2008－MPS－72 (9)
2008－BIO－15 (9)
2008／12／17

Linear GP with Redundancy-removed Recombination for Synthesis of Image Feature Extraction Programs

Ukrit Watchareeruetai, Yoshinori Takeuchi, Tetsuya Matsumoto, Hiroaki Kudo and Noboru Ohnishi

Department of Media Science, Graduate School of Information Science, Nagoya University
Furo-cho, Chikusa-ku, Nagoya 464-8603 Japan
ukrit@ohnishi.m.is.nagoya-u.ac.jp, {takeuchi,matumoto,kudo,ohnishi}@is.nagoya-u.ac.jp

We describe an evolutionary synthesis of feature extraction program for object recognition. The evolutionary synthesis method is based on linear genetic programming with the use of redundancy-removed recombination. The evolutionary synthesis can automatically construct feature extraction programs for a given object recognition problem without any domain-specific knowledge. An experiment was done on a lawn weed detection problem. The result shows that performances of the synthesized feature extraction program are comparable with those of the conventional lawn weed detection methods.

1. Introduction

To design a feature extraction program for a given object recognition problem is not an easy task and time-consuming. Usually, it is done by human experts who have to consider what features are appropriate for the problem at hand and how to extract such features from images. Human experts generally do this task based upon their knowledge and experience, and sometime under time-limitation. Therefore un-conventional but potential features may be ignored.

Many researches attempted to cope with the difficulties in designing not only feature extraction programs but also image classification and image transformation programs by using various evolutionary algorithms such as genetic algorithm (GA) [1], tree-based genetic programming (GP) [2,3,4], graph-based GP [5,6], and linear GP [7]. In this work, we focus on the approach based on linear GP because its representation is simple but powerful enough to represent graph-structure programs. Moreover, there is an algorithm for removing introns, i.e., the operations that have no effect on program output, in run-time [8]. Therefore we can avoid wasteful execution of such operations.

We have developed an automatic system for synthesis of feature extraction programs based on linear GP and proposed a way to improve its performance by a recombination strategy, named redundancy-removed recombination [9]. The objective of this work is to evaluate performances of the feature extraction programs synthesized by this system compared with feature extraction (and entire image recognition) methods proposed in related works. We have conducted an experiment with a lawn weed detection problem [10,11].

The rest of this paper is organized as follows. Section 2 describes the linear GP based synthesis of feature extraction program. Section 3 explains the redundancy-removed recombination. Section 4 is about target problem and compared methods. Section 5 shows experimental results. Section 6 concludes the paper.

2. Linear GP Based Synthesis of Feature Extraction Programs

In this work, a feature extraction program is represented by using a fixed-length sequence of basic image processing operations, e.g., thresholding, filtering, edge detection, histogram equalization, which are used as primitive operators (POs). In the execution process, each operation is sequentially executed based on a set of shared register like program execution in modern microprocessors. Namely, an operation fetches inputs from registers, processes them, and stores the processed result into a register. Two types of registers are used; image and numerical registers.

Each operation is encoded by using four components, i.e., one operation code (op-code) which describes PO to be executed and three arguments which describe input, output register indexes and parameter of the operation. One linear program consists of multiple sub-programs. Each one is executed independently of each other and generates one feature image. Intensity values of a pixel (x, y) in all feature images are integrated to construct a feature vector for that pixel. A classifier is adopted to decide to which class that pixel belongs, based upon the feature vector. In this work, Bayesian classifier with histogram approximation method is adopted.

3. Redundancy-removed Recombination

In [9], we have described about redundancies in linear GP representations and proposed a way to transform the representations into canonical forms in which redundancies are removed. Once we know the canonical forms of two individuals, we can easily verify whether or not they represent the identical program. It can be done by comparing the lengths of the two canonical forms. If they are the same, we compare their (genotypic) content byte-by-byte. The two individuals that represent the identical program will have exactly the same canonical form.

The redundancy-removed recombination [9] exploits the canonical form to verify whether the generated offspring represent programs already discovered in the evolutionary search. A generated offspring that represents a program discovered before will not be allowed to survive in the next generation. We call such offspring a redundant offspring. For the case of redundancy-removed crossover, generated redundant offspring is ignored. We then re-select parents (randomly) and generate new offspring instead. If the new offspring is still redundant, this process will be repeated. If maximum number of repeats is reached and no non-redundant offspring is generated, the latest offspring is allowed to survive in the next generation. For the case of redundancy-removed mutation, we apply mutation operation on the same offspring until it becomes the non-redundant offspring. By using the redundancy-removed recombination, we can significantly improve the search performance of linear GP based synthesis of feature extraction programs [9].

4. Lawn Weed Detection Problem

Image processing based weed detection methods have been widely studied, especially for agriculture fields. Automatic weed control systems exploit weed detection method to locate the area of weeds in the field so that the systems can precisely get rid of weeds from the field, e.g. spraying herbicide onto the area of detected weed only instead of spraying in the entire area.

We are interested on weed detection in lawn fields. Up to now, various methods have been proposed. Ahmad et al. [12] proposed a weed detection method based on gray-scale uniformity analysis (denoted by UA) which distinguishes weed surfaces from lawn surfaces based on the difference in gray-value distributions of both surfaces. As a post-processing process, a blob inspection method is exploited after weed detection by uniformity analysis to remove misclassified blobs. In [10], we have proposed two lawn weed detection methods. One is Bayesian classifier based method (denoted by BC) which detect weeds based on two textural features, i.e., mean and variance of edge strength. The other one is morphological operation based method (denoted by MO) which segments weed from lawn background by using morphological image processing techniques such as closing and opening. Moreover, in [11], the BC method is slightly modified; support vector machine is adopted instead of Bayesian classifier and we found that it is better than the BC method in some situations. We denote this modified method by SVM. All these methods (UA, BC, MO, and SVM) were compared with the feature extraction program synthesized by linear GP.

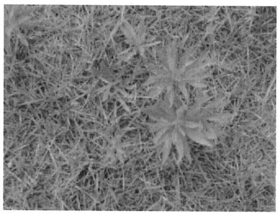

Figure 1 A lawn weed image

Table 1 Parameter setting

Population size	100
Maximum generation	500
Number of features	2
Length of sub-program	10
Crossover rate	0.75
Mutation rate	0.25
Number of image registers	6
Number of numerical registers	6
Number of POs	51

Table 2 Segmentation accuracies (%) of the synthesized programs

Trial no.	Training set (five images)	Validation set (25 images)
1	97.49%	98.50%
2	97.72%	98.64%

5. Experimental Results

Table 1 describes the parameters of the linear GP with redundancy-removed recombination which is used to synthesize feature extraction program. The lawn weed database used in this experiment is the database 1 used in [10]. It consists of training and validation sets. The training set contains five images, whereas the validation set contains 25 images. Image size is 640 × 480 pixels, covering the lawn area of size 274 × 205 mm. Figure 1 shows an example of a lawn image that includes some weeds.

Table 2 shows segmentation accuracy, which is the ratio of the number of correctly segmented pixels to the total number of pixels, of the synthesized programs. Only two independent GP runs were conducted due to the high computational complexity of the system. The synthesized programs can segment the area of weeds from lawn with accuracies of around 98%.

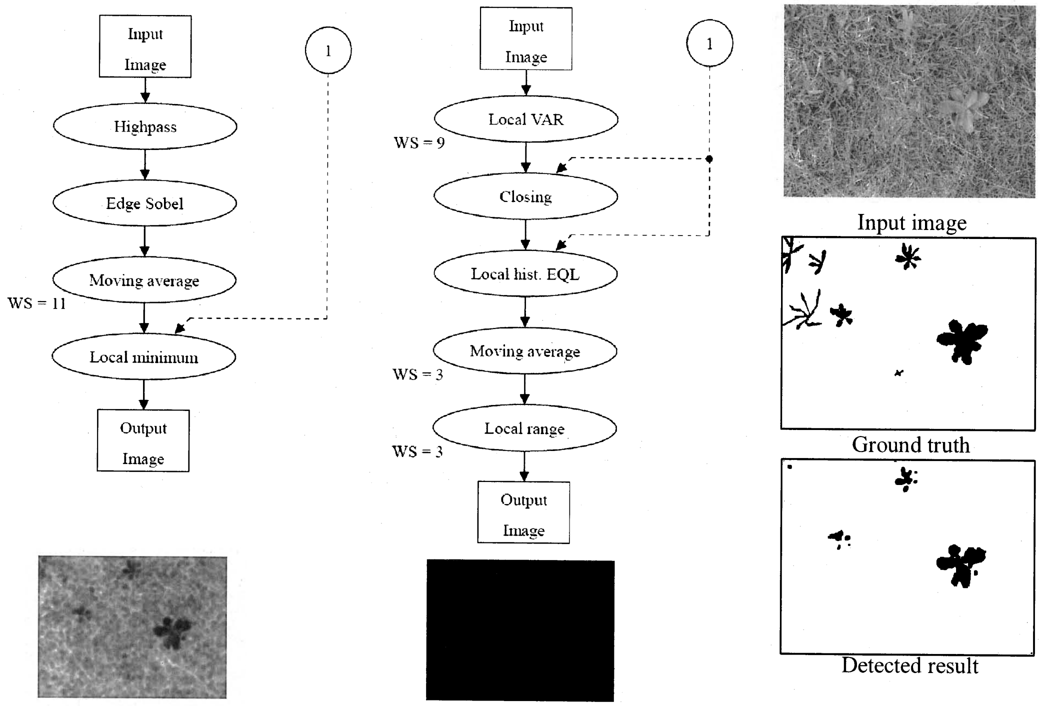

Figure 2 Flowchart of the synthesized feature extraction program (of the trial no.2)

Table 3 Performance comparison of lawn weed detection methods on a chemical-based weed control system

Rank	Method	Dataset1 (the total number of weeds = 58, the number of weed blocks = 1533)			
		Killed weed rate (# of killed weeds)	Correct spray rate (# of correct sprayed blocks / # of sprayed blocks)	False spray rate (# of false sprayed blocks / # of sprayed blocks)	Herbicide reduction rate
1	SVM	87.93% (51)	98.40% (1172/1191)	1.60% (19/1191)	92.55%
2	BC	87.93% (51)	95.94% (1252/1305)	4.06% (53/1305)	91.84%
3	UA	86.21% (50)	95.52% (1321/1383)	4.48% (62/1383)	91.35%
4	**Synthesis**	**86.21% (50)**	**93.87% (1347/1435)**	**6.13% (88/1435)**	**91.03%**
5	MO	81.03% (47)	91.59% (1383/1510)	8.41% (127/1510)	90.56%

Table 4 Performance comparison of lawn weed detection methods on an electrical-based weed control system

Rank	Method	Dataset1 (total weed number = 58)		
		Killed weed rate (# of killed weeds)	Correct spark rate (# of correct sparks / # of spark)	False spark rate (# of false sparks / # of spark)
1	BC	79.31% (46)	94.90% (93/98)	5.10% (5/98)
2	**Synthesis**	**77.59% (45)**	**93.83% (76/81)**	**6.17% (5/81)**
3	UA	75.86% (44)	95.60% (87/91)	4.40% (4/91)
4	SVM	75.86% (44)	96.77% (120/124)	3.23% (4/124)
5	MO	58.62% (34)	91.53% (54/59)	8.47% (5/59)

In Fig. 2, the flowcharts of feature extraction programs synthesized by the trial no. 2 are described. The left flowchart consists of four POs, and generates a feature image in which the difference between weed and lawn area is enhanced (the weed area become darker). The right flowchart consists of five POs but its output does not relate with the input image; it is always a black image. This means that only one feature is really effective. However, based on these features, the classifier can accurately distinguish the areas of weeds from lawn in a certain level (see the detected result in Fig. 2).

We compared the synthesized program of the trial no. 2 (as shown in Fig. 2) with the lawn weed detection methods in the literatures, i.e., the BC, MO, SVM, and UA methods. Two types of simulated weed control systems are considered. One is a chemical-based system which destroys weeds by spraying herbicide, and the other is an electrical-based system which destroys weeds by applying a high-voltage spark discharge onto the weeds [10].

The performance of weed control the chemical-based and electrical-based systems are shown in Tables 3 and 4, respectively. The main values we compared are the killed weed rate, which shows weed destruction performance, and # of false sprayed block (or # of false sparks) which describes destruction error. Ranking was done based on these values. The ranking criterion we used is the same as in our previous work [11], i.e., the method that gives errors lower than an acceptable value and gives higher weed destruction performance will have better rank. The acceptable values of the chemical-based and electrical-based systems are 160 false sprayed blocks and five false sparks, respectively. Parameters of the compared methods were manually adjusted to obtain the best result for each type of weed control system.

In the case of the chemical-based system, the synthesized program is in the 4th rank among all five methods. Its weed destruction performance is slightly lower than that of the best one (SVM) and the error is more than that of the SVM by around five points (false spray rate). In the case of the electrical-based system, the synthesized program is in the 2nd rank. Its weed destruction performance is slightly lower than the best one (BC) but it gives the same number of false sparking.

The results show that the feature extraction program, which was automatically generated by the linear GP based system without any domain-specific knowledge, is comparable with the lawn weed detection designed by human experts.

6. Conclusion

We have described an evolutionary synthesis of feature extraction program based on linear GP with redundancy-removed recombination [9]. Lawn weed detection problem was considered here. We compared performance of the feature extraction program synthesized by the system with the detection methods proposed in literatures. Experimental results show that the performance of synthesized program is comparable with those of the compared methods. In future, we plan to improve the performance of the evolutionary synthesis of feature extraction programs by using multi-objective optimization technique.

Acknowledgement

The authors would like to thank the Hori Information Science Promotion Foundation for a research grant.

References

[1] T. Nagao and S. Masunaga, "Automatic construction of image transformation processing using genetic algorithm," Proc. ICIP-96, vol.3, Lausanne, Switzerland, pp.731-734 (1996)

[2] S. Aoki and T. Nagao, "Automatic construction of tree-structural image transformations using genetic programming," Proc. ICAIP-99, Venezia, Italy, pp.136-141 (1999)

[3] M. Zhang, V. Ciesielski, P. Andreae, "A domain-independent window approach to multiclass object detection using genetic programming," EURASIP J. on Applied Signal Processing, vol.8, pp.841-859 (2003)

[4] M. Roberts and E. Claridge, "A multi-stage approach to cooperatively coevolving feature construction and object recognition," In F. Rothlauf et al. (eds.): Application of Evolutionary Computing, LNCS 3449, Springer-Verlag, pp.396-406 (2005)

[5] A. Teller and M. Veloso, "PADO: a new learning architecture for object recognition," In K. Ikeuchi and M. Veloso (eds.): Symbolic Visual Learning, Oxford University Press, pp.77-112 (1997)

[6] S. Shirakawa and T. Nagao, "Genetic image network (GIN): automatically construction of image processing," Proc. IWAIT-2007, Bangkok, Thailand, pp.643-648 (2007)

[7] K. Krawiec and B. Bhanu, "Visual learning by evolutionary and coevolutionary feature synthesis," IEEE Trans. Evolutionary Computation, vol.11. no.5, pp.635-650 (2007)

[8] M. Brameir and W. Banzhaf, "Linear Genetic Programming," Springer (2007)

[9] U. Watchareeruetai, Y. Takeuchi, T. Matsumoto, H. Kudo and N. Ohnishi, "Improving search performance of linear genetic programming based image recognition program synthesis by redundancy-removed recombination," Proc. of SMCia/08, Muroran, Japan, pp.393-398 (2008)

[10] U. Watchareeruetai, Y. Takeuchi, T. Matsumoto, H. Kudo and N. Ohnishi, "Computer vision based methods for detecting weeds in lawns," Machine Vision and Applicaitons, vol.17, no.5, pp.287-296 (2006)

[11] U. Watchareeruetai, "Lawn weed detection methods using image techniques for automatic weed control systems," Master Thesis, Graduate School of Information Science, Nagoya University (2007)

[12] U. Ahmad, N. Kondo, M. Monta, S. Arima and K. Mohri, "Weed detection in lawn field based on gray-scale uniformity," J. of the Japanese Society of Environmental Control in Biology, vol.36, no.4, pp.227-237 (1998)