8 \ \Phi_2 = 0.09$.

3.5.2 Optimising bi-modal test problem.

During course of the next experiments, we applied all three PSO variants to solve the bi-modal test problem (as shown in Figure 3.6). Although the search space in both unimodal/bimodal cases are identical, but it appears, that the convergence timings has been adversely affected by the inherent complexity of the landscape (e.g., the convergence time for CPSO have been recorded to be 0.0439 and 1.3831 seconds for uni-modal and bi-modal test cases respectively).

The prime reason for this higher complexity (in view of the author) is mainly due to the changing assignments of the global best (see Figure 3.23), and is in accordance with the text descriptions (Section 3.2.2). In particular, and with references to Figures 3.50-3.53, we have analysed that SPSO (see Equations 3.12 and 3.13) has the lowest recorded mean convergence timing in all 100 algorithmic instances. The relevant error plots confirmed that all PSO variants successfully converged to the global optimum. The algorithmic errors were significantly lower than the single solution based approaches (Figures 3.16, 3.17).

Further investigations would be carried out (in the next series of experiments) in order to study the effects of the landscapes onto the convergence timings (with a view of real time tracking in mind). As it was proclaimed earlier that the social hierarchy created by the swarming particles is mainly a simulative gesture, and scale free search experiences (inline with the David Kolb learning experiences) are relevantly better strategies, to confirm this, we will also implement and administer the three fundamental characteristic/properties developed (in contrast to the immortality assumption in a general PSO algorithm) earlier in Section 3.3 to these test cases.

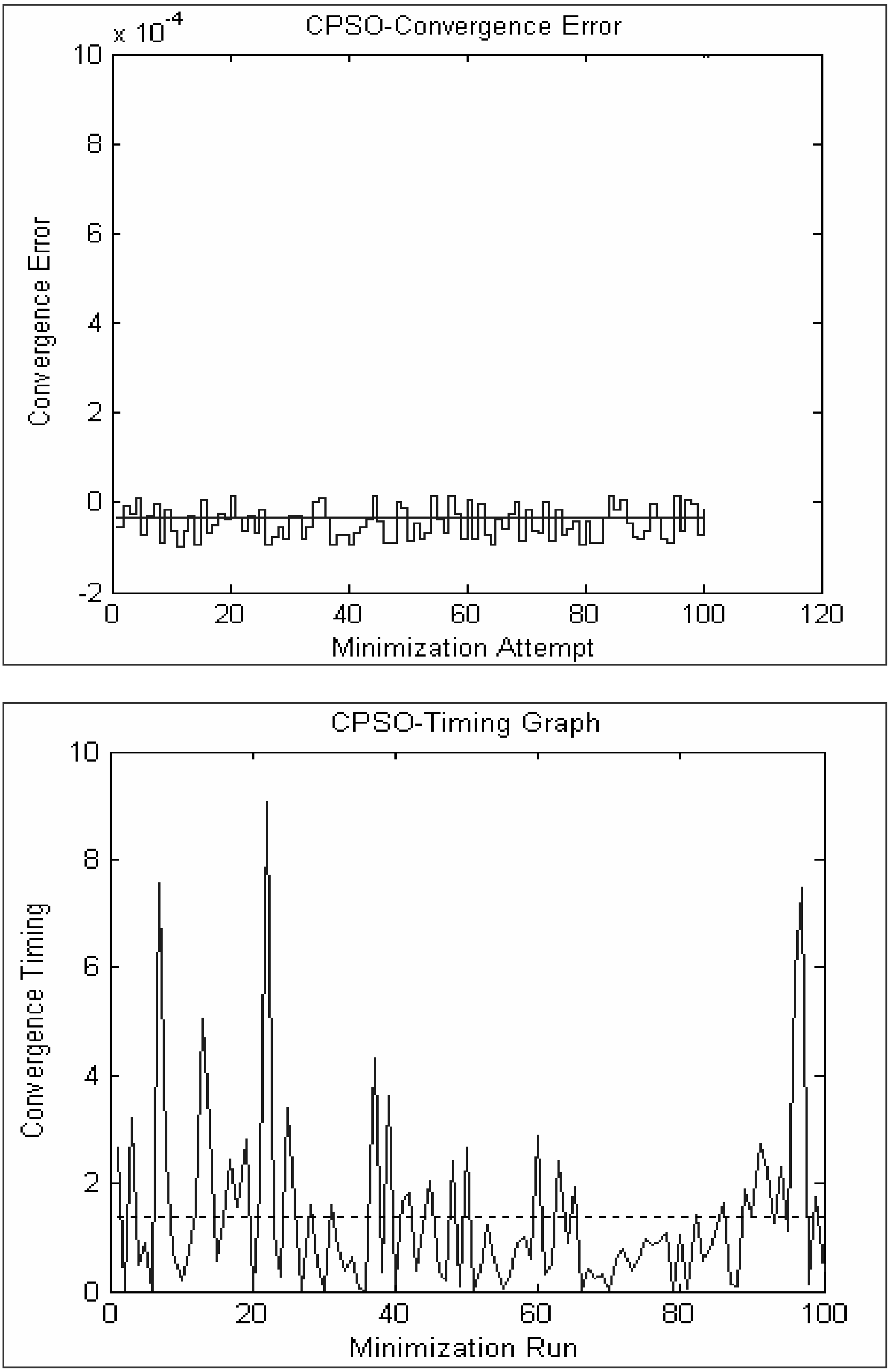
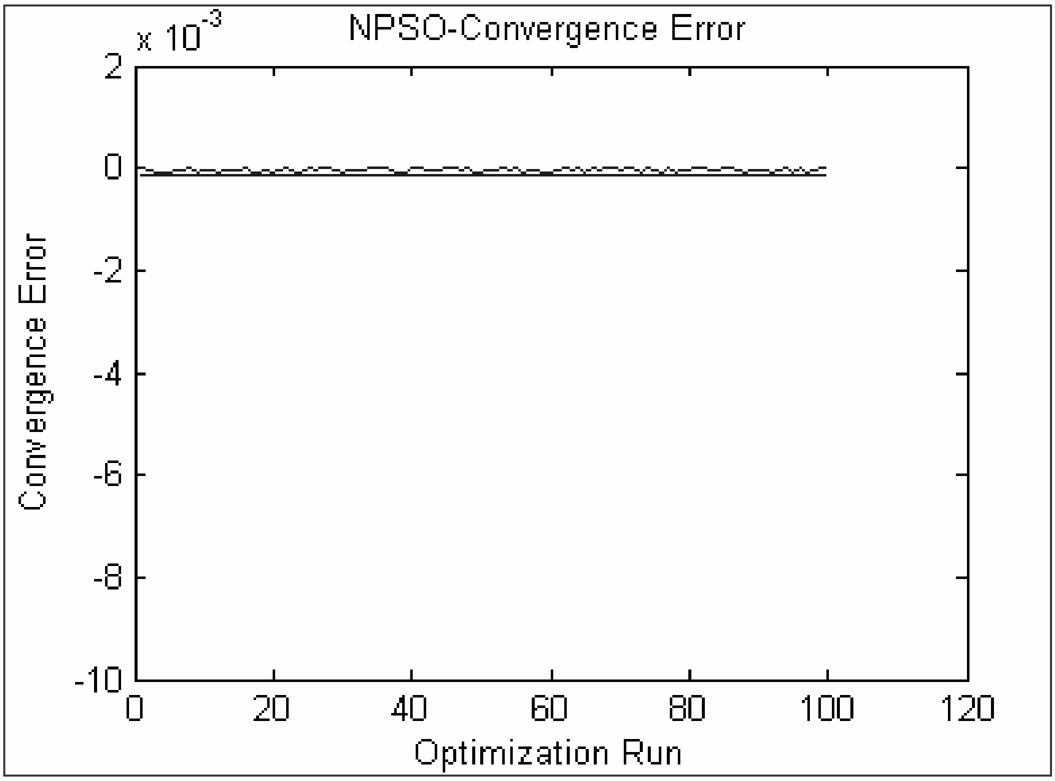


Figure 3.50: Using CPSO to solve a bimodal test case as shown in Equation (3.2).



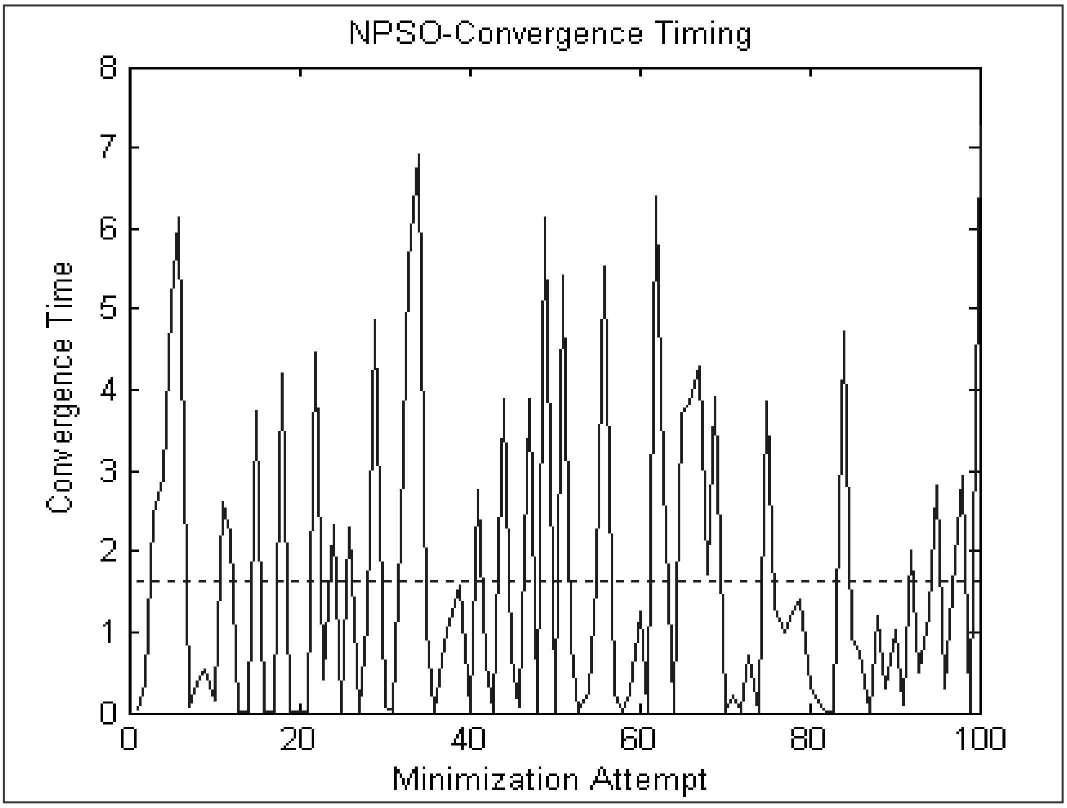
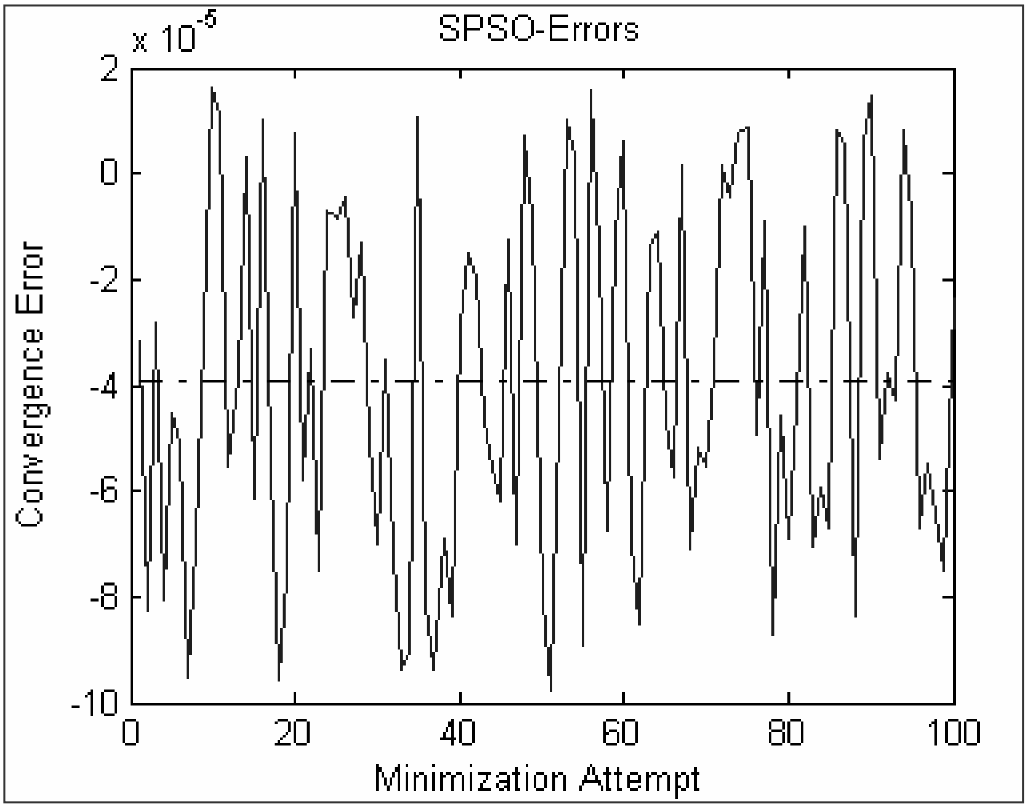


Figure 3.51: Hundred iterations of the NPSO algorithm to solve the bimodal test case.



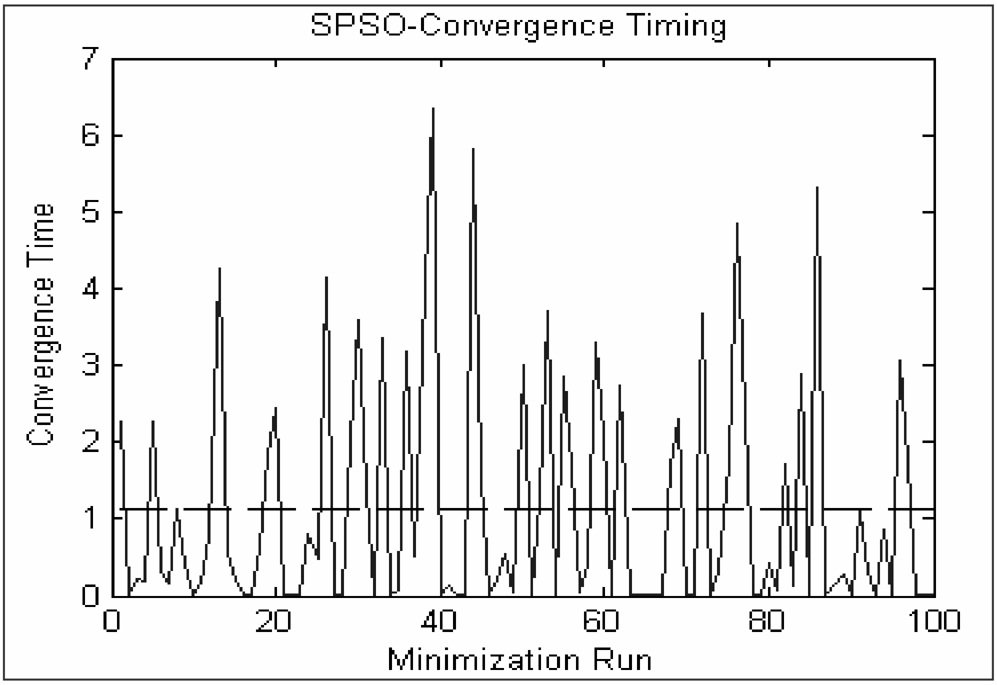


Figure 3.52: The application of SPSO in 100 algorithmic runs to solve bimodal test case.

3.5.3 Optimising the Egg-Crate Function.

In the next experiment PSO variants are applied to minimize the Egg-Crate function as presented earlier in Figures 3.22 and 3.33. The results are presented in this section.

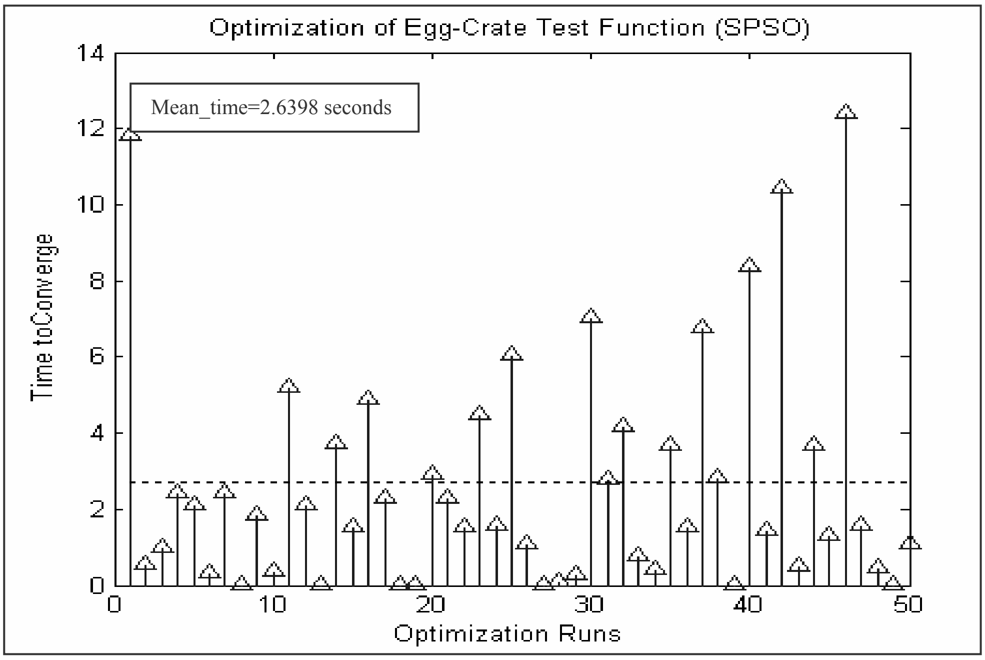


Figure 3.53: Solving the Egg-Crate minimization problem using SPSO (50 serial runs).

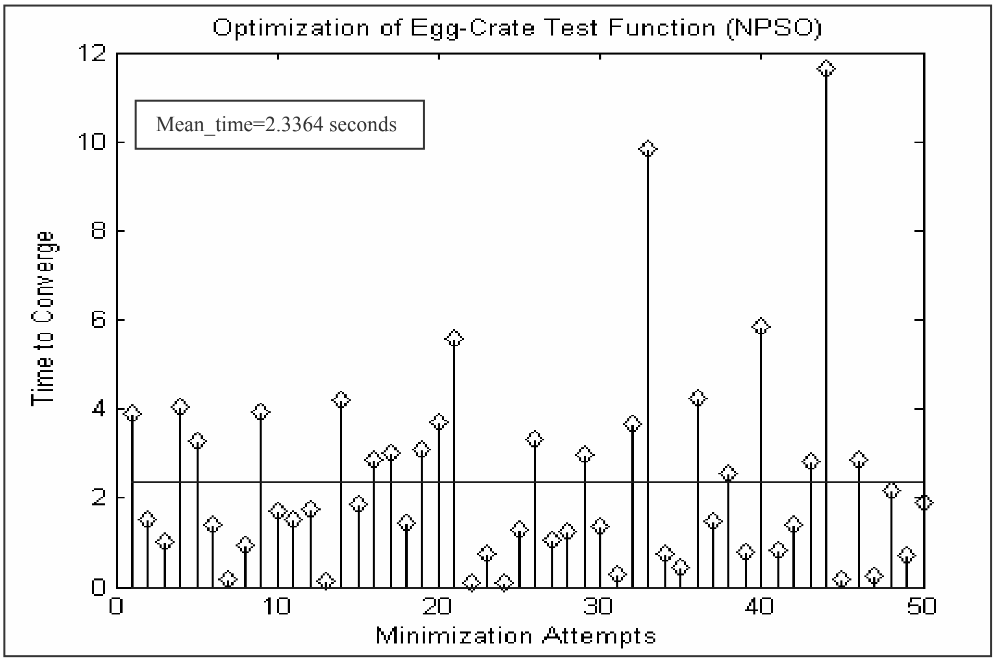


Figure 3.54: Solving the Egg-Crate minimization problem using Nostalgic PSO.

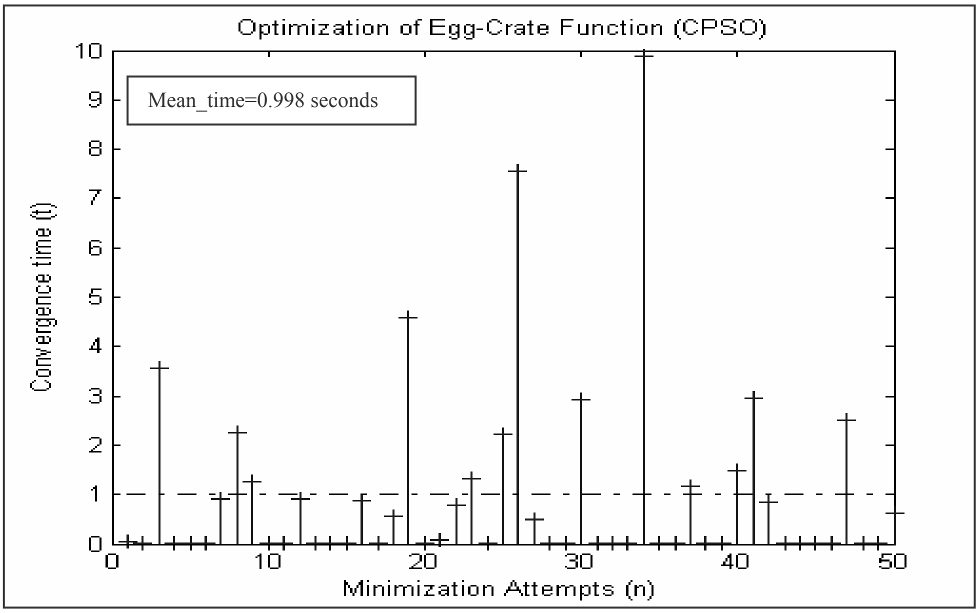


Figure 3.55: Solving the Egg-Crate minimization problem using CPSO.

Despite of a rather difficult landscape, all PSO variants (we implemented using Matlab) successfully converged to the global optimal solution as depicted in the 50 algorithmic runs (Figures 3.53-3.55). Both the Nostalgic and Stochastic variants of PSO were transcended by the CPSO version in this test case, which took only 0.998 sec mean time to converge to the global optimal solution. Although the convergence tests conducted here have shown the dominance of particle based methods, but still there are profound challenges to be addressed in order to develop a faster tracking system.

3.5.4 Solving the highly oscillatory Rastrigin function.

The Rastrigin function (RF) creates another challenging landscape and is commonly used to analyse the convergence characteristics of the nature inspired algorithms [210]. The behaviour of RF is highly rippled and induces an inordinately multimodal scenario with a very narrow locus to the optimal solution (refer to Section 3.2 and Figure 3.3). Therefore the mode seeking algorithms (Section 2.5.4) would not be successfully if are applied in such highly oscillatory problems (as explained earlier in the context of Figures 2.19 and 2.25). The subtle landscape of RF has been plotted in Figure 3.56.

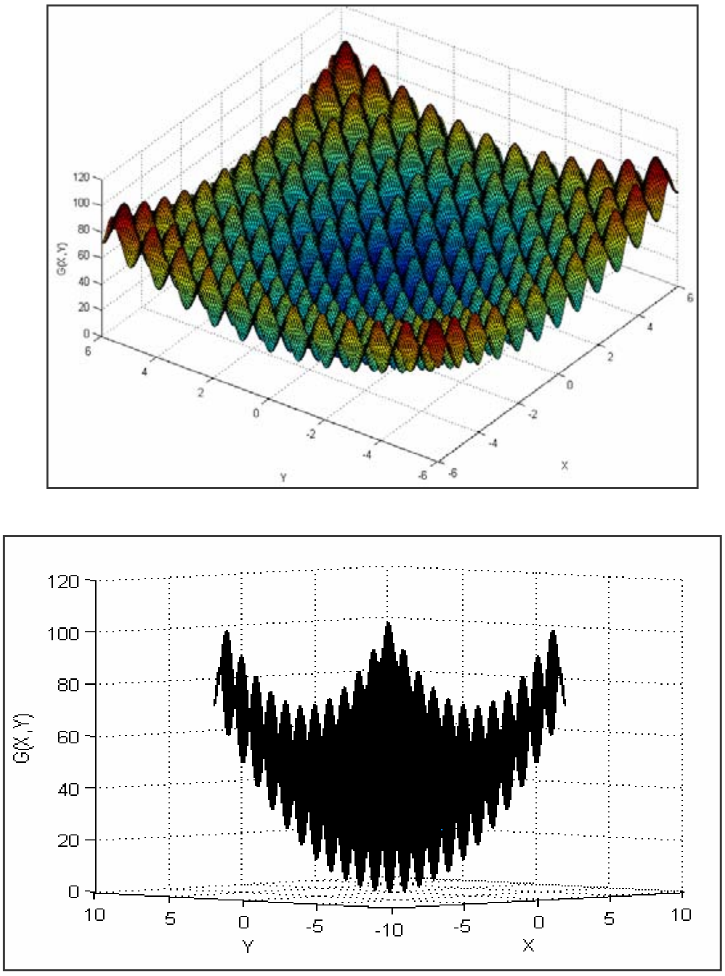


Figure 3:56: The landscape of the Rastrigin Function (RF).

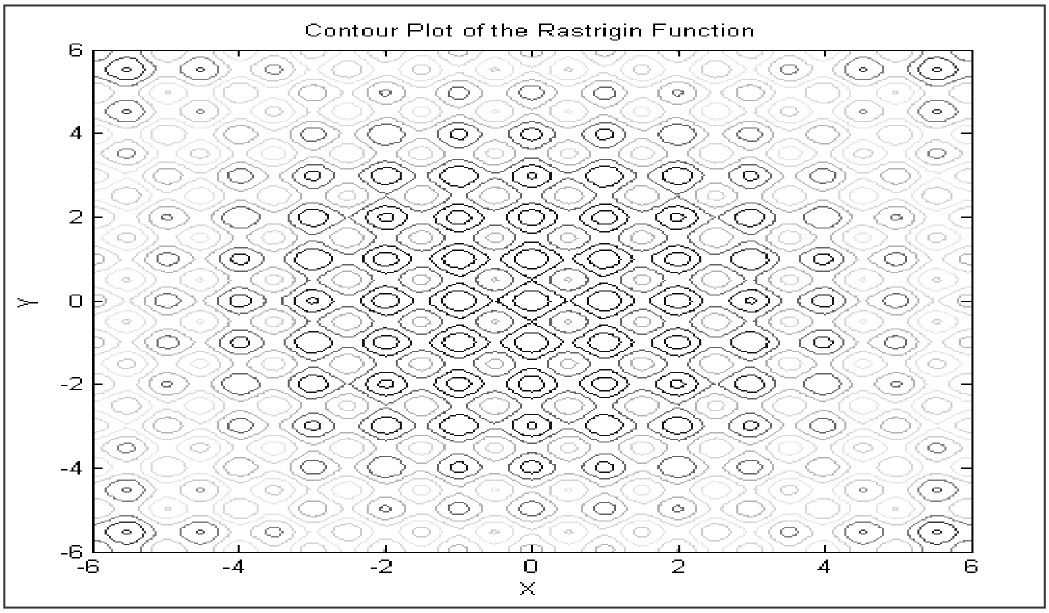


Figure 3:57: The contour plot of the RF.

The contour plot of the Rastrigin functionality is drawn in Figure 3.57; it shows the locus of abundant local distracters along with only a single global solution. The RF is implemented in Matlab here, and the parameters (shown in Equation 3.14) are selected to be $(A = 10, d = 2, x = data\_po\text{int})$. The initial aim is to optimise the RF using CPSO and SPSO alone by completely avoiding the nostalgic criterion (which has been deemed ineffective and an overhead in the previous analysis) in the standard PSO. It is also a prerequisite to define the tuning parameter for both these PSO methodologies (as discussed earlier) in order for the subsequent convergences to be meaningful due to the highly rippled landscape (and also within allowable time and computational iterations).

$$G(x, y) = A.d + \sum_{i=1}^{d}[x_i^2 - A\cos(2\pi x_i)] \qquad 3.14$$

The optimization of RF was initiated by choosing a highly populated solution, and subsequently, we experimented with the tuning parameters $\Phi_1$and Φ_2(in Equation 3.5) in order to understand the convergence phenomenon in this specific problem. Figures 3.58 and 3.59 demonstrate the significance of choosing appropriate tuning parameters in particle swarm optimization. According to the results displayed in the graphical format underneath, the most consistent and error free solutions were observed in the ranges $0.35 < \Phi_1 < 0.63$ and $0.01 \le \Phi_2 < 0.06$, the peaks in the graphs exhibit that beyond these specified regions, the intensity of errors grows resulting in the failure of the minimization objective (the location of GB in RF is $G_{(x,y)} = 0$ at $X = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$).

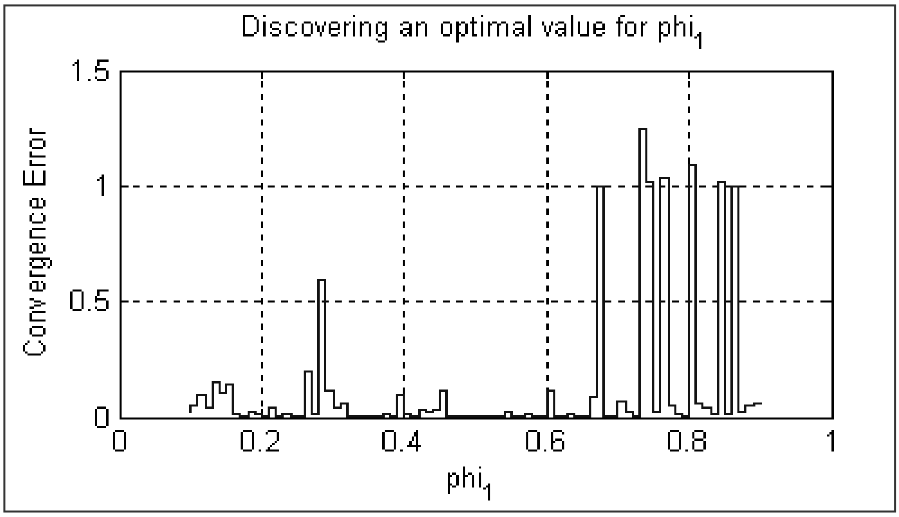


Figure 3:58: A number of experiments were conducted (91) to determine the best Φ_1 value.

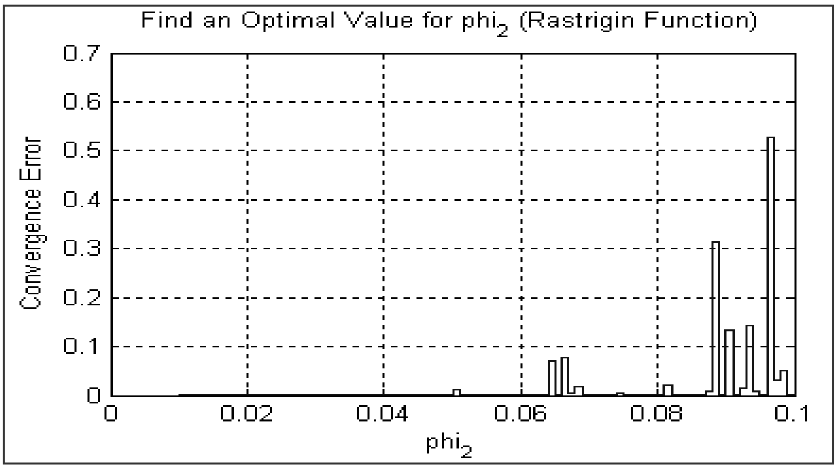


Figure 3:59: A wide variety of tests used to choose the most suitable parametric value for Φ_2

The graph in Figure 3.60 shows that the best convergence time is observed when $0.35 \le \Phi_1 \le 0.45$. Hence, a suitable choice (after studying the responses in both Figures

3.58 and 3.60) could obviously be $\Phi_1 = 0.40$. Whereas, by analysing Figures 3.59 and 361, we reached to a conclusion that $\Phi_2 = 0.028$ would also be a wiser selection for reducing both execution time and convergence errors in the minimization of the RF.

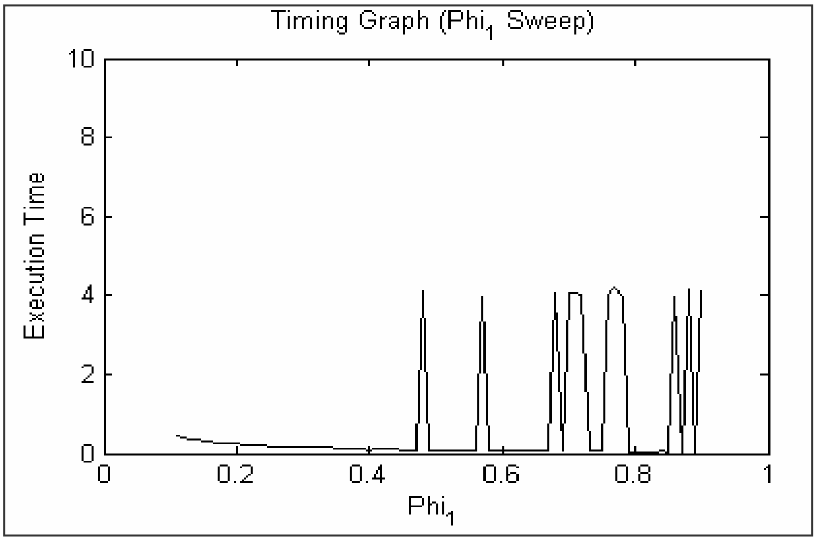


Figure 3:60: The convergence parameter Φ_1 and the timing graph in the CPSO method.

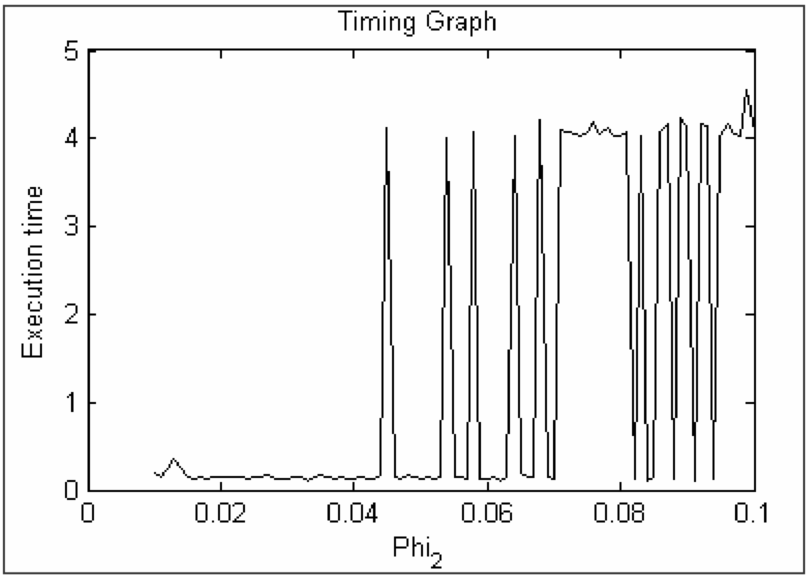


Figure 3:61: The effects of the convergence parameter Φ_2 onto the algorithmic timing.

As explained earlier in this section, the appropriate choice of population strength (n) is another key criterion to be met in the tracking systems. The three dimensional landscape of the RF is (as a matter of fact) remarkably identical to the video camera frames in the object tracking scenarios. The translational movement of an object under observation (similar to $\begin{bmatrix} x \\ y \end{bmatrix}$ independent parameters of selection in RF) takes place in a two dimensional grid as well, where the successful determination of the locus (to the minimum in vision tasks) depends on the size of a region. As the feasible space in RF is narrow, therefore if RSO successfully detects the global optimal solution, it certainly would prove useful to solve vision problems.

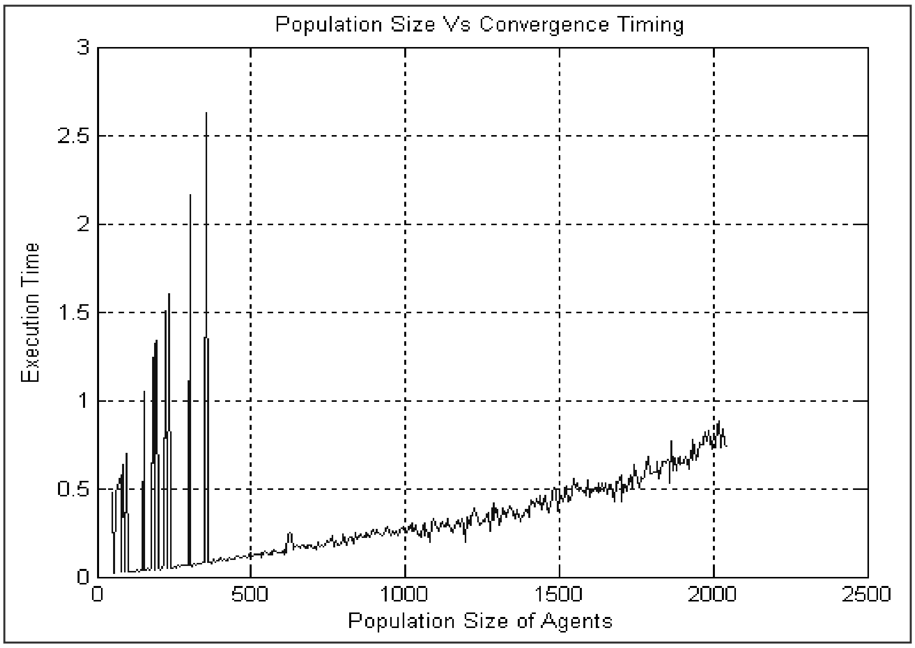


Figure 3:62: The population sweep is employed to analyse the effects on execution timing.

In Figure 3.62, the minimization of the Rastrigin problem was initiated with a population of n=50. During next iterations, the population was increased by 5 and particles were repositioned in the solution space to conduct searches. It is evident from the timing graph (Figure 3.62), that the solution failed to converge initially (where the peaks represent that the maximum permissible time has lapsed). Once a suitable population size (n=350) was reached, the minimization gained momentum, and further increments have resulted in higher execution times (without any optimization benefits) reaching to 0.9 second at N=2050.

Having considered the idealistic tuning parameters $(n = 350, \Phi_1 = 0.42, \Phi_2 = 0.028)$, we executed the minimization algorithm, and the error trajectories for both CPSO and SPSO are displayed in Figure 3.63. For the CPSO, the mean convergence time (Figure 3.63a) was recorded to be 0.2911 second, whereby in SPSO, the mean convergence time (Figure 3.63b) was 0.2500, which is slightly better than CPSO. Also the trajectory/slope of the errors in SPSO is steeper and aggressive compared to that of CPSO. Hence, we can safely conclude that both SPSO and CPSO performed competitively in solving the RF, which is an example of a highly oscillatory and multimodal test function.

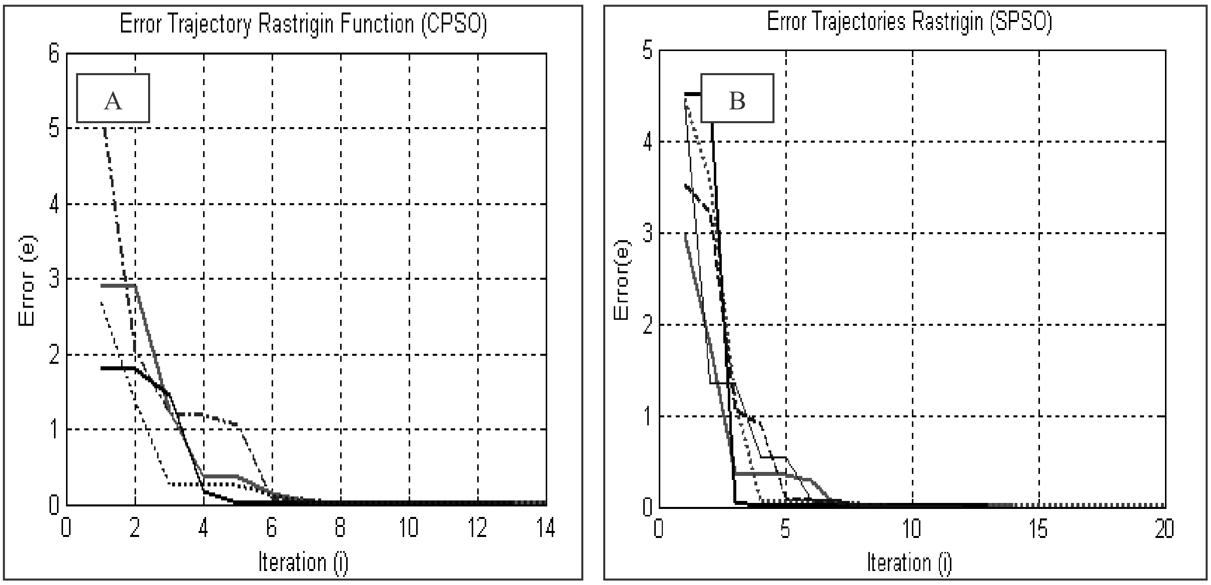


Figure 3:63: The CPSO algorithm generally converged in 6th but SPSO in the 5th iterations.

3.5.5 Solving complex mathematical test problems using radical searches.

We have reached to a pinnacle phase in Chapter-3, where with the aid of the experimentations we would analyse the efficacy of the radical/virtual particles in a variety of test problems as implemented earlier. The Rastrigin function (as shown in Figure 3.56) is a highly multimodal three dimensional test function, where the search space (in our demonstrations) spans between the regions $-6 \le (x, y) \le 6$, this problem is also reckoned to be befitting test bench for the tracking applications due to its narrow feasible space, and hence illustrates a classic detection scenario. The location of the optimal solution has been shown in the search space, the presence of nearby competing peaks pose a huge optimization risk, as a slight drift in the function variables could translate the solution in its entirety to the local optimal areas (Figure 3.64). The RSO (see Algorithm 3.1) is applied to solve the Rastrigin problem, and the results are displayed in Figures 3.65 and 3.66.

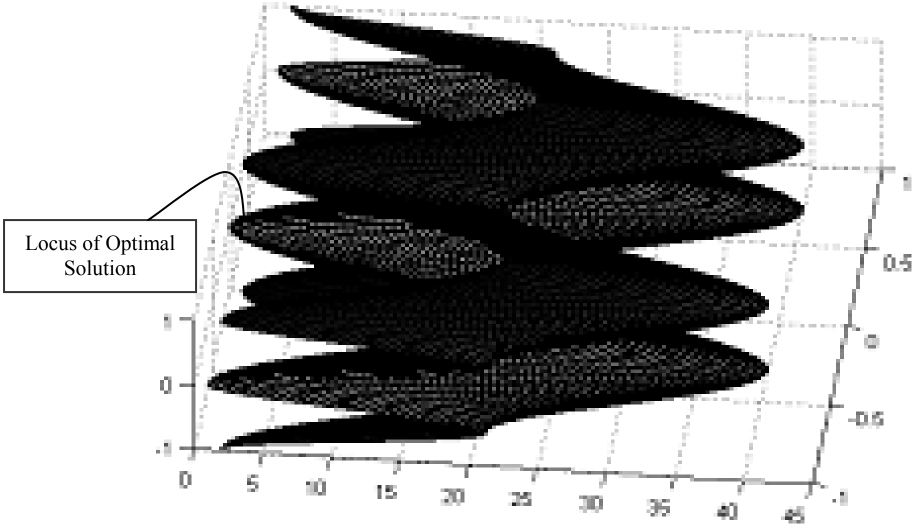


Figure 3:64: A rotated zoomed in mesh plot of the RF.

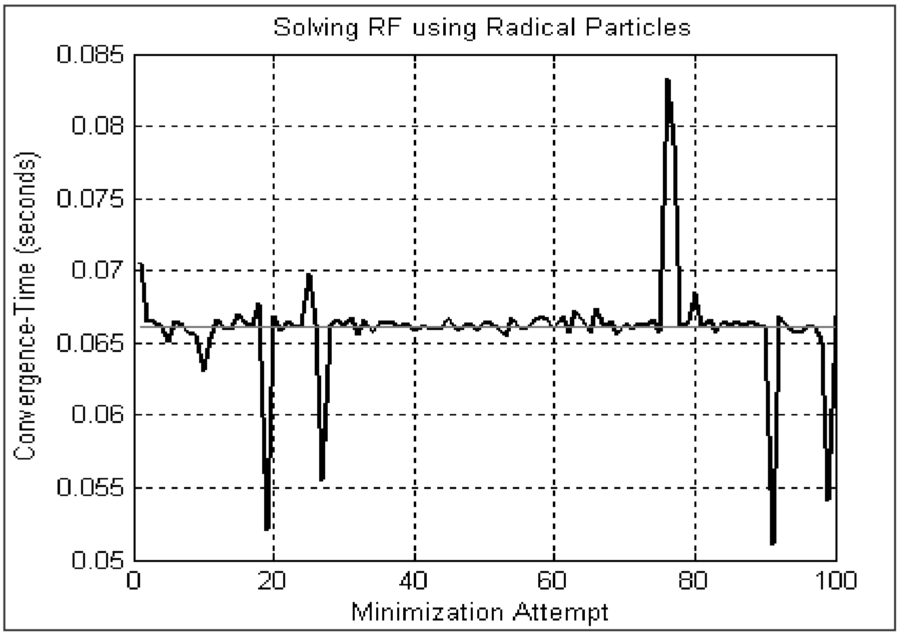


Figure 3:65: Solving the RF using RSO and the relevant convergence timing graph.

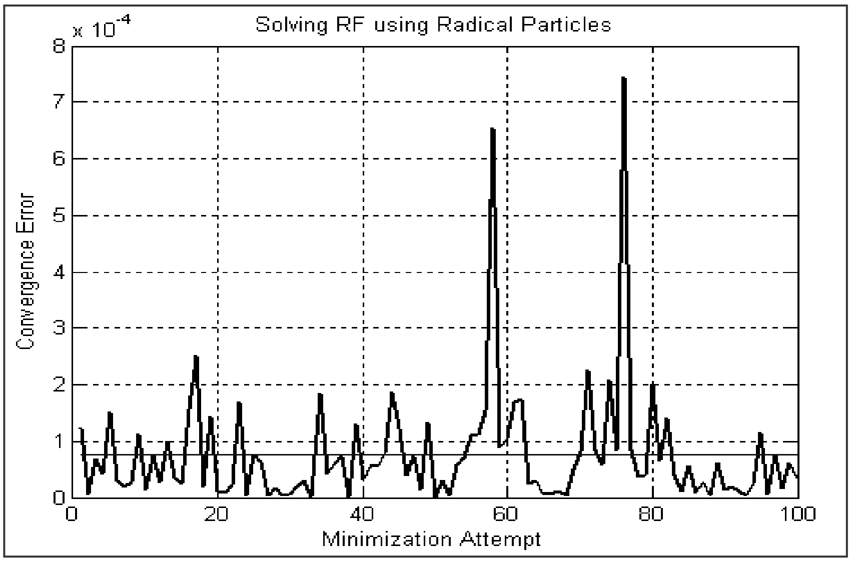
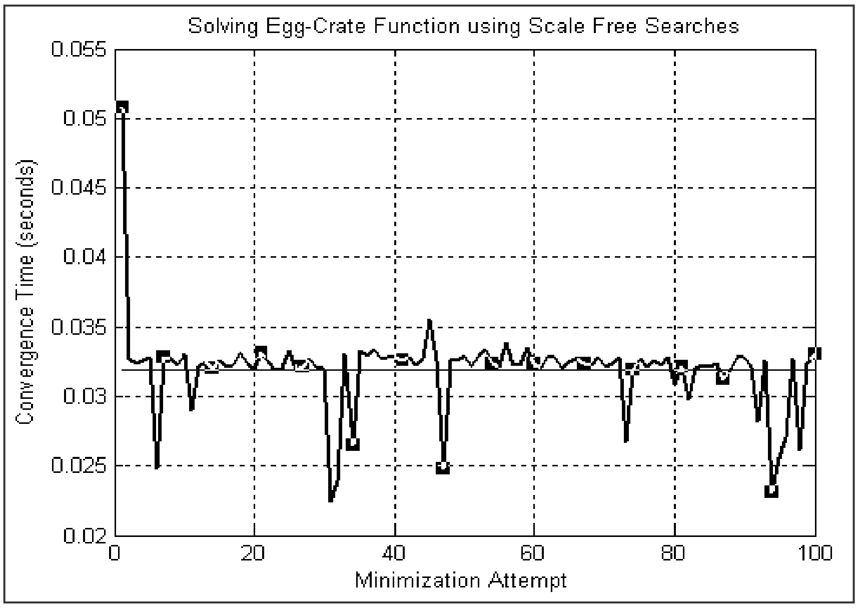


Figure 3:66: The error graph showing 100 optimisation runs in order to solve the RF with RSO.

The mean convergence time for 100 optimization runs (Figure 3.67) was recorded to be 0.0661seconds which is significantly lower than both the CPSO and SPSO algorithms (which were 0.2911 and 0.2500 respectively in the Figures 3.58 and 3.59). The convergence errors (mean=7.65e-005) in Figure 3.66 are also very competitive and significantly lower than the ones observed using the PSO methods. We also applied the RSO to the Egg-Crate function and the results are presented in the Figure 3.67.



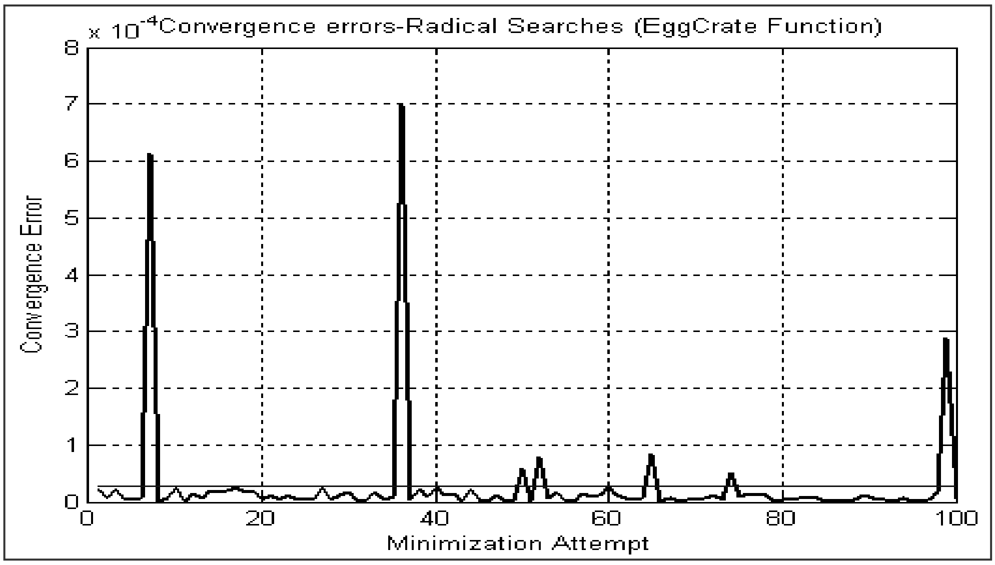


Figure 3:67: The timing and the error graphs for minimizing the Egg-crate function using RSO.

The mean convergence time for 100 optimization runs was recorded to be 0.0319 seconds, whereby CPSO took almost 2.31 seconds (Figure 3.49), and when SPSO was applied (Figure 3.51), a mean convergence time of 0.99 seconds was recorded, and the convergence errors are also notably lower than the threshold set in our experiments. The results of implying the radical searches to solve the bimodal and unimodal cases are shown in Figure 3.68.

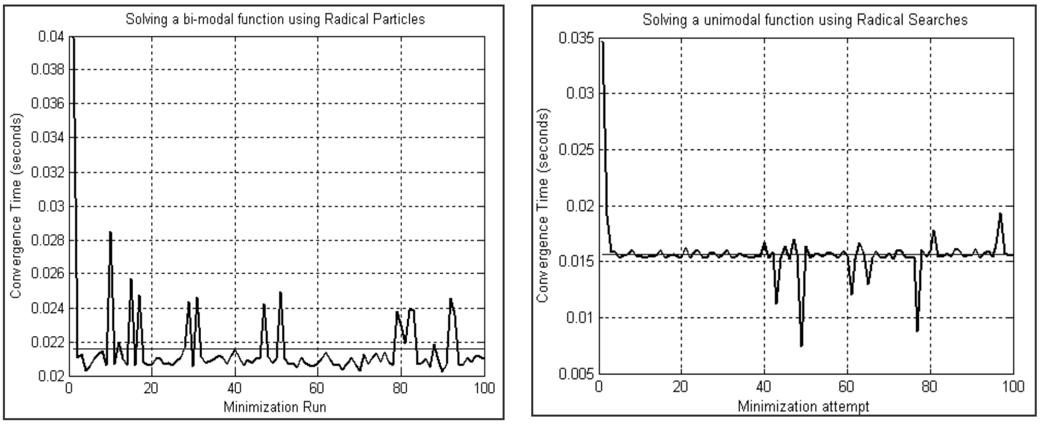


Figure 3:68: Applying RSO to bimodal and unimodal functions

The time consumed by 100 optimisation runs to solve the bimodal test problem using RSO was 63 and 51 times faster than CPSO and SPSO respectively (as were analysed in Figures 3.46-3.48). By comparing the graphs in Figures 3.43-3.45, we reached to a similar conclusion that the unimodal convergence timings using RSO are 2.8, 3.5 times faster than CPSO and

SPSO respectively.

3.6 Conclusions.

Chapter 3 started with a discussion which emphasized on the importance of the experiential learning (David Kolb's model). The wider search experiences have also been witnessed in many natural and biological life forms, the findings of the Starflag project also confirmed that the marvellous flight displays of starlings are due to their scale free correlations. The optimisation problems encountered in the gradient based methods were discussed, and it was analysed through experiments, that in the absence of reliable gradient vector field the minimization processes usually converges to the non-optimal solutions. A major prospect to resolve convergence issues is to initialise the solutions at multiple points in the search space (Figure 3.15). However, significantly better results were achieved using the population based methods (e.g., CPSO and SPSO).

In view of the author, there are a number of design flaws and misconceptions in multiple agent based systems, and readers were introduced to a novel RSO method which is based on three key characteristics and deemed suitable in artificial environment (Section 3.3). Later on in Section 3.4, we demonstrated using evolutionary test cases that scale free search experiences could indeed outperform social swarming methods by huge margins. We also highlighted a tuning free optimisation paradigm that could be used as a meta-heuristics over traditional swarms to improve convergence errors. No evidences were found in our tests that the social calling and nostalgic memory helps convergence. Instead in the view of the experiments social elements are computational overheads.

Chapter 4

Tracking in the Context of Dynamic Optimization

Many real world problems are dynamic in nature, and the convergence at any moment in time could not assert that the same solution would be applicable during the future instances. The goal of any static optimization problem (analysed in Chapter 3) is to find a value for $x^* \in S \in \Re^n$ (x^* is the optimal solution from a feasible set S in the n-dimensional Euclidean space $\Re^n$) in order to maximize (or minimize) the objective function i.e. $f(x^*) \geq f(x)$ (or $f(x^*) \leq f(x)$). The detection of the object of interest in a distinct video frame is a static optimization problem, and once the solution has converged to the optimal, we would generally require some mechanism to address the dynamicity introduced in the next frame. In brief, the convergence was a desired property from static optimization point of view but could become problematic with a dynamical perspective in mind [196].

Therefore to address a dynamic optimisation problem (DOP), some kind of diversity is needed to be artificially introduced in the search space, so that the particles hypotheses do not collapse onto the restricted areas in the search space. The landscape of a visionary optimization problem (a pedestrian tracking problem from a computer vision repository) is shown in Figure 4.1. At each frame, the peak of the density (optimal) drifts randomly, and therefore a particle splitting mechanism (anti convergence order) is required to detect the object like features in the subsequent video frames. In general, computer vision systems often see a deluge of such time varying parameters, this may include changes in the size of the object during incoming frames, the abrupt translational and rotational movements along with unprecedented changes in the object models (in feature space) which could invalidate tracking results.

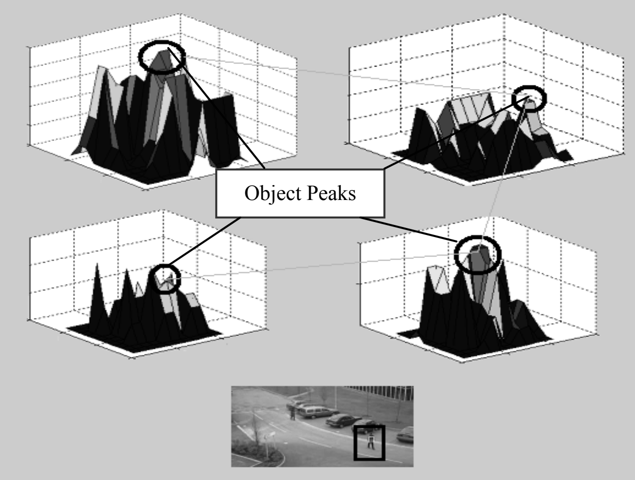


Figure 4.1: The tracking of a pedestrian is a dynamic optimization problem (DOP).

4.1 Diversity indulgence techniques

A range of propositions have been suggested in the evolutionary branch of mathematics to address the catastrophes related to the dynamically changing environments [197] [198]. Some of the most extensively researched themes in this regard involve increasing the diversity of the population by artificial injections [199], and the particles are also restricted from an absolute collapse using anti-convergent measures [200] [201]. Alternatively, it is also possible to exploit an auxiliary agent population (e.g., RSO which serves as an antidote, see Section 3.3.2) to retain a suitable level of diversity, which as a matter of fact, could also decrease the computational complexity for a real time convergence. The scale free experiences (described in Chapter 3) have an inherent feature to address the changing environmental conditions. In multimodal landscapes, another dominant methodology is to watch the competing peaks with random scouts [202], which are arbitrarily solutions introduced to pertain aspirations about the changing objective functions.

$$F = f(x, \phi, t) \qquad 4.1$$

A general DOP problem is usually described mathematically as shown in Equation (4.1) (ϕ is a set of strategy control parameters, which stipulates the spread of the particles in the fitness landscape at any time instant, and x is a solution from a feasible set i.e. $x \in S^n$. A rather simplistic strategic control parameter is the multi-variate measure of the relevant Euclidean distance (from the last known mean of a converged population). If the changes in the strategy parameter are represented by$\Delta\phi$, then at time instant$t+1$, the dynamic optimization problem is expressed as in Equation (4.2), the fusions of the strategy control variables $\phi_t \oplus \Delta\phi$ may also have much broader implications on the future convergence instances of the algorithm [203].

$$f(x,\phi,t+1) = f(x,\phi_t \oplus \Delta\phi,t) \qquad 4.2$$

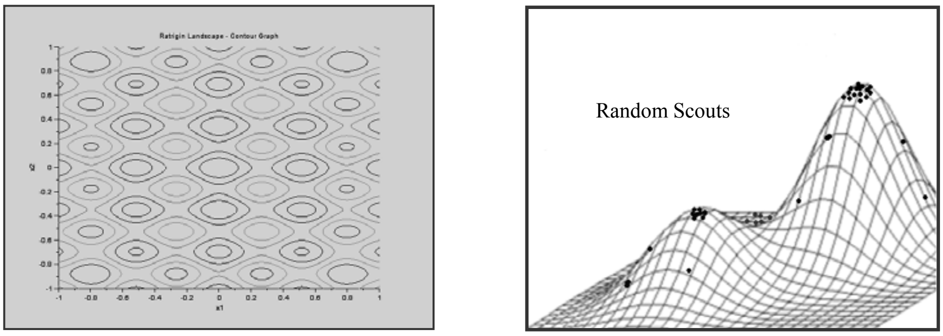


Figure 4.2: Watching multi-modal landscape along time using Random Scout Population

The most fascinating resolution to the changing goal post problem, which is also valid in many tracking scenarios, is applied with the introduction of the scout populations in the search space (with an aim to watch competing regions of interest as mentioned earlier). An optimization scenario is shown in Figure 4.2, where contending peaks are watched by scouts, therefore when the peaks move in time the changes are readily detected and addressed accordingly. The standard iterative optimization methodologies (e.g., the Newton method [204] shown in Equation 4.3) would certainly had failed in such circumstances (especially in the regions of the valleys) where it is extremely difficult to establish the gradient direction $f'(x_n)$.

$$x_{n+1} = x_n - \frac{f(x_n)}{f^{'}(x_n)} \qquad 4.3$$

Similarly, a reactive-proactive tracking algorithm adapts a futuristic kind of metaphor to comprehend challenging tracking scenes. The non-linearity of the landscape is resolved using a knowledgeable approach that also identifies other relative optimum around the region of interest. The diversified artificial injections could also facilitate the restoration and recovery of the lost tracks. Furthermore populations are also systematically prioritised using elitism approaches [205] primarily with a prospective to detect the non-linear movement patterns during the manoeuvrability phases. A tracking algorithm built on the DOP perspectives therefore has the following key potentials.

- A set of anti convergence features of the population are defined for each video frame f_n which generally are used for scene specific recovery phases.
- To actively learn the motion model of the object of interest. This would strongly affect the mutation strengths and other relative parametric control measures that define the population behaviours for a robust and expedited convergence.
- To implement an effective change detection procedure, however if the change detection principle and corresponding decision making process are too slow (e.g., it require collating opinions from all agents before making a decision), then this would deter the real time convergence properties of the algorithm.
- It is fundamentally important to declare the correct level of population diversity.

If the diversifying elements/agents have trapped themselves into the local minima, then the solution would have no alternate means to come out of these local traps. The severity of changes in the past could be used to determine the next set of strategy parameters. The projections using the motion history is one way to learn the required diversity levels.

However, to answer completely randomised motions, extra steps and many elaborative diversification phases are needed so that the underlying non-linear movements could be tackled in the video frames. The tracking failure could also result from the noisy environment (e.g., due to the sensor noise, camera resolution and background clutter [206]), it may also be due to consequence of the objects going under occlusion for a length of period. In recursive and repetitive object movement patterns memory based techniques (e.g., Tabu search [207]) could also be utilised in the timely detection during occlusion phases.

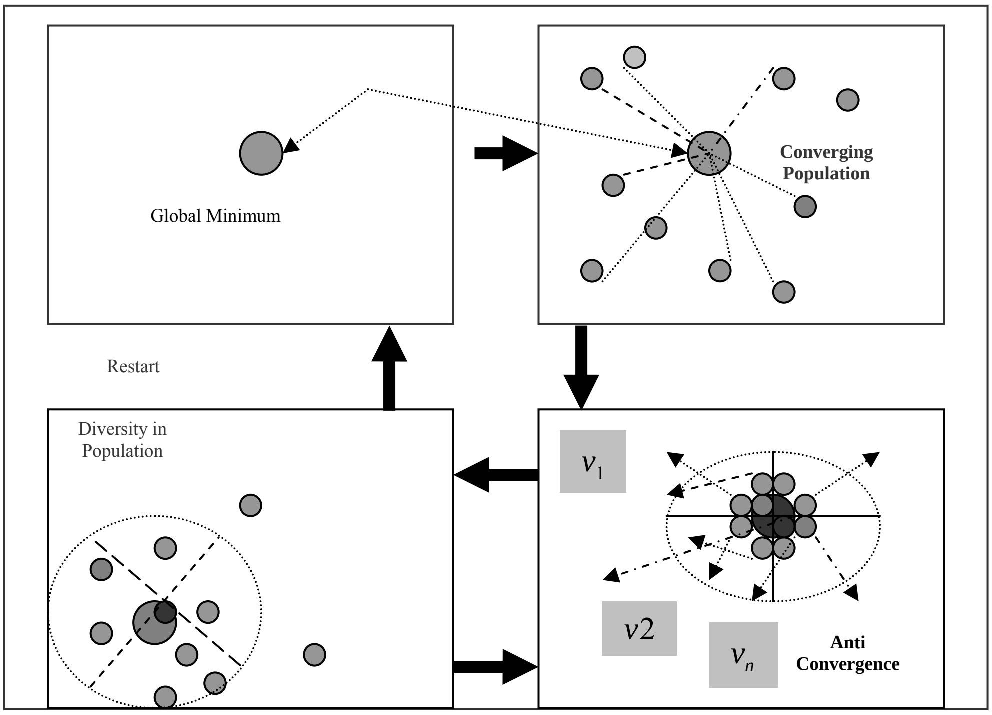


Figure 4.3: Anti-convergent parameters help timely convergence in subsequent frames.

Figure 4.3 shows a generic problematic condition in the particle based systems, after the discovery of a global optimal solution the particles rush towards the surrounding landscapes with an intention to exploit the nearby space. However, the ramifications of the converged population is devastating for tracking in the next video frame (bottom right of Figure 4.3), and need remedial velocity declarations to define their splitting behaviours. A resolute technique to address the dynamic environment is to prohibit the particles from convergences using repulsive forces (as mentioned earlier) and is explored further using Figure 4.3 (bottom left). In natural unforeseen tracking cases, where there are more chances of frame corruptions due to the noise, it is much more feasible to facilitate and develop a restart strategy with strategic placements of agents to reduce the tracking time. Therefore, the tactical initialization of particles plays a significant role in both static (Section 3.2) and DOP scenarios.

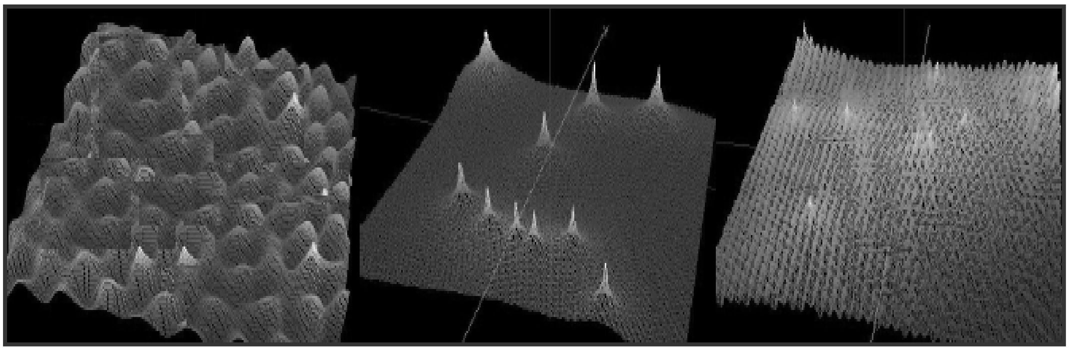


Figure 4.4: Artificially created DOP landmark using Moving Peak benchmark problem.

The mathematical description of the evolutionary dynamic optimisation test bench problems (EDOP) would go beyond the scope of this thesis. However, a typical scenario is portrayed in the context of Figure 4.4. The height and width of the peaks are controlled in this moving peak benchmark test problem (MPBP) [208] using time changing assignments of the global best solution (ranging from a much cluttered environment on the left to a rather simplistic landscape on the right). Figure 4.5 is also presented here in order to relate the EDOP and a general tracking scenario, a cluttered tracking scenario is presented, where, the foundation rules of a combinatorial optimisation are applied to differentiate among various moving objects (in order to differentiate a white car enclosed within a tracking window).

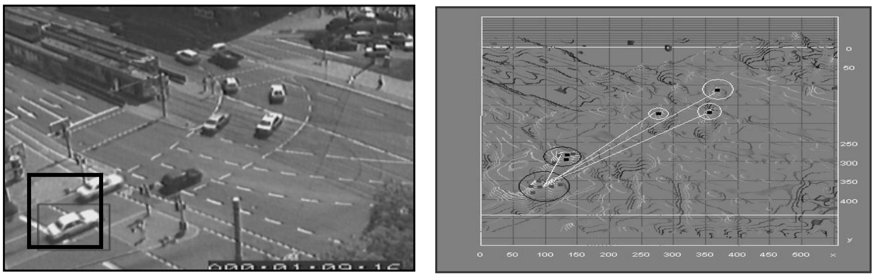


Figure 4.5: Tracking of a white Car using EDOP and its corresponding contour landscape plot.

The tracking terrain could also be more rugged as shown in Figures 4.6 and 4.7. A frame from the ant tracking sequence has been presented, which exhibit the severity of the task due to the resultant shadows from the maze boundaries. The histogram ball plot of the ant (on the right of Figure 4.6) shows the colour distribution of the object in the RGB feature space. One major differentiating criterion that could also be applied to ensure optimal convergence is related to the observation of the individual histogram balls within a confined area (suspected of harbouring an optimal region, this would also be explored further using penalising guided search). In Figure 4.7, the landscape of a video frame stipulates the similarity of tracking frames and the MPBP.

By studying the contents of Figures 4.6 and 4.7 we may reach to a vital analysis that a task oriented computer vision algorithm is generally composed of two parallel operations. The first stage deals with the feature based characteristics to create a unique identity of a region. In contrast, a second stage relies on parallel detections (ideally through the scale free searches explained in Chapter 3), to discriminate among the potential global optimal. If the scene conditions are dynamic in nature than the profile updates are also mandatory.

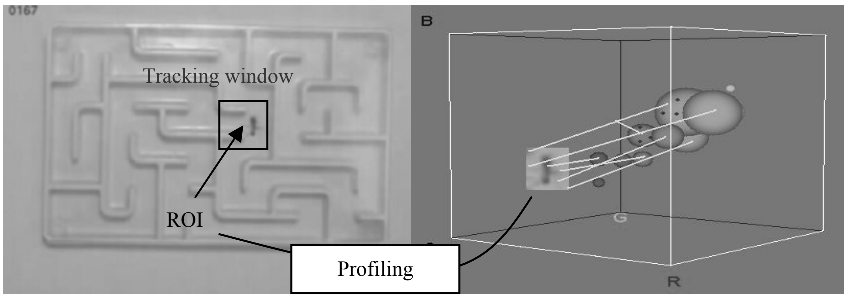


Figure 4.6: Ant tracking sequence with corresponding feature Histogram plot in the inset.

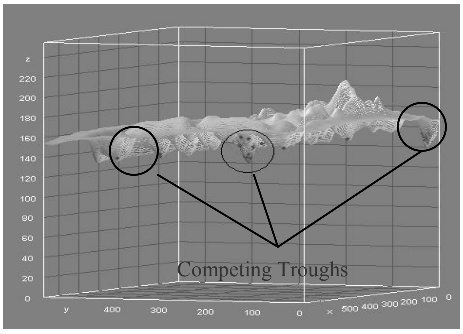


Figure 4.7: Feature Space diversification with multiple scout population in Ant sequence.

4.2 Properties of a proactive tracking system

The fundamental reason behind adapting nature inspired approaches in tracking applications (in this thesis) is mainly due to the significance of their natural abilities to address the dynamically changing environment. Biological life forms have to rigorously tackle the environmental peculiarities on a daily basis in order to survive in harsh conditions. The challenges imposed on the natural colonies and inhabitants of wilderness are enormous, it could range from the daily hunt for food [209], fending off predators, and to devise suitable action plans to tackle natural threats and selection of secure nests [210]. The RSO method in Chapter 3 is an autonomous search strategy motivated by the movements of natural foragers, and several search models are explored in Chapter 4 with an aim to apply those in tracking applications. The most important characteristics of our novel reactive-proactive tracking algorithm are displayed in Figure 4.8. RSO is also inherently a strategically better dynamical optimisation strategy compared to standard particle swarm optimisation.

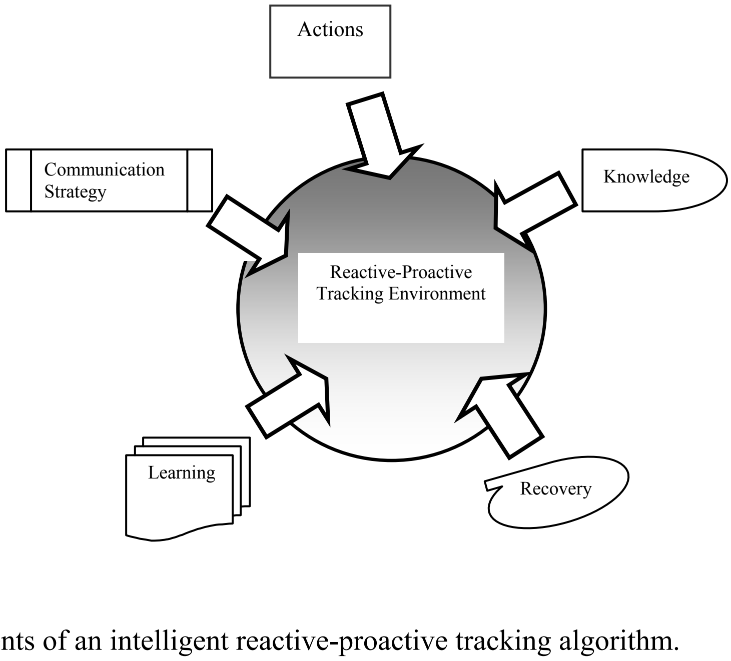


Figure 4.8: Elements of an intelligent reactive-proactive tracking algorithm.

4.2.1 Actions for optimal convergence timing

In a discrete computer grid a larger parametric jump is attributed to exactly the same computational complexity as an alternative shorter flight between two points P and R (Figure 4.9). The course of actions needed to apply the RSO as a meta-heuristics over the swarm based methods are specified in Algorithm 4.1. Once the feasibility of a specific optimal solution has been established, Algorithm 4.1 devises a distance modulated convergence strategy (steps 3-4). The relatively larger jumps are introduced to reach the feasibility space (distance d_{AB} in Figure 4.9) and steps are repeated for the whole population.

The random perturbations (e.g., using Gaussian models) are also introduced at times to enhance the exploration of the algorithm (steps 5, 8). The distance modulation scheme (represented in Figure 4.9 and Algorithm 4.1) does not rely on the tuning parameters Φ_1, Φ_2 as shown in Equation (3.5). Therefore, instead of several small convergence steps ($a_1, a_2, ... a_n$) a larger exploratory jump is preferred to speed up the convergence process (lines 3-7); both Iterative local searches (ILS) and local search methods would also be discussed in Chapter 4.

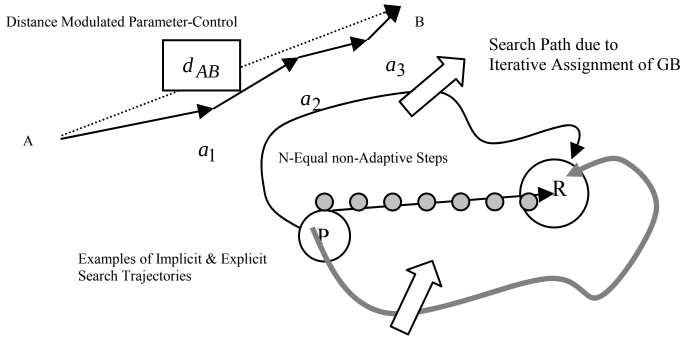

Distance Modulated Parameter-Control
B
Search Path due to
Iterative Assignment of GB
d_{AB}
a_3
a_2
A
a_1
N-Equal non-Adaptive Steps
R
P
Examples of Implicit & Explicit
Search Trajectories

Figure 4.9: The distance modulated parametric control in the agent based systems.

Algorithm 4.1

```
1-   Repeat
2-    For all Agents in Population do
3-       Calculate Distance (…) to local optimal solutions
4-       Devise Distance Modulated Search Scheme
5-       Add random covariance in search paths for exploration
6-       If solution not discovered yet
7-            Define appropriate search regions around potential solutions
8-            Use exploitation algorithms to adaptively search optimal e.g., Using ILS-LS [11]
9-       End if;
```

```
        Evolve Population in accordance with the above *rules*
    End For;
Until Converged;
```

4.2.2 Recovery of the lost tracks.

The experiments conducted in Chapter 3 are broader attempts to find the optimal solutions through detection algorithms, as there is an absence of a propitious scheme in which the gradient information would have been exerted. A similar detection mode could also be implemented in the object tracking, and if it is prompt enough, it might also facilitate a real time tracking (a prime objective). When the particle population is colossal, every agent occupies a place in space, and then such situations are conventionally referred to as brute force searches (BFS) [211]. One of the traditional BFS techniques is the standard mean shift algorithm (MS) (Section 2.5.4). One of the repercussions of the absence of a recovery stage in MS is the absolute search fiasco, such MS adversities are frequently observed in the circumstances when no objects like features are detected within an observational window. To incorporate robustness, hybrid methods could be applied (to both expedite the MS convergence and to rectify errors), static and dynamic particles could be utilised in this context [212] (as shown in Figure 4.10).

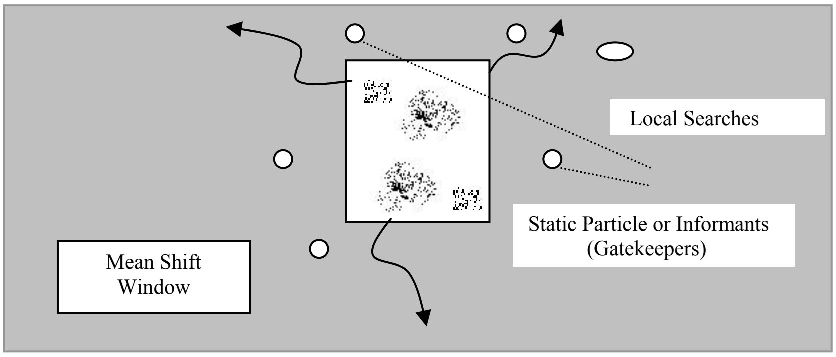


Figure 4.10: A hybrid robust MS variant resulting from integrating particles and density mode.

By incorporating both static and dynamic sensors [213], a recovery phase is induced onto the standard MS algorithm. The task of the static particles (which are strategically placed around the MS window) is to act as gatekeepers, and develop a Meta-heuristics by granting a sense of direction to the tracking window (arrows in Figure 4.10). The role of the dynamic particles is to assert the dominant modes within the MS window (Figure 4.10). The flow diagram of the recovery procedure is shown in Figure 4.11.

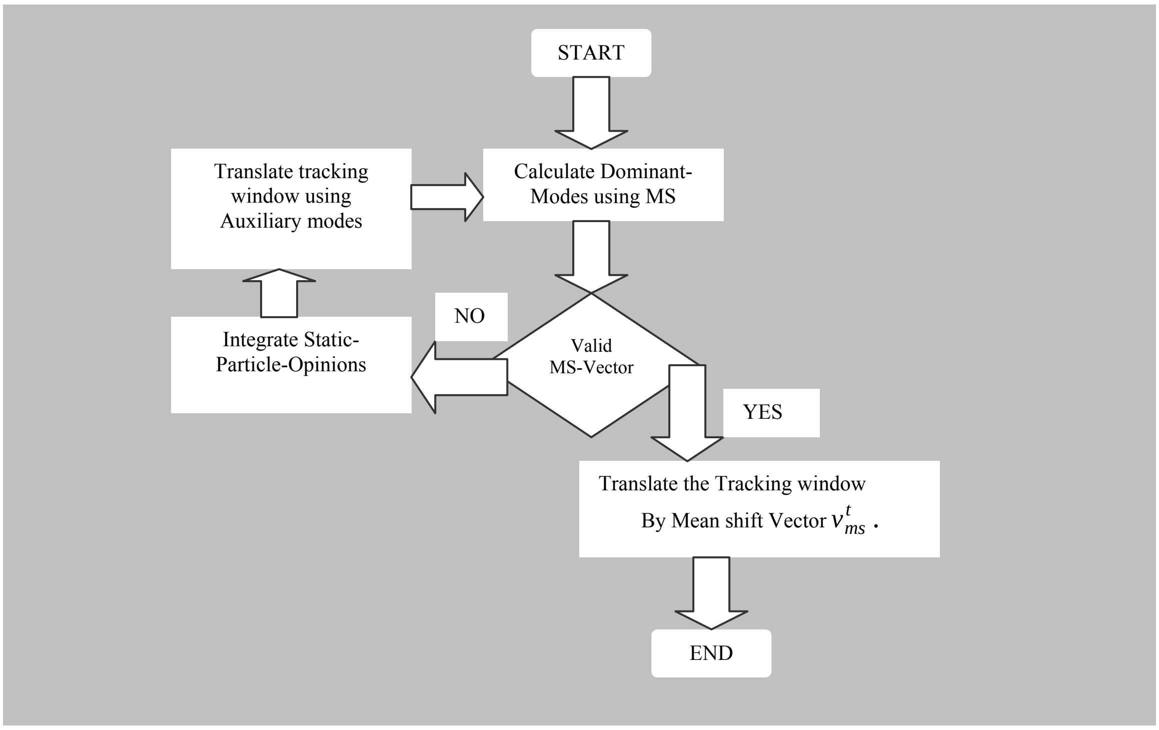

START
Translate tracking window using Auxiliary modes
Calculate Dominant-Modes using MS
NO
Integrate Static-Particle-Opinions
Valid MS-Vector
YES
Translate the Tracking window By Mean shift Vector v_{ms}^{t} .
END

Figure 4.11: A detection and tracking cycle of a hybrid MS algorithm.

In Figure 4.11, a recuperative process (over the MS algorithm) ensures that the algorithm robustly catches up with the dynamic object, and therefore the lost window is timely recovered in a video frame. First, the mean shift vector v_{ms} is calculated using Equations (2.17)-(2.19), and in case no object like feature are detected then the opinions of static particles are collated. The tracking window could also be shifted in Auxiliary mode to gather better local information of the hypothesis projected by the static particles. Furthermore, another remarkable characteristic of particle based methods is that both dynamic and static particles could be employed as computational agents. Static particles (similar to random scouts in Figure 4.2) are computationally more effective methods in order to anticipate motion in complex vision applications. The objective function (e.g., matching RGB colour histograms in tracking) values in the vicinities of the static particles are also calculated. The detection window normally hovers around the potential optimal regions until a dominant density mode is detected in an image frame. Theoretically speaking, the widened dynamical searches are homogeneous to the covariance matrix adaptations in the standard Kalman filters. However, due to the hybrid nature of the tracking window, and because of its auxiliary or flexible nature, it is more diverse and robustly adjusts its scale of measurements without complicated matrix operations.

4.2.3 Learning the motion model.

Many real world tracking problems disclose heterogeneity in their movement sequences. An automatic surveillance system [214] has to extensively entail both detection as well as tracking phases (due to the nature of the operation, where the tracked body might change its shape, disappears in several frames, and re-emerge at an entirely different search area). In contrast, tracking the flow of traffic on the motorways has a certain degree of linearity, and the direction of travel could be predicted to an extent as the movements are generally in compliance with the local traffic laws. An airborne object on the other hand might not be reprimanded by a stringent course of actions, and due to an extra degree of freedom may be allowed to move in a non-linear manner. In Biology, trackers are used in labs to understand the behavioural pattern of reptiles and insects, and their motions appear to be predominantly random with no deterministic components [215].

The inter-frame displacements in video sequences therefore could take place in a specific Euclidean plane (the motions due to the gravity alone) or might enumerate linearity as well as the projectile like characteristics [216]. The tracking essence therefore is in conceiving the motion peculiarities of the relevant situation (projectile/linear). Generally, the efficacy of tracking algorithms could be significantly improved if the larger dynamics are already known to the analyst programmer. In the computer vision systems, historical motions could effectuate the crucial task of narrowing down the search space. In contrast to the Markov assumption [217] (where the current state of the system s_t is independent of the ones observed during $s_{t-1}, s_{t-2}, \ldots S_{t-n}$), the previously discerned measurements and models (in feature space) could also be applied to prioritise the search phases in the object tracking modules.

Due to the exuberant motions observed in the world objects, both deterministic and stochastic trackers are needed to promote a balance between an area specific and to develop diversified search rapports. Therefore, a more concerning tracking strategy is to indulge all possible rectifying measures to scrutinize the search space (see Section 2.5.2). One major research question that was undertaken in this thesis was to analyse the roles of the motion models onto the tracking efficiency. An inherent flaw in the Monte-Carlo particle methods [218] (in the view of the author) is that the particles are dispersed in space-time using pre-determined models (Section 2.6), which in majority of the cases prove counter productive (because of the re-sampling and allocation of preferences).

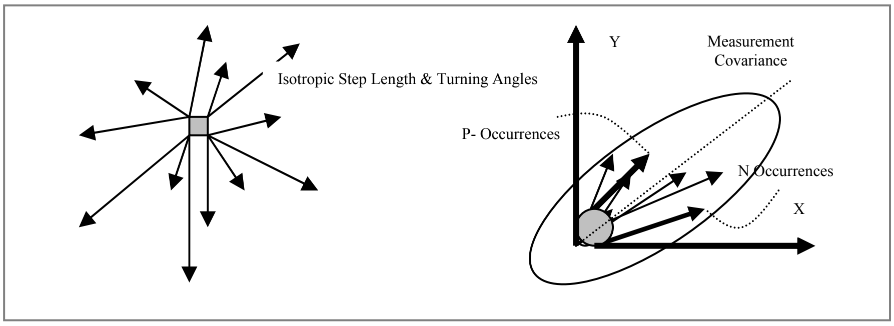


Figure 4.12: Two different characteristic shows randomness and predictable motion models.

The RSO tracking (on the other hand) is entirely based on the foraging behaviours [219] and therefore no explicit models are generally required to track an object (in a generic scenario). Furthermore, the nested searches in RSO cooperate in discovering the dominant modes of motions and guide the optimising process. However, as it was analysed in Sections 3.3 and

3.4 that the search correlations reduce the computation complexity, the historical motion are seemingly beneficial for achieving a real time tracking. The anticipated motion vectors are stored in our algorithm to prioritize searches as shown in Figure 4.12. The determination of the motions in the ant tracker (on the left) displays a randomised pattern of movements, whereby, in a pedestrian tracker (right) displacement logic could be emphasised to correlate search. The historical motion trajectories are stored in memory arrays (using Matlab) and are validated as the tracking progresses in time.

4.2.4 Environmental knowledge for the optimal particle convergences.

There are several important paradigms that must be entertained in order to reinforce the particle awareness in their search space. The partial evaluation of the objective is one such possible technique to boost the environmental know-how of the sparsely distributed agents. The detection of a dynamic body through its bin identification function alone $b(x_i^*)$ (Equation 2.13) is an example of the partial evaluation of its colour density model. Furthermore, the size of the object of interest could also be utilised to differentiate between targets and distracters.

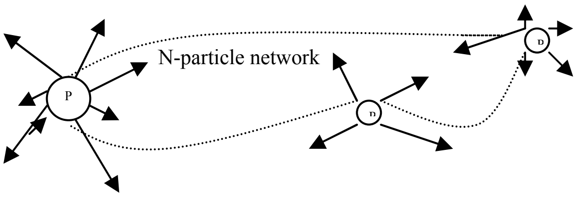


Figure 4.13: The network of n-knowledgeable particles.

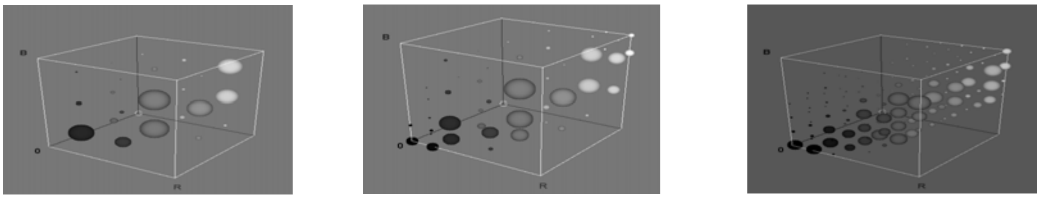


Figure 4.14: The partial information about the histogram distribution of the colour model.

The swarm based methodologies (in their typical format) generally lack an inbuilt feature to penalise distracters, therefore at times, converge to the local solutions (until an agent specifically reaches to a target area). Therefore, the optimal may remain oblivious (for a length of time), and a badly tuned algorithm with ineffective population strength further exacerbates the situation (Section 3.3). As shown in Figure 4.13, a particle's environmental knowledge could be profoundly heightened by the random walks conducted in its immediate neighbourhoods, and the objective is to integrate the landscape data with an underlying aim to translate its own position in the search space.

The allocation of partial objectives (e.g., the relevant histogram balls in Figure 4.14 are all essential components of the overall information) enhances the environmental scanning abilities of the particles, and such knowledge is exploited to determine the depth of the field (tracked object). By composing a circumstantial particle network (Figure 4.13), the algorithmic complexity of the algorithm is substantially reduced, as the non-sampled areas (e.g., ridges and valleys) in the search space could be identified using this knowledge (e.g., line of sight) in Figure 4.15.

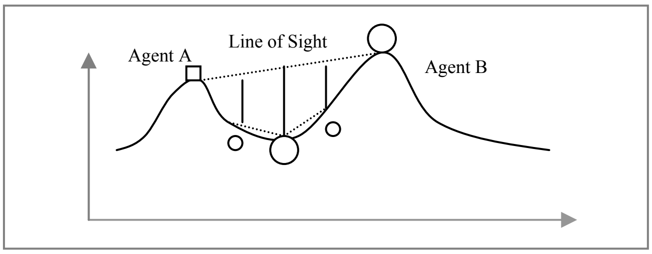


Figure 4.15: The ridges and valleys are easier to be sampled using line of sight trajectories.

A rather conflicting optimization scenario in PSO [220] is portrayed in Figure 4.16 (the bottom equations replicate the velocity/positional updates in the PSO). The optimum trajectory for one particular particle (from its initial position) is represented by the bold arrow on the top of Figure 4.16. Instead of adapting the shortest possible trajectory, the state vector of the particle $\begin{bmatrix} x_i \\ v_1 \end{bmatrix}$ is wrongly modified due to the changing global assignments (as represented by the arrows). However (as evident in Figure 4.16) the size of the optimal (larger circle) is remarkably different than the rest of the objects (triangle, square etc), therefore if the shape information is encoded into the tracking algorithm at an earlier stage, the unnecessary iterations could be avoided for an optimal tracking time.

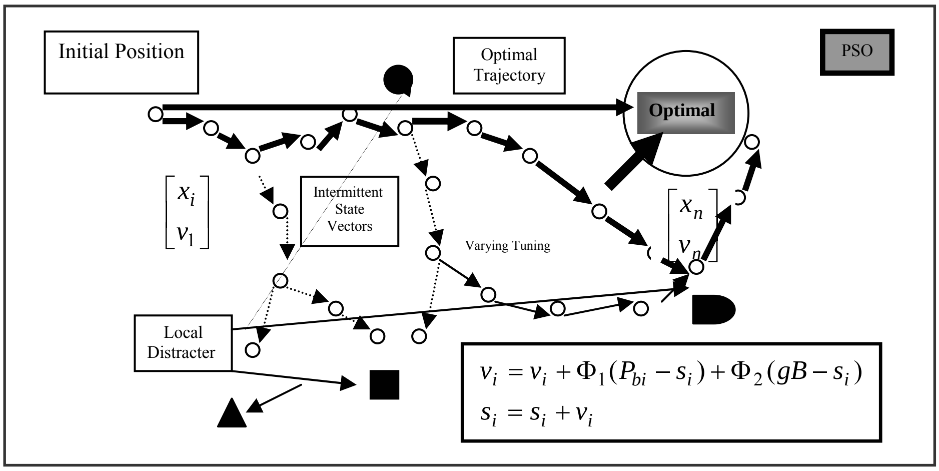

Initial Position
Optimal Trajectory
PSO
Optimal
$\begin{bmatrix} x_i \\ v_1 \end{bmatrix}$
Intermittent State Vectors
Varying Tuning
$\begin{bmatrix} x_n \\ v_n \end{bmatrix}$
Local Distracter
$v_i = v_i + \Phi_1(P_{bi} - s_i) + \Phi_2(gB - s_i)$
$s_i = s_i + v_i$

Figure 4.16: The divergence of particles away from optimal due to varying global assignments

To incorporate the penalising models, we researched along the directions imposed by the Guided local search algorithm (GLS) [221]. As explained in Chapter 3, the radical searches are autonomous behaviours inspired by the natural foragers (ants and bees). The foremost important factor in these spectacular natural agents is that, they appear to have a genetic assignment to differentiate between resourceful directions (e.g., viable food locations) [222]. The manner in which the non-optimal solutions are discarded by the follower ants and bees is a matter of larger interest for a transpiring tracking system. The activation signals generated by the successful hunter bees (e.g., in nest site selection) are substantially similar to the mathematical GLS as expressed in Equation (4.1).

$$g(s) = f(s) + \lambda a \sum_{1 \le j \le m} I_j(s) p_j \qquad 4.1$$

The augmented cost function $g(s)$ uses a penalising approach to distinguish targets from the relative distracters. In terms of tracking application, we can interpret $I_i(s).p_i$ as a target localization map of the object of interest, whereas, $0 < \lambda < 1$ is used to specify diversity levels.

More thorough searches are conducted around regions where object like features are detected, whereby ‘a’ is a problem specific constant in Equation (4.1). The penalising costs calculated by weighing the indicator functions using the p_i models (and summation is carried out along all dimensions as in Equation 4.1) is one manner through which the non-optimal convergence are rectified in agent centred tracking. In this thesis, we have successfully programmed the bin strengths (as an indicator function) using the object model, $\sum b_i$ as an indicator function to penalise clutter.

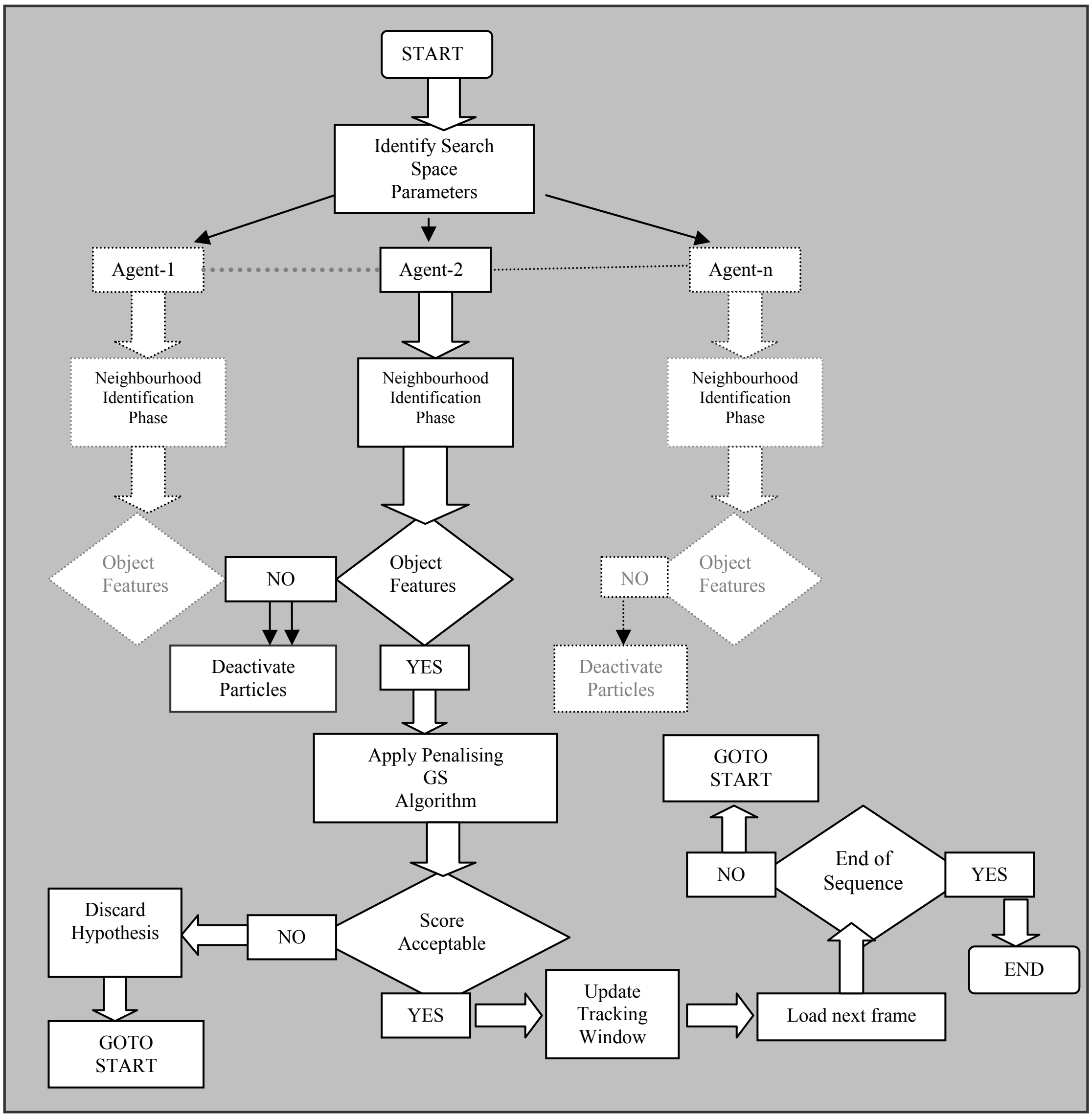


Figure 4.17: Tracking using a hybrid (GLS and RSO) algorithm in Section 6.8-6.14.

Further test benches would be presented to the readers in Chapter 5, which would also facilitate to understand the roles of the partial function evaluation (PFE) and GLS. The flowchart in Figure 4.17 deploys n initial agents in the search space (RSO could be used in this context to narrow down the region of interest). During the next phase, particles conduct local searches and collate neighbourhood information (Figure 4.13). The mean positions of agents are updated using the local information, and the particles which do not meet the fitness criterion (the candidate/detected RGB models do not match the stored priors) are deactivated. Only the winning particles therefore enter into the next phases in our algorithm. The penalising GS approach is then applied to compare the identity of the object with the stored memory models (the object size and the observations of relevant bins $\sum_n b(i)$ are simultaneously applied).

The tracking window is updated once a required GS score is achieved; otherwise, this proposed hypothesis is rejected, and parameters of search are altered to incorporate more diversified searches (as shown in Figure 4.17). The tracking algorithm (Figure 4.17) relies on all three fundamental characteristics devised in Section 3.3. The underlying logic (in this real time system) is that the computational cost is significantly lessened due to the deactivated particles as majority of searches are non-nostalgic in nature. The tracking algorithm finally terminates when the end of sequence is detected.

4.2.5-The communication channels in the particle based methods.

The communicational line among population agents plays a crucial role in the success probabilities of a converging evolutionary algorithm. Through these communications, a computational agent deduces a kind of emotional intelligence [223] about its role in particle societies, and of its peers [224]. However the communication in computational environments (e.g., in the Cartesian space) does not have to deal with the usual perks and challenges experienced by alive natural environment agents. The search behaviour of the natural world foragers would be analysed in the next sections, and some implicit communication channels (e.g., trail pheromone [225]) are critically discussed. Similarly, the waggle dances of the forager bees are redundant phenomenon (in our view) and impractical approach. Once a suitable food resource is detected, a forager does not have to return to its base (to solicit) in artificial landscapes. Furthermore, in contrast to a direct channel, an implicit form of communication may also exist by a third party channel (e.g., the entity c could be a trail path) as in Figure 4.18.

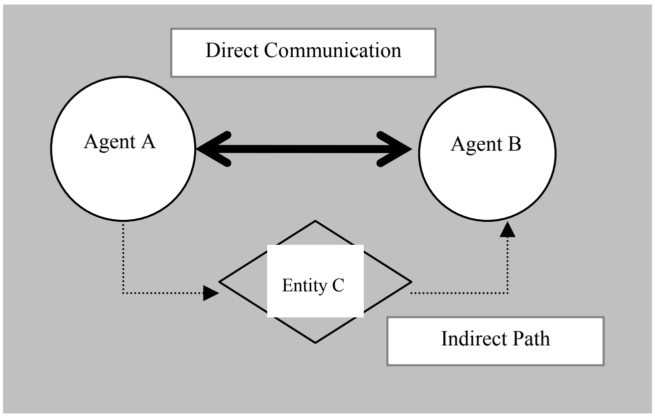


Figure 4.18: The direct (explicit) and implicit communication between particles.

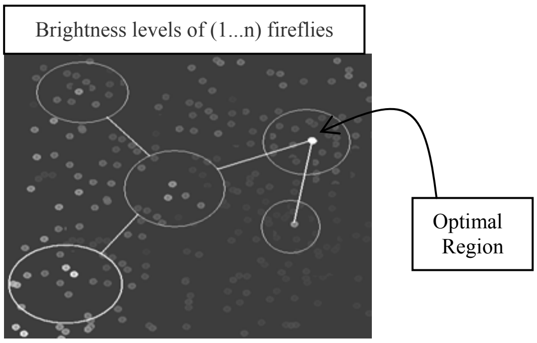


Figure 4.19: The communication radiuses in the firefly algorithm (FA).

To establish a further insight into the communication methods, the readers are encouraged (a detailed discussion is beyond the scope of this thesis) to investigate the operational principle behind the Firefly algorithm (FA) [226]. The core theme in the FA is to spread the particles in the search space, and the fitness level of each and every firefly is calculated using their current positional coordinates. In the context of 3D test functions described earlier in Chapter

3 (e.g., see Figures 3.6, 3.22 and 3.52), the fitness indicators are the function calculations $G_{(X,Y)}$ at each positional coordinate of a translating firefly in the predefined search space. However, in object tracking algorithms the corresponding objective is to match the object colour density in 3D (RGB) colour space. Once all the fitness values of fireflies are analysed, then similar to the real world insects the best ones are illuminated at higher/brighter intensities than the worst ones. During the next stage, implicit channels are established using the natural light attenuation characteristics, and therefore, only neighbouring fireflies are attracted towards the highly glowing insects (an optimal is drawn in Figure 4.19). The type of competitiveness is usually referred to (in the evolutionary literature) as a combinatorial optimisation methodology/framework.

4.3 Heuristic searches in video tracking problems.

The engineering problems are riddled with unprecedented levels of uncertainties, and often a unique solution to the problem is unattainable. In cases where solving a problem with precision mathematical techniques is a valid option, the dimensionality of the space and complex inter-correlations among data variables complicate the problem to an extent that the real time solutions are challenging . The removal of outliers using regularization is a possibility to smooth out the noisy measurements. However, finding an exact analytical solution with a conditioned data field is still gigantically complex to commence in safety specific applications. To tackle high dependencies, sometimes it is feasible to use alternative shortcuts to detect acceptable solutions to the problem. The heuristic shortcuts (HS) [227] are intuitive decisions which could facilitate problem solving in complicated visual systems. Similar to the intuitional decisions made by human beings (where an inner guidance is sought to replace complex cognitive processes [228]), heuristic searches reduce the dimensionality issues through intuitional adjustments of the independent variables. It would be appropriate to write that, heuristics object tracking applies trial-error rules to find an acceptable solution to this computer vision task.

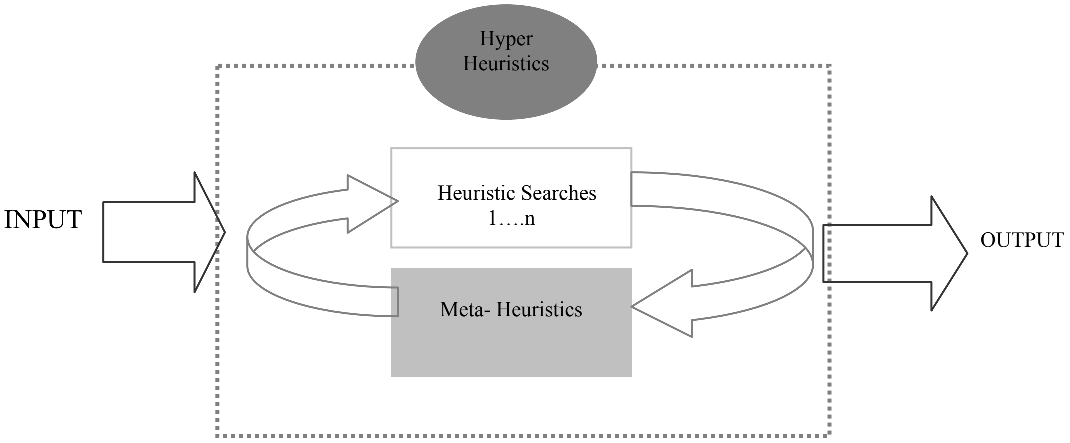


Figure 4.20: The framework of heuristic searches in RSO.

Figure 4.20 is a general HS framework where n parallel/serial heuristics are applied to secure a solution. The role of the meta-heuristic (MH) stage is to identify appropriate search areas to apply the inner heuristic searches. The readers are referred to Section 3.2.3, where the honey-bird and human relationship is discussed, which in our view could be attributed as a strong meta-heuristics. Similarly hyper-heuristics (HH) [229] is a top order rule that supervises the heuristics based solutions, with an aim to supplement the probabilities of success.

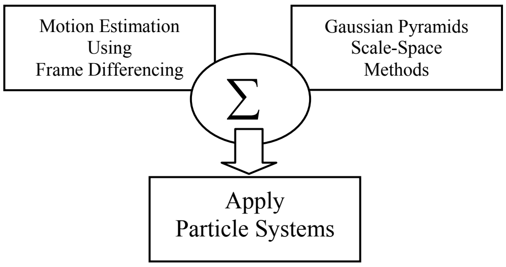


Figure 4.21: Top order stages in the combinatorial optimised tracking systems.

In terms of the honey-bird human scenario, we could also refer the hyper stages as an experiential learning model discussed in Chapter 3, the interpretation of the body language of the birds and further refinements overtime is a hyper stage. The gradient descent methods in Sections 2.5.4 and 3.2.1 are short sighted due to an absence of a hyper-heuristics. Through the experimentation in Chapter 3, we reached a conclusion that the RSO heuristics generates a superior directional sense in particle systems (verified in solving complex test problems, Section 3.5). In computer vision applications, similar methods have been applied (in this thesis) to track an object in real time.

In Figure 4.21, both standard frame differencing and the ones using time differed Gaussian convolved images are applied in order to identify the suspected object movements. Once suitable regions are identified by the top order information fusion stage, the particles are deployed in those competing areas and the result of this combinatorial optimisation process is analysed, and the best solution is chosen as an object of interest. Therefore, despite of an absence of particular dynamical models, objects are tracked with high precision and frame rates in this thesis. Many resolute evolutionary methodologies (as shown in Figure 4.22) also act in an iterative undaunted manner to address dimensionality issues by exploiting the deep-seated heuristics.

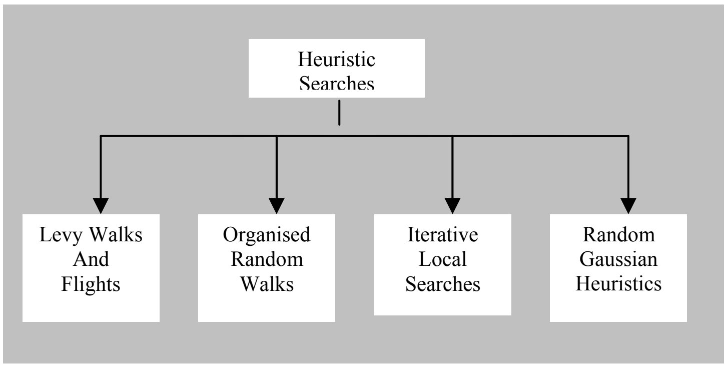


Figure 4.22: Some of the most prominent heuristic searches.

The quantum cloud by Antoney Gromley (Figure 1.1) is a delineation of the organised random walks (ORW) [230], first the solution space is divided into multiple subspaces, and then organised random walks are carried out to characterize the optimal sketch of the image. The foraging strategies deployed through the Gaussian cored heuristics are (on the other hand) highly randomised, and due to their parametric nature, often need explicit definitions of the search covariance. The iterative local search (ILS) [231] is a pseudo-random proposition that imposes local search irregularities to perform objective pattern matching. The ILS

(Figure 4.23) uses non-linear projections (using variability in step sizes and turning angles) to detect translational motions in tracking, and could increase the tracking robustness when applied as a heuristics over the short sighted algorithms.

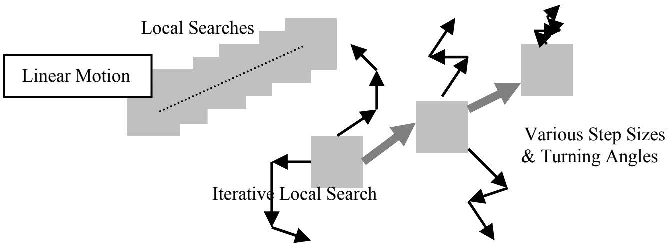


Figure 4.23: An iterative local search algorithm VS the local search method.

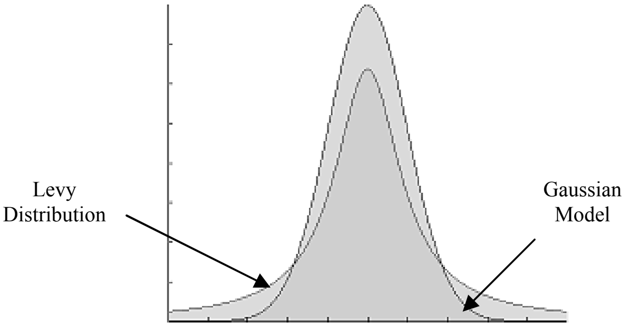


Figure 4.24: A Levy distribution (model) with heavy-tail characteristics.

Some recent studies have also unveiled the fact that many 21st century North African hunter gatherers use Levy like walks and flights in order to reach amicable food resources [232]. The comprehensive mathematical description of the Levy model is out of the scope (readers are referred to [233] [234]), but a general graphical comparison of the Levy model versus Gaussian distribution is presented in Figure 4.24. The heavy tailed Levy function indulges more diversified searches in comparison to the Gaussian probability distributions.

Many foraging insects (bees, ants and termites) also use similar patterns in their daily hunts for food and in the respect of their nest site selections (NSS) [235]. Crist et al investigated the behaviour of forager ants and an excerpt from his renowned paper has been presented here for the sake of this discussion [236], “Individual ants exhibited fidelity in both search site and native seed species. Spatial analysis of foraging movements showed a highly oriented travel path while running, and an area-restricted path while searching. Searching ants moved in a manner consistent with a correlated random walk. The deterministic component of path fidelity and the stochastic component of search may override energetic foraging decision in individuals P.Ooccidentalis ants".

The forager bees also apply local and randomised heuristics to collect nectar from sporadically distributed food resources. It has been observed that only $(2-5)\%$ insects (from an overall population) depart their hives at any one time to search for food, and on their returns perform Waggle dances to convince remaining foragers to fly towards the competitive directions [237]. Therefore, the bee colonies reciprocate a natural inclination towards competitive-cooperative population structures. The RSO (Chapter 3) could make explicit use of the distributions represented in Figure 4.22, with an objective to improve the convergence timing. The recruitment processes in the natural colonies could be one-to-one (e.g., the tandem runs, in which a forager ant guides a single novice recruit towards a food resource [238]), or a group of recruits are led by the successful forager ants.

4.4 Novel propositions.

Having considered the fundamental drawbacks in the general video tracking (Sections 2.1-2.5), and after laying the foundations for a new particle oriented theory (RSO) in Chapters 3 (see Sections 4.1, 4.2 as well), we are now in a privileged position to present our key propositions in this report. The characteristics defined in Section 4.3 are also the main paving stones/building blocks that may lead to a better understanding of our adapted approach. Furthermore, this section also acts as a framework for forthcoming discussion of the tracking flowcharts, and provides a general guidance about the experiments conducted in Chapter 6.

I. *Prediction dynamics and state transition models are not crucial / mandatory in the object tracking scenarios. The proliferated errors during the predictive phases augment the algorithmic complexity, and are usually counter-productive gestures. The plant dynamics are therefore replaced (in this report) by competing particle swarms and through random trials (e.g., based on the iterative and guided local searches).*

II. *The partial function evaluation using the bin identification numbers are expedited objective functionalities, and more insightful than the procrastinations witnessed during the explicit similarity determinations (among two densities). A voting strategy based on a multivariate density evaluation at data points, and a Euclidean measure in* L^2 *norm (e.g.,* $\sqrt{(x-a)^2+(y-b)^2}$ *) in colour-space could prove equally effective in the tracking applications.*

III. *Projection of belief space using information fusion and dynamic optimisation techniques, e.g., using nested RSO and parallel populations, random scouts, artificial injections and memory based methods are novel affectations to detect an object of interest in a computerised vision. The .fusion schemes (e.g., of colour and motion models) reduces the search space effectively, hence explorative swarms could be* *initialised in those search regions to increase the particle effectiveness.*

IV. *Hybrid methods e.g., using the particle assisted calculation of meanshift vectors outclassed the standard meanshift in both space and time. Therefore, the meanshift operational basin is automatically adjusted based on the iterative projections of the belief space.*

V. *Simple but competitive heuristics that converges to the local optimal solutions (e.g., formulated around potential regions of interest as circular level sets or spherical structures in higher dimensions) are effective methodologies to address both translational and scale changes between two frames.*

VI. *Standard particle swarm method does not incorporate the local intelligence, e.g., in natural world, all entities have an olfactory sense (OS) that enables agents to alter their search trajectories, and when OS is not taken into account, lead the solutions to relative best regions. The well aware particles have an ability to hop around and autonomously decide their own trajectories, and serve a resolute alternative to the predetermined movements applied in the particle swarm optimization. Therefore the search (in our approach) becomes a distance modulated scheme utilizing a collective intelligence.*

VII. *It is of paramount importance to emphasize that the iterative evolution of agents is not mandatory to resolve complex multimodal problems. Instead, the particle rebirths in projected areas through RSO are more logical approaches to address convergence timing.*

VIII. *The heuristic searches could also be applied in the segmentation applications. The curve evolution timing could be significantly reduced due to the fact that such methods do not require explicit calculations of the curvature (or normal vector field using re-initialisations using signed distance transforms).*

4.5 System Diagrams.

The core theme of this thesis is described along various sections in this report, and it revolves around the fact that computer vision is not a precision oriented science. Perhaps, as emphasised in Sections 2.4.1, a significant portion of the tracking procedure constitutes of integrating tighter and flexible phases (Section 2.5.2). Furthermore, it is more feasible to monitor different phases using a strategy controller so that a correct balance between the speed of convergence and algorithmic accuracy could be achieved (Figure 2.10). One way to achieve this balance is to use experiential variations (embedded in RSO) in accordance with the Kolb model to achieve better convergences.

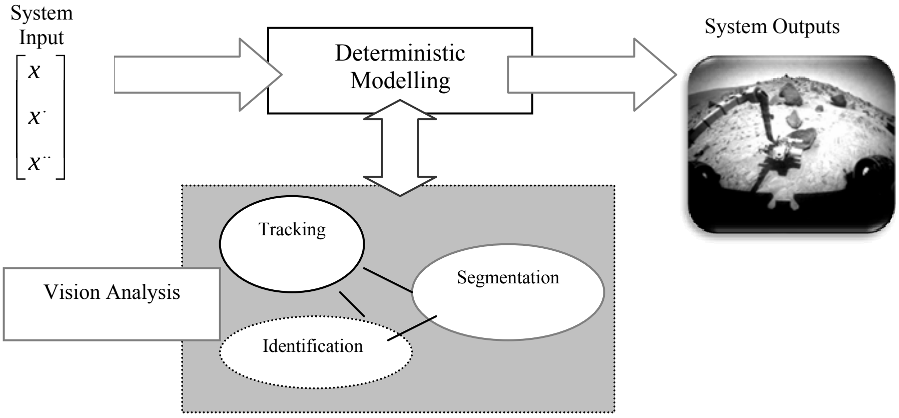


Figure 4.25: A layout of a control problem facilitated with an artificial vision.

Figure 4.25 also highlights the fact that any visual analysis must not be embedded within the system dynamics for a robotic application to prove more robust. Although modelling of the deterministic components is essential to generate precise movements (e.g., to achieve traction control in the electrical motors under various load conditions), a computer vision analysis must be an autonomous entity and should not be part of the overall dynamical model. Tracking as well as many other computer vision task (e.g., segmentation) is therefore a scale free search process (that is successfully applied in Chapter 3 to solve related 3D problems), Especially on the mobile platform, a distributed process using roaming particles overcome many inherent drawbacks of the single solution based approaches (which usually need extensive modelling in matrix formats). The main characteristics of a multiple agent based tracking are shown in Figure 4.26.

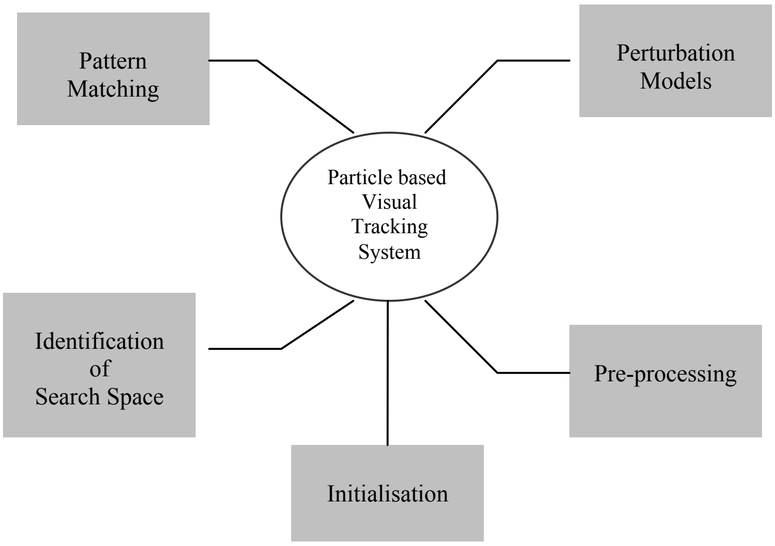

Pattern Matching
Perturbation Models
Particle based Visual Tracking System
Identification of Search Space
Pre-processing
Initialisation

Figure 4.26: Properties of particle based visual tracking applied in Chapter 6.

The timely identification of a relevant search space is important to cut algorithmic complexity, and the underline motive a pre-processing step is to eliminate the unnecessary noisy observations in an image to accomplish faster convergences in Figure 4.26, therefore these two processes work in close conjunction. The most fundamental part of the algorithm is to apply the relevant perturbation models (RSO, PSO and other heuristic techniques are appropriate exemplifications in this context). In a computational environment we need a pattern matching stage (such process are extremely efficient in a biological visual system and equally hard to understand) to establish identities as reflected in Figures 3.58-3.68.An overall picture of the complete algorithm could be portrayed as in Figure 4.27.

Instead of using tighter and precise deterministic components of the dynamics (Sections 2.5.1, 2.5.2), the visual tracking (in this approach) resembles a state model in Figure 4.27. This section is devoted to describe the major sections in diagram 4.27; however a brief detail is presented here. As experimented in Section 3.5, the best perturbation model is generally the one which employs least memory operations (Figure 4.27) as it has been established that there is no algorithmic advantage in recording trajectory changes (e.g., a virtual particle has been found more efficient than all PSO variants in difficult test cases). The strategic controller indulges the required flexibility in tracking (case based software constructs are applied to introduce diversified tracking), and performance based measures could be selected for more meaningful convergences.

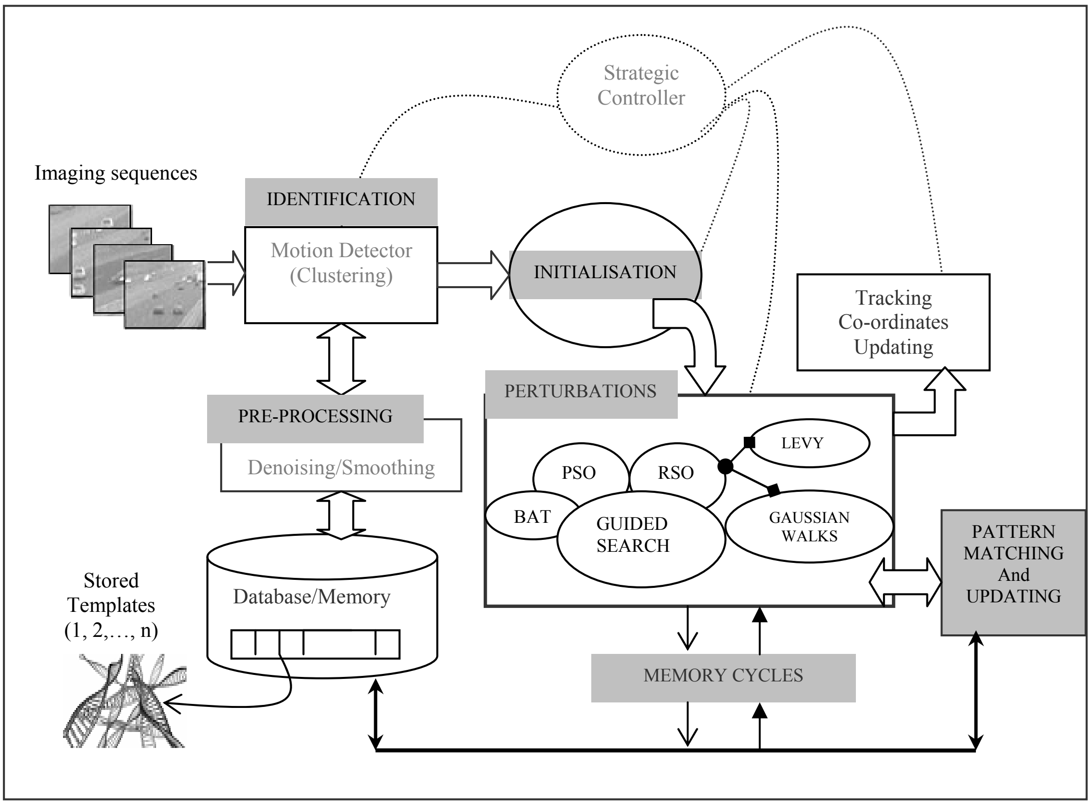


Figure 4.27: Structure of the proposed tracking algorithm.

The identification stage uses top order/meta-heuristics (frame differencing and clustering techniques) to first subdivide image into prioritised clusters/regions, which are later analysed using perturbing agents. At a later stage individual measurements collated by using diffusing particles are integrated matched against stored templates and the tracking coordinates are updated. S. Therefore without using any hardwired logic (e.g., present in Mean shift and particle filter based tracking), the system still facilitate superior and automatic recovery of the tracking window. The metaphorical and systematic measurements only prioritise particles that matched the core features of a stored pattern.

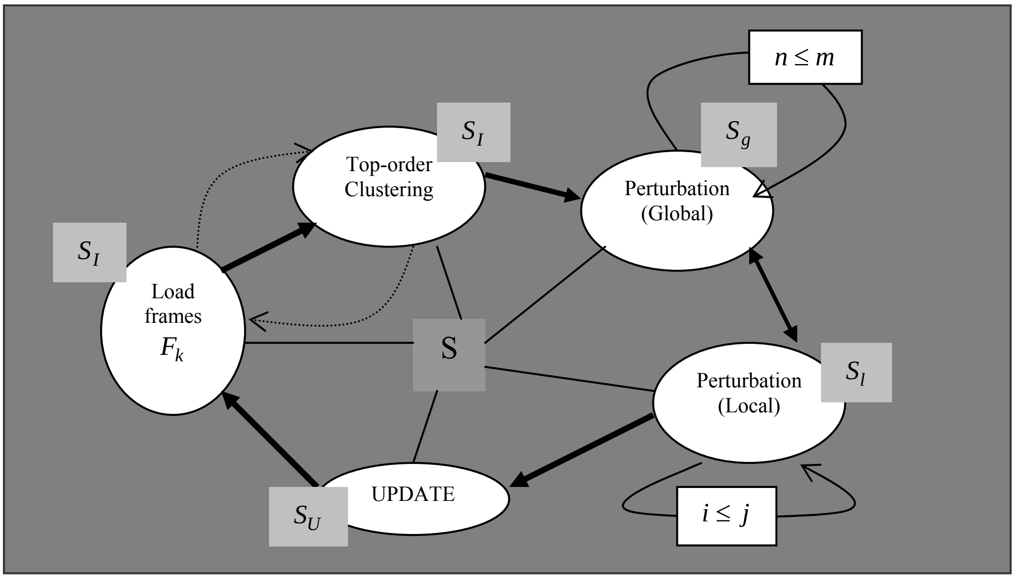


Figure 4.28: Dynamic control policy in the mobile tracking applications.

Figure 4.28 is a simplistic controller strategy; the eccentric control introduced by using a supervisory scheme (S) ensures that unnecessary calculations are avoided by allowing more flexibility in the state jumps. Therefore, the preference to stay at a particular state (e.g., s_g) is dynamically altered using a centralised controller, and depends on the scene conditions (e.g., by manipulating variables (n,i) and (i,j)). The nested searches are implemented using local and global perturbations, and the search parameters (e.g., particle population n_p) are also strategically allocated along with the number of frames needed to impose a higher order heuristics.

4.5.1 Identification of a feasible search space

The task of creating computationally effective tokens is of momentous nature in the field of computer vision. The uniqueness of a region is established by describing patterns in a mathematical format, and such identifiers are generally stored in memory. Colour, texture and shapes are common criteria to group pixels (into specific objects), which are then used to penalise video frames and to track objects. Despite of occupying distinct areas in a feature space, tokens belonging to the similar object (under observation) are unified (Figure 2.3). However, the static analysis of an image (e.g., pixel to pixel searches) is generally a very costly process, and an initial frame portioning could be done by exploiting the time dimensionality by comparing two (or more) sequentially generated images in a digital video.

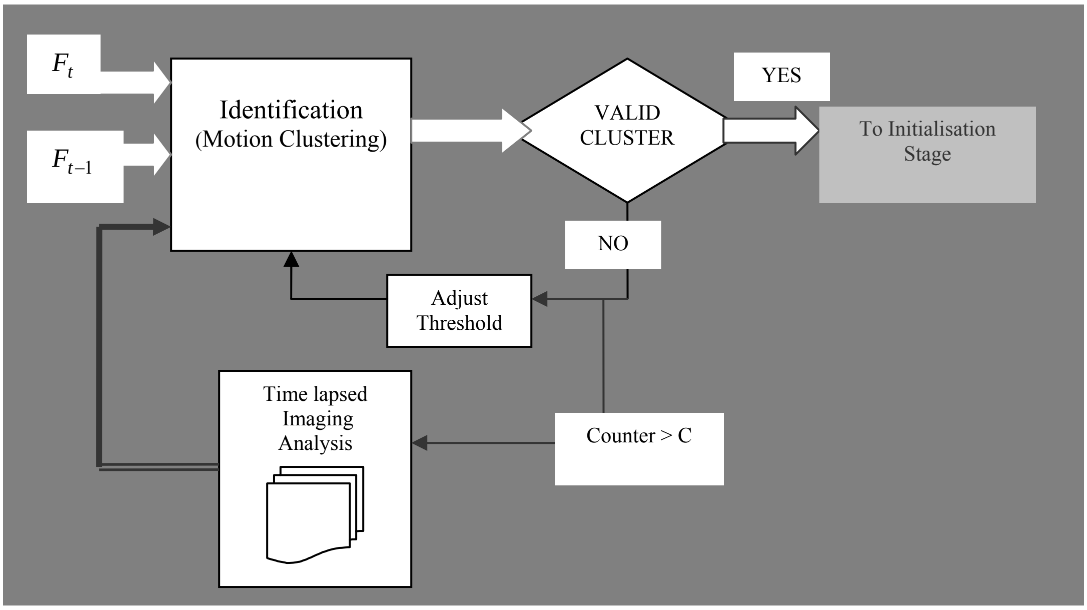


Figure 4.29: A flowchart to formulate a top order clustering and metaheuristics.

Initially two subsequent frames (f_t, f_{t-1}) are utilised to identify suitable clusters (of pixels) exhibiting a correlated motion (however, any past or an averaged image could be chosen to perform this higher order clustering), the threshold parameter (n_c) is readjusted as an initial strategy if no viable clusters are identifiable. However, if this scheme fails (using counter variable C) the motion based clustering is performed with a variety of previous frame information and relevant subtractions to observe changes in a current frame.

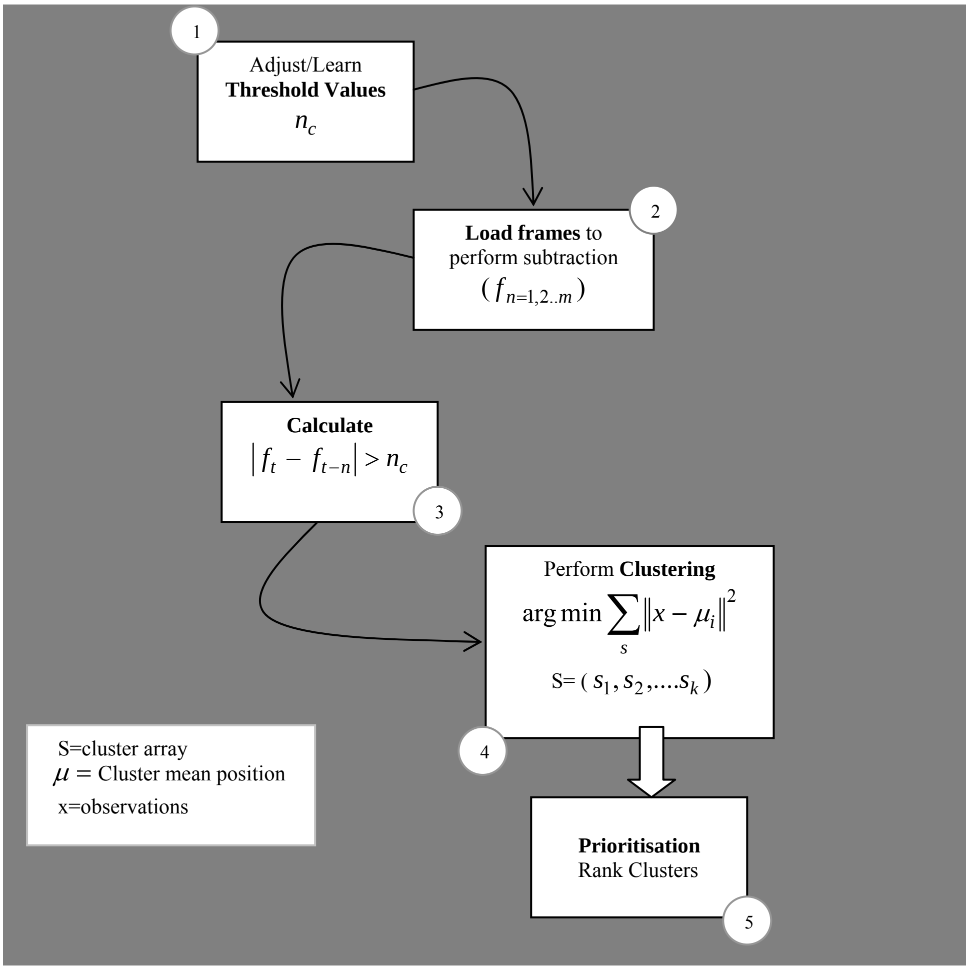


Figure 4.30: The generic flow sequence applied in the frame clustering.

The flow diagram in Figure 4.30 performs top order clustering using frame subtractions, the changes in the intensity levels (after differencing two frames) is analysed using a pre-learnt threshold. 'n_c'. The relevant information is grouped into 'k' regions by minimizing Euclidean/L^2 norm $(\arg\min \sum_s \|x - \mu_i\|)$ of the observations at stage 4. The resultant clusters are then prioritised at stage 5, and submitted for further analysis using appropriate particle initializations (e.g., random Gaussian) in those regions as shown in Figure 4.27. In Figure 4.31, the moving clusters (in a pedestrian crossing situation) are determined using two time lapsed images (27 and 20). The initial surface plots (middle row) are quite complicated, and it may prove much very time consuming to eliminate the static objects through a sequential or brute force search. However, by using just 2 threshold levels (stage 3 in Figure 4.30); the prioritised particle initialisation generates a meta-heuristics to scrutinize the image space in real time.

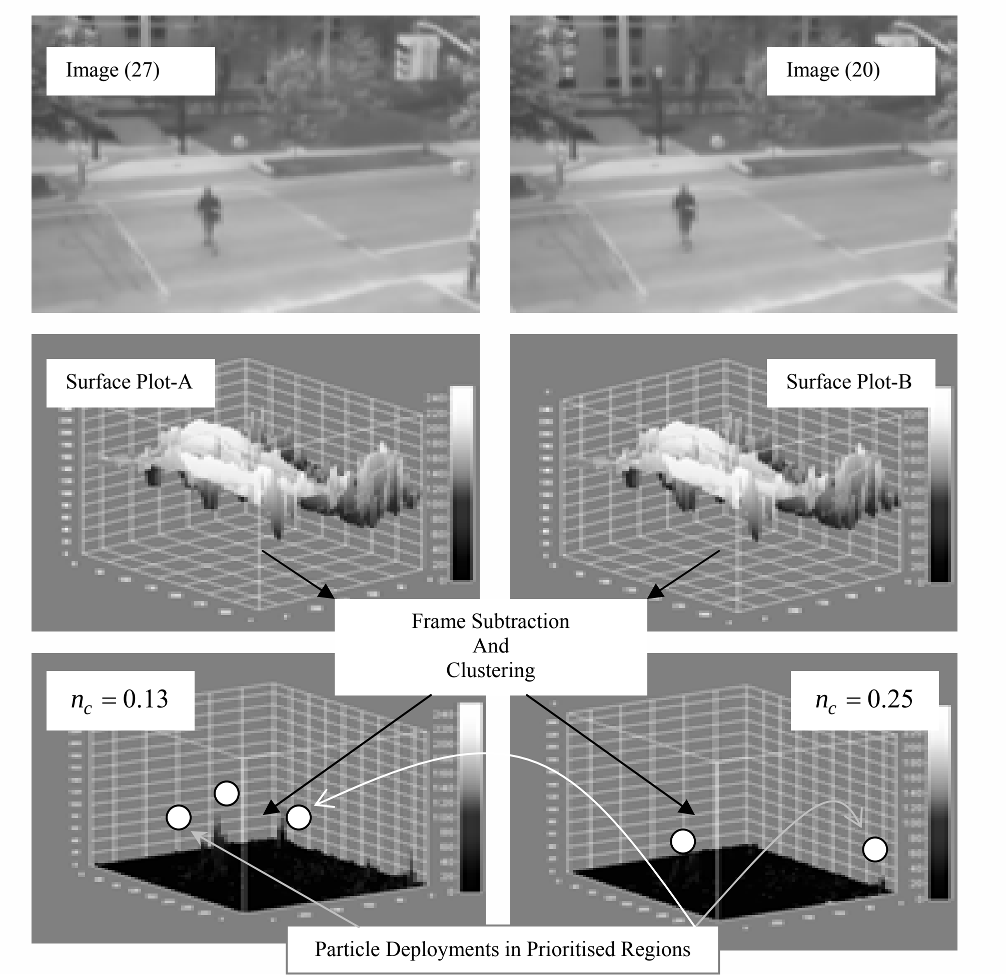


Figure 4.31: Particle initialisations using frame difference and priority clustering.

4.5.2 Pre-Processing.

In order to highlight the role of this thesis, the scope of research in the field of digital imaging is discussed in Section 2.2. Generally, the image enhancement and pre-processing techniques are computationally intense, and could contradict with our prime objective of tracking in real time. However, some cost effective image processing (IP) techniques are discussed here for a self contained reading. The main IP routines used in this thesis revolve around the non-penalising approach of pixel diffusions using isotropic Gaussian functionalities. The term ‘diffusion’ in imaging refers to the flow (as a general mathematical diffusive process) in which a specific pixel neighbourhood is applied to assign values to a central pixel, and this produces a blurring effect is used to define the scale of information and to remove discontinuities. Equation (4.2) assigns average intensity (in a neighbourhood N) to a pixel x

$$x = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad 4.2$$

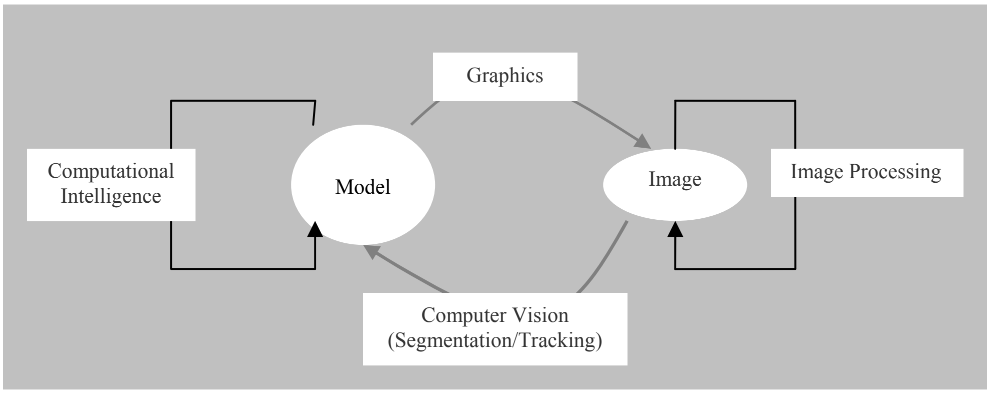

Graphics
Computational Intelligence
Model
Image
Image Processing
Computer Vision (Segmentation/Tracking)

Figure 4.32: The role of image processing routines in the computer vision literature.

The flow diagram presented in Figure 4.32 highlights that the computer vision tasks attempt to infer a hidden model (e.g., a model of object motion) after conditioning frames using image processing techniques (the computer graphs is a reverse process).

$$x = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad 4.2$$

Denoising and blurring (to produce scale space image pyramids) are two main pre-processing stages applied to track objects in Chapter 6 (Sections 6.1-6.13). Image denoising could be broadly classified into two categories, and five common filters are categorised into isotropic and anisotropic filters (in this respect) as shown in Figure 4.33. In traditional mean image filtering and denoising (Equation 4.2), each pixel RGB channel data is replaced with a mean intensity level which is calculated using a mask (Figure 2.8) of N neighbouring pixels.

A median filter (on the other hand) reorganises pixel data into a list of ascending values, and the median intensity level is used to replace noisy pixel value. Whereas in Gaussian smoothing (applied in Equation 4.3), an original noisy image $I_0(x,y)$ produces a family of digital images $I(x,y,t)$ after applying mathematical convolution to itself with several different Gaussian kernels $G(x,y;\Sigma)$ of changing variances Σ. Due to the isotropic nature of this particular kernel, the resulting images exhibit a blurring effect which generally reduces the information content of an image (generating a scale-space image pyramid). The wide-ranging imaging information contents (finer to much coarser level) are then used to locate a region of interest using secondary foraging/perturbing particles as mentioned earlier.

$$I(x,y,t) = I_0(x,y) \bullet G(x,y;\Sigma) \qquad 4.3$$

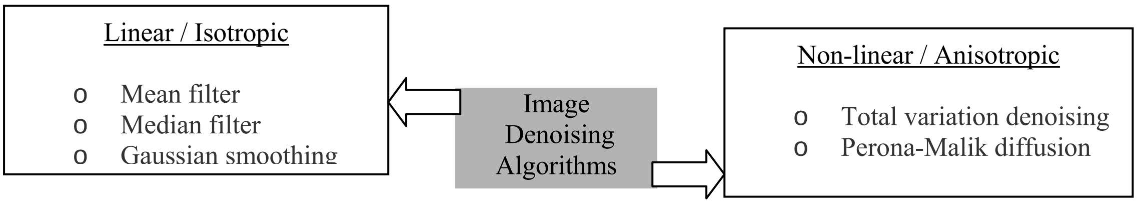


Figure 4.33: Classification of denoising algorithms in image processing applications.

Figure 4.34 depicts the effects of introducing Gaussian blurs into a campus image (Aston-Webb, University of Birmingham). The filter parameters (mask sizes and standard deviation) are identified at the bottom (Figure 4.34). The image pyramid (in this case) consists of only three subsequent scale-space images. However, in practical recognition systems (e.g., scale invariant feature transform (SIFT) [239]), it usually consists of tens of sequentially blurred images which facilitate prominent feature matching process of an object. In Section 6.1, a range of similar scale-space images are applied to detect top order object movements.

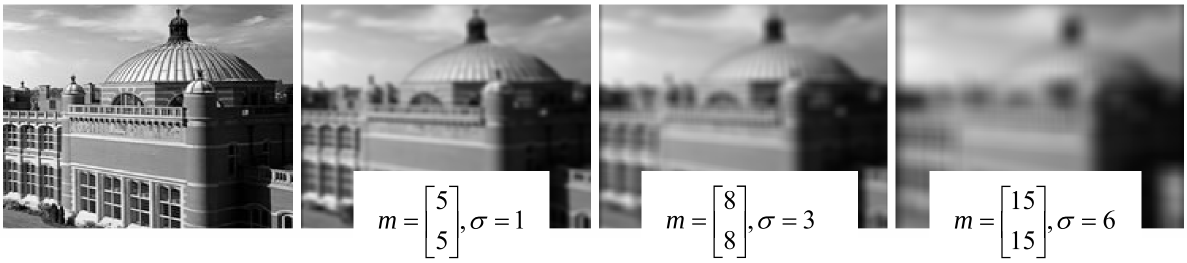
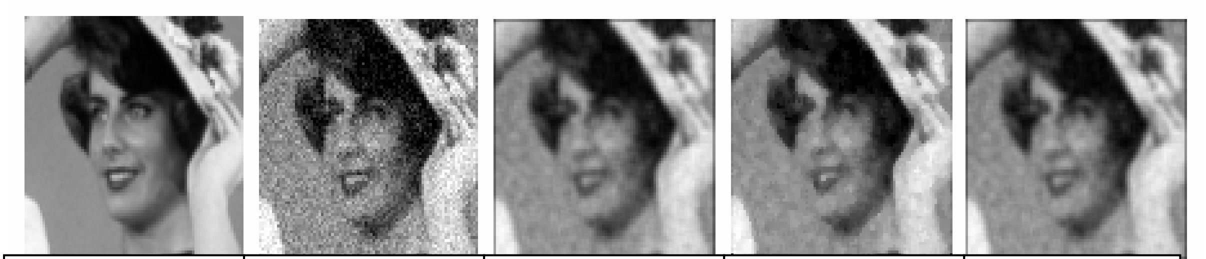

$m = \begin{bmatrix} 5 \\ 5 \end{bmatrix}, \sigma = 1$
$m = \begin{bmatrix} 8 \\ 8 \end{bmatrix}, \sigma = 3$
$m = \begin{bmatrix} 15 \\ 15 \end{bmatrix}, \sigma = 6$

Figure 4.34: Applying different Gaussian blurring operators (means and variances).



Original	Gaussian Noise $\mu = 0, \sigma = 0.005$	Mean Filter 0.003 seconds	Median Filter 0.0021 seconds	Gaussian Filter 0.00183 seconds

Figure 4.35: Effects of applying various denoising filters on a library image of Lena.

In Figure 4.35, a Gaussian white noise ($\mu = 0, \sigma = 0.005$) was introduced into an original image of Lena, and the responses of various filters (mentioned earlier) are replicated on the right. All linear denoising filters (when applied on this typical computer vision library image of resolution 14400 pixels) worked very efficiently, although, more blurring is introduced by the isotropic Gaussian filter (mask size= 25 pixels, $\sigma = 1.5$). Furthermore, all three linear filters introduced in this section are remarkably efficient in terms of algorithmic timings (as shown in Figure 4.35), and could be applied in the context of real time tracking applications.

One possible layout of a scale-space frame difference (SSD) scheme is shown in Figure 4.36. There are $'N'$number of filters in this bank, and both an incoming frame (at time instant$t$) and n previous frames (stored in memory) are used to detect possible movements. The state vector in a general tracking application comprises of mean positional coordinate vector or centroid of a region of interest, i.e., $m_h = \begin{bmatrix} y \\ x \end{bmatrix}$, and the spread of measurement is $\begin{bmatrix} S_y \\ S_x \end{bmatrix}$ (which forms a feasible space for particle based searches). A data fusion module (Σ) collates opinions of $N - SSD$ modules (Section 4.4.1). A possible fusion strategy is the simplistic majority voting or consensus (e.g., if total votes in favour of a current hypothesis $> \frac{N}{2}$). In hybrid tracking particles are therefore initialised using the proposed state vector at this stage and further refined by applying perturbation models.

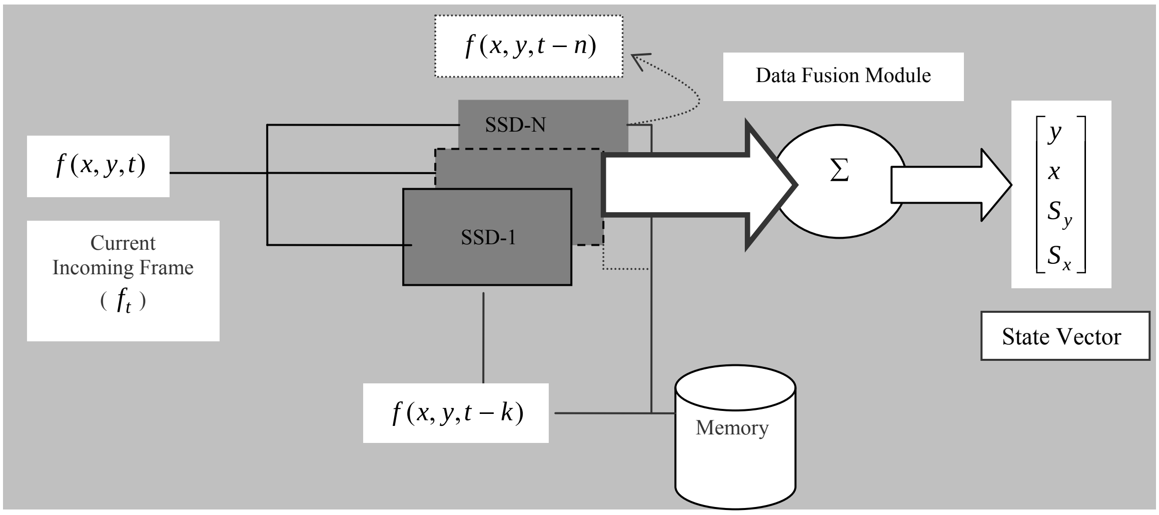


Figure 4.36: Schematics to apply SSD filter-banks in hybrid (particle and motion) tracking.

Imposing meta-heuristics (similar to the honeybird-human scenario described in Figure 3.25) by integrating motion estimations is a very strong tool at our disposal; it helps to cut down convergence timing in a significant manner. On the other hand, anisotropic diffusion (Figure 4.33) is more suitable for off-line segmentation. One dominant advantage of non-linear/discriminatory diffusions is that the object boundaries are perfectly preserved.

$$I_t(x,y) = c\Delta I(x,y) = c.(I_{xx} + I_{yy}) \qquad 4.3$$

Equation (4.3) demonstrates linear image diffusion (using Laplacian), and is inspired primarily by the solution of the general heat equation [240]. When such diffusion is applied to an image I_t (where I_{xx} and I_{yy} are the second spatial derivatives along x and y image dimensions, c is the conduction coefficient) all high contrast edges are smoothed out resulting in a steady state image version. Figure 4.37 shows the effect of such simplistic diffusion (prescribed in Equation 4.3); clearly, all edge information has been lost in the process.

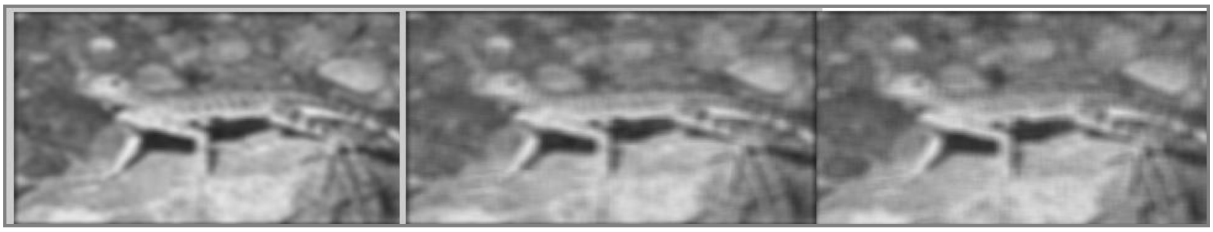
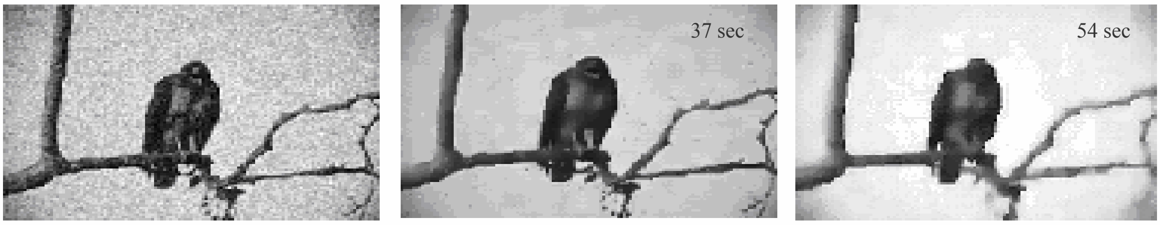


Figure 4.37: Most of the high contrast (edges) information is lost in the linear image diffusions.

Any keen readers are briefly introduced here to an active research area related to anisotropic diffusion. The pioneering research work in this regard was carried out by Perona-Malik [241] endeavouring non-linear diffusivity into digital images, the intensity of gradients and gradient vector field are applied as conduction coefficients (in their approach) to retain edges. Rudin-Fatemi applied calculus of variations to reduce noisy variations from digital images whilst preserving image information/fidelity [242]. Mumford-Shah performed contour based tracking and non-linear denoising within a single operation [243]. The processing timing and frame rates achieved (using anisotropic diffusion) in our experiments (Figure 4.38) are found to be inadequate for real time convergences. However, the extraordinary feature of preserving edges could be pivotal for segmentation purposes in computer vision (evident in Figure 4.38).



.Figure 4.38: Result of applying Perona-Malik diffusion using two unique conduction coefficients.

4.5.3 Pattern Matching using partial function evaluations.

A significant research is needed to investigate the advantages of sequentially populating observational densities in computer vision routines. In RSO, sparsely distributed measurements are systematically integrated (along time) to conduct denser searches at later stages. Practical vision systems encounter a number of bottlenecks when comprehensive discrete density estimations (described in Sections 2.5.1 and 2.5.4) are implemented in an iterative computer program. The fundamental aim of particle hierarchy (in this thesis) is therefore to highlight the advantage of collating measurements of sparse agents (similar to murmuring starlings in Figure 3.1), and to prove their superiority (in all major aspects) over single solution based video tracking.

Section 2.5.4 presented relative drawbacks of applying such costly measurement procedures (right from start) in a typical dominant mode seeking tracking stage. In MS, a weighted discrete density and a candidate model (using kernel estimations) is matched against stored patterns. Generally, Bhattacharyya measure (dot product of two density vectors) is implemented to discover a missing link by applying Equations (2.13)-(2.19). The search processes in both Kalman filter and MS are in fact brute force searches (conducted within an operational basin) which hinder timely convergences as mentioned earlier.

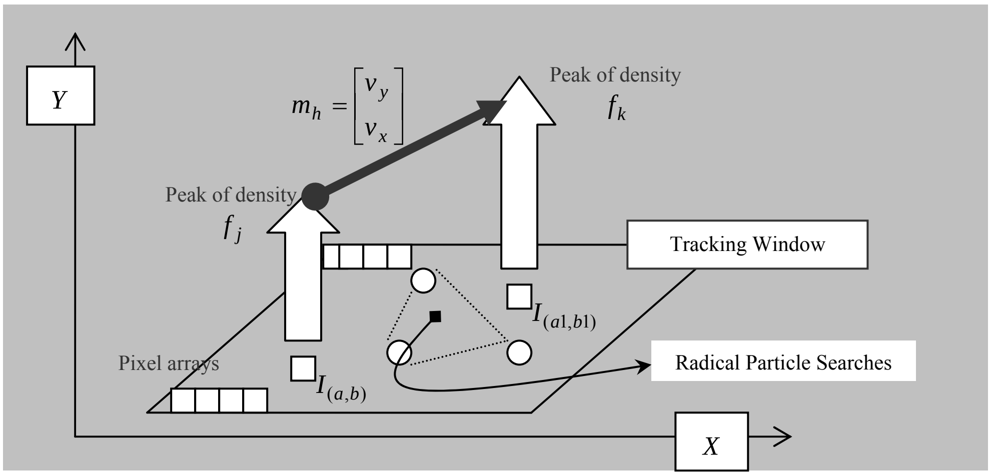


Figure 4.39: Pictorial comparison of brute force searches and agent cored pattern matching.

Figure 4.39 shows how minute and tiny/partial bits of information (gathered by hopping particles hierarchy) could be iteratively employed to predict overall spatial density transitions (shown using block arrows). The fundamental objective is therefore to allocate optimal magnitudes to the velocity components $\begin{bmatrix} V_y \\ V_x \end{bmatrix}$ that also expedites window translations between frames f_j and f_k. At steady state, a tracking window aligns itself perfectly with the newly discovered peak at spatial location $I_{(a1,b1)}$, where $'X'$ and $'Y'$ are image resolutions in Figure 4.39. The costs associated with sequential memory access and calculating Bhattacharyya dissimilarity are therefore significantly reduced in cases where virtual foraging behaviours are implemented as exemplified in Figure 4.40.

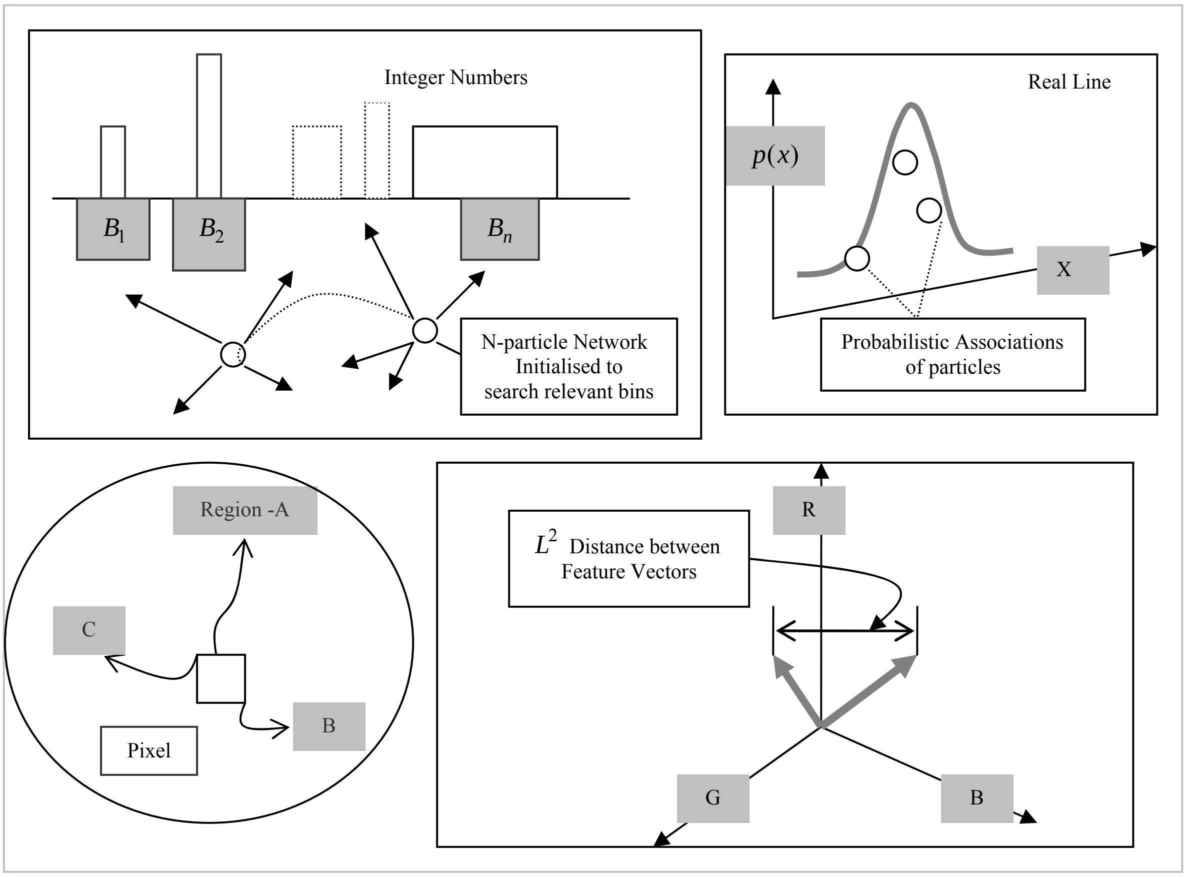

Integer Numbers
B_1
B_2
B_n
N-particle Network Initialised to search relevant bins
Real Line
$p(x)$
X
Probabilistic Associations of particles
Region -A
C
B
Pixel
L^2 Distance between Feature Vectors
R
G
B

Figure 4.40: Tree prominent partial function evaluations in colour feature matching.

$$\hat{q_u} = C\sum_{i=1}^{n} k(\| x_i \|^2)\delta[b(x_i^*) - u] \qquad 4.4$$

$$d_{(j,k)} = \sqrt{\sum_{i=1}^{n}(j_i - k_i)^2} \qquad 4.5$$

$$p(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp(\frac{-(x-\mu)^2}{2\sigma^2} \qquad 4.6$$

Temporal/sequential segmentation is a feature matching process in which every pixel of a digital image is grouped into their respective regions (foreground or background) as shown at the bottom left of Figure 4.40. Equations (4.4)-(4.6) represent how only partial function evaluations could be utilised to detect a variety of objects in videos. Equation (4.4) is reproduced from Section 2.5.4; here only bin identifications $b(x_i^*)$ are allocated as objectives to exploring particles (shown on the top left of Figure 4.40).

The kernel weights $k(\| x_i \|^2)$ in this approach are stored in the form of look-up tables (which is more cost effective than performing calculations in Equation 4.4). Therefore, denser searches are conducted in the vicinity of particles that have discovered those core bins. Equation (4.5) calculates the Euclidean distance between two feature vectors (either grey levels or RGB colour vectors shown in Figure 4.40). Whereas in Equation (4.6), a Gaussian distribution is used to identify an object (calculated at each particle position), and are used as probabilistic associations of particles with the dynamic objects (top right of Figure 4.40).

In our experiments (Chapter 6) we preferred implementing $b(x_i^*)$ as an objective for foraging particles, as in this case, only integer tags are required (compared to the other two partial evaluation cases represented in Figure 4.40) to detect a specific object of interest in an imaging frame. The particle based pattern matching process starts by randomly allocating objective bins $(b_1, b_2, ..., b_n)$ to a particle population. During the next phase, particles are initialised within regeneration areas (identified in Section 4.4.1), and then hop (using perturbation models described in Section 4.4.4) to match designated integer objectives which is also displayed pictorially in Figure 4.40. The flow diagram of this algorithm is drawn in

Figure 4.41.

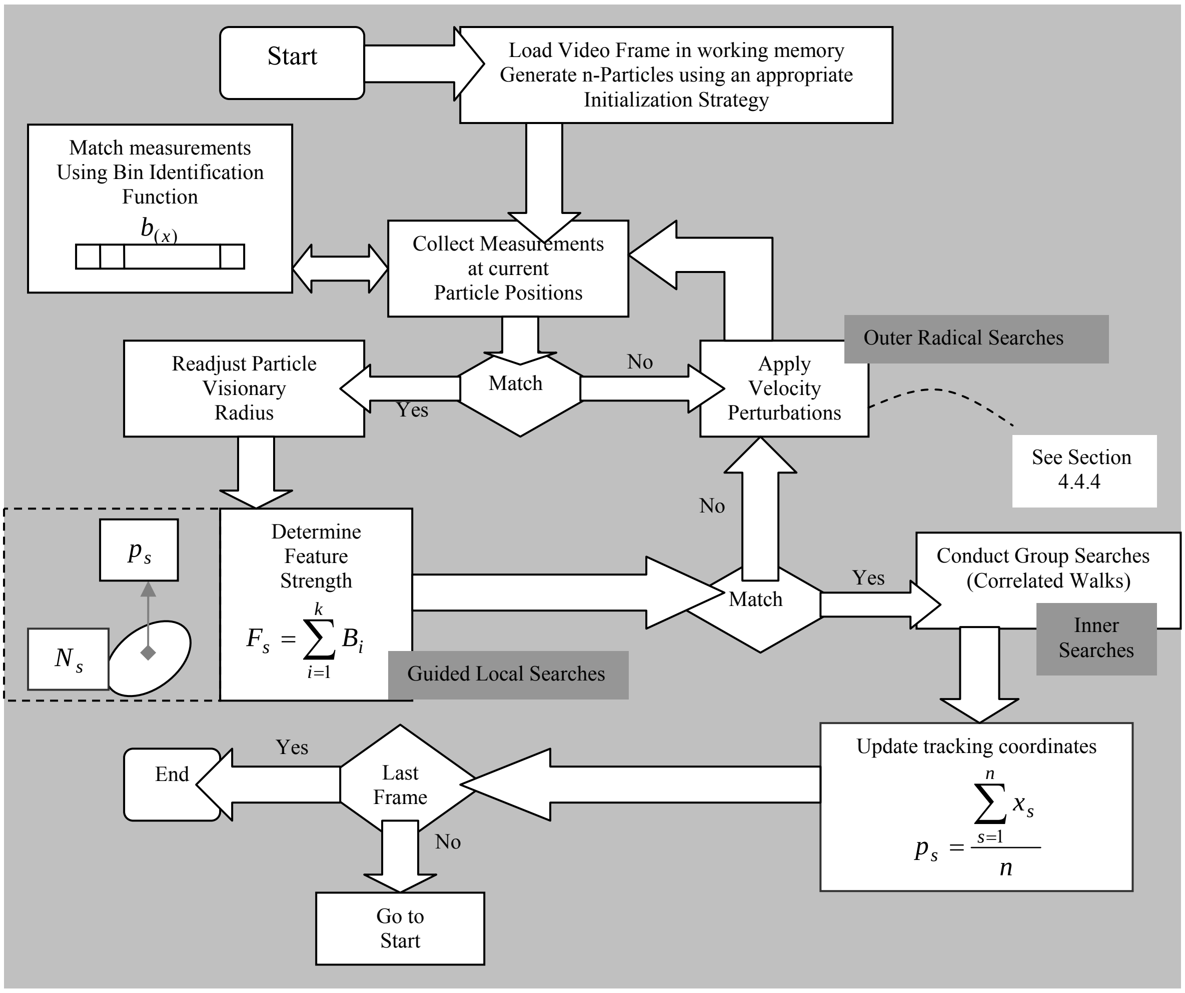


Figure 4.41: A nested tracking algorithm based on bin function evaluations.

A possible layout of this novel tracking algorithm is shown in Figure 4.41 (to portray a complete picture readers are also referred to Figure 4.17 and Algorithm 3.1), and this is the basis of the detection experiments in Chapter 6. The process starts by loading appropriate number of frames into working memory, and once a feasible search space is identified using top order clustering (explained in earlier sections), particles are generated into such regions (for a relevant sampling/initialisation strategy see Figure 3.14). However, if no relevant bins are detected at n-particle positions (p_s^n), suitable velocity perturbations (as explained in Section 4.4.4) are applied to particles until more meaningful patterns are discovered. The algorithm is constituted of two local searches; this includes a Guided local search (shown in Equation 4.1) stage in which feature strength F_s is analysed (within a closed neighbourhood region N_s formulated around a particle positions p_s^n). A suitable choice for colour based features and patterns is $F_s > \frac{B_n}{2}$, hence, if more than half of relevant integer bins (stored as feature Identities) are discovered (in predefined neighbourhoods), the region under question is labelled as a legitimate target (otherwise the hypothesis is rejected and further outer velocity modulations are applied). Finally, the position of tracked object is refined using correlated movements (explained in Section 4.4.4 and Figure 3.44), and the target position is updated using the newly calculated mean positional vector p_s. The algorithm is repeated for all frames as shown in Figure 4.4.1.

4.5.4 Perturbation Models

'Perturbation theory' has been historically applied to a number of situations where finding an exact solution (to a mathematical problem) is next to impossible [244]. The particle dynamics (e.g., applied in PSO and DE) are complementary computational rhetoric (introduced in an analogues context) to address inversion (inverse problems) in a computerised visual system (please refer to Sections 1.1-1.3). A significant portion of this thesis has already been dedicated to the cause and effects of indulging velocity variations, which are predominantly driven by social factors (abundant in natural agents). Sections 3.1-3.4 are structured to devise an alternative that nullifies the overheads in swarm cored analytics.

Later on in Section 3.4, a newly improved perturbation model was introduced (to the readers based on virtual dynamics observed in foraging agents) that has proven more effective during experiments conducted along the course of Section 3.4.5.The concept of 'mortality of particles' introduced in Section 3.3.2, and comparative discussion presented using relevant ecological and geological disturbances (a possible synonym for the word 'perturbation') paved the way to understand how vital balances are kept in natural systems. Some further perturbation models are introduced in this section for more detailed analytical comparisons to be performed in Chapter 6. A case study will be presented (using bee nest site selection/BNSS algorithm [246]) for relating natural and computational worlds.

(a)-Search Strategies in the Virtual Bats

During 2010, Yang proposed a novel meta-heuristic algorithm inspired by the urban living micro bats [245]. Bat algorithm is mainly built around the echo localization behaviour of micro bats. The bat algorithm tries to conduct much wider range of searches compared to the PSO algorithm, with an inherent/key advantage that it has to deal with less strategic control parameters compared to the PSO technique. Echolocation is a type of sonar ranging between 25 kHz to 100 kHz which lasts for only a few milliseconds, during this time, short bursts of pulses are discharged by the bat which could be as high as 20-30 bursts per second.

Bats also have good vision, and rely on their sense of smell to plan their trajectories. Therefore, both auditory and olfactory senses are applied by the bats to effectively scan their environments. In the original bat algorithm, bats deploy echolocation techniques to sense the distances to the food and prey, and use the reflected pulses to differentiate between preys and to plan their paths through the complex terrains. The virtual bats therefore also apply loudness and a variable pulse rate to explore the search space. The velocity and the frequency of pulses at any moment in time are determined by the following Equations (4.7)-(4.9).

$$f_i = f_{\min} + (f_{\max} - f_{\min})\beta \tag{4.7}$$

$$v_i^t = v_i^{t-1} + (x_i^t - x_*) f_i \tag{4.8}$$

$$x_i^t = x_i^{t-1} + v_i^t \tag{4.9}$$

In Equation (4.7), $\beta \in [0,1]$ is a random variable drawn from a uniform distribution. The frequency of pulses is adjusted by a randomization process which assigns values by taking into consideration the maximum and the minimum allowable frequencies in a specific search space (Equation 4.7). The velocities of the virtual bats are calculated through Equation (4.8). Finally the newly calculated velocities are used to update bat positions in Equation (4.9). In order to get a right balance between the exploration and the exploitation phases, the loudness and pulse rates are varied in accordance with Equations (4.10) and (4.11). All bats explore the search space using variable loudness values which could range between $A_{\min}$ - $A_{\max}$ as shown in Equation (4.10).

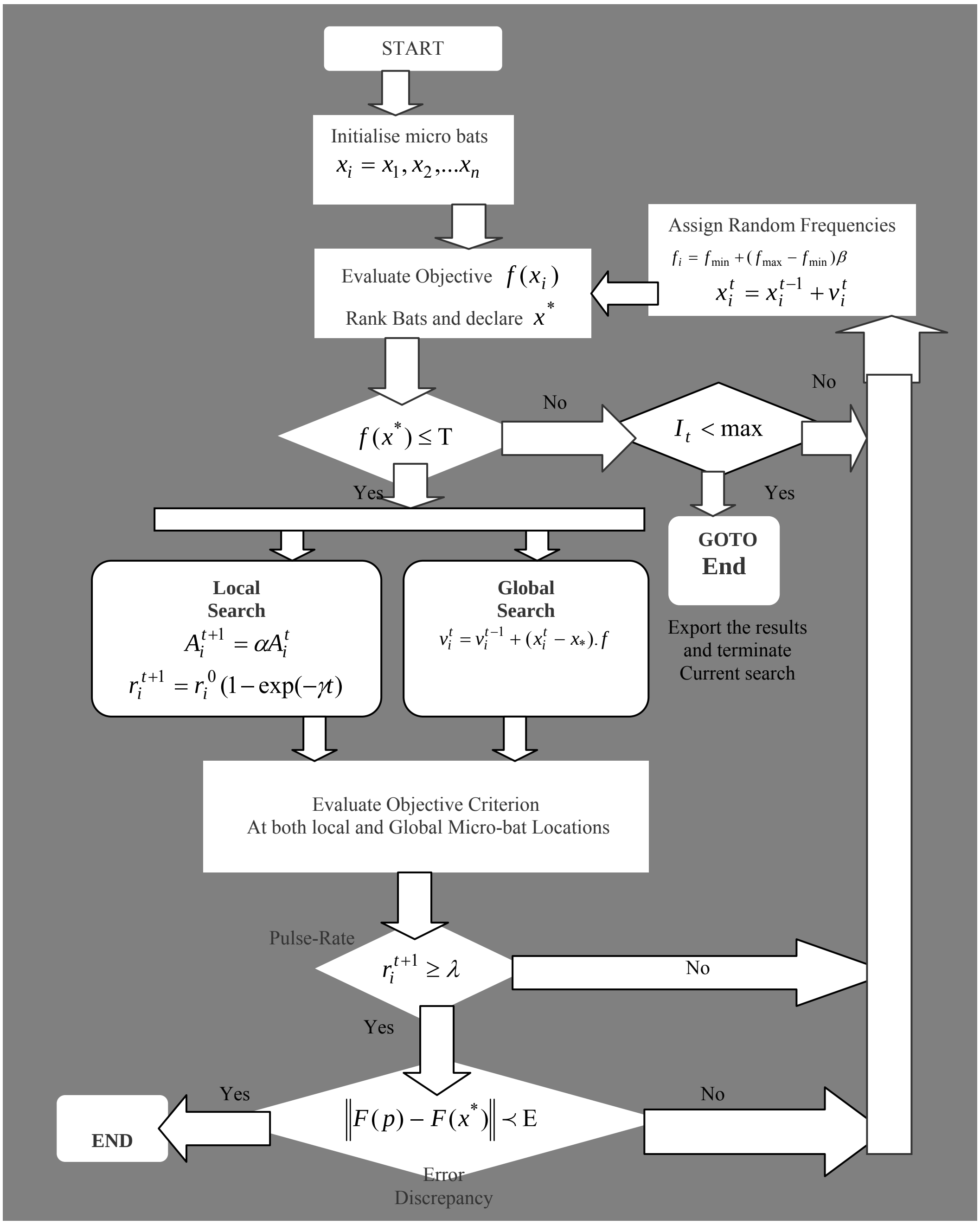
$$A_i^{t+1} = \alpha A_i^t \text{ And } \alpha \in [0,1] \tag{4.10}$$

$$r_i^{t+1} = r_i^0 (1 - \exp(-\gamma t)) \qquad \gamma > 1 \tag{4.11}$$

When micro bats get closer to their assigned objectives (e.g., bin identities or other pattern matching objectives in Figure 4.40), they vary rates of emitted pulses in order to perform localized searches (using exponential function in Equation 4.11). In contrast, the loudness decreases towards zero at a near optimal point (where a bat gets almost stationary). The loudness and pulse rates are indulged for local searches, whereas, the variable frequency assignments are embedded meta-heuristics which broaden searches, and provides opportunity for a bat population to be attracted towards promising regions (within space boundaries).

The flowchart of micro bats based optimisation algorithm is shown in Figure 4.42. The algorithm starts by implanting n-micro bats in the search area. Next, the corresponding objective function values are evaluated at each bat location, moreover, solutions are ranked in ascending/descending orders to nominate best matches. If the function value $f(x)$ at a proposed global optimal position is below (or is equal to) a pre set threshold value (i.e., $f(x^*) \leq \mathrm{T}$), then, 2 parallel search phases are initiated (as shown in Figure 4.42).

The aim of these two simultaneous processes is to further refine the newly projected global optimal solution $f(x^*)$ using Equations (4.10)-(4.11), and at the same time, global searches continue searching for alternative solutions using Equations (4.7)-(4.9). Once a bat accomplishes its designated task (or gets nearer to achieve its allocated task), the pulse rate is incremented, and bat search frequencies are more smoothly varied (in a linear fashion, similar to the simulated annealing stages described in Section 3.2.2) which enables all bats to search within local proximities (and restricts them jumping beyond the current search space). As the pulse rate is the confidence measure of a particular bat (that it has find its objective), therefore, in searches where $r_i^t < \lambda$, the local search is terminated and the algorithm enters into an initial global search phase. Finally, the process terminates if the error discrepancy (between objective and calculated values) is $\|F(p) - F(x^*)\| \prec \mathrm{E}$.



START
Initialise micro bats
$x_i = x_1, x_2, ...x_n$
Assign Random Frequencies
$f_i = f_{\min} + (f_{\max} - f_{\min})\beta$
$x_i^t = x_i^{t-1} + v_i^t$
Evaluate Objective $f(x_i)$
Rank Bats and declare x^*
No
No
$f(x^*) \leq \mathrm{T}$
$I_t < \max$
Yes
Yes
GOTO
End
Export the results
and terminate
Current search
Local
Search
$A_i^{t+1} = \alpha A_i^t$
$r_i^{t+1} = r_i^0(1 - \exp(-\gamma t)$
Global
Search
$v_i^t = v_i^{t-1} + (x_i^t - x_*).f$
Evaluate Objective Criterion
At both local and Global Micro-bat Locations
Pulse-Rate
$r_i^{t+1} \geq \lambda$
No
Yes
Yes
No
END
$\|F(p) - F(x^*)\| \prec \mathrm{E}$
Error
Discrepancy

Figure 4.42: Flow chart of the original micro bat algorithm applied in Chapter 6.

(b)-Correlated Walks

Correlated walks (CW) are one further type of particle perturbations (beside PSO, BAT and RSO and guided searches) that will be applied to track and detect object boundaries in Chapter 6. The correlated walks could prove particularly useful in parallel hypothesis pruning in the combinatorial optimisation technique described in Figure 4.19. Once the top order clustering (mentioned in Section 4.4.1) has been performed, the particles are regenerated around the vicinities of those clusters, and both inwards and outwards correlated walks (depicted through $\pm$ in Equation 4.12) are performed to detect object boundaries, where P_g is the current choice of the global optimal, and x_i^t is the position of a particle at time instant t.

$$x_i^t = g(x_i^{t-1} - P_g) \pm x_i^{t-1} \qquad 4.12$$

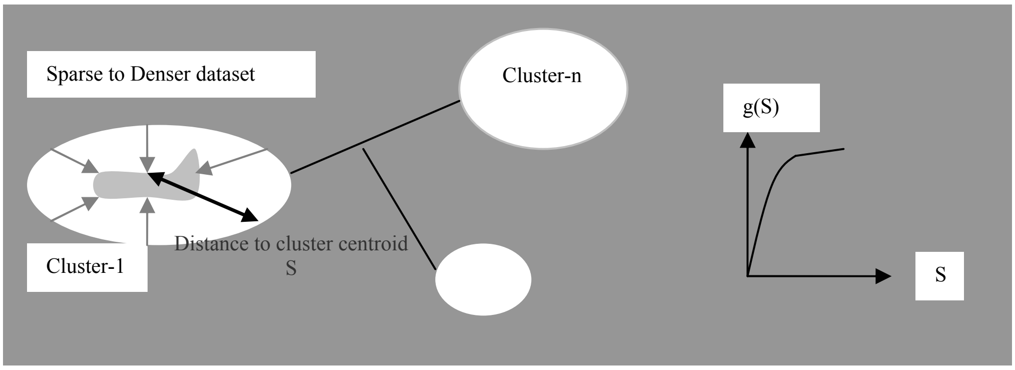


Figure 4.43: New correlated movements in particles for combinatorial detections and tracking.

The velocity of a particle (Figure 4.43) at time x_i^t is calculated by the distance modulation scheme (also proposed earlier in Algorithm 4.1), in this approach the particle velocities $g(S)$ (at an instant of time) are non-linear function of their respective distances $S = (x_i^{t-1} - P_g)$ from a prospective global optimal (as shown in Equation 4.12, and through the plot of the function $G(s)$ in Figure 4.43). Therefore, as particles travel towards the centroid of the cluster, there velocity are dampened and become stationary at the boundaries. Finally, the tracker window is updated by finding the mean position of the particles using Equation (4.13). All particles use bin functions $B(X)$ as colour templates to discriminate between object and background regions (see Figure 4.40).

(c)-Case Study using Bee Nest Site Selection (BNSS)

In BNSS [246], only a smaller proportion of population is activated at any time in contrast to the PSO. The search strategies of the forager bees could be versatile and a range of local and global searches (e.g., isotropic and correlated Gaussians, signal modulated Levy or pure randomisation) could be employed in this context as shown in Figure 4.44. All scouts keep looking for the nest site (four correlated walks from swarm position p_{SWARM} are shown in the top left corner) until they come across a potential nest site (object). At this stage, scouts recruit a number of followers to investigate further (e.g., by deploying CW in Figure 4.43).

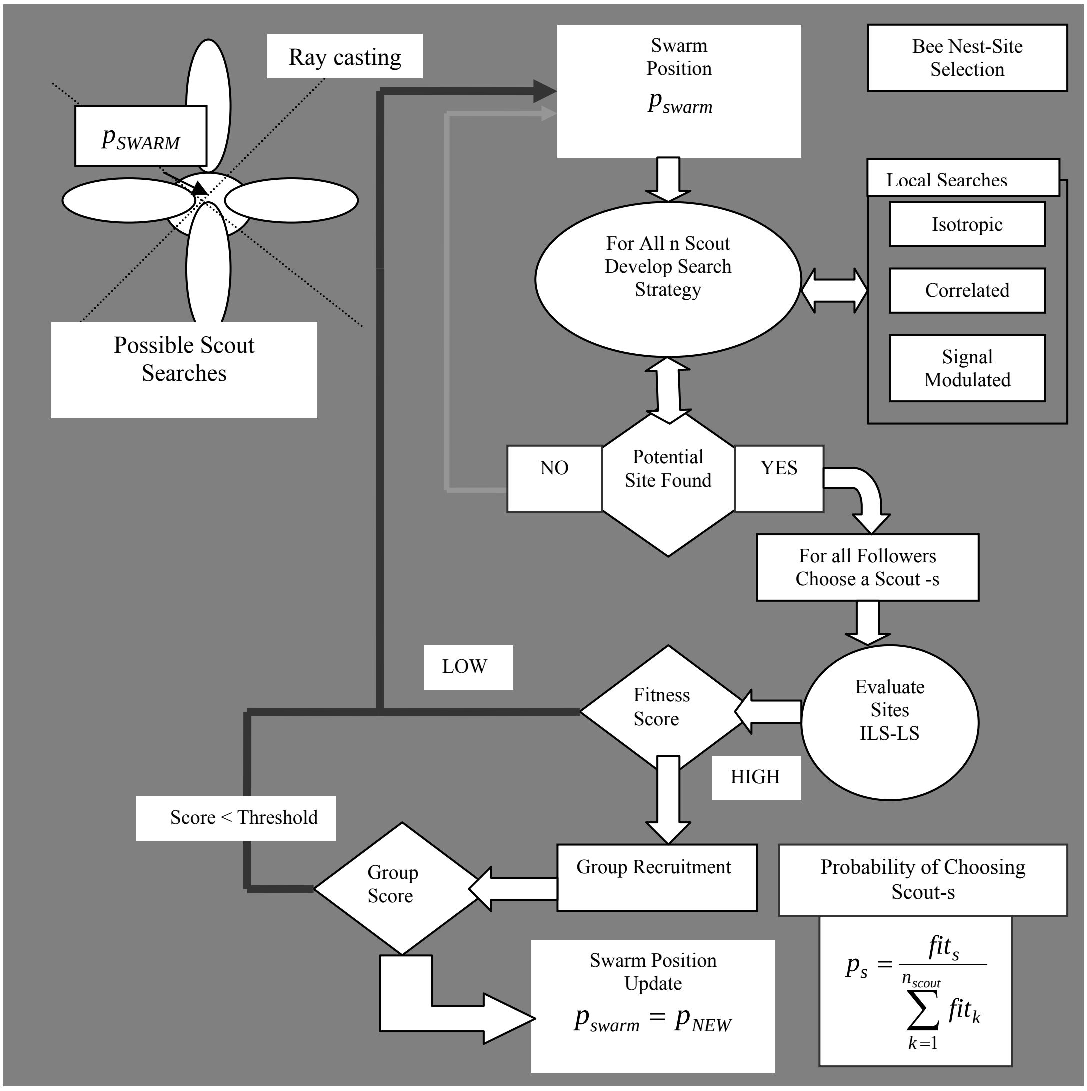


Figure 4.44: Tracking algorithm motivated by the bee nest site selection (BNSS).

The ray castings are however tighter and precision line searches that could also be used in this regard. The fitness scores of scouts are ranked according to $p_s = \frac{fit_s}{\sum_{k=1}^{n_{scout}} fit_k}$, and more followers are recruited if higher fitness score is achieved by a particular scout. In the standard BNSS algorithm, two additional guided stages are also performed during the optimisation process, here successively increasing populations and group fitness are evaluated (e.g., by controlling follower population) to establish if a potential site meets all the necessary conditions (e.g., object size and range of colour distributions in tracking-similar to the natural hives). The systematic evaluation at both individual and group levels is therefore used to select nests in BNSS. Therefore BNSS is a more advanced approach that reduces the associated computational costs and is an excellent example of a dynamical controlled environment. The RSO very much resembles to the BNSS due to the inbuilt guided search stage. The main fitness score in RSO is however based on the observations of a variety of colour bins within a candidate region (Figure 4.41), and the relevant multivariate colour distributions (Figure 4.40) are used as a pattern matching process. Finally the swarm position (or tracking window in an image) is updated upon a successful match.

4.6 Conclusions

The purpose of this chapter is to introduce to the readers the key characteristics of a dynamic optimization environment. The dynamic optimization rules are applied to expedite a tracking stage by conducting more diversified searches in the feasible region.

Chapter 5

Test cases-image sequences

This chapter introduces several challenging detection and tracking scenarios that would be later applied in Chapter 6; the aim is to select a wide variety of scenes so that the response of proposed methods could be analysed. Several plots are presented in this Chapter that defines each problem in the search space. The spatial search space in tracking and detection algorithms is the corresponding image dimension in which objects undergo translations (Figures 2.1-2.5), whereby, a feature space is used in matching the unique patterns of an object of interest (e.g., using colour, texture and shape based features).

It is established through a variety of experiments that formulating an efficacious communication between social expatriates (similar to foraging hierarchy in bee hives), and a converging population (e.g., PSO) could address the common challenges in the deterministic systems [247]. The effectuated diversity introduced by the scale free correlations could culminate accurate movements in heterogeneous applications (hence results in increased portability mentioned in Section 1.1). Figure 5.1 shows a range of visual challenges in this context where various detection and tracking experiments would be conducted in Chapter 6.

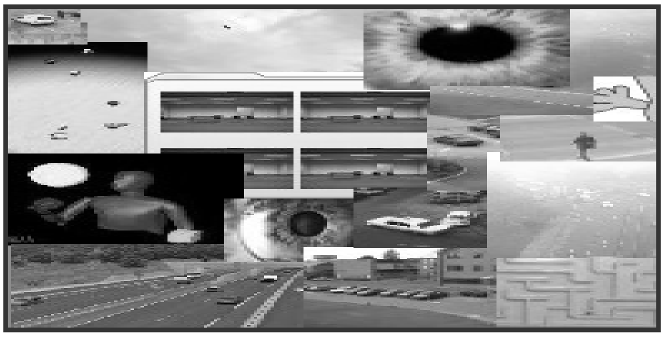


Figure 5.1: Tracking scenarios to be applied in our experiments (taken from vision repository [248]).

5.1 Frame Rates in video tracking applications

The frame rate of a capturing device is an important characteristic in the tracking applications. When high speed cameras are deployed to capture the underline motions, the precise and minute movement details are perfectly preserved onto a capturing plate/film. The motions appear quite smooth and jitter free (to a human observer) if visual information is sampled and presented at 25+ FPS. However, the tracking rate achieved in a tracking algorithm is an entirely different aspect, and this scenario is presented in Figure 5.2 to clarify. There could be f_n number of frames that have been used to sample the motion of a dynamic object during one second of time interval. However, the tracking frame rate explicitly depicts the processing speed (in FPS) at which motion trajectories has been derived from a stored video sequence. Hence, if the tracking is observed to be effectively working at $250FPS$, then it means that such number of frames are being processed at a tracking stage, and could be utilised in situations with the respective camera speeds (e.g., in a live monitoring). Generally, the underline displacements are described in terms of the pixel space (the top left pixel of a digital image is $\begin{bmatrix}0\\0\end{bmatrix}$, and the bottom right is a corresponding pixel at $\begin{bmatrix}y_n\\x_n\end{bmatrix}$ spatial location (where n is the resolution of a square image).

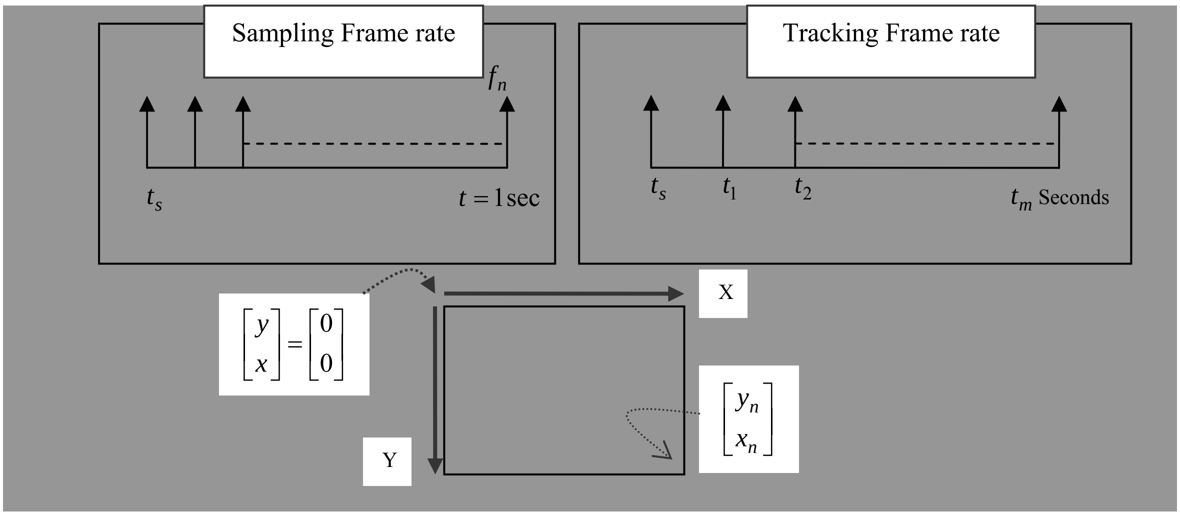

Sampling Frame rate
f_n
t_s
$t = 1\sec$
Tracking Frame rate
t_s
t_1
t_2
t_m Seconds
X
$\begin{bmatrix}y\\x\end{bmatrix} = \begin{bmatrix}0\\0\end{bmatrix}$
$\begin{bmatrix}y_n\\x_n\end{bmatrix}$
Y

Figure 5.2: Pictorial difference between the sampling and tracking frame rates.

5.2 Test Benches

Several computer vision test problems will be specified in this section. The displacements patterns in the form of generalized motion trends are presented along with the corresponding quantitative characteristics in both spatial (ground truth of motions) and feature space (histogram ball plots) formats. Several related factors including the sampling frame rates, average and maximum displacements (in Euclidean pixel distances) are highlighted for each test scenario. Therefore, the selected patterns exhibit a great deal of versatility and diversified styles of movements (to be further tested in Chapter 6).

5.2.1 Ant Sequence

In this sequence an ant randomly traverses the maze, and occasionally slips and falls off the maze walls, and generally alters its direction of travel in a non-deterministic fashion. The shadows from the boundary walls and the illumination changes further exacerbate the volatility of this scene. The models stored in the memory therefore frequently become invalidated and (during adverse stages) could not be applied to differentiate between the object and the background clutter. Hence, an alternative fusion methodology is needed in order to regain control of the window. An ample tracking system could be built using the nested heuristic searches (Sections 3.3, 4.2), and by treating each potential region as parallel hypothesis with only the top order solution being accepted as an object of interest.

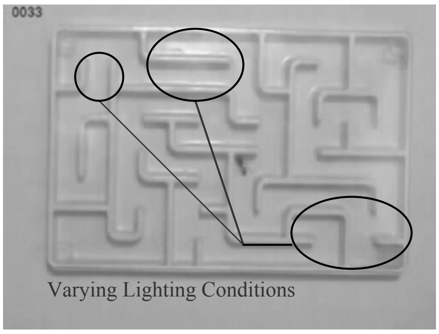


Figure 5.3: Tracking a randomly moving insect under lab condition (taken from repository [249]).

Figure 5.4 shows the surface plot of a particular tracking frame (shown in Figure 5.3), the landscape of this sequence is highly rippled, and for comparative analysis, the terrain generated by the Egg-crate function is also plotted [250]. Due to the similarity of the landscapes, the previously implied test processes (Section 3.5) are generally applicable in the tracking domain as well. The ground truth (movement coordinates) of the ant movement history has been shown in the Figure 5.5 along with its feature plot in a RGB colour space. The respective histogram bins in the feature domain (799, 1073, and 1421 etc and marked with the arrows), and the distracter features have been identified as well. In contrast to the density related comparisons in the mode seeking algorithms (Section 2.5.3), particles will be assigned shorter binary objectives (to detect dominant modes using the respective bins), the frequency of the respective bins is indicated by the radius of the balls. All histogram ball plot diagrams presented in Chapter 5 are interpreted in a similar manner.

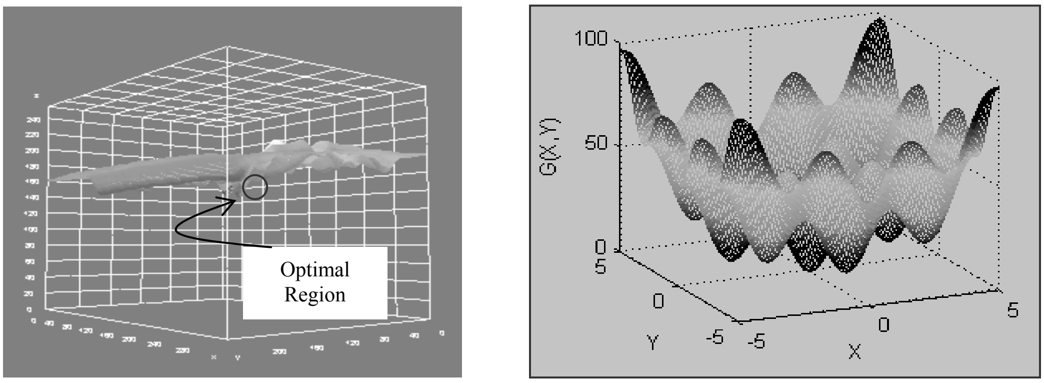


Figure 5.4: The similarity between an image tracking and a 3D evolutionary test case.

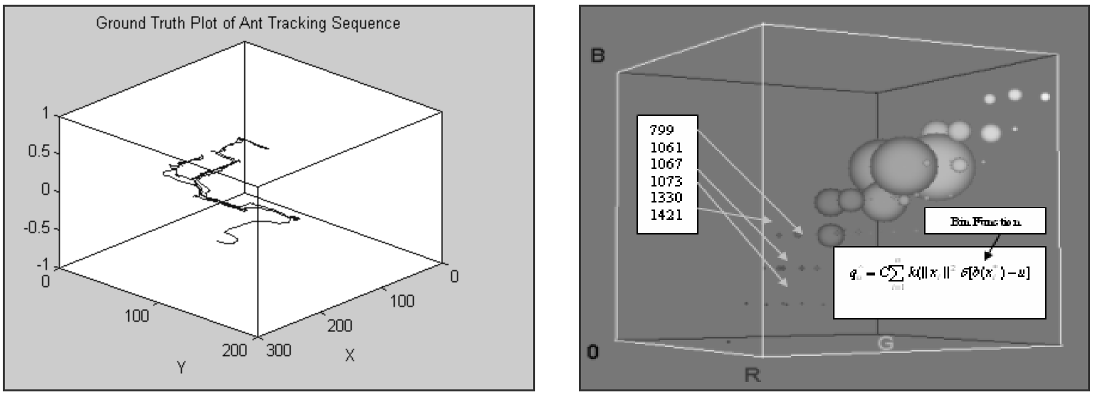


Figure 5.5: Plot of the ground truth in the ant test case (sequence consists of 774 image frames).

The quantitative characteristics of the ant sequence are presented in Table 5.1. The sequence is composed of 773 images captured at a frame rate (camera speed) of 25.8 FPS. The total displacement in Table 5.1 refers to the pixel distance this object has traversed within its spatial space during 30 seconds. The displacement refers to the Euclidean distance it has traversed during its trajectory, whereas the maximum displacement observed between any two frames is around 12 pixels. The diversified and random style of movement is evident in the Matlab plot drawn in Figure 5.6; the plotted vectors show both magnitude and direction of the movements between all video frames (in an image space XY). Tables 5.1-5.4 could also be interpreted as described in the above paragraph.

Table 5.1

Sequence Length	773 frames
Sampling Frame Rate	25.8 FPS
Sequence Time	30 Seconds
Total Displacement (Pixels)	1.5057e+003 Pixels
Maximum Displacement	12.53 Pixels

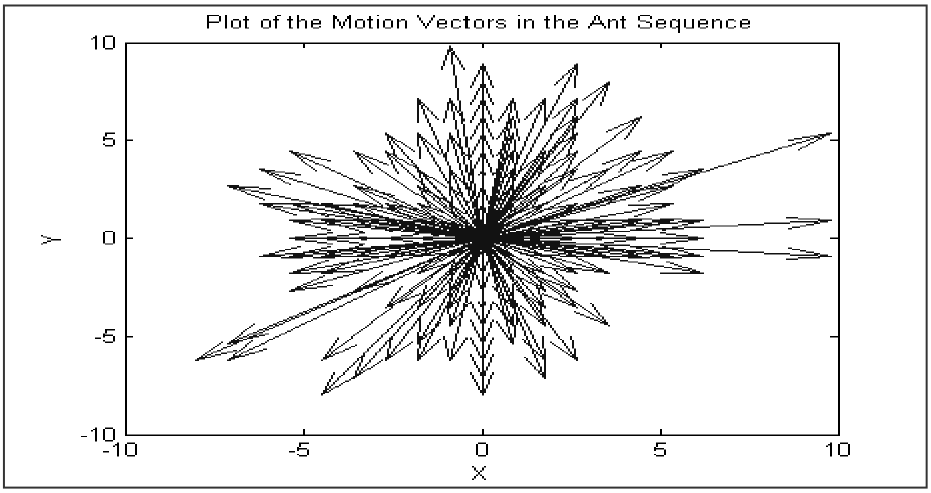


Figure 5.6: The vector plot of motions observed between frames in the ant sequence.

5.2.2 PETS Pedestrian Sequence-1.

This pedestrian test scenario has been adapted from the PETS video repository [251], and is a standard problem to detect the tracking efficiency, and its speed of convergence. The profiles of the tracked people are manually formulated using the discrete density histogram models [252]. The Bhattacharyya similarity measure will be used to match the estimated density to the ones stored in the memory. The movement trajectory of the pedestrian-3 has been plotted on the right in Figure 5.7, which shows a near-linear translational during the entire 195 video frames. This clip constitutes of 195 frame captured at a Frame rate of 24FPS (Table 5.2).

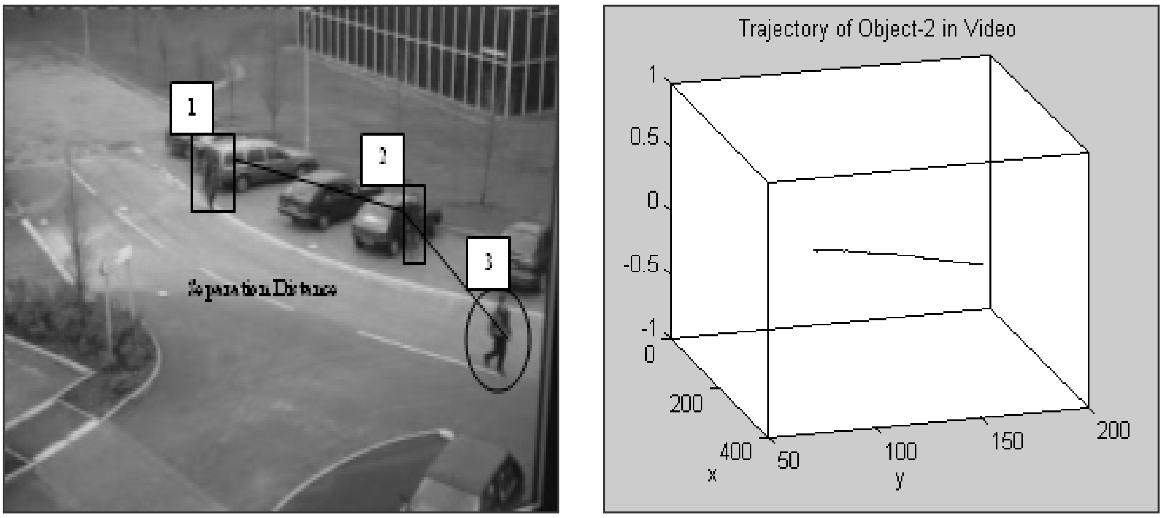


Figure 5.7: Tracking of pedestrians and the corresponding movement trajectory (pedestrian-3)

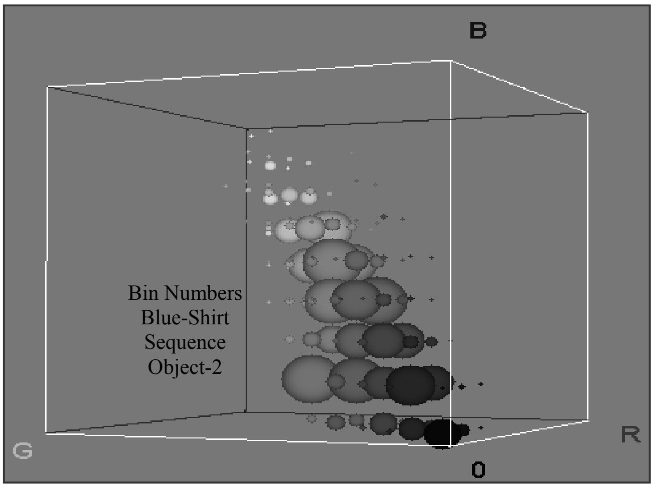


Figure 5.8: The histogram ball plot of the colour profile of the pedestrian-3.

The histogram ball diagram for the pedestrian-3 has been plotted in the Figure 5.8. The numerical data in the plot stipulates a variety of histograms components. Therefore, the aim is to match these numerical values with the ones detected during the online tracking stage. The 3d plot (Figure 5.9) shows the general data trend in profiling the pedestrian in normalised RGB-space. Larger variance has been observed along the vertical dimension and therefore the changes are accordingly reflected into the objective function.

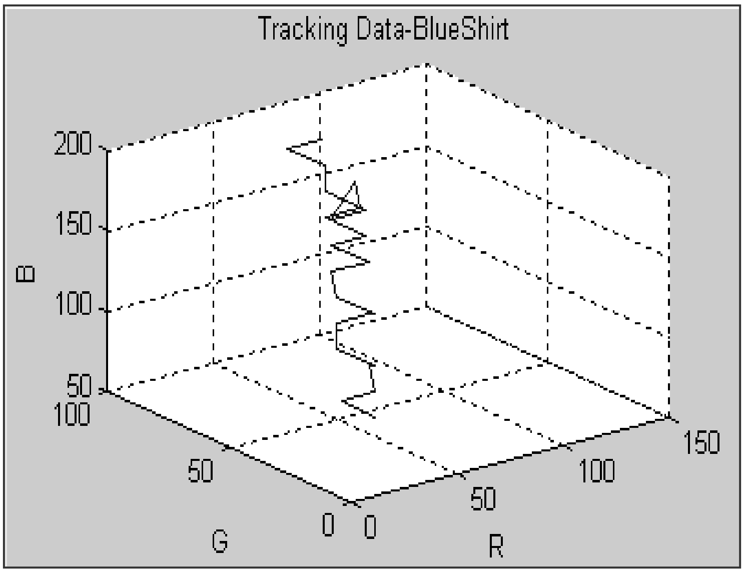


Figure 5.9: Plot of the colour characteristics of object 3 in Figure 5.5.

Table 5.2

Sequence Length	195 Frames
Sampling Frame Rate	24 FPS
Sequence Time	8 Seconds
Total Displacement (Pixels)	587.89 Pixels
Maximum Displacement	8.361 Pixels

5.2.3 Highway Sequences.

Some highway videos have also been selected as test benches, and four particular images from this sequence are shown in Figure 5.10 [253]. In this sequence a variety of motor vehicles are tracked, and the trajectory of a green van has been identified with the arrows. The histogram ball plot (where the radius of a ball show the frequency of observations) of the object has also been presented in Figure 5.11. The movement of this motor vehicle during 145 frames are plotted in Figure 5.12, and the overall motion appears to be of a curvilinear in nature compared to the pedestrian sequence, and the tracking data in a quantitative format is presented in Table 5.3. This particular sequence is captured with a high speed camera, therefore, a smaller (maximum) movement of 1.89 is observed between frames.

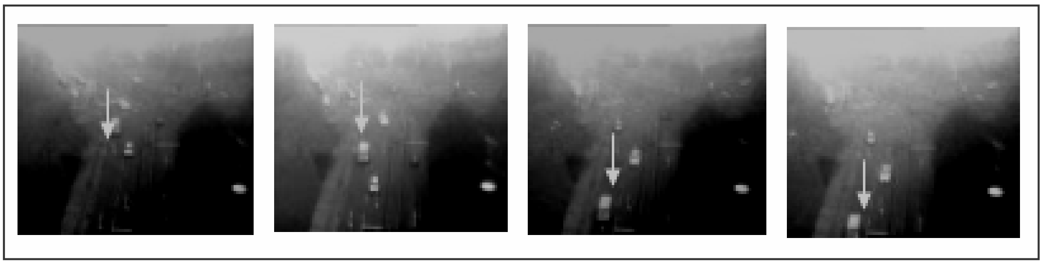


Figure 5.10: Four frames of the highway tracking sequence.

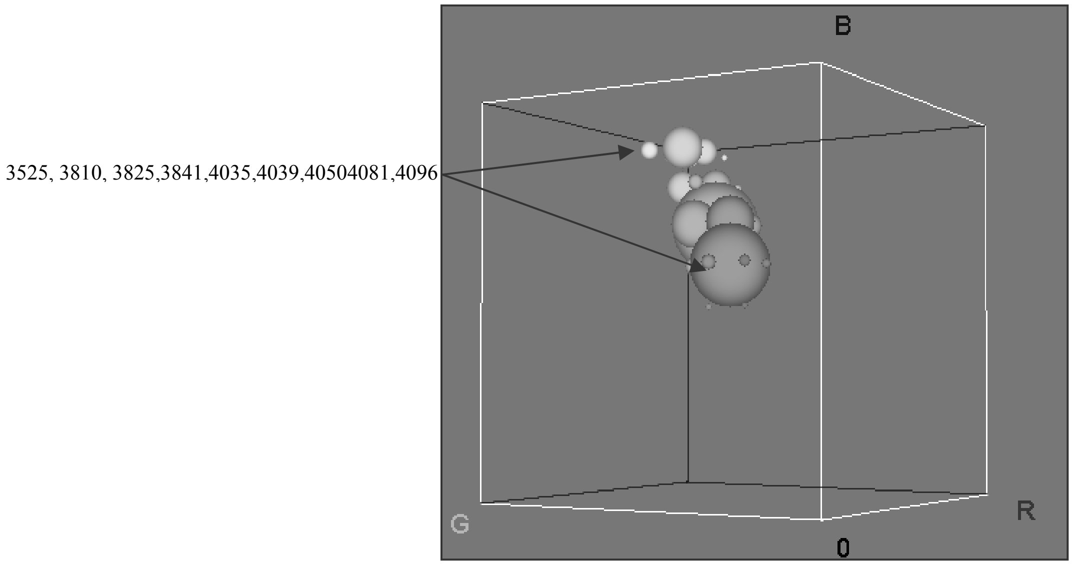


Figure 5.11: The histogram ball diagram of a green vehicle in the RGB colour space.

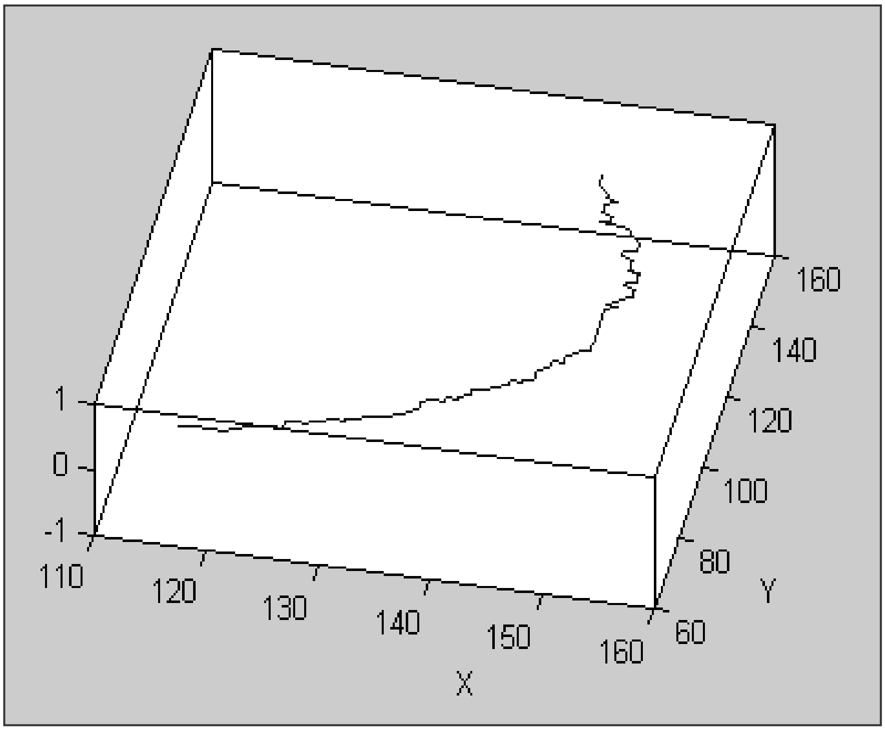


Figure 5.12: The curvilinear movement and path of the object of interest.

Table 5.3

Sequence Length	145 Frames
Sampling Frame Rate	36.25 FPS
Sequence Time	4 Seconds
Total Displacement (Pixels)	0.545e+003 Pixels
Maximum Displacement	1.89 Pixels

5.2.4 Quads-Copter/Drone Sequence.

This video sequence was created at a local park in Edgbaston, Birmingham. The underlying aim is to investigate the applicability of the RSO in objects subjected to the scale, translational and rotational movements. This platform is an idealistic test bench to determine the penalising strength of the guided local searches (Figure 4.17). Due to being an airborne vehicle, the quad-copter frequently blends with the clutter, and therefore is a challenging tracking problem. As the drone drifts away from the dynamic camera, it undergoes sudden changes in its scale, and at its furthest distance, it dramatically reduces to only a few imaging pixels. The poor visibility and outpouring conditions demand an update in the feature space to increase the tracker stability and its robustness.

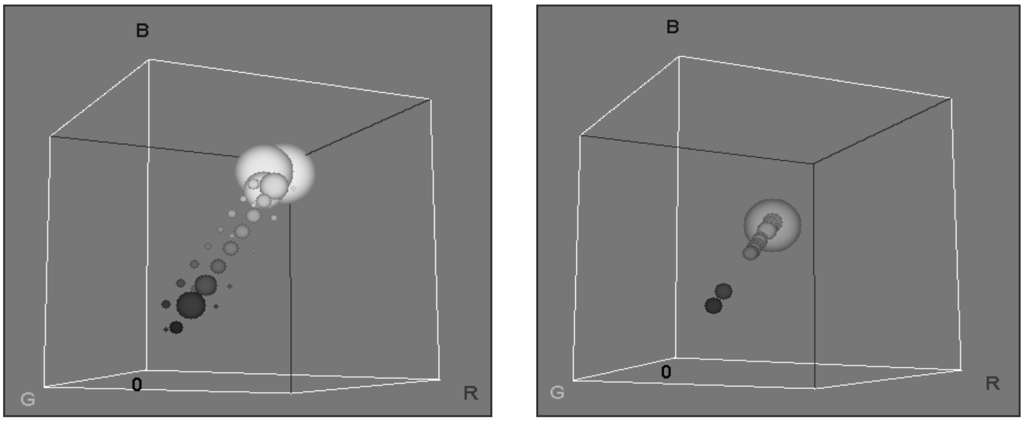


Figure 5.13: The histogram ball plot of an airborne vehicle shown in the Figure 5.14

Figure 5.13 shows the changes in the object histogram model. The deviations in the circumferences of the balls indicate the scale of the observed discrepancies between two video frames. Along with the changes in the frequency of measurements, a significant reduction in the histogram components is observed within a very short time period. In cases where larger model invalidation is observed, it is often more feasible to narrow down the search space using the data fusion techniques (e.g., using motion and feature detection sub-modules). A number of frames from the drone tracking problem (the tracking test case is constituted of 879 image frames) are presented in Figure 5.14, the airborne vehicle often merges with the low and fast moving cloud, and therefore tracking window is dynamically altered in order to diversify the measurements. The footage was captured on a Nokia mobile phone with a maximum sampling rate of 7FPS. Therefore, this particular sequence is especially challenging due to the jittering platform (held in hand) and low sampling frequency and higher displacements between all frames as expressed in Table 5.4.

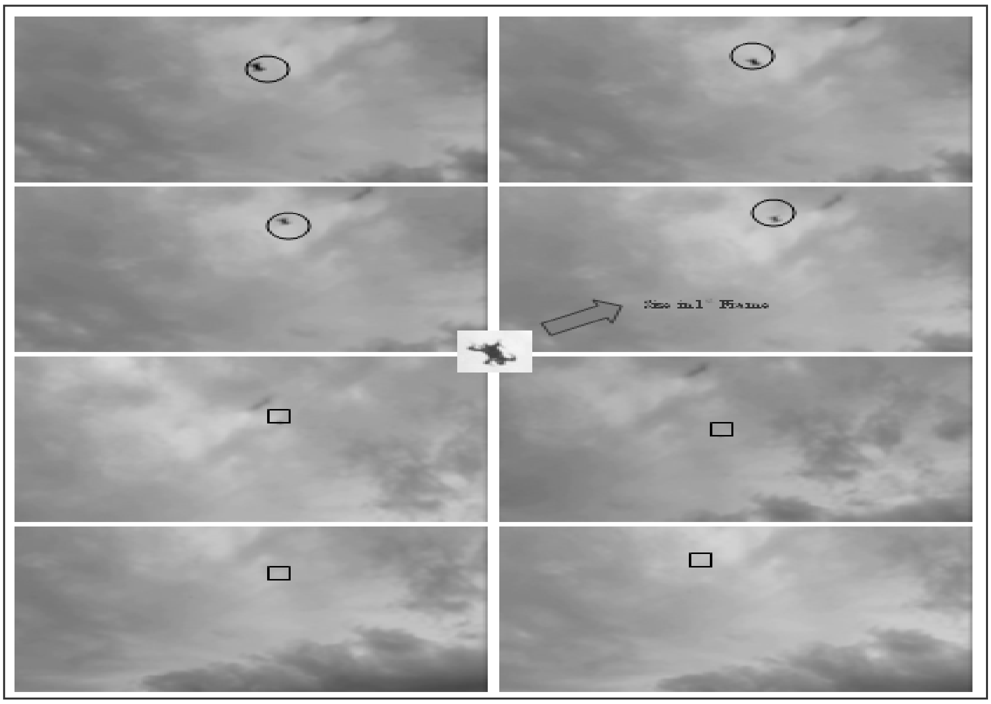


Figure 5.14: The scale changes in eight video frames taken from the drone sequence.

Table 5.4

Sequence Length	879 Frames
Sampling Frame Rate	7 FPS
Sequence Time	125 Seconds
Total Displacement (Pixels)	9.879e+003 Pixels
Maximum Displacement	35.8 Pixels

5.3 Conclusions.

This chapter was written with an aim to introduce readers to some most common and challenging test cases utilised in the computer vision community. Several difficult video tracking and detection problems have been identified in this chapter. The ground truth values are plotted along each test case, and would be used in the next chapter to determine the accuracies of our detection/tracking algorithms.

Chapter 6

Natural Detection and Tracking

It has been a fervent desire of human beings to learn from the nature's artistic designs [254]. We apply our biological vision to accomplish almost every task that frequently incorporates visual tracking of bodies in order to enhance our anticipation of the environment. The salient features are derived and availed to promptly recognize a dynamical object in our vision [255]. A partial inspiration in this report came from our visual references and consequently the realization of a denser semantic knowledge to handle complex situations. Solving the visual jigsaw in machine vision is more elaborate and decades of scientific research is needed to gain the proficiencies of the natural vision.

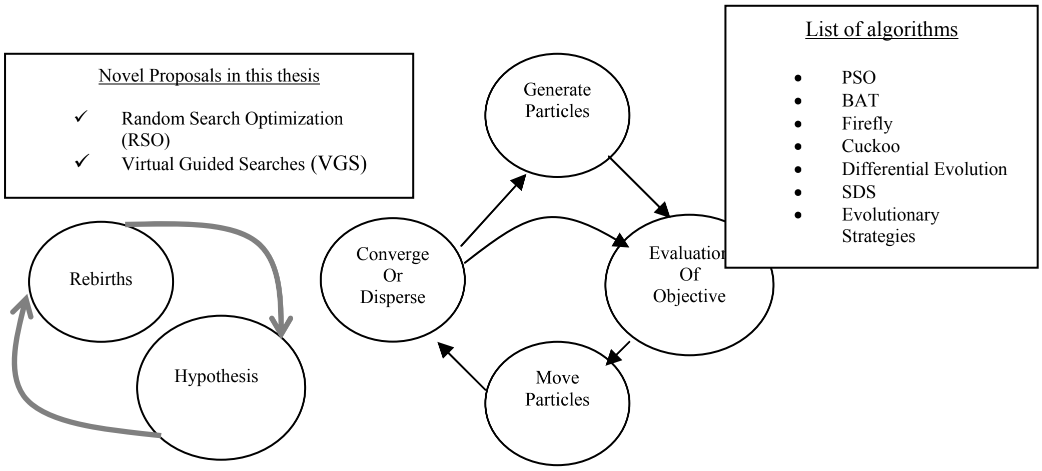


Figure 6.1: Strategic differences in general optimisation and methodologies applied in this thesis.

Some inferences about a biological vision could be deduced from self analysis. The association of the computer vision systems and our visionary perspectives could be a key to implement the fundamental aspects effectuating our visual prolepsis during highly cluttered scenes. One of the major components of a biological vision is the presence of a systematic hierarchy (similar to the RSO we applied to solve test problems) in which sparse observations are generally imputed to perpetuate lower and denser solutions. It is not much deviation from the reality to write that it seemingly appears that we apply dominant scene characteristics (in our vision) to detect a top order movement. After further visual refinements, a human brain conjectures a belief and reaches to a specific conclusion regarding the visual depth of an object.

We would like to conclude the main theoretical expositions presented in this thesis before conducting further experimentation. Some complicated evolutionary test cases were successfully solved in Chapter 3 by applying RSO and stochastic variations (Section 3.4). The novel system (RSO/VGS) stipulated the preferences of space based convergences (Section 3.1-3.3) over the traditional calls in the social particles. A major source of the misconceptions in the evolutionary algorithms (RHS of Figure 6.1) is due to wrongfully considering particles as immortal agents. Furthermore, storing intermediate states of particles, and mathematically calculating the complex social interactions are analysed as redundant features in this thesis.

Therefore, a search driven optimisation based on nested particle behaviours is a novel technique which first integrates observations of the virtual swarms (Section 3.4), and later applies guided searches (Section 4.3) to solve complicated vision problems. The two optimization processes shown in Figure 6.1 are unique within themselves in the sense that instead of converging or dispersing particles, the priority is on particle proliferations using rebirths (in our approaches) around the feasible area in a search space. The exploration and exploitation (EAE) phases in a particle system are shown in Figure 6.2. In traditional swarm methods (e.g., PSO [256]) both EAE are simultaneously carried out in all iterations (top right side of the Figure 6.2). The incorporation of EAE based on social calling is the main source of complexity in many nature inspired algorithm (as verified using experiments in Section 3.5). Therefore the priority in this thesis is to decouple EAE into separate phases to increase search efficiency. Such kind of optimisation process is theoretically identical to the searches carried out by the independent foragers in the solution space as shown in Figure 6.3.

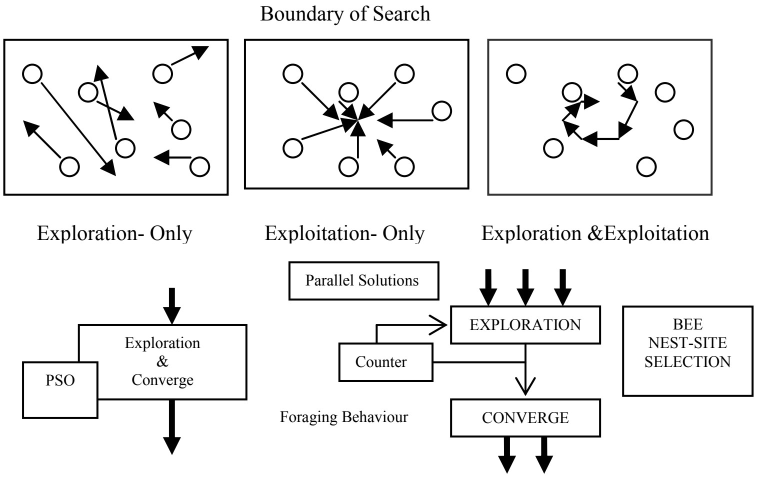


Figure 6.2: Selection of tuning parameters in swarm based approaches.

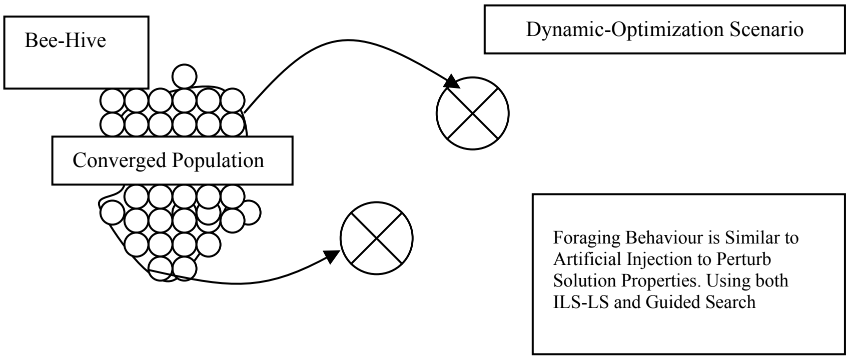

Bee-Hive
Dynamic-Optimization Scenario
Converged Population
Foraging Behaviour is Similar to Artificial Injection to Perturb Solution Properties. Using both ILS-LS and Guided Search

Figure 6.3: Foraging and iterative search strategies without activating the population.

Figure 6.3 replicates the hunting pattern of independent agents which depart from converged colonies (bee-hives/ant nests) in order to find suitable food resources. The larger circles are drawn to display the local olfactory radiuses of the foragers during their flight paths. The scale free searches carried out by autonomous particles therefore are analogous to the artificial injections in dynamic optimisation methodologies [257]. Similar to determining the optimal population size (Figure 3.57), it would also be interesting to analyse the forager particles needed to detect an object.

To supplement the earlier arguments (especially regarding the particle rebirth phenomenon), we have portrayed a pictorial representation of the ant colony optimisation method [258] in Figure 6.4. The path of a wining forager ant is drawn along with those of the unsuccessful ones. The major obstruction in the route (river and mud) enforces the search agents to seek alternative shortest permissible paths to a potential nest site. The larger pheromone trails are therefore deposited (due to the collated opinions of a large number of recruited ants) along the optimal trajectory to the new nest site. The implicit communication method uses pheromone trails to guide the nervous ants (at the nest site) towards a solution. However, one major research question to answer is that whether such implicit methodologies are absolutely mandatory in artificial vision based tracking.

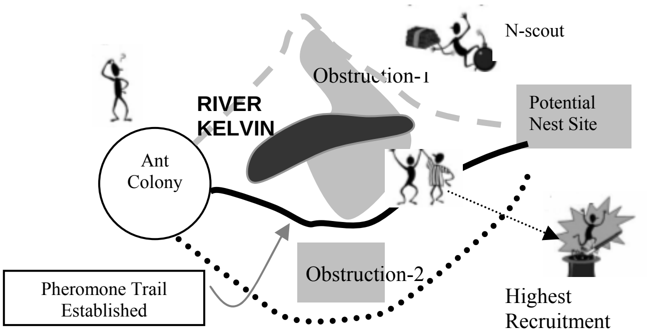
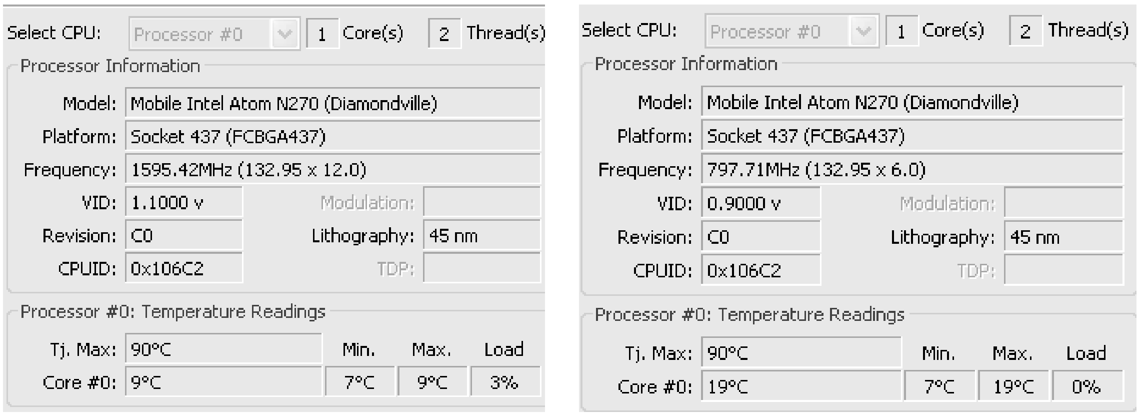


Figure 6.4: The competitive and collaborative environment of ACO.

6.1 Experimental Setting

The aim of Chapter 6 is to demonstrate with experiments that the RSO and virtual agents (memory-less particles) are capable of tracking objects in diversified scenarios. This chapter is organised as follows, the scale space methods (Gaussian blurred images) are applied in Section 6.1 to track a pedestrian in a video sequence. Then in Section 6.2 we will apply the foraging patterns (organised random walks) to track the movements of the same pedestrian. In Sections 6.1-6.13 several detection and tracking experiments are performed on randomly moving objects. First we have applied PSO and BAT algorithms in order to detect and track motions, and later on the same objects are tracked using RSO and VGS to analyse the tracking efficiencies in the space-time dimensions. Finally, the process is repeated for a number of other dynamical objects.



Select CPU: Processor #0 | 1 Core(s) | 2 Thread(s)

Processor Information

Model: Mobile Intel Atom N270 (Diamondville)
Platform: Socket 437 (FCBGA437)
Frequency: 1595.42MHz (132.95 x 12.0)
VID: 1.1000 v
Modulation:
Revision: C0
Lithography: 45 nm
CPUID: 0x106C2
TDP:

Processor #0: Temperature Readings

Tj. Max: 90°C
Core #0: 9°C | Min. 7°C | Max. 9°C | Load 3%

Select CPU: Processor #0 | 1 Core(s) | 2 Thread(s)

Processor Information

Model: Mobile Intel Atom N270 (Diamondville)
Platform: Socket 437 (FCBGA437)
Frequency: 797.71MHz (132.95 x 6.0)
VID: 0.9000 v
Modulation:
Revision: C0
Lithography: 45 nm
CPUID: 0x106C2
TDP:

Processor #0: Temperature Readings

Tj. Max: 90°C
Core #0: 19°C | Min. 7°C | Max. 19°C | Load 0%

Figure 6.5: The properties of the platform on which the experiments are conducted.

The aim of this tracking project is to devise tuning free real time tracking for the embedded platforms (e.g., smart phones). A mobile Intel device (N270-798/1600 MHz) with 1GB of memory was used in our analysis (Figure 6.5) [259]. The tracking does not take into consideration the display characteristics. Therefore experiments are independent of the graphical renderings, accelerations and storage capabilities of the graphic cards.

6.2: Tracking experiment using Motion Estimations.

The detection of the pedestrians and culminating their trajectories is an important task in the emerging vision technologies (e.g., driverless cars [260]). The main ideology (industrial perspective) of any engineering system is to invent a jargon free design that guarantees an optimal performance within the economical constraints (speed and budget) A higher level corroborative module in embedded platforms could be based on analysing movements using the Gaussian convolved image differences. The identification and associations of the relevant information clusters obtained through the frame subtractions $f_t - f_{t-n}(1 \leq n \leq t-1)$ is a cost effective technique to solve complex tracking landscapes.

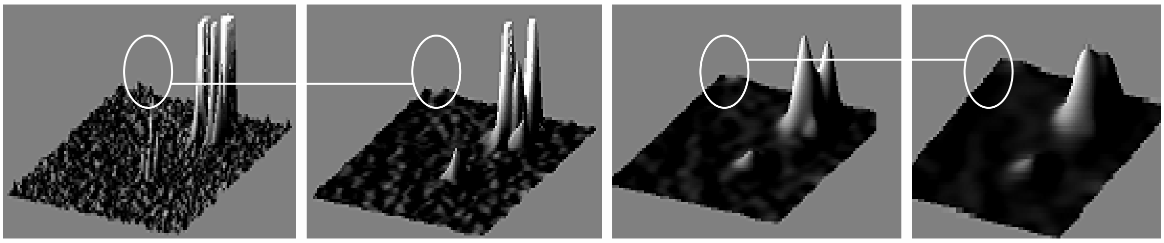


Figure 6.6: The pyramid of information by fusing Gaussian smoothed images [261].

Figure 6.6 shows an information pyramid generated by subtracting two images taken from the pedestrian sequence introduced in Section 5.2.2. The clusters reciprocate the suspected movements and could be further exploited at various scale-space (by convolving with Gaussian models) to analyse motions. The translational movements of the pedestrian-3 in Figure 5.5 are identified by narrowing down the relative pyramids in Figure 6.6. Furthermore, it is quite straightforward to relate contextually different visual problems (e.g., Figure 6.6) to combinatorial optimisation methods applied in the test cases (Figures 3.6, 3.22 and 3.52). Many historical trackers applied inter-connected sub modules to track movements of people [262]. We have presented one typical solution where a Kalman filter was used to track the movements (detailed discussion could be found in [263] and are out of the scope in this thesis). The tracking relies on the state-space model (shown by numerical data in brackets) which utilizes the general equations of motion (kinematics) to analyse movements as shown in Figure 6.7. The results of the tracker have been shown in the top right hand corner of Figure 6.7.

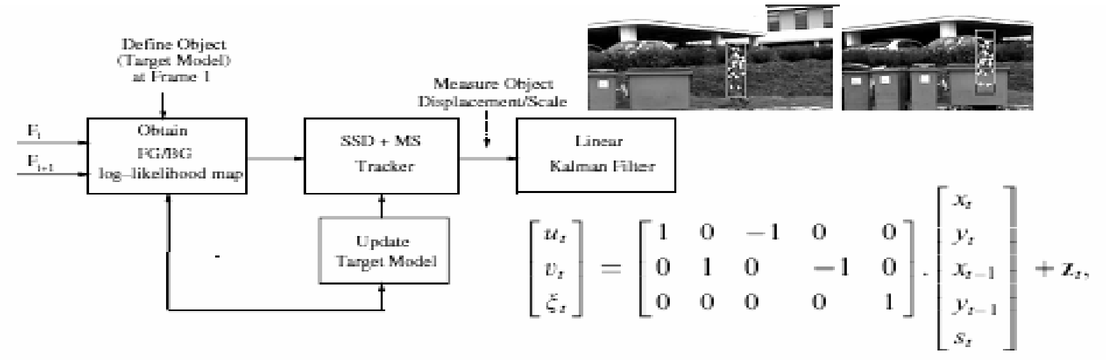

Define Object
(Target Model)
at Frame 1
F_i
F_{i+1}
Obtain
FG/BG
log–likelihood map
SSD + MS
Tracker
Measure Object
Displacement/Scale
Linear
Kalman Filter
Update
Target Model
$\begin{bmatrix} u_t \\ v_t \\ \xi_t \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_t \\ y_t \\ x_{t-1} \\ y_{t-1} \\ s_t \end{bmatrix} + z_t,$

Figure 6.7: Estimating Motion by applying SSD and MS phases to a Kalman filter stage.

To analyse the applicability of a proposed scheme (based on information pyramids shown in Figure 6.6) we used Gaussian convolved frames to deduce frame information and the consequent tracking results (pedestrian-3 in Figure 5.5) have been drawn underneath. The trajectory of the tracking window (using cluster analysis) is presented in Figure 6.8 alongside the ground truth (in Figure 5.5). The overlapping graphs prove that the information fusion by Gaussian convolutions is a valid higher order tracking technique.

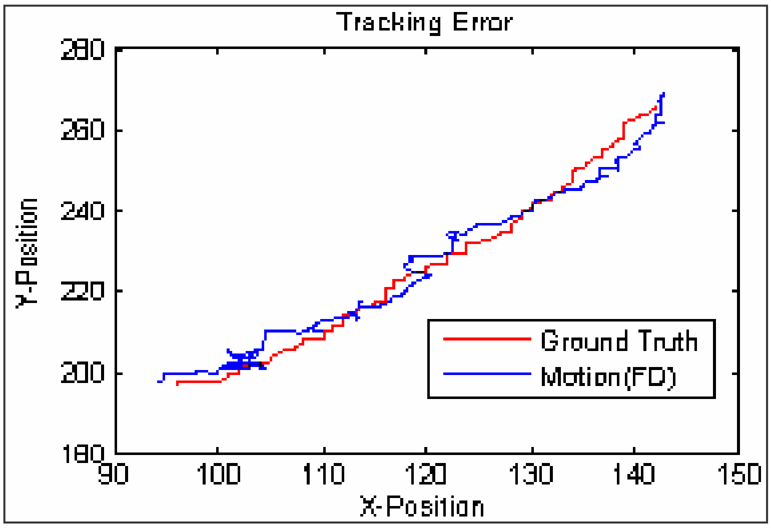


Figure 6.8: Tracking a pedestrian using Gaussian convolved images.

Figure 6.10 shows the timing plot of the Gaussian scale-space based tracking method. The overall convergence time (for a sequence containing *143* images) using frame differences (FD) was observed to be around *5.775* seconds in this experiment. Therefore tracking of a pedestrian was performed at 25 frames per second (FPS), this is quite remarkable finding in itself that could also be applied in narrowing down the search space in more complex frames (we used a window of size *10x10 pixels* around the last known position of pedestrian as shown in Figure 6.9). Therefore, in the pedestrian tracking cases, the FD cluster analysis appears to be an effective approach (as apparent from the graphs). The mean pixel distance error $\begin{bmatrix} y_e \\ x_e \end{bmatrix}$ for *143* frames (in Figure 6.8) was observed to be $\begin{bmatrix} 1.86 \\ 2.37 \end{bmatrix}$, which is quite remarkable for a tracker based on FD alone. However, there is a significant margin for further improving the frame rates as in Section 6.2.

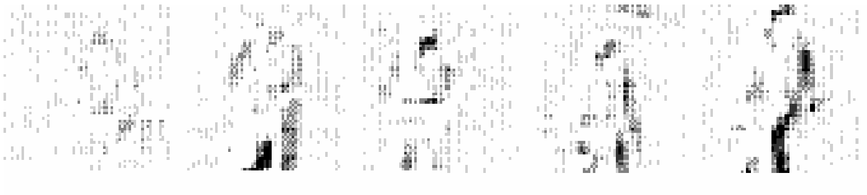


Figure 6.9: Information deductions from a variety of video frames using scale space methods.

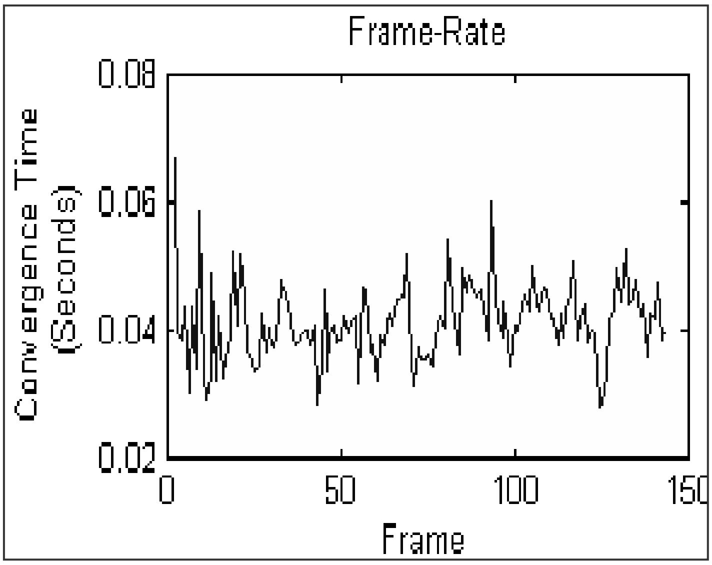


Figure 6.10: Timing efficiency graph in a pedestrian sequence.

6.3: The Motion Estimations using Foraging Patterns.

The real time tracking approach in this thesis predominantly relies on a fact that the search space could be narrowed down using information fusion (IF) techniques [264] (as learnt in Sections 6.1 and 3.4). The deep-seated reason for the success of the nested RSO (proposed in the Section 3.1-3.3) is that it applies various IF stages to select a best hypothesis. The frame subtraction method is an exemplification of an elitism [265] based meta-heuristics (similar to the honey-bird human relationship in Section 3.2.3) that could also be applied in the real time applications. In the coming demonstrations, we will analyse with experiments the fact that how IF in visual tracking could conjecture search reductions. The partial function evaluation of the colour distribution (introduced in Section 4.1.4) will be used as an objective criterion for the foraging particles in this case.

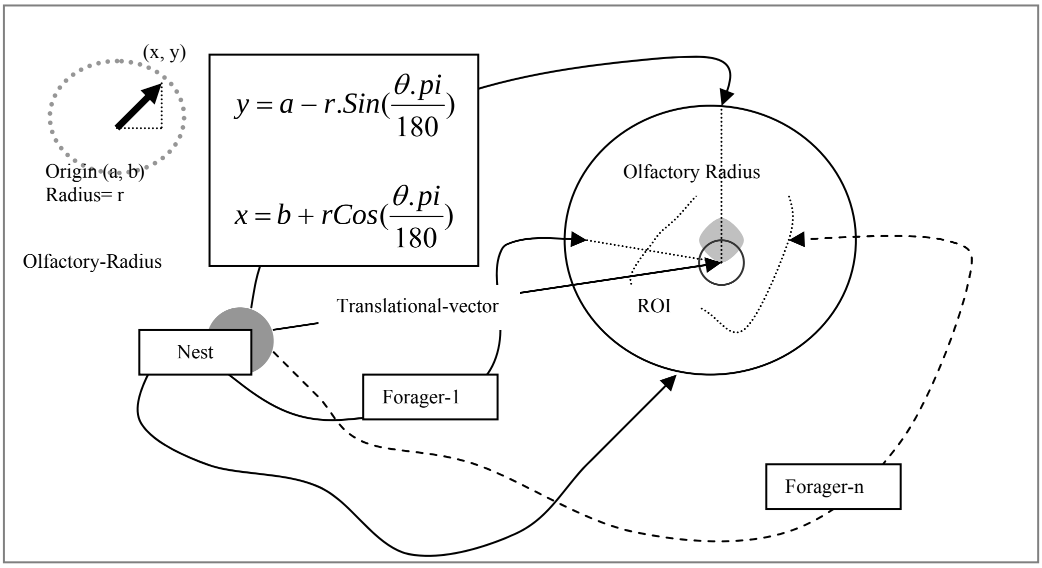


Figure 6.11: Tracking an ROI in video sequence using only a few foraging-particles.

An alternative to usual analytical deterministic drifts is presented in Figure 6.11. There are n-foragers in the search space which form a cluster around a potential region of interest (ROI), and perform iterative local searches within the olfactory radius (determined by a higher level process, e.g., clusters formulated by frame subtractions) to detect the object. The radiuses of the olfactory senses are learnt along time by taking into consideration the historical motions

(Section 4.1.3), and the initial position of particles $\begin{bmatrix} y \\ x \end{bmatrix}$ are calculated by implementing a circular symmetric model (in this scenario) as drawn in Figure 6.11. At a point in time, n foragers leave the nest and are represented by arrows. However by exploiting the properties 1-3 (Section 3.3.2), all intermittent and transitional velocity updates are avoided to achieve a real time convergence.

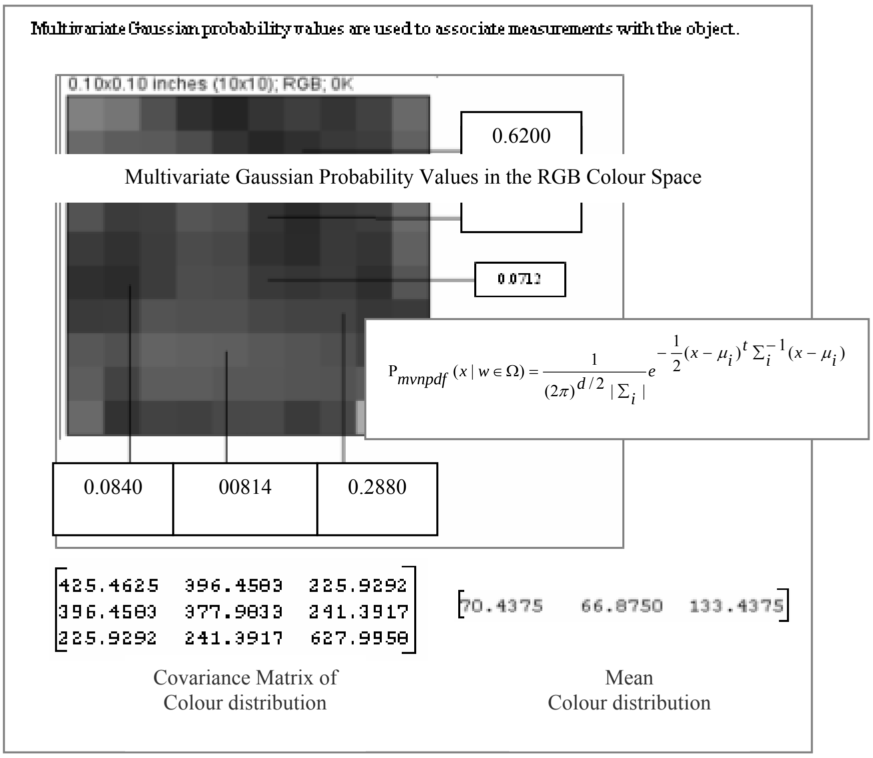

Multivariate Gaussian probability values are used to associate measurements with the object.
0.10x0.10 inches (10x10); RGB; 0K
0.6200
Multivariate Gaussian Probability Values in the RGB Colour Space
0.0712
$P_{mvnpdf}(x \mid w \in \Omega) = \frac{1}{(2\pi)^{d/2} \mid \Sigma_i \mid} e^{-\frac{1}{2}(x-\mu_i)^t \Sigma_i^{-1} (x-\mu_i)}$
0.0840
00814
0.2880
$\begin{bmatrix} 425.4625 & 396.4583 & 225.9292 \\ 396.4583 & 377.9833 & 241.3917 \\ 225.9292 & 241.3917 & 627.9958 \end{bmatrix}$
$\begin{bmatrix} 70.4375 & 66.8750 & 133.4375 \end{bmatrix}$
Covariance Matrix of Colour distribution
Mean Colour distribution

Figure 6.12: The partial function evaluation using multivariate Gaussian PDF.

Once main foragers have presented their hypothesis about the validity of a potential nest-site (similar to foraging bees) further evaluations are carried out using a small group of recruits. The recruits fly around autonomously within the designated space to detect the clusters belonging to the object. All forging particle use partial function evaluations of the colour distributions (as described in Section 4.4.3) in Figure 6.12. Similar to the probabilistic associations established by using grey level or intensity changes in Figure 4.40, the corresponding RGB variations are modelled using a multivariate Gaussian distribution [266]. Moreover, the particles utilise non-linear speed function that makes them stationary in the vicinities where object like features are detected (refer to Figure 4.44). Due to the simplicity of this feature matching process, only the encoded values of mean and covariance matrix of the colour variations (within a region using RGB space) are required to assign probability values to each pixel in the search space (as shown in Figure 6.12). Figure 6.13 shows how the particles search space are manipulated over the course of time to calculate the underline movements in this video.

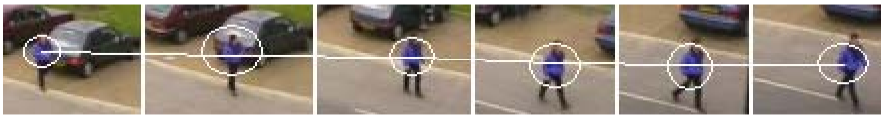


Figure 6.13: Six tracking frames from the pedestrian sequence using olfactory radiuses.

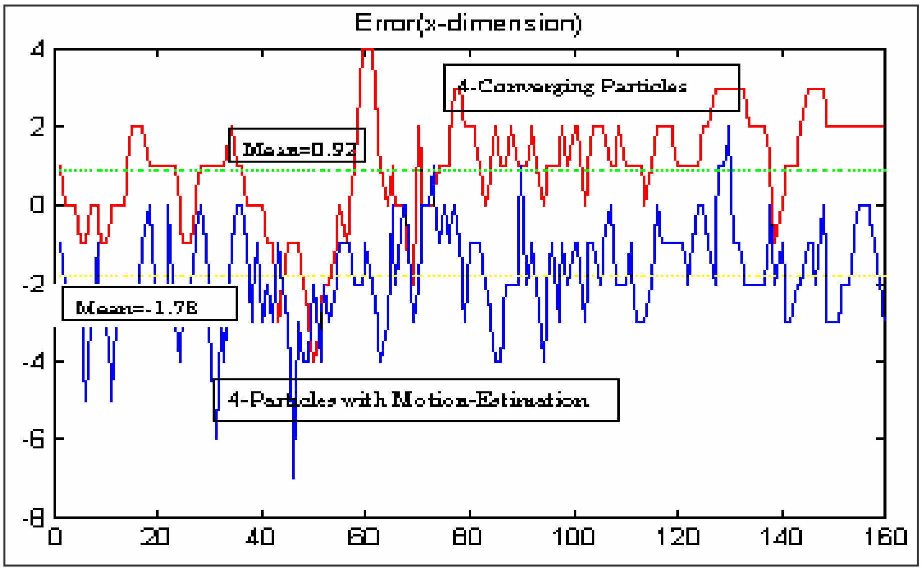


Figure 6.14: Tracking errors in the x-dimension using 4 scouting particles.

The tracking results in the graphical formats are presented in Figure 6.14. In the first scenario, particles are deployed using pyramid structures (obtained through FD as a top order heuristic) to narrow searches. The results in XY search space dimensions are compared to the ground truth (GT) in Figure 6.14, and the corresponding accumulated errors are presented in Figure 6.15. The errors in the XY dimensions (when FD and foraging behaviour [267] were applied together) are observed to be -1.78 and -2 pixels respectively. The discrepancies are further reduced to a mean of 0.92 and 0.2 when only 4-scout (RSO) particles are initiated in the relevant space without analysing any top level clusters through the Gaussian pyramids [268].

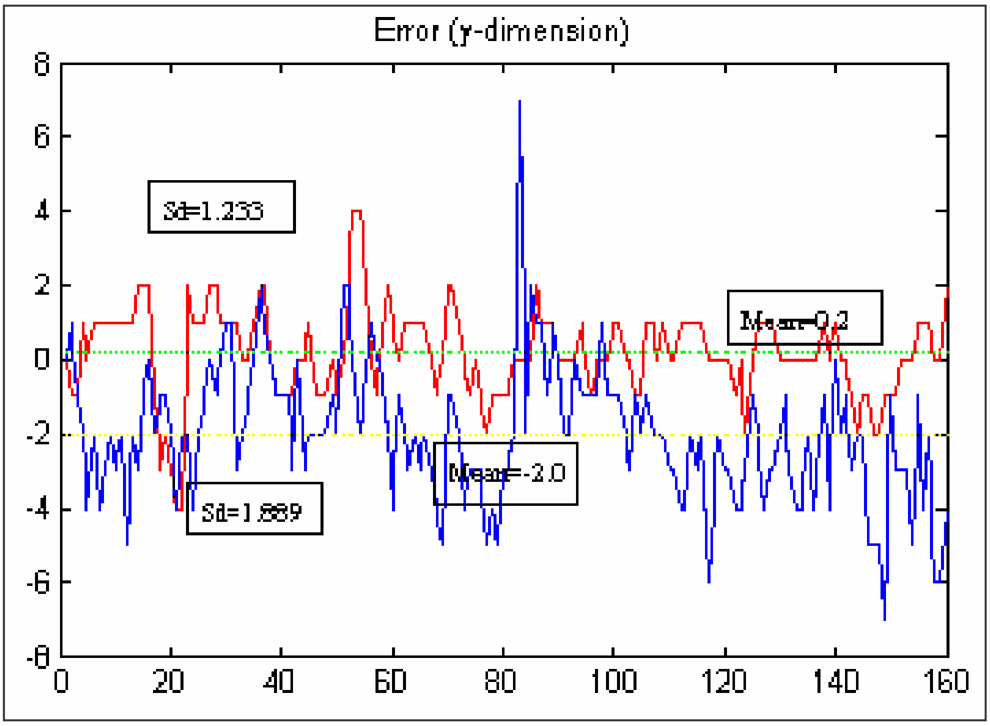


Figure 6.15: Tracking errors in the y-dimension using only 4 scouting particles.

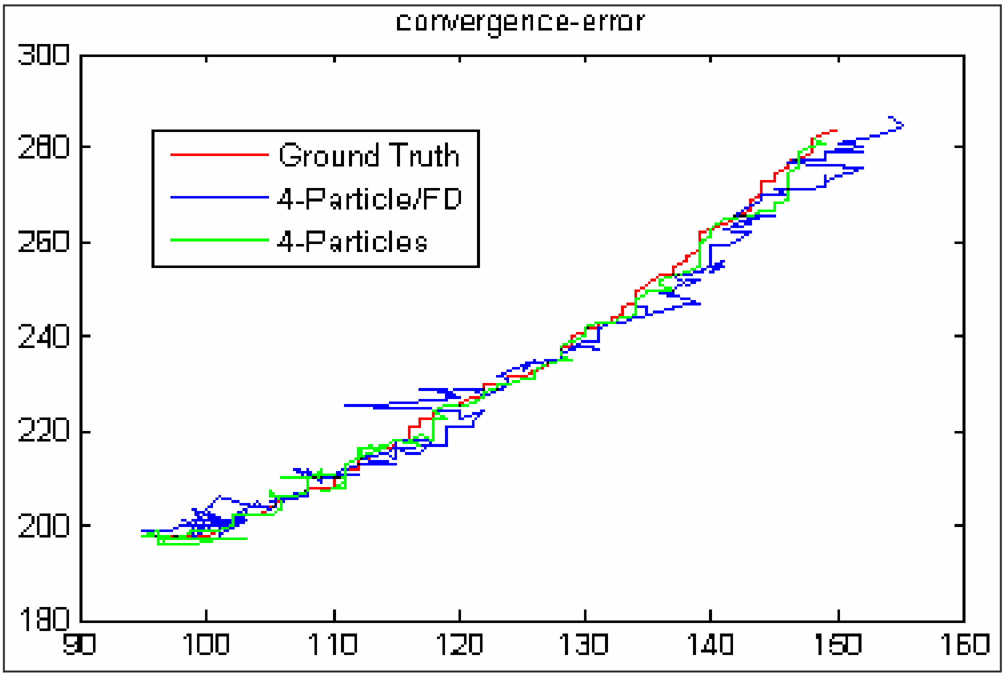


Figure 6.16: Tracking results of 4 particles using FD and when only scouts were deployed.

In Figure 6.16, the ground truth positions of the pedestrian are plotted alongside two further scenarios when the ROI was being tracked using foraging behaviours alone, and with frame subtraction as a higher order heuristics. The results are convincing and stipulate an earlier hypothesis that the nested RSO could significantly outperform single solution based analytical methods (e.g., Figure 6.7).

The timing graphs are also presented in Figures 6.17 and 6.18. When the mobile processor is running at its peak operating frequency (Figure 6.5), we achieved a mean tracking speed of 100 frames per second (FPS), which is significantly higher than the system implemented in Figure 6.7. The dominating performance of a simplistic natured inspired approach (foraging behaviours) is therefore very conclusive in this application.

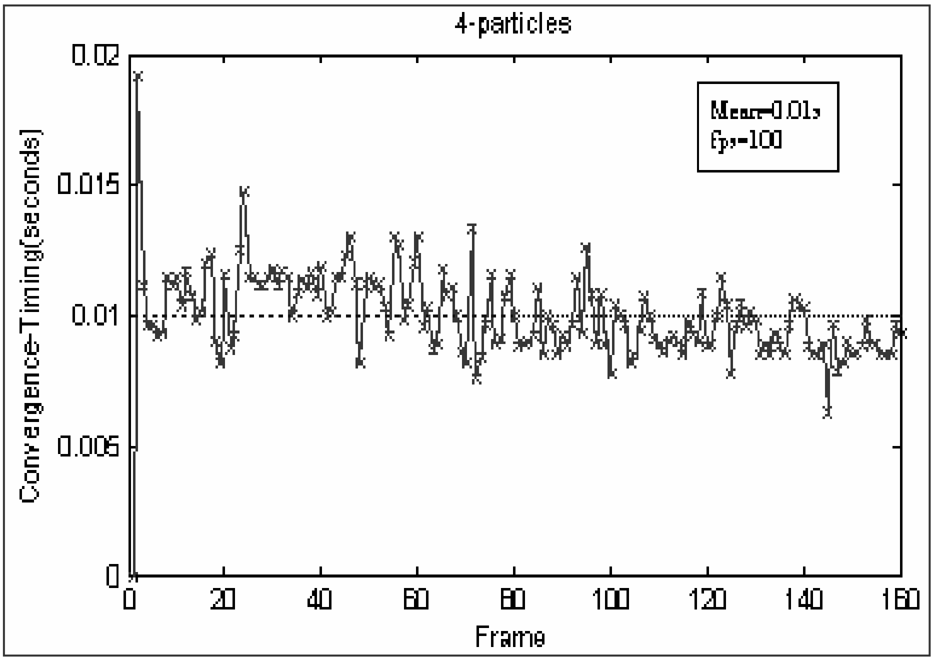


Figure 6.17: The graph of the convergence timing at a processor speed of 1595 MHz.

The tracking results for 15 frames are shown in Figure 6.19. The position of the tracking windows represents the mean pedestrian location detected in a frame. The pictorial data also confirms that this tracker works remarkably well (at impressive frame rates) in detecting the pedestrian movements during the entire length of this sequence.

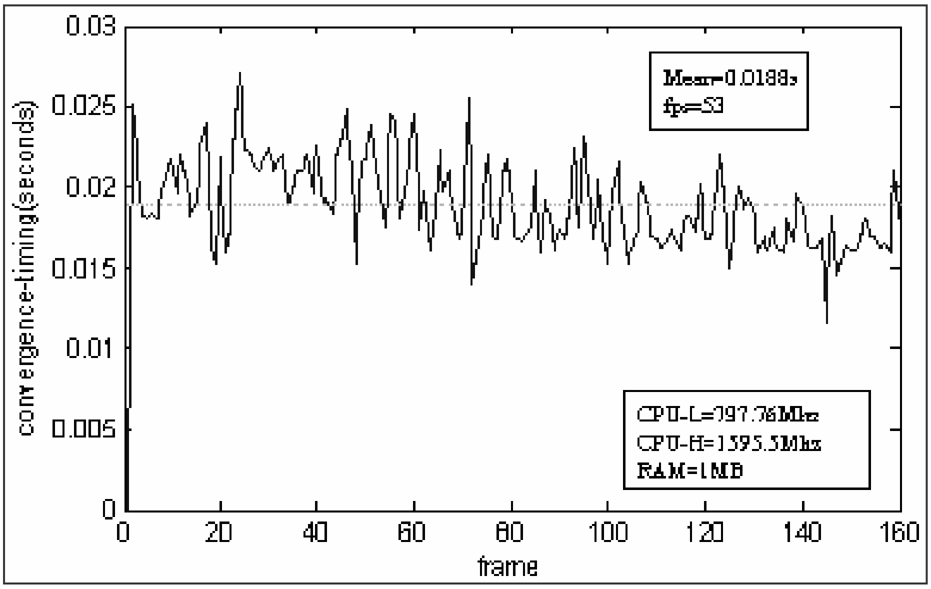


Figure 6.18: The convergence timing graph for a processor running at 798 MHz.

The tracking results for 15 frames are shown in Figure 6.19. The position of the tracking windows represents the mean pedestrian location detected in a frame. The pictorial data also confirms that this tracker works remarkably well (at impressive frame rates) in detecting the pedestrian movements during the entire length of this sequence.

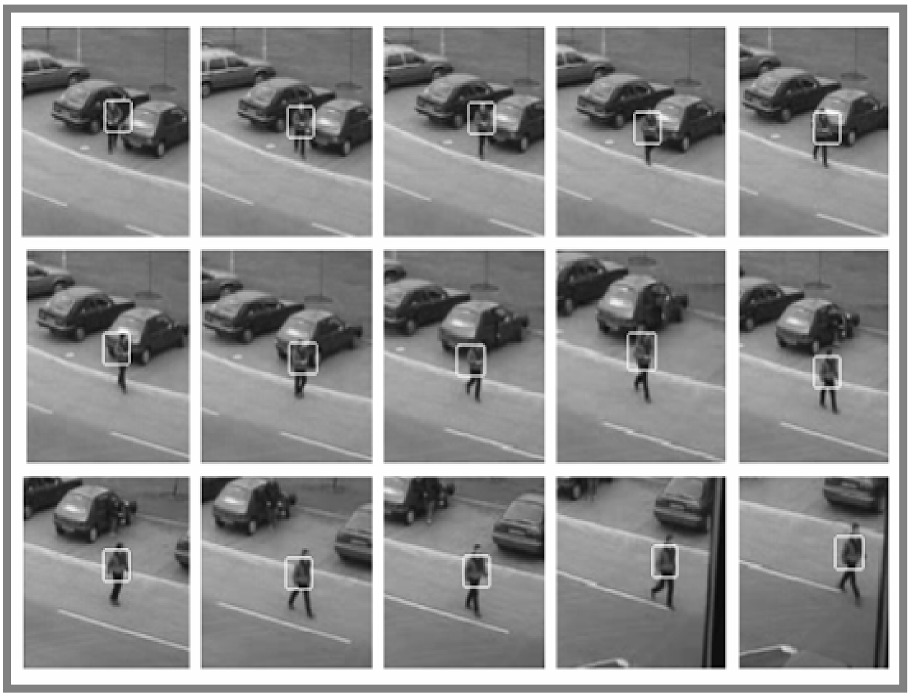


Figure 6.19: Tracking results using only 4-foragers at an FPS of around 100.

The analysis in Figure 6.17 was repeated for a range of other particle populations, and the tracking results are provided in Figures 6.20 and 6.21. By comparing Figure 6.20 with Figure 6.17, we reached to a conclusion that no significant improvements (in the tracker accuracy) are observed by increasing the population size of the foraging particles in this particular pedestrian tracking problem.

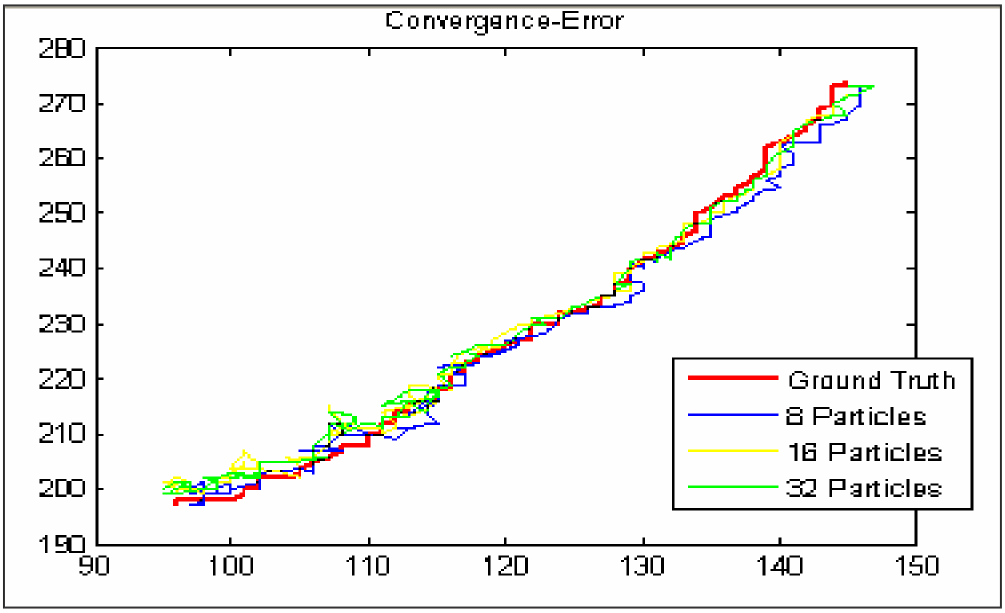


Figure 6.20: Tracking results using 8, 16, and 32 numbers of foragers.

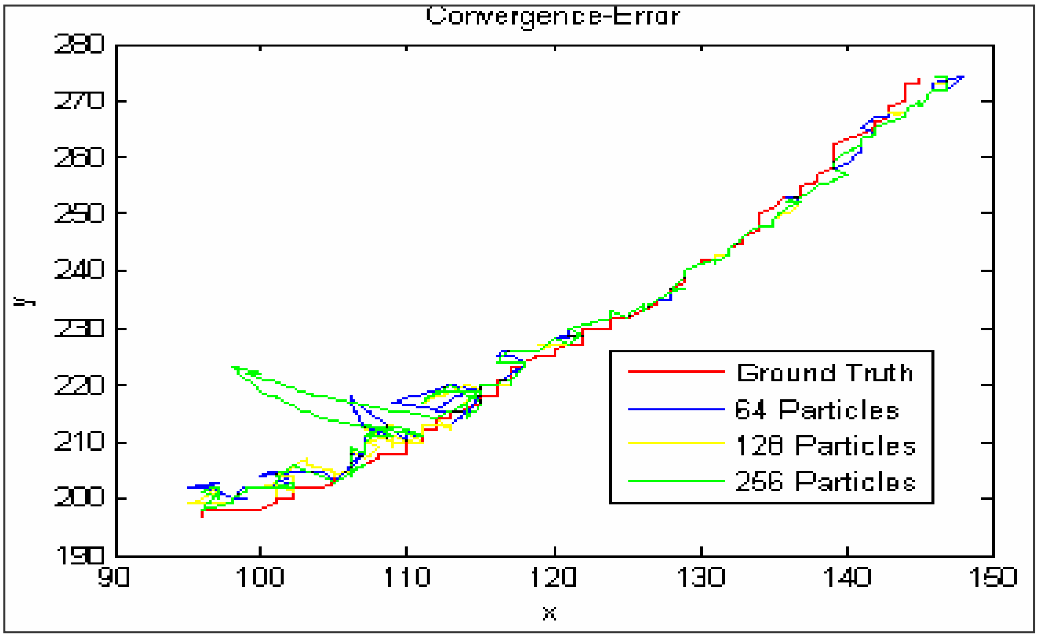


Figure 6.21: Pedestrian tracking using 64, 128, and 256 numbers of foragers.

The effects of the particle population strengths on the convergence timings of the tracking algorithm are drawn in Figure 6.22. The mean convergence time for 8 particles was recorded to be around 0.147 sec, and by increasing the particle size to 32, the convergence time jumped to 0.88 second (Figure 6.22). The timing graphs for 64, 128 and 256 particles are also plotted in Figure 6.23 (with worsening convergence timings). Hence, no particular improvement in terms of the convergence accuracy is ever recorded in this test by incrementing the foraging population sizes (Figures 6.20 and 6.21).

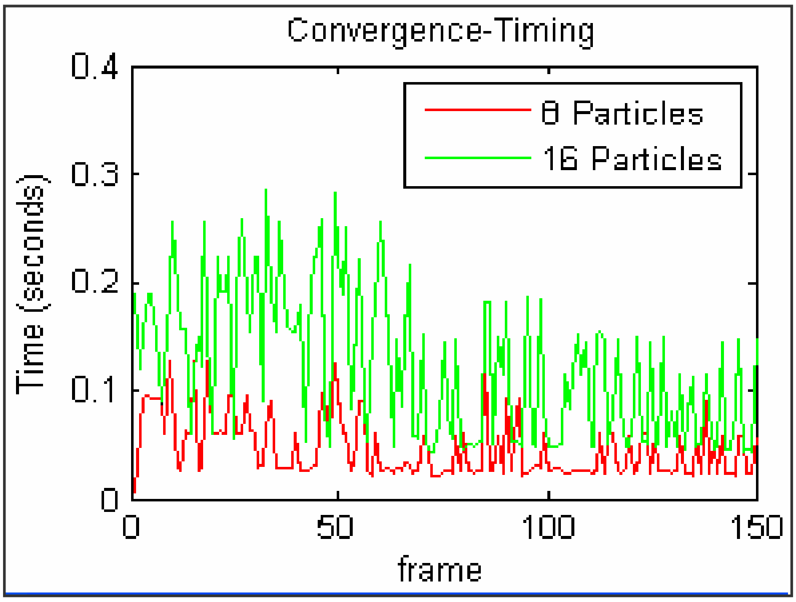


Figure 6.22: The convergence timing graph for 8, 16 particles.

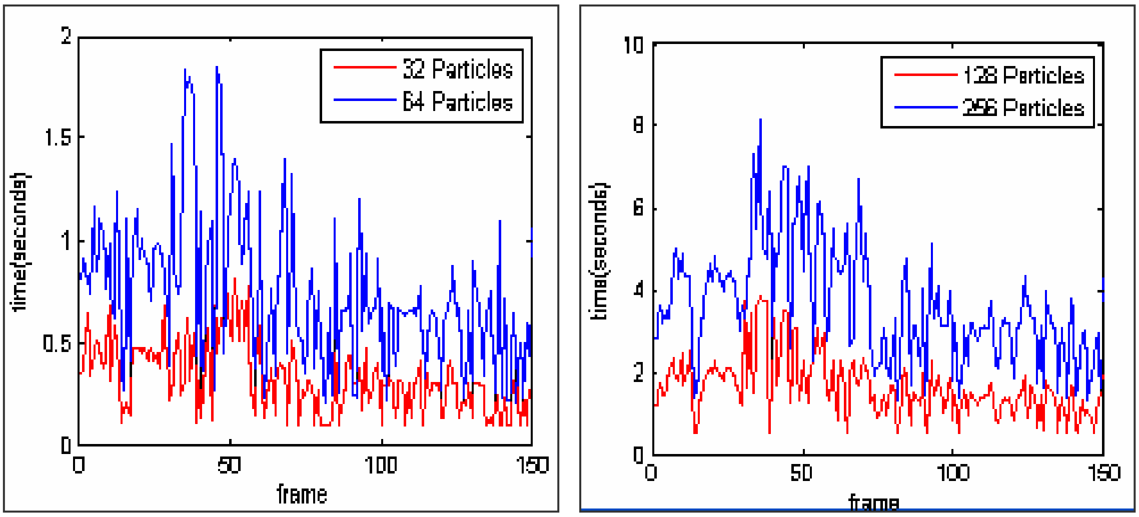


Figure 6.23: The convergence timing graph for 32, 64, 128 and 256 foragers.

6.4: Solving the detection problem using virtual guided searches (VGS).

The detection of an object in an image often becomes a challenging multi-modal problem. The underline idea of this section is to apply RSO proposed in Chapter 3 to find an object of interest. The inherent philosophy of the devised RSO method is to highlight the fact that learning of dynamics using scale free experiences is a preferable optimization method than the deterministic drifts employed in both standard Kalman and particle filters. It was demonstrated earlier (Figures 3.63-3.66) that the nostalgic factors (e.g., previous best positions) do not contribute as much as it was previously thought in the optimisation of evolutionary test cases. Other major sources of functional discrepancies in the agent based methods are the memory operations needed to update the particle's positions [269].

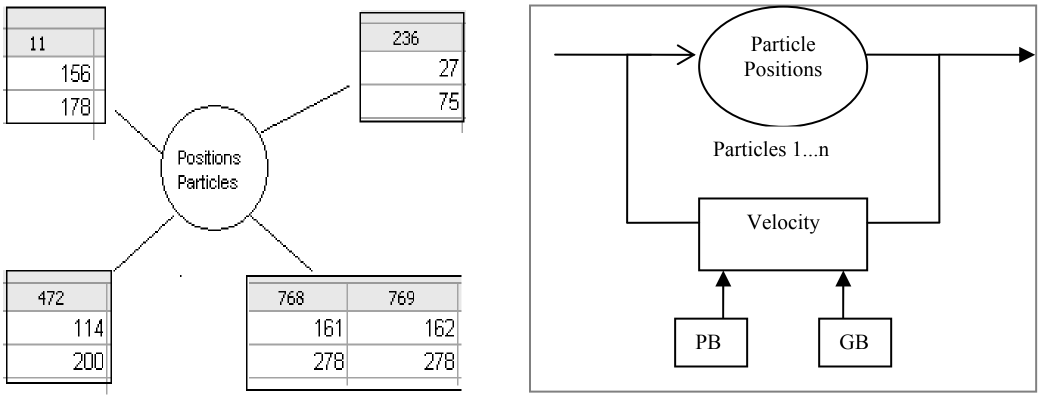


Figure 6.24: Understanding the notion of particle positioning and recording in memory.

A predominant source of the computational complexity in particle centred schemes (e.g., PSO, BAT and Firefly algorithm) is that the agent's positions are stored in complex data records (Figure 6.24) which are not mandatory at all in the view of the author. In particle swarm optimisation, the velocity modelling relies on the personal best (PB) and global best (GB) positions, and particle positions are updated by applying newly calculated velocities to the stored positions. A virtual particle on the other hand does not require such complicated data structures, and no particular velocity models are kept in the record to specify their search behaviours in the feasible space.

The virtual particles (similar to foraging insects) fly around intermittently (e.g., based on stochastic Gaussian models) without broadcasting their positions to the rest of the population. Once a suitable region is identified using broader searches, a virtual agent utilises the available communicational method to advertise its findings. The denser guided searches (Equation 4.1) are then applied in order to accept or reject a hypothesis. The idea therefore revolves around devising an effective recovery phase to regain control of the tracking windows (as was explained in Sections 4.1, 4.2). The virtual guided search (VGS) is hence a novel idea presented here to solve the vision based problems. We aim to apply both VGS and the RSO in the coming sections.

The operational environments for the visual detection problems are presented in Figure 6.25. The detection challenges are exacerbated in the drone sequence due to the variations in scale and background clutter. Furthermore, the illumination and shadows complicate the process of identifying the ant positions within allowable time in Figure 6.25. Both ant and drone sequences would be extensively used in the coming sections to analyse the most plausible tracking method.

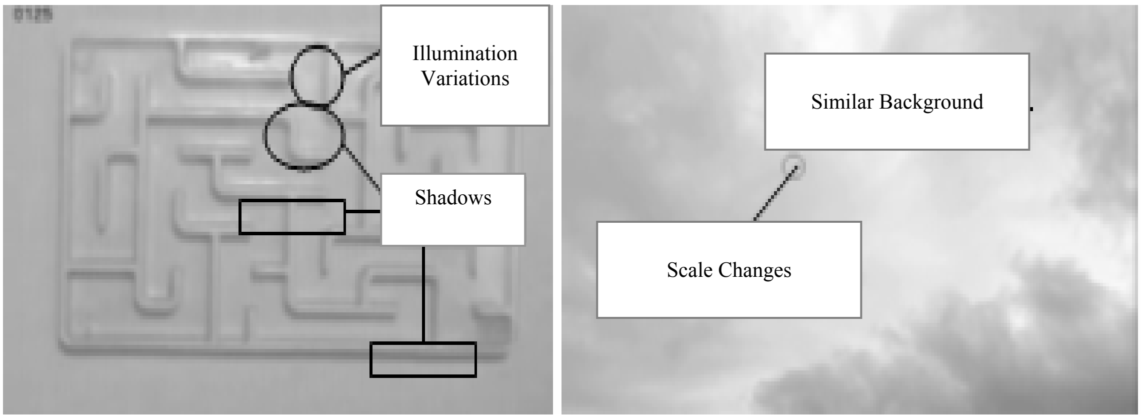


Figure 6.25: Detection of object of interest becomes a blind search in case the trail is lost.

6.4.1 Detection Experiment (the ant in a maze sequence).

In this experiment we kept the window size to $\begin{bmatrix}30\\30\end{bmatrix}$ pixels in both x-y dimensions, and the aim here is to analyse the effectiveness of the exploratory features of both PSO and Bat algorithms. The virtually guided particle behaviours are then applied in order to research into the most suitable tracking methodology for a real time performance. A complicated frame was chosen from the video (introduced in Section 5.2.1) for the detection runs. The surface plot of the problem is shown in Figure 6.26, and it is evident that this problem is highly rippled with similar peaks present in the surroundings.

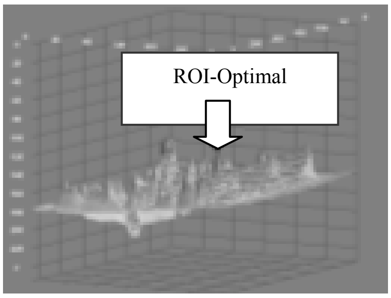


Figure 6.26: The surface plot of the search space shows peaks of matching densities.

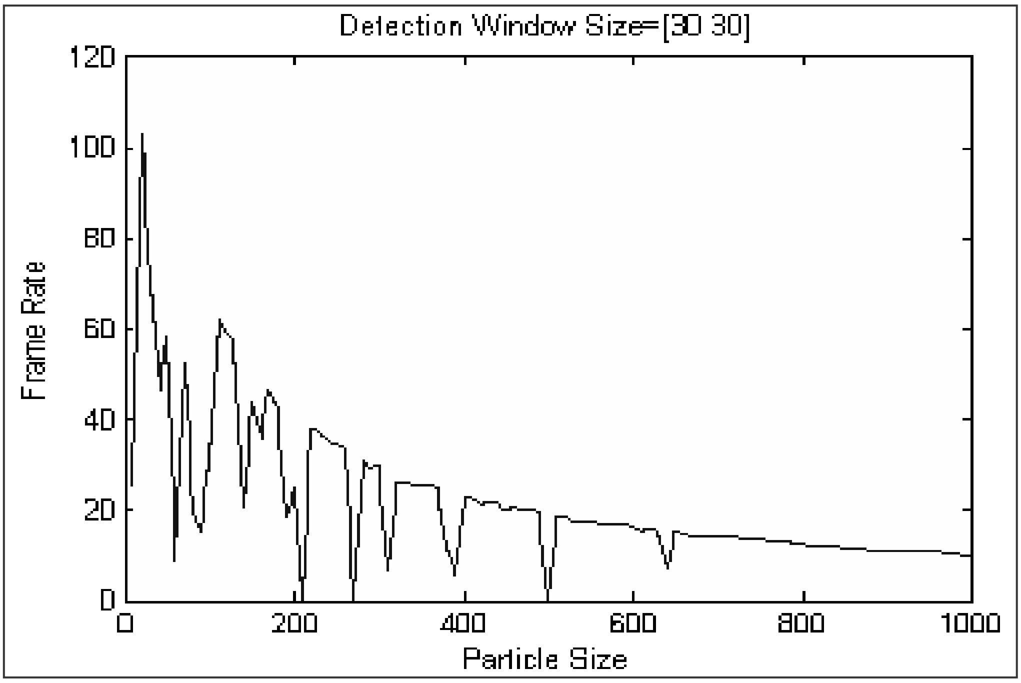


Figure 6.27: The visual detection of ant is carried out by using a particle sweep in PSO.

Figure 6.27 shows the detection results when the newly devised PSO format (based on the stochastic variations defined in Equations 3.12-3.13) is applied to the ant trapped in a maze image sequence. A population sweep (similar to Figure 3.57, where a particle population is sequentially increased to study the consequences on an optimisation process) was conducted, and the detection behaviour is plotted for 5-1000 particles. The general trend that could be observed in Figure 6.28 is that the computational complexity increases almost exponentially with the proliferating agent population. The best detection speeds are however observed when the particle population is ranging between the limits 50-150, and within this population span, a mean detection rate of 41 fps are recorded.

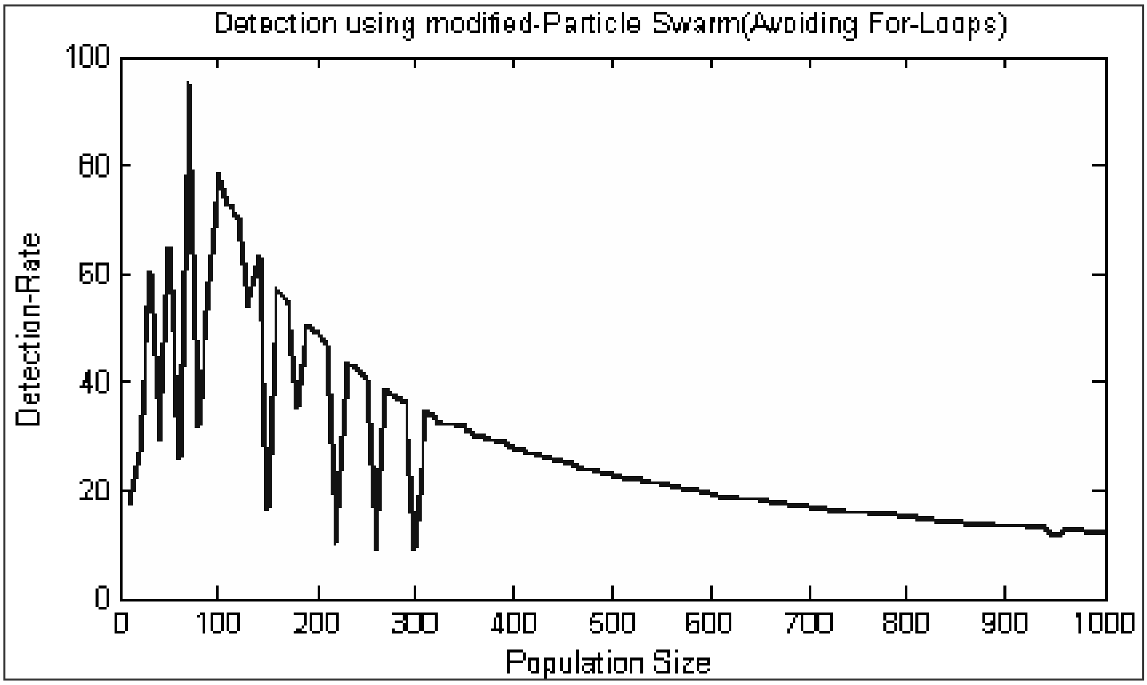
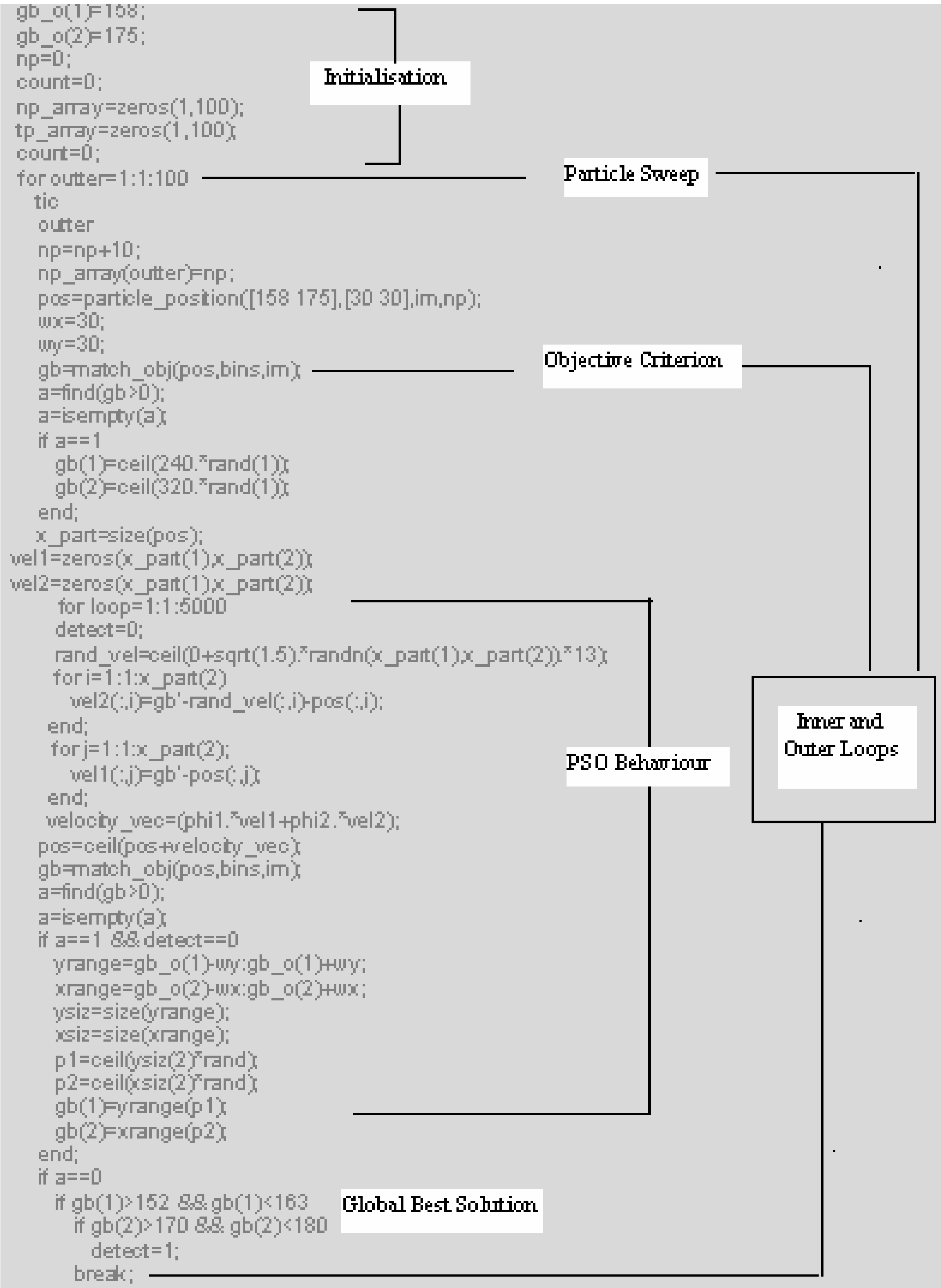


Figure 6.28: The convergence behaviour of a modified PSO without using for-loops.

A code optimisation (CO) procedure [270] was carried out onto the PSO program (applied in Figure 6.27). The main idea of CO is to avoid the conventional ‘for loops’ (a type of programming construct) which are extensively applied during particle translations in a search space (Figure 6.29). The code in Figure 6.29 consists of five major sections, an initialization phase (particles are spread in the space), an outer loop to sequentially increment the population sizes, and the calculation of the objective values in the RGB space, a PSO behavioural implementation (Equations 3.5, 3.12 and 3.13), and finally calculating the Euclidean distances between the observed positions and the ground truth value of $\begin{bmatrix}160\\175\end{bmatrix}$. The detection results after applying CO are shown in Figure 6.28, the detection rates were increased to a mean of 52.31 fps.



```
 gb_o(1)=158;
 gb_o(2)=175;
 np=0;
 count=0;
 np_array=zeros(1,100);
 tp_array=zeros(1,100);
 count=0;
 for outter=1:1:100
   tic
    outter
    np=np+10;
    np_array(outter)=np;
    pos=particle_position([158 175],[30 30],im,np);
    wx=30;
    wy=30;
    gb=match_obj(pos,bins,im);
    a=find(gb>0);
    a=isempty(a);
    if a==1
      gb(1)=ceil(240.*rand(1));
      gb(2)=ceil(320.*rand(1));
    end;
   x_part=size(pos);
vel1=zeros(x_part(1),x_part(2));
vel2=zeros(x_part(1),x_part(2));
      for loop=1:1:5000
      detect=0;
      rand_vel=ceil(0+sqrt(1.5).*randn(x_part(1),x_part(2)).*13);
      for i=1:1:x_part(2)
        vel2(:,i)=gb'-rand_vel(:,i)-pos(:,i);
     end;
     for j=1:1:x_part(2);
        vel1(:,j)=gb'-pos(:,j);
     end;
     velocity_vec=(phi1.*vel1+phi2.*vel2);
    pos=ceil(pos+velocity_vec);
    gb=match_obj(pos,bins,im);
    a=find(gb>0);
    a=isempty(a);
    if a==1 && detect==0
      yrange=gb_o(1)-wy:gb_o(1)+wy;
      xrange=gb_o(2)-wx:gb_o(2)+wx;
      ysiz=size(yrange);
      xsiz=size(xrange);
      p1=ceil(ysiz(2)*rand);
      p2=ceil(xsiz(2)*rand);
      gb(1)=yrange(p1);
      gb(2)=xrange(p2);
    end;
    if a==0
      if gb(1)>152 && gb(1)<163
        if gb(2)>170 && gb(2)<180
          detect=1;
        break;
```

Figure 6.29: A section of the detection code written in Matlab (image processing toolbox).

Figure 6.30 shows the positions of the particles with respect to the object of interest for various tuning parameter selections (Equations 3.12-3.13). The video frames are taken after the maximum allowable time lapses during the runs. The non-convergent characteristic of the virtual particles could be observed in Figure 6.30d. On the other hand particles collapsed at a point in the situation c (where the parametric values are chosen as $\Phi_1 = 0.8$ and $\Phi_2 = 0.2$). Figure 6.31 shows the convergence timing for a detection experiment with $\Phi_2 = 0.1$. Higher convergence rate (129.7 FPS) is observed in the range where the tuning parameter was $0.01 \le \Phi_1 \le 0.45$, whereby selecting its values between $0.45 \le \Phi_1 \le 0.8$ resulted in the detection rates dropping to a mean of 53 FPS. Therefore the experiment (in Figure 6.31) also verified that visual detections are highly sensitive to the parametric choices in the swarm based methods (as suspected).

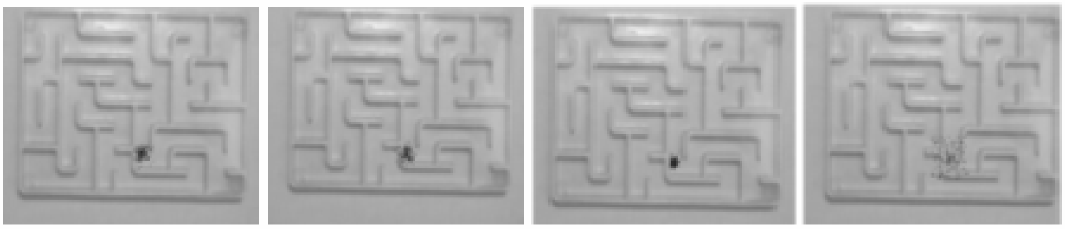


Figure 6.30: (a) Phi1=0.6, Phi2=0.2 (b) phi1=0.6, phi2=0.3 (c) phi1=0.8, phi2=0.2 (d) Virtual-particles

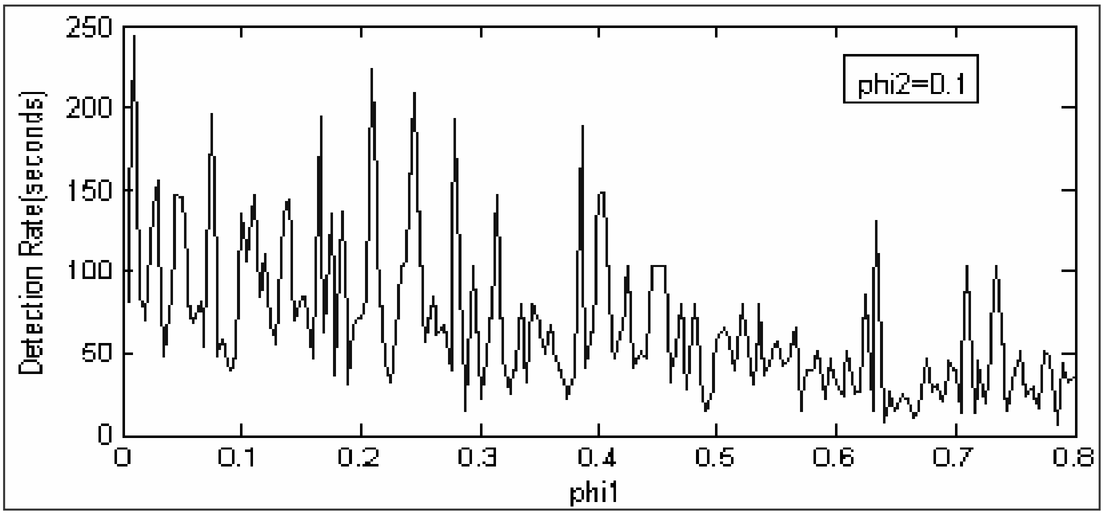


Figure 6.31: A modified PSO without using for-loops in the velocity and positional updates.

Similar to Figure 6.27, the bat algorithm was implemented to detect the object in the relevant landscape (Figure 6.26), and the results are presented in Figure 6.32. The mean convergence timing for a population size of 50-150 particles was observed to be around 143 fps, which is significantly higher than both PSO variants. Finally the detection was repeated for the virtual particle behaviours, and the results (Figure 6.33) are remarkably higher (as expected) compared to both Bat and PSO algorithms. The detection rate of 200 fps using virtual particles also verifies the earlier proposition (Section 6.3) that the memory operations in PSO and Bat algorithms are computational overheads with no optimisation impacts whatsoever.

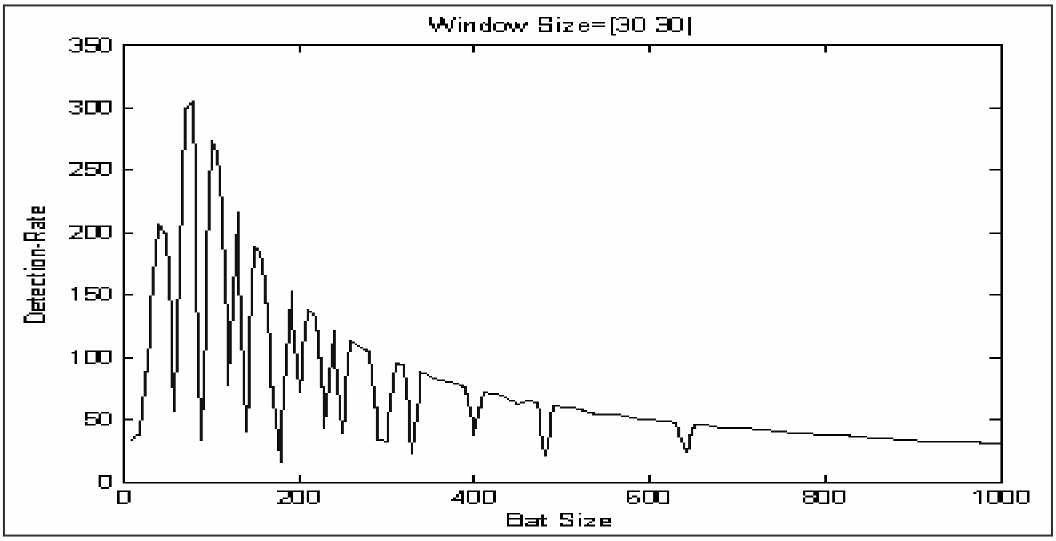


Figure 6.32: The detection timing when Bat algorithm was used to find the ant in a maze.

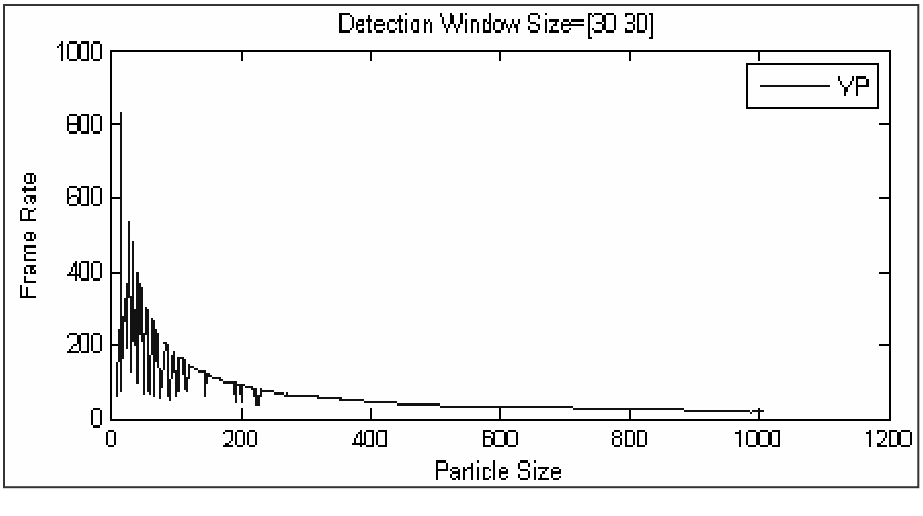


Figure 6.33: Frame rate/ detection time using only virtual-particles (VPs).

6.5: Solving the Detection problem in the widened Search Space.

The operational basin in the meanshift (MS) visual detections remains static throughout the tracking (Section 2.5.4). One way to rectify the inherent weakness of the MS algorithm is to automatically alter the window sizes to incorporate more diversified measurements. However widening of the search space gives birth to an expedited accumulation of errors, as more clutter is usually introduced into the measurements as shown in Figure 6.34 (when the search space widens from Ω_1 to Ω_n, a lot more distracters enter into the tracking window). Figure 6.35 has been specifically drawn to relate the above mentioned problem in a typical detection case. As the tracking window widens, more surface abnormalities corrupt the measurements, and calculation of the correct MS vector is difficult to achieve.

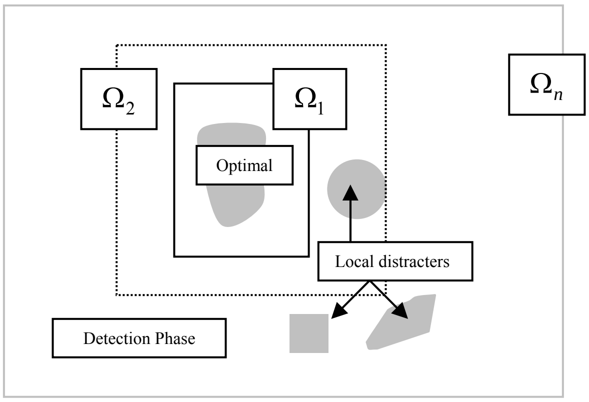


Figure 6.34: Search space selection and incorporation of local clutter in the measurements.

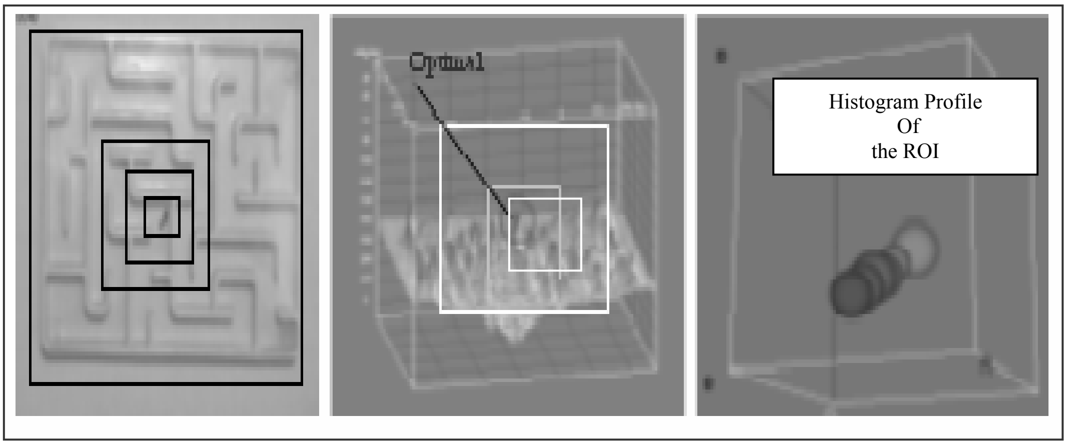


Figure 6.35: (a) ROI in frame. (b) Surface Plot. (c) A depiction of discrete density model.

In this section we will conduct further experimentations to check that whether the Guided local search (GLS) could be used to identify local clutter and the changing histogram profiles in Figure 6.35 (c). We will iteratively increase the window sizes as shown in Figure 6.35(a) to penalise the distracters (using the guided local searches as in Equation 4.1). In the first case, detections are carried out by widening the tracking window ($\begin{bmatrix}10\\10\end{bmatrix}$-$\begin{bmatrix}210\\210\end{bmatrix}$) without the GLS module, and the results are presented in Figure 6.36.

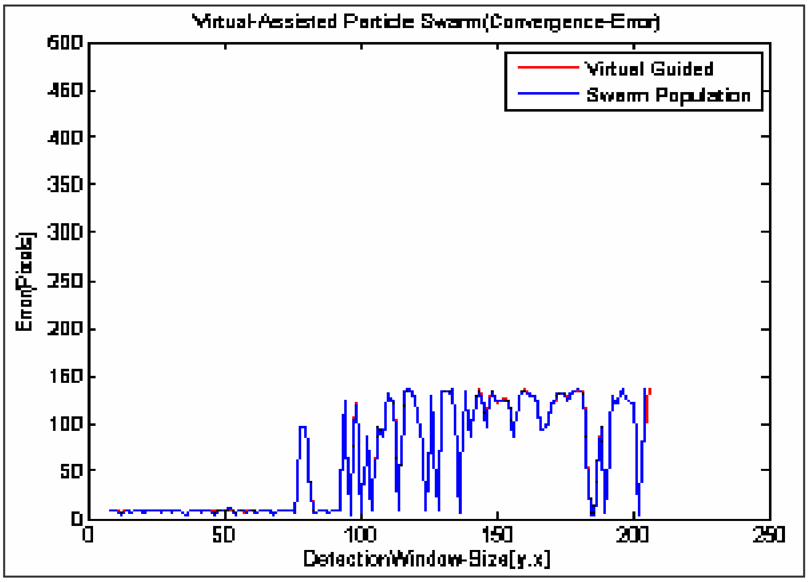


Figure 6.36: The detection using PSO failed drastically after 80th frame.

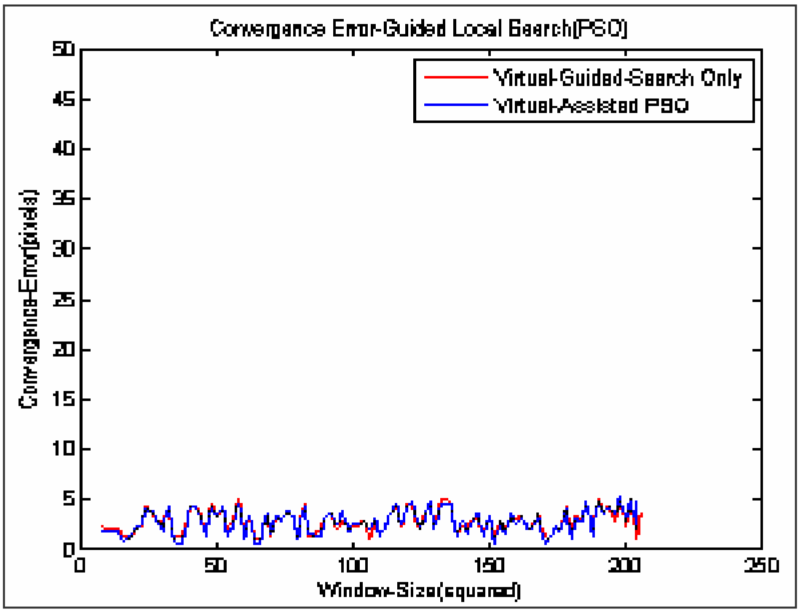


Figure 6.37: The guided search maintained the position of object as the window is enlarged.

The guided search was then applied in Figure 6.37 exploiting two optimisation paradigms. First virtual guided search (VGS) was used as a meta-heuristic over the PSO, and in the second case, the experiments were repeated using VGS on its own. The results in Figure 6.37 show that both algorithms worked competitively well and errors are significantly reduced. The tests are repeated for the Bat algorithm and VGS algorithm in Figures 6.38-6.39 (two plots exhibit VGS assisted Bat and the VGS tracking on its own). The plot in Figure 6.39 shows how GLS facilitated the recovery of the Bat detections carried out in Figure 6.38.

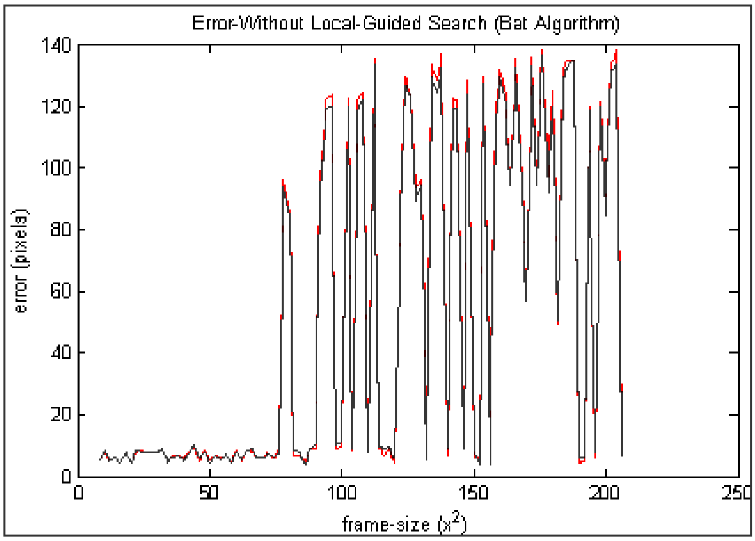


Figure 6.38: The error plot of detection using Bat algorithm in Figure 6.35.

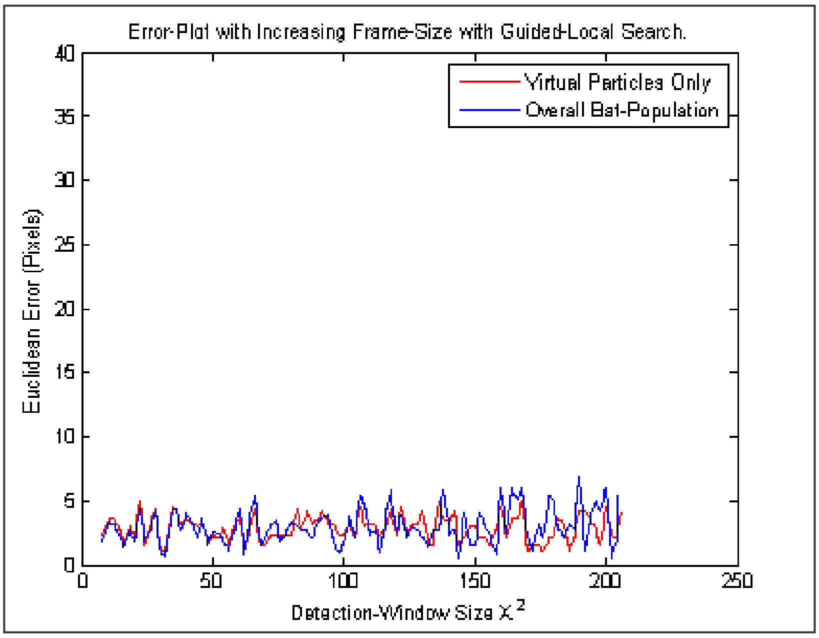


Figure 6.39: Errors are significantly reduced by introducing the guided searches.

Figures 6.36-6.39 were generated by programming a meta-heuristics (VGS) over the standard Bat and PSO algorithms. The core idea of these hybrid tests was to analyse if swarming phenomenon increases the precision. However from the experimental data (Figure 6.39), it was affirmed that the virtual particles (with a guided local heuristic) are capable within themselves to precisely detect and track an OOI (overlapping results in Figure 6.39). Further timing experiments were conducted by varying the population sizes in the VGS algorithm (to maximum permissible window size-$\begin{bmatrix}240\\320\end{bmatrix}$), and the results are presented underneath. It appears that the convergence timings of the VGS algorithm are less prone to the population strengths as displayed in Figure 6.40. The ant was successfully detected in all 205 runs (despite of the clutter) at an average rate of 0.2-0.3 seconds.

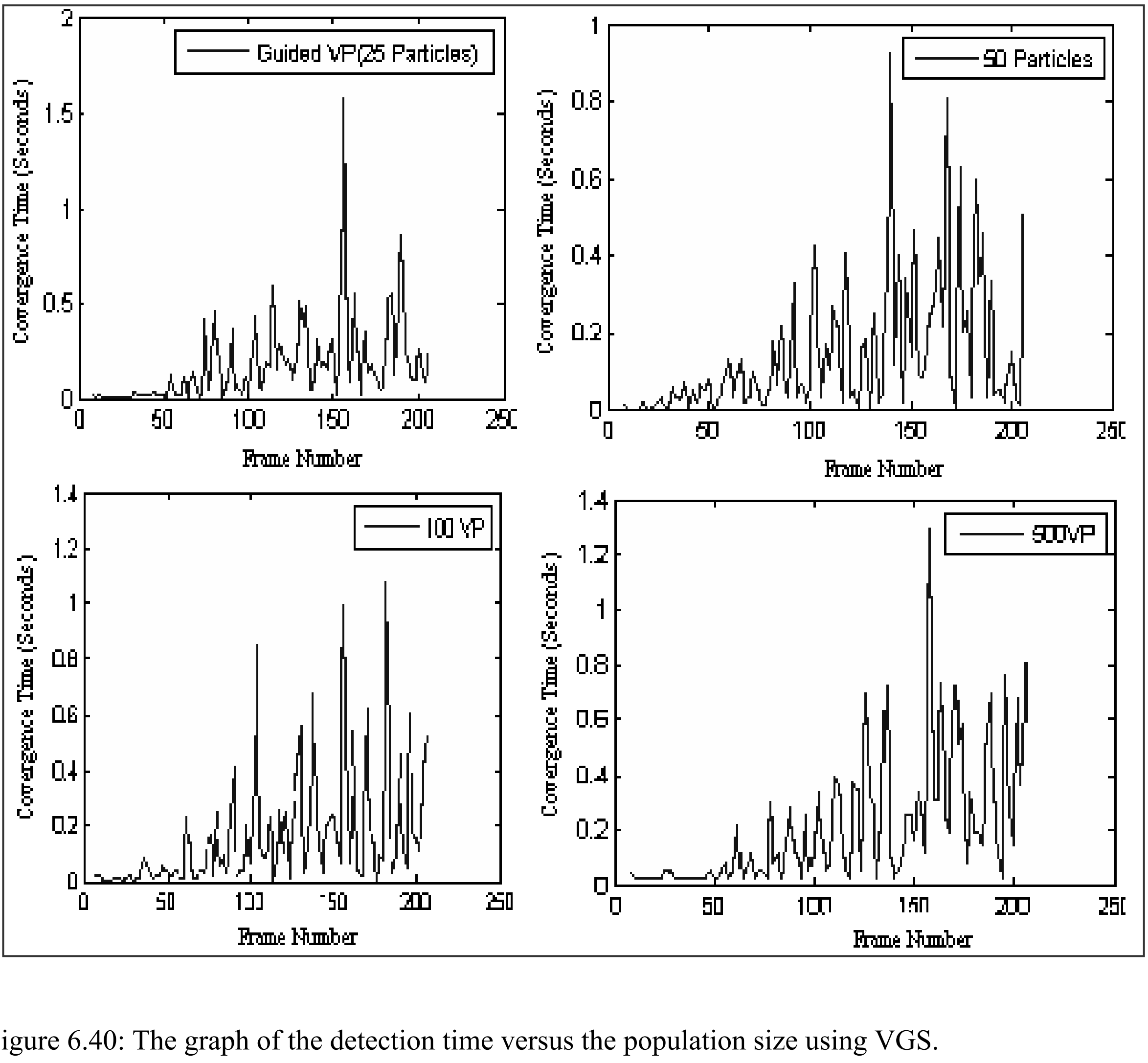
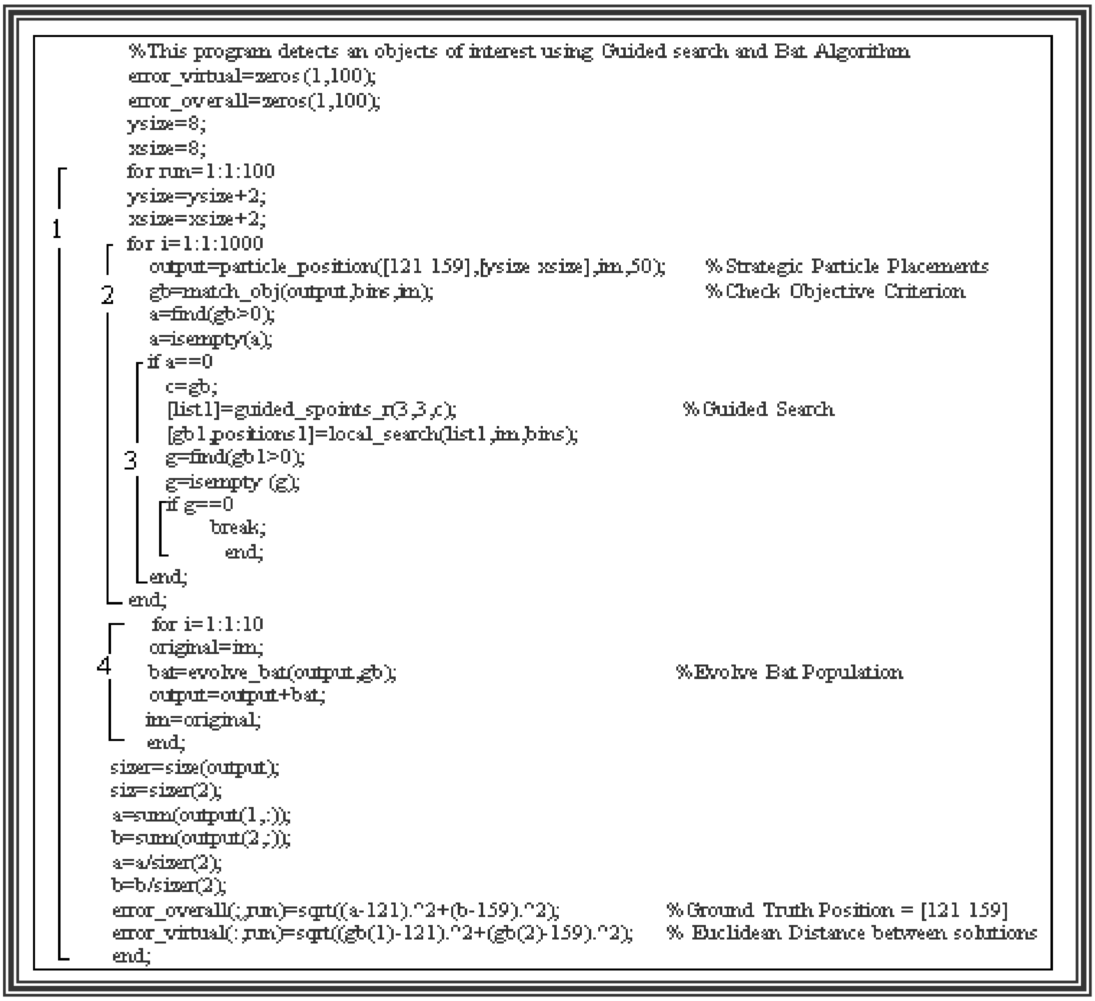


Figure 6.40: The graph of the detection time versus the population size using VGS.

The program structure used in the virtual assisted Bat experiment is partly shown in Figure 6.41. The outer loop-1 is used to increase the windows size, whereas loop-2 strategically places the required number of particles in the search space and calculates the corresponding objective function values. The guided local search is then implemented in the construct-3, whereby, further refinements (nested searches) are sought by reproducing a swarm of Bats in closer vicinity projected by the VGS.



```
%This program detects an objects of interest using Guided search and Bat Algorithm
error_virtual=zeros(1,100);
error_overall=zeros(1,100);
ysize=8;
xsize=8;
for run=1:1:100
ysize=ysize+2;
xsize=xsize+2;
for i=1:1:1000
   output=particle_position([121 159],[ysize xsize],im,50);     % Strategic Particle Placements
   gb=match_obj(output,bins,im);                                % Check Objective Criterion
   a=find(gb>0);
   a=isempty(a);
  if a==0
     c=gb;
     [list1]=guided_spoints_n(3,3,c);                          % Guided Search
     [gb1,positions1]=local_search(list1,im,bins);
     g=find(gb1>0);
     g=isempty (g);
    if g==0
          break;
            end;
   end;
end;
    for i=1:1:10
    original=im;
    bat=evolve_bat(output,gb);                                 %Evolve Bat Population
    output=output+bat;
   im=original;
    end;
sizer=size(output);
siz=sizer(2);
a=sum(output(1,:));
b=sum(output(2,:));
a=a/sizer(2);
b=b/sizer(2);
error_overall(:,run)=sqrt((a-121).^2+(b-159).^2);          % Ground Truth Position = [121 159]
error_virtual(:,run)=sqrt((gb(1)-121).^2+(gb(2)-159).^2);   % Euclidean Distance between solutions
end;
```

Figure 6.41: The VGS assisted Bat detection algorithm.

6.6: Applying RSO and VGS in the Ant tracking sequence.

Two predominantly common problems in the sequential pattern matching algorithms (e.g., vision tracking) are the changes in the object profiles in the feature space, and the exhibition of non-linear dynamics during the video tracking stages, which often translates an object outside the scope of an observational window or operational basin. The aim of the forthcoming sections is to analyse the applicability of both RSO and VGS algorithms in order to regain control of the tracker in challenging scene conditions. If the feature space changes are not timely incorporated into the object model, the tracking algorithm is normally unable to detect the object under observation. Similarly, when the object travels outside the search basin, the tracking algorithm fails to match the known patterns, and therefore, the window generally keeps roaming around in the sub-optimal regions.

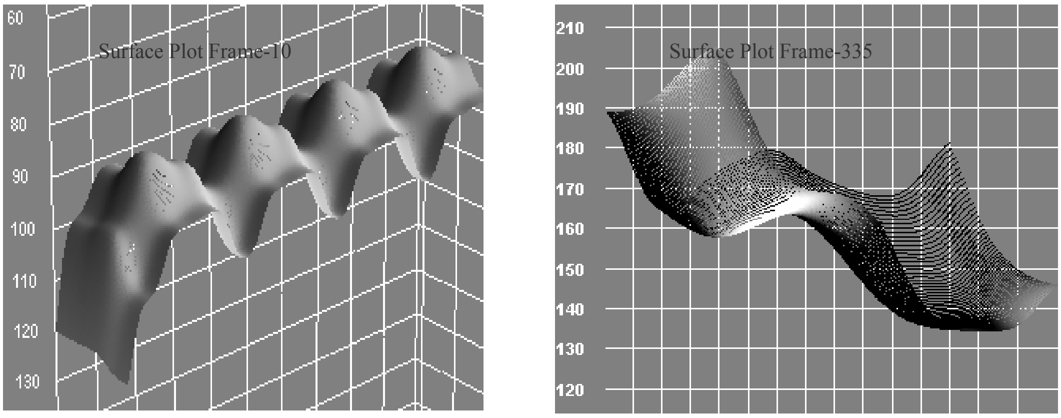


Figure 6.42: The feature space plots of an object of interest during lighting change.

Figure 6.42 shows the effects of lighting on the object model during two different frames of the sequence (Section 5.2.1). The 3d surface plots of the normalised colour intensities show the extent of the non-linear shifts which render the objects to an extent that they become undetectable in frames. One solution to the problem displayed in Figure 6.42 is to provide motion assistances using scale space differences (see Figure 6.6) until the original conditions have returned.

In cases of permanent shifts the only resilient solution (to address the dynamic problem) is to remodel the objects in the feature space. Optimistically speaking, the VGS and RSO (Sections 3.4, 4.1, 6.4) based detections would enable us to address any unpredictable movement (due to a scale free approach) during manoeuvrability phases, and these scenarios will be further tested in the coming experiments. Figure 6.43 shows the sequence of these demonstrations. The experiments start by allocating the dynamical windows in Section 6.6, and a final conclusive stage (Section 6.10) would help to stipulate the applicability of the tuning free approach (adapted in this thesis), and the aim is to test the performance of the novel RSO-VGS algorithm.

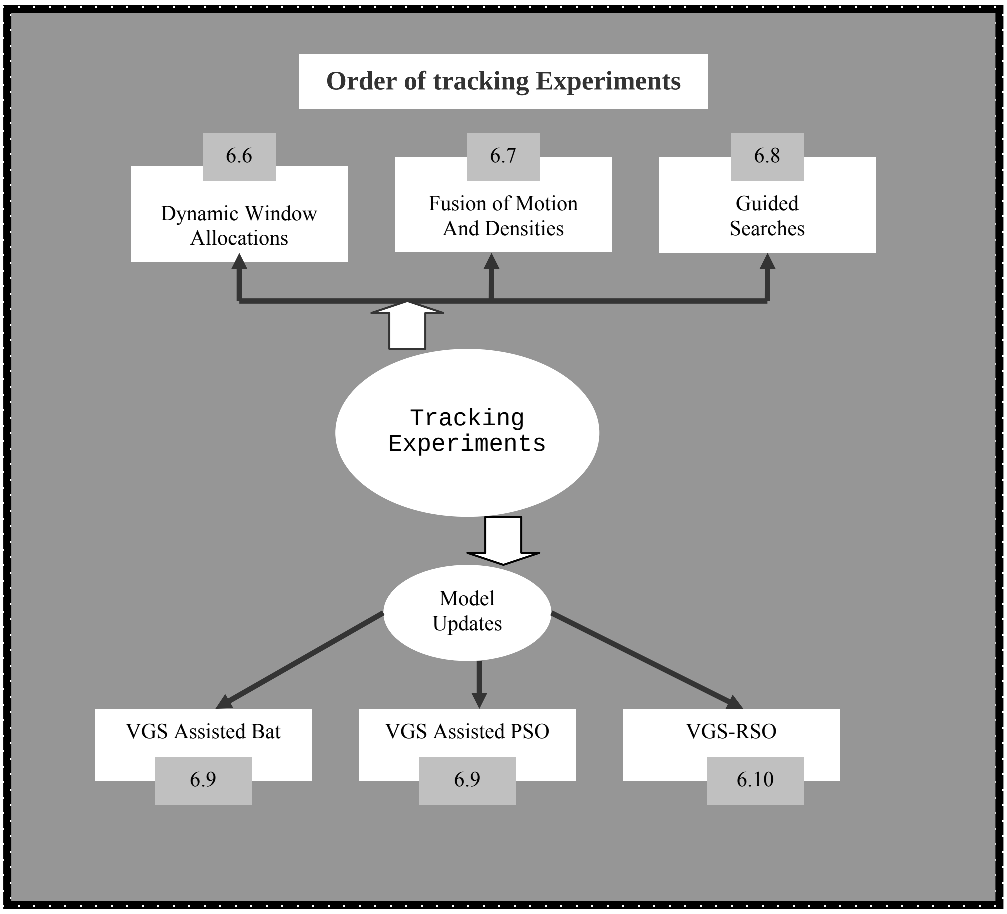


Figure 6.43: The sequence of the tracking experimentations in coming sections.

6.7: Tracking using the dynamically changing window size.

In this experiment we applied the standard Bat algorithm to track an object of interest. The core idea in this approach is to alter the window sizes in order to regain control of the tracking window during the lost frames. In the first experiment, a window size of $\begin{bmatrix} 10 \\ 10 \end{bmatrix}$ pixels is selected to track an ant in a maze as shown in Figure 6.44. Figure 6.44 reveals that once the tracking window was lost in the frame (205th frame), the tracker was never able to recover, and the window roams around in the search space. The errors $\begin{bmatrix} E_x \\ E_y \end{bmatrix}$ (Figure 6.45) are plotted for this video sequence in Figure 6.44.

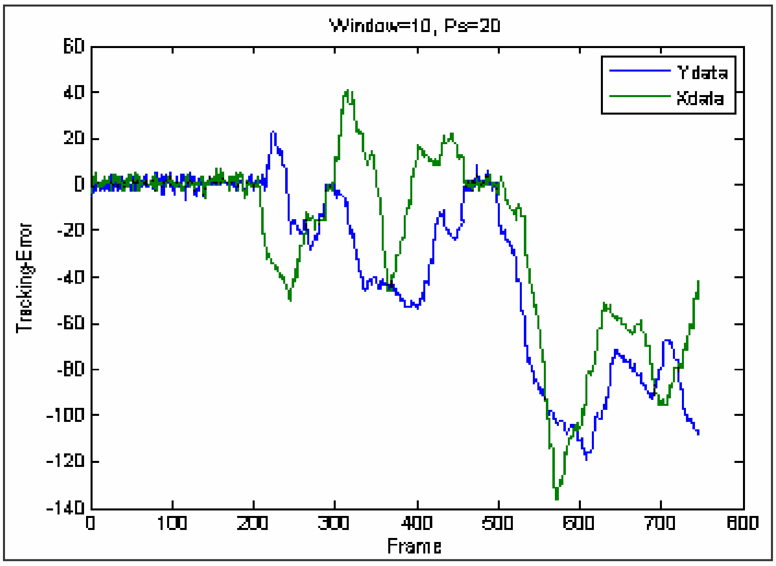


Figure 6.44: The errors E_x and E_y are plotted for the 774 frames.

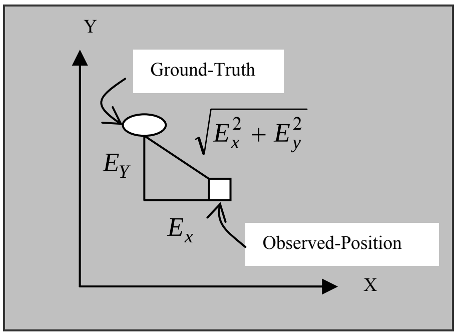


Figure 6.45: The calculation of the Euclidean error distance VS the errors E_x and E_y.

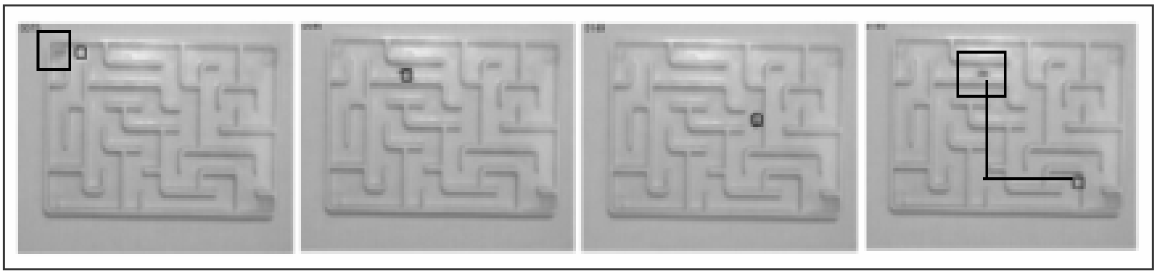


Figure 6.46: The pictorial view of the tracking errors E_x and E_y in four imaging frames.

Some frames from the tracking sequence are presented in Figure 6.46. The larger squares are used to identify the true positions (ground truth) of the ROI, whereas, the smaller window replicate a tracked positions along the 2 dimensional (XY) space. In the next experiment, we have altered the swarm population size from $p_s = 50$ to $p_s = 100$, and the tracking results are plotted in Figure 6.47. Remarkably, the tracking window successfully recovered after 417th frame in this sequence. The 2 dominant error phases are plotted in Figures 6.47-6.48. In case of a window size $W_s = \begin{bmatrix} 20 \\ 20 \end{bmatrix}$, the error plots in Figure 6.47 display a robust recovery.

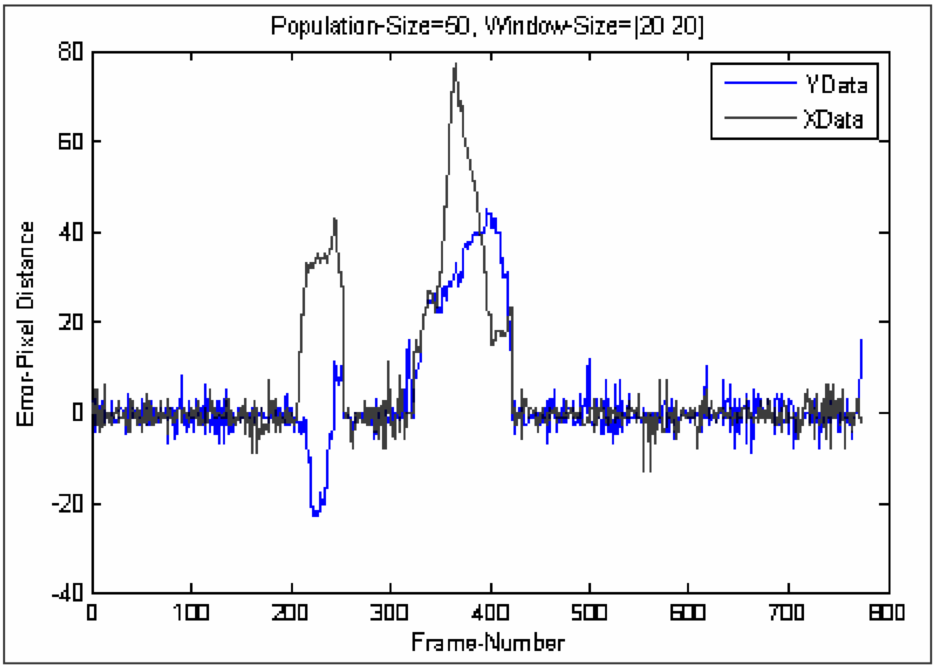


Figure 6.47: The object tracking with a window size $\begin{bmatrix} 20 \\ 20 \end{bmatrix}$ and swarm of 50 BATS.

Finally, the tracking window was adjusted to $w_s = \begin{bmatrix} 50 \\ 50 \end{bmatrix}$ with $p_s = 100$ bats, and Figure 6.48 shows the tracking errors observed in this particular detection situation. This experiment shows the scale of the errors when a tracking window is widened in order to recover a lost position (Section 6.4). The problem has also been identified in Figure 6.48, which exhibits that the tracker converged to the non-optimal regions/shadows. The tracker has never been successful in locating a ROI (as in Figure 6.49).

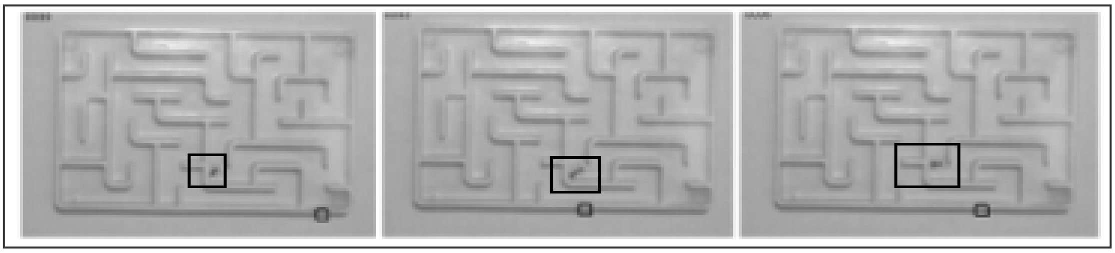


Figure 6.48: The pictorial/graphical view 3-tracking frames.

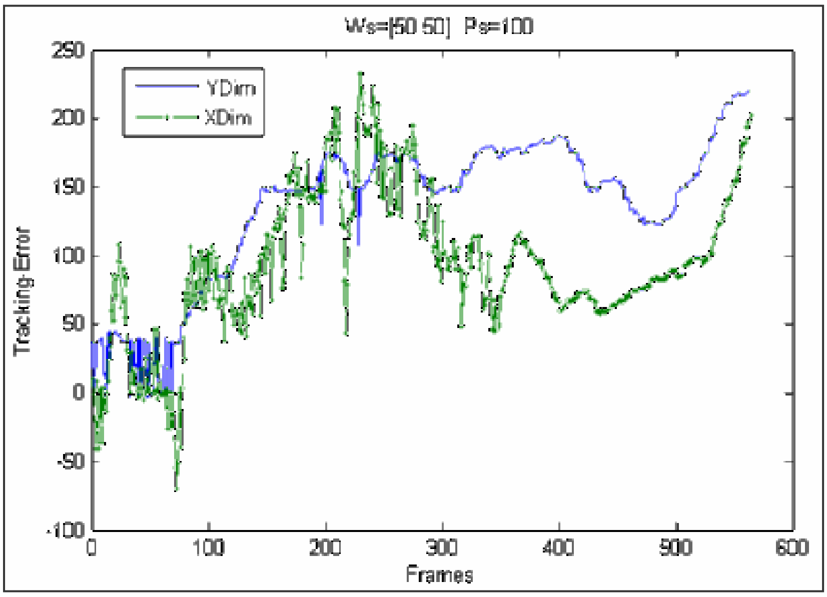


Figure 6.49: The object tracking with a window size $\begin{bmatrix} 50 \\ 50 \end{bmatrix}$ and a swarm of 100 BATS.

6.8 Tracking using state vector fusions.

One way to resolve the model related problem (exemplified in Figure 6.42) is to fuse the state vectors of parallel processes with an aim to increase the tracker robustness. The undeterminable changes in the object models could be addressed by collating the motion and density vectors. The opinion polling mechanism [271] was used in this experiment to integrate location based information. Both Euclidean and x-y errors are plotted in Figure 6.51. The performance enhancement is evident when the results are compared to Figure 6.49. The performance of the tracker is somewhat marginalised by a low mean frame rate of 21.73 FPS. The stability of the tracker during adverse lighting phases could be confirmed in Figure 6.50 as well.

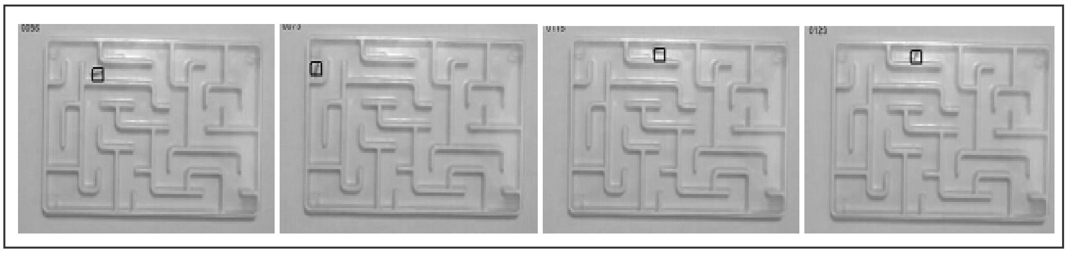


Figure 6.50: ROI was successfully tracked in the region of large illumination changes.

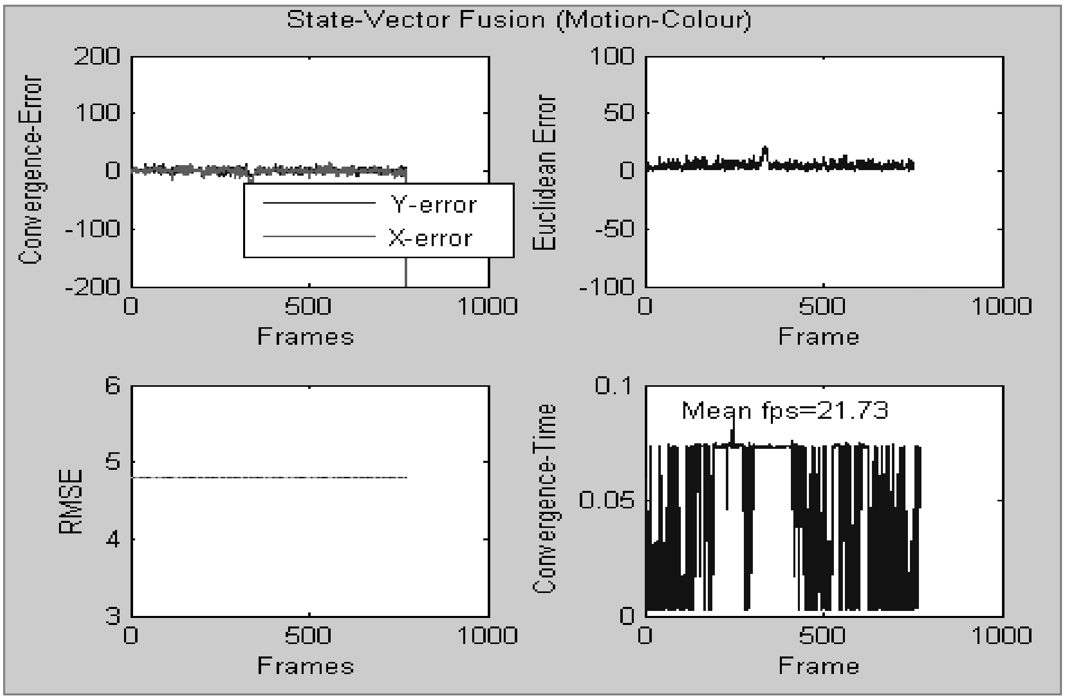


Figure 6.51: The fusion of colour and motion models increases stability in the ant tracking.

6.9 Tracking using penalising approaches.

The virtual assisted PSO (PSO-VA) and the virtually assisted Bat (Bat-VA) are novel meta-heuristic algorithms devised to entail optimal convergence of particles. The RSO performs random searches in the search space, and the guided search (Sections 4.2-4.3) is applied once the object characteristics are discovered. During the next stages, particle rebirths (see Figure 3.36) take place and both PSO and Bat convergences are used to refine the object locations. The x-y tracking errors are plotted in Figures 6.51 and 6.52. Both Bat and virtually assisted PSO recovered from seven major illumination changes as plotted in Figures 6.52 and 6.53.

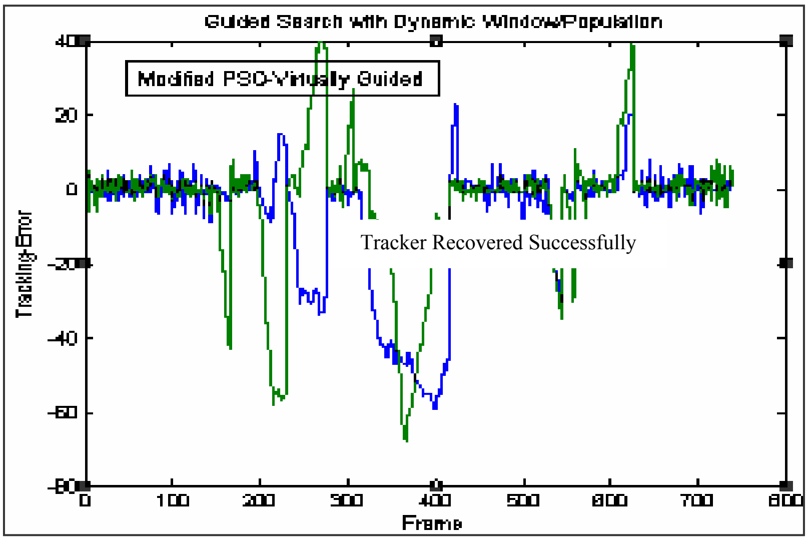


Figure 6.52: The performance of the virtual assisted PSO in the ant tracking sequence.

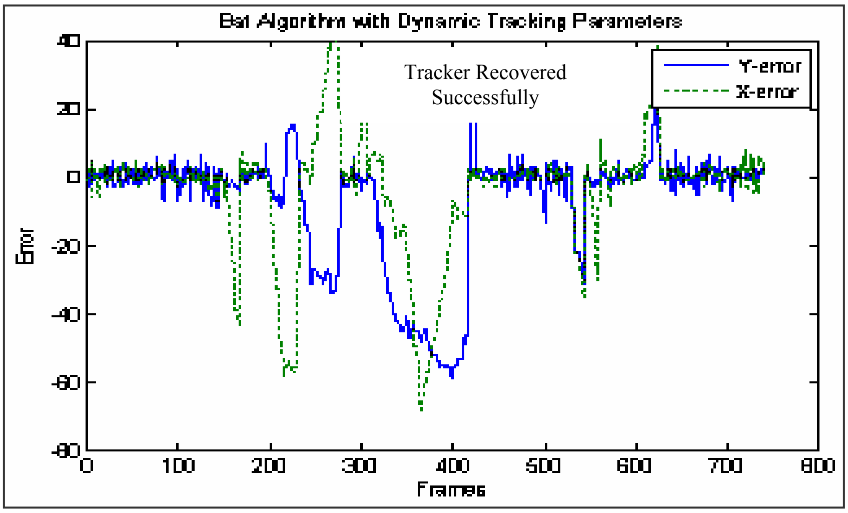


Figure 6.53: The performance of the virtual assisted Bats in detecting an ant trapped in a maze.

6.10 Tracking using Model-Updates and VGS as a metaheuristic.

The population based methods generally lack an ability to penalise clutter, and therefore due to an absence of a guiding procedure, the process suffers from a tendency to converge towards local optimal regions. The radical search optimization (RSO) (Section 3.4) is a novel scale free optimization methodology which is more effective than the particle swarms (Section 3.3). Moreover, the virtual guided search (VGS) is a memory free particle hierarchy (Figure 6.24) that enables the agents (conducting RSO) to perform detailed local searches and penalise the non-optimal regions. Figure 6.54 exhibits the scale of changes in the RGB space experienced in two consecutive frames. The underline theme behind the model updates is to reconstruct the feature profiles at the moment an adversity is found in a video frame. In this section, the profile updates will be applied using PSO-VA and Bat-VA.

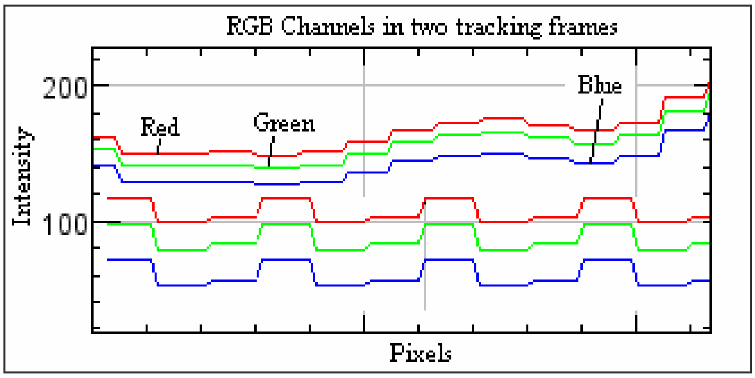


Figure 6.54: The discrepancies in the RGB colour channels of the object profile.

The results of the tracking by subjecting the PSO-VA and Bat-VA along with the feature updates are shown in Figures 6.55 and 6.56. Both algorithms maintained good tracks of the moving object, and the resulting plots show that the convergence timing was also significantly increased from 21.73fps (Figure 6.51) to 50fps in the PSO-VA case with RMSE=6.9pixels. The Bat-VA was comparatively more effective and frames were tracked at a rate of 66fps (Figure 6.56) with reduction in the errors (RMSE=4.42 pixels).

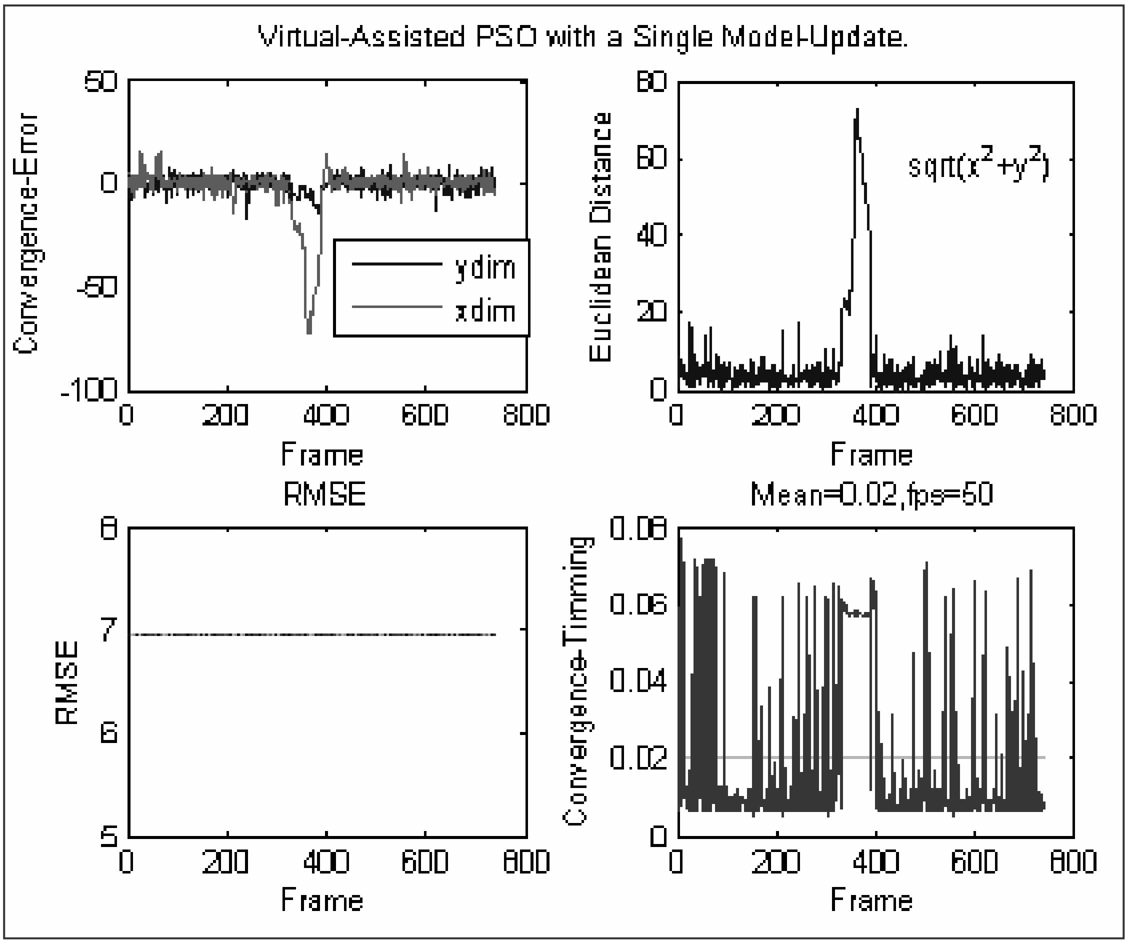


Figure 6.55: Frame rate of 50fps and a RMSE of 6.9 pixels were observed in ant sequence.

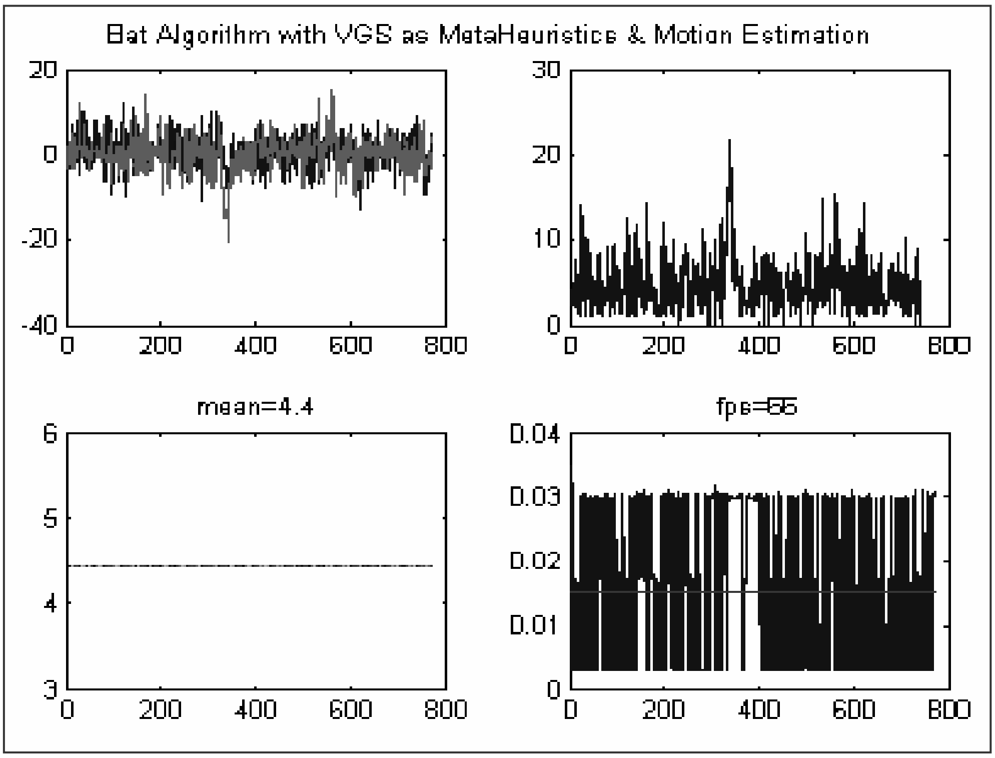


Figure 6.56: BAT algorithm using virtual guided search as meta-heuristics and bin updates.

6.11 Tracking using only VGS with Model-Update.

The aim of this experiment is to analyse both the tracking accuracy and the convergence timing using VGS and RSO routines without swarm deployments. The performance plots in Figure 6.57 highlights the fact that the errors in both x-y dimensions are significantly reduced with a mean error vector $\begin{bmatrix} 2.58 \\ 4.23 \end{bmatrix}$. The illumination variations have been compensated by rebuilding the feature profile during adverse frames. Compared to Figure 6.55, the errors peaks between 335-373 frames are therefore less significant. A mean convergence time of 0.004 seconds was recorded in Figure 6.58. Therefore the RSA-VGS produced significantly accurate results at 250fps in this vision tracking sequence. The performance of RSO is also in line with the results obtained in optimising the evolutionary test cases (Figures 3.63-3.66). A number of frames (32) have been selected and the tracking results are displayed in pictorial formats in Figure 6.59.

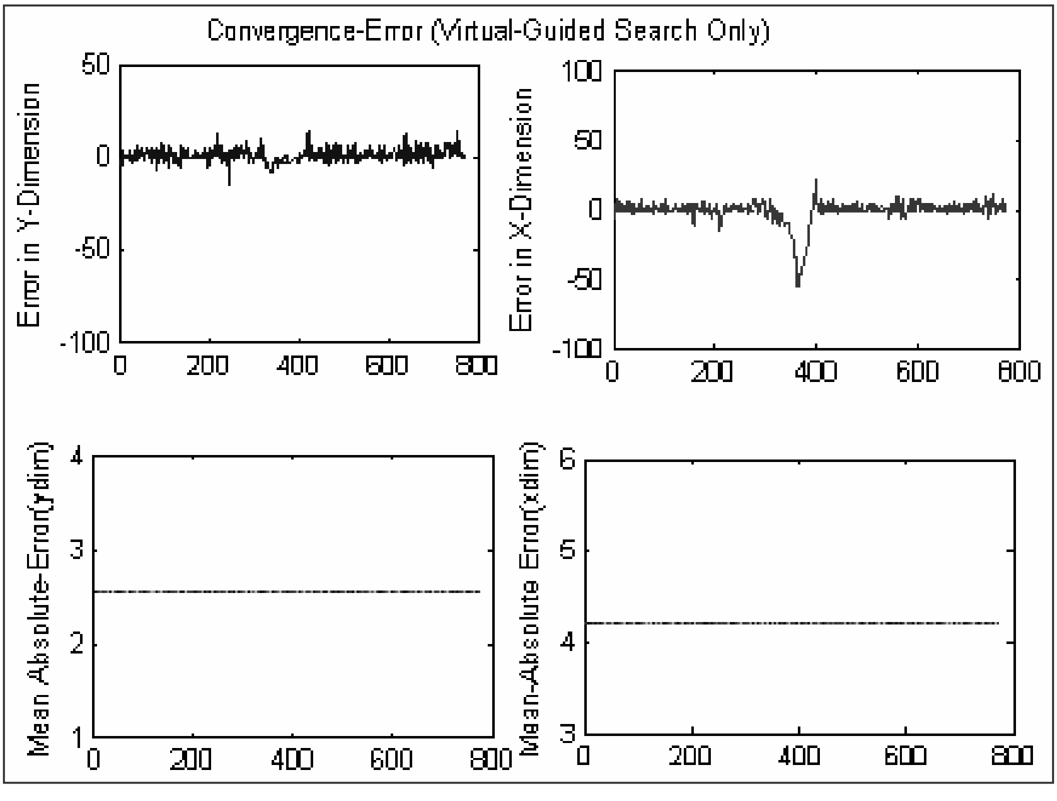


Figure 6.57: VGS tracking was performed with fewer errors than the PSO-VA and Bat-VA.

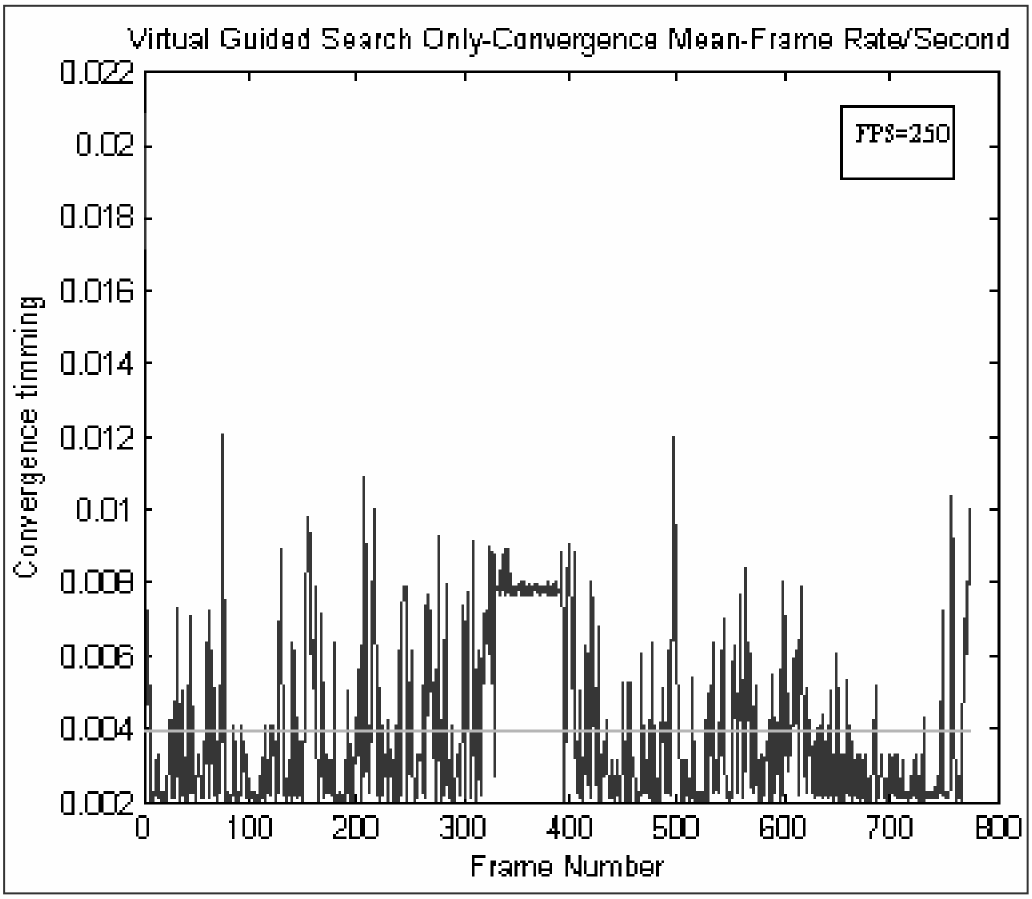


Figure 6.58: The VGS algorithm managed to attain a mean FPS=250, VGS/RSO has outclassed both BAT and PSO (inline with the findings presented in Section 3.5) by huge margins. This is partly due to the fact that no record of velocity and positions is kept as explained in Section 6.3.

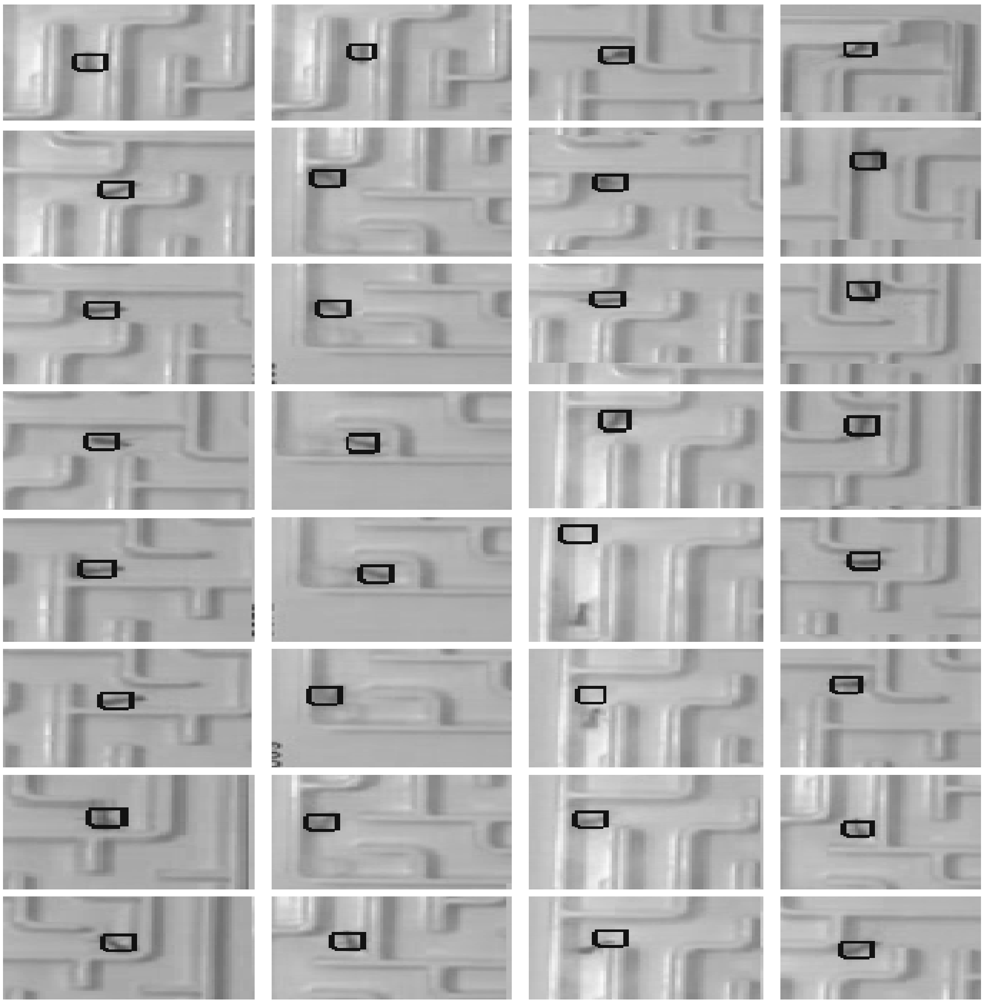


Figure 6.59: Tracking ant sequence at a frame rate of between 200-250 using VGS alone, the algorithm quickly regained control after the lost frames using RSO heuristics.

6.12 Scale free searches in the Drone Sequence.

The motion modelling using kinematics (Sections 2.1-2.4) has been a popular methodology for many years in the video tracking problems. However, one of the key adversities (that standard tracking algorithms based on motion modelling have been unable to address) is the partial and complete occlusions of an object of interest. When an object becomes undetectable in an observational window for several frames, the tracking algorithm generally lags behind the true position and the detection fails. Figure 6.60 shows this problem graphically, the state of any tracking application comprises of the window size parameter$\begin{bmatrix} w_y \\ w_x \end{bmatrix}$, and its mean position$\begin{bmatrix} m_y \\ m_x \end{bmatrix}$. Therefore the aim of tracking is to minimise the distance d_{xy} between the mean window position and the object centroid. The role of the RSO particles is to provide the required loose control based on the Bernsteinian control theory (Figure 2.9), and later on, the strict disciplinary nature of the VGS projects a much tighter control to efficiently detect and track an object.

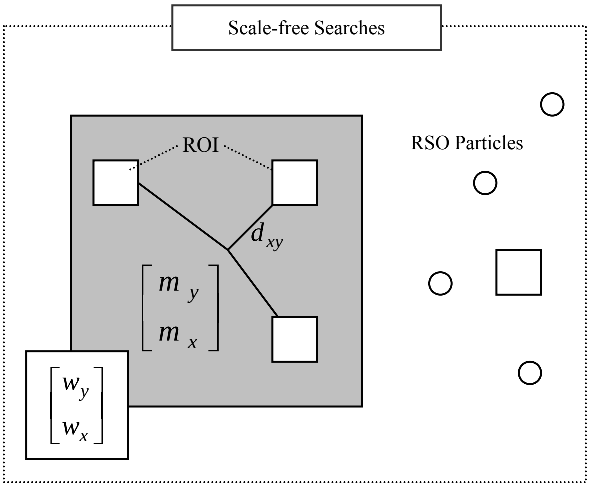

Scale-free Searches
ROI
RSO Particles
d_{xy}
$\begin{bmatrix} m_y \\ m_x \end{bmatrix}$
$\begin{bmatrix} w_y \\ w_x \end{bmatrix}$

Figure 6.60: The scale free searches using RSO

We applied the scale free searches based on RSO and VGS to the drone sequence, the severity of the problem with dynamical windows have been identified in Figure 6.61. The enlargement of the tracking window (in this test) provides an ideal platform to study the detection efficiency of the RSO-VGS duo. In these experiments the window size will be varied from $\begin{bmatrix}10\\10\end{bmatrix}$ to $\begin{bmatrix}300\\300\end{bmatrix}$, and the penalising VGS approach will be applied to distinguish between the low cloud and drone.

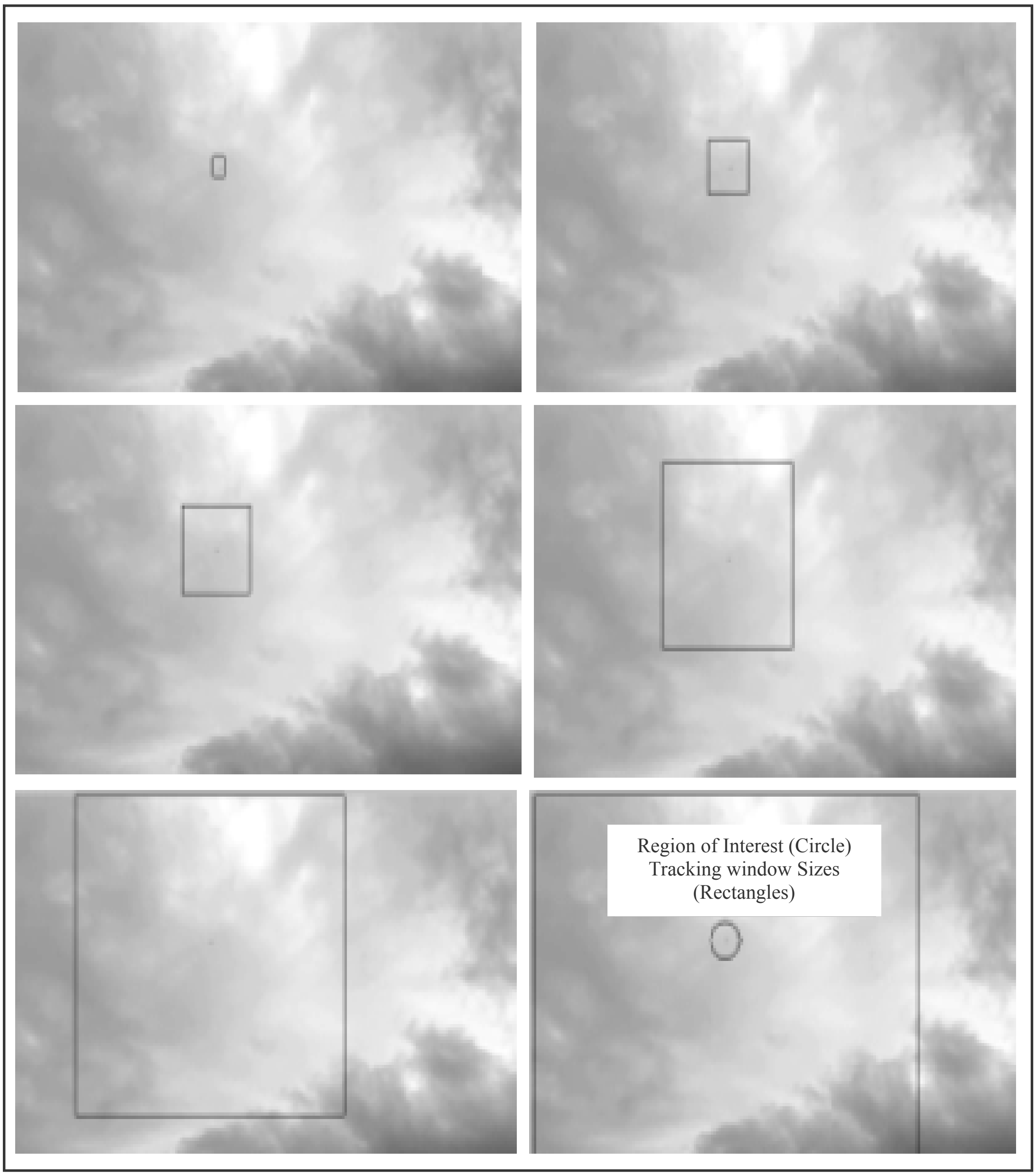


Figure 6.61: The scale of the search windows in these experiments compared to the ROI.

The tracking window size is varied as shown in Figures 6.59 and 6.60, both the detection errors and the convergence timings are plotted for each case.

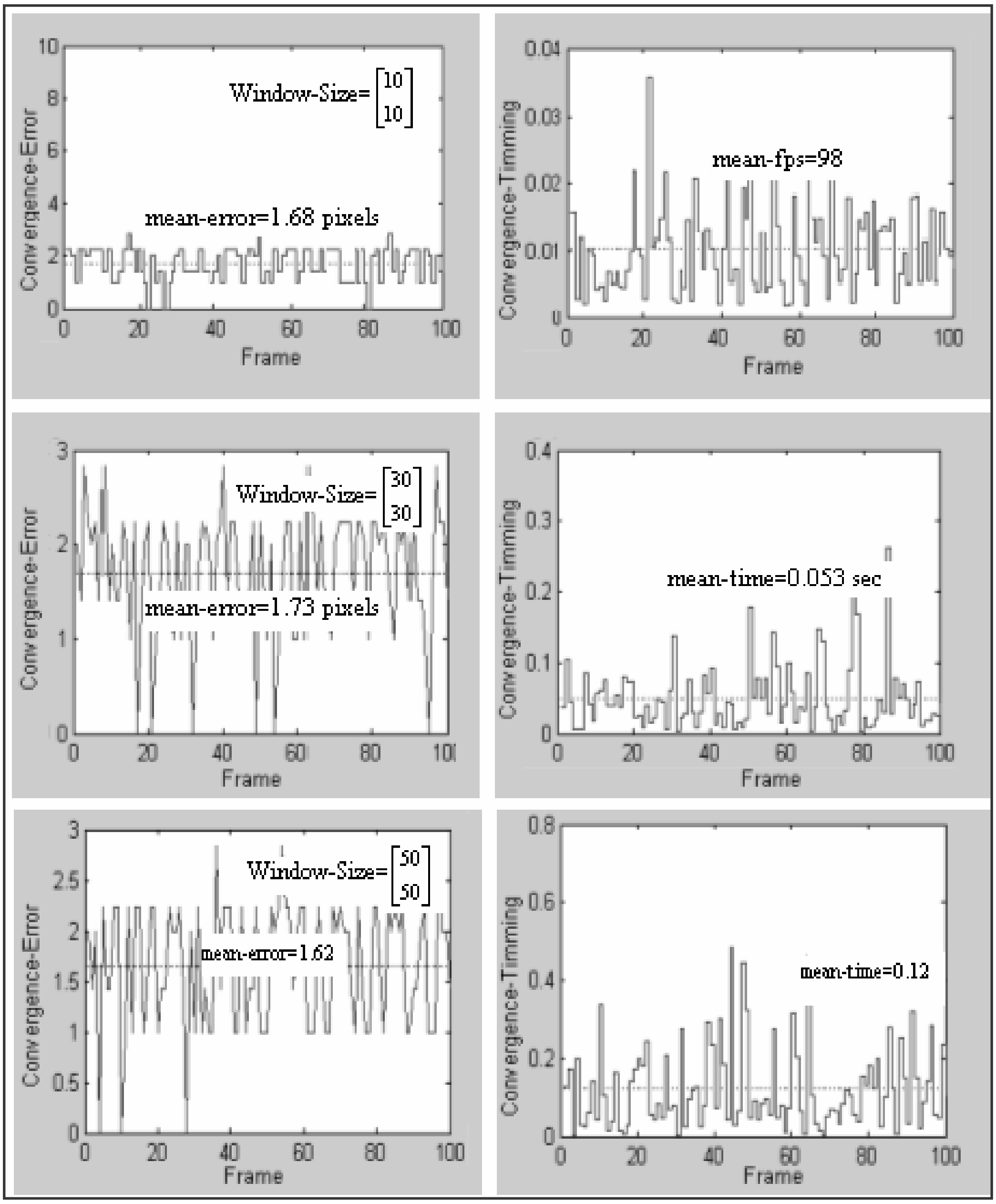


Figure 6.62: Effects of the window size selection $\begin{bmatrix}10\\10\end{bmatrix}$ to $\begin{bmatrix}50\\50\end{bmatrix}$ onto the frame/detection rates and the time needed to converge.

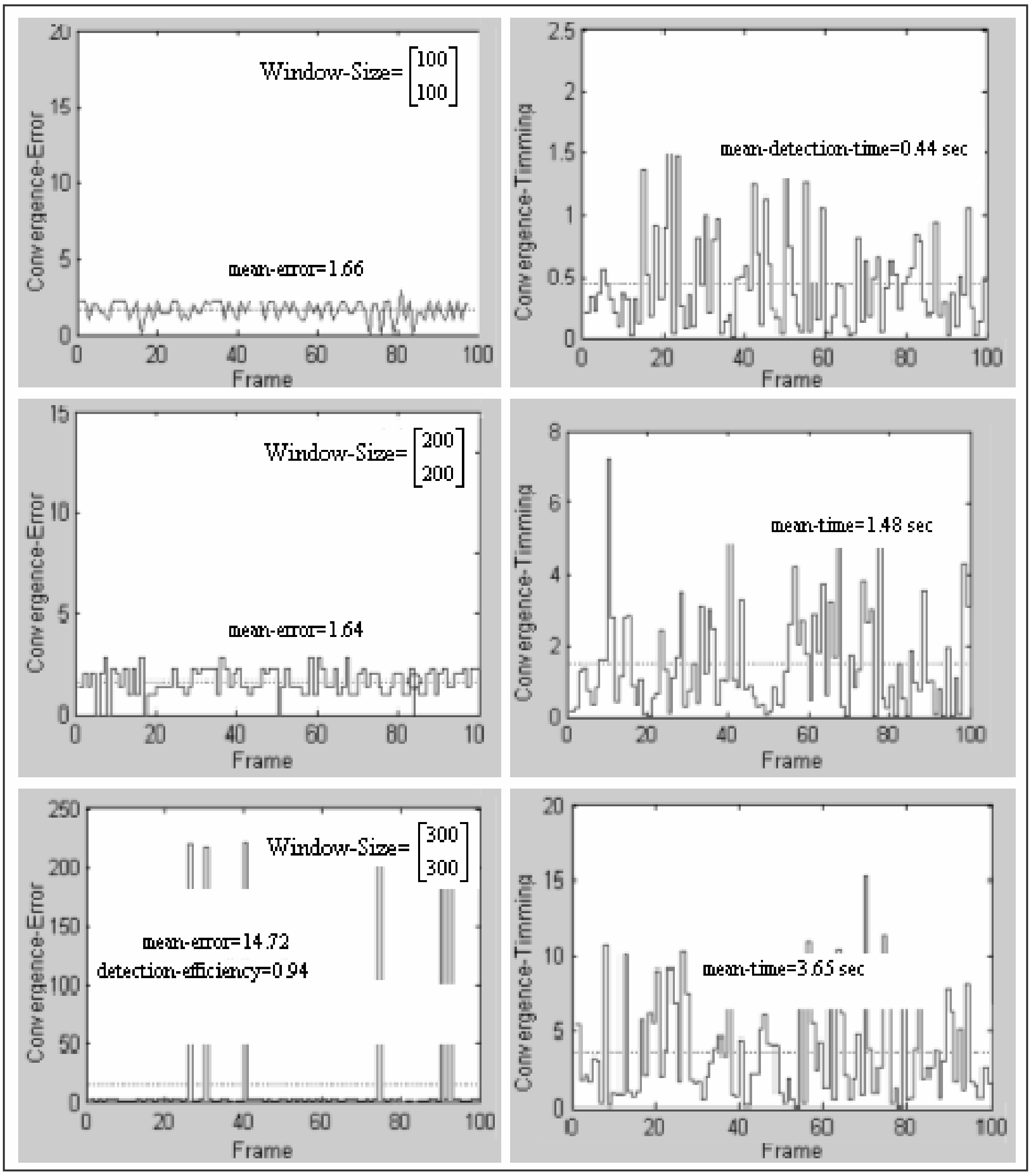


Figure 6.63: Effects of window size selections $\begin{bmatrix}100\\100\end{bmatrix}$-$\begin{bmatrix}300\\300\end{bmatrix}$ onto the detection errors and the convergence timings (drone).

In Figures 6.62 and 6.63 the mean convergence rate is recorded to be between 0.01-3.65 seconds when the window size was varied between 100-90k pixels. The detection efficiency of 0.94 with a window size of $\begin{bmatrix}300\\300\end{bmatrix}$ shows the stability of the proposed RSO-VGS methodology. The detection (after occlusion) in the largest window (Figure 6.63) took only 3.65 seconds to converge to the optimal. A real time convergence rate of 98fps was observed with a $\begin{bmatrix}10\\10\end{bmatrix}$ window size. Further fascinating factor in these tests was that the experiments were performed with only $N_p = 3$ particles that simulated virtual swarms in the search space

To analyse the effects of particle strengths in VGS tracking, a population sweep test was conducted $3 \le N_p \le 203$ with a window size of $\begin{bmatrix}100\\100\end{bmatrix}$, and the convergence results are presented in Figure 6.64. From the shape of error plots we can conclude that the accuracy of the RSO-VGS trackers is also less dependent on the population strengths. However the frame rate peaked when 150-203 particles are deployed in this experiment.

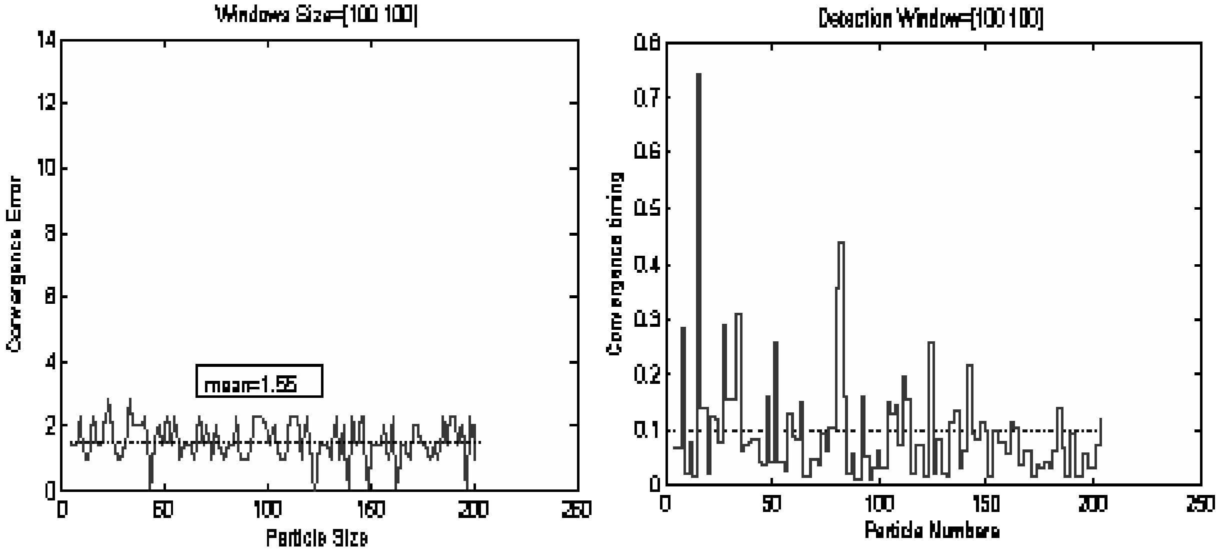


Figure 6.64: Effects of the particle sizes implying the VGS-RSO tracking in the drone Sequence.

The tracking results with variable window selections are drawn in Figure 6.62. The algorithm dynamically changes complexity by evaluating the scene using the VGS. If the pattern matching fails in an image, the algorithm dynamically alters the windows size to incorporate wider RSO searches. The drone was successfully tracked at an average rate of 35fps, and some tracking frames are presented in Figure 6.65. On a particular occasion the drone got occluded during several frames by a low cloud, and it recovered from this complete occlusion in less than 2.44 seconds at a later stage by using the RSO.

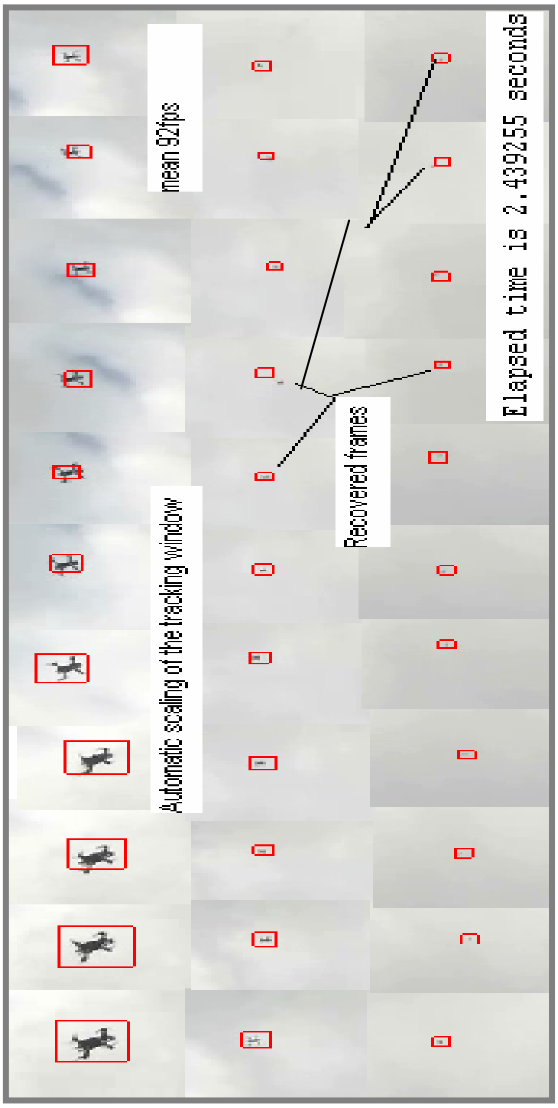

mean 92fps
Automatic scaling of the tracking window
Recovered frames
Elapsed time is 2.439255 seconds

Figure 6.65: Tracking of the drone sequence using auto-varying window sizes.

6.13 Comparison with Mean Shift Tracking.

In this section the meanshift (MS) tracking was implemented to track a pedestrian as shown in Figure 6.66. Due to the intensive nature of the meanshift searches (which usually takes place in a fixed operational basin) it only managed a maximum tracking rate of only 21FPS in pedestrian tracking sequence at a mean convergence error of 1.34 pixels.

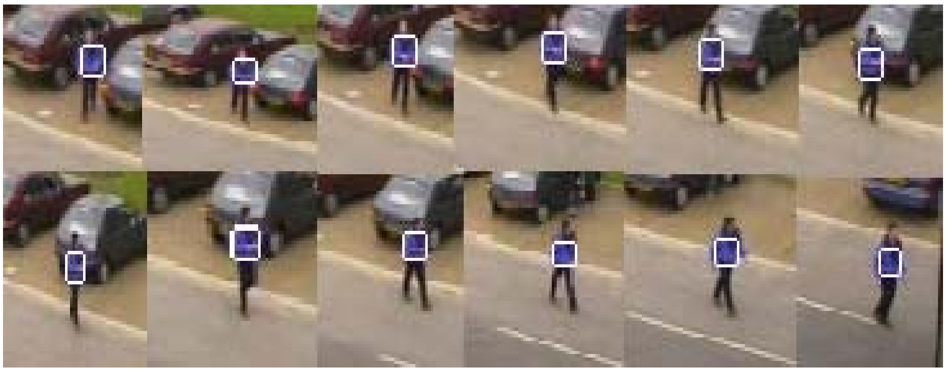


Figure 6.66: Tracking in pedestrian sequence using uniform meanshift kernel.

The application of MS in the ant tracking sequence (Figure 6.67) managed a frame rate of only 17FPS in Figure 6.66, and the convergence errors are averaged out and found to be in the vicinity of 1.23 pixels (using the Euclidean distance norm). When both MS and motion estimation were utilised, the convergence timing improved to a mean value of 23FPS (Figure 6.67) which is still way less than what we achieved using the RSO-VGS algorithm in earlier analysis performed in Chapter 6.

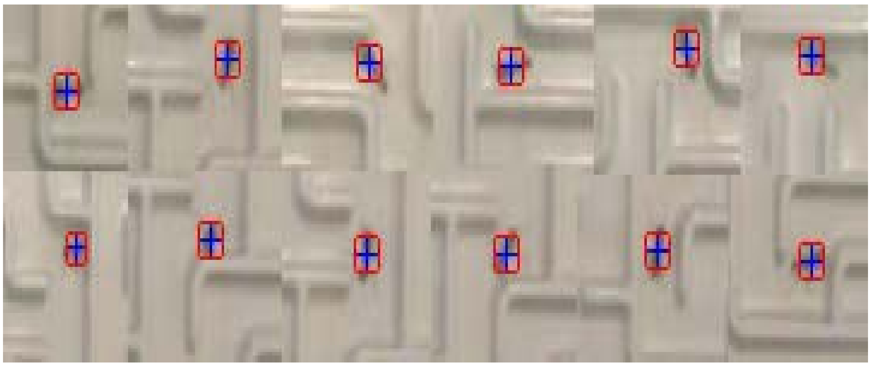


Figure 6.67: Tracking ant using MS at a frame rate of 17 and RMSE of 1.2 pixels.

6.14 Tracking Highway Sequence.

Several vehicles were also accurately tracked using the RSO-VGS methodology proposed in Chapters 3-6. The results approved that our tracking methodology is suitable for a variety of detection and tracking scenarios. In the top sequence various motor vehicles were correctly identified and tracked, whereby a green van was tracked at a rate of 60FPS with a mean pixel discrepancy of 2.13. The RSO tracking was also successfully applied to a synthetic motorway sequence (Figure 6.68).

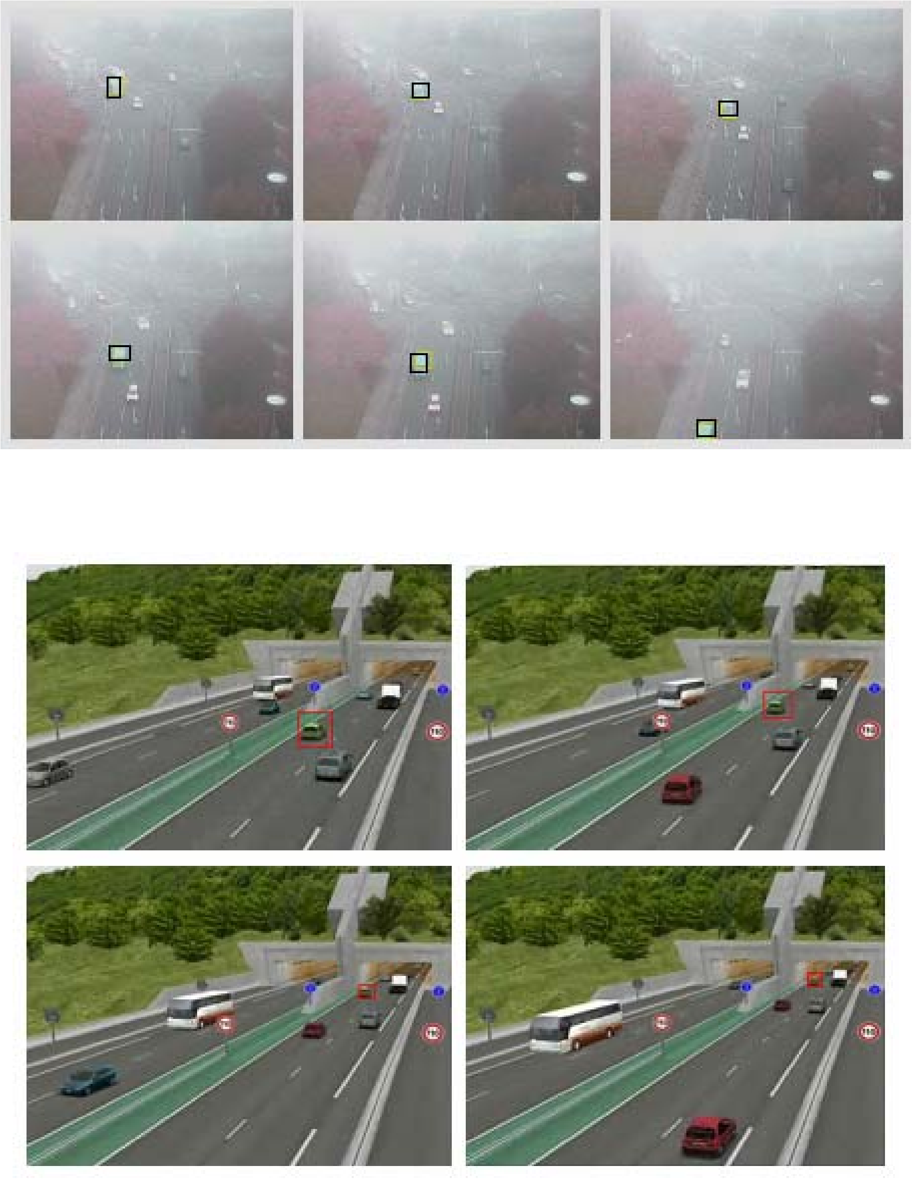


Figure 6.68: Real time tracking in Highway Sequences at frame rates of 57-70.

6.15 Results of some further Experiments.

To establish further applicability of the VGS-RSO, several experiments were conducted in order to track and detect a variety of objects. Detection test precisely located various balls in the juggling sequence. Tracking algorithms were applied to a video game sequence (in Figure 6.69) achieving real time performances of 17-31fps. The detection experiments (50 runs) were carried out to identify various regions in the cartoon images (Figure 6.70) achieving detection times of between 0.013-0.023 seconds. Finally, the partial occlusion was successfully resolved in the walking man video (Figure 6.71). Hence we conclude safely that our novel tuning free particle method is widely applicable.

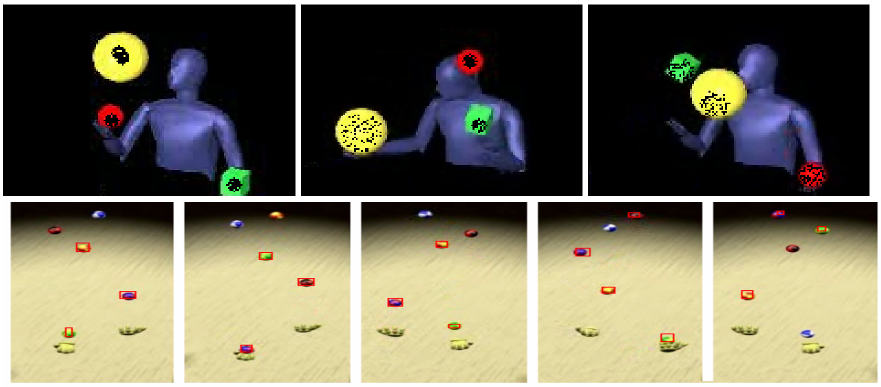


Figure 6.69: Multiple object tracking in synthetic sequences using bat and virtual particles.

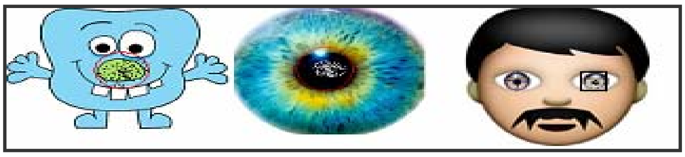
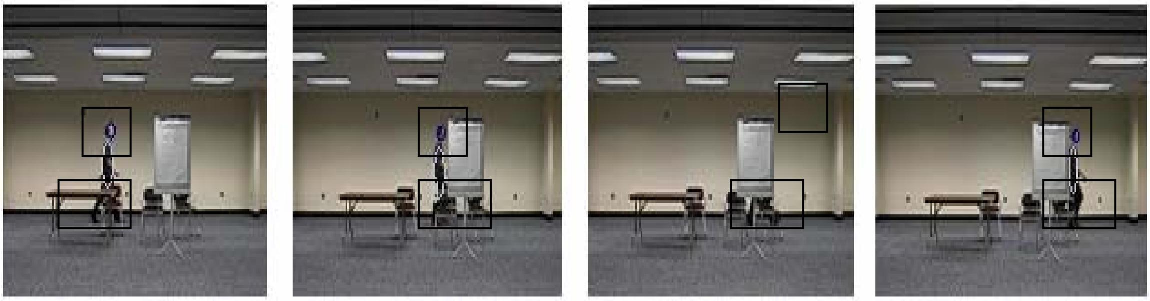


Figure 6.70: Applying visual detection to various cartoon images (converged particles drawn).



.Figure 6.71: Detection of Walking Man after phases of occlusion was robust and accurate direction.

Chapter 7

Conclusions

Automatic tracking is a fundamental branch of vision science and it deals with the identification of a region of interest, and inferences are made to determine the concealed knowledge (e.g., motion patterns) on the sequential arrivals of digital images. To meet the challenges and demands of developing a real time tracking algorithm, author had to deal with a number of misconceptions basically stemming from the technological limitations of the past century. Many common linear estimation processes and relative stochastic controllers (an era beginning from the renowned work of Rudolf Emil Kalman [28]) generally tend to ignore the historical inferences (as integrated memory was reckoned to be a luxury during that time) in order to gain an insight into the future behaviour of a dynamic system. Today, the human generation is rocketing towards a new technological phase, and the role of self evolving systems and collaborative intelligence (among entities) is becoming much more apparent, therefore paving further a path to self correction and towards an embedded reconfiguration.

7.1 Research Background

Like many real world agents (birds, humans, wild animals and insects), a particle is a basic computational agent that is deployed to solve computational problems arising when an exact solution does not exist in reality (Sections 1.1-1.3). Scientists are intrigued to designate versatile behaviours to the basic search element, and the particle roles are generally depicted from the natural life observations. The literature survey conducted in this thesis spanned from the correlative and mesmerising flypasts of the murmuring starlings to the bioluminescence among mating fireflies. However, one key aspect that captivated our attention (we may call it a discrepancy), and comes after the realization that a lot of (so called) optimisation scenarios are in fact just simulations of the natural agents, and exhibit their changing social behaviours in a natural world in order to deal with the calamities. In view of this research, a futuristic strategy and approach should rectify the core misconceptions in order to develop faster and efficient optimisation routines. For a new generation of scientists and engineers, a clearer insight into this problem is important, and there is an imperative need to differentiate among the particle roles in both simulative and optimisation scenarios.

This thesis developed a practical approach to devise some novel optimisation strategies that are tested in both evolutionary and imaging test cases. From a broad range of experimentations conducted in this thesis, it has come to our attention that tracking applications are extremely sensitive to the parametric selections in regards to the discovery of solutions that satisfy both space and time constraints. Particularly, in the work of Kennedy and Eberhart [146], the right parametric settings in Equation (3.5) are extremely difficult to be handled in live real time video scenarios; this effect is studied in detail and demonstrated in relation to Figures 3.55, 6.31 for solving rigorous evolutionary test cases.

7.2 Methodology

Along with the functional level difficulties witnessed in general optimisation algorithms, many optimisation contingencies arise particularly from a grass root level in the evolutionary branch of mathematics. During the course of this research, a meticulous comparative analysis is performed in Chapter 3 (Figures 3.3-3.17), and the underline aim was to alleviate the convergence issues resulting from the empathetic social velocity vectors in multi-dimensional test bench problems. It is later learnt through experiments that the pragmatic initializations (e.g., by using multiple solutions) could address the elementary challenges encountered when unfamiliar/online optimisation scenarios are being handled. Furthermore, for an optimal and timely convergence, a broader diversity is needed in the imaging algorithms. The gradient based approaches (despite of their inbuilt reassurance to find at least a local optimal solution) are unreliable techniques that usually prohibit timely discovery of a correct locus in multi-modal visual systems (Figures 1.3 and 3.23). The particle injections into discerned image areas, and consequently, the application of systematic perturbations (within search space boundaries) have much greater tendency to solve vision problems in a robust fashion. Hence, the deterministic shifts using proliferating modelling in analytical mathematics has very limited role in the computer vision applications, as the problem at hand is more of a non-linear in nature. In order to further reduce the computational complexity, the swarming characteristic is impersonated by deployments of a limited particle population. An optimal particle based technique therefore is embedded through the virtual swarming illusion in which particles flew repeatedly (in the feasible space) reducing a population size.

The computational equivalence could be established by investigating the search problem in the light of no free lunch theorem of Macready and Wolpert [191]. Moreover, as a virtual agent can be anywhere anytime, we are not bounded to explicitly calculate relative transitional energies (of a migrating population unlike natural entities) between search areas. Two fundamentally important enhancements (needed to make particle searches more efficient) are related to the manner in which a particle establishes its correlations with the surrounding landscape, and the way collateral artefacts of colonies (of agents) are declared within global environmental variables. Unlike social convergence of particle in PSO (invented by Kennedy and pictorially drawn in Figure 3.28), a scale free correlation (termed as RSO or RSA in Chapter 3) of particles is an autonomous and foraging strategy inspired by natural agents (as mentioned earlier). Therefore, unlike mandatory hierarchy in social agents, the detachment could be the key in an artificial visual tracking (e.g., in robotic vision, and in accordance with the experimental findings of Chapter 3 and 6), in which nostalgic operations and social gestures are bypassed for reduced optimisation timings.

7.3 Convergence Issues in Tracking Applications

The pre-mature stagnation (convergence to sub-optimal regions which restricts explorations of the search space) is one such unwanted feature that usually occurs in the swarm based approaches. The convergence issues (in the view of this thesis) occur due to the changing assignments of the global best solutions, as once the particles have converged to confined areas in the solution space; their diversity levels falls treacherously, and finally their vision is limited to an extent that they remain oblivious of even the surrounding landscape. Therefore, the core policy in RSO is to ensure that all feasible regions are explored using parallel search strategies, and afterwards, only prominent regions are exploited in a nested fashion (for the detection of a global optimal solution). The core building blocks of the evolutionary dynamic optimisation theory are discussed in Chapter 4 with a prospective to expedite frame rates in Chapter 6. By indulging the core feature discussed earlier, we experienced a tracking rate of as high as 200-250 frames per second during the tracking assignments.

The resultant intelligent particle roaming and emergent behaviour are significantly better optimisation methodologies compared to the traditional swarm based methods. Some rigours and challenging multi-modal test problem (e.g., Rastrigin) are solved in Section 3.5 by first using the standard PSO and then by applying its variants, later on, the problems are tackled by applying RSO. The results presented in Figures 3.45-3.66 are quite intriguing/shocking (with some prediction, as we anticipated that the social components among particles are not contributable optimisation factors) for the author as well.

The new particle roles and the search space expeditions relied on the novel agent features that are prescribed in accordance with the properties defined in Section 3.4. By analysing the convergence graphs in Section 3.5, we reached to a conclusion that the RSO algorithm transcended all main variants of the particle swarm optimisation in both time and space, Figures 3.63-3.66 delineates both the novelty and applicability of RSO in key test cases.

7.4 Registration of Searches

The second problem is merely a function level attribute which sees every particle position to be stored in the working memory, and Figure 6.24 was particularly drawn to highlight this fundamental flaw. In reality, we discovered that the storing of particle positions is not required (at all), and instead, a better focus must be on the rebirth phenomenon which facilitates particles proliferating in only specified regions of interest, thus saving valuable computational time. The virtual particle (VP) therefore does not store intermittent positions (Section 3.4), and it bypasses slow memory operations which helped us to attain a real time objective in almost every tracking problem (see Figure 6.58). The detections based on the virtual guided searches (VGS) also entailed timely recovery of the object of interest (a core problem in the vision applications). The penalising ability introduced by the VGS and RSO helped identification of the background clutter (low fast moving cloud) in the drone sequence (Section 6.11).

Finally, we would like to conclude the discussions presented in Chapter 7 in more simplistic wordings. It would help to recap the contributions of this thesis. We altered the definition and role of particles (using RSO-VGS), which sees a tracking problem in the light of pure optimisation without any unwanted simulative gesture. A radical particle (RP) is therefore an autonomous agent (responsible for its own behaviour) which scrutinizes the search space by using both olfactory and visual senses. As a RP can glance much further into a search space, a nominal population of particles could scrutinize a broader search space eliminating the need to employ hundreds and thousands of nostalgic particles. The virtual particles are mortal elements of a colony (unlike real life agents), and therefore as they do not require the convergent or dispersion energies, and thus saving these high computational resources a more task oriented feature was seen in our experiments. The tuning free method of RSO-VGS is applicable in almost every test scenario, and is not scene dependent, and is built to escape local traps. The issue of pre-mature stagnation is inherently addressed in our novel strategy, as usually, it is the space which converges rather than the particles.

7.5 Future Research

The changing roles of particles could be utilised in many other computer vision applications (e.g., establishing shape identities and contour based matching/segmentation). The most common problems in the curve based recognition could be rectified by granting data points a sense of evolution (Figure 3.34). Therefore, the curvature reducing measures (portrayed in Figure 3.34 as tighter metallic rings) are not required (in the opinion of this research) for an evolving contour aiming to identify a region of interest. Hence by implementing radical searches in Figure 3.34b, the evolution time could be reduced significantly. It was highlighted in the categorization part of this thesis (Section 2.4) that an evolving contour is nothing more than an ensemble of particles (data points), hence, by enforcing tighter control policies (at all times), we are in a conflicting situation against Norbert's and Bernstein's research findings. A meaningful optimal control is nothing more than "an act of steering" (a Norbert Weiner quote-Section 2.4.1).

I witnessed my drone enthusiast peer spending ludicrous amount of time and finances to modify its impulse response to achieve the level of stability in its flights like a humming bird (but in vain, as natural designs are more reinventing). The findings of this thesis could be utilised to address many automatic control problems as well. The ramifications of tighter deterministic drifts could be avoided by bringing in the notion of flexibility, which creates a better distributed learning environment. The high speed tracking (achieved through the techniques developed in this thesis) could be applied in many automatic resource allocation problems. The faster detection phases (based on VGS) could be deployed for better camera focus, and general video capturing devices could also benefit from this technique. However, a great deal of learning is possible from the natural expositions of life around us.

References

[1]-F. Barnard 1913. Advertising Trade Journal Printers Ink. New York USA.

[2]-Badawy, Wael; Jullien, Graham A., eds. (2003). System-on-Chip for Real-Time Applications. Kluwer international series in engineering and computer science, SECS 711.

[3]-Hurlburt, Miller, Voas An Ethical Analysis of Automation, Risk, and the Financial Crises of 2008, IEEE Reliability Society 2008 Annual Technology Report

[4]-K. R. Reddy, K. H. Priya and N. Neelima, "Object Detection and Tracking -- A Survey," 2015 International Conference on Computational Intelligence and Communication Networks (CICN), Jabalpur, 2015, pp. 418-421.

[5]-C. Hermes, J. Einhaus, M. Hahn, C. Wöhler and F. Kummert, "Vehicle tracking and motion prediction in complex urban scenarios," 2010 IEEE Intelligent Vehicles Symposium, San Diego, CA, 2010, pp. 26-33.

[6]-K. P. Karmann, "Time-recursive motion estimation using dynamical models for motion prediction," [1990] Proceedings. 10th International Conference on Pattern Recognition, Atlantic City, NJ, 1990, pp. 268-270 vol.1.

[7]-https://en.wikipedia.org/wiki/Quantum_Cloud

[8]-E. R. Pike, J. G. McWhirter, M. Bertero and C. de Mol, "Generalised information theory for inverse problems in signal processing," in Communications, Radar and Signal Processing, IEE Proceedings F, vol. 131, no. 6, pp. 660-667, October 1984.

[9]-R. Hagege and J. M. Francos, "Linear Estimation of Sequences of Multi-Dimensional Affine Transformations," 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, Toulouse, 2006, pp. II-II.

[10]-Emmanuel J. Cand`es & David L. Donoho, Recovering Edges in Ill-Posed Inverse Problems: Optimality of Curvelet FramesDepartment of Statistics Stanford University March, 2000

[11]-Aster, Richard; Borchers, Brian, and Thurber, Clifford (2012). Parameter Estimation and Inverse Problems, Second Edition, Elsevier. ISBN 0123850487, ISBN 978-

[12]-S. Ambikasaran, “A fast direct solver for dense linear systems,” https://github.com/sivaramambikasaran/HODLR, 2013.

[13]-D. Terzopoulos, "Regularization of Inverse Visual Problems Involving Discontinuities," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.PAMI-8,no.4,pp.413-424,July1986.doi: 10.1109/TPAMI.1986.4767807

[14]-Walter, E.; Pronzato, L. (1997). Identification of Parametric Models from Experimental Data. London, UK: Springer-Verlag.

[15]-Nocedal, Jorge; Wright, Stephen J. (1999). Numerical Optimization. Springer-Verlag. ISBN 0-387-98793-2.

[16]-Jan A. Snyman (2005). Practical Mathematical Optimization: An Introduction to Basic Optimization Theory and Classical and New Gradient-Based Algorithms. Springer Publishing. ISBN 0-387-24348-8

[17]-S. Besson, M. Barlaud, and G. Aubert, “Detection and tracking of moving objects using a new level set based method,” Proc. ICPR, vol. 3, pp. 1100 – 1105, Sept 2000.

[18]-D. Comaniciu, P. Meer. Robust analysis of feature space: Color image segmentation. In IEEE Conf. Computer vision and Pattern Recognition, 750 – 755, 1997

[19]-Z. Han, R. Zhang, L. Wen, X. Xie and Z. Li, "Moving Object Tracking Method Based on Improved Camshift Algorithm," 2016 International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII), Wuhan, 2016, pp. 91-95.

[20]-Blanco, J.L.; Gonzalez, J.; Fernandez-Madrigal, J.A. (2008). An Optimal Filtering Algorithm for Non-Parametric Observation Models in Robot Localization. IEEE International Conference on Robotics and Automation (ICRA'08). pp. 461–466

[21]-Särkkä, Simo (2013). Bayesian Filtering and Smoothing (PDF). Cambridge University Press.

[22]-Trelea, I.C. (2003). "The Particle Swarm Optimization Algorithm: convergence analysis and parameter selection". Information Processing Letters. **85** (6): 317–325.

[23]-Pham, D.T., Castellani, M. (2009), The Bees Algorithm – Modelling Foraging Behaviour to Solve Continuous Optimisation Problems. Proc. ImechE, Part C, 223(12), 2919-2938.

[24]-M. Dorigo, Optimization, Learning and Natural Algorithms, PhD thesis, Politecnico di Milano, Italy, 1992.

[25]-R. S. Parpinelli, H. S. Lopes and A. A Freitas, "Data mining with an ant colony optimization algorithm," IEEE Transaction on Evolutionary Computation, vol.6, no.4, pp.321-332, 2002.

[26]-Abbott, K. R.; Dukas, R. (2009). "Honeybees consider flower danger in their waggle dance". AnimalBehaviour. **78** (3): 633–635.

[27]-Tsai, P. W.; Pan, J. S.; Liao, B. Y.; Tsai, M. J.; Istanda, V. (2012). "Bat algorithm inspired algorithm for solving numerical optimization problems". Applied MechanicsandMaterials.148-149:

[28]-D.E. Catlin. Estimation, Control and the Discrete Kalman Filter. Springer Verlag, 1984.

[29]-P. W. Tsai, J. S. Pan, B. Y. Liao, M. J. Tsai, V. Istanda, Bat algorithm inspired

algorithm for solving numerical optimization problems, Applied Mechanics and Materials, Vo.. 148-149, pp.134-137 (2012)

[30]-Wilson, C. E. (2003). Kinematics and dynamics of machinery. Pearson Education. ISBN 978-0-201-35099-9.

[31]-Bar-Shalom, Yaakov; Li, X. Rong; Kirubarajan, Thiagalingam (July 2001). Estimation with Applications to Tracking and Navigation. New York: John Wiley & Sons. pp. 308–317.

[32]-C. Fernandes, L. Gurvits, and Z. Li. Near optimal nonholonomic motion planning for a system of coupled rigid bodies. IEEE Transactions on Automatic Control, March 1994.

[33]-G. F. Franklin, J. D.Çengel, Yunus A., and Michael A. Boles. "9-8." Thermodynamics: An Engineering Approach. 7th ed. New York: McGraw-Hill, 2011. 510.

[34]-Davim, J. Paulo, editor (2011) Mechatronics, John Wiley & Sons ISBN 978-1-84821-308-1 .

[35]-Kleppner, D.; Kolenkow, R. J. (1973). An Introduction to Mechanics. McGraw-Hill. ISBN 0-07-035048-5.

[36]-Galley, Chad R. (2013). "Classical Mechanics of Nonconservative Systems". Physical Review Letters. **110** (17): 174301. arXiv:1210.2745.

[37]-Arnol'd, V. I.; Kozlov, V. V.; Neĭshtadt, A. I. (1988), Mathematical aspects of classical and celestial mechanics, **3**, Springer-Verlag

[38]-Zimek, A.; Schubert, E.; Kriegel, H.-P. (2012). "A survey on unsupervised outlier detection in high-dimensional numerical data". Statistical Analysis and Data Mining. **5** (5): 363–387. doi:10.1002/sam.11161.

[39]- mars.nasa.gov/mer/home/

[40]-D. Simon. Evolutionary Optimization Algorithms. Wiley, 2013.

[41]-http://www.mobileye.com

[42]- S, Bull, N, Cooke & A, Mabbott. (2007). Visual Attention in Open Learner Model Presentations An Eye-Tracking Investigation, in C. Conati, K. McCoy & G. Paliouras (Eds), User Modeling 2007:11th International Conference, Springer-Verlag, Berlin Heidelberg, 187-196.

[43]-European Roadmap Smart Systems for Automated Driving, European Technology Platform on Smart Systems Integration (EPoSS), 2015.

[44]-http://microbetracker.org/help/helpMicrobeTracker.htm

[45]-Shen,C.Galazia (2015): Automated tracking and analysis of behavior in restrained insects, Journal of Neuroscience Methods, Volume 239, 15 January 2015, Pages 194–205

[46]-T. E. Boult, R. J. Micheals and X. Gao, “Into the Woods: Visual surveillance of noncooperative and camouflaged targets in complex outdoor settings,” Proc. IEEE, vol. 89, no. 10,pp 1382-1402,2001

[47]-Joubert, J. and Sharma, D. Using CMOS Cameras for Light Microscopy. Microscopy Today 19(4):22-29, 2011.

[48]-Gérard Medioni; Sing Bing Kang (2004). Emerging Topics in Computer Vision. Prentice Hall. ISBN 0-13-101366-1.

[49]-Autodesk, Inventor® : CAD program, computer software, 2013. http://www.autodesk.com/products/autodesk-inventorfamily/overview [12] Rippmann M.,

[50]-Norman, G.R., Brooks, L.R., Coblentz, C.L and Babcock, C. J. “The correlation of feature identification and category judgments in diagnostic radiology,” Mem Cognit, 20,344-355(1992)

[51]-Le Dinh, Phuc-Tue; Patry, Jacques (February 24, 2006). "Video compression artifacts and MPEG noise reduction". Video Imaging DesignLine. Retrieved April 30, 2010.

[52]-Alessi, A. Sudano, D. Accoto, E. Guglielmelli, "Development of an autonomous robotic fish," In Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEE RAS & EMBS International Conference on Robotics(pp. 1032-1037).IEEE

[53]-Bunge, Mario; "A general black-box theory", Philosophy of Science, Vol. 30, No. 4, 1963, pp. 346-358. jstor/186066

[54]-Hendrik Richter; Andries P. Engelbrecht (2014). Recent Advances in the Theory and Application of Fitness Landscapes. ISBN 978-3-642-41888-4.

[55]-Pacheco, J. M., Traulsen, A. & Nowak, M. A. 2006 Coevolution of strategy and structure in complex networks with dynamical linking Phys. Rev. Lett. 97, 258103. (doi:10. 1103/PhysRevLett.97.258103)

[56]-http://www.aylward.org/notes/open-access-medical-image-repositories

[57]-Rosenfeld , A. C. Kak (1982), Digital Picture Processing, Academic Press, Inc., ISBN 0-12-597302-0

[58]-W,Norbert; Cybernetics or control and communication in the animal and the machine,professor of mathematics, the Massachusetts institute of technology,second edition.

[59]-Latash, Mark L. (ed.) Progress in Motor Control: Bernstein's Traditions in Movement Studies, Vol. 1

[60]-Bernstein, Nikolai (1967). The Coordination and Regulation of Movements. Oxford: Pergamon Press.

[61]- W.E.Dyer. Inspiration your ultimate calling, Hay House Publications 2/28/06. ISBN 1-4019-0721-0

[62]-Stone, P., and Veloso, M. 1999. Task decomposition, dynamic role assignemnt, and low-bandwidth communicaiton for real-time strategic teamwork. Artificial Intelligence 110(2):241–273.

[63]-C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149), Fort Collins, CO, 1999, pp. 252 Vol. 2.

[64]-O. D. Escoda, A. Petrovic and P. Vandergheynst, "Segmentation of natural images using Scale-Space representations: A linear and a non-linear approach," 2002 11th European Signal Processing Conference, Toulouse, 2002, pp. 1-4.

[65]-Y. Shi and W. C. Karl, "Real-time tracking using level sets," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005, pp. 34-41 vol. 2.

[66]-Xue Zhou, Xi Li and Weiming Hu, "Level set tracking with dynamical shape priors," 2008 15th IEEE International Conference on Image Processing, San Diego, CA, 2008, pp. 1540-1543.

[67]-Tse Min Chen, R.C. Luo, Tsu Hung Hsiao. Visual tracking using adaptive color histogram model. IECON '99, Vol. 3: 1336 -1341, 1999.

[68]-David J. Fleet & Yair Weiss (2006). "Optical Flow Estimation". In Paragios; et al. Handbook of Mathematical Models in Computer Vision (PDF). Springer. ISBN 0-387-26371-3.

[69]-J. Cox and S. L. Hingorani. An efficient implementation of Reid’s multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. PAMI

[70]-Mikolajczyk, C., Schmid, C., and Zisserman, A. 2004. Human detection based on a probabilistic assembly of robust part detectors. ECCV, vol. I, pp. 69–82

[71]-R. Malladi, J. A. Sethian, and B. C. Vemuri, \Shape modeling with front propagation: A level set approach," Center for Pure and Applied Mathematics, Report PAM-589, Univ. of California, Berkeley, August 1993.

[72]-R.Bowden and M.Sarhadi “A non-linear Model of Shape and Motion for Tracking Finger Spelt American Sign Language” Image and Vision Computing,2002

[73]-K. R. Reddy, K. H. Priya and N. Neelima, "Object Detection and Tracking -- A Survey," 2015 International Conference on Computational Intelligence and Communication Networks (CICN), Jabalpur, 2015, pp. 418-421.

[74]-Yilmaz, “Object tracking a survey” Ohio State University.

[75]- Masreliez, C. Johan; Martin, R D (1977). "Robust Bayesian estimation for the linear model and robustifying the Kalman filter". IEEE Transactions on Automatic Control. **22** (3): 361–371. doi:10.1109/TAC.1977.1101538

[76]-Lee, Peter M. (2012). "Chapter 1". Bayesian Statistics. Wiley. ISBN 978-1-1183-3257-3.

[77]-Kalman, R. E. (1960). "A New Approach to Linear Filtering and Prediction Problems". Journal of Basic Engineering. **82**: 35. doi:10.1115/1.3662552

[78]-Masreliez, C. Johan; Martin, R D (1977). "Robust Bayesian estimation for the linear model and robustifying the Kalman filter". IEEE Transactions on Automatic Control. **22** (3): 361–371. doi:10.1109/TAC.1977.1101538

[79]-Kroese, D. P.; Taimre, T.; Botev, Z. I. (2011). Handbook of Monte Carlo Methods. John Wiley & Sons.

[80]-http://nmarkou.blogspot.co.uk/2011/11/particle-filter-snippet.html

[81]-Anderson and Moore 1979. Optimal Filtering. Prentice Hall.

[82]-M. Isard and A. Blake. CONDENSATION: conditional density propagation for visual tracking. International Journal of Computer Vision, 29(1), 1998.

[83]-Y. Cheng. Mean shift, mode seeking, and clustering. PAMI, 17(8), 1995.

[84]-Fast Gradient Methods, lecture notes by Prof. Lieven Vandenberghe for EE236C at UCLA

[85]-Sutton, R. S. (1986). Two problems with backpropagation and other steepest-descent learning procedures for networks. Proc. 8th Annual Conf. Cognitive Science society

[86]-Atkinson, Kendell A. (1988). "Section 8.9". An introduction to numerical analysis (2nd ed.). John Wiley and Sons. ISBN 0-471-50023-2.

[87]-Comaniciu, Dorin; Peter Meer (May 2002). "Mean Shift: A Robust Approach Toward Feature Space Analysis". IEEE Transactions on Pattern Analysis and Machine Intelligence. IEEE. **24** (5): 603–619. doi:10.1109/34.1000236.

[88]-Epanechnikov, V.A. (1969). "Non-parametric estimation of a multivariate probability density". Theory of Probability and its Applications. **14**: 153–158. doi:10.1137/1114019.

[89]-Gut, Allan (2009) An Intermediate Course in Probability, Springer. ISBN 9781441901613 (Chapter 5)

[90]-Y. Benezeth , B. Emile, C. Rosenberger. Comparative study on foreground detection algorithms for human detection. International Conference on Image and Graphics, pages 661–666, 2007.

[91]-H. Tong, H. Zhang, H. Meng, X. Wang, Multitarget tracking before detection via probability hypothesis density filter, in: Int. Conference on Electrical and Control Engineering, Wuhan, China, 2010, pp. 1332–1335.

[92]-Bhattacharyya, A. (1943). "On a measure of divergence between two statistical populations defined by their probability distributions". Bulletin of the Calcutta Mathematical Society. **35**: 99–109. MR 0010358

[93]-D. Comaniciu, V. Ramesh, P. Meer, Kernel-based object tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(5), pp564-575, 2003

[94]-Deza, Elena; Deza, Michel Marie (2009). Encyclopedia of Distances. Springer. p. 94.

[95]-Lindeberg, Tony, "Scale-space for discrete signals," PAMI(12), No. 3, March 1990, pp. 234-254.

[96]-B. Tamersoy (September 29, 2009). "Background Subtraction – Lecture Notes" (PDF). University of Texas at Austin.

[97]-Gordon, N. J.; Salmond, D. J.; Smith, A. F. M. (1993). "Novel approach to nonlinear/non-Gaussian Bayesian state estimation". IEE Proceedings F on Radar and Signal Processing. **140** (2): 107–113. doi:10.1049/ip-f-2.1993.0015. Retrieved 2009-09-1

[98]-Cetnarowicz, K., Kisiel-Dorohinicki, M. & Nawarecki, E. 1996. The application of evolution process in multi-agentworld (MAW) to the prediction system. InProceedings of the 2nd International Conference on Multi-AgentSystems (ICMAS96)M. Tokoro (ed.), 2632.AAI Press.

[99]-Cavagna, A., & Giardina, I. (n.d.). The seventh starling. Significance, 5(2), 62-66. doi:10.1111/j.1740-9713.2008.00288.x

[100]-Sumpter, D.J.T. (2010). Collective Animal Behavior (Princeton University Press).

[101]http://www.princeton.edu/main/news/archive/S36/02/56I00/index.xml?section=topstories

[102]-Couzin I, Krause J, Franks N, Levin S (2005) Effective leadership and decision-making in animal groups on the move. Nature 433: 513–516

[103]-Erber J, Kierzek S, Sander E, Grandy K (1998) Tactile learning in the honeybee. J Comp Physiol A: 737±744

[104]-Bilchev B, Parmee IC. The ant colony metaphor for searching continuous design spaces. In: Fogarty TC, editor. Proceedings of the AISB workshop on evolutionary computation. Lecture Notes in Comput Sci, vol. 993. Berlin: Springer; 1995. p. 25–39.

[105]-Ostermeyer, G.-P. Many particle systems. German Polish Workshop, IPPT PAN,Warszawa, (1995).

[106]-Kolb, A., & Kolb, D. A. (1999). Bibliography of research on experiential learning theory and the Learning Style Inventory Department ofOrganizational Behavior, Weatherhead School of Management, Case Western Reserve, University, Cleveland, OH.

[107]-Schunk, D. (2004). Learning theories: An educational perspective (4th ed.). Upper Saddle River, NJ, USA: Pearson, p. 220, ISBN 0130384968.

[108]-Aaron R. Bradley and Zohar Manna. 2007. The Calculus of Computation: Decision Procedures with Applications to Verification. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[109]-Hazewinkel, Michiel, ed. (2001), "Point of inflection", Encyclopedia of Mathematics, Springer, ISBN 978-1-55608-010-4

[110]-Adams, A. Adams; Essex, Christopher (2009). Calculus: A Complete Course. Pearson Prentice Hall. p. 744. ISBN 978-0-321-54928-0.

[111]-Whittaker, E. T. and Watson, G. N. "Forms of the Remainder in Taylor's Series." §5.41 in A Course in Modern Analysis, 4th ed. Cambridge, England: Cambridge University Press, pp. 95-96, 1990.

[112]-Bertsekas, Dimitri P. (2009). Convex Optimization Theory. Belmont, MA.: Athena Scientific. ISBN 978-1-886529-31-1.

[113]-S. Bubeck. Introduction to online optimization. Lecture Notes, 2011.

[114]-K. Fukunaga and L.D. Hostetler, “The estimation of the gradient of a density function, with applications in pattern cognition,” IEEE Truns. Infi,rmution Theory, vol.21, pp. 32-40, 1975.

[115]-T. McInerney, D. Terzopoulos. (1996) Deformable models in medical image analysis. Proceedings of the Workshop on Mathematical Methods in Biomedical Image Analysis, 171-180.

[116]-C. Xu and J.L. Prince, "Gradient Vector Flow: A New External Force for Snakes," Proc. IEEE Conf. on Comp. Vis. Patt. Recog. (CVPR), Los Alamitos: Comp. Soc. Press, pp. 66–71, June 1997

[117]-Osher, S., and Sethian, J.A.,Fronts Propagating with Curvature Dependent speed: Algorihms Based on Hamilton-Jacobi Formulation, Journal of Computational Physics, Vol. 79, pp. 12-49,1988.

[118]-M.R.Rajamani.Data-based Techniques to Improve State Estimationin Model Predictive Control. University of Wisconson Press, 2007, ppii-ii

[119]-A.Jameson, "Gradient Based Optimization Methods",MAE TechnicalReport No. 2057, Princeton University, 1995.
[120]-Larson, Ron; Edwards, Bruce H. (2009). Calculus (9th ed.). Brooks/Cole. ISBN 0-547-16702-4.

[121]-Černý, V. (1985). "Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm". Journal of Optimization Theory and Applications. **45**: 41–51. doi:10.1007/BF00940812.

[122]-A.P.Godse and U.A.Bakshi (2009). Communication Engineering. Technical Publications. p. 36. ISBN 978-81-8431-089-4.

[123]- Alam,I "Biologically inspired object tracking: A modular approach with distributed particle like sensors," *Third International Conference on Innovative Computing Technology (INTECH 2013)*, London, 2013, pp. 23-28.doi: 10.1109/INTECH.2013.6653720

[124]-Carter, J. J. Mor´e, "The MINIPACK-2 Test Problem Collection,"Mathematics and Computer Science Division, Agronne National Laboratory, TechnicalMemorandum No. 150, 1991.B. M. Averick, R. G.

[125]-http://imgur.com/

[126]-
https://www.youtube.com/watch?v=SN5igku_kGk&list=PL2MyffdNOEsCdbJ2crWNk1MtTcULe_PJX&index=2

[127]-Spottiswoode, C.N., Begg, K.S. & Begg, C.M. 2016 Reciprocal communication in human-honeyguide mutualism. Science 353: 387-389.

[128]-Springer, Information Fusion in Data Mining (2003), ISBN 3-540-00676-1

[129]- Alam,I "Object tracking in video sequences using information fusion principles. Meanshift kernel implementation using fuzzy rules," *2013 5th Computer Science and Electronic Engineering Conference (CEEC)*, Colchester, 2013, pp. 146-151.doi: 10.1109/CEEC.2013.6659462

[130]-http://www.snowcrystals.com/science/science.html

[131]-Cavagna, A., Giardina,V. (2008) The STARFLAG handbook on collective animal behaviour: Part I, empirical methods. Animal Behaviour, to be published. Availablefromhttp://arxiv.org/abs/0802.1668.)

[132]-Ballerini, M., Cabibbo, (2008) Interaction ruling collective animal behaviour depends on topological rather than metric distance: evidence from a field study. Proceedings of the National Academy of Sciences of the USA105, 1232–1237.

[133]-http://www.bbc.co.uk/programmes/b0861m8b

[134]-R. Oftadeh et al. (2010), A novel meta-heuristic optimization algorithm inspired by group hunting of animals: Hunting search, 60, 2087–2098.

[135]-Benkő A., Dósa G., Tuza Z. (2010), Bin Packing/Covering with Delivery, Solved with the Evolution of Algorithms, Proc. 2010 IEEE 5th International Conference on Bio-Inspired Computing: Theories and Applications, BIC-TA 2010, pp. 298–302.

[136]-Ashlock, D. (2006), Evolutionary Computation for Modeling and Optimization, Springer, ISBN 0-387-22196-4.

[137]-Bäck, T. (1996), Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms, Oxford Univ. Press.

[138]-Lones, Michael A. (2014). "Metaheuristics in Nature-Inspired Algorithms" (PDF). GECCO '14. doi:10.1145/2598394.2609841.

[139]-Feldman, D. H. (1999). "The Development of Creativity". In Sternberg, R.J. Handbook of Creativity. Cambridge University Press.

[140]-Jackson, Mike (2000). "Wherefore Gadolinium? Magnetism of the Rare Earths" (PDF). IRM Quarterly. Institute for Rock Magnetism. **10** (3): 6.

[141]-Feynman, Richard P.; Robert B. Leighton; Matthew Sands (1963). The Feynman Lectures on Physics, Vol. I. USA: California Inst. of Technology. pp. 37.5–37.6. ISBN 0465024939.

[142]-Vincent Conitzer and Tuomas Sandholm. An algorithm for single-agent deterministic automated mechanism design without payments. In IJCAI-03 workshop on Distributed Constraint Reasoning (DCR), Acapulco, Mexico, 2003.

[143]-Cavagna.A, Parasi.G,"The scale free correlations in starling flocks"(2010) PNAS Alerts, vol. 107 no. 26 11865–11870, doi:10.1073/pnas.1005766107

[144]-http://www.peterbeerli.com/classes/images/4/4e/AmSci2011Hayes.pdf

[145]-Holland J. (1975), "Adaptation in Natural and Artificial Systems", Ann Arbor: University of Michigan Press.

[146]-Kennedy, J.; Eberhart, R.C. (2001). Swarm Intelligence. Morgan Kaufmann. ISBN 1-55860-595-9.

[147]-Herschbach, M. On the role of social interaction in social cognition: a mechanistic alternative to enactivism. Phenomenol. Cogn. Sci. **11**, 467–486 (2012).

[148]-Wilson and B.F. Skinner, Developments in Primatology: Human Sociobiology: The Essential E.O. Wilson, Progress and Prospects, DOI 10.1007/978-0-387-89462-1 2, C Springer Science+Business Media, LLC 2009

[149]-Kennedy, J.; Eberhart, R. (1995). "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks. **IV**. pp. 1942–1948. doi:10.1109/ICNN.1995.488968

[150]-Cleveland, Harlan and Jacobs, Garry, The Genetic Code for Social Development". In: Human Choice, World Academy of Art & Science, USA, 1999, p. 7.

[151]-Heppner, F. and Grenander, U. (1990). A stochastic nonlinear model for coordinated birdflocks.The Ubiquity of chaos,pages 233–238

[152]-Rakitianskaia and A. P. Engelbrecht, "Cooperative charged particle swarm optimiser," 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), Hong Kong, 2008, pp. 933-939.

[153]-C. V. García-Mendoza, M. G. Villarreal-Cervantes, O. Peñaloza-Mejía and G. Sepúlveda-Cervantes, "Adaptive Control of a DC Motor Based on Swarm Intelligence," 2015 International Conference on Computational Science and Computational Intelligence (CSCI), Las Vegas, NV, 2015, pp. 192-197. doi: 10.1109/CSCI.2015.50

[154]-S. Banerjee, K. Da Zhao, W. Rao and M. Žefran, "Decentralized self-balancing systems," 2013 IFIP/IEEE 21st International Conference on Very Large Scale Integration (VLSI-SoC), Istanbul, 2013, pp. 340-343.

[155]-T. K. Roy, H. R. Pota, M. Garratt and H. Teimoori, "Robust control for longitudinal and lateral dynamics of small scale helicopter," Proceedings of the 31st Chinese Control Conference, Hefei, 2012, pp. 2607-2612.

[156]-Glatzmaier, Gary. "The Geodynamo". University of California Santa Cruz. Retrieved 20 October 2013.

[157]-http://www.sciencephoto.com

[158]-Kanter, R. (1994) Collaborative Advantage: The Art of Alliances Harvard Business Review, 72:4 pp.96-108

[159]-Fischer, Hank (1995). "From Varmints to Rock Stars".Wolf Wars—The Remarkable Inside Story of the Restoration of Wolves to Yellowstone. Helena, MT: Falcon Press Publishing Co. Inc. pp. 35–43. ISBN 1-56044-352-9.

[160]-Mitchell, Melanie (1996). An Introduction to Genetic Algorithms. Cambridge, MA: MIT Press. ISBN 9780585030944.

[161]-Holmes, S. J., and H. M. Loomis. 1909. The heredity of eye color and hair color in man. Biological Bulletin 18: 50Ð65.

[162]-Eiben, A. E. et al (1994). "Genetic algorithms with multi-parent recombination". PPSN III: Proceedings of the International Conference on Evolutionary Computation. The Third Conference on Parallel Problem Solving from Nature: 78–87. ISBN 3-540-58484-6.

[163]-Lawrence, Eleanor (2005) Henderson's Dictionary of Biology. Pearson, Prentice Hall. ISBN 0-13-127384-1

[164]-J. R. Koza. Genetic Programming: On the Programming of Computers by means of Natural Evolution. MIT Press, Massachusetts, 1992.

[165]-Hall, Dave L.; Llinas, James (1997). "Introduction to Multisensor Data Fusion". Proceedings of IEEE. **85** (1): 6–23.

[166]-Crowley, J. L. and Sanderson, A. C. "Multiple resolution representation and probabilistic matching of 2-D gray-scale shape", IEEE Transactions on Pattern Analysis and Machine Intelligence, 9(1), pp 113-121, 1987

[167]-Schwefel H-P and Rudolph G (1995) Contemporary evolution strategies. In: Morana F,

[168]-Bernardo, J.M., Smith, A.F.M. (2000) Bayesian Theory'.' Wiley. ISBN 0-471-49464-X (pages 209, 366)

[169]-Lipowski, Roulette-wheel selection via stochastic acceptance (arXiv:1109.3627)

[170]-De Jong, K. (2006). Evolutionary Computation:A Unified Approach. MIT Press. ISBN 9780262041942.

[171]-Kenneth Price, M. Storn, Differential Evolution: A Practical Approach to Global Optimization, Springer-Verlag New York, Inc. Secaucus, NJ, USA ©2005 ISBN:3540209506

[172]-M. M. al-Rifaie, D. Joyce, S. Shergill and M. Bishop, "Investigating stochastic diffusion search in data clustering," 2015 SAI Intelligent Systems Conference (IntelliSys), London, 2015, pp. 187-194.doi: 10.1109/IntelliSys.2015.7361143

[173]-Zhan, Z-H.; Zhang, J.; Li, Y; Chung, H.S-H. (2009). "Adaptive Particle Swarm Optimization" (PDF). IEEE Transactions on Systems, Man, and Cybernetics. **39** (6): 1362–1381. doi:10.1109/TSMCB.2009.201595

[174]-Storn, R.; Price, K. (1997). "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces". Journal of Global Optimization. **11**: 341–359. doi:10.1023/A:1008202821328

[175] orman, Donald & Verganti, Roberto. (2014). Incremental and Radical Innovation: Design Research vs.
Technology and Meaning Change. Design Issues. 30. 78-96. 10.1162/DESI_a_00250.

[176] Bailey, R., Brake, M. (Eds.) (1981), Radical Social Work and Practice. London: Sage

[177] BECKER, G. (1983): A Theory of Competition among Pressure
Groups for Political Influence. In: Quarterly Journal
of Economics 98 (3): 371-400.

[178] https://dictionary.cambridge.org/dictionary/english/radical

[179]-M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, Internat. J. Comput. Vision 1 (4) (1988) 321–331.

[180]-A.-R. Mansouri. Region tracking via level set pdes without motion computation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(7):947– 961, 2002

[181]-L.-T. Cheng, "An Efficient level set method for constructing wavefronts in three space dimensions," UCLA Computational and Applied Mathematics Report (06-15), April 2006.

[182]-S. Osher and J. A. Sethian. Fronts propagating with curvature dependent speed: Algorithms based in Hamilton-Jacobi formulations. Journal of Computational Physics, 79:12–49, 1988.
[183]-Boyer D, Miramontes O, Ramos-Fernández G (2008) Evidence for biological Levy flights stands. arXiv:0802.1762.

[184]-De Jager M, Weissing FJ, Herman PMJ, Nolet BA, van de Koppel J (2011) Lévy walks evolve through interaction between movement and environmental complexity. Science 332(6037):1551–1553.

[185]Healey CIM, Pratt SC (2008) The effect of prior experience on nest site
evaluation by the ant Temnothorax curvispinosus. Animal Behaviour 76:893–899

[186] Gancarz, G. (1996). Guided Search 3.0: A model of visual search catches up with Jay Enoch 40years later. In V. Lakshminarayanan (Ed.), Basic and clinical applications of vision science (pp. 189–192). Dordrecht, Netherlands: Kluwer Academic.

[187]Yang, X. S. (2008). Nature-Inspired Metaheuristic Algorithms. Luniver Press. ISBN 1-905986-10-6.

[188] Lourenço, H.R.; Martin O.; Stützle T. (2003). "Iterated Local Search". Handbook of Metaheuristics.
Kluwer Academic Publishers, International Series in Operations Research & Management Science. 57: 321–353.

[189] Viswanathan, G.; Afanasyev, V.; Buldyrev, S.; Havlin, S.; Daluz, M.; Raposo, E.; Stanley, H. (2000).
"Lévy flights in random searches". Physica A: Statistical Mechanics and its Applications. 282: 1–12. doi:10.1016/S0378-4371(00)00071-6

[190] Ben-Avraham, D.; Havlin, S. (2000). Diffusion and reaction in disordered systems. Cambridge University Press.

[191] Wolpert, D.H., Macready, W.G. (1997), "No Free Lunch Theorems for Optimization", IEEE Transactions on Evolutionary Computation 1, 67.

[192] Wolfe, J.M. (2014). Approaches to visual search: feature integration theory and guided search. The Oxford handbook of attention.
Oxford: Oxford University Press. pp. 11–50. ISBN 9780199675111.

[193]M. Malek; M. Huruswamy; H. Owens; M. Pandya (1989). Serial and parallel search techniques for the traveling salesman problem. Annals of OR: Linkages with Artificial Intelligence.

[194] C. Igel. Evolutionary kernel learning. In Encyclopedia of Machine Learning. Springer-Verlag, 2010

[195]Evolution strategies, A comprehensive introduction, HANS-GEORG BEYER and HANS-PAUL SCHWEFEL, Department of Computer Science XI, University of Dortmund, Joseph-von-Fraunhoferstr. 20,D-44221 Dortmund, Germany

[196]-Collard, P., Escazut, C., & Gaspar, A. (1997). An evolutionary approach for time dependant optimization. International Journal on Artificial IntelligenceTools,6(4), 665–695.

[197]-Branke, J., & Mattfeld, D. (2000). Anticipation in dynamic optimization: The scheduling case. In M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. LuttonJ. J. Merelo, & H.-P. Schwefel (Eds.)

[198]-Cobb, H. G. (1990).An investigation into the use of hypermutation as an adaptive operator in genetic algorithms having continuous, time-dependent non stationary environments(Tech. Rep. No. 6760 (NLR Memorandum)). Wash-ington, D.C.: Navy Center for Applied Research in Artificial Intelligence.

[199]-Zheng, H. Liu, A different topology multi-swarm PSO in dynamic environment, in: IEEE International Symposium on IT in Medicine Education, 2009 ITIME ’09, vol. 1, 2009, pp. 790–795.

[200]-Branke, A Evolutionary approaches to dynamic environments updated survey, in: GECCO Workshop on Evolutionary Algorithms for Dynamic Optimization Problems, 2001, pp. 27–30.

[201]-K. Weicker, An analysis of dynamic severity and population size, in International Conference on Parallel Problem Solving from Nature, PPSN, Vol. 1917 of Lecture Notes in Computer Science, Springer, 2000.

[202]-W. Tang, Q. Wu, J. Saunders, Bacterial foraging algorithm for dynamic environments, in: IEEE Congress on Evolutionary Computation CEC2006, 2006, pp. 1324–1330

[203]-J. Mehnen, G. Rudolph, T. Wagner, Evolutionary optimization of dynamic multi objective functions, Technical Report FI-204/06, Universitat Dortmund, Dortmund, Germany, 2006

[204]-Endre Süli and David Mayers, An Introduction to Numerical Analysis, Cambridge University Press, 2003. ISBN 0-521-00794-1.

[205]-S. Yang, "Genetic Algorithms with Memory- and Elitism-Based Immigrants in Dynamic Environments," in Evolutionary Computation, vol. 16, no. 3, pp. 385-416, Sept. 2008.

[206]-Cavallaro and T. Ebrahimi. Video object extraction based on adaptive background and statistical change detection. In Proceedings of SPIE Electronic Imaging - Visual Communications and Image Processing, pages 465–475, San Jose, California, USA,2001

[207]-F. Glover; M. Laguna (1997). Tabu Search. Kluwer Academic Publishers.

[208]-I.Moser , R.Chiong; Dynamic Function Optimization: The Moving Peaks BenchmarkMetaheuristics for Dynamic Optimization, SCI 433, pp. 35–9.springerlink.comcSpringer-Verlag Berlin Heidelberg 2013

[209]-Fernandez PC, Farina WM (2005): Collective nectar foraging at low reward conditions in honeybees Apis mellifera. Apidologie 36, 301–311.

[210]-Dukas R, Visscher PK (1994): Lifetime learning by foraging honey bees. Animal Behaviour 48, 1007–1012

[211] http://intelligence.worldofcomputing/brute-force-search Brute Force Search, December 14th, 2009.

[212]-P. Le, A. D. Duong, H. Q. Vu and N. T. Pham, "Adaptive Hybrid Mean Shift and Particle Filter," 2009 IEEE-RIVF International Conference on Computing and Communication Technologies, Da Nang, 2009, pp. 1-4.

[213]-Jingxin Du, Jun Zhou, Chang Li and Lin Yang, "An overview of dynamic data mining," 2016 3rd International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS), Jinzhou, 2016, pp. 331-335.

[214]-W. Hu et al., “A Survey on Visual Surveillance of Object Motion and Behaviors,” IEEE Trans. Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol. 34, no. 3, 2004, pp. 334-352.

[215]-X. Xue, “Video-Based Animal Behavior Analysis,” Ph.D., The University of Utah, United States -- Utah, 2009.

[216]-https://www.aapt.org/Conferences/newfaculty/upload/Coop-Problem-Solving-Guide.pdf

[217]Howard, Ronald A. (1960). Dynamic Programming and Markov Processes (PDF). The M.I.T. Press.

[218]-Del Moral, Pierre (2013). Mean field simulation for Monte Carlo integration. Chapman & Hall/CRC Press. p. 626. Monographs on Statistics & Applied Probability

[219]-Wilson EO, Hölldobler B (2005). "The rise of the ants: A phylogenetic and ecological explanation". Proceedings of the National Academy of Sciences. **102** (21): 7411–7414. Bibcode:2005PNAS..102.7411W. doi:10.1073/pnas.0502264102. PMC 1140440. PMID 15899976.

[220]-Qianying Pi and Hongtao Ye, "Survey of particle swarm optimization algorithm and its applications in antenna circuit," 2015 IEEE International Conference on Communication Problem-Solving (ICCP), Guilin, 2015, pp. 492-495.

[221]-C. Voudouris and E. P. K. Tsang, "Guided local search joins the elite in discrete optimisation", in Proceedings, DIMACS Workshop on Constraint Programming and Large Scale Discrete Optimisation, 1998.

[222]-Deb, K. and Goldberg, D. E. (1989). An investigation of niche and species formation in genetic function optimization. In Schaer, J. D., editor, Proc. Third Int. Conf. on Genetic Algorithms, pp. 42{ 50. Morgan Kaufmann.

[223]-Dixon, B. (2001). "Animal emotions. Ethics and the Environment". Ethics. **6** (2): 22–30. doi:10.2979/ete.2001.6.2.22.

[224]-Engelbrecht, "Heterogeneous Particle Swarm Optimisation," in ANTS 2010, Brussels, 2010.

[225]-M.S. Blum 1970 . The chemical basis of insect society. In Chemicals controlling Insect Behavior. M. Beroza, Ed Academic, New York, pp. 61-94

[226]-X. S. Yang, "Firefly Algorithms for Multimodal Optimization," in Stochastic algorithms: foundations and applications, 2009

[227]-Apter, Michael J. (1970). The Computer Simulation of Behaviour. London: Hutchinson & Co. p. 83.

[228]-Kahneman, Daniel (1982). Judgment under uncertainty: heuristics and biases. Cambridge New York: Cambridge University Press. ISBN 0521284147.

[229]-E. K. Burke, E. Hart, G. Kendall, J. Newall, P. Ross, and S. Schulenburg, Hyper-heuristics: An emerging direction in modern search technology, Handbook of Metaheuristics (F. Glover and G. Kochenberger, eds.), Kluwer, 2003, pp. 457–474.

[230]-Weiss, George H. (1994). Aspects and Applications of the Random Walk. Random Materials and Processes. North-Holland Publishing Co., Amsterdam. ISBN 0-444-81606-2. MR 1280031.

[231]-Lourenço, H.R.; Martin O.; Stützle T. (2003). "Iterated Local Search". Handbook of Metaheuristics. Kluwer Academic Publishers, International Series in Operations Research & Management Science. **57**: 321–353.

[232]-Panter-Brick, C.; R. H. Layton; eds. (2001). Hunter-gatherers: an interdisciplinary perspective. Cambridge University Press. ISBN 0-521-77672-4.

[233]-Sims, David W.; Humphries, Nicolas E.; Bradford, Russell W.; Bruce, Barry D. "Lévy flight and Brownian search patterns of a free-ranging predator reflect different prey field characteristics". Journal of Animal Ecology. **81:** 432–442. doi:10.1111/j.1365-2656.2011.01914.x.

[234]-Viswanathan, G.M.; Raposo, E.P.; da Luz, M.G.E. (September 2008). "Lévy flights and superdiffusion in the context of biological encounters and random searches". Physics of Life Reviews. **5** (3): 133–150. doi:10.1016/j.plrev.2008.03.002

[235]-STAUFFER, D. F., AND L. B. BEST. 1982. Nest-site selection by cavity-nesting birds of riparian habitats in Iowa. Wilson Bull. 94~329-337.

[236]- T. O. Crist J. A. Mac-Mahon Individual foraging components of harvester ants: movement patterns and seed patch fidelity Journal Volume 38, Issue 4, 1991ISSN: 0020-1812 (Print) 1420-9098

[237]-R Schürch, MJ Couvillon**,** D Burns, K Tiasman, D Waxman, & FLW Ratnieks (2013). Incorporating variability in honey bee waggle dance decoding improves the mapping of communicated resource locations Journal of Comparative Physiology A 199, 1143-1152.

[238]-Schultheiss, P., Raderschall, C. A., & Narendra, A. (2015). Follower ants in a tandem pair are not always naïve. Scientific Reports, 5.

[239]Lowe, David G. (1999). "Object recognition from local scale-invariant features" (PDF). Proceedings of the International Conference on Computer Vision. 2. pp. 1150–1157. doi:10.1109/ICCV.1999.790410.

[240]Cannon, John Rozier (1984), The One–Dimensional Heat Equation, Encyclopedia of Mathematics and Its Applications, 23 (1st ed.), Reading-Menlo Park–London–Don Mills–Sidney–Tokyo/ Cambridge–New York–New Rochelle–Melbourne–Sidney: Addison-Wesley Publishing Company/Cambridge University Press, pp. XXV+483, ISBN 978-0-521-30243-2, MR 0747979, Zbl 0567.35001.

[241] Pietro Perona and Jitendra Malik (November 1987). "Scale-space and edge detection using anisotropic diffusion". Proceedings of IEEE Computer Society Workshop on Computer Vision,. pp. 16–22.

[242] Rudin, L. I.; Osher, S.; Fatemi, E. (1992). "Nonlinear total variation based noise removal algorithms". Physica D. 60: 259–268. doi:10.1016/0167-2789(92)90242-f.

[243] Mumford, David; Shah, Jayant (1989), "Optimal Approximations by Piecewise Smooth Functions and Associated Variational Problems", Communications on Pure and

Applied Mathematics, XLII (5): 577–685, doi:10.1002/cpa.3160420503, MR 0997568, Zbl 0691.49036

[244] Y. Peleg, R. Pnini, E. Zaarur, Shaum's Outline of Theory and Problems of Quantum Mechanics, McGraw-Hill, 1998.

[245]-Yang, X. S. (2010). "A New Metaheuristic Bat-Inspired Algorithm, in: Nature Inspired Cooperative Strategies for Optimization (NISCO 2010)". Studies in Computational Intelligence. 284: 65–74. arXiv:1004.4170

[246]-Seeley, T. and Buhrman, S. (2001). Nest–site selection in honey bees: how well do swarms implement the"best-of-n" decision rule. Behavioural Ecology and Socio-biology, 49:416–427.

[247]-G. Rudolph, Convergence Properties of Evolutionary Algorithms, Verlag Dr. Kovac, Hamburg (1997)

[248]- http://homepages.inf.ed.ac.uk/rbf/CVonline/Imagedbase.htm

[249]- http://www.vision.ee.ethz.ch/datasets/

[250]- http://benchmarkfcns.xyz/benchmarkfcns/eggcratefcn.html

[251]- http://www.cvg.reading.ac.uk/slides/pets.html

[252]- Michael Swain and Dana Ballard. Color indexing. International Journal of Computer Vision, 7(1):11–32, 1991.

[253]- http://www.computervisiononline.com/datasets

[254]-Vincent, JFV, Bogatyreva, OA, Bogatyreva, NR (2006)- Biomimetics- its practice and theory. Journal of the Royal Society Interface 3 471-482.

[255]-Purves D, Lotto B (2011) Why We See What We Do Redux: A Wholly Empirical Theory of Vision (Sinauer Associates, Sunderland, MA)

[256]-S. Kannan, M. R. Slochanal, P. Subbaraj, and N. P. Padhy, "Application of particle swarm optimization technique and its variants to generation expansion planning problem," Elect. Power Syst. Res., vol. 70, no. 3, pp. 203–210, 2004.

[257]-Nguyen, Trung Thanh, Yang, Shengxiang and Branke, Jürgen. (2012) Evolutionary dynamic optimization : a survey of the state of the art. Swarm and Evolutionary Computation, Volume 6 . pp. 1-24. ISSN 2210-6502

[258]-M. Dorigo, ANTS' 98, From Ant Colonies to Artificial Ants : First International Workshop on Ant Colony Optimization, ANTS 98, Bruxelles, Belgique, octobre 1998.

[259]-http://www.samsung.com/uk/support/model/NP-N130-KA01UK

[260]-Reimer, B. (2014). Driver Assistance Systems and the Transition to Automated Vehicles: A Path to Increase Older Adult Safety and Mobility? Public Policy & Aging Report, 24(1), 27-31.

[261]-Lindeberg, T. and Bretzner, L. Real-time scale selection in hybrid multi-scale representations, Proc. Scale-Space'03, Isle of Skye, Scotland, Springer Lecture Notes in Computer Science, volume 2695, pages 148-163, 2003.

[262]-Davies, A., Yin, J. and Velastin, S., 1995. Crowd monitoring using image processing. IEEE Electronic and Communications Engineering Journal 7 (1), pp. 37–47.

[263]-Venkatesh.B., Perez,P,Bouthemy.P.,2003' Robust tracking with motion estimation and kernel-based colour modelling'. IRISA/INRIA Campus de Beaulieu 35042 Rennes Cedex, France

[264]-Klein, Lawrence A. (2004). Sensor and data fusion: A tool for information assessment and decision making. SPIE Press. p. 51. ISBN 0-8194-5435-4.

[265]-K.Kavitha, A.Tejaswini, Background Detection and Subtraction for Image Sequences in Video, International Journal of Computer Science and Information Technologies, vol.3, Issue 5, pp. 5223-5226, 2012

[266]-Genz, Alan (2009). Computation of Multivariate Normal and t Probabilities. Springer. ISBN 978-3-642-01689-9.

[267]-Bertram BCR (1979) Serengeti predators and their social systems. In: Sinclair ARE, Norton-Griffiths M (eds) Serengeti: dynamics of an ecosystem. University of Chicago Press, Chicago, pp 221-248

[268]-Crowley, James L., and Alice C. Parker, "A representation for shape based on peaks and ridges in the difference of lowpass transform," IEEE Trans. on Pattern Analysis and Machine Intelligence, 6, 2 (1984), pp. 156–170.

[269]-Eberhart, R. C. and Shi, Y. (1998). Comparison between genetic algorithms and particle swarm optimization. In Porto, V. W., Saravanan, N., Waagen, D., and Eibe, A., editors, Proceedings of the Seventh Annual Conference on Evolutionary Programming, pages 611–619. SpringerVerla

[270]-R.Allen and K. Kennedy. Optimizing Compilers for Modern Architectures. Morgan and Kaufman, 2002.

[271]-David Culler, J.P. Singh, and Anoop Gupta, Parallel Computer Architecture : A Hardware/Software Approach, Morgan Kaufmann, 1998.