Automatic Generation of Image-Processing Programs for Production Lines

● Tsuyoshi Nagato ● Tetsuo Koezuka

Image-processing systems are applied at many stages of production lines, such as manufacturing/assembly and visual inspection. Because changes take place frequently in production lines, as in facility or component alterations, image-processing programs need to be developed and/or modified promptly to accommodate those changes. Using one of the machine learning techniques—Genetic Programming—Fujitsu Laboratories is pursuing the development of technology to automatically create image-processing programs simply by feeding in input images and the correct data that is the intended processing result. One of the biggest challenges in applying genetic programming to production lines was the learning speed. In tackling this challenge, we attempted to restrict the hierarchical depth of the program structure, and developed an appropriate evaluation method according to the shapes of target objects and an auto-selection system for the necessary learning data. As a result, we succeeded in generating an automatic program for high-precision image processing applied to alignment technology in component assembling equipment while managing to reduce the time required for the learning process to one-tenth that of conventional systems. In this paper, we explain this technology to automatically generate an image-processing program, and describe the evaluation of its application to a production site.

1. Introduction

In the factory automation field, image-processing technology using cameras has been introduced for various applications, such as product manufacturing/assembly and visual inspection. In recent years, automation through the use of robots has been progressing, particularly for post-processes, and there has been a rising need to meet requirements in terms of quality improvement and flexible high-mix low-volume production. To this end, improvements of the performance and versatility of image-processing technology are strongly desired.

The process of development and modification of image-processing programs for manufacturing equipment is shown in **Figure 1**. During the start-up of manufacturing equipment, image-processing program developers create a program according to its intended purpose(s) while checking the images captured by the camera(s). If high-accuracy measurement of the distance between component A and component B of a given product in a captured image is desired, they create an image-processing program that detects the straight edges of each component. As the developed manufacturing equipment goes on producing products, the lighting conditions might change due to adjustments made to the equipment or simply the passage of time, causing a change in brightness of the captured images. Because the detection performance of the image-processing program drops as a result, it becomes necessary to adjust the various parameters of the program. Moreover, besides changes in the equipment environment, the specifications of the products themselves change also. Therefore, simply adjusting the parameters might not be enough, and it may be necessary to rebuild the program.

When unexpected external factors such as described above arise, developers go to the production site, and while they are modifying the program, all too often production efficiency drops or the production line stops altogether. Were it possible to automatically modify programs at the production site, situations requiring program adjustment could be handled quickly at any time. Therefore, for the purpose of quick market launches and the stabilization of mass production quality, technology that allows automatic program creation and modification at production sites without any professional knowledge is desired.

2. Technology for automatic generation of image-processing programs

As a technology for the automatic generation of image-processing programs, automatic construction of tree-structural image transformation (ACTIT),[1),2)] which applies genetic programming (GP),[3)] a machine learning technique, to image programming, has been proposed. This technique automatically generates tree structure programs for the intended purpose by treating image-processing programs as tree structure programs configured of multiple basic image-processing functions and optimizing combinations of such functions through GP on a computer.

The more varied the types and the greater the number of image-processing functions, which are the basic constituents, the better it is for automatically generating programs for diverse purposes, but this results in the buildup of an enormous number of combinations of tree structure programs at great computational cost (in terms of time). Using existing technology, it takes about 3 or 4 days to create the desired image-processing program, which, considering application to production sites, inevitably causes a drop in production efficiency during the program creation period. Given that quick improvement of production efficiency is desired, making automatic program creation technology work faster is crucial.

Further, methods to evaluate whether the programs created through automatic program creation with GP are suitable for the intended use also need to be developed. Since existing technology is designed to convert input images into different images, image quality based program evaluation methods are used. However, such evaluation methods cannot be applied for purposes other than image conversion, such as shape recognition and judgment processing. Moreover, program evaluation requires multiple input images as learning data. These input images should ideally constitute a body of images that depict a variety of content under many different lighting conditions. For that reason, evaluation methods designed for the intended purpose as well as methods allowing efficient selection of learning data are needed.

This paper describes a technique for the automatic creation of image-processing programs that was developed for application to production lines and meets all the above requirements.

3. Developed technology

Figure 2 shows an outline of the newly developed technology for the automatic creation of image-processing programs through GP. In this technology, each

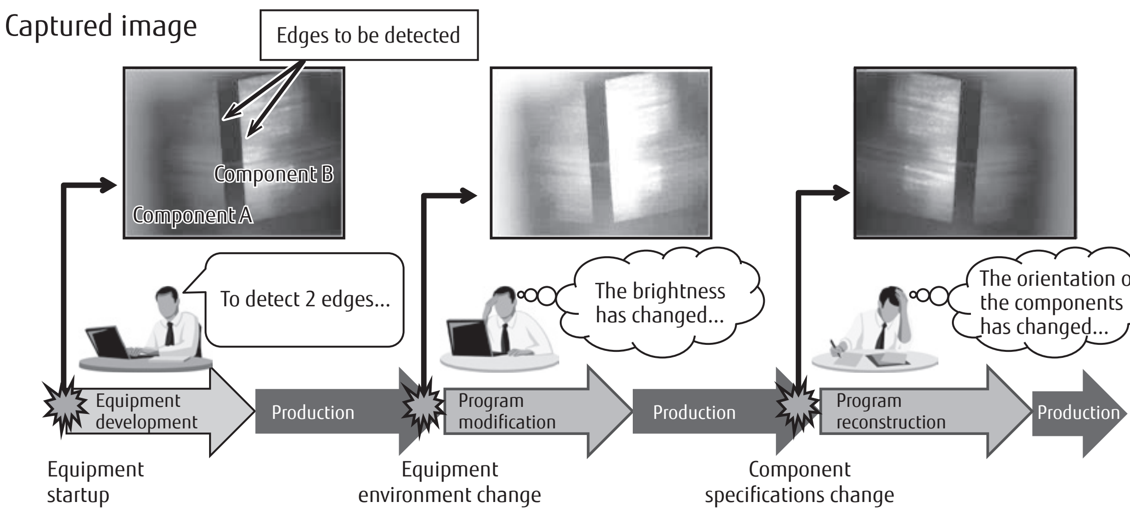


Figure 1
Image-processing program development and modification process.

image-processing program is considered to consist of a combination of basic image-processing functions, and to have a tree structure that yields one output result per input image. To create a program, it is necessary to prepare in advance, as learning data, input images for learning, and the target images that one wants the program to automatically generate as the correct output result. The flow of tree structure optimization by GP is explained below. First, multiple programs built at random (initial individuals) are generated to form the parent program. Then, two of the parent programs are selected, crossover operation that interchanges parts of the programs' structures, and mutation operation that causes local structure alterations, are performed to generate new programs as children (evolution). The outputs results of the generated programs and the target images are compared, and the fitness score used to evaluate the performance of the programs is calculated. Then, depending on their fitness, the child programs are used as next-generation parents (generation change). The aim is to achieve program structures of high fitness that output processing results close to the target images by repeating this process of evolution and generation change.

This section describes the newly developed program generation method, evaluation method, and learning data selection method for application to production lines.

1) Method of program generation by hierarchization

The number of image-processing functions, which are the basic constituents for combinations, and the size of the tree structure are both important for the generation of image-processing programs for various purposes. However, as these increase, so does the number of combinations explosively increase, and as a result, the learning time required until the automatic generation of programs of suitable fitness for the intended application grows longer. Considering this, shortening the learning time by specifying the general flow of image processing for the intended application in advance, and setting a limit on the number of combinations of the tree structure, was sought.

Figure 3 shows the program structure hierarchized in line with the general processing flow and a sample list of image-processing functions at each level. In general image recognition, preprocessing is performed through a flow that consists in enhancing the features of the recognition target, and after these features have been extracted through threshold processing, by forming their shape through binary image processing. Image-processing functions, which are the program constituents, also can be divided into three

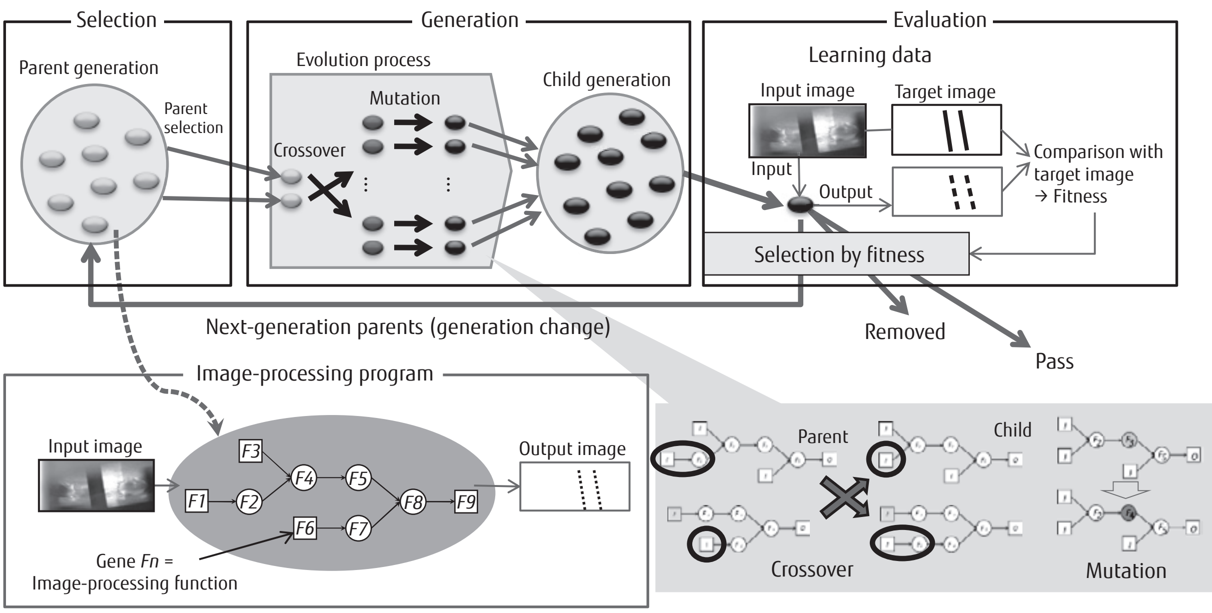
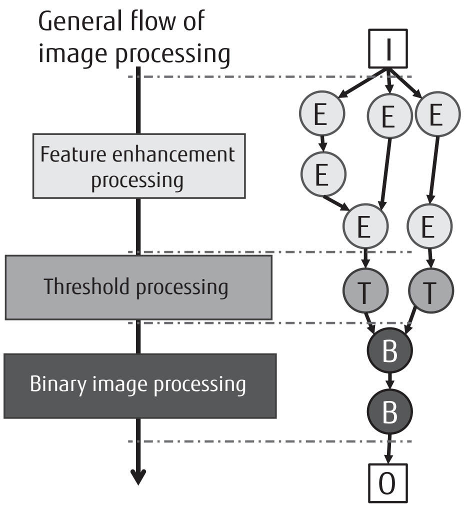


Figure 2
Automatic program generation by genetic programming.



Sample list of image-processing functions

E	T	B
Smooth (smoothing)	Thresh (fixed binarization)	Erode (erosion)
Sobel (first derivation)	Auto (automatic binarization)	Dilate (dilation)
Laplace (second derivation)	Edge (edge detection)	Mask (mask processing)
Inverse (grayscale inversion)		And (AND operation)
Sub (subtraction)		Or (OR operation)
Add (addition)		Not (inverse operation)
Diff (absolute value of difference)		

Figure 3
Hierarchization of processing procedure.

main types, i.e. feature enhancement processing, threshold processing, and binary image processing. Thus acceleration of learning by narrowing down selection candidates by setting constraints in such a way that the above-described flow is not disrupted, when carrying out crossovers and mutations during the process of parent program generation and evolution in the early learning phase, was aimed for.

2) Program evaluation method according to shape

If the program to be automatically generated is to perform shape detection processing for shapes such as straight lines and circles, the ultimate goals is not to obtain images consisting of extracted shapes, but detection of the position and inclination of the shapes. The evaluation of programs for detecting straight lines is explained below. The conventional evaluation method compares the target image and output image pixel by pixel. If this method is applied for straight line detection, images obtained by extracting the straight line parts might be prepared as target images. In that case, there is the problem that the evaluation criteria of the program change depending on the thickness or length of the straight lines.

To solve this problem, a technique to perform evaluation not through pixel-by-pixel comparison but instead using parameters such as the position and inclination of the extracted straight lines was developed.[4] This allows evaluation of program performance by using the final output result according to the purpose as the evaluation index.

3) Learning data selection method

The proposed technology evaluates program performance by using several sets of learning data. For that reason, automatically generated programs often are ineffective for images with features that differ from the learning data. Further, in the case of automatic generation using a large amount of learning data, the learning period grows greatly, which is problematic from an operation viewpoint. Therefore, to automatically and quickly generate programs that have good performance, learning needs to take place after the suitable selection of learning data that grasps the features of the images acquired from the manufacturing equipment.

The flow of the automatic selection of learning data under the proposed technology is described below. First, the respective image feature amounts are calculated from several hundred captured images taken from the manufacturing equipment. Given that most of the image-processing functions that are program constituents are spatial filters, and that the overall image features are brightness and texture,[5] the average luminance, luminance histogram, spatial frequency component and the like are selected for use for the image feature amount. Next, normalization and clustering are applied to each feature amount. The k-means method is used as the clustering method here. As the result of clustering, the center sample of the feature space is selected as learning data. The distribution of each captured image in the feature space and an example of the image selected for each class are shown in **Figure 4**. Captured images are broadly divided into two groups, and in the group with both a strong spatial frequency component and average luminance, each of the samples can be seen to consist of images that have an extremely bright left border and depict the capture target as having a characteristic pattern. As a result, learning data that grasps the target features can be selected.

Use of the above three technologies allows automatic generation of high-performance programs with a learning time of just a few hours.

4. Evaluation of application of the developed technology to production lines

To verify the validity of this technology, the proposed automatic program generation technology was applied to the image recognition unit of component assembly equipment, and its performance was evaluated. The purpose of the image recognition was to measure the distance between component A and component B in the capture image in Figure 1, and it was decided to automatically generate a program for detecting the straight edge on the component A side. The learning data was to consist of the images captured on the same equipment and the edge position information on these images, and 8 sets of images out of the sample data consisting of about 300 images were selected by the above-described automatic selection technique for learning data. For the edge position information, each image was displayed on a screen and the coordinates of two points on each edge were taught. The program hierarchy limit and image-processing functions serving as constituents shown in Figure 3 were selected for use, and for evaluation of the program, three parameters, i.e. the position and inclination of the detected straight lines, and the gradient direction of the edge, were selected to calculate the fitness of the program.

The time required for automatic generation was found to be approximately 8 hours, a reduction by a factor of 10 compared with conventional methods. **Figure 5** shows the structure of the program generated using the proposed technology and the recognition results obtained from the captured images. As seen from the output result, the generated program extracted the edge on the component A side in white, and the recognition result correctly shows the edge. Further, verification of the component A edge detection performance by applying the automatically generated program to the 300 evaluation image samples that were not used for learning showed that, in all samples, straight lines were recognized with an accuracy level meeting specification requirements. Notwithstanding

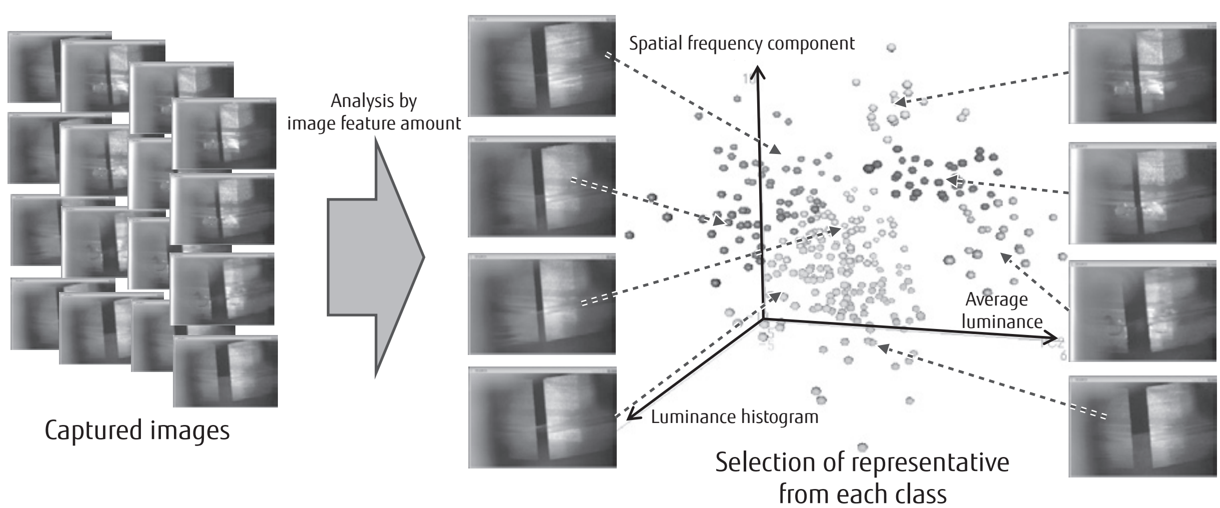


Figure 4
Automatic selection by image feature amount.

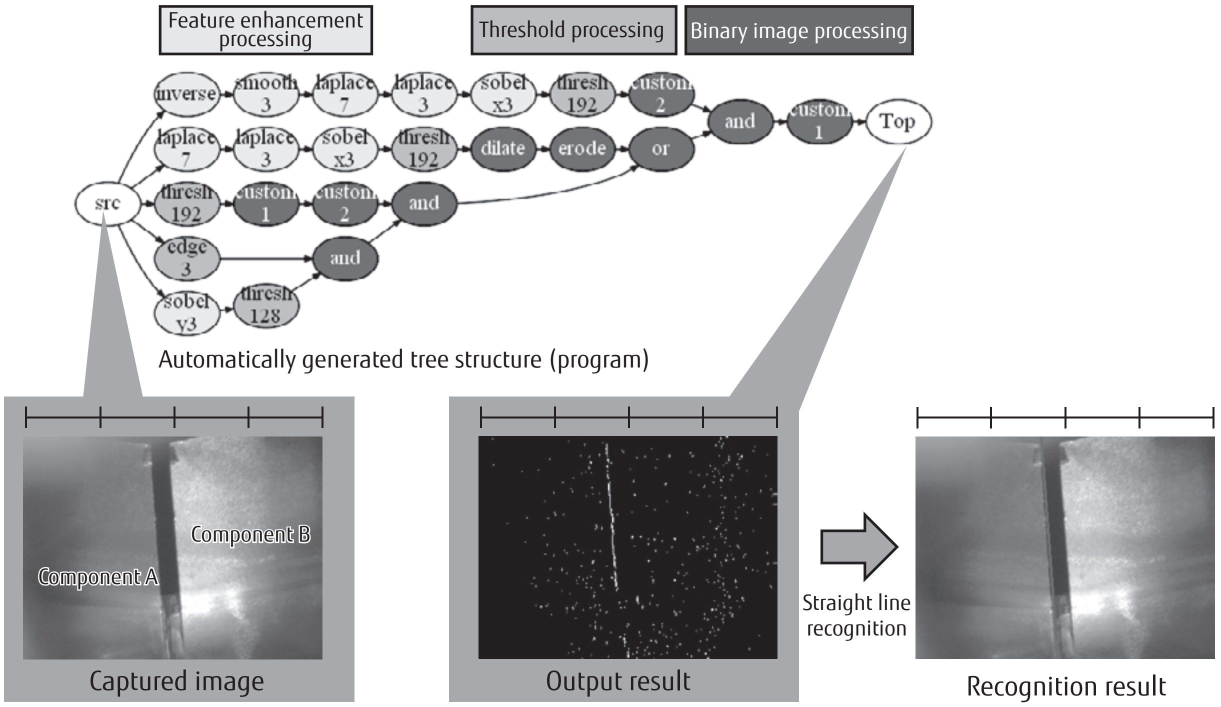


Figure 5
Automatic program generation result.

observable contamination of components, chipped edges, and differences in brightness in the evaluation images, performance was not affected and correct edge detection was performed. Based on the above, it was confirmed that high-performance programs can be generated quickly by this technology.

Further, a trial evaluation was conducted on an actual production line by using the present technology for 5 positioning tasks performed by component assembling equipment. This evaluation showed that learning at start-up and relearning when modifications occur can be done in 8 hours or less. Thus, equipment adjustment and performance verification work that previously took approximately one week to accomplish by hand can be taken care of in one night through automatic program generation using the proposed technology, which demonstrates that the learning period is fully practicable. As for the straight line recognition performance of the automatically generated program, an approximate doubling of position recognition accuracy was achieved compared with positioning tasks that until now could not be automated, and straight line position recognition error was within specification requirements, with a recognition rate of 97% or higher. Further, reduction of assembly work time by two-thirds through automation of the positioning process was achieved as a secondary effect.

5. Conclusion

This paper describes automatic generation technology for image-processing programs based on genetic programming developed by Fujitsu Laboratories, and a trial evaluation of this technology applied to a production line. An evaluation method using a hierarchy limit for the program structure and shape parameters was established through this trial evaluation, and the fact that it is possible to quickly generate high-performance programs through the introduction of an automatic selection technique for learning data based on image feature amount was also established.

Going forward, we aim to further enhance this technology for automatically generating all kinds of image recognition programs, and to expand the scope of application of the developed technology beyond just positioning.

References

1) S. Aoki et al.: Automatic Construction Method of Tree-structural Image Conversion Method ACTIT. Journal of the Institute of Image Information and Television Engineers, Vol. 53, No. 6, pp. 888–894, 1999.
2) T. Nagao: Evolutionary Image Processing. Shokodo, 2002.
3) J. R. Koza: Genetic Programming, Cambridge. MIT Press, 1992.
4) T. Nagato et al.: Technology for Automatic Generation of Image Processing Programs for Component Alignment. 2014 Japan Society for Precision Engineering Autumn Meeting Proceedings. Tottori, Japan Society for Precision Engineering, 2014, pp. 151–152.
5) K. Kushima et al.: Image Content Inspection Technology Based on Amount of Superficial Features such as Color and Shape. IPSJ Journal, Vol. 40, No. SIG3 (TOD1), pp. 171–184, 1999.

Tsuyoshi Nagato
Fujitsu Laboratories Ltd.
Mr. Nagato is currently engaged in research on image processing and recognition systems for factory automation.

Tetsuo Koezuka
Fujitsu Laboratories Ltd.
Mr. Koezuka is currently engaged in research on image processing and recognition systems for factory automation.