JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 25 No. 3, pp. 19-26, March 2020
https://doi.org/10.9708/jksci.2020.25.03.019

Automatic Metallic Surface Defect Detection using ShuffleDefectNet

Avlokulov Anvar*, Young Im Cho*

*Student, Dept. of Computer Engineering, Gachon University, Seongnam, Korea
*Professor, Dept. of Computer Engineering, Gachon University, Seongnam, Korea

[Abstract]

Steel production requires high-quality surfaces with minimal defects. Therefore, the detection algorithms for the surface defects of steel strip should have good generalization performance. To meet the growing demand for high-quality products, the use of intelligent visual inspection systems is becoming essential in production lines. In this paper, we proposed a ShuffleDefectNet defect detection system based on deep learning. The proposed defect detection system exceeds state-of-the-art performance for defect detection on the Northeastern University (NEU) dataset obtaining a mean average accuracy of 99.75%. We train the best performing detection with different amounts of training data and observe the performance of detection. We notice that accuracy and speed improve significantly when use the overall architecture of ShuffleDefectNet.

▸ **Key words**: Defect detection, Deep Learning, ShuffleNet, Light-weight modules

[요　약]

일반적으로 품질 관리는 많은 제조 공정, 특히 주조 또는 용접과 관련된 공정의 기본 구성 요소가 된다. 그러나 사람이 일일이 수동으로 품질 관리 절차를 하는 것은 종종 시간이 걸리고 오류가 발생하기 쉽다. 최근 고품질 제품에 대한 요구를 만족시키기 위해 지능형 육안 검사 시스템의 사용이 생산 라인에서 필수적이 되고 있다. 본 논문에서는 이를 위해 딥 러닝 기반의 ShuffleDefectNet 결함 감지 시스템을 제안하고자 한다. 제안된 결함 검출 시스템은 NEU 데이터 세트의 결함 검출에 대한 여러 최신 성능들보다 높은 평균 정확도 99.75% 정도를 얻는다. 이 논문에서 여러 다른 트레이닝 데이터로부터 최상의 성능을 탐지하고 탐지 성능을 관찰하였다. 그 결과 ShuffleDefectNet의 전체 아키텍처를 사용할 때 정확성과 속도가 크게 향상됨을 알 수 있었다.

▸ **주제어**: 결함탐지, 딥러닝, 셔플넷, 라이트웨이 모듈

• First Author: Avlokulov Anvar, Corresponding Author: Young Im Cho
*Avlokulov Anvar (barselona4448@gmail.com), Dept. of Computer Engineering, Gachon University
*Young Im Cho (yicho@gachon.ac.kr), Dept. of Computer Engineering, Gachon University
• Received: 2019. 12. 31, Revised: 2020. 02. 21, Accepted: 2020. 02. 24.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

In the steel industry, the detect of steel surface defect plays an important role in finding the cause of defects in the manufacturing process. It is possible to eliminate the defects by utilizing the cause of defects of the product in real-time for the manufacturing process. As a result, mass defect in the continuous manufacturing process can be drastically reduced. Besides, process optimization can be performed by analyzing the pattern of defects according to the operating conditions.

Processes such as casting and welding can introduce defects in the product which are detrimental to the final product quality [1]. Common casting defects include air holes, foreign particle inclusions, shrinkage cavities, cracks, wrinkles, and casting fins [2]. Early detection of these defects can allow faulty products to be identified early in the manufacturing process, leading to time and cost savings [3]. Automated quality control can be used to facilitate consistent and cost-effective inspection. The primary drivers for automated inspection systems include faster inspection rates, higher quality demands, and the need for more quantitative product evaluation that is not hampered by the effects of human fatigue.

Many factors make real-time detection of steel strip surface defects particularly difficult, such as the high-speed production line, diversity and large scale changes of defects, random distribution and non-defective interferences (oil stains and dust on the surface of steel strips). Using ShuffleDefectNet, we can automatically extract multi-scale features of steel strip surface defects with good generalization and high accuracy by using a general-purpose learning procedure [4]. Using a trained network, defect regions can be detected in milliseconds. Therefore, SheffleNet v2 [5] can provide an accurate, real-time detection method for surface defects in steel strip product quality of steel strips.

For automatic metallic surface defect detection, we are using notification ShuffleNet V2 module. Because ShuffleNet V2 [5] is faster than the other networks, especially on GPU. For example, at 500MFLOPs ShuffleNet V2 [5] is 58% faster than MobileNet v2 [6], 63% faster than ShuffleNet V1 [7] and 25% faster than Xception [8]. On ARM, the speeds of ShuffleNet V1 [7], Xception [8] and ShuffleNet V2 [6] are comparable; however, MobileNet V2 [6] is much slower, especially on smaller FLOPs, because MobileNet v2 [6] has higher MAC, which is significant on mobile devices. Our method called ShuffleDefect and architecture is modification of ShuffleNet V2. For reaching high accuracy and speed we change 1 × 1 GConv to 1 × 1 Conv and 3 × 3 AVG Pool to 3 × 3 DWConv.

The rest of the paper is organized as follows. Next is the Related works section, we review some existing works and describe a dataset. CNN architecture description and show performed result from the experiment part in Section 3. Our conclusions are described in Section 4.

II. Related Works

1.1 Overview

Recent progress in the field of neural networks was triggered by several achievements. Deep learning architectures were enabled by increasing computing power and provide more and higher levels of representation [9]. Data augmentation, e.g. addition of artificial training data derived from the existing data through distortions, proved to be a powerful tool to avoid overfitting [10]. Committee methods can reduce the error rate by a combination of several networks, especially when the individual predictions are uncorrelated [11]. Finally, unsupervised methods for learning features and representations became very popular and solved problems with purely supervised training, e.g. dependence on random initialization, slow convergence, etc. [12]

Recently Masci et al. introduced Max-Pooling CNN model approach for supervised steel defect classification [13]. On a different approach, Soukup and Huber-M¨ork trained a CNN with stereo imaging to detect steel surface defects [14]. But the stereo acquisition method limits the application and cause the inference speed of this approach to be slow. Ke et al. tried using CNN-based defect-recognition in banknote images [15]. Even though the CNN performs better than traditional methods in results, study of the single type of (circular) defect, limits the usage in similar problems. Faghih-Roohi et al. used deep learning approaches with multiple CNN models to detect and classify rail surface defects and achieved 92% accuracy with 5 classes of defects [16]. Park et al. had a more holistic approach to surface inspection systems with their CNN-based system for surface defect inspection [16]. Park et al. show that even though CNN-based classifiers perform better than traditional methods with 92% accuracy, the inference time of 217ms is inferior to traditional methods [16]. Stricker Weimer et al. used multiple CNN models to automate the feature extraction in inspection systems [17]. Even though results show remarkable accuracy on textured images, they only used the simple circular and linear type of defects but not complex defect types.

Currently, the neural network architecture design is mostly guided by the indirect metric of computation complexity, i.e., FLOPs. However, the direct metric, e.g., speed, also depends on the other factors such as memory access cost and platform characteristics. Thus, this work proposes to evaluate the direct metric on the target platform, beyond only considering FLOPs. Based on a series of controlled experiments, this work derives several practical guidelines for efficient network design. Accordingly, a new architecture is presented, called ShuffleNet V2. Comprehensive ablation experiments verify that our model is state-of-the-art in terms of speed and accuracy tradeoff respect to these models.

1.2 Dataset

NEU surface defect is a defect classification dataset was opened seven years ago [18]. There are six types of defects from hot-rolled steel plates, including crazing, inclusion, patches, pitted surface, rolled-in scales, and scratches. The database includes 1800 grayscale imagesEach class has 300 images, but it does not mean that an image consists of a single defect. Examples of defect images are shown in Figure 1. To perform defect detection tasks, we provide annotations saved as XML files. With them, the classification dataset is upgraded to a detection dataset. The annotation marks the class and bounding box of each defect appearing in an image. Each bounding box is regarded as a ground truth box, which is represented by its top left and bottom right coordinates. There are nearly 5000 ground-truth boxes in total. For simplicity, we call the original dataset NEU-CLS and the complemented dataset NEU-DET.

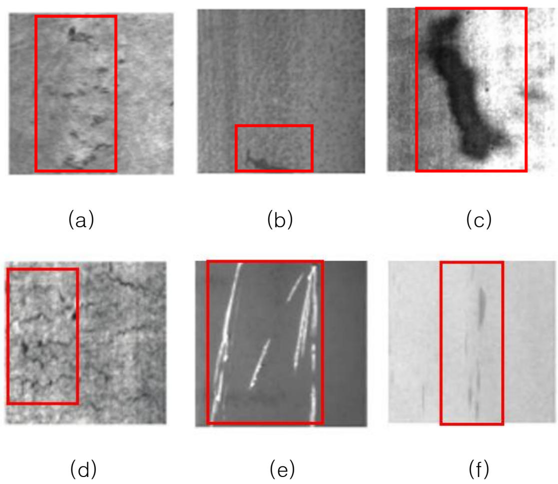

Fig. 1. Examples of the NEU defect dataset. Each column represents a type of defect, and the defect areas are labeled by the red bounding boxes. (a) rolled-in scale; (b) pitted surface; (c) patches; (d) crazing; (e) scratches; (f) inclusion.

III. The Proposed Scheme

3.1 Light Weight module Designs

A lightweight network structure is designed for ShuffleDefectNet with NEU surface defect detection. The network structure, which enhances the ability of speed and improve detection accuracy.

3.1.1 Point-wise Convolution

The point-wise convolution is the standard 1×1 convolution. It is used to aggregate information on different channels. The standard convolution convolves the input feature map both in the spatial-wise and the channel-wise dimensions. The depth-wise convolution could convolve the input feature map in the spatial-wise dimension, but it loses the information exchange among the different channels. Therefore, depth-wise convolutions and point-wise convolutions are complementary to each other. By using convolution factorization has a effects, we factorize each standard convolution layer of the baseline detection network into a depth-wise convolution layer and a point-wise convolution layer. The convolution factorization has two advantages over directly adopting standard convolutions. Firstly, the parameters of the network become much less. We suppose the input channels is Nin and the output channels are Not, the regular 3×3 convolution has 3×3×Nin×Nout parameters. After convolution factorization, the parameters become 3×3×Nin+1×1×Nin×Nout parameters. Secondly, the computation complexity is largely reduced. We use FLOPS as the index of computation complexity. We suppose the above convolution layer's input and output feature maps' spatial resolution are both H∗W. The FLOPS of regulation convolution would be H×W×3×3×Nin×Nout, while after convolution factorization, the FLOPS would be H×W×3×3×Nin+1×1×Nin×Nout × H × W. Since the convolution factorization could reduce much computation costs, we adopt it in all convolutional layers of the face detection network except conv1, including the backbone and the predicting layers. The regular convolutions with the same number of input and output channels can easily be factorized into depth-wise and point-wise convolutions. For the convolution which has a different number of channels between its input and output feature maps, the channel number transformation is accomplished at the point-wise convolutions.

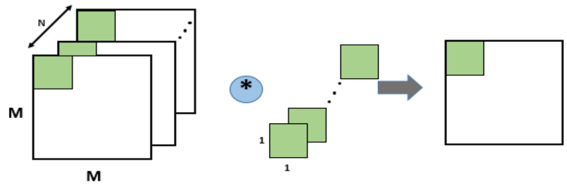

Fig. 2. Point-wise Convolution

3.1.2 Depth-wise convolution

In this section, we have described the depth-wise convolution. Depth-wise separable convolution layer has been proposed as an efficient alternative to the standard convolution operation. By replacing a standard 3-D convolution with a 2-D depth-wise convolution followed by a 1-D point-wise convolution, an efficient class of NN called as MobileNets. ShuffleNets utilize depth-wise convolutions on shuffled channels along with groupwise 1x1 convolutions to improve the accuracy with compact models. MobileNets V2 further improved the efficiency by adding shortcut connections, which help in convergence in deep networks. Overall, there have been many efficient neural network architectures proposed, which can be leveraged when developing a NN model specific for our hardware budget

3.1.3. GConv group convolution

Group convolution Figure 4 is a special case of a sparsely connected convolution. It was first used in the AlexNet [20] architecture and has more recently been popularized by its successful application in ResNeXt [21].

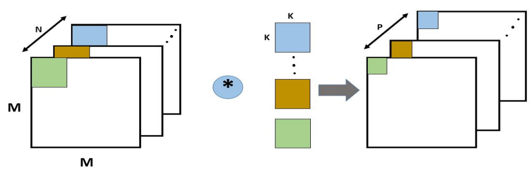

Fig. 3. Depth-wise convolution

Standard convolutional layers generate O output feature maps by applying convolutional filters overall I input feature maps, leading to a computational cost of I × O. In comparison, group convolution reduces this computational cost by partitioning the input features into G mutually exclusive groups and each group produces its outputs—reducing the computational cost by a factor G to I×O/G. ShuffleNet V2 suggests that the group number should be carefully chosen based on the target platform and task. It is unwise to use a large group number simply because this may enable using more channels because the benet of accuracy increase can easily be outweighed by the rapidly increasing computational cost. Channel shuffle operation makes it possible to build more powerful structures with multiple group convolutional layers. In the subsection, we will introduce an efficient network unit with channel shuffle and group convolutions.

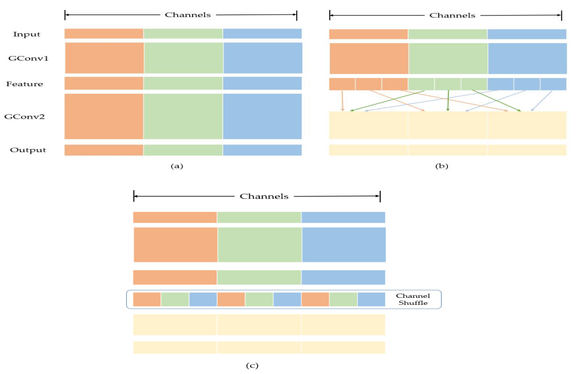

Fig. 4. Channel shuffle with two stacked group convolutions. GConv stands for group convolution. (a) Two stacked convolution layers with the same number of groups. Each output channel only relates to the input channels within the group. No cross-talk; (b) input and output channels are fully related when GConv2 takes data from different groups after GConv1; (c) an equivalent implementation to (b) using channel shuffle.

3.2. ShuffleDefectNet

Taking advantage of the channel shuffle operation, we propose a novel ShuffleDefectNet unit specially designed for small networks. We start from the design principle of the bottleneck unit in Fig 5 (c). It is a residual block. In its residual branch, for the 3 × 3 layer, we apply a computational economical 3 × 3 depthwise convolution on the bottleneck feature map. Then, we replace the first 1 × 1 layer with pointwise group convolution followed by a channel shuffle operation, to form a ShuffleDefectNet unit, as shown in Fig 5 (d).

ShuffleDefectNet (Table 1) is not only efficient, but also accurate. There are two main reasons. First, the high efficiency in each building block enables using more feature channels and larger network capacity.

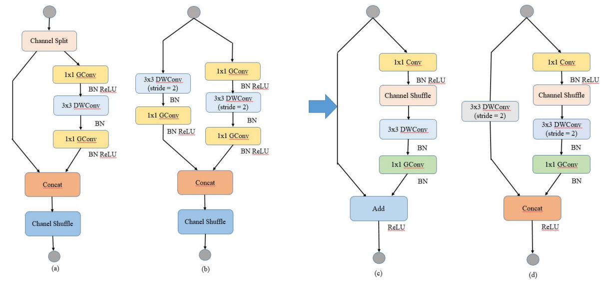

Fig. 5. Building blocks of our ShuffleDefectNet (c)(d).Old ShuffleNet V2 blocks (a)(b)

An extremely efficient convolutional neural network for mobile devices [19]. The main challenge for light-weight networks is that only a limited number of feature channels is affordable under a given computation budget (FLOPs). To increase the number of channels without significantly increasing FLOPs, two techniques are adopted in [19] pointwise group convolutions and bottleneck-like structures. A “channel shuffle” operation is then introduced to enable information communication between different groups of channels and improve accuracy. The building blocks are illustrated in Figure 4(a)(b).

Table 1. Overall architecture 200×200 image Resized to 192×192 of ShuffledefectNet with NEU dataset four different levels of complexities.

Layer	Output size	Ksize	Stride	Repeat	Output channels			
					0.5x	1x	1.5	2x
Image	192x192				3	3	3	3
Conv1 MaxPool	96x96 56x56	3x3 3x3	2 2	1	24	24	24	24
Stage2	48x48 24x24		2 1	1 3	48	116	176	244
Stage3	12x12 12x12		2 1	1 5	96	232	352	488
Stage4	6x6 6x6		2 1	1 3	192	464	704	976
Conv5	1x1	1x1	1	1	1024	1024	1024	2048
Globalpool		6x6						
FC					1000	1000	1000	1000
FLOPs					38M	136M	276M	579M
# of Weights					1.2M	2.1M	3.2M	6.9M

3.3 Experiments

For experimental comparisons, we use the standard network ShuffleDefectNet. All models of networks were implemented using the TensorFlow framework and trained on a single NVIDIA GeForce GTX 1080 Ti GPUs, Intel(R) Core(TM) i7-8700K CPU @ 3.20GHz. In the experiments, 300 images are used for training and 27 images are for testing.

As with many deep learning tasks, it takes a large amount of labeled data to train an accurate detection. We train the best performing classifiers with different amounts of training data and observe the performance of detection. We notice that accuracy and speed improve significantly when use the overall architecture of ShuffleDefectNet(Table 1).

The proposed defect detection system exceeds the ResNet34 model from [22] in terms of accuracy and evaluation time. The improvement in accuracy is thought to be largely due to benefits arising from a joint prediction of bounding boxes and segmentation masks. Both systems take a similar amount of time to evaluate on the CPU, but the proposed system is faster than the ResNet34 system when evaluated on a GPU and we can see our network has best results than AECLBP[23], BYEC[24], OVERFEAT[25], Decaf[26], MVM-VGG[27]. As we know, stronger performance on defect classification should be positively correlated with stronger performance on defect detection. A good classification result is a prerequisite for subsequent defect detection experiments. As we know, stronger performance on defect classification should be positively correlated with stronger performance on defect detection. A good classification result is a prerequisite for subsequent defect detection experiments. There is an inherent tradeoff between speed and accuracy in most modern object detection networks systems based on the ShuffleDefectNet framework with NEU defect dataset. As shown in Table2, our architecture increases high accuracy. By the way, changing 1 × 1 GConv to 1 × 1 Conv is reaching high speed. The proposed defect detection system exceeds the previous state-of-the-art performance on casting defect detection reaching an accuracy of 99.75% in Table2. The proposed architecture is easy to use and possible to apply to any standard metal surface defect detection with considering time and accuracy complexity by limiting the number of the blocks.

Table 2. Detection results on NEU dataset

Model	Accuracy%
BYEC	96.30
OVERFEAT	98.70
AECLBP	98.93
ResNet34	99.33
MVM-VGG	99.50
ShuffleNet v2	99.64
Decaf	99.70
SuffleDefectNet	99.75

IV. Conclusions

In this paper, the defect detection network (ShuffleDefectNet), a defect inspection system for metal steel surfaces this system is the network that can obtain the specific category and detailed location of a defect by fusing the multilevel features. For defect detection tasks, our system can provide detailed for quality assessment systems, such as the quantity, category, complexity, and area of a defect. The proposed defect detection system exceeds state-of-the-art performance for defect detection on the Northeaster University (NEU) dataset obtaining a mean average accuracy of 99.75%. The defect detection system described in this work can detect casting and welding defects with very high accuracy. Future work could involve training the same network to detect defects in other materials such as wood or glass.

ACKNOWLEDGEMENT

This work was supported by NRF(project number is 2018R1D1A1A09084151) and the MSIT, Korea, under the ITRC support program(IITP-2020-2017-0-01630).

REFERENCES

[1] R. Rajkolhe and J. Khan, "Defects, causes and their remediesin casting process: A review," vol. 2, no. 3, pp. 375-383.

[2] X. Li, S. K. Tso, X.-P. Guan, and Q. Huang, "Improving automatic detection of defects in castings by applying wavelet technique," vol. 53, no. 6, pp. 1927-1934.

[3] S. Ghorai, A. Mukherjee, M. Gangadaran, and P. K. Dutta, "Automatic defect detection on hot-rolled flat steel products," vol. 62, no. 3, pp. 612-621.

[4] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. nature, 521(7553), 436.

[5] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, Jian Sun "ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design" (Submitted on 30 Jul 2018)

[6] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Inverted residuals and linear bottlenecks: Mobile networks for classification, detection, and segmentation. arXiv preprint arXiv:1801.04381 (2018)

[7] Zhang, X., Zhou, X., Lin, M., Sun, J.: Shufflenet: An extremely efficient convolutional neural network for mobile devices. arXiv preprint arXiv:1707.01083 (2017)

[8] Chollet, F.: Xception: Deep learning with depthwise separable convolutions. arXiv preprint (2016)

[9] Ciresan, D., Meier, U., Schmidhuber, J.: Multi-column deep neural networks for im-age classification. In: Proc. of Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3642-3649 (June 2012)

[10] Arel, I., Rose, D.C., Karnowski, T.P.: Deep machine learning - a new frontier inartificial intelligence research. IEEE Computational Intelligence Magazine 5(4),13-18 (2010)

[11] Simard, P.Y., Steinkraus, D., Platt, J.C.: Best practices for convolutional neural networks applied to visual document analysis. In: Proc. of International Conference on Document Analysis and Recognition (ICDAR), pp. 958-963 (2003)

[12] Coates, A., Lee, H., Ng, A.Y.: An analysis of single-layer networks in unsupervised feature learning. In: Proc. of International Conference on Artificial intelligence and statistics (AISTATS) (2011)

[13] Soukup, D., Huber-M□work, R.: Convolutional neural networks for steel surface defect detection from photometric stereo images. Advances in Visual Computing 8887 (2014) 668{677 3

[14] Ke, W., Huiqin, W., Yue, S., Li, M., Fengyan, Q.: Banknote Image Defect Recognition Method Based on Convolution Neural Network. International Journal of Security and Its Applications 10(6) (2016) 269{280 3

[15] Faghih-Roohi, S., Hajizadeh, S., Nu~nez, A., Babuska, R., Schutter, B.D.: Deep convolutional neural networks for detection of rail surface defects. In: 2016 International Joint Conference on Neural Networks (IJCNN). (2016) 2584{2589 3

[16] Park, J.K., Kwon, B.K., Park, J.H., Kang, D.J.: Machine learning-based imaging system for surface defect inspection. International Journal of Precision Engineering and Manufacturing-Green Technology 3(3) (Jul 2016) 303{310 3

[17] Weimer, D., Scholz-Reiter, B., Shpitalni, M.: Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection. CIRP Annals - Manufacturing Technology 65(1) (Jan 2016) 417{420 4

[18] K. Song and Y. Yan, "A noise-robust method based on completed local binary patterns for hot-rolled steel strip surface defects," Appl. Surface Sci., vol. 285, pp. 858-864, Nov. 2013.

[19] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Inverted residuals and linear bottlenecks: Mobile networks for classification, detection, and segmentation. arXiv preprint arXiv:1801.04381 (2018)

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages 1097-1105, 2012

[21] S. Xie, R. Girshick, P. Dollar, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on, pages 5987-5995. IEEE, 2017.

[22] Yu He, Kitchen Song, Qinggang Meng, Yunhui Yan. An End-to-end Steel Surface Defect Detection Approach via Fusing Multiple Hierarchical Features. DOI 10.1109/TIM.2019.2915404, IEEE

[23] K. Song and Y. Yan, "A noise-robust method based on completed local binary patterns for hot-rolled steel strip surface defects," Appl. Surface Sci., vol. 285, pp. 858-864, Nov. 2013.

[24] M. Xiao, M. Jiang, and G. Li et al., "An evolutionary classifier for steel surface defects with a small sample set," EURASIP J. Image Vid. Process., vol. 2017, no. 1, pp. 48-61, Dec. 2017

[25] P. Chen, and S.S. Ho, "Is over feat useful for image-based surface defect classification tasks?" in Proc. IEEE Int. Conf. Image Process. (ICIP), AZ, USA, Sep. 2016, pp. 749-753.

[26] R. Ren, T. Hung, and K.C. Tan, "A generic deep-learning-based approach for automated surface inspection," IEEE Trans. Cybern., vol. 48, no. 3, pp. 929-940, Mar. 2018.

[27] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436-444, May 2015.

Authors

Avlokulov Anvar received B.C Degree in Tashkent University of Information Technology, dept. of Software Engineering. Avlokulov Anvar joined the faculty of the Department of Computer Engineering at Gachon University, in 2017. His research areas: Artificial Intelligence, Computer Vision, Pattern Recognition and Machine Learning

Young Im Cho received the B.S., M.S. and Ph.D. degrees in Computer Science and Engineering from Korea University, Korea, in 1988, 1990 and 1994, respectively. She is a Professor at Gachon University. Her research interest includes AI, Big Data, Information Retrieval, Smart City, etc.