

Automatic Construction of Accurate Image Processing using AdaBoost

Jun Ando and Tomoharu Nagao

Graduate School of Environment and Information Sciences, Yokohama National University
79-7, Tokiwadai, Hodogaya-ku, Yokohama, Kanagawa, 240-8501, Japan
email:andou@nlab.sogo1.ynu.ac.jp

Abstract—Image processing and recognition technologies are required to solve various problems. We have already proposed the system which automatically constructs image processing with Genetic Programming (GP), Automatic Construction of Tree-structural Image Transformation (ACTIT). However, it is difficult to construct an accurate image processing for all training image sets if they have various characteristics. In this paper, we propose ACTIT-Boost which automatically constructs an accurate image processing by employing Adaptive Boosting (AdaBoost) to ACTIT. It learns training image sets and their areas which are difficultly approximated to target images in particular. We show experimentally that ACTIT-Boost is more effective in comparison with ordinary ACTIT.

I. INTRODUCTION

In the realization of machine intelligence, image processing and recognition technologies are gaining in importance. However, it is difficult to construct image processing in each problem. In this case, a general-purpose method that constructs image processing without depending on problems is necessary.

On the other hand, Evolutionary Computation studies[1][2][3] are widely applied to image processing[4]. Evolutionary Computation is an optimizing algorithm inspired by evolutionary processes of living things. We have previously proposed a system that automatically constructs an image-processing filter: Automatic Construction of Tree-structural Image Transformation (ACTIT)[5]. In this system, ACTIT approximates a required image processing by combining tree-structurally several image-processing filters prepared in advance with Genetic Programming (GP)[3], which is a type of Evolutionary Computation. We have ever shown that ACTIT is an effective method for many problems.

However, it is difficult that ACTIT constructs an accurate image processing which is applied to all images in case there are many and various images. Therefore, it is necessary to combine many subroutines of image processing for construction of an accurate image processing. In this paper, we propose ACTIT-Boost which automatically constructs an accurate image processing by employing Adaptive Boosting (AdaBoost)[6] to ACTIT. We describe the experimental results of comparison of ordinary ACTIT and ACTIT-Boost.

This paper is composed of the followings. Section 2 explains the related works, ACTIT, and AdaBoost. Section 3 describes ACTIT-Boost, which is the proposed system in this paper. Section 4 shows the experimental results and the analysis of

an image processing which the proposed system constructs. Finally, section 5 describes conclusions and future works.

II. RELATED WORKS

A. ACTIT

Automatic Construction of Tree-structural Image Transformation (ACTIT)[5] is a study of image processing using Genetic Programming (GP)[3]. It automatically constructs a tree-structural image transformation by combining several image-processing filters prepared in advance with GP by referring to training image sets. The individual in GP is a tree-structural image transformation. A tree-structural image transformation is composed of input images as terminal nodes, non-terminal nodes in the form of several types of image-processing filters, and a root in the form of an output image.

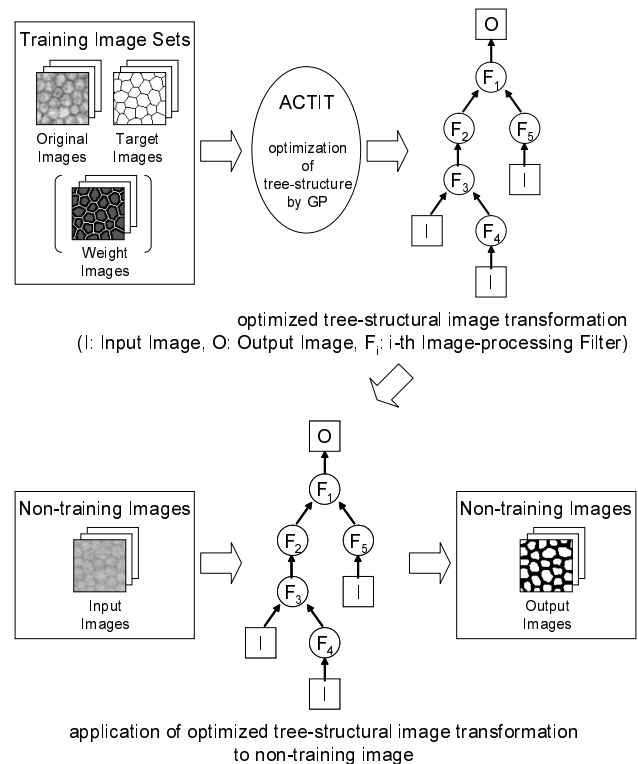


Fig. 1. The processing flow of ACTIT system.

Figure 1 shows the processing flow of the ACTIT system. Training image sets are prepared, including several original

images, their target images and weight images that indicate the important degree of pixel. We set the parameters that GP uses to optimize the tree structure and feed the training image sets to ACTIT. Then, ACTIT optimizes the tree-structural image transformation by means of GP. The fitness of individual is the difference of output images and target images with weight images. As a result, we can obtain an optimized tree-structural image transformation that has maximum fitness.

The tree-structural image transformation applies a certain processing mechanism to images that have the same characteristics. If the constructed tree-structural image transformation is appropriate, we can expect similar effects to the images that have the same characteristics as those learned. We have ever shown that ACTIT is an effective method for a number of problems, such as 2D image processing for the detection of defects and 3D medical image processing[7].

B. AdaBoost

Adaptive Boosting (AdaBoost)[6] is a machine learning algorithm, proposed by Freund and Schapire. AdaBoost is a kind of ensemble learning classifier, which constructs a strong classifier by combining weak classifiers. AdaBoost learns mistakable training data in particular. Figure 2 shows the illustration of AdaBoost learning.

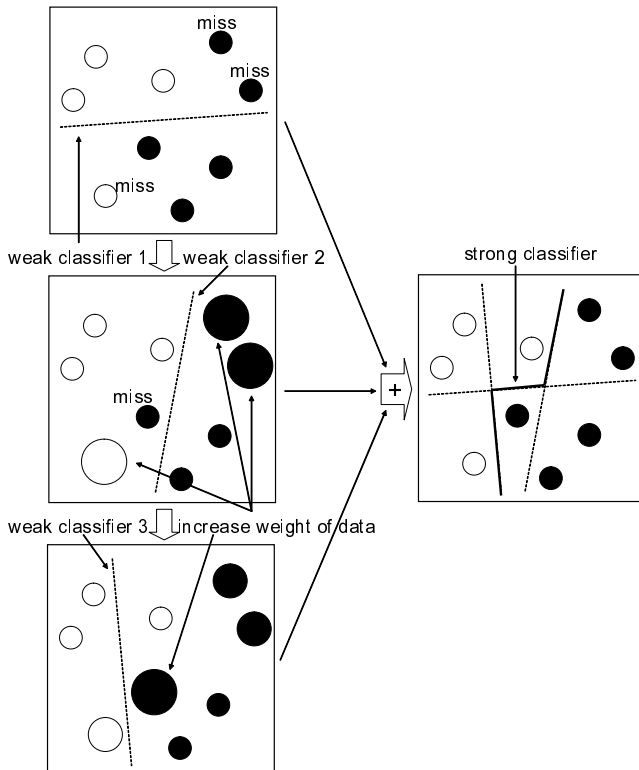


Fig. 2. The illustration of AdaBoost learning.

The following shows AdaBoost algorithm.

1. Prepare the training data sets:

$$D = (x_{11}, x_{12}, \dots, x_{1M}, y_1), (x_{21}, x_{22}, \dots, x_{2M}, y_2), \dots,$$

$(x_{N1}, x_{N2}, \dots, x_{NM}, y_N)$, where x is input feature data and y is output label data.

2. Initialize the weight of data: $w_i = 1/N, i = 1, 2, \dots, N$.
3. For $t = 1$ to T :

- a) Construct weak classifier $h_t(x)$ basing upon weight of data w_i .
- b) Calculate error rate with weight:

$$\varepsilon_t = \sum_{i=1}^N w_t(i) \frac{|h_t(x_i) - y_i|}{2} (< 0.5). \quad (1)$$

- c) Calculate confidence rate:

$$\alpha_t = \frac{1}{2} \log \frac{1 - \varepsilon_t}{\varepsilon_t} (> 0). \quad (2)$$

- d) Modify weight of data:

$$w_{t+1}(i) = w_t(i) \exp\{-\alpha_t y_i h_t(x_i)\}. \quad (3)$$

- e) Normalize w_i .

4. Output classifier label:

$$f_T(x) = \sum_{t=1}^T \alpha_t h_t(x). \quad (4)$$

If there are weak classifiers whose error rate with weight is smaller than 0.5 enough, AdaBoost has been proved to be extremely successful in producing accurate classifiers.

III. ACTIT-BOOST

We propose the system which automatically constructs an accurate image processing. The system is inspired by AdaBoost algorithm. AdaBoost has been proved to be extremely successful in producing accurate classifiers. Therefore, ACTIT-Boost constructs an image processing which completely approximates training image set by employing AdaBoost to ACTIT.

TABLE I
THE CORRESPONDENCE OF ADABOOST AND ACTIT-BOOST.

| AdaBoost | ACTIT-Boost |
|-----------------|--|
| Data | Pixels |
| Weight of data | Weight images |
| Weak classifier | Optimized tree-structural image transformation |
| Classifier rate | Fitness of the best individual |
| Classes | Black or white as pixel values |

Table I shows the correspondence of AdaBoost and ACTIT-Boost, and Fig. 3 shows the processing flow of ACTIT-Boost. The processing flow of ACTIT-Boost is similar to that of AdaBoost. Data as AdaBoost is pixels as ACTIT-Boost.

Firstly, weight images which is weight of data as AdaBoost is initialized by constant value. A small tree-structural image transformation which is weak classifier as AdaBoost is constructed basing upon training image sets. We name them weak tree-structural image transformations. Error rate and confidence rate are calculated from fitness which is difference of a target image and an output of image processing. Weight

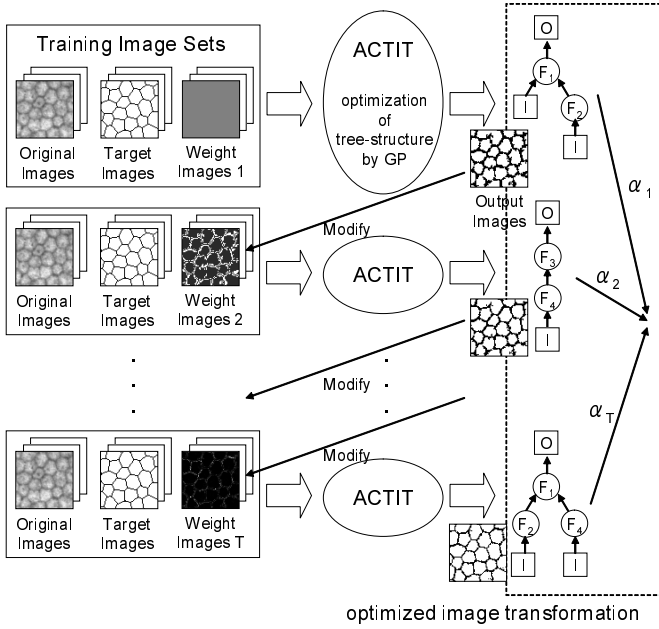


Fig. 3. The processing flow of ACTIT-Boost.

images are modified basing upon confidence rate. The multiple weak tree-structural image transformations are constructed by repeating the above process.

Lastly, we obtain output images by combining outputs of multiple weak tree-structural image transformations as follows.

$$O_T(x, y) = \text{sign}\left(\sum_{t=1}^T \alpha_t \{O_t(x, y) / V_{max} \times 2 - 1\}\right) \quad (5)$$

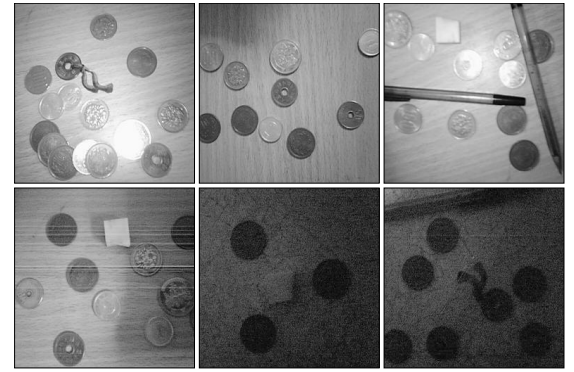
V_{max} is maximum of value (= 255). We expect that an accurate image processing is constructed by learning mistakable training image sets and their areas in particular.

In ordinary ACTIT, we experimentally determine weight images. However, it is difficult that we experimentally determine optimal weight images. On the other hand, in the proposed system, weight images are adaptively updated. It is an advantage of ACTIT-Boost against the ordinary ACTIT.

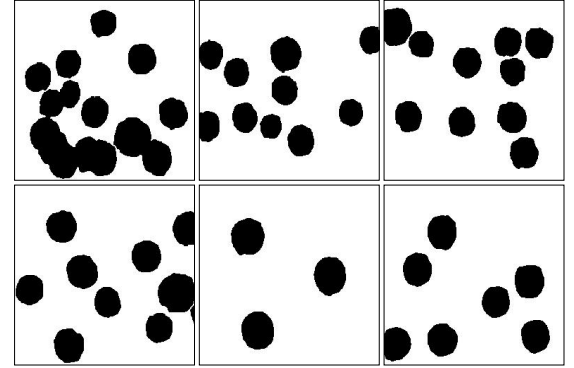
IV. EXPERIMENTS

A. Experimental setting

We compared the performance of the proposed system with ordinary ACTIT. We employed six coins segmentation images and six concrete cracks detection images as training image sets. Figure 4 and Fig. 5 show training image sets. It is difficult to construct an image processing for these images because these images have various characteristics, object color, background, lightness, and a border between textures. In proposed method, we don't have to prepare optimal weight images. However, we used the weight images generated by hand in ordinary ACTIT and the proposed method because of equal calculating fitness.

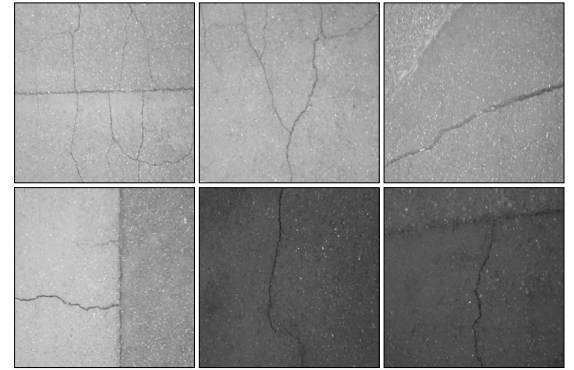


(a) original images

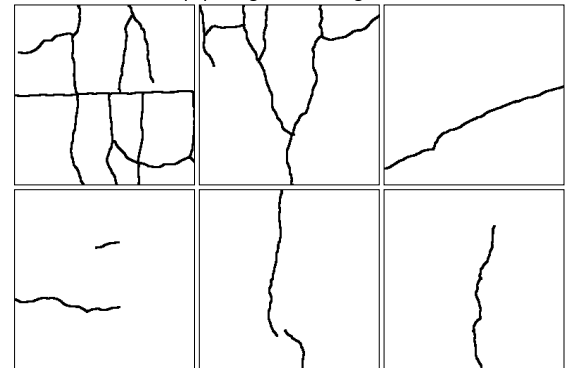


(b) target images

Fig. 4. Coins segmentation images.



(a) original images



(b) target images

Fig. 5. Concrete cracks detection images.

The total number of trees (weak classifiers) is 1000. In ordinary ACTIT, GP parameters are employed general values. In the proposed method, number of generations, population size, and maximum number of node are small because of constructing weak tree structural image transformations. Table II shows GP parameters in ordinary ACTIT and the proposed method. Besides, we employed Graphics Processing Unit (GPU)[8][9] to ordinary ACTIT and the proposed method for the purpose of reducing processing time[10]. We used 37 types of one or two input and one output simple image-processing filters on GPU (mean filter, linear transformation of histogram, difference filter, and so on).

TABLE II
GP PARAMETERS SETTING.

| | ordinary ACTIT | ACTIT-Boost |
|---|----------------|-------------|
| Number of generations | 30000 | 1000 |
| Population size | 200 | 50 |
| Number of individual evaluations per a generation | 40 | 5 |
| Maximum number of node | 40 | 10 |
| Number of tree-structural image transformations | 1 | 1000 |
| Crossover rate | 1.0 | 1.0 |
| Mutation rate (for individual) | 0.9 | 0.9 |

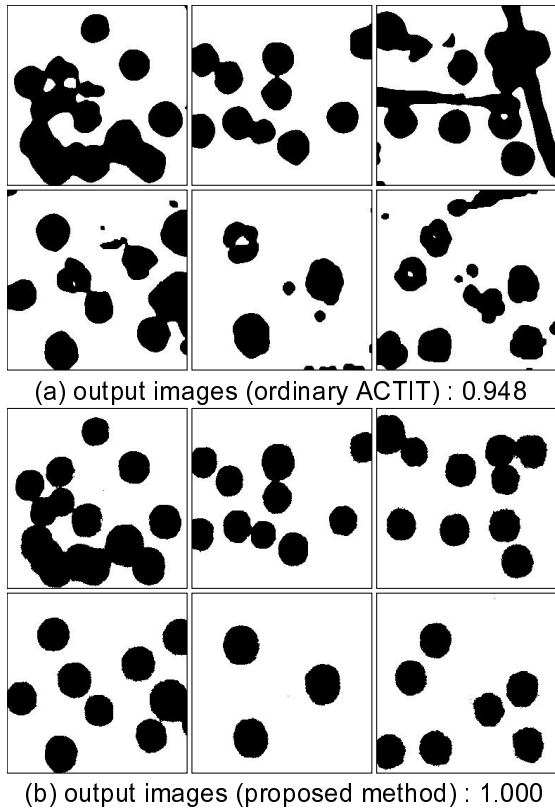


Fig. 6. Experimental results of comparison of ordinary ACTIT and the proposed method in case of coins segmentation images.

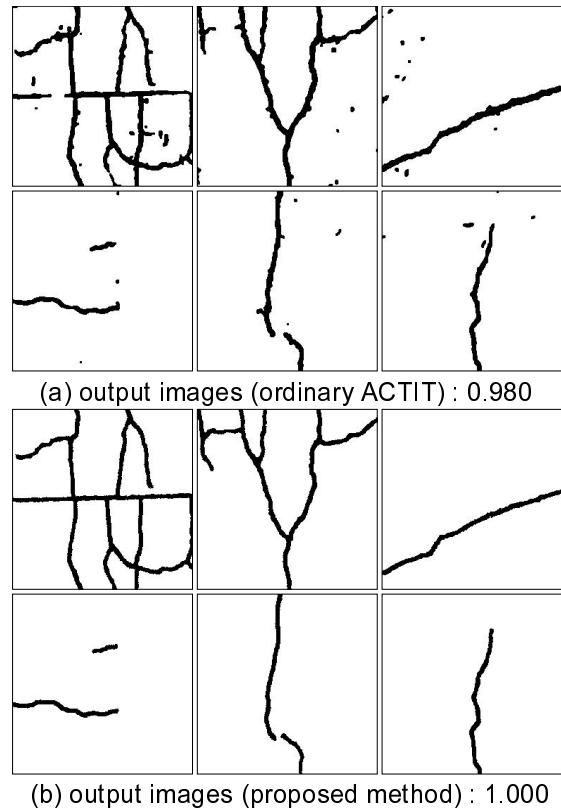


Fig. 7. Experimental results of comparison of ordinary ACTIT and the proposed method in case of concrete cracks detection images.

B. Experimental results

Firstly, we refer to the experimental results of application of optimized tree-structural image transformations to training image sets. Figure 6 and Fig. 7 show output images and the fitness of image processing. The fitness of image processing are average value of five trials.

In case of coins segmentation images, coins are successfully extracted in both ordinary ACTIT and the proposed method. However, in ordinary ACTIT, there are many noises on images and coins expand and connect to neighbor coins. On the other hand, in the proposed method, there are no noises on images and coins do not transform. In case of concrete cracks detection images, cracks are successfully detected in both ordinary ACTIT and the proposed method. However, in ordinary ACTIT, there are a few noises on a border between textures. On the other hand, in the proposed method, there are no noises on a border between textures.

Lastly, we refer to the experimental results of application of optimized tree-structural image transformations to non-training images. Figure 8 and Fig. 9 show output images and the fitness of image processing. The fitness for non-training images are calculated by comparing output images with target images and weight images generated by hand for non-training images.

In case of coins segmentation images and concrete cracks detection images, there are miss extraction and noises in

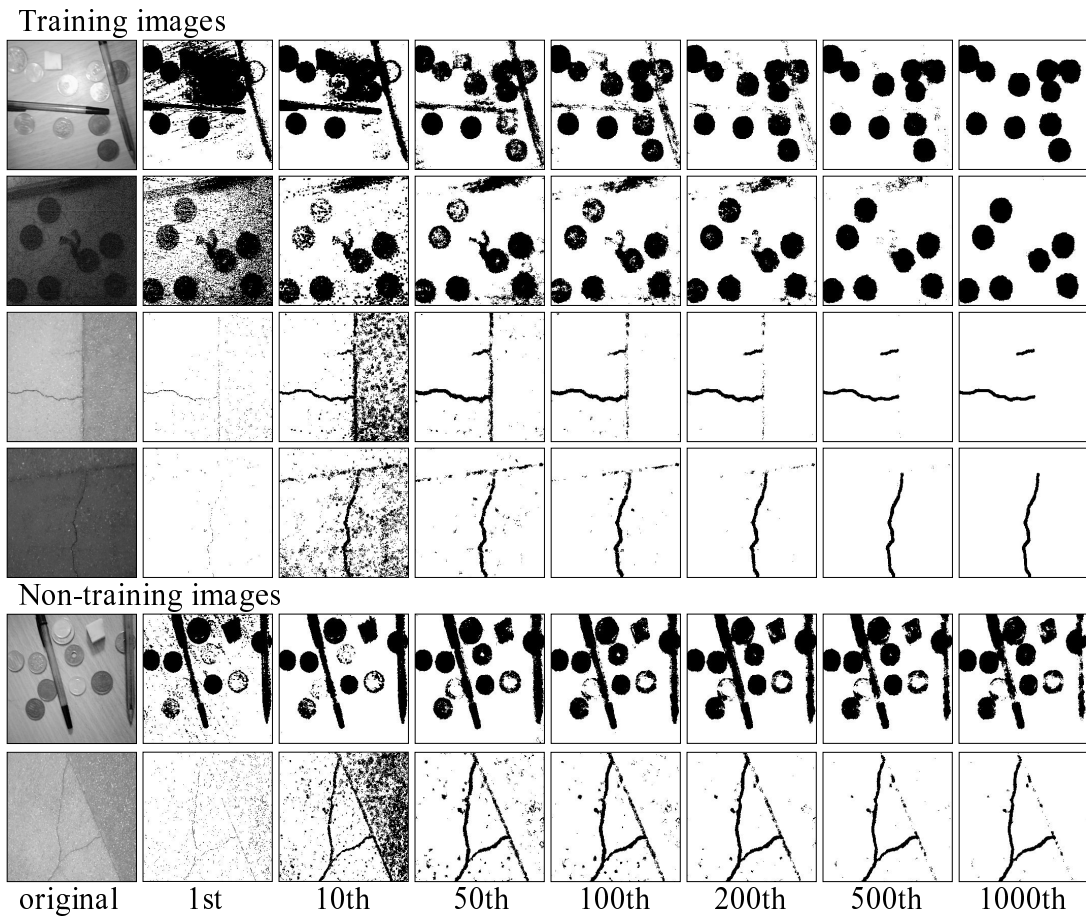
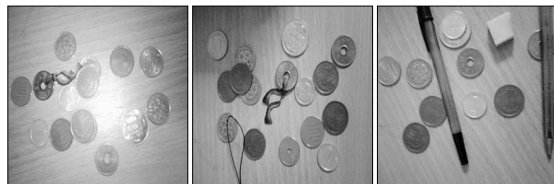
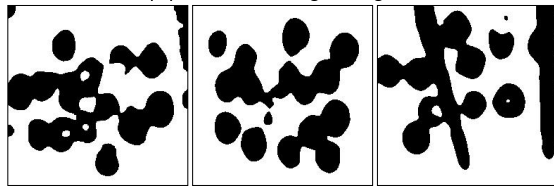


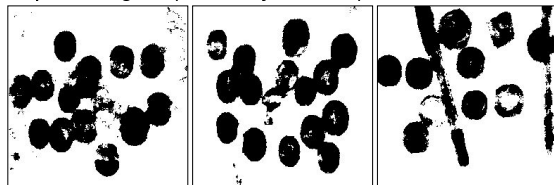
Fig. 10. Output images of each tree-structural image transformation.



(a) non-training images

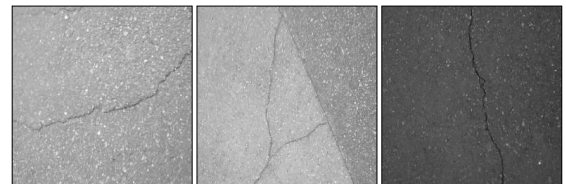


(b) output images (ordinary ACTIT) : 0.922, 0.971, 0.863



(c) output images (proposed method) : 0.960, 0.961, 0.858

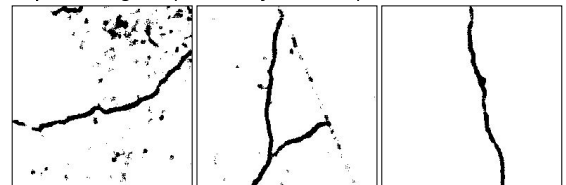
Fig. 8. Application of optimized tree-structural image transformations to non-training images in case of coins segmentation images.



(a) non-training images



(b) output images (ordinary ACTIT) : 0.813, 0.978, 0.999



(c) output images (proposed method) : 0.936, 0.987, 0.994

Fig. 9. Application of optimized tree-structural image transformations to non-training images in case of concrete cracks detection images.

both ordinary ACTIT and the proposed method. However, the quality of output images of the proposed method was better than that of ordinary ACTIT. These experimental results show that the proposed method is also effective for non-training images.

C. The optimized image processing structure

We discuss about the optimized image processing structure. Figure 10 shows output images of each tree-structural image transformation. According as weak tree-structural image transformations increase, noises on images decrease in case of training images. However, according as weak tree-structural image transformations increase, noises on images do not completely decrease in case of non-training images. We suppose that the proposed method excessively learns training images.

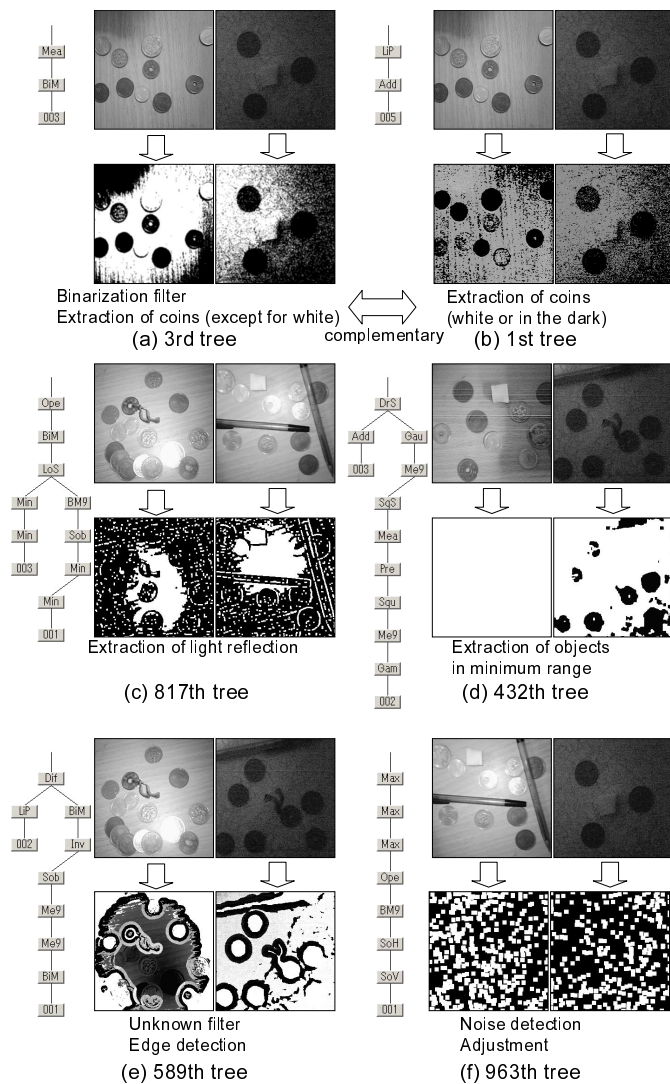


Fig. 11. Weak tree-structural image transformations.

Next, Fig. 11 shows the examples of weak tree-structural image transformations. We pay attention to 3rd tree which extract coins except for white ones and 1st tree which extract

white coins. Two kind of weak tree-structural image transformation complement each other. It shows that AdaBoost effectively work in ACTIT system. On the other hand, there are unknown filters too. We suppose that they work for adjustment of pixels.

V. CONCLUSION

We proposed ACTIT-Boost which automatically constructs an accurate image processing by employing AdaBoost to ACTIT. The quality of output images of the proposed method was better than that of ordinary ACTIT. We experimentally showed that the proposed method was effective for both training image sets and non-training images. Besides, we analyzed the optimized image processing structure. We examined that AdaBoost effectively worked in ACTIT system.

In future works, it is necessary that we apply the proposed system to many real images. We propose construction of image-processing algorithms which are effective for AdaBoost. And we aim to construct a fast evolutionary image processing system for many and various images.

REFERENCES

- [1] J. H. Holland, "Adaptation in Natural and Artificial Systems", *the Univ. Michigan Press*, 1975, *MIT Press*, 1992.
- [2] D. E. Goldberg, "Genetic Algorithms in Search, Optimization & Machine Learning", *Addison Wesley*, 1989.
- [3] J. R. Koza, "Genetic Programming on the Programming of Computers by Means of Natural Selection", *MIT Press*, 1992.
- [4] T. Nagao, "Evolutionary Image Processing", *Shokodo*, Japan, 2002.
- [5] S. Aoki and T. Nagao, "ACTIT; Automatic Construction of Tree-structural Image Transformation", *The Institute of Image Information and Television Engineers*, Vol. 53, No. 6, pp 888-894, 1999.
- [6] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting", *Journal of Computer and System Sciences*, 55(1):119-139, August 1997.
- [7] Y. Nakano and T. Nagao, "Automatic Extraction of Internal Organs Region from 3D PET Image Data using 3D-ACTIT", *International Workshop on Advanced Image Technology 2006*, 2006.
- [8] R. Fernando, "GPU Gems: Programming Techniques, Tips, and Tricks for Real-Time Graphics", *NVIDIA Corporation*, 2004.
- [9] M. Pharr and R. Fernando, "GPU Gems 2: Programming Techniques for High-Performance Graphics and General-Purpose Computation", *NVIDIA Corporation*, 2005.
- [10] J. Ando and T. Nagao, "Fast Evolutionary Image Processing using Multi-GPUs", *Proc. of The 2007 IEEE International Conference on Systems, Man, and Cybernetics*, pp 2927-2932, 2007.