International Journal of Multimedia and Ubiquitous Engineering
Vol.10, No.6 (2015), pp.53-76
http://dx.doi.org/10.14257/ijmue.2015.10.6.07

An Enriched 3D Trajectory Generated Equations for the Most Common Path of Multiple Object Tracking

Israa Hadi[1] and Mustafa Sabah[2]

[1]*First author, Professor College of Information Technology University of Babylon, Israa_hadi1968@yahoo.com*
[2]*Corresponding Author Ph.D. Student, College of Information Technology University of Babylon, muustafa_bayat@yahoo.com*

Abstract

Object tracking is important and challenging task in many computer vision applications such as surveillance, vehicle navigation, and autonomous robot navigation. Video surveillance in a dynamic environment, especially for humans and vehicles, is one of the current challenging research topics in computer vision. It is a key technology to fight against terrorism, crime, public safety and for efficient management of traffic.

In this paper, in order To get an accurate description of the trajectory points, regression analysis technique is used. This technique has the ability to summarize the collection of trajectory points by fitting it to mathematical models which will accurately describe these points and consequently describe object behavior. The regression analysis technique uses the least square method to obtain the best fit of equations for the given set of trajectory points. The least square method assumes that the best fit curve has the minimal sum of the deviations squared error from the given set of data.

In this paper, propose new method to deal with the trajectory by converting the trajectory points into 3D approximation function using best fit plane after interpolation the time factor this method offers high flexibility as well as statistical tools for the analysis behavior of object. Planar regression calculates the best fit plane through a group of 3 or more data points. The plane is calculated by minimizing the residuals (or errors) between the plane and the original points using least squares minimization. The objective of this paper was to develop methods for optimization of least square best fit geometry for planes.

Keywords*: Object tracking, curve fitting, 3D approximation function, and optimization*

1. Introduction

Moving object tracking in a video sequence is one of the most important areas of research in the field of computer vision. Typical applications include video surveillance, man machine communication, video indexing, automatic vehicle control and so forth. Generally, color, edge, texture and optical flow are used as features to identify an area inside a video sequence as an object. These features are highly influenced by the domain of the application. A moving object can be represented by an array of control points or by shape information of the object itself. In the last two decades, a good number of papers have reported in this domain. However, the problem is quite challenging due to the complex characteristics of the motion of an object in a natural video sequence. Moreover, it may be more complicated due to image noise, occlusions by other object(s) or the complex shape of the object [1]. On the other hand, object tracking can be mapped into an optimization problem. Recently, k-shortest path optimization [2] or evolutionary optimization (particle swarm optimization) [3] techniques have been used for multiple object tracking, cat swarm optimization [4].

ISSN: 1975-0080 IJMUE

The objective of curve fitting is to theoretically describe experimental data with a model (function or equation) and to find the parameters associated with this model. Models of primary importance to us are mechanistic models. Mechanistic models are specifically formulated to provide insight into a chemical, biological, or physical process that is thought to govern the phenomenon under study. Parameters derived from mechanistic models are quantitative estimates of real system properties (rate constants, dissociation constants, catalytic velocities *etc.*). It is important to distinguish mechanistic models from empirical models that are mathematical functions formulated to fit a particular curve but whose parameters do not necessarily correspond to a biological, chemical or physical property [5-6].

In fitting data with an approximating function, there are two basic approaches**.** The first approach involves passing an assumed function (preferably a polynomial**)** through every data point. However, such approaches suffer from three limitations such as unexpectedly large deviations from a smooth curve (wiggle**)**, an nth-order polynomial for n + 1 data points is too complex for large n values and experimental data are subject to errors and to pass a polynomial through every point is inappropriate.
Obviously, the first two of these limitations can be eliminated by using cubic splines, unfortunately, there is little that can be done to address the third of these problems. Alternatively, we may resort to the second approach. This involves assuming a function which best describes the shape of the curve representing the given data set, and having it pass as close as possible, but not necessarily through every data point [7].

Historians attribute the phrase regression analysis to Sir Francis Galton (1822-1911), a British anthropologist and meteorologist, who used the term regression in an address that was published in Nature in 1885. Galton used the term while talking of his discovery that offspring of seeds "did not tend to resemble their parent seeds in size, but to be always more mediocre [*i.e.,* more average] than they. The experiments showed further that the mean filial regression towards mediocrity was directly proportional to the parental deviation from it."

Regression Analysis refers to the study of the relationship between a response (dependent) variable, Y, and one or more independent variables, the X's. When this relationship is reasonably approximated by a straight line, it is said to be linear, and we talk of linear regression. When the relationship follows a curve, we call it curvilinear regression. Usually, you assume that the independent variables are measured exactly (without random error) while the dependent variable is measured with random error. Frequently, this assumption is not completely true, but when it cannot be justified, a much more complicated fitting procedure is required. However, if the size of the measurement error in an independent variable is small relative to the range of values of that variable, least squares regression analysis may be used with legitimacy.

In order to relate with a trajectory point as a function, Curve fitting function will be an essential component of any mathematical interface. The main advantage of such mathematical interfaces over other methods is that they can more naturally embody the usual two-dimensional mathematical notations. This method of evaluating empirical formulas was developed over a century ago, the procedure involves approximation a function such that the sum of the squares of differences between the approximation function and the actual function [5-7].

Change" Finally the overall paper is organized as follows: Section 2 summarizes studies that are related to the proposed algorithm. In Section 3 holds a brief description about the least square method. Section 4 provides backgrounds to curve fitting. Section 5 describe the linear regression. Section 6 holds a brief description about the CSO algorithm. Section 7 provides brief review to the parallel cat swarm optimization and Section 8 presents the concept of Average-Inertia Weighted CSO. Section 9 provides brief review to Fitness approximation method while Section 10 exposes the final proposed algorithm.

Section 11 demonstrates experimental results for the proposed approach over tested sequences and some conclusions are drawn in Section 12.

2. Related Work

Video tracking is an important topic in computer vision and it has been studied for several decades. Also mathematical model in Tracking is an important topic in computer vision and it has been studied for several decades. In this section some studies that related to proposed strategy have been summarized below:

Israa hadi and Mustafa sabah [8] present an improved algorithm by mixing two concepts, first concept found in parallel cat swarm optimization (PCSO) method for solving numerical optimization problems. The parallel cat swarm optimization (PCSO) method is an optimization algorithm designed to solve optimization problems Based on cats' cooperation and competition for improving the convergence of Cat Swarm Optimization,, the second concept found in Average-Inertia Weighted CSO (AICSO) by adding a new parameter to the velocity update equation as an inertia weight and used a new form of the position update equation in the tracing mode of algorithm. The performance of ICSO is sensitive to the control parameters selection.

Mrs. N. G. Chitaliya and Prof A. I .Trivedi [9] propose a novel, simple and fast block matching algorithm using predictive motion vector for object tracking. Color histogram is used for matching criteria for the motion tracking. The proposed algorithm is considered with a flexible size of block as well as pixel displacement.

Israa hadi and Mustafa sabah [10] Propose a new algorithm based on Hybrid Cat Swarm Optimization (HCSO) to reduce the number of search locations in the BM process. In proposed algorithm, the computation of search locations is drastically reduced by adopting a fitness calculation strategy which indicates when it is feasible to calculate or only estimate new search locations.

Kalyan Goswami, Gwang-Soo Hong and Byung-Gyu Kim [11] introduce a global mesh generation technique which can construct a hierarchical mesh on moving object(s) inside a video frame. This technique is efficient enough to solve the problem to a great extent and can identify a moving object in a video frame without any previous knowledge; the generated mesh is triangular and probabilistic in nature. The granularity of the mesh depends upon the level of hierarchy, neighborhood triangle decomposition, gradient and temporal information of the video sequence.

Darren Caulfield and Kenneth Dawson-Howe [9] propose an innovation to mean-shift tracking that combines the background exclusion constraint with multi-part appearance models. The former constraint prevents the tracker from moving to regions where no foreground objects are present, while the multi-part nature of the models enforces a spatial structure on the tracked object. Also a simple formula to determine the scale of the object in each video frame has been used, and notes the importance of setting an appropriate convergence condition.

Emilio Maggio and Andrea Cavallaro [12] present an effective target representation based on multiple colour histograms computed on semi-overlapping image areas. This solution introduces spatial information in the representation, without compromising the benefits of the histograms. In particular, target rotation and scaling can be accounted for, thus improving the tracker robustness to false targets. The proposed target representation outperforms the standard single histogram model and non-overlapping multi-part representations, using state-of-the-art tracking algorithms.

Chang hyun Choi, Seung-Min Baek and Sukhan Lee [13] demonstrates that a real-time solution of 3D pose estimation become feasible by combining a fast tracker such as KLT with a method of determining the 3D coordinates of tracking points on an object at the time of SIFT based tracking point initiation, assuming that a 3D geometric model with SIFT description of an object is known a-priori. Keeping track of tracking points with

KLT, removing the tracking point outliers automatically, and reinitiating the tracking points using SIFT once deteriorated, the 3D pose of an object can be estimated and tracked in real-time. This method can be applied to both mono and stereo camera based 3D pose estimation and tracking. The former guarantees higher frame rates with about 1 ms of local pose estimation, while the latter assures of more precise pose results but with about 16 ms of local pose estimation. The experimental investigations have shown the effectiveness of the proposed approach with real-time performance.

Marcus Baum and Uwe D. Hane beck [14], this paper is about tracking an extended object or a group target, which gives rise to a varying number of measurements from different measurement sources. For this purpose, the shape of the target is tracked in addition to its kinematics. The target extent is modeled with a new approach called Random Hyper surface Model (RHM) that assumes varying measurement sources to lie on scaled versions of the shape boundaries. In this paper, a star-convex RHM is introduced for tracking star convex shape approximations of targets. Bayesian inference for star-convex RHMs is performed by means of a Gaussian-assumed state estimator allowing for an efficient recursive closed-form measurement update. Simulations demonstrate the performance of this approach for typical extended object and group tracking scenarios.

Dong Wang, Huchuan Lu [15] In this paper propose a generative tracking method based on a novel robust linear regression algorithm. In contrast to existing methods, the proposed Least Soft threshold Squares (LSS) algorithm models the error term with the Gaussian Laplacian distribution, which can be solved efficiently.

Israa hadi and Mustafa sabah [6] propose new method to deal with the trajectory by converting the trajectory points into approximation function using curve fitting function to smooth the data, improve the appearance of the trajectory and extract important features such as slope, intersection point.

3. Least Squares Method

This method of evaluating empirical formulas was developed over a century ago and has been in use for many years. Like the method of cubic splines, it attempts to fit a simple function through a set of data points without the wiggle problem associated with high-order polynomials. Unlike the cubic splines technique, it presupposes that the derived functional relationship does not necessarily pass through every data point. The procedure involves approximating a function such that the sum of the squares of the differences between the approximating function and the actual functional values given by the data is a minimum. The basis for the method is represented graphically in Figure 1.

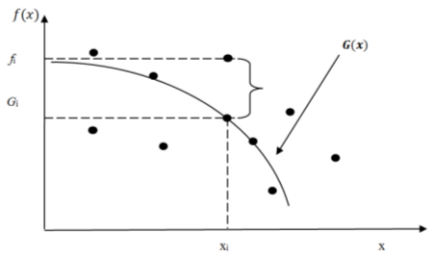


Figure 1. Graphical Interpretation of the Least-Squares Method

Here, given a set of N data points, we wish to fit an approximating function of the form G(x) having the following form:

$$G(x) = a_1 + g_1(x) + \ a_2 + g_2(x) + \cdots + a_m g_m(x) \qquad (1)$$

Where $m \leq N and g_1(x), \ldots, g_m(x)$ are assumed functions of the independent variable x. The problem is then to evaluate the regression coefficients $a_1, \ldots, a_m$. the method of least squares suggests that these can be easily calculated by minimizing a deviation function D define as follows:

$$D = \sum_{i=1}^{N} [f(x_i) - G(x_{i)}]^2 \tag{2}$$

Note that since one can fit different G functions through the data set then $a_1, \dots, a_m$may be looked at as variables. Therefore, using the method of calculus we can minimize the function given by equation (2) as shown below:

$$\frac{\partial D}{\partial a_1} = 0$$
$$\frac{\partial D}{\partial a_2} = 0 \tag{3}$$
$$\frac{\partial D}{\partial a_m} = 0$$

The set of mXmlinear algebraic equations given by equations (3) can therefore be solved for the unknowns $a_1, \dots, a_m$these coefficients are then substitutes into equation (1) to give the desired approximating function [6].

4. Curve Fitting

In many branches of applied mathematics and engineering sciences we come across experiments and problems, which involve two variables. For example, it is known that the speed *v* of a ship varies with the horsepower *p* of an engine according to the formula $p = a + bv^3$. Here a, b are the constants to be determined. For this purpose we take several sets of readings of speeds and the corresponding horse powers. The problem is to find the best values for *a, b* using the observed values of *v* and *p*.

Thus, the general problem is to find a suitable relation or law that may exist between the variables *x* and *y* from a given set of observed values (x_i, y_i), i = 1, 2,..........*n* Such a relation connecting *x* and *y* is known as empirical law, For above example, $x = v$ and $y = p$.

The process of finding the equation of the curve of best fit, which may be most suitable for predicting the unknown values, is known as curve fitting. Therefore, curve fitting means an exact relationship between two variables by algebraic equations. One of the most important aspects of regression (least squares) analysis is related to the question of what functional relationship G(x) should be assumed. Unfortunately, there are no clear criteria for determining the type of function that can best represent an arbitrary data set. However, the following rules should be observed whenever possible: first plot the data and look for obvious trends such as linear, quadratic, or higher-order behavior. This task can also be accomplished by examining the first few differences to see if higher differences tend to zero. If so, then a polynomial approximation may be appropriate. Second see if the data are symmetrical. Symmetry with respect to *f* may indicate polynomials of even powers only. Furthermore, symmetry can in some cases be achieved by transforming the data. Third consider periodicity. Trigonometric functions may be possible. Forth consider plotting the data on a semi log and/or log-log scale. This may provide information on whether logarithmic or exponential functions can be assumed. Finally break the data set into groups and consider the possibility of assuming different functions for the data subsets.

These rules are helpful in the absence of more suitable criteria, but should be used with discretion. Perhaps the most important rule is that of common sense. That is, if the given data are related to a particular physical phenomenon for which a functional relationship is known, then it makes no sense to assume something different. For example, when examining the relationship between stress and strain for a steel specimen, we know that the initial portion of this relationship is linear and fully described by Hooke's law.

Therefore, if the experimental data indicate nonlinear behavior initially, then the data are in error and a linear function must assume [5, 6, 16].

5. Linear Regression

In order to fully understand the method of least squares and be able to interpret the deviation function given by equation (2) let us consider figure 2.Given a set of N data points, a linear function is assumed to exist between the dependent variable $f$and the independent variable x. That is,

$$G(x) = a_1 + a_2 x \tag{4}$$

Where:

$g_1(x) = 1\ and g_2(x) = x$.

It is evident that many straight lines can be fitted through any two of the data points. Such an arbitrary procedure will undoubtedly be biased and dependent on individual preference

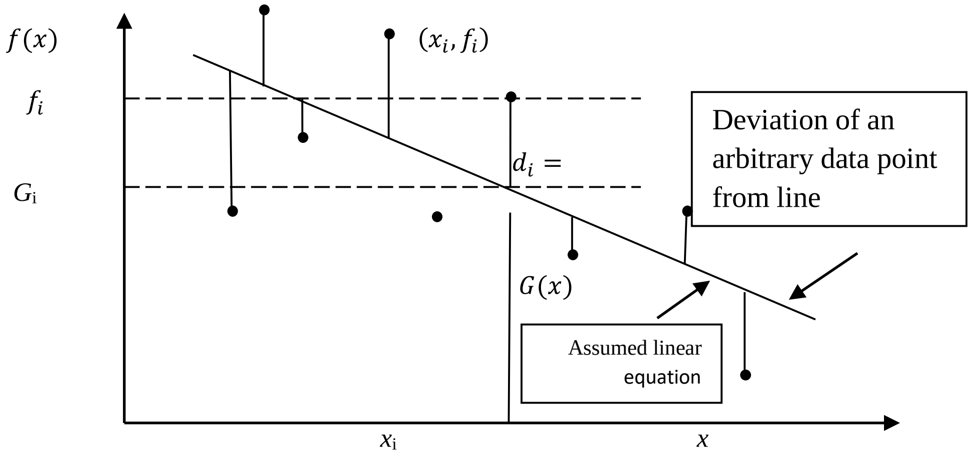


Figure 2. Schematic Representation of Linear Regression

In fact, there are $\boldsymbol{S}$linear equations that can be determined where

$$S = \frac{N!}{2!\,(N-2)!}$$

Where N is the number of data points. The implication is that there are S different sets of values for $a_1 and a_2$that can be evaluated. This implies those $a_1 and\ a_2$ are indeed variables and must be fixed in such a way that the derived function is the best possible solution to the problem. Such solutions will have the smallest deviations $d_i$from the actual data. The set of deviations is given by:

$$d_1 = f_1 - G(x_1) = f_1 - a_1 - a_2x_1$$
$$d_2 = f_2 - G(x_2) = f_2 - a_1 - a_2x_2$$
$$\vdots$$
$$d_i = f_i - G(x_i)\ = f_i - a_1 - a_2x_i$$
$$\vdots$$
$$d_N = f_N - G(x_N) = f_N - a_1 - a_2x_N$$

The object is to minimize the sum of all of the deviations. Unfortunately, this is not possible because the negative deviations will reduce the positive ones. However, if the sum of the squares of the deviations is determined then a true estimate of the total deviation is possible. This is precisely what equation (2) states. Consequently for an assumed linear approximation we have:

$$d^2 = D = \sum_{i=1}^{N}[f(x_i) - G(x_i)]^2 = \sum_{i=1}^{N}(f_i - a_1 - a_2x_i)^2 \tag{5}$$

Note that $f(x_i) = f_i and G(x_i) = a_1 + a_2x_i$. Differentiating with respect to a_1 and a_2 then setting the resulting expressions to zero gives:

$$\frac{\partial D}{\partial a_2} = 0 = 2\sum_{i=1}^{N}(f_i - a_1 - a_2 x_i)(-x_i)$$
$$\frac{\partial D}{\partial a_1} = 0 = 2\sum_{i=1}^{N}(f_i - a_1 - a_2 x_i)(-1)$$

Simplifying and rearranging yields the following set of 2 x 2 linear algebraic equations in the unknownsa_1 $and a_2$:

$$\sum_{i=1}^{N} f_i = a_1 N + a_2 \sum_{i=1}^{N} x_i \qquad (6a)$$
$$\sum_{i=1}^{N} f_i x_i = a_1 \sum_{i=1}^{N} x_i + a_2 \sum_{i=1}^{N} {x_i}^2 \qquad (6b)$$

Equation (6) can therefore be solved to give

$$a_1 = \frac{\sum f_i \sum {x_i}^2 - \sum x_i \sum f_i x_i}{N \sum {x_i}^2 - (\sum x_i)^2} \qquad (7a)$$

$$a_2 = \frac{N \sum f_i x_i - \sum x_i \sum f_i}{N \sum {x_i}^2 - (\sum x_i)^2} \qquad (7b)$$

Note that for simplicity the limits have been omitted from the sums in equation (7). It is evident that the assumed linear functional relationship is now completely defined in terms of the given data set [6, 16, 17].

6. Cat Swarm Optimization (CSO)

Swarm Intelligence (SI) is a novel artificial intelligence approach inspired by the swarming behaviors of groups of organisms such as ants, termites, bees, birds, fishes in foraging and sharing the information with each other. SI focuses on the collective intelligence of a decentralized system consisting of a group of organisms interacting with each other and their environment. So, by means of their collective intelligence swarms are able to effectively use their environment and resources. SI is also a mechanism that enables individuals to overcome their cognitive limitations and solve problems which are difficult for individuals to resolve alone. Swarm intelligence algorithms are essentially stochastic search and optimization techniques and were developed by simulating the intelligent behavior of these organisms. These algorithms are known to be efficient, adaptive, robust, and produce near optimal solutions and utilize implicit parallelism approaches [8, 10, 18].

One of the more recent optimization algorithm based on swarm intelligence is the Cat Swarm Optimization (CSO) algorithm. The CSO algorithm was developed based on the common behavior of cats. It has been found that cats spend most of their time resting and observing their environment rather that running after things as this leads to excessive use of energy resources. To reflect these two important behavioral characteristics of the cats, the algorithm is divided into two sub-modes and CSO refers to these behavioral characteristics as ―seeking mode‖ and ―tracing mode‖, which represent two different procedures in the algorithm. Tracing mode models the behavior of the cats when running after a target while the seeking mode models the behavior of the cats when resting and observing their environment [19].

Furthermore, previous researches have shown that the CSO algorithm has a better performance in function minimization problems compared to the other similar optimization algorithms like Particle Swarm Optimization (PSO) and weighted-PSO [20].

Cat Swarm Optimization algorithm has two modes in order to solve the problems which are described below:

6.1. Seeking Mode: Resting and Observing

For modeling the behavior of cats in resting time and being-alert, we use the seeking mode. This mode is a time for thinking and deciding about next move. This mode has four main parameters which are mentioned as follow:

Seeking memory pool (**SMP**), seeking range of the selected dimension (**SRD**), counts of dimension to change (**CDC**) and self-position consideration (**SPC**) [21]. The process of seeking mode is describes as follow:

Step1: Make j copies of the present position of catk, where j = SMP. If the value of SPC is true, let j = (SMP-1), then retain the present position as one of the candidates.

Step2: For each copy, according to CDC, randomly plus or minus SRD percent the present values and replace the old ones.

Step3: Calculate the fitness values (FS) of all candidate points.

Step4: If all FS are not exactly equal, calculate the selecting probability of each candidate point by (8); otherwise set all the selecting probability of each candidate point is 1.

Step5: Randomly pick the point to move to from the candidate points, and replace the position of catk.

$$Pi = \frac{|\mathrm{SSEi} - \mathrm{SSEmax}|}{\mathrm{SSEmax} - \mathrm{SSEmin}} \tag{8}$$

If the goal of the fitness function is to find the minimum solution, FSb = FSmax, otherwise FSb = FSmin

6.2 Tracing Mode: Running After a Target

Tracing mode is the second mode of algorithm. In this mode, cats desire to trace targets and foods. The process of tracing mode can be described as follow: [22]

Step1: Update the velocities for every dimension according to (9).

Step2: Check if the velocities are in the range of maximum velocity. In case the new velocity is over-range, it is set equal to the limit.

$$V_{k,d} = V_{k,d} + r_1 c_1 (X_{best} - X_{k,d}) \tag{9}$$

Step 3: Update the position of cat k according to (10).

$$X_{k,d} = X_{k,d} + V_{k,d} \tag{10}$$

$X_{best,d}$ is the position of the cat, who has the best fitness value, $X_{k,d}$ is the position of cat_k , c_1 is an acceleration coefficient for extending the velocity of the cat to move in the solution space and usually is equal to 2.05 and r_1 is a random value uniformly generated in the range of [0,1].

7. Parallel Cat Swarm Optimization (PCSO)

Tsai *et al.* [23] have proposed the parallel cat swarm optimization (PCSO) method for solving optimization problems. The basic idea of the PCSO method utilizes the major structure of the cat swarm optimization (CSO) method proposed by Chu *et al.* [24]. The CSO method has two modes, *i.e.*, the seeking mode and the tracing mode, for simulating the behaviors of cats to move the individuals in the solution space. By adjusting the parameter MR, the ratio of individuals moved by the seeking process and the tracing process can be controlled, where MR 2 [0, 1]. Some methods for splitting a population into several sub-populations to construct a parallel structure have been presented, such as HCSO algorithm [10], IHCSO algorithm [8], the parallel genetic algorithm [25], the ant colony system with communication strategies [26] and the parallel particle swarm optimization algorithm with communication strategies [27]. Each of the sub-populations evolves independently and shares the information they have occasionally. It results in the reducing of the population size for each sub-population and the benefit of cooperation is achieved.

In the PCSO method, the individuals are separated into a predefined number of groups in the initial process to construct the virtual parallel space for the individuals. If we let the predefined number of groups be equal to 1, then the PCSO method becomes the CSO method due to the fact that there is only one group. The individuals in the same group provide a local near best solution for their group in every generation, and the global near best solution found so far can be discovered by comparing the local near best solutions collected from the parallel groups. The individuals in a group can only access the near best solution discovered by their own group, but when the process of information exchanging is applied, the parallel groups can receive a near best solution from another randomly picked group. The difference between the PCSO method and the CSO method is described as follows. At the beginning of the PCSO method, N individuals are created and then they are separated into G groups. The calculation of the PCSO method in the tracing mode is different from that of the CSO method and there exists an information exchanging process. The parallel cat swarm optimization (PCSO) method is an optimization algorithm designed to solve numerical optimization problems under the conditions of a small population size and a few iteration numbers.

7.1 Parallel Tracing Mode Process

Since the virtual cats are divided into isolated groups, they can be treated as groups of small-scale CSO clusters. Agents in different clusters should only share their own near best solution. Thus, in the parallel tracing mode process has the following steps:

Step 1: Update the velocities for every dimension $v_{k,d}(t)$ for the cat_k at the current iteration, according to Eq. (11):

$$V_{k,d}= V_{k,d}(t-1) + r_1c_1(X_{lbest,d}(t-1) - X_{k,d}(t-1)),\ d=1,2,\ldots\ldots,M \tag{11}$$

Where $X_{lbest,d}$ denotes the coordinates of the near best solution in one cluster.

Step 2: Check whether the velocities are in the range of maximum velocity. The new velocity is bounded to the maximum velocity in case the new velocity is over-range.

Step 3: Update the position of cat_k according to Eq. (12):

$$X_{k,d} = X_{k,d}(t-1) + V_{k,d} \tag{12}$$

7.2 Information Exchanging Process

In the information exchanging process, the near best solutions may have change to be copied into different clusters. A parameter called *ECH* is defined to trigger off the information exchanging process. Hence, in PCSO, the information exchanging process is involved every *ECH* iterations. This process can be described in 3 steps:

Step 1: Sort the virtual cats for every cluster by their fitness values.

Step 2: Randomly pick a near best solution from all clusters and replace the virtual cat, which has the worst fitness value in the cluster. But the near best solution and the virtual cat should not come from the same cluster.

Step 3: Repeat step 2 for all clusters

8. Average-Inertia Weighted Cat Swarm Optimization (AICSO)

In the pure CSO, a condition on the velocity equation should be put in order to control the velocities of the cats for every dimension and check whether the velocities are in the range of maximum or not.

For modifying this part, a parameter as an inertia weight to handle this problem will be used. Here the value of inertia weight (w) will be chosen randomly and experimental results indicate that it is better to choose w in the range of [0.4, 0.9].

So selecting the largest value for w in the first iteration (w = 0.9) and then it will be reduced to 0.4 in the next iterations.CSO with inertia weight can converge under certain conditions even without using v_{max}.

For w>1, velocities increase over time, causing cats to diverge eventually beyond the boundaries of the search space. For w<1, velocities decrease over time, eventually reaching 0, resulting convergence behavior. So the new position update equation can be written as

$$V_{k,d}= WV_{k,d} +r_1c_1(X_{best}-X_{k,d}) \quad (13)$$

Where $c_1$is acceleration coefficient and usually is equal to 2.05 and $r_1$is a random value uniformly generated in the range of [0, 1] and w is inertia weight (ICSO).

Next step, a new form of the position update equation composing two terms will be used. In the first term, the average information of current and previous position and in the second, the average of current and previous velocity information will be used(AICSO). So new position equation is described below: [28]

$$.X_{i+1} = \frac{X_{i+1}+X_i}{2} + \frac{V_{i+1}+V_i}{2} \quad (14)$$

9. Fitness Approximation Method

Evolutionary algorithms that use fitness approximation aim to find the global minimum of a given function by considering only a small number of function evaluations and a large number of estimations. Such algorithms commonly employ alternative models of the function landscape in order to approximate the actual fitness function. The application of this method requires that the objective function fulfills two conditions: a heavy computational overhead and a small number of dimensions (up to five) [29].

Recently, several fitness estimators have been reported in the literature [30–32] in which the number of function evaluations is considerably reduced to hundreds, dozens or even less. However, most of these methods produce complex algorithms whose performance is conditioned to the quality of the training phase and the learning algorithm in the construction of the approximation model.

In this paper, we explore the use of a local approximation scheme based on the nearest-neighbor-interpolation (NNI) for reducing the function evaluation number. The model estimates fitness values based on previously evaluated neighboring individuals which have been stored during the evolution process. At each generation, some individuals of the population are evaluated through the accurate (actual) fitness function while other remaining individuals are only estimated. The positions to be accurately evaluated are determined either by their proximity to the best individual or regarding their uncertain fitness value.

9.1 Updating the Individual Database

In a fitness approximation method, every evaluation or estimation of an individual produces one data point (individual position and fitness value) that is potentially considered for building the approximation model during the evolution process. In the proposed approach, all seen-so-far evaluations are kept in a history array **T** which is employed to select the closest neighbor and to estimate the fitness value of a newer individual. Since all data are preserved and potentially available for their use, the model construction is faster because only the most relevant data points are actually used by the approach.

9.2 Fitness Calculation Strategy

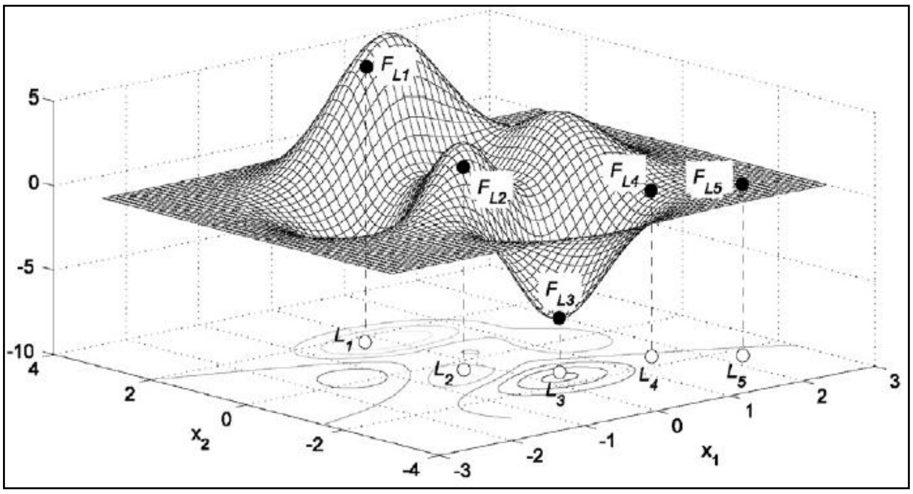
This section discusses details about the strategy to decide which individuals are to be evaluated or estimated. The proposed fitness calculation scheme estimates most of fitness values to reduce the computational overhead at each generation. In the model, those individuals lying nearer to the best fitness value holder, currently registered in the array **T** (step 1), are evaluated by using the actual fitness function. Such individuals are relevant as they possess a stronger influence on the evolution process than others. On the other hand, evaluation is also compulsory for those individuals lying in a region of the search space which has been unexplored so far (step 2). The fitness values for such individuals are uncertain since there is no close reference (close points contained in **T**) to calculate their estimates. The rest of the individuals, lying in a region of the search space that contains enough previously calculated points, must be estimated using the NNI (step 3). This rule indicates that the fitness value for such individuals must be estimated by assigning the fitness value from the nearest individual stored in **T**. Therefore, the fitness calculation model follows three important rules to evaluate or estimate fitness values:

Rule 1: Exploitation rule (evaluation): If a new individual (search position) P is located closer than a distance d with respect to the nearest individual Lq (q = 1, 2, 3, . . . , m; where m is the number of elements contained in **T**) with a fitness value FLq that corresponds to the best fitness value seen-so-far, then the fitness value of P is evaluated using the actual fitness function. Figure. 3b draws this rule procedure.

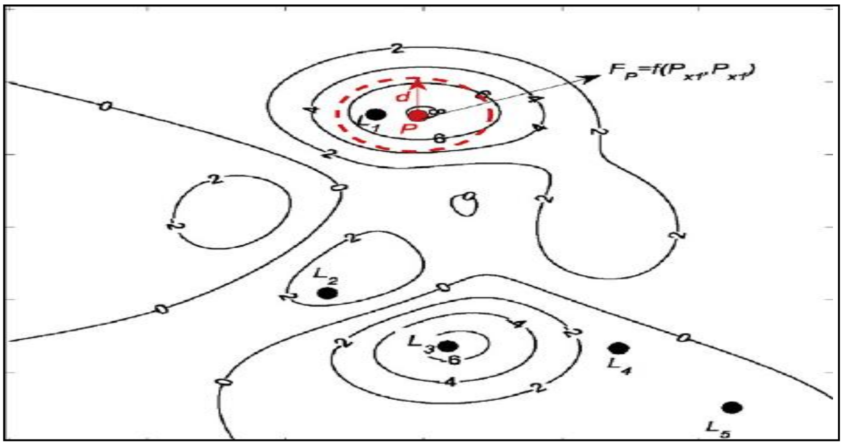
Rule 2: Exploration rule (evaluation): If a new individual P is located further away than a distance d with respect to the nearest individual Lq, then its fitness value is evaluated by using the actual fitness function. Figure. 3c outlines the rule procedure.

Rule 3:NNI rule (estimation): If a new individual P is located closer than a distance d with respect to the nearest individual Lq, whose fitness value FLq does not correspond to the best fitness value, then its fitness value is estimated by assigning the same fitness that Lq (FP = FLq). Figure. 3d sketches the rule procedure.

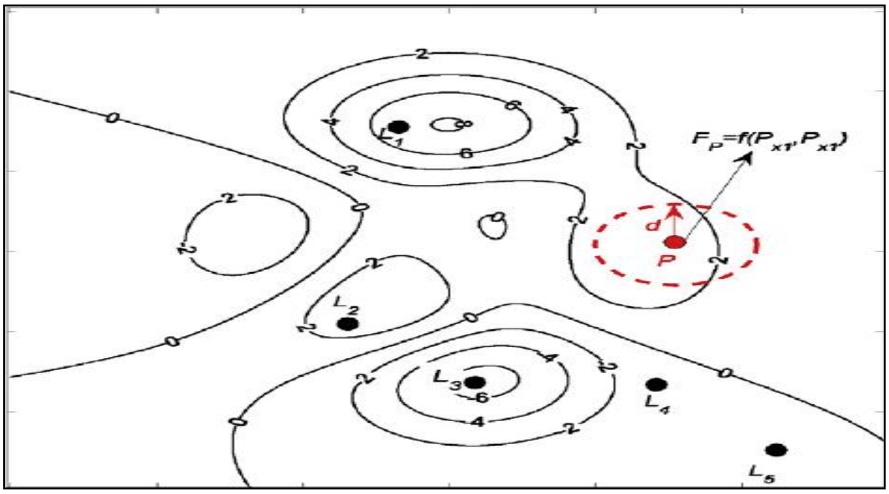
The d value controls the trade-off between the evaluation and the estimation of search locations. Typical values of d range from 2 to 4. Figure.3 illustrates the procedure of fitness computation for a new solution (point P). In the problem (Figure. 1a), it is considered the fitness function f with respect to two parameters (x1, x2), where the individuals database array **T** contains five different elements (L1 - L5) and their corresponding fitness values (FL1 - FL5). Figures. 3(b) and (c) shows the fitness evaluation (f (x1, x2)) of the new solution P, following the step 1 and step 2 respectively, whereas Figure. 3(d) presents the fitness estimation of P using the NNI approach which has been laid by step 3 [34].



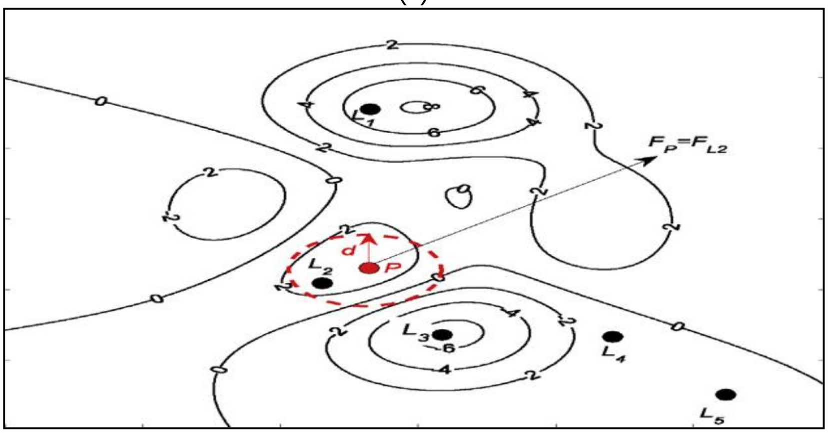
(a)



(b)



(c)



(d)

Figure 3. The Fitness Calculation Strategy. (a) Fitness Function and the History Array T Content. (b) According to the Rule 1, the Individual (search position) P is Evaluated as it is Located Closer than a Distance d with Respect to the Best Individual L1. (c) According to the Rule 2, the Search Point P is Evaluated because there is no Reference within its Neighborhood. (d) According to Rule 3, the Fitness Value of P is Estimated by the NNI-Estimator, Assigning FP = FL2.

10. Proposed Algorithm

In This proposed algorithm, the block matching will not be used, but some important characteristics from block matching will be extracted and used such as search window and matching criteria concept.
In order to achieve the tracking technique in efficient manner, there are some important issues, first the modify EHCSO will be used that decrease overall computational complexity and increase the accuracy of motion estimation. Second issues reduced the span of search window in order to find best match. A third issue is used accumulated histogram as a new cost function to find the best match. also a multi part object representation will be used in order to achieve the robustness to partial occlusions, finally3D curve fitting will be used in order to build an equation that describe the flow of most common path of multiple objects.

10.1 Multi –Part Object Representation

A histogram consider as a sample set of a population with respect to a measurement represents the frequency of quantized values of that measurement among the samples. Finding the distance, or similarity, between two histograms is an important issue in computer vision. A number of measures for computing the distance have been used.

There are two classifications in histogram distance measures: vector and probabilistic. In the vector approach, a histogram is treated as a fixed-dimensional vector, Euclidean or intersection can be used as distance measures.

The probabilistic approach is based on the fact that a histogram of a measurement provides the basis for an empirical estimate of the probability density function (pdf) (spectral information). This is equivalent to measuring the overlap between two pdf's as the distance. There is much literature regarding the distance between pdfs, an early one being the Bhattacharyya distance between statistical populations. Bhattacharyya coefficient is a popular method that uses color histograms to correlate images or two sections of images, and is believed to be the absolute similarity measure for frequency coded data and it needs no bias correction.

In this paper the accumulation histogram will be used to describe the target region moving characteristics instead of the general histogram. The accumulation histogram reflecting the relationship between distance in color-axes and the similarity of color distributions, distinguishes colors more accurately, in the target tracking domain, accumulation histograms are proposed form of target representation, because of their independence from scaling and rotation, and robustness to partial occlusions.

In this paper an effective approach based on a multi-part model will be used. The target is divided into four regions in order to increase the tracker sensitivity to rotations and anisotropic scale changes this solution introduces spatial information in the representation and to reflect real moving object by determine one distinguish feature for each region, So the model to be tracked should be represented by a number of histograms (Multi-part appearance model), so that some element of the spatial layout of the object to be tracked is recorded thus improving the tracker robustness to false targets because the lack of spatial information in histograms can be a problem in color based tracking The tracker is attracted to false targets with similar color distributions. So the information related to the spatial distribution of the colors is fundamental for a correct tracking.

10.2 The Search Algorithm

The processing of block matching is looking for the best position within the search window, in which a point of the minimum of MSE needs to be found. In order to reaching a better MSE, the more positions within the search window will be matched; however, the more computation times will be spent on searching. A better matching algorithm should spend less computation time on searching and obtain the better position. In this paper, the aim of the application of the HCSO algorithm to ME is to accelerate matching search, obtain higher accuracy, faster computation speed and reach a better ME. Strong curiosity to moving objects and the outstanding skill of hunting are the two distinctive features of a cat. These two behavioral traits of cats are modeled by CSO: seeking mode and tracing mode, which reflects the cooperation between "cats". However, in order to further improve the CSO optimization speed and prediction accuracy, HCSO absorbs the advantage of parallel computing to improve the tracing mode such that a parallel tracing mode is adopted. HCSO establishes a plurality of CSO to search the best parameters in the prediction the next block independently and simultaneously by dividing the "cat swarm" into some groups. At the same time, it adds information exchanging mode such that the CSOs can exchange information occasionally, which reflects the cooperation between groups. The information exchanging process aims to share the isolated near best solution between different groups of virtual cats. Hence, HCSO is particularly suitable for optimization problems, because it makes full use of computer resources and obtains the optimal result quickly. When HCSO is running, the "cats" are randomly distributed in the prediction search space. Inevitability, it results in a state such that there more "cats" in some areas and less in others. But sometimes in some cases pure CSO takes a long time to find an acceptable solution. So it affects on performance and convergence of the algorithm. Therefore high speed processor is needed for getting reasonable result.

In this study, proposed a new algorithm (HCSO) in order to improve the performance and achieve better convergence in less iteration. By adding a new parameter to the position equation as inertia weight that will be chosen randomly, then by making a new form of the velocity equation to improve searching ability in the vicinity of the best cats. By using this parameter, a balance between global and local search ability can be made. A large inertia weight facilitates a global search while a small inertia weigh facilitates a local search. First a large value will be used and it will be reduced gradually to the least value. So the maximum inertia weight happens in the first dimension of the each iteration and it will be updated decreasingly in every dimension, the velocity update equation for each cat to a new form can be changed. Also the proposed fitness calculation strategy, seen from an optimization perspective, favors the exploitation and exploration in the search process.

For the exploration, the method evaluates the fitness function of new search locations which have been located far away from previously calculated positions. Additionally, it also estimates those which are closer. For the exploitation, the proposed method evaluates the actual fitness function of those new individuals which are located nearby the position that holds the minimum fitness value seen-so-far, aiming to improve its minimum. After several simulations, the value of $d = 3$ has shown the best balance between the exploration and exploitation inside the search space (in the context of a BM application); thus it has been used in this paper.

The enhanced HCSO optimization method declares the incorporation of the fitness calculation strategy to the CSO algorithm is presented. Only the fitness calculation scheme shows the difference between the conventional HCSO and the enhanced approach. In the modified HCSO, only some individuals are actually evaluated (rules 1 and 2) at each generation. The fitness values for the rest are estimated using the NNI-approach (rule 3).

The estimation is executed using individuals that have been already stored in the array **T**.

The proposed algorithm based on EHCSO for ME is summarized as follows:

Step1: A population of four groups cats is generated with random positions within the searching window in the previous frame, the search area called search window which is usually a region centered on the current block position; and then random velocities are assigned to each cat, initialize the individuals database array T as an empty array.

Step 2: pick number of cats and set them into tracing mode according to *MR*, and the others set into seeking mode.

Step 3: The fitness of each groups are then evaluated according to the objective function. In the processing of block matching, the MSE as the (matching criterion) will be chosen.

Step 4: Evaluate the fitness value of each group cat by applying the positions of cats into the fitness function, which represents the criteria of our goal, and keep the best cat into T.

Step 5: Since all individuals of the initial population fulfill rule 2 conditions, they are evaluated through the actual fitness function by calculating the actual MSE values.

Step 6: Pick up a group of cats sequentially and sort the cats in this group according to their fitness values.

Step 7: Update new evaluations in the individual database array **T**.

Step 8: Move the cats according to their *flags*, if cat_k is in seeking mode, apply the cat to the seeking mode process, otherwise apply it to the tracing mode process.

Step 9: Compute fitness values for each copy in seeking mode by using the fitness calculation strategy presented in Section 7

Step 10: Set the CDC value for each group into 25% because each group responsible for three direction only from proposed 12 direction sets.

Step 11: Each time, choose the inertia weight (w) randomly in range of [0.4, 0.9] in order to controlling excessive roaming of cats outside the searching window.

Step 12: Pick the near best solution from the neighbor group and replace the virtual cat, which has the worst fitness value in the group that appear in array T. But the near best solution and the virtual cat should not come from the same group.

Step 13: Repeat steps (9 -13) for all groups.

Step 14: Each time in the parallel tracing mode process, the velocity update step as declare in equation (13) instead of equations 3, 5, 6:

$$V_{k,d}= V_{k,d}(t-1) +r_1c_1(X_{lbest,d}(t-1) - X_{k,d}(t-1)),\ d=1,2,\ldots\ldots,M \qquad (15)$$

Where $X_{lbest,d}$ denotes the coordinates of the near best solution in one cluster.

Step 15: use a new form of the position update equation composing two terms. In the first term, the average information of current and previous position and in the second, the average of current and previous velocity information will be used. So new position equation is described below:

$$X_t = \frac{X_t+X_{t-1}}{2} + \frac{V_t+V_{t-1}}{2} \qquad (16)$$

Step 16: Termination criteria. If the number of iteration equals to the maximum (I_{max}), or MSE of the matching less than a given small number ε, then iteration terminate; Otherwise go back to step 3.

10.3. Curve Fitting

In this part, in order To get an accurate description of the trajectory points, regression analysis technique is used. This technique has the ability to summarize the collection of trajectory points by fitting it to mathematical models which will accurately describe these points and consequently describe object behavior. The regression analysis technique uses the least square method to obtain the best fit of equations for the given set of trajectory points. The least square method assumes that the best fit curve has the minimal sum of the deviations squared error from the given set of data.

In this part, propose new method to deal with the trajectory by converting the trajectory points into 3D approximation function using best fit plane after interpolation the time factor this method offers high flexibility as well as statistical tools for the analysis behavior of object.

The result equation that identify the flow of the path of moving multi object can support the analyzer to study the be behaviours of traffic such as bottlenecks without need to train the huge set of date. This step increase the capability and flexibility of the analysis method under multi situation such as overlap of motion objects is present and full or partial occlusion.

3D Curve fitting is one of the mainly influential analysis tool. so it examines a relationship among one or more predictors alternatively called independent variables as well as a response variable alternatively called dependent variable.

A planar regression calculates the best fit plane during groups of 3 data points. A plane is intended by minimizing a residuals or errors among the plane and the original points by procedure called least squares minimization pro. In this part the three parameters was the x, y position of the object and z represent the time factor, as shown in Figure 4 below.

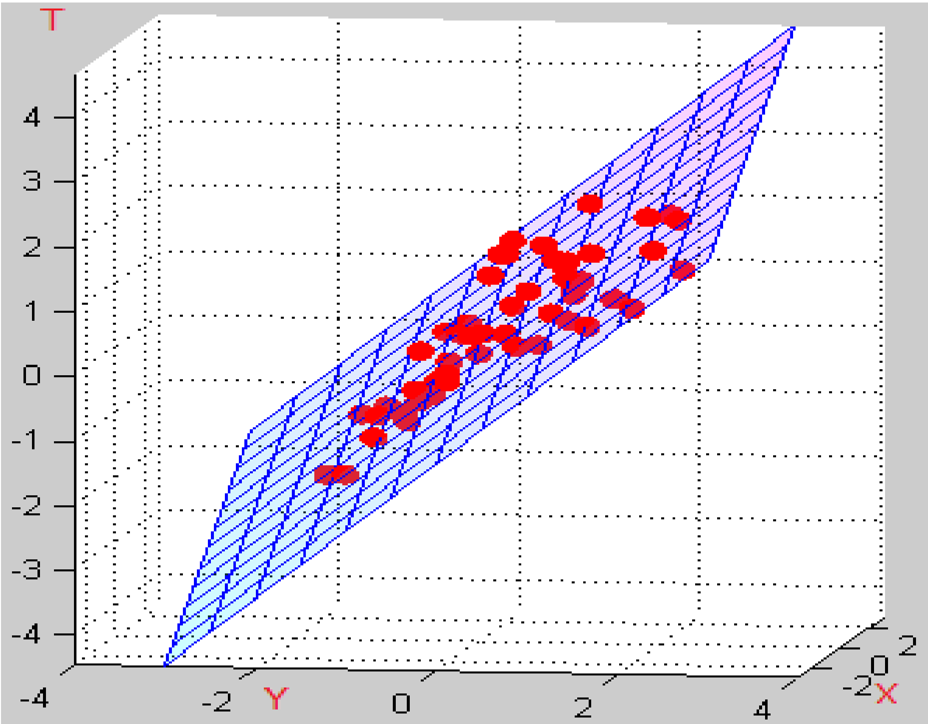


Figure 4 : A Plane of 3D Data Sets

The least squares minimization equations is represented as follow:

$$\varepsilon^2=\sum_{i=1}^{n}(f(x_i,y_i)-z_i)^2 \qquad (17)$$

Where z_i are the observed values (time parameter of moving object through the video) as well as f (x_i, y_i) is the y value of a surface at x_i,y_i. The equation 18 of the plane is:

$$Z=Ax+By+C \qquad (18)$$

Putting this value in to the regression equation gives the equation 19 below:

$$\varepsilon^2 = \sum_{i=1}^{n}(Ax_i + By_i + C - z_i)^2 \qquad (19)$$

In order to discover a minimum residual error, the derivative of the residuals equation should be equal to zero, which means all of a partial derivatives with respect to every coefficient should be equal to zero as shown in equations 20 - 22 below:

$$\frac{d^{\varepsilon\varepsilon}}{dA}=\sum_{i=1}^{n}(Ax_i + By_i + C - z_i)\cdot x_i = 0 \qquad (20)$$

$$\frac{d^{\varepsilon\varepsilon}}{dB}=\sum_{i=1}^{n}(Ax_i + By_i + C - z_i)\cdot y_i = 0 \qquad (21)$$

$$\frac{d^{\varepsilon\varepsilon}}{dC}=\sum_{i=1}^{n}(Ax_i + By_i + C - z_i) = 0 \qquad (22)$$

These equations be able to be expressed in matrix shape:

$$\begin{bmatrix} \sum x_{i^2} & \sum x_i y_i & \sum x_i \\ \sum x_i y_i & \sum y_{i^2} & \sum y_i \\ \sum x_i & \sum y_i & \sum 1 \end{bmatrix} \cdot \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} \sum x_i z_i \\ \sum y_i z_i \\ \sum z_i \end{bmatrix}$$

Solving for

$$\begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} \sum x_{i^2} & \sum x_i y_i & \sum x_i \\ \sum x_i y_i & \sum y_{i^2} & \sum y_i \\ \sum x_i & \sum y_i & \sum 1 \end{bmatrix}^{-1} \cdot \begin{bmatrix} \sum x_i z_i \\ \sum y_i z_i \\ \sum z_i \end{bmatrix}$$

Therefore these parameter will be interpolated into main equation to build the flow of the path equation.

10.4 The Main Algorithm

The main algorithmic steps can be summarized as follows:

1) Define a rectangle block on the region of interest object R0 in the first frame of a video sequence.
2) The target is divided into four regions $R0_1$, $R0_2$, $R0_3$ and $R0_4$
3) Compute the Accumulated histogram h1 for each region of the target (without included the background for each region)
4) Apply Enhanced Hybrid Cat Swarm Optimization Based on Fitness Approximation Method algorithm to find the search region, in this paper the new search pattern(according to the direction set) will be proposed in order to limit the CDC parameter in seeking mode, the new search pattern will be shown in Figure 4 below.
5) Divide the overall cats into 4 groups, each group work in specific part so reducing the search points. The algorithm chooses as search points only those locations that iteratively minimize the error-function (accumulated histogram value).
6) Given a small square region of fixed size W (*e.g.* W = 3) around the rectangle block on the region of interest object, the EHCS algorithm search the new locations in 12 directions. These directions according to search pattern that was shown in Figure 5below.
7) Each group of cats search in 3 direction only, as example the group of cats that is located in $R0_1$ search only in (DIR 1, DIR 2 and DIR 1) in order to reduce the computational complexity because the span of search window to find the best match will be reduced.
8) Displacement value of original location pixel, then form a new rectangle block and check appropriate similarity part for new detected locations using similarity measure by Histogram matching using Bhattacharya coefficient is applied

$$\rho\,[p,\mathrm{q}] = \sum_{j=1}^{Nb} \sqrt{p^{(j)}\ .\ q^{(j)}} \tag{23}$$

Where $p^{(j)}$ and $q^{(j)}$ are the model and candidate accumulated histograms and the larger ρ is, the more similar the distributions are. This step leads to Decrease the computational overhead for every search point where the matching cost (SAD operation) is replaced by an accumulated histogram this feature will be less complexity. Since only a fraction of pixels enters into the matching computation, the use of such regular sub-sampling techniques can seriously affect the accuracy of the detection of motion vectors due to noise or illumination changes.

9) Repeat the steps (4-8) for all groups until the suitable part is found.
10) The object will be considered as a static if the matching error is zero between the current region and same region in reference frame.
11) Check the global accumulated histogram of all regions in target in order to achieve the reliability.

12) $\rho\,[p,q] = \frac{\sum_{j=1}^{N} p[p^{(j)}, q^{(j)}]}{N}$ (24)
13) Where N is the number of regions, $p^{(j)}$ and $q^{(j)}$ are the model and candidate accumulated histograms .
14) Store all date sets of al l moving objects.
15) Set for each direction a probability ratio depending on the number of the path (s) that included for each direction.
16) Apply 3D curve fitting to the direction that has superior probability than others in order to built an a non linear equation for the flow of most common path of multiple objects.

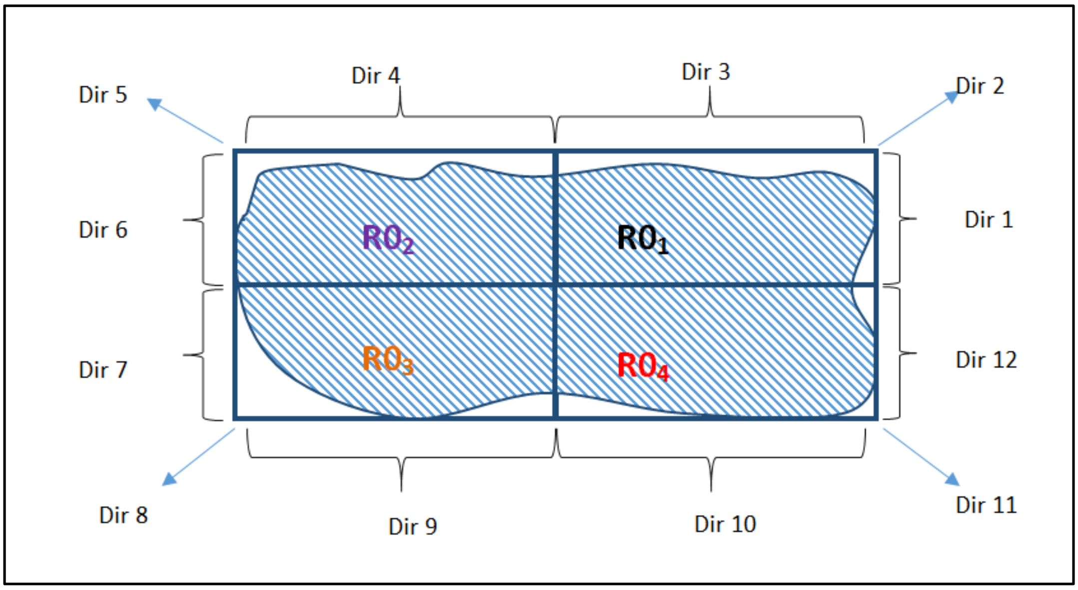


Figure 5. Proposed Model

11. Simulation Results

To illustrate the performance and to evaluate the proposed tracking algorithm, in this methodology only single object tracking is used, an example of video has sequence (AVI 25 frame/second 720x576).

Figures 6 and 7 give the selected motion object which accumulated histogram was computed for each part of object (four parts proposed in this study) from this step, the distinguish feature was obtained for each region in order to use it in proposed tracking algorithm and to describe correctly the target when partial occlusion was occurred. This important feature could help to improve the effectiveness of the proposed work.

The proposed study combines the multipart accumulated histogram feature with EHCSO in order to increase the efficiency of object tracking in view of computation complexity. The proposed study was computed by C sharp (Search window size: 30). Also the compare computation times between the proposed algorithm with PSO for the same video sequence have been implemented. The results are shown in Table. 1. From this table it is quite clear that our proposed algorithm gives superior results (more enhancement) than the previously algorithm. This is because more exploitation of the object motion characteristics has been considered in this proposed work. We have included four features to calculate a more accurate probability value which is reflected in our results.

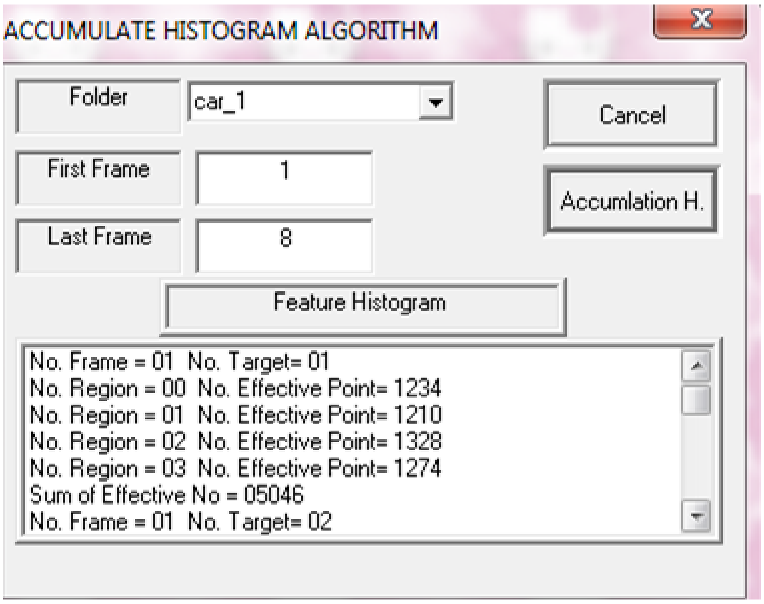


Figure 6. Accumulated Histogram for Each Region for the Frame 1

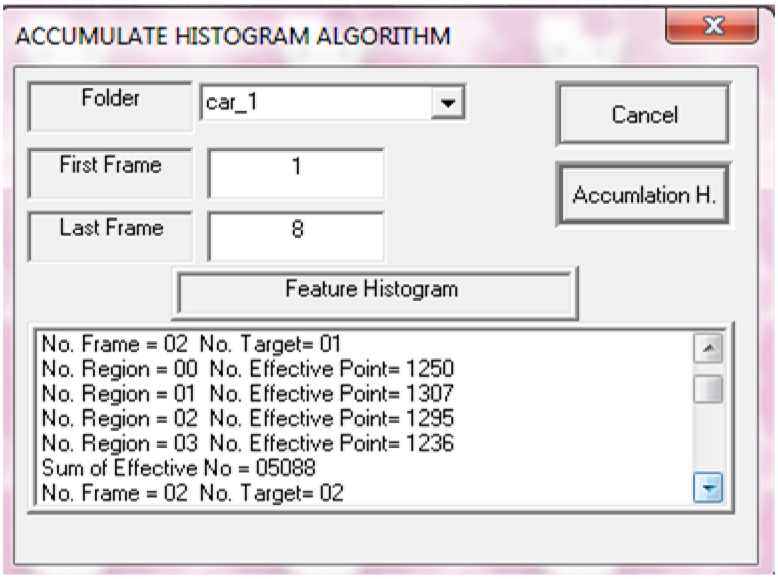


Figure 7. Accumulated Histogram for Each Region for the Frame 2

Table 1. Computation Times per Pixels and the MSE per Pixel

Algorithms	PSO	Proposed algorithm
Computations	13.94	12.00
MSE	11.92	10.32

Figure 8 below show the tracking technique for multiple objects using IHCSO algorithm, in this step a huge data sets will be stored.

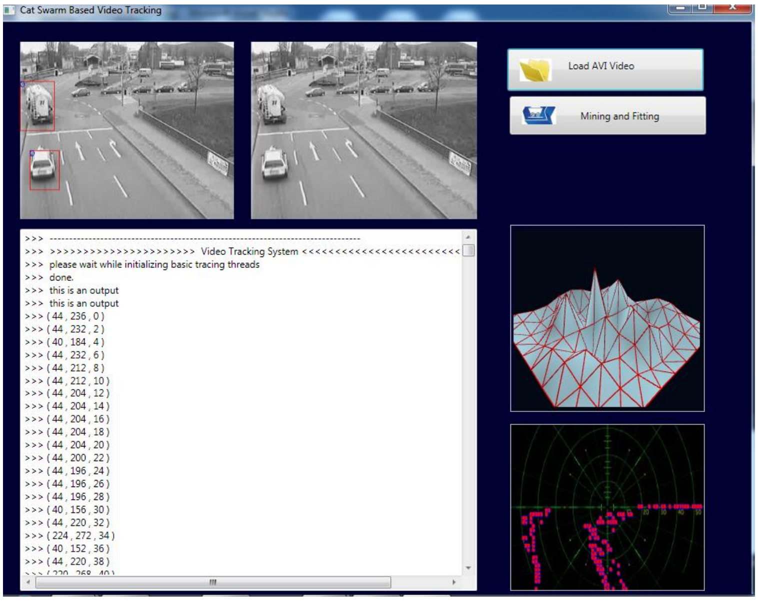


Figure 8. Tracking Technique for Multiple Objects

Finally Figure. 9 below show the 3D equation in term of(x),(y) ant(t) parameters for all data sets that describe the flow of most common path of multiple objects.

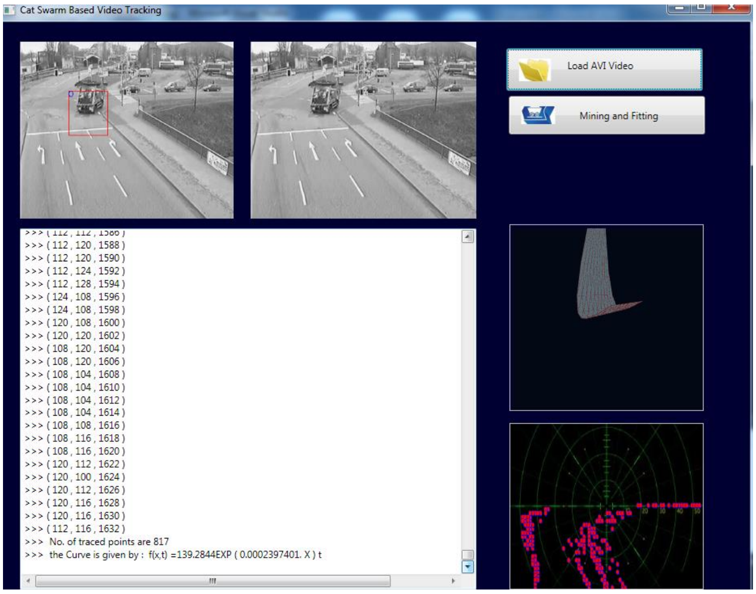


Figure 8. 3D curve fitting

12. Conclusion

A solution to upgrade the performance of object tracking over an EHCSO has been presented. An improved object tracking method based on accumulation histogram and multi-part appearance models is presented; these make some significant improvements in EHCSO algorithm. It uses the accumulation histogram to describe the object moving characteristics instead of the general histogram; the accumulation histogram distinguishes colors more accurately, while the multi-part nature of the models enforces a spatial structure on the tracked object and introduces spatial information in the representation thus improving the tracker robustness to false targets especially with partial occlusion.

In this study a multi part representation based on the computation of four accumulated histogram will be used which overcomes the drawbacks of global color histogram based tracking, this representation method decrease the value of CDC value in seeking mode in EHCSO into 25% because each group of cats responsible for 3directions from overall 12 proposed directions. In this paper the block matching will not be used, instead of this, some important features from this method will be used such as search window and matching criteria these leads to increase the efficiency in search algorithm (EHCSO). Also the proposed algorithm is considered with a flexible size of block (the size of each region in the target) as well as pixel displacement.

Experimental results show that the proposed representation achieves an improvement of tracking accuracy and a reduction of track losses, without increasing significantly the computational complexity.

Further development and more evaluation should be corporate to implement the algorithm for real time application. Also includes investigating an appropriate adaptive representation based on target content, in order to weight differently the parts of the representation.

Several studies have focused on the semantic properties of trajectories. All of them deal with trajectory as a set of points this assumption leads to huge data and increase the storage file and increase overall overhead and complexity. The novelty of this paper is the creation “trajectory function” for each object in sequence of frames that make able to build knowledge base to infer new and more knowledge about important places of trajectories.

In this paper we used to Fit data to built-in and user-defined fitting functions, Do non-linear regression, Virtually unlimited number of fit coefficients in user-defined fitting functions, Automatic calculation of the model curve, curve fit residuals, and confidence and prediction bands. These curves can be automatically added to a graph of your data, Optional automatic calculation of confidence limits for fit coefficients, set and hold the value of any fit coefficient and weighted data fitting. We are able to infer the behavior of the moving object and understand the goals of his/her trajectory.

The experimental results validate our solutions by showing the usefulness of using numerical analysis with video tracking and improvement in performance compared to trajectory points approach. The results demonstrated that the proposed method is more robust to noise and to missing object observations. Our future work is focused on event modeling using trajectory clustering depend on some important features that extract from trajectory function.

Finally the 3D curve fitting equation can be able the analyzer to make a clear understanding of the flow of multiple objects in order to put the solutions if there is a problem appeared in the road to make a correct decision.

References

[1] Kalyan G., Gwang-Soo H. and Byung-Gyu K.,” A Novel Mesh-Based Moving Object Detection Technique in Video Sequence”, Future Technology Research Association International, Journal of Convergence, vol. 4, no. 3, September **(2013)**.

[2] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua, “Multiple Object Tracking Using K-Shortest Paths Optimization,” ,IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 9, **(2011)**.

[3] X. Zhang, W. Hu, W. Qu, and S. Maybank, “Multiple Object Tracking Via Species-Based Particle Swarm Optimization,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 20, no. 11, **(2010)**.

[4] Israa H. and Mustafa S., “An Enhanced Video Tracking Technique Based on Nature Inspired Algorithm”, International Journal of Digital Content Technology and its Applications (JDCTA), vol. 8 no. 3, June **(2014)**, pp. 32-42.

[5] Press W. H., Flannery B. P., Teukolsky S. A., and Vetterling W. T., “*Numerical Recipes in C”,* Cambridge University Press, New York **(1988)**.

[6] Israa H. and Mustafa S., “Behavior Formula Extraction for Object Trajectory using Curve Fitting Method”, International Journal of Computer Applications (0975 – 8887), vol. 104 no.2, October **(2014)**.

[7] Chambers J. M., Cleveland W. S., Kleiner B., and Tukey P. A. "*Graphical Methods for Data Analysis"*, Duxbury Press, Boston **(1983)**.
[8] Israa H. and Mustafa S., "Improvement Cat Swarm Optimization for Efficient Motion Estimation", International Journal of Hybrid Information Technology, vol.8 no.1, **(2015)**, pp. 279-294.
[9] Mrs. N. G. Chitaliya and Prof. A. I. Trivedi, "Novel Block matching algorithm using Predictive motion vector for Video Object Tracking based on color histogram", Electronics Computer Technology (ICECT), 3rd International Conference on**(2011)**, pp. 81-85.
[10] Israa H. and Mustafa S., "Enhanced Hybrid Cat Swarm Optimization Based on Fitness Approximation Method for Efficient Motion Estimation", International Journal of Hybrid Information Technology, vol. 7, no. 6, **(2014)**, pp. 345-364.
[11] Kalyan G., Gwang-Soo H. and Byung-Gyu K., "A Novel Mesh-Based Moving Object Detection Technique in Video Sequence",Future Technology Research Association International, Volume 4, Number 3,pp 20-24, September **2013**.
[12] Emilio M. and Andrea C., "Multi-part target representation for color tracking", Image Processing, IEEE International Conference on , vol. 1, **(2005)**, pp I - 729-32.
[13] Changhyun C., Seung-Min B. and Sukhan L., "Real-time 3D Object Pose Estimation and Tracking for Natural Landmark Based Visual Servo", IEEE/RSJ international conference, **(2008)**, pp. 3983-3989.
[14] Marcus B., Uwe D. H., "Shape Tracking of Extended Objects and Group Targets with Star-Convex RHMs", 14th IEEE international conference, (**2011)**, pp. 1-8.
[15] Dong W., Huchuan L., Ming-Hsuan Y., "Least Soft threshold Squares Tracking", IEEE international conference, **(2013)**, pp. 2371-2378.
[16] Steve W., "Numerical method for engineering", **(2006)**.
[17] Chafik S., P. A. Absil, Anuj S., Eric K., "A Gradient-Descent Method for Curve Fitting on Riemannian Manifolds", paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programmer, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its authors. This research was supported in part by AFOSR FA9550-06-1-0324 and ONR N00014-09-10664.March 9, **(2011)**.
[18] S. C. Chu, P. W. Tsai, and J. S. Pan, "Cat swarm optimization," in PRICAI 2006: Trends in Artificial Intelligence. Springer, **(2006)**, pp. 854–858.
[19] S. C. Chu and P. W. Tsai, "Computational intelligence based on the behavior of cats," International Journal of Innovative Computing, Information and Control, vol. 3, no. 1, **(2007)**, pp. 163–173.
[20] M. A. Khanesar, M. Teshnehlab, and M. A. Shoorehdeli, "A novel binary particle swarm optimization", in Control & Automation, 2007. MED'07, Mediterranean Conference on, IEEE, **(2007)**, pp. 1–6.
[21] P. W. Tsai, J. S. Pan, S. M. Chen and B. Y. Liao, "Enhanced parallel cat swarm optimization based on the Taguchi method", Expert Systems with Applications, vol. 39 no. 7, **(2012)**, pp. 6309-6319.
[22] J. P. T. Yusiong, "Optimizing Artificial Neural Network Using Cat Swarm Optimization Algorithm", International Journal of Intelligent Systems and Applications, vol. 5, no. 1, **(2012)**, pp. 69-80.
[23] P. W. Tsai, J. S. Pan, S. M. Chen, B. Y. Liao and S. P. Hao, "Parallel Cat Swarm Optimization", In Proceedings of the 7th International Conference on Machine Learning and Cybernetics, **(2008)**, pp. 3328-3333.
[24] S. C. Chu and P. W. Tsai, "Computational intelligence based on the behavior of cats", International Journal of Innovative Computing, Information and Control, vol. 3, no. 1, **(2007)**, pp. 163–173.
[25] D. Abramson, and J. Abela, "A Parallel Genetic Algorithm for Solving the School Timetabling Problem", Technical Report, Division of Information Technology, CSIRO, **(1991)**.
[26] M. Dorigo, and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem", IEEE Trans. on Evolutionary Computation, vol.26 no.1, **(1997)**, pp. 53-66.
[27] Jui-Fang C., Shu-Chuan C., John F. R. and Jeng-Shyang P., "A Parallel Particle Swarm Optimization Algorithm with Communication Strategies", Journal of Information Science and Engineering, vol.21, no.4, **(2005)**, pp.809-818.
[28] Maysam O., Mohammad M. and Mohammad T., "Average-Inertia Weighted Cat Swarm Optimization", ICSI 2011, Part I, LNCS 6728, **(2011)**, pp. 321–328.
[29] Z. Zhou, Y. Ong, M. Nguyen and D. Lim, "A study on polynomial regression and Gaussian process global surrogate model in hierarchical surrogate-assisted evolutionary algorithm", in: IEEE Congress on Evolutionary Computation (ECiDUE'05), Edinburgh, United Kingdom, September 2–5, **(2005)**.
[30] A. Ratle, "Kriging as a surrogate fitness landscape in evolutionary optimization", Artificial Intelligence for Engineering Design, Analysis and Manufacturing, vol. 15 **(2001)**, pp. 37–49.
[31] D. Lim, Y. Jin, Y. Ong and B. Sendhoff, "Generalizing surrogate-assisted evolutionary computation", IEEE Transactions on Evolutionary Computation, vol. 14 no. 3, **(2010)**, pp. 329–355.
[32] Y. Ong, K. Lum and P. Nair, "Evolutionary algorithm with hermite radial basis function interpolants for computationally expensive adjoint solvers", Computational Optimization and Applications, vol. 39 no. 1, **(2008)**, pp. 97–119.
[33] Israa H. and Mustafa S., "Enhanced Hybrid Cat Swarm Optimization Based on Fitness Approximation Method for Efficient Motion Estimation", International Journal of Hybrid Information Technology, vol. 7 no. 6, **(2014)**, pp. 345-364.

[34] Erik C., Daniel Z., Marco P. C., Humberto S. and Valentín O., “Block matching algorithm for motion estimation based on Artificial Bee”, Applied Soft Computing, Elsevier, vol. 13, **(2013)**, pp. 3047–3059.

Authors

Israa Hadi, she received her Ph.D. degree from University of Babylon. She is currently a Professor in Software Department, College of Information Technology, Babylon University, Iraq. She is primary research interests cover video tracking, data mining and information hiding.

Mustafa Sabah, he is currently a Ph.D. student of College of Information Technology University of Babylon. His primary research interests cover video tracking and swarm optimization.