Survey of Object Tracking and Feature Extraction Using Genetic Algorithm

Ashish Shingade, Archana Ghotkar
1Dept. of Computer Engineering, Pune Institute of Computer Technology, Pune, India

Abstract
Object tracking and feature extraction are the important steps in any computer vision application. With availability of various object tracking and feature extraction techniques, in this paper genetic algorithm technique is explored. Genetic algorithm is used for optimization, searching, feature extraction, segmentation and classification. Traditional methods search from a single point where as genetic algorithm search parallel from a population of points and find the global optimum in relatively few evaluations. Genetic algorithm is robust and efficiently searches complex search space. Comparative study of feature extraction techniques with genetic algorithm is presented to give guidelines for the researchers in the same domain.

Keywords
Genetic Algorithm, Object Detection, Tracking, Feature Extraction

I. Introduction
Genetic Algorithms (GAs) are search techniques that are based on concept of natural selection and natural genetics [1]. GAs are a particular class of evolutionary algorithms that use the techniques inspired from evolutionary biology. The main advantage of the GA design is that the accurate results may be obtained using a very simple algorithm. It is a method of finding a good answer to the problem, based on feedback received from its repeated attempts to a solution. GA was developed to simulate process observed in natural evolution, a process that operates on chromosomes (string, individual) [2]. The difference between GA and other search methods is that it searches among a population of points, and works with a coding of parameter value themselves. The transition scheme of GA is probabilistic, whereas traditional methods use gradient information. Because of these features of GA, they are used as general purpose optimization algorithm and also provide means to search irregular space and hence are applied to a variety of functions optimization, feature extraction, classification, segmentation, parameter estimation, computer vision and machine learning applications.

Object detection and tracking process are the starting steps for any video processing application, such as animation, gaming and surveillance. Object tracking in videos can be defined as a method of following an object of interest through successive image frames to determine its relative movement with respect to other objects. The challenges of object detection and tracking are handling of orientation and occlusion. Object tracking is an easy task when the shape, direction and speed of moving object are constant. Object can be a person, vehicle or any entity that is of interest.

The two major sources of information in video that can be used to track object are visual features (such as color, texture and shape) and motion features. The object detection and tracking have many applications such as video surveillance, robot vision, traffic monitoring, vehicle navigation and animation. In this paper, GA is introduced with background subtraction to improve the efficiency of overall system. GAs advantages are to improve robustness, accuracy and flexibility of system [1]. GA can be used for feature extraction, providing those features that maximize the performance of a system. Feature extraction is a special form of dimensionality reduction. The GA based feature extraction technique provides best feature to be extracted that reviews the original input compared with the other feature extraction techniques. It maximizes the performance of the system.

This paper gives the survey on use of GA in different applications, object tracking and shape feature extraction. The paper is organized as follows. Section II describes concept of GA, section III present object tracking using GA, section IV gives the comparative study of feature extraction techniques and in section V we present shape feature extraction using GA and conclusion is presented in section VI.

II. Genetic Algorithm
GAs are methods used in computing to find true or appropriate solution to search and optimization problems based on principles of natural selection and genetics. Here GA and associated terminology is explained.

GA uses expressions from natural genetics. The individuals (or genotype) in a population are called as strings or chromosomes. Chromosomes are made of units- genes (also features, characters, or decoders), every gene controls the inheritance of one or more character. The genes of characters are located in a chromosome, which are called loci (string positions). Any features of an individual (such as skin color) can manifest itself in different ways the gene have states called alleles (feature values) as shown in Table 1.

Each genotype (a single chromosome) represents a solution to a problem; a biological process runs on a population of chromosomes corresponding to a search over a space of possible solutions [5]. A GA performs a search by maintaining a population of potential solutions and inspiring information formation and exchange between these directions. The population undergoes a simulated development; at each generation the good solution is reproduced and the bad solutions are ignored. To distinguish between the solutions, an objective (fitness) function is used, which plays the role of the selection process in the biological environment.

A GA starts with an initial set of population. Each individual in the population is a chromosome, representing a solution to the problem. A chromosome is a string structure, typically a list of binary digits representing a coding of control parameters of a given problem. The chromosomes progress through successive iterations called generations. The chromosomes of each generation are evaluated using measure of fitness. To create the next generation, new chromosomes called offspring are formed by either a crossover operator or a mutation operator. A new generation is formed by selecting, according to fitness value, some of the parents and offspring, and rejecting others not fit in order to keep the population size constant. Chromosomes having higher fitness values are selected. Normally, the roulette-wheel selection approach is used. After several generations, the algorithm meets to the best chromosome, which represents the optimal or near
optimal solution to the problem [5].

Table 1: Explanation of Genetic Algorithm Terms

<table>
<thead>
<tr>
<th>Genetic Algorithm terms</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chromosome (string, individual)</td>
<td>Solution</td>
</tr>
<tr>
<td>Genes (bits)</td>
<td>Part of Solution</td>
</tr>
<tr>
<td>Locus</td>
<td>Position of gene</td>
</tr>
<tr>
<td>Alleles</td>
<td>Values of gene</td>
</tr>
<tr>
<td>Phenotype</td>
<td>Decoded solution</td>
</tr>
<tr>
<td>Genotype</td>
<td>Encoded solution</td>
</tr>
</tbody>
</table>

The Algorithm works as follow
1. Choose the initial population of the individuals.
2. Evaluate the fitness of each individual in that population.
3. Repeat on this generation until termination (time limit, sufficient fitness achieved etc.).
   a. Select the best fit individual for reproduction.
   b. Produce new individuals through crossover and mutation operation to give birth to offspring.
   c. Evaluate the individual fitness of new individuals.
   d. Replace least fit population with new individuals.

The fig. 1 shows flowchart of implementation of GA for any application.

![Flowchart of Genetic Algorithm](image)

Fig. 1: Flowchart of Genetic Algorithm

Applications of Genetic Algorithm
As seen from the above description of the working principle of GA, GA is different from the other traditional optimization methods. GA works with a string-coding of variables instead of the variables. The advantage of working with a coding of variables is that the coding discretizes the search space. On other hand, since GA require only function values at various discrete point, a discrete or discontinuous function can be handled with no extra burden. This allow GA to be applied a wide variety of problems. The difference between GA and many traditional optimization methods is that GA work with population of points instead of a single point. Because there is more than one string being processed simultaneously, it is very likely that expected GA solution may be a global solution. Due to such advantages GAs are applied to many scientific, engineering problems, in business and entertainment, including following applications:

- **Computer Vision:** GA has been used in classification, image segmentation, feature extraction of images and various other computer vision applications [3][4].
- **Machine Learning:** GA has being used in many machine learning applications, including classification and prediction. GA is also implemented to design neural networks, to design and control robots.
- **Automatic programming:** GA has been used to develop computer programs for specific task, and to design other computational structures.
- **Optimization:** GA can be applied to numerical optimization, combinatorial optimization problems such as traveling salesman problem (TSP), circuit designing, job shop scheduling, and video and sound quality optimization [2].

The Table 2 shows the use of GA by different authors in various applications and the efficient use of GA.

Table 2: Survey of Applications of Genetic Algorithm Used by Different Authors

<table>
<thead>
<tr>
<th>Use of Genetic Algorithm</th>
<th>Result</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature selection</td>
<td>GAs give good feature selection compared with other algorithms</td>
<td>[14]</td>
</tr>
<tr>
<td>Feature selection</td>
<td>Genetic algorithm maximizes the classification percentage</td>
<td>[13]</td>
</tr>
<tr>
<td>Segmentation</td>
<td>Good segmentation was achieved using Genetic Algorithms</td>
<td>[15]</td>
</tr>
<tr>
<td>Feature Selection</td>
<td>Advantage in Performance and speed using GAs</td>
<td>[12]</td>
</tr>
<tr>
<td>Feature Selection, Feature Extraction and Classification</td>
<td>Best feature selection, feature extraction and classification achieved using genetic algorithm</td>
<td>[8]</td>
</tr>
<tr>
<td>Feature Extraction and Classification</td>
<td>Improved performance of feature extraction and classification using genetic algorithm</td>
<td>[3]</td>
</tr>
</tbody>
</table>
## III. Object Tracking

In this paper we propose a method for object detection and tracking using background subtraction and GA. The [28, 30-33] shows the implementation of various object tracking algorithms. The steps in this system are image acquisition, object detection, object tracking and object processing, the object detection includes some sub phases as background model, foreground detection, object and feature extraction. The proposed system block diagram is shown in fig. 2.

### A. Image Acquisition

Image acquisition is the initial step in which frames from real time video are taken as input to the system and provided for further steps foreground detection and background model for object detection.

### B. Object Detection

The object detection phase consists of following sub phases

1. **Background Model**
   - The centroid (x, y) of the object is computed
   - Genetic algorithm is used to accurately search the object in successive frames
     - If genetic algorithm accurately searches the object in frame i to i+1
     - Then the motion vector is evaluated as (x’-x, y’-y) where (x’, y’) is the estimated centroid of the object in frame i+1
     - The process is repeated until the object is present in the frame
     - Else terminate

   The input to the GA is the detected object (from the frame) and the centroid (x, y) of the object.

   Genetic algorithm terms:
   1. Population Initialization: The centroid value of the detected object is encoded as the chromosome
   2. For every chromosome
   3. Selection: Select chromosomes for mutation or crossover
   4. Crossover: All chromosomes that are chosen for cloning are directly taken into the next generation
   5. Mutation: All Chromosomes chosen for mutation are replaced with uniformly distributed random centroid values
   6. Termination: If number of generations exceeds a predefined threshold value

   Background model learns the background image, which is taken at the time of initialization. The image is taken when there are no any moving objects; only stationary objects and background is taken in consideration. The background image is then given to object detection phase.

2. **Object Tracking**

   Once the object is detected in each frame, the tracking algorithm is applied to track the object from frame to frame.

   In this section we present GA for object tracking. Here the input to GA is the detected object from frames with feature as centroid. The algorithm for tracking object is stated below.

   **Algorithm**
   1. For every frame i
   2. The centroid (x, y) of the object is computed
   3. Genetic algorithm is used to accurately search the object in successive frames
      - If genetic algorithm accurately searches the object in frame i to i+1
      - Then the motion vector is evaluated as (x’-x, y’-y) where (x’, y’) is the estimated centroid of the object in frame i+1
      - The process is repeated until the object is present in the frame
      - Else terminate

3. **Object and Feature Extraction**

   This step extracts useful features from the sequence of frames; feature extraction is done to reduce the amount of information to be processed [34-35]. The object detection is performed by extracting the features of each object. In this paper centroid and shape of object are used as major features for object detection. Once the features are extracted the object is detected on the base of features and the bounding box is applied to represent the detected object.

### C. Object Tracking

Once the object is detected in each frame, the tracking algorithm is applied to track the object from frame to frame.

In this section we present GA for object tracking. Here the input to GA is the detected object from frames with feature as centroid. The algorithm for tracking object is stated below.

**Algorithm**

1. For every frame i
2. The centroid (x, y) of the object is computed
3. Genetic algorithm is used to accurately search the object in successive frames
   - If genetic algorithm accurately searches the object in frame i to i+1
   - Then the motion vector is evaluated as (x’-x, y’-y) where (x’, y’) is the estimated centroid of the object in frame i+1
   - The process is repeated until the object is present in the frame
   - Else terminate

The input to the GA is the detected object (from the frame) and the centroid (x, y) of the object.

Genetic algorithm terms:
1. Population Initialization: The centroid value of the detected object is encoded as the chromosome
2. For every chromosome
   - Fitness Evaluation: is given by \( \text{Diff}_{\text{frame}} = |V(x, y, t+1) - V(x, y, t)| \)
3. Selection: Select chromosomes for mutation or crossover
4. Crossover: All chromosomes that are chosen for cloning are directly taken into the next generation
5. Mutation: All Chromosomes chosen for mutation are replaced with uniformly distributed random centroid values
6. Termination: If number of generations exceeds a predefined threshold value
D. Object Processing
The tracked object is to be processed; the processing of object is done using bounding box. Bounding box is applied on the tracked object to get object of interest from the input video.

E. Mathematical Model
The mathematical model for the proposed system is stated below.

Let S be the system, such that

\[ S = \{I_n, O_u, F_u, S_u, F_a\} \]

1. Input
   \[ I_n = \{i_1, i_2, i_3, \ldots, i_n\} \]
   Set of input sequence of frames from video.

2. Output
   \[ O_u = \text{Set of bounding box applied on tracked object on each frame.} \]

3. Functions
   \[ F_u = \{f_1, f_2, f_3, f_4\} \]
   \[ f_1 = \text{background} \_\text{model}() \]
   \[ f_2 = \text{foreground} \_\text{model}() \]
   Background subtraction with frame differencing
   \[ \text{Frame differencing at time} \ t+1 \text{ is} \]
   \[ \text{Diff} \_\text{frame} = |V(x, y, t+1) - V(x, y, t)| \]
   \[ |V(x, y, t) - V(x, y, t+1)| > T_h \]
   \[ V = \text{pixel intensity} \]
   \[ T_h = \text{Threshold} \]
   \[ f_3 = \text{feature} \_\text{extraction}() \]
   \[ \text{Centroid} = (x, y) \]
   \[ x = \frac{\sum (An \times x)}{\sum (An)} \]
   \[ y = \frac{\sum (An \times y)}{\sum (An)} \]
   \[ f_4 = \text{object} \_\text{tracking}() \]

4. Success
   \[ S_u = \text{Correct tracking of the object}. \]

5. Failure
   \[ F_a = \text{False tracking of object or object not tracked due to occlusion, environmental factors (like fog, lightning, rainfall, snowfall).} \]

IV. Comparative Study of Feature Extraction Techniques
There are many methods for feature extraction mentioned in [10, 11, 16-29]. Table 3 reviews some of the works published for feature extraction. Some of the commonly used techniques are Principal Component Analysis (PCA), hough transformation (HT), Independent Component Analysis (ICA), Linear Discriminant Analysis (LDA), and Genetic Algorithm (GA). The following table shows the use of various feature extraction techniques.

<table>
<thead>
<tr>
<th>Feature Extraction Technique</th>
<th>Application</th>
<th>Number of Feature Points</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hough transform</td>
<td>Iris Recognition</td>
<td>Outer iris and pupil</td>
<td>[28]</td>
</tr>
</tbody>
</table>

Table 3: Comparative Study of Feature Extraction Techniques by Various Methods

V. Shape Feature Extraction
Feature extraction is a method in which the original set of features is transformed to provide a new set of features. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data, in order to perform the chosen task using this reduced representation instead of the full size input. The objective of this is to find a reduced subset among the original N features such that useful class discriminatory information is included and redundant class information and noise is excluded [3-4]. Shape feature extraction is a particular set of feature extraction.

To describe shapes, geometric features can be used. A shape can be described by different features. These shape parameters are Center of gravity, Axis of inertia, Solidity, Digital bending energy,
Eccentricity, Circularity ratio, Elliptic variance, Rectangularity, Convexity, Euler number, Profiles, Hole area ratio [7].

Genetic Algorithm for Shape Feature Extraction
The input to a shape feature extraction algorithm consists of N points in Cartesian space and the equation defining the particular geometric shape to be extracted. Some examples of geometric shapes are lines, circles, ellipses in 2D and planes, spheres, cones, cylinders in 3D. The extraction process will find the best shape of given data. Single geometric shape acts as an individual and the chromosome is a set of p genes, each of which is one of the points in the minimal subset defining the shape. Since there are N geometric data points, index is used to identify each minimal point, which is in the range from 1 to N; the process of shape feature extraction is shown in figure 3.

For GA to execute, a fitness function (objective function) must be provided. The input to the algorithm is chromosome defining the shape and the output is a scalar which defines the fitness of the shape. The fitness function counts the number of geometric data points in a fixed-sized template around the geometric shape. These points define a valid Shape. To create the initial population the geometric data are first divided into small subsets, then to get the initial population random sampling is performed on each subset. Two individuals (i.e. shapes) are chosen for reproduction from the population using a fitness function. The operation of crossover and mutation are applied to the chosen individuals to create new individuals. The individuals are replaced with the new individuals and the weak individuals are discarded.

The fitness function is used to rank the two new individuals which are then added to pool of population. The worst fit individuals are removed to keep the population pool size fixed. The GA is repeated till there is some indication of convergence. The final result of this process is population member with best score. The parameter vector is computed from the chromosome of this individual. To label an individual (shape), parameter vector and fitness function are used. GA is run repeatedly on the remaining geometric data points to extract all the shapes. [9].

The above study shows the efficient use of GA for feature extraction hence the GA methodology can be used to extract a wide range of geometric shapes.

VI. Conclusion
In this paper we explore that GA plays an important role in object tracking and feature extraction. Comparative study of use of GAs in various applications and its advantages are mentioned. We proposed an object tracking technique using GA and background subtraction. Different techniques for feature extraction are studied and proposed a GA approach for shape feature extraction. The study shows that the GA can be applied for wide range of problems.
References


