Improved Methods for Fundamental Matrix Estimation Based on Evolutionary Agents

Mingxing Hu [1], Gordon Dodds[1], Baozong Yuan [2]

1. Virtual Engineering Centre, Queen's University Belfast, Belfast, BT9 5HN, United Kingdom;
2. Institute of Information Science, Beijing Jiaotong University, Beijing, P. R. China, 100044
Email: m.hu@qub.ac.uk

ABSTRACT

This paper presents two evolutionary agent-based approaches to fundamental matrix estimation. In order to improve the search ability and computational efficiency of the simple evolutionary agent, new methods, Competitive Evolutionary Agent (CEA) and Finite Multiple Evolutionary Agent (FMEA), are proposed by applying better evolutionary strategies and decision rules. CEA mainly focuses on the reproduction behavior and FMEA concentrates on the diffusion process. Experiments show that the improved approaches perform better than the original one in terms of accuracy and speed, and are more robust to noise and outliers.

1. INTRODUCTION

Estimation of the fundamental matrix that describes a relationship, across uncalibrated perspective cameras, is a fundamental problem in computer vision. It encapsulates the epipolar geometry, and can be used for motion segmentation, 3D reconstruction, camera calibration and view synthesis, etc.

Due to the great importance of fundamental matrix, its accurate and robust estimation has become an important and very productive research area [1-6]. Recently, with the development of evolutionary computation in the field of computer vision, some evolutionary algorithms have been implemented to provide the estimation, and have demonstrated excellent performance. Firstly, Chai addressed a simple genetic algorithm (sGA) for parameter estimation [4]. In sGA, each gene stands for a pair of correspondences and a chromosome with eight genes is considered as a sufficient subset to estimate the fundamental matrix. The authors also proposed a variable genetic algorithm (vGA), which employs different strategies for computing according to the length of chromosome, and the minimal subset for estimation consists of only seven genes [5]. GAs-based approaches can effectively explore a vast solution space by using properly defined genetic operators, and efficiently detect the outliers that are in gross disagreement with a specific postulated model. However, without exploiting the intrinsic parallelism, the calculation time for GAs is too long for many applications. This is a deficiency of nearly all GA-based applications.

The author also proposed an evolutionary agent-based approach, which improves the robustness of geometry estimation and reduces the computational expense as well [6]. The correspondences are viewed as a one-dimensional cellular-environment in which the agents inhabit, evolve and execute some evolutionary behaviors, such as reproduction and diffusion, to find the optimal result. Our approach is different from some other agent-based techniques, which mainly search in 2D digital images and use agents only to label feature pixels in the neighborhood [7-8]. But through the experiments it is found that because of the simplicity of evolutionary operators, when the parent agents breed more than one offspring, the agents of next generation will increase almost exponentially at early stages. Moreover, if there is no rule about termination, it will take a long time for convergence to occur.

In this paper, we present two improved approaches based on evolutionary agents for parameter estimation: (1) Competitive Evolutionary Agent (CEA). The differences between CEA and Simple Evolutionary Agent (SEA) are mainly in reproduction behavior. When a parent agent breeds a finite number of offspring agents, the offspring will compete with each other, and only the one with minimum cost will survive and proceed with diffusion operator. (2) Finite Multiple Evolutionary Agent (FMEA). The differences between FMEA and SEA lie in diffusion behavior. The offspring first compare their cost with their parents' to determine which of these will be kept in the environment. Then the successful agents are sorted by their cost, and the lowest cost n_{best} agents are kept active for further evolutionary processing. Experiments with both synthetic data and real images show that the improved methods work better than the original SEA and are appropriate for different situations. CEA produces results commensurate with, or superior to, SEA in accuracy, but the computation time decreases greatly (almost thirty percent). With higher computation efficiency, FMEA can obtain much better results than SEA in terms of accuracy and robustness.

The organization of the paper is as follows. In section 2, we give a brief introduction to epipolar geometry and agents. Then two improved approaches based on evolutionary agents are presented in detail. Section 4 deals with the experimental results obtained from synthetic data and real images. Finally, the conclusions are drawn in section 5.

2. BACKGROUND

Consider the case of two images acquired from a 3D scene. An image point $\mathbf{m}_i$ in the left view corresponds to an image point $\mathbf{m}'_i$ in the right. According to the well-known epipolar constraint, we have [1]

$$\mathbf{m}'^T_i \mathbf{F} \mathbf{m}_i = 0$$

where $\mathbf{F}$ is a 3×3 matrix called the fundamental matrix, $\mathbf{m}_i = (x_i,\ y_i,\ 1)^T$, $\mathbf{m}'_i = (x'_i,\ y'_i,\ 1)^T$ are the homogenous vectors for the corresponding points. $\mathbf{F}$ can be obtained by using only seven correspondences, which form the data matrix $\mathbf{Z} = (\mathbf{z}_1,\ \Lambda\ ,\ \mathbf{z}_7)$, and $\mathbf{z} = \left(x'_i x_i,\ x'_i y_i,\ x'_i,\ y'_i x_i,\ y'_i y_i,\ y'_i,\ x_i,\ y_i,\ 1\right)^T$, $i = 1,\ \Lambda\ ,\ 7$. The null space of the moment matrix $\mathbf{M} = \mathbf{Z}^T\mathbf{Z}$ is dimension 2, barring degeneracy ($\mathbf{Z}$ is 7×9). It defines a one-parameter family which is exactly fit for the seven correspondences: $\alpha\mathbf{F}_1 + (1-\alpha)\mathbf{F}_2$. Introducing the constraint $\det|\mathbf{F}| = 0$ leads to a cubic in α

$$\det\left|\alpha\mathbf{F}_1 + (1-\alpha)\mathbf{F}_2\right| = 0$$

which has 1 or 3 real solutions for α . Ideally, every possible subsample (seven correspondences) of all the correspondences n should be considered to get the optimal result, but this is usually computationally infeasible.

3. IMPROVED APPROACHES BASED ON EVOLUTIONARY AGENTS

In this section, we first give a common definition of evolutionary agents, then focus on the evolutionary behaviors of the improved approaches.

3.1 Common definition of evolutionary agent

Suppose that $\mathbf{S}$ is the dataset of all the correspondences $\{(\mathbf{m}_i, \mathbf{m}'_i) \mid i = 1, \Lambda, n\}$ and may be viewed as a one-dimensional grid lattice for agents to inhabit and evolve. The evolutionary agent is defined as follows

$$Agent = \langle \mathbf{p},\ a,\ D_{\cos t},\ fml,\ \mathrm{Rep},\ Diff,\ Die \rangle$$

which includes seven parameters to denote its structure and evolutionary behaviors. $\mathbf{p}$ stands for the positions of an agent in $\mathbf{S}$, which is a seven-dimensional vector, and the entry $p_k = \{i \mid i = 1, \Lambda, n\},\ k = 1, \Lambda, 7$, is the index number of correspondence lattice $\mathbf{S}$, as shown in Figure 1. a denotes the age of an agent; fml represents the family index, which indicates where an agent comes from. Rep denotes the reproduction behavior; $Diff$ represents the diffusion behavior; while Die indicates that an agent has a life span, and it may die like a living thing.

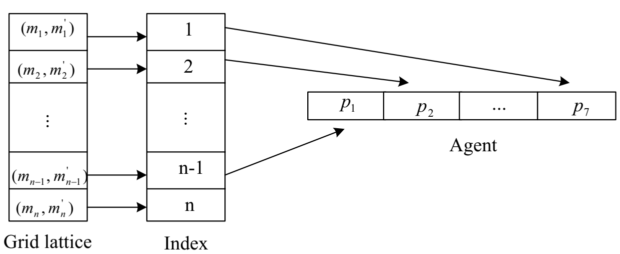

Figure 1. Agent representation for the 7 correspondences

$D_{\cos t}$ symbolizes the agent's cost, which indicates the adoptability of an agent, and can be computed by using the $\mathbf{F}$ matrix obtained from the correspondences of $\mathbf{p}$

$$D_{\cos tt} = \sum_{i=1}^{n} \omega_i d_i + \sum_{i=1}^{n} \beta(1 - \omega_i) \qquad (1)$$

where ω_i satisfies the following equation

$$\omega_i = \begin{cases} 1 & if\ \ d_i \le \beta \quad (inlier) \\ 0 & otherwise \quad (outlier) \end{cases}$$

d_i is the Sampson distance of correspondences i [9]. Sampson distance gives a first-order approximation to geometric error, the orthogonal distance of a point to the quadric variety determined by fundamental matrix.

$\beta = 1.96\delta = 1.96 \times 1.4828(1 + \frac{5}{n-7})\sqrt{med_i \mid d_i \mid}$ is the threshold for inliers and outliers, obtained from a maximum likelihood estimation. Readers are referred to the work [3] for more details.

In general, we mainly consider the Sampson distances of inliers and the outliers make a little contribution to the cost function.

3.2 Evolutionary behaviors of competitive evolutionary agent

Evolutionary agents adapt to their environment mainly by way of two behavioral responses, namely, reproduction and diffusion. We employ $A^{(g)}$ to stand for the set of all active agents in generation g .

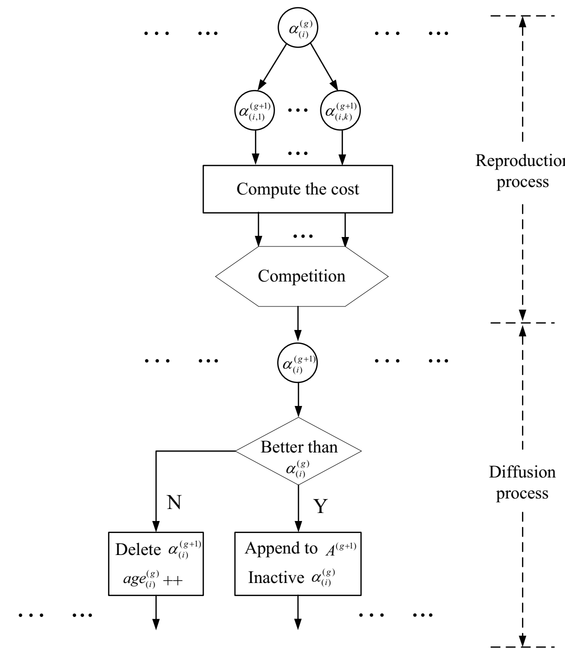

Figure 2. Evolutionary behaviors of CEA

(1) Reproduction: In the reproduction process, each active agent $\alpha^{(g)}$ (g denotes the generation of the evolutionary process) will breed a finite number, m , of offspring agents, $\alpha^{(g+1)}$. The differences between $\alpha^{(g)}$ and $\alpha^{(g+1)}$ are mainly in the position vectors, that is, two elements of $\mathbf{p}^{(g)}$ are selected and changed into the index numbers of $\mathbf{S}$ by a random number generator. Computation time will increase dramatically for a large m if no parallel algorithm is applied. So in CEA， the cost of the new generated agents are first computed by using equation (1), then the siblings of the same family, $\alpha_{(l)}^{(g+1)}$ (l denotes the family number), compete with each other. Only the agent with minimum cost will survive and move on to diffusion process. That is to say, there will be only one agent, the best one, kept by each family. Thus the total active agents of each generation will not increase, but better offspring have been created and selected by breeding. Thus this method of optimization, using two steps, reduces the complexity of the calculations.

(2) Diffusion: The diffusion behavior is important for an agent to search for new positions in correspondence lattice. After the reproduction process, the successful agents $\alpha^{(g+1)}$ will further compare their cost with that of their parent $\alpha^{(g)}$. If the offspring have the less cost, they will be appended to the agent set $A^{(g+1)}$,

and their parents will become inactive and be removed from the evolutionary environment. If not, the offspring will be deleted and the age of their parent will be increased by one.

3.3 Evolutionary behaviors of Finite multiple evolutionary agent

(1) Reproduction: The reproduction process of FMEA is simpler than that of CEA. Each active agent $\alpha^{(g)}$ breeds m offspring $\alpha^{(g+1)}$, and two entries of $\mathbf{p}^{(g)}$ are changed into randomly generated index numbers, just as SEA does.

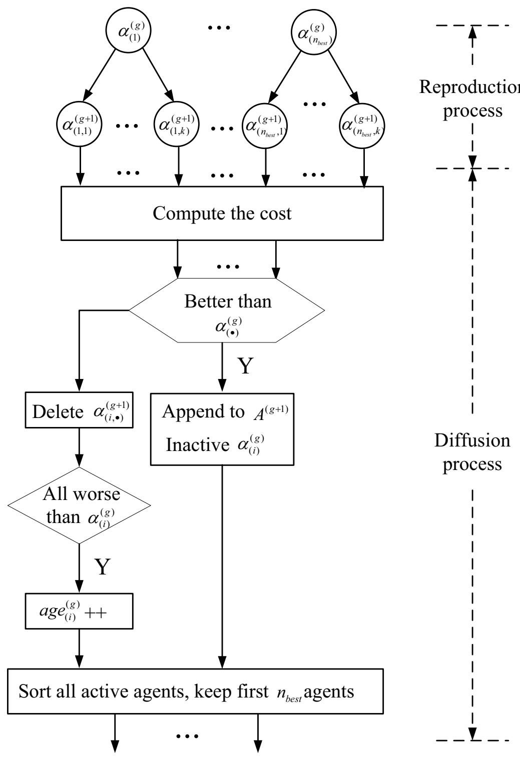

Figure 3. Evolutionary behaviors of FMEA

(2) Diffusion: First of all, the cost of the offspring is computed according to equation (1). Then the offspring agents of family l, $\alpha^{(g+1)}_{(l)}$, compare their cost with their parent's respectively. If the offspring wins, they will be added to the agent set, and their parents are removed. If none of the offspring of family l is better than their parent, they will be deleted and the age of their parents $\alpha^{(g)}_{(l)}$ will increase. That is to say, in FMEA, the offspring of family l maybe all survive from the comparison if with less cost. But here in contrast to CEA, after each generation there won't be more than one active agent of each family. Finally, the successful agents $\alpha^{(g+1)}$ are sorted ascendingly according to their cost, and the first n_{best} are kept for evolutionary process of next generation. Thus, we avoid the dramatic increase of active agents in the environment, especially at early stages. Each evolutionary setup will begin with the same number of active agents, and ends with the same number of offspring of better quality.

We would also emphasize that after each generation we will check the ages of active agents. If the age of an agent exceeds its life span, it will be removed from the environment, which avoids endless trial-and-error and thus reduces useless computation. If there is no active agent in the evolutionary environment, the whole process halts.

4. EXPERIMENTAL RESULTS

In this part, our improved approaches are compared with several typical methods, including LMedS [2], MAPSAC [3], and sGA [4]. In the comparison, part of the source code that is provided by X. Armangue [10] is used.

4.1 Experiments with synthetic data

In the experiments with synthetic data, the correspondences are randomly generated by space points in the region of $\Re^3$ visible to two different positions of a synthetic camera: $\mathbf{P}_1 = \mathbf{C}[\mathbf{I} \mid \mathbf{R}]$ ($\mathbf{C}$ stands for camera intrinsic matrix) and $\mathbf{P}_2 = \mathbf{C}[\mathbf{R} \mid \mathbf{t}]$, where the camera makes a rotation $\mathbf{R}$ and a translation $\mathbf{t}$. Here the total number of correspondences is 100, and there are only 10 agents in $\{\alpha^{(0)}\}$. The number of agents for initialization may be larger than 10, but it will take more time for computation and ten agents has been found in practice to be good enough for real applications. The experiments are divided into two groups:

(G1): Six different ranges of Gaussian noise are added to the projective correspondences, whose means are 0 and variances vary from 0.5 to 3.0 (in steps of 0.5), as shown in Table 1.

(G2): The means and variances of Gaussian noise are fixed to 0, 1, respectively; the percentage of outliers disturbed by the noise and false matches are varied from 10% to 50% (in steps of 10%), as shown in Table 2.

Table 1 Sampson distance under variable variance of noise

	LMedS	MAPSAC	sGA	SEA	CEA	FMEA
0.5	2.4893	0.3322	0.2825	0.2748	0.2689	0.2421
1.0	3.1307	0.8971	0.5532	0.5271	0.5139	0.4671
1.5	3.4586	1.5853	0.9426	0.9096	0.8821	0.8248
2.0	3.3881	1.9167	1.4093	1.3556	1.3179	1.2641
2.5	4.3218	2.3176	1.6802	1.6459	1.6196	1.5517
3.0	4.4311	2.7388	1.9850	1.9357	1.9033	1.8181

Table 2 Sampson distance under different percentage of outliers

	LMedS	MAPSAC	sGA	SEA	CEA	FMEA
10%	3.1401	1.2186	0.7225	0.6136	0.5642	0.5245
20%	3.7694	1.4756	0.9875	0.7855	0.7503	0.6969
30%	3.9844	1.9037	1.2141	1.0508	1.0146	0.9402
40%	4.0806	2.3307	1.3916	1.1879	1.1340	1.0507
50%	4.7508	2.7166	1.9770	1.7179	1.7020	1.5730

Table 3 Average computing times for two groups (Sec.)

Group	LMedS	MAPSAC	sGA	SEA	CEA	FMEA
G1	0.2748	0.4619	0.5594	0.3675	0.2433	0.3237
G2	0.2772	0.4766	0.5696	0.3209	0.2293	0.3118

Table 1 and 2 show the experimental results of (G1) and (G2), respectively. And Table 3 shows the average computation time spent by the methods in (G1) and (G2). From these tables, we can notice that EA-based approaches perform better than other typical methods, better even the simple genetic algorithm. And in the three EA-based methods, CEA turns out to be the quickest one, the results of which are slightly better than those of SEA, but the computation time is decreased by 33.8% in the noise-perturbation test, and 28.5% in the outlier-perturbation test, as shown in Table 1 and 2, respectively. Also it can be seen that, FMEA works as fast as SEA, but obtains the most accurate results. For instance, compared with SEA, the Sampson distance of FMEA decreases 11.4% and 6.08% when the variance of noise is 1.0 and 3.0 respectively, and 11.4% and 8.43% when there are 10% and 50% outliers involved respectively.

4.2 Experiments with medical images

Three different pairs of medical images are taken from a laparoscopic operation. Figure 4 illustrates the first pair of images we use and the white circles denote the feature points obtained by corner detection and matching.

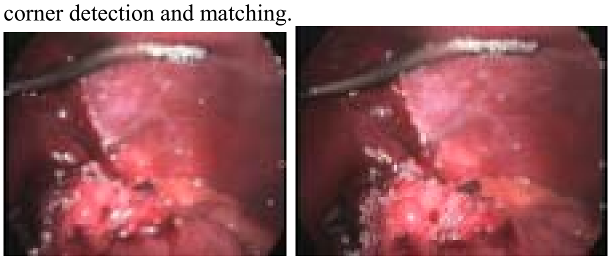

Figure 4 The medical images from two viewpoints

Table 4 Sampson distance of different pairs of medical images

Group	LMedS	MAPSAC	sGA	SEA	CEA	FMEA
MG1	3.5031	2.3201	1.6677	1.5240	1.4952	1.4546
MG2	3.7667	2.7987	1.8458	1.7727	1.7268	1.7003
MG3	3.0981	2.5948	1.6841	1.6773	1.6206	1.5749

Table 5 Average computation time for medical image testing (Sec.)

	LMedS	MAPSAC	sGA	SEA	CEA	FMEA
Time	0.6639	0.8487	1.0374	0.7755	0.6723	0.7905

Table 4 and 5 show the Sampson distances and computation time of the medical image testing, respectively. We can see that EA-based methods also perform best in the real image experiments. The mean Sampson distances of LMedS, MAPSAC, sGA, SEA and CEA are 2.192, 1.631, 1.099, 1.052, 1.024 times as much as that of FMEA. As to the computational efficiency, CEA works so fast that the computation time for CEA is 0.866, 0.851 as much as those of SEA and FMEA, respectively.

From the experiments above, we can conclude that:

(1) The competition between siblings in CEA effectively keeps the dramatic increase of new agents within limits, without decreasing the searching ability of evolutionary agent in the environment.

(2) The survival-of-finite-fittest in FMEA provides a richer population and more exploration to avoid unfavorable local minima than SEA and CEA, but with the same computation expense as SEA.

In other words, the evolutionary strategies of the improved methods help agents search for a fit parameter set in the uncertain solution space, and move more efficiently toward the global optimum by gradually reducing the chance of reproducing an unfit dataset.

5. CONCLUSION

In this paper, we describe two evolutionary agent-based approaches to fundamental matrix estimation, which employ new evolutionary strategies and decision rules. CEA mainly focuses on the reproduction behaviors to reduce the computing time, which produces results commensurate with, or superior to, SEA. FMEA focuses on the diffusion behaviors to obtain more accurate results, which is also appropriate in the various noise or outlier situations. The results of experiments in fundamental matrix estimation indicate that the improved methods attain high level of performance in terms of accuracy and computational efficiency. It can obtain an optimal (or near optimal) result in the solution space and is robust to the outliers disturbed by position noise and false matches.

6. ACKNOWLEDGEMENTS

Thanks to Royal Belfast Hospital for provding the laparoscopic images. The research is partly supported by VEC, National Natural Science Foundations (No. 69789301) and Doctoral Foundations of China (No. 97000409).

7. REFERENCES

[1] H.C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections", Nature, vol. 293, no. 10, pp. 133-135, 1981.

[2] Z. Zhang, "Determining the epipolar geometry and its uncertainty: a review", International Journal of Computer Vision, vol. 27, no. 2, pp.161-198, 1998.

[3] P.H.S. Torr, "IMPSAC: Synthesis of importance sampling and random sample consensus", IEEE Trans. On Pattern Analysis and Machine Intelligence, vol. 25, no. 3, pp. 354-364, 2003.

[4] J.X. Chai and S.D. Ma, "Robust epipolar geometry estimation using genetic algorithm", Pattern Recognition Letters, Vol. 19, pp. 829-838, 1998.

[5] M.X Hu, Q. Xing, B.Z. Yuan and X.F. Tang, "Epipolar geometry estimation based on genetic algorithm under different strategies", International Conference on Signal Processing, vol. 1, pp. 885-888, 2002

[6] M.X. Hu, G. Dodds, B.Z. Yuan, "Evolutionary agents for epipolar geometry estimation", International Conference on Image Processing, vol.2, 1843-1846, 2004.

[7] J.M Liu, Y.Y. Tang and Y.C. Cao, "An evolutionary autonomous agents approach to image feature extraction", IEEE Trans. On Evolutionary Computation, vol. 1, no. 2, pp. 141-158, 1997.

[8] C.J. Veenman, M.J.T. Reinders and E. Backer, "a cellular coevolutionary algorithm for image segmentation", IEEE Trans. Image Processing, vol. 12, no. 3, pp. 304-316, 2003.

[9] P.D. Sampson, "Fitting conic sections to very scatered data: a iterative refinement of Bookstein algorithm", Computer Graphics and Image Processing, vol. 18, pp. 97-108, 1982.

[10] Matlab Toolkit: Http://eia.udg.es/~armangue/research