Two view NURBS reconstruction based on GACO model

Deepika Saini¹ · Sanoj Kumar² · Manoj K. Singh² · Musrrat Ali³

Received: 20 September 2020 / Accepted: 20 May 2021 / Published online: 5 June 2021
© The Author(s) 2021

Abstract

The key job here in the presented work is to investigate the performance of Generalized Ant Colony Optimizer (GACO) model in order to evolve the shape of three dimensional free-form Non Uniform Rational B-Spline (NURBS) curve using stereo (two) views. GACO model is a blend of two well known meta-heuristic optimization algorithms known as Simple Ant Colony and Global Ant Colony Optimization algorithms. Basically, the work talks about the solution of NURBS-fitting based reconstruction process. Therefore, GACO model is used to optimize the NURBS parameters (control points and weights) by minimizing the weighted least-square errors between the data points and the fitted NURBS curve. The algorithm is applied by first assuming some pre-fixed values of NURBS parameters. The experiments clearly show that the optimization procedure is a better option in a case where good initial locations of parameters are selected. A detailed experimental analysis is given in support of our algorithm. The implemented error analysis shows that the proposed methodology perform better as compared to the conventional methods.

Keywords NURBS · Stereo images · Inverse reconstruction · Triangulation · Ant Colony Optimization

Introduction

Two view reconstruction of a three dimensional object is a very well studied but still a challenging problem in computer vision [1,2]. The main task in the reconstruction process may be thought as to obtain the geometrical structure of any scene from the provided pair of two dimensional images. Generally, the two view reconstruction process is treated as ill posed problem due to geometric inverse in nature. Hence very small approximation error may cause a very large deflection in the final result. This is the main reason of very less interest shown

by the researchers towards the multi view reconstruction. The basic approach for the reconstruction of the locations of corresponding three dimensional points is known as point-based reconstruction algorithm [3,4]. The core principle of point-based reconstruction method is as follows: the points are located in two view planes, and the three dimensional position of this point may be found as the intersection of the two emitted projection rays. The three dimensional shape of any object may be recovered by repeating the principle for several points. This two view configuration requires a complete prior knowledge of camera positions, camera parameters and their relative orientation.

For algebraic curves, each pair of corresponding points in both views yields the same three dimensional point. But it is not possible in case of free-form shapes or objects. Generally, it is happened due to the presence of some fallacies like difficulties in locating the correct correspondence between the points, complex shapes, geometrical errors, very large amount of corrupted data, etc. To reduce the complexity of free-form reconstruction, it may be formulated as: Find the locations of three dimensional points whose back projections fit the measured data points in all the view planes very well. This formulation involves a very sensitive part, named curve fitting through the measured data points. This fitting is a very crucial problem itself in a wide region of Science and Engi-

✉ Sanoj Kumar
sanoj.kumar@ddn.upes.ac.in

Deepika Saini
deepikasaini@geu.ac.in

Manoj K. Singh
mkumar@ddn.upes.ac.in

Musrrat Ali
mkasim@kfu.edu.sa

¹ Department of Mathematics, Graphic Era Deemed to be University, Dehradun, Uttarakhand 248002, India

² Department of Mathematics, University of Petroleum and Energy Studies, Dehradun, Uttarakhand 248007, India

³ Department of Basic Sciences, PYD, King Faisal University, Al Ahsa 31982, Saudi Arabia

neering. The fitting is generally termed as the error function for the given data. Various approaches such as least-square fitting [5,6], quadratic programming [7] etc. were taken to model this problem. However, they served the purpose but the main drawback of these algorithms were time consumption and the second drawback is the complex nature of the reconstruction problem.

While talking about the free-form shape reconstruction, some approximation techniques come into existence, for example Cubic Splines, Bernstein polynomials, B-Splines, Non Uniform Rational B-Spline or NURBS [8,9]. Here NURBS can represent the algebraic curve as well as the free-form shapes very well. It is capable of representing the complex geometrical shapes also. The main advantage of using this model is that it can be completely generated by its control points, hence the reconstruction problem in three dimensional space is simply converted to the reconstruction of control points associated with NURBS representation of the corresponding shape. Xiao et al. [6] gave a method to reconstruct free-form NURBS curve in three dimensional space using stereo views. For the fitting process of digitized data, they have adopted least-square approximation in the orthographic projections of a NURBS model. The applied approach was efficiently able to reconstruct the three dimensional structures directly from their stereo pictures but they need to consider more constraints in their optimization framework to resolve the stereo ambiguity. Peng et al. [10] developed an algorithm based on the concept of projective depth estimation and simplified iterative closest point for the reconstruction of the three dimensional structure using a sequence of images. They said that the depth can be detained as a linear factor for any pair of three dimensional points. Through the experiment validation it was shown that the proposed algorithm mostly suited for the cases with less noise. In case of large noise, the approach required large iteration time. Cai et al. [11] presented an energy based algorithm for the reconstruction, called snake-based energy minimization approach using NURBS. They recovered the three dimensional figure using the inverse optimization rule to lessen the matching process. They had also shown that how the errors in two dimensional views influenced the reconstructed shape in three dimensional space. Besides control points, all other parameters like knots and weight vectors have been fixed in the optimization framework. Being inspired by the energy based algorithms, Saini et al. [12] also presented the two view reconstruction methodology based on NURBS-snake model. For the optimization process they used Levenberg-Marquardt method. Earlier they have also used Quadratic programming [7] to get the optimum values of control points and weights, called two-step procedure. This optimization framework was purely designed for two view reconstruction using NURBS. But the proposed method was not applicable in case of partially occluded data. Lu et al. [13] used orthog-

onal perspective views to model the same task in the form of NURBS-snake energy minimization but the same problem was not dealt in their approach also.

From the past few decades, researchers are using some evolutionary algorithms and physics based algorithms very frequently to serve the purpose of reconstruction process. Ning et al. [14] applied simulated annealing approach to solve the three dimensional reconstruction problem using two infinite-source orthogonal projections. Ogura et al. [15] used simulated annealing technique to visualize the accurate posterioric angular assignment structure of protein projections. Chen et al. [16] presented a three dimensional image reconstruction from a limited number of projections by simulated annealing. The volume image was represented by a set of 3D Gaussian functions and the reconstruction was done by estimating parameters of kernel functions. Voisin et al. [17] developed Genetic Algorithm techniques to reconstruct the three dimensional shapes with the cloud data points. To handle the noisy data, they defined a fitness function with tolerance threshold. Koch et al. [18] proposed an evolutionary-based algorithm for three dimensional panoramic reconstruction using uncalibrated stereovision system. Their algorithm consists two steps: In first step, they extracted the points of interest in pairs of images which are captured by two consecutive cameras of the system. And in second step, they used evolutionary algorithm to compute the transformed matrix between the two images and the respective depth of the points of interest. Singh et al. [19] proposed a non-linear optimization model to reconstruct the three dimensional curve and surface using Gravitational Search Algorithm (GSA). The objective function was modeled as an error function between the given measured points and the points on the generated NURBS curve or surface. Another optimization approach for curve reconstruction is given by Alazzam et al. [20] where Average Uniform Algorithm (AUA) was used. Their algorithm was principally constructed using uniform distribution to generate random solutions, and then averaging the best solutions to obtain the optimal value for the objective function. As the evolutionary approaches and physics-based algorithms are very popular, easy to implement and able to solve the complex optimization problems, but these approaches have high implementation cost and usually require huge number of iterations. The major drawback of the most of the evolutionary approaches discussed above is that they are totally three dimensional reconstruction techniques using the direct three dimensional data. The goal which is three dimensional reconstruction using two views is neglected throughout in their work due to its complex nature.

In the recent scenario, the nature inspired swarm based meta-heuristic optimization techniques are also proven to be effective in many fields of computer and engineering. These techniques seems more relevant as compared to the tradi-

tional methods to solve the optimization problems in large search spaces [21–23]. The inspiration behind the swarm based optimization techniques is the foraging characteristic of the species (such as real ants). At the initial step of food search, ants move in their surroundings randomly. As soon as an ant get any food source, it collects the information about the food quality and its quantity. During the return to its nest, it releases some pheromone trail on the base. If the quantity of the pheromone deposited is high then the probability to follow that path by the other ants will also be high. This type of indirect communication among the ants by the means of pheromone trails helps to get the shortest paths between the food source and the nest. This behavior of real ants is transferred into a form of algorithm which is known as Ant Colony Optimization (ACO). It was introduced by Dorigo [24,25] in early 1990s. Various types of ACO approaches are existed in literature such as Max-Min, Ant Q, Ant-tabu, Fast ant system [26,27] etc. These approaches are very well used for data fitting also and hence for reconstruction purpose. In [28], Xiao et al. shown the NURBS fitting process in three dimensional space based on the ACO algorithm. They have used a modified ant colony optimization algorithm to estimate the weight vectors by the minimization of the sum square residual error between the fitted and target three dimensional surface. It must be noted here that their proposed optimization model does not involve the control points as the optimization variable. In reverse engineering problems, control points play an important role. It is a member of a set of points used to determine the shape of the curve and surface or more precisely any higher dimensional NURBS structure. Another thing is the ACO algorithm, used for the optimization process, comprised only the local search space using Tabu search and then refined the solutions of all iterations. The global search is not chosen as prior in the search space. Chrysostomou et al. [29] presented a decent approach to model the three dimensional complex structures using the multiple and calibrated photos of the same scene. Their theorem presented some characteristics of space curve algorithm. But the complications with space carving techniques generally arise when images are segmented incorrectly, due to the fact that when a voxel is incorrectly removed at initial stage, it could emerge as a hole in the final 3D model which was not present in the real input images. In the first part of their algorithm, they reconstructed the structures using lightness compensating image comparison method using several input images while in the second part they used ACO algorithm for the further refinement of the reconstructed models. Some modified forms of ACO algorithm like Simple ant colony optimization (SACO) [30], Global ant colony optimization (GCO) [31,32] have also been presented to overcome the complexities of the previously existed form [24]. A comparison of the proposed reconstruction algorithm with other relevant approaches is presented in Table 1. In this table, all reconstruction method-

ologies are compared based on their objectives, number of views, method and the other reconstruction highlights.

In this study, we have used a nature inspired meta-heuristic optimization approach to reconstruct the three dimensional shapes using two views. The main focus of the proposed algorithm is towards the compressive NURBS reconstruction of three dimensional shapes based on GACO model [26]. The most notable novelty of the proposed algorithm lies in the framework of ant colony optimization technique in reverse engineering problems to reconstruct the three dimensional models from its two dimensional stereo images. While other researchers avail themselves to reconstruct the three dimensional object using the three dimensional cloud data points directly. This work shows the potential of using nature inspired methods to provide more accurate three dimensional objects from its stereo views to deal the reverse engineering problems. This model is a hybridization of two optimization algorithms named as SACO and GCO approaches. This new hybrid algorithm is totally based on distance matrix, new colony generation, pure foraging behavior and the continuous effort of ants. We have used this model to get the optimal values of NURBS parameters in two views. The optimized parameters are further used in Triangulation procedure [7] to reconstruct the shape in three dimensional space. The whole paper comprises different sections. “Preliminaries” briefs us a short note on NURBS, the parametrization process and the modified Generalized Ant Colony Optimizer (GACO) model. Our proposed algorithm is described in “Curve reconstruction by GACO model”. It includes detailed explanation about the curve fitting problem, selection of parameters and research methodology to determine the three dimensional parameters for reconstruction process via GACO model. “Experimental results and discussions” floats our experimental models with detailed error analysis. This section also includes the comparison of the proposed algorithm with some previously existed approaches. Finally the paper ends with conclusion, the main contribution and the future scope.

Preliminaries

NURBS function

Let us consider a sequence $(t_0, t_1, \dots, t_{s-1}, t_s)$ of real numbers, where $t_j \leq t_{j+1}$, $j = 0, 1, \dots, s-1$. This sequence is known as knot sequence or knot vector whereas the term t_j including within the sequence is called knot. The number of times this knot repeated itself in a sequence is called the multiplicity of the knot and this multiplicity defines an important role in the shape of the NURBS curve. There are basically two groups of knot vectors, uniform knot vector and non uniform knot vector. In uniform knot vector, the knots inside the sequence are equally spaced and each knot appears only one

Table 1 Comparison with existing approaches in terms of objective and implementation details

Approaches	Objective	Views	Method	Optimization technique	Result highlights
Xiao et al. [6]	3D curve reconstruction	Stereo	NURBS reconstruction	LMA [†]	Control points and weights computation
Chrysostomou et al. [29]	3D object reconstruction	Multiple	Space carving algorithm	ACO [♣]	Luminosity-compensated dissimilarity measurement
Xiao et al. [28]	3D NURBS fitting	—	NURBS	ACO [♣]	Weight and knot computation
Alazzam et al. [20]	Curve fitting	—	Quadratic model	AUA [‡]	Curve parameters (confidants) computation
Singh et al. [19]	3D curve/surface reconstruction	—	NURBS	GSA [*]	Control points and weights
Proposed	3D curve/surface reconstruction	Two views	NURBS	GACO [‡]	Control points and weights computation

LMA[†] Levenberg-Marquardt Algorithm, ACO[♣] Ant Colony Optimization, AUA[‡] Average Uniform Algorithm, GSA^{*} Gravitational Search Algorithm, GACO[‡] Generalized Ant Colony Optimization

time in the sequence. Whereas, in non uniform knot vector, the knots are unequally spaced and/or knot may appear more than one times in the sequence. In case of non uniform knot sequence, generally we find the non-periodic case where the end knots repeat itself according to the order of the NURBS curve and inner knots appear only at one time.

The B-spline basis functions of degree $l - 1$, which are described by Cox-De Boor Recurrence relations [9] are given as

$$B_{j,1}(\zeta) = \begin{cases} 1 & \text{if } t_{j-1} \leq \zeta < t_j, \\ 0 & \text{otherwise.} \end{cases}, \quad (1)$$

and for $l > 1$,

$$B_{j,l}(\zeta) = \frac{\zeta - t_j}{t_{j+l} - t_j} B_{j,l-1}(\zeta) + \frac{t_{j+l+1} - \zeta}{t_{j+l+1} - t_{j+1}} B_{j+1,l-1}(\zeta), \quad (2)$$

where Eq. (1) shows that within the interval $[t_j, t_{j+1})$, the B-spline Basis function $B_{j,1}$ is a piecewise continuous function with value as 1, and 0 elsewhere. This is known as the support of the function. If both the knots in a knot vector $[t_{j-1}, t_j]$ are same then this support reduces to an identity. It is clear from Eq. (2) that the B-spline basis function $B_{j,l}(\zeta)$ of order l is a combination of B-spline basis function of previous orders with coefficients as the linear factors in ζ . While its support is the union of the supports of the former basis functions of order $l - 1$. Note that the condition $\frac{0}{0} = 0$ is applicable in Eq. (2) wherever it is necessary.

Now, define the rational B-spline basis function as

$$r_{j,l}(\zeta) = \frac{w_j B_{j,l}(\zeta)}{\sum_{j=0}^n w_j B_{j,l}(\zeta)}, \quad j = 0, \dots, n, \quad (3)$$

where $B_{j,l}$ is the B-spline basis function of order l and w_j is the weight vector. Hence for a given set of two dimensional control points p_j and a knot vector $(t_0, t_1, \dots, t_{s-1}, t_s)$, the non uniform rational B-spline function or NURBS curve of order l is defined as

$$c(\zeta) = \sum_{j=0}^n p_j r_{j,l}(\zeta), \quad (4)$$

where the B-spline basis functions are given by Eqs. (1) and (2). NURBS function have so many important properties, for example non-negativity, partition of unity, local support, strong convex hull property, affine invariant, etc. and it can be converted into a simple B-spline basis function or Bezier curve [8,9].

Parametrization process

Another important property which is related to the NURBS curve is named as parametrization process. It helps us to assign a numerical value to each point on a NURBS curve. Hence each point of the NURBS curve has a set of some numerical value or parameter value that describe the location of the point on the NURBS curve. The chord length parametrization process is given by

$$\zeta_0 = 0, \quad \zeta_j = \frac{|t_j - t_{j-1}|}{\sum_{j=2}^n (t_j - t_{j-1})} \quad \text{for } j = 1, \dots, n \quad \text{and} \quad \zeta_n = 1. \quad (5)$$

The chord length parametrization lies in the range $[0, 1]$ i.e. the parameter ζ takes the values within the range $[0, 1]$. Besides the chord length parametrization process, there are many more methods for the same, such as uniform parametrization [9,33], centripetal parametrization [9], hybrid parametrization [34] etc.

GACO model

The Generalized Ant Colony Optimizer (GACO) model mainly consists two approaches: Simple ant colony optimizer and Global ant colony optimizer. In this model ant's foraging behavior plays an important role. With this type of behavior ants are able to approach to the optimal path without any coordination, advice and corporation. Dorigo et al. [30] have presented the foraging behavior of ants by means of a formal binary path selection model. According to that model: Let us consider the paths E_1 and E_2 with moving ants N_{E_1} and N_{E_2} respectively at any time t . Now, at time $(t + 1)$, the probability of choosing the path E_1 by the new upcoming ants as the target path is given by

$$p_{E_1(t+1)} = \frac{(N_{E_1}(t) + k)^\alpha}{(N_{E_1}(t) + k)^\alpha + (N_{E_2}(t) + k)^\alpha}, \quad (6)$$

$$= 1 - p_{E_2(t+1)},$$

where the term α denotes the pheromone deposit by the ants. As we increase the value of α , the probability of choosing that path will certainly increase. The constant c denotes the degree of the attraction of the path. According to the foraging model, when the ants move towards the food search, each ant leave a certain amount of pheromone value. Shorter or nearest path will get more amount of pheromone deposited as compare to the longer paths and all the following ants quickly adopt that shorter path. The main algorithm can be divided into three stages or parts: (a) Ant based solution construction; (b) Pheromone update; and (c) Daemon actions. Let us consider a set of artificial ants $n = (a_1, a_2, \dots, a_n)$ located in a colony.

Ant based solution construction stage: The solution construction starts with a random partial solution. After that, at each construction step, the solution is extended by adding some solution component from the set of feasible solution V_n^i to the partial solution. The choice of solution component is done probabilistically at each construction step. Let us consider that the ant n is currently at the point i and its probability to transit in the next point $j \in V_n^i$ is given by

$$p_n^{i,j}(t) = \begin{cases} \frac{z_{i,j}^\alpha(t)}{\sum_{j \in V_n^i} z_{i,j}^\alpha(t)}, & j \in V_n^i; \\ 0, & j \in V_n^i. \end{cases} \quad (7)$$

Each and every ant has its own decision policy to select the next shifting point. The value of $z_{i,j}$ denotes the pheromone value at the path (i, j) whereas α denotes the impact of pheromone deposit, and it must be a positive constant value. All the ants built the entire path by linking the source points to their destination points. Now remove the entire path cycles of individual ants but they can be retraced from their sources as all the paths (i, j) have their pheromone amount deposited. In this way the pheromone table may be constructed. This table contains a real valued information for each of the destination point and for each of the adjacent path. The real valued information is thought of as a practicing marks of traveling over the adjacency paths on the way to the destination points.

Pheromone update stage: In the beginning of the process, all the ants will find the nearest point randomly. And the ants with successfully achieved the destination points have the right to update the total path length $\mathcal{L}$ in a pheromone table. Let us consider $\mathcal{L}^n(t)$ as the path length traveled by the ant n . The total pheromone added or deposited by the ant n on this path length is $\Delta z_{i,j}^n(t)$ which is given by

$$\Delta z_{i,j}^n(t) \propto \frac{1}{\mathcal{L}^n(t)}.$$

The pheromone table is a continuously updating table by the ants in the search of the possible solutions. The pheromone update is to be performed by the rule

$$z_{i,j}(t+1) = z_{i,j}(t) + \sum_{n=1}^{a_n} \Delta z_{i,j}^n(t+1). \quad (8)$$

The upcoming ants will take the help from this updated pheromone table in search of their destinations.

Daemon action stage: Let us denote the optimal solution by $\mathcal{O}^n(t)$ and quality solution by $f(\mathcal{O}^n(t))$. In this algorithm, local search space as well as global search space are taken into consideration. The local pheromone amount $\Delta z_{i,j}^n(t+1)$

is calculated by the rule

$$\Delta z_{i,j}^n(t+1) = \begin{cases} \frac{1}{\mathcal{L}^n(t+1)}, & \text{if } n\text{th ant follows the path } (i, j); \\ 0, & \text{otherwise.} \end{cases}$$

In the above Eq. (9), $\mathcal{L}^n(t+1)$ is the total path length traveled by the ant n . The role of local pheromone update is to make the transition paths attractive by dynamically changing the path tours. The transition of the path by the ants also occurred after every iteration based on the deposited pheromone amount. Thus, the local search is always performed in order to get the shorter paths. The global pheromone update is performed using the similar rule as for the local search. Once all the ants complete their paths, global search is performed to get the best path as follows:

$$\Delta z_{i,j}^n(t+1) = \begin{cases} \frac{1}{\mathcal{L}_{\text{best}}(t+1)}, & \text{if } (i, j) \text{ is the best path;} \\ 0, & \text{otherwise,} \end{cases} \quad (10)$$

where $\mathcal{L}_{\text{best}}$ is the shortest path. Let a_l ants and a_g ants perform the local and global search respectively. Now denote $r(i)$ as the target points and the function $f(r(i))$ as the fitness function. All the target points are assigned a quality solution, and a certain amount of pheromone value z_i is also initialized for each target point. It is observed in almost all the ant colony optimization problems that they fall into the local optimum so easily. Hence crossover and mutation [35] play an important role to avoid the prematurity of the algorithm. They certainly increase the diversity of the ants. In this study, 95% of the a_g perform crossover and mutation while the remaining 5% are involved in pheromone trial distribution. The probability of each n of the local ants a_l selects $r(i)$ target points, which is partially towards the good target points, is as follows:

$$p_i^n(t) = \frac{z_i^\alpha(t) v_i^\beta(t)}{\sum_{j \in N_i^n} z_j^\alpha(t) v_j^\beta(t)}, \quad (11)$$

where $v_i^\beta(t)$ represents the attractiveness of the move from source to destination and N_i^n denotes the feasible solution. Now, we calculate the fitness function for the target points. If the target points get the better fitness, then the upcoming ants follow the same path otherwise choose new direction randomly with the increment in the age of target points i.e. the weakness of the particular solution.

Curve reconstruction by GACO model

Curve fitting

- Let us assume that we want to fit a provided set of two dimensional data points $q = \{q_\alpha\} = (x_\alpha, y_\alpha)_{\alpha=0, \dots, p}$ with the help of a NURBS curve $c(\zeta)$ of order l such that ($s < p$)

$$q_\alpha = \sum_{j=0}^n p_j r_{j,l}(\zeta_\alpha) \quad \text{for all } \alpha = 0, \dots, p. \quad (12)$$

For this fitting we must use an arrangement of parameter values ζ_α to each of the measured data points $\{q_\alpha\}$ and a suitable choice of weight vectors w_j . Thus the problem is now reduced in finding the curve which minimizes the following expression (weighted least-squares):

$$f = \sum_{\alpha=0}^p [q_\alpha - c(\zeta_\alpha)]^2. \quad (13)$$

We may think of the above problem as: There are the data points arranged in a vector form $\{q_\xi = (x_\xi, y_\xi)\}$ with $\xi = 0, \dots, D$ where D indicates the number of given data points. After providing the parameter values ζ_ξ and the weights w_ξ to each of the data points q_ξ , the fitting problem can be expressed as

$$f = \sum_{\xi=0}^D [q_\xi - c(\zeta_\xi)]^2 = \sum_{\xi=0}^D \left[q_\xi - \frac{\sum_{j=0}^n w_\xi p_\xi b_{j,l}(\zeta_\xi)}{\sum_{j=0}^n p_\xi b_{j,l}(\zeta_\xi)} \right]^2. \quad (14)$$

From the above Eq. (14), we get an over-constrained system of linear equations (with p_ξ as the unknown in above system). A least-square solution [5,6] is widely chosen for the solution of these kind of system. But the B-spline basis functions involved in the system are highly non-linear in nature and also there is a huge number of unknowns for a given large data points. Thus in accordance to the mentioned challenges it will be much more suited to apply a non-linear optimization process to get the desired solution.

The matrix representation of Eq. (12) is

$$\begin{pmatrix} q_0 \\ q_1 \\ \dots \\ q_p \end{pmatrix} = \begin{pmatrix} r_{0,l}(\alpha_0) & r_{1,l}(\alpha_0) & \dots & r_{n,l}(\alpha_0) \\ r_{0,l}(\alpha_1) & r_{1,l}(\alpha_1) & \dots & r_{n,l}(\alpha_1) \\ \dots & \dots & \dots & \dots \\ r_{0,l}(\alpha_p) & r_{1,l}(\alpha_p) & \dots & r_{n,l}(\alpha_p) \end{pmatrix}$$

$$\begin{pmatrix} p_0 \\ p_1 \\ \vdots \\ p_n \end{pmatrix} \text{ OR } [\mathbf{Q}] = [\mathbf{R}][\mathbf{P}]. \quad (15)$$

Pre-Multiply by $[\mathbf{R}]^T$ both sides (due to the over-determined nature of the above system), we get

$$[\mathbf{R}]^T [\mathbf{Q}] = [\mathbf{R}]^T [\mathbf{R}][\mathbf{P}]. \quad (16)$$

Now we are able to get the classical least-square minimization system which provides the best fit coefficients in the weighted least-square manner to the given data points. Hence we can say that in the fitting problem, generally we have to perform the following tasks:

- A suitable and careful selection of parameters (order, no. of control points, knot vectors) is required to generate the B-spline basis function first.
- The positive weights corresponding to each control point must be obtained.
- The proper parametrization of the given data has to be obtained.
- At last, the control points must be obtained.

In the next sections, we describe the procedures to solve the above subproblems by the GACO model. This algorithm presents the solution of fitting problem in a general and unified way.

Selection of parameters

For the solution of above subproblems, the inputs are:

- The collection of two dimensional data points, $\mathbf{Q}$.
- The order of the NURBS curve l
- The number of control points $n + 1$.

For the dimension of search space, we have the following details

- The total number of control points to be calculated are $(n + 1)$, each of which has two coordinates plus the corresponding weight vector. That forces the search space to $3(n + 1)$.
- With the total data points $(p + 1)$, each with one parameter value ζ , the search space is $(p + 1)$.

Hence, the search space dimension in the case of curve fitting is directed to $3(n + 1) + (p + 1)$. Now, the order of the B-spline basis functions and the numbers of control points are chosen based on the nature of the given data points. As we know that

the NURBS are piecewise polynomial functions so we can randomly choose the numbers and places of the control points for simple as well as complex shapes. The inner knots are set within the range $[0, 1]$ while the weight vectors are also set normalized within the same range $(0, 1)$. The NURBS curve parameters (weights, control points), which were initially set to some values, are evaluated numerically by solving the Eq. (16) with the method of two step linear process as described in [7]. Hence the choice of control points is not critical at all. Similar relevant is the value of the initial pheromone which is set as $z_{i,j} = 0.1$ and $\alpha = 1, \beta = 1, z_k(0) = 1$.

Methodology

In this section, the adopted methodology is described. Many conventional techniques have been used by the researchers [8,9] for curve fitting where a constant value (typically 1) is always chosen for the weight vectors of the respective control points. This step generates a automatic B-spline curve from the NURBS curve. Hence some of the important properties of the NURBS curve [9] will be lost automatically. This comes out to be the one of the major weaknesses of the conventional approaches. In the presented approach, at the starting of proposed algorithm we have given some random values [7] to the control points and their corresponding weight vectors. After that GACO model will optimize the results accordingly. The procedure of the proposed algorithm is shown in Fig. 1. The GACO algorithm is described as follows:

ALGORITHM

- Step 1 Start
- Step 2 Generate data points $\mathbf{Q}$
- Step 3 NURBS curve of order l with n control points will be constructed to fit the data points $\mathbf{Q}$. The first step to fit the data points is to calculate the knot vectors which is set as $(0, \dots, 0, t_1, \dots, t_{n-p}, 1, \dots, 1)$ i.e., in the range of $(0,1)$
- Step 4 From the generated data points, the values of the parameters are calculated using chord length parametrization method given by Eq. (5)
- Step 5 Generate random control points and weight of the NURBS curve
- Step 6 Evaluate the fitness of the objective function using Eq. (14)
- Step 7 Start the GACO algorithm
- Step 8 Create ants
- Step 9 Put ants on an entry state
- Step 10 link empty path lists for each ant
- Step 11 Select next state for each ant using Eq. (7)
- Step 12 Are the path linkage completed if no then go to Step 11
- Step 13 For each ant add pheromone values in the pheromone table using Eq. (8)

Fig. 1 Methodology Of NURBS curve fitting using GACO model

- Step 14 Calculate the objective function (Eq. (14)) and sort the fitness for all target points in decreasing order
- Step 15 Post 95% of a_g for crossover and mutation
- Step 16 Post 5% for trail diffusion
- Step 17 Update pheromone values by using the Eq. (10) and also update the weakness of a_g
- Step 18 Send a_l ants to pick good target points by Eq. (11)
- Step 19 For each a_l calculate the fitness function, if fitness improve then go to Step 20 otherwise go to Step 21
- Step 20 Update the pheromone value by Eq. (9), and move to good target points
- Step 21 Increase the age of target points and select new direction randomly
- Step 22 Repeat the steps until terminating criteria (to get a small predefined small value) is reached
- Step 23 End

Experimental results and discussions

Reconstruction results

In this section, the reconstruction results based on GACO algorithm are discussed. The proposed approach is tested for synthetic (Helix, Testsurface) as well as real world objects (Vase, Tsukuba, Giraffe). The code for the reconstruction algorithm is developed in MATLAB and a computer with 2.53 GHz Intel(R) I3 processor with 3.0 GB RAM is used for computation. For two view reconstruction process mainly the following steps are taken into consideration:

- Generate the digitized data points in both views by projecting the synthetic three dimensional data points with the help of projection matrices.
- Start NURBS fitting [7] and obtain the initial parameter values of NURBS.
- Perform GACO algorithm in both image planes to get NURBS control points and corresponding weights.
- Use third view [7] for establishing the correspondence between the control points in two images.
- Reconstruct the control points of the NURBS shape (curves or surfaces) in three dimensional space using Triangulation process [7]. Moreover, assign the appropriate weights to each reconstructed three dimensional control point to generate the complete NURBS shape.

Model 1: Helix The very first mean for the validation of our approach is a helix of the form (shown in Fig. 2(i)):
$$\begin{cases} X = 50 + 3 \sin t \\ Y = 30 + 5 \cos t \\ Z = 10 + 0.5 t \end{cases} \quad t \in [0, 2\pi]$$
 We have projected the above parametric curve onto the left image plane using

the projection matrix $T^L = \begin{pmatrix} 1 & 0 & 0 & 50 \\ 0 & 1 & 0 & 20 \\ 0 & 0 & 1 & 0 \end{pmatrix}$ and onto the

right image plane with matrix $T^R = \begin{pmatrix} 1 & 0 & 0 & -20 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$.

Now the data points in the two dimensional image planes (as shown by $\cdot$ in Fig. 2a and b are obtained with help of above projection matrices. The inputs for NURBS fitting [7] are set as: (i) Order of the fitted two dimensional NURBS curve: 4; (ii) Non-uniform Knot vector: $[0 \ 0 \ 0 \ 0 \ \frac{1}{10} \ \frac{1}{10} \ \frac{2}{10} \ \frac{3}{10} \ \frac{5}{10} \ \frac{6}{10} \ \frac{8}{10} \ \frac{9}{10} \ \frac{9}{10} \ 1 \ 1 \ 1 \ 1]$; (iii) The number of control points: 13. The search space in this case is a hyperspace of dimension 80. The algorithm [7] generates the control points and corresponding weight vectors. And through this NURBS fitting a two dimensional curve is fitted, and points are calculated for different parametric values. Now the error function or objective function is formed as the error between data points and the calculated points. The objective is to minimize this error function by GACO model. The values of the algorithm parameters are already set as $z_{i,j} = 0.1$, $\alpha = 1$, $\beta = 1$, $z_k(0) = 1$. The termination criterion is set as 100 iterations or minimum threshold error. Figure 2c and d show the starting random positions of the control points by $\diamond$. Using the presented theory, the updated positions of the control points are shown by $\diamond$ in the Fig. 2e and f. Using third view [12], the three dimensional control points (shown by red colored $\diamond$ in Fig. 2g) and weight vectors are obtained. Finally the reconstructed NURBS curve with the obtained parameters is shown in the Fig. 2h. This represents the best fit error of the NURBS curve of order 4 with 13 control points.

Model 2: Testsurface As we know that the NURBS surface differs from a NURBS curve in a way that NURBS surface has two parametric directions instead of one. And also the order and the knot vector must be defined for both parameters. With the help of similar concept we may extend the presented theory for surfaces also. For the validation, we have considered the following testsurface (Fig. 3g):

$$\begin{cases} X = \text{linspace}(-2, 2, 10) \\ Y = \text{linspace}(-2, 2, 10) \\ (X \ Y) = \text{meshgrid}(X, Y) \\ Z = X(e^{-X^2-Y^2}) \\ \text{surf}(X, Y, Z) \end{cases}$$

For the generation of the data points (shown with blue colored $\square$ Fig. 3a, b) in both image planes, we have taken the left and right projection matrices as $T^L = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$ and

Fig. 2 Reconstruction results of Helix: **a, b** Two dimensional projected data points in left and right views; **c, d** Randomly posed control points in both image planes; **e, f** Updated positions of control points in both views; **g** Reconstructed control points in space; **h** Reconstructed NURBS curve in space; **i** Helix

$$T^R = \begin{pmatrix} 1 & 0 & 0 & -10 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}. \text{ The parameters for NURBS fit-}$$

ting are set as: (i) NURBS surface order: (3, 3); (ii) The knot vectors: $[0 \ 0 \ 0 \ \frac{1}{4} \ \frac{1}{4} \ \frac{2}{4} \ \frac{2}{4} \ \frac{3}{4} \ \frac{3}{4} \ 1 \ 1 \ 1]$ and $[0 \ 0 \ 0 \ \frac{1}{3} \ \frac{1}{3} \ \frac{2}{3} \ \frac{2}{3} \ 1 \ 1 \ 1]$; (iii) The number of control points: 10×8 . Rest of the parameters for GACO algorithm are set as same as in above model. With the proposed approach, the optimized locations of the two dimensional control points and their corresponding weights are obtained in both image planes (shown with blue * in Fig. 3c, d). Using third view, the required parameters are evaluated in three dimensional space (shown by colored * in Fig. 3e). The generated NURBS surface through the parameters is shown in Fig. 3f.

Model 3: Vase Model For the validation of the algorithm with real examples, first we have taken two views (images)

of the same vase (or flask) as shown in Fig. 4a and b. Both views or images are captured with the help of a camera with different viewpoints. The objective is to construct the three dimensional model or surface corresponding to the provided two dimensional images. Before starting the GACO algorithm we have set the following inputs: (i) Order of the fitted NURBS surface: (4, 4); (ii) The knot vectors: $[0 \ 0 \ 0 \ 0 \ \frac{1}{4} \ \frac{1}{4} \ \frac{2}{4} \ \frac{2}{4} \ \frac{3}{4} \ 1 \ 1 \ 1 \ 1]$ and $[0 \ 0 \ 0 \ 0 \ \frac{1}{3} \ \frac{1}{3} \ \frac{2}{3} \ \frac{2}{3} \ 1 \ 1 \ 1 \ 1]$; (iii) The number of control points: 9×8 . Rest of the GACO parameters are same as mentioned in “Selection of parameters”. The three dimension model or surface corresponding to two views with the reconstructed control points (shown with black *) and corresponding weights is shown in Fig. 4c. The main highlight of the presented algorithm in this case is the reduction of third view. We have not used the concept of third view here, still we get very good visual result.

Fig. 3 Reconstruction results of surface: **a, b** Two dimensional projected data points in left and right views; **c, d** Updated control points by GACO model in both image planes; **e** Reconstructed control points in space; **f** Reconstructed NURBS surface; **g** Testsurface

Fig. 4 Reconstruction results of Vase: **a, b** Two dimensional views of Vase; **c** Reconstructed NURBS Vase Model

Model 4: Tsukuba Statue For another example, we have taken the Tsukuba image which is publicly available several years ago. Again for the two view reconstruction we have taken two subsequent frames (Fig. 5a, b) of the Tsukuba image captured from slightly different viewpoints. The main objective here is to reconstruct the three dimensional curve of the boundaries of the Tsukuba statue. Here the inputs are given in the following form: (i) Order of NURBS: 4; (ii) The knot vector: $[0 \ 0 \ 0 \ 0 \ \frac{1}{10} \ \frac{1}{10} \ \frac{2}{10} \ \frac{3}{10} \ \frac{5}{10} \ \frac{6}{10} \ \frac{8}{10} \ \frac{9}{10} \ \frac{9}{10} \ 1 \ 1 \ 1 \ 1]$; (iii) The number of control points: 13. Rest of the parameters are same as before. To the total of 429 two dimensional data points for boundary curve reconstruction, the optimized locations of NURBS parameters by GACO model in both views are shown by $\diamond$ in Fig. 5c and d. Finally the NURBS

curve of Tsukuba statue by GACO model is shown in Fig. 6e.

Model 5: Giraffe Image In the next example, we have taken two views of a Giraffe (shown in Fig. 6a, b). In this example total 1914 two dimensional data points are taken into consideration for the reconstruction process of outer boundary of the giraffe. With the proposed approach, the reconstructed outer shape of the body of the giraffe is shown in the Fig. 6c.

Error analysis

For the quantitative error analysis, we have evaluated Mean, Median, Best, Worst and Standard deviation (SD, σ) of the values of objective function determined by the GACO algo-

Fig. 5 Reconstruction results of Tsukuba statue: **a, b** Two dimensional views of Tsukuba model; **c, d** Updated control points by GACO model in both views; **e** Reconstructed NURBS Tsukuba Model

Fig. 6 Reconstruction results of Giraffe: **a, b** Two dimensional images of Giraffe; **c** Reconstructed boundary of NURBS curve

rithm. All the relevant terms are calculated over 100 runs and displayed in Table 2. It is observed from this table that the reconstruction errors are very less and can be inferred as almost negligible which shows the effectiveness of the proposed algorithm very well. But overall performance of any algorithm will be tested in presence of noise. We have also checked the robustness of the presented theory.

The procedure to add the noise in the reconstruction process is as follows: the white Gaussian noise (Mean=0 and different variance) is added to the data points of both views

(2D image planes). This step will perturb the actual position of the data points and induce some error in the whole reconstruction process. We have to study how much it will going to affect the actual reconstruction process. This study has been presented in Table 3. We have induced different types of noise level i.e., variances ($\sigma = 0.1$ to 1). Under each noise level, mean, median, best, worst and SD are reported and it is observed that the reconstruction errors are significantly small in presence of different level noise also. Figure 7a shows the robustness of the presented theory in case of helix. Here

Table 2 Mean, median, best, worst and standard deviation (SD) of errors over 100 runs

Experiment	Mean	Median	Best	Worst	SD
Helix	$1.13E-08$	$1.12E-08$	$1.12E-08$	$1.12E-07$	$4.92E-18$
Testsurface	$2.25E-15$	$1.94E-15$	$1.94E-15$	$1.32E-14$	$2.85E-19$

mean of the reconstruction errors are shown under different amount of induced noise. There is no deflection in the curve upto the noise level $\sigma = 0.6$. After that small changes in the deflections are shown but still no rapid change. Similar analysis has been done in case of the surface example also (shown in Fig. 7b). From Fig. 7b, it may be concluded that the induction of higher levels of noise does not produce any kind of rapid increment in the reconstruction errors. Hence this analysis presents the robustness of the presented theory very well. The convergence behavior of the presented study has been demonstrated by the means of error plots. The errors (mean, median, SD) are observed over iterations. The convergence plot for the first synthetic example is shown in Fig. 8a. It shows that almost after the 60^{th} iteration, the errors are constant. Whereas in case of the surface example (Fig. 8b), the errors are converging very fast i.e., almost after the 15th iteration.

Comparison study

As discussed in the above sections, the proposed GACO approach towards the two view reconstruction process performs very well. To support our claim, a detailed comparison analysis is presented in this section. For this study, we have compared the reconstruction results for the curve as well as the surface. The curve is taken of the form:

$$\begin{cases} X = 2 \cos t \\ Y = 2 \sin t \\ Z = 2(t + 1) \end{cases} \quad t \in [0, \frac{5\pi}{4}].$$

This curve is projected in left and right image planes using these two projection matrices: $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$ and $\begin{pmatrix} 1 & 0 & 0 & -20 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$.

The number of data points used for comparison are 31. The parameters for GACO algorithm are set as follows: (i) NURBS order: 3; (ii) The knot vector: $[0 \ 0 \ 0 \ \frac{1.45}{3} \ \frac{2.38}{3} \ 1 \ 1 \ 1]$; (iii) The number of control point: 5. Figure 9 represents the qualitative comparison. Left part of the Fig. 9 is the reconstruction result based on point-based methodology [3] and right part of the Fig. 9 is the result by the proposed GACO approach. This comparison gives the support to our claim to get smooth and flexible reconstruction.

Another example for the comparison is considered as the

$$\text{given surface: } \begin{cases} X = \text{linspace}(-2, 2, 20) \\ Y = \text{linspace}(-2, 2, 20) \\ (X \ Y) = \text{meshgrid}(X, Y) \\ R = (\sqrt{X^2 + Y^2}) \\ Z = \sin(R)/R \\ \text{surf}(X, Y, Z) \end{cases}$$

This surface is projected in left and right image planes

using the projection matrices: $\begin{pmatrix} 1 & 0 & 0 & 40 \\ 0 & 1 & 0 & 20 \\ 0 & 0 & 1 & 1 \end{pmatrix}$ and

$\begin{pmatrix} 1 & 0 & 0 & -20 \\ 0 & 1 & 0 & -10 \\ 0 & 0 & 1 & 1 \end{pmatrix}$. The parameters are set for 400 data

points: (i) NURBS order: (3, 3); (ii) The knot vectors $[0 \ 0 \ 0 \ \frac{1}{3} \ \frac{1}{3} \ \frac{2}{3} \ \frac{2}{3} \ \frac{2}{3} \ \frac{2}{3} \ 1 \ 1 \ 1]$ and $[0 \ 0 \ 0 \ \frac{1}{4} \ \frac{1}{4} \ \frac{2}{4} \ \frac{2}{4} \ \frac{3}{4} \ \frac{3}{4} \ 1 \ 1 \ 1]$; (iii) Number of control points: 80. To illustrate the advantages of the proposed algorithm, we have compared the reconstruction result of this surface with ACO approach [28]. In their study, simple ACO algorithm was used for three dimensional NURBS fitting. To get the input for two view NURBS fitting, we have projected the three dimensional data [28] onto the base planes (view planes) with the help of same projection matrices. The qualitative analysis is shown in Fig. 10. Figure 10a is the reconstruction result from the reference [28] while Fig. 10b corresponds to the proposed GACO approach. The obtained results also show the better smoothness and flexibility. Another mode of comparison i.e. quantitative analysis is also presented here for both the reference examples. For this again Mean, Median, Best and SD are obtained for the reconstruction results under various levels of noise. Table 4 represents the obtained error in both cases, where each block indicates the Mean, Median, Best and SD in decreasing orders. From this table, it is observed that reconstruction errors in both examples by GACO approach is much smaller than the other considered methods. The computational efficiency is also displayed through the Table 5. In this table, the reconstruction time periods for all the reference examples (Figs. 2,3,9,10) are shown. This table also shows that the total computation time for the proposed algorithm is smaller than the other existed state of the art algorithms. Hence with all the presented analysis, we may conclude that our proposed algorithm outperforms others in terms of accuracy, efficiency and flexibility.

Table 3 Reconstruction errors in presence of various amounts of noise

Experiment	$\sigma = 0.1$	$\sigma = 0.2$	$\sigma = 0.3$	$\sigma = 0.4$	$\sigma = 0.5$
Helix	$\begin{pmatrix} 1.14E-08 \\ 1.13E-08 \\ 1.14E-08 \\ 1.13E-07 \\ 4.95E-18 \end{pmatrix}$	$\begin{pmatrix} 1.23E-07 \\ 1.24E-07 \\ 1.24E-07 \\ 1.26E-07 \\ 5.85E-17 \end{pmatrix}$	$\begin{pmatrix} 1.34E-05 \\ 1.33E-05 \\ 1.34E-05 \\ 1.33E-06 \\ 4.93E-16 \end{pmatrix}$	$\begin{pmatrix} 1.55E-04 \\ 1.55E-04 \\ 1.58E-04 \\ 1.59E-05 \\ 8.55E-15 \end{pmatrix}$	$\begin{pmatrix} 2.04E-04 \\ 2.03E-04 \\ 2.04E-04 \\ 2.03E-05 \\ 11.33E-15 \end{pmatrix}$
	$\begin{pmatrix} 2.85E-15 \\ 2.78E-15 \\ 2.34E-15 \\ 2.12E-14 \\ 3.85E-19 \end{pmatrix}$	$\begin{pmatrix} 2.65E-14 \\ 2.24E-14 \\ 2.94E-14 \\ 2.32E-12 \\ 5.85E-17 \end{pmatrix}$	$\begin{pmatrix} 3.25E-13 \\ 2.94E-13 \\ 2.94E-13 \\ 3.32E-12 \\ 7.85E-16 \end{pmatrix}$	$\begin{pmatrix} 5.10E-11 \\ 4.87E-11 \\ 4.90E-11 \\ 6.23E-10 \\ 8.30E-13 \end{pmatrix}$	$\begin{pmatrix} 6.25E-10 \\ 6.94E-10 \\ 6.94E-10 \\ 6.32E-09 \\ 10.85E-12 \end{pmatrix}$
	$\sigma = 0.6$	$\sigma = 0.7$	$\sigma = 0.8$	$\sigma = 0.9$	$\sigma = 1.0$
	$\begin{pmatrix} 2.14E-03 \\ 2.32E-03 \\ 2.22E-03 \\ 2.36E-04 \\ 10.37E-11 \end{pmatrix}$	$\begin{pmatrix} 4.22E-02 \\ 4.03E-02 \\ 4.28E-02 \\ 5.77E-03 \\ 9.29E-09 \end{pmatrix}$	$\begin{pmatrix} 6.99E-02 \\ 7.01E-02 \\ 6.87E-02 \\ 6.98E-03 \\ 10.91E-08 \end{pmatrix}$	$\begin{pmatrix} 6.99E-01 \\ 7.01E-01 \\ 6.87E-01 \\ 6.98E-01 \\ 10.91E-04 \end{pmatrix}$	$\begin{pmatrix} 10.89E-01 \\ 11.12E-01 \\ 10.87E-01 \\ 16.34E-01 \\ 11.01E-03 \end{pmatrix}$
	$\begin{pmatrix} 6.01E-08 \\ 6.03E-08 \\ 6.10E-08 \\ 6.99E-07 \\ 10.21E-10 \end{pmatrix}$	$\begin{pmatrix} 6.31E-06 \\ 6.23E-06 \\ 6.30E-06 \\ 7.29E-05 \\ 15.11E-08 \end{pmatrix}$	$\begin{pmatrix} 8.21E-04 \\ 7.99E-04 \\ 8.02E-04 \\ 9.99E-05 \\ 21.35E-07 \end{pmatrix}$	$\begin{pmatrix} 6.45E-02 \\ 6.64E-02 \\ 6.92E-02 \\ 6.99E-03 \\ 15.55E-04 \end{pmatrix}$	$\begin{pmatrix} 1.51E-01 \\ 1.45E-01 \\ 1.53E-01 \\ 1.33E-01 \\ 2.34E-02 \end{pmatrix}$

Each column vector shows the mean, median, best, worst and SD in descending order

Fig. 7 Mean errors of reconstruction under different noise levels in case of **a** Helix; **b** Testsurface **Fig. 8** **a** Convergence behaviour of Helix; **b** Convergence behaviour of Testsurface

Fig. 9 Comparison analysis of a curve by: **a** Point-based Approach; **b** Our Algorithm

Fig. 10 Comparison analysis of a surface by: **a** Point-based Approach; **b** Our Algorithm

Table 4 Errors of reconstruction with corrupted data in case of curve (Fig. 9) and surface (Fig. 10)

Noise level	Curve (Fig. 9)		Surface (Fig. 10)	
	Our approach	Point-based [3]	Our approach	ACO [28]
$\sigma = [0.1]$	0.0389	0.5218	0.2112	0.6105
	0.0301	0.5111	0.2002	0.5913
	0.2462	0.7132	0.3484	0.7643
	0.0422	0.1233	0.3328	0.6003
$\sigma = [0.2]$	0.0549	0.7118	0.3105	0.7321
	0.0533	0.6126	0.2918	0.6723
	0.2681	0.8182	0.3778	0.9897
	0.0556	0.2219	0.3623	0.7125
$\sigma = [0.3]$	0.0685	0.9112	0.3889	0.9825
	0.0532	0.8138	0.3562	1.2352
	0.2883	0.9116	0.4157	2.3232
	0.0670	0.3412	0.4025	1.1235
$\sigma = [0.4]$	0.0885	1.1235	0.4562	2.5232
	0.0798	1.0923	0.4102	2.0234
	0.3530	2.0135	0.5330	4.0813
	0.0770	1.1895	0.4638	3.1926
$\sigma = [0.5]$	0.0922	6.1111	0.5823	8.2333
	0.0892	5.3229	0.5205	6.1923
	0.3892	8.1352	0.6833	10.1526
	0.0855	4.6264	0.5287	7.1925

Table 5 Average time in reconstruction process

Model	Our approach	Compared With
Helix (Fig. 2)	105.1004	116.2932 [7]
Testsurface (Fig. 3)	129.3269	240.3052 [7]
Curve (Fig. 9)	025.2378	082.1685 [3]
Surface (Fig. 10)	114.2523	123.0009 [28]

Conclusion

In this study, we have presented a reconstruction methodology for free-form shapes in space using only two views. In particular, through a given set of data points in stereo views, a NURBS curve of definite order is assumed to fit. GACO model is used for the optimization of the fitting error between the data points and the fitted NURBS curve. The optimization process results in the best values of control points and their corresponding weights in both views. Finally, Triangulation method has been applied to get the related parameters (control points and weight vectors) in space, and thus actually returning the reconstructed NURBS curve. This method has been extended for the NURBS surface also. A detailed experimental study and error analysis has been conducted in support of the proposed approach. The main highlights of the presented algorithm are floated here:

- **Very Effective** Proposed method has been tested over several synthetic as well as real images. We did not foist any restriction to the input images or data points. The methodology is proven to be very effective in all cases. Even in case of the images with lost data (as shown in Fig. 4), the performance is remarkable.
- **On the Mark** All the examples taken for the analysis shows that the measured errors are very less. It means that the method performs well in terms of numerical accuracy. Moreover the reconstruction method reconstructs the shape of the objects with high accuracy.
- **General** The method is very general and can be applied to all complex problems. In case of curve, surface or boundary evolving problems, the method gives accurate results.
- **Robust** As we have discussed in the above study that the reconstruction problem in three dimensional space is an ill-posed problem. So small error in the fitting process in two dimensional space produce a large deviation in the reconstructed shape. But in the presented error analysis, it is clear that the method performs extremely good in case of large noisy input data also.

In this work, we have used the optimized values of NURBS control points and their corresponding weight vectors for

required result. There are some other parameters also which we have fixed during our procedure such as knot vectors, parametric values of data points etc. Type of parametrization method of data points is also a major concern here. In near future, we want to switch to some other parametrization methods also like centripetal, universal etc. and will try to give more emphasis on the mentioned parameters also for the reconstruction process.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Faugeras O (1993) Three-dimensional computer vision: a geometric viewpoint. MIT Press, Cambridge
2. Forsyth DS, Ponce J (2011) Computer vision: a modern approach, 2nd edn. Pearson Education, Singapore
3. Aharchi M, Ait Kbir M (2020) A review on 3D reconstruction techniques from 2D images. https://doi.org/10.1007/978-3-030-37629-1_37, 510–530
4. Ummenhofer B, Brox T (2013) Point-based 3D reconstruction of thin objects. In: IEEE International Conference on Computer Vision 969–976
5. Ma WY, Kruth JP (1995) Parametrization of randomly measured points for least-squares fitting of B-spline curves and surfaces. *Comput Aided Design* 27:663–675
6. Xiao Y, Li FY (2005) Optimized stereo reconstruction of free-form space curves based on a non-uniform rational B-spline model. *J Opt Soc Am* 22(9):1746–1762
7. Saini D, Kumar S, Gulati TR (2017) NURBS-based geometric inverse reconstruction of free-form shaped objects. *JKSU-Comput Inf Sci* 29(1):116–133 (Elsevier)
8. Piegel L (1991) On NURBS: a survey. *IEEE Comput Graphics Appl* 11:55–71
9. Piegel L, Tiller W (1995) The NURBS book. Springer, Berlin
10. Peng K, Chen X, Zhou D, Liu Y, Zhai Y (2012) 3D reconstruction using image sequences based on projective depth and simplified iterative closest point. *Opt Eng* 51(2):021110
11. Cai Y, Su Z, Li Z, Sun R, Liu X, Zhao Y (2011) Two-view curve reconstruction based on the Snake model. *J Comput Appl Math* 236:631–639
12. Saini D, Kumar S, Gulati TR (2015) Reconstruction of free-form space curves using NURBS-snakes and a quadratic programming approach. *Comput Aid Geometric Design* 33:30–45
13. Lu Y, Yong JH, Shi KL, Song HC, Ye TY (2016) 3D B-spline curve construction from orthogonal views with self-overlapping projection segments. *Comput Graph* 54:18–27
14. Ning J, McClean S, Cranley K (2001) 3D reconstruction from two orthogonal views using simulated annealing approach. In: Third

- international conference on 3D digital imaging and modeling. pp 309–313
15. Ogura T, Sato C (2006) A fully automatic 3D reconstruction method using simulated annealing enables accurate posterior angular assignment of protein projections. *J Struct Biol* 156(3):371–386
 16. Chen YW, Kohatsu T (2007) 3D image reconstruction from limited projections by simulated annealing. In: Second international conference on innovative computing, information and control, 2007. ICICIC'07, 456–456
 17. Voisin S, Abidi MA, Fofou S, Truchetet F (2009) Genetic algorithms for 3D reconstruction with supershapes. In: 16th International conference on image processing, IEEE, 529–532
 18. Koch A, Dipanda A (2012) Evolutionary-based 3D reconstruction using an uncalibrated stereovision system: application of building a panoramic object view. *Multimedia Tools Appl* 57(3):565–586
 19. Singh A, Deep K (2018) Reconstruction of 3D curves and surfaces using new variants of gravitational search algorithm. *J Inf Optimiz Sci* 1–23
 20. Alazzam A, AlOmar B (2017) Using average uniform algorithm to model educational data. In: The fourth HCT information technology trends (ITT 2017), Dubai, UAE, Oct., 25–26, 30–34
 21. Shtovba SD (2005) Ant algorithms: theory and applications. *Program Comput Softw* 31(4):167
 22. Balamurugan R, Natarajan AM, Premalatha K (2015) Stellar-mass black hole optimization for biclustering microarray gene expression data. *Appl Artif Intell Int J* 29(4):353–381
 23. Bianchi L, Dorigo M, Gambardella LM, Gutjahr WJ (2009) A survey on meta-heuristics for stochastic combinatorial optimization. *Nat Comput* 8(2):239–287
 24. Dorigo M (1992) Optimization, learning and natural algorithms (in Italian). Ph.D. Thesis, Dipartimento di Elettronica, Politecnico di Milano, Italy
 25. Dorigo M, Maniezzo V, Colnari A (1996) Ant system: optimization by a colony of cooperating agents. *IEEE Trans Syst Man Cybernet Part B* 26(1):29–41
 26. Kumar A, Bawa S (2019) Generalized ant colony optimizer: swarm-based meta-heuristic algorithm for cloud services execution. *Computing* 101:1609–1632
 27. Blum C, Roli A (2003) Meta-heuristics in combinatorial optimization: overview and conceptual comparison. *ACM Comput Surv* 35(3):268–308
 28. Xiao R, Zhang J, Liu H (2012) NURBS fitting optimization based on ant colony algorithm. *Adv Mater Res* 549:988–992
 29. Chrysostomou D, Gasteratos A, Nalpantidis L, Sirakoulis GC (2012) Multi-view 3D scene reconstruction using Ant Colony Optimization Techniques. *Meas Sci Technol* 23(114002):1–11
 30. Dorigo M, Blum C (2005) Ant Colony Optimization theory: a survey. *Theor Comput Sci* 344(2–3):243
 31. Socha K, Dorigo M (2008) Ant Colony Optimization for continuous domains. *Eur J Oper Res* 185(3):1155
 32. Merkle D, Middendorf M (2003) Ant Colony Optimization with global pheromone evaluation for scheduling a single machine. *Appl Intell* 18(1):105
 33. Hj SSM, Ahmed M, Smian (2006) NURBS skinning surface for ship hull design based on new parametrization method. *Int J Adv Manuf Technol* 28:936–941 (**Springer**)
 34. Shamsuddin SMH and Ahmed M (2004) A hybrid parametrization method for NURBS. In: Proceedings international conference on graphics, imaging and visualization 2004 (CGIV2004), IEEE Press, 26–29 July
 35. Zhang X, Zhang H, Gao M (2010) Continuous Ant Colony Optimization Algorithm based on crossover and mutation. In: 2010 Sixth international conference on natural computation (ICNC 2010), pp 2605–2608

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.