An efficient image matching algorithm based on culture evolution

Qinghua Wu\textsuperscript{1,2}, Jin Zhang\textsuperscript{1,2}, Wenzhi Huang\textsuperscript{1,2} and Yuxin Sun\textsuperscript{1,2}

\textsuperscript{1}Hubei Provincial Key Laboratory of Intelligent Robot, Wuhan Institute of Technology, Wuhan, China
\textsuperscript{2}School of Computer Science and Engineering, Wuhan Institute of Technology, Wuhan, China

ABSTRACT

In image matching research, how to ensure that best match’s accuracy of the premise and a significant reduction in the amount of computing is the focus of concern by researchers. Search strategy to find the best match location of the image matching process to determine the amount of computing of image matching, in the existing image matching method are used to traverse search strategy, it is difficult to reduce the amount of computing. This is a common defect of the existing image matching algorithm. Cultural Algorithms are a class of computational models derived from observing the cultural evolution process in nature, compared with genetic algorithm the cultural algorithms have high convergence speed. In this paper, compared with genetic algorithm on the Benchmarks function, the results show that the new algorithm is efficient, we also use the new algorithm to solve image matching problem and the experiment results show the new algorithm is effective for this problem.

Key words: Image Matching, Particle Swarm Optimization, Population, Fitness Function

INTRODUCTION

Image matching technology is a key technology used in aircraft or missile map matching navigation and positioning. The image matching is this process, use two different sensors to obtain two different sizes of images from the same area or the same scene respectively, and then compare these two images with the spatial location in order to determine the relative translation between these two images. In image matching the images obtained in advance, we refer to as base image, in the matching process we can obtain images in real time or online, we refer to as real time image. In image matching research, how to ensure that best match’s accuracy of the premise and a significant reduction in the amount of computing is the focus of concern by researchers. Search strategy to find the best match location of the image matching process to determine the amount of computing of image matching, in the existing image matching method are used to traverse search strategy, it is difficult to reduce the amount of computing. This is a common defect of the existing image matching algorithm [1].

Cultural Algorithms (CA) proposed by Reynolds in 1994 [2]. Cultural algorithm is in-depth analysis of the superiority of the original evolution theory on the basis of drawing on the social (cultural) evolution theory in the social sciences and has achieved broad consensus on the research results, and proposed a new algorithm. Cultural algorithm is used to solve complex calculations of the new global optimization search algorithms, cultural algorithms in the optimization of the complex functions of its superior performance.

1. CULTURAL ALGORITHM

1.1 BASIC CULTURAL ALGORITHM

Cultural Algorithms are a class of computational models derived from observing the cultural evolution process in nature [2-4]. In this algorithm, individuals are first evaluated using a performance function. The performance information represents the problem-solving experience of an individual. An acceptance function determines which individuals in the current population are able to impact, or to be voted to contribute, to the current beliefs. The
experience of these selected individual is used to adjust the current group beliefs. These group beliefs are then used to guide and influence the evolution of the population at the next step, where parameters for self-adaptation can be determined from the belief space. Information that is stored in the belief space can pertain to any of the lower levels, e.g. population, individual, or component. As a result, the belief space can be used to control self-adaptation at any or all of these levels. The cultural algorithm is a dual inheritance system with evolution taking place at the population level and at the belief level. The two components interact through a communications protocol. The protocol determines how the updated beliefs are able to impact and influence the adaptation of the population component.

The Cultural Algorithm is a dual inheritance system that characterizes evolution in human culture at both the macro-evolutionary level, which takes place within the belief space, and at the micro-evolutionary level, which occurs at the population space. CA consists of a social population and a belief space. Experience of individuals selected from the population space by the acceptance function is used to generate problem solving knowledge that resides in the belief space. The belief space stores and manipulates the knowledge acquired from the experience of individuals in the population space. This knowledge can control the evolution of the population component by means of the influence function. As a result, CA can provide an explicit mechanism for global knowledge and a useful framework within which to model self-adaptation in an EC system. The population level component of the cultural algorithm will be Evolutionary Programming (EP). The global knowledge that has been learned by the population will be expressed in terms of both normative and situational knowledge as discussed earlier.

A pseudo-code description of the Cultural Algorithms is described as follows:

Framework of cultural algorithm
1 BEGIN
2 t=0;
3 Initialize population P(t);
4 Initialize belief space B(t);
5 Repeat
6 Evaluate P(t);
7 Update(B(t), accept(P(t)))
8 Generate (P(t), influence(B(t)))
9 Select P(t) from P(t-1);
10 t+=1;
11 Until (termination condition)
12 END

In this algorithm, first the belief space and the population space are initialized. Then, the algorithm will repeat processing for each generation until a termination condition is achieved. Individuals are evaluated using the performance function. The two levels of Cultural Algorithm communicate through the acceptance function and the influence function. The acceptance function determines which individuals from the current population are selected to impact the belief space. The selected individuals' experiences are generalized and applied to adjust the current beliefs in the belief space via the update function. The new beliefs can then be used to guide and influence the evolutionary process for the next generation.

Cultural algorithms as described above consist of three components. First, there is a population component that contains the social population to be evolved and the mechanisms for its evaluation, reproduction, and modification. Second there is a belief space that represents the bias that has been acquired by the population during its problem-solving process. The third component is the communications protocol that is used to determine the interaction between the population and their beliefs.

1.2 DESIGN BASIC CULTURAL ALGORITHM
In the basic cultural algorithms, the belief space uses the \( \{S,N\} \) structure represented [4]. The formal syntax for the belief space, \( B \), used in this study is: \( B=S|N\{S,N\} \), where \( S \) denotes structure for situational knowledge and \( N \) denotes structures for normative knowledge. The definition above means the belief space can consist of situational knowledge only, normative knowledge only, or both. The situational knowledge \( S \) is represented formally as a pair wise structure: \( S = \left\{ E_1, E_2, ..., E_s, adjust_s(e) \right\} \), where \( E_i \) represent an \( i \)th best exemplar individual in the evolution history. There can be \( s \) best exemplars in \( S \) as a set that constitutes the situational knowledge. Each exemplar individual has \( n \) parameters and a performance value. \( adjust_s(e) \) is the belief space operator applied to
update number of exemplar individuals in $S$. The normative knowledge, $N$, a set of interval information for each of the $n$ parameters is defined formally as 4-tuple: $N = \{I_j, L_j, U_j, \lambda\}_{j=1}^{n}$, where $I_j$ denotes the closed interval of variable $j$, that is a continuous set of real numbers represented as an ordered number pair: $I_j = [l_j, u_j] = \{x \mid l_j \leq x \leq u_j, x \in \mathbb{R}\}$. $L_j$ (lower bound) and $U_j$ (upper bound) are initialized by the given domain values. $L_j$ represents the performance score of the lower bound $l_j$ for parameter $j$. $U_j$ represents the performance score of the upper bound $u_j$ for parameter $j$.

For the update function, we defined like this: $S = \{s_{\text{best}}\}$, select the best individual $s_{\text{best}}$ update the situation knowledge $S$ in belief space. The update process follows the equation (1):

$$s^{t+1} = \begin{cases} s_{\text{best}}^t & f(s_{\text{best}}^t) < f(s^t) \\ s^t & \text{otherwise} \end{cases}$$

where $s_{\text{best}}^t$ denotes $t$th best individual.

Update the normative knowledge $N$ in belief space uses the equation (2):

$$L_{\text{current}}^t = \begin{cases} l_j & \sigma(f(x_j) < L_j) \\ \text{others} & \end{cases}$$

$$L_{\text{current}}^t = \begin{cases} u_j & \sigma(f(x_j) < U_j) \\ \text{others} & \end{cases}$$

$$x_{\text{current}}^t = \begin{cases} x_{\text{current}}^t + \lambda \cdot \text{size}(I_j) \cdot N(0, 1) & x_{\text{current}}^t < L_j \\ x_{\text{current}}^t - \lambda \cdot \text{size}(I_j) \cdot N(0, 1) & x_{\text{current}}^t > U_j \\ x_{\text{current}}^t + \lambda \cdot \text{size}(I_j) \cdot N(0, 1) & \text{otherwise} \end{cases}$$

(2)

In basic CA, the knowledge represented in the belief space can be explicitly used to influence the creation of the offspring via an influence function. In our sliding window model, the strategy can be simply described as follows. The first is if a parent is in a promising region, the offspring are created by randomly changing the problem parameters of the parent just a little. In this case, the normative knowledge applies. The offspring $x_{j,i}^{t+1}$, will be created by using this normative knowledge as a beacon to attract the parent $x_{j,i}^t$ to move a copy toward the current sliding window, the influence function defined by the equation (3).

$$x_{j,i}^{t+1} = \begin{cases} x_{j,i}^t + \lambda \cdot \text{size}(I_j) \cdot N(0, 1) & x_{j,i}^t < L_j \\ x_{j,i}^t - \lambda \cdot \text{size}(I_j) \cdot N(0, 1) & x_{j,i}^t > U_j \\ x_{j,i}^t + \lambda \cdot \text{size}(I_j) \cdot N(0, 1) & \text{otherwise} \end{cases}$$

(3)

The second is if a parent is in an unpromising region, moving a copy of the parent to a more promising region can be used to create a new offspring. In this case, the constraint knowledge applies. The creation of offspring will be affected by the characteristic of the cells within the sliding window, the influence function defined by the equation (4).

$$x_{j,i}^{t+1} = \begin{cases} x_{j,i}^t + \lambda \cdot \text{size}(I_j) \cdot N(0, 1) & x_{j,i}^t < S_j \\ x_{j,i}^t - \lambda \cdot \text{size}(I_j) \cdot N(0, 1) & x_{j,i}^t > S_j \\ x_{j,i}^t + \lambda \cdot \text{size}(I_j) \cdot N(0, 1) & \text{otherwise} \end{cases}$$

(4)

Footnotes should be typed in single-line spacing at the bottom of the page and column where it is cited. Footnotes should be rare.
1.3 IMPLEMENTATION OF BASIC CA

Compared with genetic algorithm (GA), the most obvious advantage of CA is that the convergence speed is very high because of the dual inheritance system that characterizes evolution. In order to verify the convergence speed of the CA, we selected four benchmarks function and compared the results with traditional genetic algorithm.

F1: Schaffer function
\[
\min f(x) = 0.5 - \frac{(\sin^2\sqrt{x_i^2 + x_j^2} - 0.5)}{[1 + 0.001(x_i^2 + x_j^2)]^2}, -100 \leq x \leq 100
\]

In this function the biggest point is in the situation where \(x_i = (0, 0)\) and the global optimal value is 1.0, the largest in the overall points for the center, and 3.14 for the radius of a circle on the overall situation from numerous major points of the uplift. This function has a strong shock; therefore, it is difficult to find a general method of its global optimal solution.

F2: Shubert function
\[
\min f(x, y) = \left(\sum_{i=1}^{10} i \cos[(i+1)x + i]\right) \times \left(\sum_{j=1}^{10} j \cos[(j+1)y + j]\right),
\]
\(x, y \in [-10, 10]\)

This function has 760 local minimum and 18 global minimum, the global minimum value is -186.7309.

F3: Hansen function
\[
\min f(x, y) = \sum_{i=1}^{5} i \cos((i-1)x + i) \sum_{j=1}^{5} j \cos((j+1)y + j),
\]
\(x, y \in [-10, 10]\)
This function has a global minimum value -176.541793, in the following nine points: (-7.589893, -7.708314), (-7.589893, -1.425128), (-7.589893, 4.858057), (-1.306708, -7.708314), (-1.306708, -1.425128), (-1.306708, 4.858057), (4.976478, -7.708314), (4.976478, -1.425128), (4.976478, 4.858057) can get this global minimum value, the function has 760 local minimum.

F4: Camel function

\[
m_{\text{min}}(x, y) = \left(4 - 2.1x^2 + \frac{x^4}{3}\right)x^2 + xy + \left(-4 + 4y^2\right)y^2,
\]

\[x, y \in [-100, 100] \]

Camel function has 6 local minimum: (1.607105, 0.568651), (-1.607105, -0.568651), (1.703607, -0.796084), (-1.703607, 0.796084), (-0.0898, 0.7126), and (0.0898, -0.7126), the (-0.0898, 0.7126) and (0.0898, -0.7126) are the two global minimums, the value is -1.031628.

### Table 1: experiment results comparison (100 runs for each case)

<table>
<thead>
<tr>
<th>Function</th>
<th>Algorithm</th>
<th>Convergence Times</th>
<th>Optimal Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1</td>
<td>GA</td>
<td>72</td>
<td>1.000000</td>
</tr>
<tr>
<td></td>
<td>CA</td>
<td>75</td>
<td>1.000000</td>
</tr>
<tr>
<td>F2</td>
<td>GA</td>
<td>75</td>
<td>-186.730909</td>
</tr>
<tr>
<td></td>
<td>CA</td>
<td>80</td>
<td>-186.730909</td>
</tr>
<tr>
<td>F3</td>
<td>GA</td>
<td>85</td>
<td>-176.541793</td>
</tr>
<tr>
<td></td>
<td>CA</td>
<td>90</td>
<td>-176.541793</td>
</tr>
<tr>
<td>F4</td>
<td>GA</td>
<td>23</td>
<td>-1.031628</td>
</tr>
<tr>
<td></td>
<td>CA</td>
<td>56</td>
<td>-1.031628</td>
</tr>
</tbody>
</table>

In the experiment, each case is repeated for 100 times. Table 1 shows the statistics of our experimental results in terms of accuracy of the best solutions. GA found the known optimal solution to F1 72 times out of 100 runs, found the known optimal solution to F2 75 times out of 100 runs, found the known optimal solution to F3 85 times out of 100 runs, found the known optimal solution to F4 23 times out of 100 runs; CA is efficiency for the four cases: found the known optimal solution to F1 75 times out of 100 runs, found the known optimal solution to F2 80 times out of 100 runs, found the known optimal solution to F3 90 times out of 100 runs and found the known optimal solution to F4 56 times out of 100 runs.
2. IMAGE MATCHING BASED ON CULTURAL ALGORITHM

2.1. CODING
In this paper, we use natural number to code the chromosome. Find the location coordinates (i, j) of the match target is the goal of image matching, but the location coordinates must be satisfied 1≤i≤P-M+1, 1≤j≤Q-N+1, Where P, Q represents the number of rows and the number of columns of base image, and M, N, denote the number of rows and number of columns of the real time image. In here, the i and j are all natural number, so in our algorithm the coding we use natural number.

2.2. FITNESS FUNCTION
Individual’s fitness is a measure of individual ability to adapt to the environment in the evolutionary algorithm, is the sole criterion to distinguish the individual merits of the population, the algorithm can be the driving force of evolution and natural selection is the fundamental basis. Therefore, the design or selection of a good fitness function is crucial to improve the efficiency of the algorithm, in this paper, we choose the fitness function as follows:

\[
X = \sum_{P-M+1}^{P} \sum_{Q-N+1}^{Q} (T(u,v) * S(u+i, v+j))
\]

\[
Y = \sum_{P-M+1}^{P} \sum_{Q-N+1}^{Q} (S(u+i, v+j) * S(u+i, v+j))
\]

\[
Z = \sum_{P-M+1}^{P} \sum_{Q-N+1}^{Q} (T(u,v) * T(u,v))
\]

\[
\text{fitness}(i, j) = X \sqrt{Y} \sqrt{Z}
\]  \( (5) \)

In here, T (u, v) expressed the real time image’s gray matrix, S (u + i, v + j) is the individual gray-scale matrix corresponds to the location coordinates in the base image.

2.3. EXPERIMENT RESULTS
In this paper, we use a 512 × 512 image as the experiment object of the algorithm. We select a 35 × 35 as a sub-image as a template image (real time image) of the matching. The result shown in Fig.5 and Fig.6, Fig.7 is our algorithm’s final result, from the final result can be seen that the evolutionary image matching algorithm has the ability to fast optimization, and can converge to the global optimum.

Fig. 5: Base image
We compare our algorithm with traditional GA [1], in Table 3 is the comparison results.

<table>
<thead>
<tr>
<th></th>
<th>Experiment Number</th>
<th>Shortest Time(s)</th>
<th>Average Time(s)</th>
<th>Correct Rate(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GA</td>
<td>50</td>
<td>10.231</td>
<td>13.104</td>
<td>87</td>
</tr>
<tr>
<td>CA</td>
<td>50</td>
<td>9.736</td>
<td>11.534</td>
<td>95</td>
</tr>
</tbody>
</table>

Can be seen from the results in Table 1, we compare our proposed matching algorithm with the traditional GA with the same experiment number: 50. The shortest time: GA is 10.231s and CA is 9.736s; the average time: GA is 13.104s and CA is 11.534s, it is almost reduce 2s; then the correct rate: GA is 87% and CA is 95%, it is improve 8%. The above comparison results illustrate the proposed algorithm are doing a great job in the fast optimization ability of the algorithm and global convergence of these two aspects.

CONCLUSION

Cultural Algorithms are a class of computational models derived from observing the cultural evolution process in nature, compared with genetic algorithm the cultural algorithms have high convergence speed. In this paper, compared with genetic algorithm on the Benchmarks function, the results show that the new algorithm is efficient, we also use the new algorithm to solve image matching problem and the experiment results show the new algorithm is effective for this problem.

Acknowledgement

This paper is supported by the Provincial Natural Science Foundation of Hubei, China (No. 2013CFB320) and Science Research Foundation of Wuhan Institute of Technology (No.12126011).

REFERENCES

Wayne State University, Detroit, Michigan, USA, 1997.