Automated detection of textured-surface defects using UNet-based semantic segmentation network

Nastaran Enshaei
Concordia Institute for Inf. Sys. Eng.
Concordia University
Montreal, Quebec H3G 1M8
nastaran.enshae@encs.concordia.ca

Safwan Ahmad
Mechanical, Aerospace & Ind. Eng.
Concordia University
Montreal, Quebec H3G 1M8
an_ibra@encs.concordia.ca

Farnoosh Naderkhani
Concordia Institute for Inf. Sys. Eng.
Concordia University
Montreal, Quebec H3G 1M8
Farnoosh.naderkhani@concordia.ca

***Abstract*—Over the recent years, developing a reliable automated visual inspection system/approach for manufacturing and industry sectors which are moving toward smart manufacturing operations faces lots of significant challenges. Traditional visual inspection techniques which are developed based on manually extracted features, can rarely be generalized and have shown weak performance in real applications in different industries. In this paper, we propose a novel and automated visual inspection system which can outperform the statistical methods in terms of detection and the quantification of anomalies in image data for performing critical industrial tasks such as detecting micro scratches on product. In particular, an end-to-end UNet-based fully convolutional neural network for automated defect detection in industrial surfaces is designed and developed. The proposed network has the capability to accept raw images as input and the output is pixel-wise masks. In order to avoid overfitting and improve the model generalization, we use real-time data augmentation approach during our training phase. To evaluate the performance of the proposed model, we use a publicly available data set containing ten different types of textured-surfaces with their associated weakly annotated masks. The findings indicate that despite working with roughly annotated labels, our results are in agreement with previous works and show improvements regarding the detection time.**

I. Introduction

Visual inspection plays an important role in process monitoring and product quality control in prognostic and health management (PHM) context. The goal of visual inspection is to verify that a defect-free product is delivered to the successive level or distributed to the final customer. In today's competitive market, significant advances in communication systems and sensor technologies have motivated manufacturing policymakers and industry owners to move toward smart manufacturing. Over recent years, automated visual inspection systems in quality control through collecting and analyzing high-dimensional data like images and videos are attracting more attention due to their performance in increasing customer's satisfaction by decreasing number of defective items/defects which lead to huge decrease in product recall cost. However, most of the existing defect detection techniques are based on manually extracting features followed by a classification task. In these approaches, feature extraction is performed for every individual case by the expert designers with prior knowledge of the domain which makes the process very time consuming and dramatically decreases the generalization of provided algorithms.

In general, surface defect detection methods can be categorized into two main groups, namely, classical and deep learning (DL)-based methods. Classical surface defect detection methods can be categorized into three groups: (i) Statistical methods where detection of surface defects is performed by evaluation of pixel intensities and their associated statistical characteristics based of fractal dimensions [1], thresholding [2] and local binary pattern [3]; (ii) Filter-based methods which are based on frequency domain or time-frequency domain representations of pixel intensities through transformation operations like Fourier transform [4] and Wavelet transform [5]; and finally (iii) Model-based methods which are based on Hidden Markov trees [6] and Weibull distribution [7]. Both statistical and filter-based methods are sensitive to diverse stochastic patterns on surfaces, but model-based techniques have a better performance on textured surfaces by transforming intensities distribution to a lower dimensional space [8]. Classical methods are highly dependent on extracted features and should be redesigned for any individual case.

Over recent decades, with emerging of new sensor generations and availability of large amount of high-dimensional data, DL frameworks have been widely applied in all aspects of manufacturing including surface defect detection and product quality control. DL approaches are usually designed based on convolutional neural networks (CNN) due to their significant capability in automatically extracting powerful features from images during training phase. Generally speaking, DL-based models applied for the purpose of surface defect detection can be classified into three main categories:

(1) Classification models: The classifier is trained to detect different types or the severity of the defects [14], [15]. For example, in [14], a deep CNN model was trained to detect three classes of surface defects including damage spot, glue mark and dust/ fiber spot on a data set of defective flat metallic surfaces.

(2) Object detection models: The algorithm learns to track the location of a defect spot inside a surface with a bounding box [16]- [19]. The most powerful algorithms for object detection tasks are Regional-CNNs (R-CNNs) and You Look Only Once (YOLO). In [17], the authors developed a Faster R-

978-1-7281-6286-7/20/$31.00 ©2020 IEEE

CNN network for detecting multiple structural damage types in an automatic visual inspection. The authors in [18], proposed an improved YOLO network containing 27 convolution layers for real-time defect detection of steel strip surfaces.

(3) Semantic segmentation models: The algorithm is trained to receive an image and through a pixel-wise classification process produce a mask containing the precise shape of a defect spot inside the image. For training a semantic segmentation algorithm, annotated labels of image data set should be provided which requires the huge amount of time and human efforts. Fully convolutional networks (FCNs) [10] is a successful semantic segmentation algorithm which contains a contracting path for extracting high-resolution features from images and an up-sampling path for localizing the information and producing the masks. One of the newly developed FCN algorithm with a symmetric contracting/expansive structure referred to as UNet algorithm is proposed by [9] in 2015. The novelty of this algorithm is the idea of skip connections such that the high-resolution features extracted in each block of contracting path are transferred to the same level in an expansive path which helps the algorithm to achieve a more precise localization. Skip connections have made a significant improvement in the performance of image segmentation networks.

Recently, some researchers have applied different extensions of FCNs for the purpose of surface defect detection [20]-[24]. Although the researchers for anomaly detection on textured surfaces in PHM context are vast, the results are not satisfactory due to existence of divers patterns in texture surfaces [23]-[24]. Therefore, in this paper we try to address this gap by developing a UNet-based model which is able to detect powerful features from images automatically and predict the defect regions inside the industrial textured surfaces. We perform the experiments by two slightly different networks. In first network, we developed a UNet-based network structure and in the second one, following [25] and [18], we replace the max-pooling layers by convolutional layers to let the algorithm to learn the proper weighted-average for down-sampling the parameters during the training phase. The experimental results are presented on ten different types of textured-surfaces. The proposed end-to-end surface defect detection algorithm is capable to extract high-resolution features from input images and produce a mask displaying the shape of the defect inside a surface. In order to avoid overfitting, we implement the real-time data augmentation strategies convenient for our data set.

The reminder of this paper is organized as follows: In Section II, we describe the structure of our image segmentation algorithm. Section III deals with the model implementation and the data set we used for our model evaluation. In section IV, the experimental results are presented and discussed. Finally, Section V concludes the paper.

II. Network Structure

In this paper, we propose two image segmentation models inspired by UNet algorithm. The proposed models contain a contracting path for extracting high-resolution features from images, a bottleneck and an expansive path for localization and building the mask. In Model 1, each block of contacting path contains a 3×3 convolution layer following by a Batch Normalization (BN) layer and then a Rectified Linear Unit (ReLU). Then, a Max-pooling layer for down-sampling following by a Dropout layer for improving generalization are applied. The procedure is shown in Figure 1. Taking the recommendation of the original BN paper proposed by [26], we applied BN transformation immediately before ReLU function. Bottleneck contains only a convolution layer followed by a BN layer and a ReLU activation function (Figure 2).

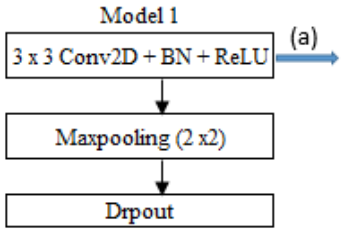

Fig. 1. A block of contracting path in model 1, (a): feature map transformation

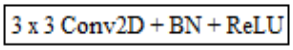

Fig. 2. Bottleneck block

In an expansive path which is shown in Fig. 3, first a 3×3 up-convolution layer is applied on the input feature maps and the results are concatenated by corresponding feature maps from contracting path. After implementing a Dropout layer, a 3×3 convolution layer following by BN layer and then RelU function are applied. We considered four blocks in both contracting and expansive paths of our algorithm. Last layer contains a 1×1 convolutional layer with the Sigmoid activation function where a class label between [0, 1] is predicted for each pixel. When evaluating the results, the values above 0.5 are considered as class 1 and the rest are considered as class zero. This completes our discussion on Model 1.

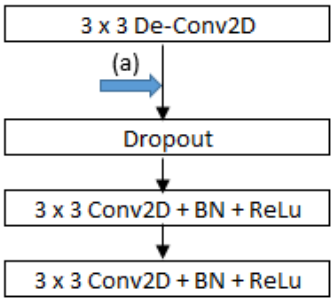

Fig. 3. A block of up-sampling path, (a): concatenating feature map.

Model 2 has the similar structure of Model 1. The only difference is that in Model 2, the max-pooling layer has been replaced with a 3×3 convolution layer with strides

2×2 following by a BN layer and a ReLU function 4. Indeed, during training phase, Model 2 learns how to perform a weighted down-sampling of information. The number of

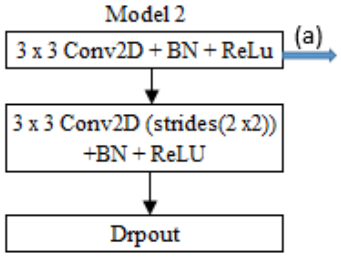

Fig. 4. A block of contracting path in model 2, (a): feature map transformation

trainable parameters inside a DL network is an important criterion for evaluating the network's efficiency. The number of Kernels in first block of down-sampling path is considered 32 and gets double in each following step such that the bottleneck contains 512 feature channels. The number of feature channels in same levels of contracting and expansive path are equal. The number of trainable parameters for Model 1 and Model 2 are 4.7 and 7.851 million, respectively.

III. Model Implementation

In this section, detailed description of implementation procedure is discussed.

A. Details of experiment

The proposed model receives an image of size 512×512 as an input and produces a 512×512 mask as an output. Developments and experiments are ran on GPU: NVIDIA GeForce RTX 2080 Ti, CPU:i7-9700, RAM: 64 G, software environment: Python 3.7 Keras, Tensorflow backend.

To capture the features non-linearity, we used ReLU as an activation function in all layers inside the algorithm. In output layer, where a label should be assigned to each pixel, the Sigmoid function is used to predict a class label in range of [0,1] for each pixel. The values above 0.5 is considered as defective area and the rest is considered as non-defective.

Since our images contain only two class labels (defect and defect-free), the binary cross-entropy loss function is applied as follows:

$$L(y, \hat{y}) = \sum_{i}^{n} [\, y_i \, log \, \hat{y}_i + (1 - y_i) \, log(1 - \hat{y}_i], \qquad (1)$$

where n is the number of samples, y_i is the ground truth label and $\hat{y}_i$ is the predicted label. Adam optimizer with initial learning rate equal to 0.001 and decay factor equal to $1e^{-}6$ has been used for network optimization.

To train our network efficiently, the real-time data augmentation is applied [12]. Data augmentation is the most common used method in machine vision to cope with lack of sufficient data and reduce overfitting. This technique, artificially generates new images from existing images by strategies like flipping, cropping, zooming, whitening and etc. Compared to off-line image augmentation techniques which the algorithm observes repetitively the same images in all epochs over training phase, in real-time data augmentation techniques, images are generated from each mini-batch images and the algorithm will see each artificial image only once that will result in significant improvements in model generalization. Saving disk space is another advantage of real-time data augmentation method particularly when different types of augmentation strategies are applied. Indeed, for each training iteration, one batch of artificially augmented images along with the original images are fed into the network and consequently over each epoch the number of artificially generated images would be equal to the number of images in the training set. In general, the number of training samples generated by data augmentation technique over the whole training process is equal to the number of images in the original training set multiplied by the number of training epochs. In this study, we use the Keras DL library from python for automatically augmenting our data during the training phase.

B. Metrics

In each DL model, a convenient metric is needed in order to fairly evaluate the performance of the model along with the model's advantageous and limitations. In classification tasks, metrics are usually calculated based on the number of True Positive (TP), False Positive (FP) and False Negative (FN) classified samples (Figure 5).

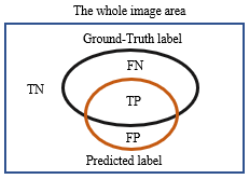

Fig. 5. Representation of TP, FP and FN

In this paper, Intersection-over-Union (IoU), F1-score and pixel-accuracy are used as our evaluation metrics that are calculated as follows:

$$\begin{aligned} \textit{Pixel-accuracy} &= \frac{TP + TN}{TP + TN + FN + FP}; \\ \textit{IoU} &= \frac{TP}{TP + FN + FP}; \\ \textit{F1-score} &= \frac{2\,TP}{2\,TP + FN + FP}. \end{aligned} \qquad (2)$$

Pixel-accuracy measures the number of pixels classified correctly relative to all the total number of pixels, while F1-score and IoU simply compare a predicted mask with the ground-truth label. F1-score and IoU are more powerful metrics for evaluation of a model performance in an imbalanced data set.

C. Data set description

The data set used for this study is a publicly available textured surfaces data set created at 2007 for DAGM competition [13]. This artificially generated data set consists of different classes of defective and non-defective textured surfaces,

each class created by a different texture and defect model. Images have been saved in grayscale 8-bit format and with the size of 512×512 pixels.

For all defective images, a mask has been provided where the defective area is roughly indicated with an ellipse. As shown in Figure 6, in each provided label parts of non-defective area also encircled by the ellipse.

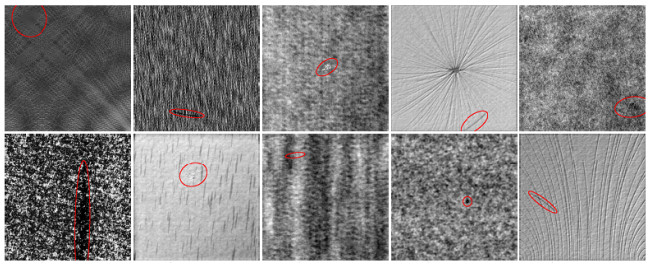

Fig. 6. Different textured surfaces and their associated weakly labels in the data set (Surface 1 (top-left) to Surface 10 (bottom-right).

IV. Experimental Results

As mentioned earlier, to evaluate the performance of our model, we use IoU, F1-score and pixel-accuracy as evaluation metrics. We calculate each metric by comparing the predicted mask for each given test image with its associated ground-truth label and take the average over the whole test set of every class. A summary of results are presented in Table I. As the average results show, there is a slight improvement when Model 2 is applied. For both models, the average of pixel-accuracy over all test sets is above 99 percent. Although Model 1 in a few classes of data yields slightly better results, the overall performance of Model 2 is better. The mean values of F1-score and mean-IoU in Model 2 is higher than model 1 which indicates model 2 where the network uses a trainable down-sampling layer instead of the conventional max-pooling one is showing better performance. Figure 7 and Figure 8 represent the variation of F1-score and IoU metrics over different types of surfaces. The minimum of IoU is 0.48 and 0.54 for Model 1 and Model 2, respectively which indicates the model is working reasonably in predicting the defective regions in diverse textured-surface.

We also make a comparison between the proposed models and model developed by [21] where the authors perform an image segmentation task on 6-class textured surfaces using a precisely manually annotated data set and achieved the mean-IoU equal to 0.73 and the detection time of 25 images per second. On the other hand, in our proposed model, the mean-IoU of 0.6835 for a 10-class data set and the detection time of 40 images per second are archived which indicates despite working with a roughly annotated dataset our results are in agreement with other works and show superiority regarding the detection time. It is worth mentioning that Our model can work reasonably on weakly annotated data which is considered a great advantage since in many industrial applications providing a high-quality annotated dataset is very time-consuming and expensive.

In Figure 9, some samples of produced masks on different classes are shown for illustration purposes where Figure 9 (a) demonstrates the most accurate mask produced by the model in that class of test data, 9(b) and 9(c) indicate the worse predicted label in each class regrading F1-score and pixel-accuracy, respectively.

TABLE I
The evaluation of model performance using IoU, F1-score and pixel-accuracy metrics

	Model 1			Model 2		
	IoU	F1_Score	Pix_acc	IoU	F1_Score	Pix_acc
C1	0.685	0.805	0.986	0.6887	0.8043	0.9863
C2	0.615	0.7332	0.9954	0.655	0.7823	0.9953
C3	0.6085	0.751	0.9931	0.615	0.7513	0.9929
C4	0.489	0.5926	0.9861	0.5495	0.6314	0.9871
C5	0.722	0.83	0.9937	0.631	0.7642	0.9911
C6	0.8708	0.9297	0.984	0.8292	0.9052	0.9833
C7	0.677	0.7974	0.984	0.7431	0.8475	0.9879
C8	0.547	0.686	0.9967	0.5962	0.734	0.9973
C9	0.839	0.9077	0.999	0.8647	0.9268	0.9995
C10	0.71	0.8259	0.9957	0.6629	0.7862	0.9954
Ave	0.6763	0.7859	0.9914	0.6835	0.7933	0.9916

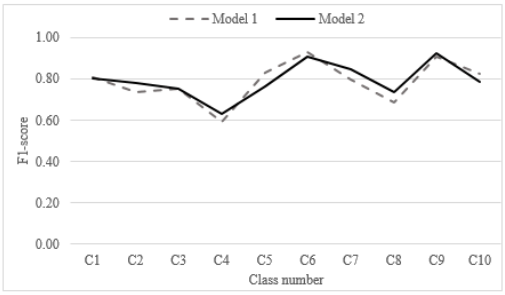

Fig. 7. F1-score metric obtained by Model 1 and Model 2 for different class surfaces

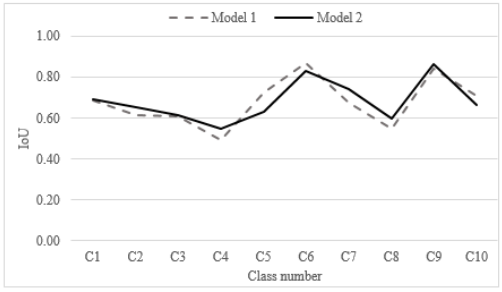

Fig. 8. IoU metric obtained by Model 1 and Model 2 for different class surfaces

V. Conclusion

This paper proposes a DL-based solution referred to as UNet-based image segmentation model for automated defect

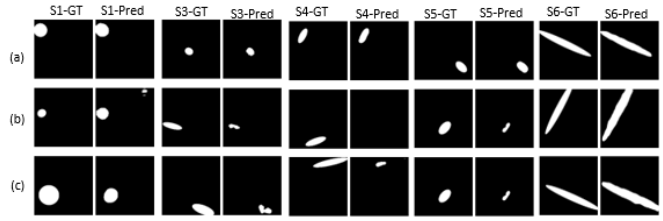

Fig. 9. A snapshot of predicted masks for different test images, For each class in first and second row the best and The worse prediction regarding F1-score and in the second row the worse prediction regarding the pixel-accuracy metric are presented

detection of textured-surfaces on a weakly annotated industrial data set. We evaluate the performance of our proposed model on 10 different types of textured surfaces. The results indicate the effectiveness of our proposed model on detecting defective regions inside the textured surfaces with roughly annotated labels. The real-time data augmentation approach are used for training phase, which significantly improves the model generalization. The great advantageous of the proposed model is its high capability to analyze and work with weakly annotated data since in many industrial applications providing high-quality ground-truth labels is not feasible and practical. In case of achieving the optimum efficiency-accuracy trade-off, the model can be further improved. We leave this for the future research.

REFERENCES

[1] Yazdchi, M., Yazdi, M. and Mahyari, A.G., 2009, March. Steel surface defect detection using texture segmentation based on multifractal dimension. In 2009 International Conference on Digital Image Processing (pp. 346-350). IEEE.

[2] Nand, G.K. and Neogi, N., 2014, December. Defect detection of steel surface using entropy segmentation. In 2014 Annual IEEE India Conference (INDICON) (pp. 1-6). IEEE.

[3] Wang, J., Li, Q., Gan, J., Yu, H. and Yang, X., 2019. Surface Defect Detection via Entity Sparsity Pursuit with Intrinsic Priors. IEEE Transactions on Industrial Informatics, 16(1), pp.141-150.

[4] Ai, Y.H. and Xu, K., 2013. Surface detection of continuous casting slabs based on curvelet transform and kernel locality preserving projections. Journal of Iron and Steel Research International, 20(5), pp.80-86.

[5] Liu, W. and Yan, Y., 2014. Automated surface defect detection for cold-rolled steel strip based on wavelet anisotropic diffusion method. International Journal of Industrial and Systems Engineering, 17(2), pp.224-239.

[6] Mukherjee, A., Chaudhuri, S., Dutta, P.K., Sen, S. and Patra, A., 2006. An object-based coding scheme for frontal surface of defective fluted ingot. ISA transactions, 45(1), pp.1-8.

[7] Timm, F. and Barth, E., 2011, February. Non-parametric texture defect detection using Weibull features. In Image Processing: Machine Vision Applications IV (Vol. 7877, p. 78770J). International Society for Optics and Photonics.

[11] Zeiler, M.D., 2012. Adadelta: an adaptive learning rate method. arXiv preprint arXiv:1212.5701.

[8] Luo, Q., Fang, X., Liu, L., Yang, C. and Sun, Y., 2020. Automated Visual Defect Detection for Flat Steel Surface: A Survey. IEEE Transactions on Instrumentation and Measurement.

[9] Ronneberger, O., Fischer, P. and Brox, T., 2015, October. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention (pp. 234-241). Springer, Cham.

[10] Long, J., Shelhamer, E. and Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3431-3440).

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

[13] https://resources.mpi-inf.mpg.de/conference/dagm/2007/prizes.html

[14] Tao, X., Zhang, D., Ma, W., Liu, X. and Xu, D., 2018. Automatic metallic surface defect detection and recognition with convolutional neural networks. Applied Sciences, 8(9), p.1575.

[15] Natarajan, V., Hung, T.Y., Vaikundam, S. and Chia, L.T., 2017, March. Convolutional networks for voting-based anomaly classification in metal surface inspection. In 2017 IEEE International Conference on Industrial Technology (ICIT) (pp. 986-991). IEEE.

[16] Zhou, F., Liu, G., Xu, F. and Deng, H., 2019. A Generic Automated Surface Defect Detection Based on a Bilinear Model. Applied Sciences, 9(15), p.3159.

[17] Cha, Y.J., Choi, W., Suh, G., Mahmoudkhani, S. and Büyüköztürk, O., 2018. Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types. Computer-Aided Civil and Infrastructure Engineering, 33(9), pp.731-747.

[18] Li, J., Su, Z., Geng, J. and Yin, Y., 2018. Real-time detection of steel strip surface defects based on improved yolo detection network. IFAC-PapersOnLine, 51(21), pp.76-81.

[19] Ren, R., Hung, T. and Tan, K.C., 2017. A generic deep-learning-based approach for automated surface inspection. IEEE transactions on cybernetics, 48(3), pp.929-940.

[20] Racki, D., Tomazevic, D. and Skocaj, D., 2018. The effect of different CNN configurations on textured-surface defect segmentation and detection performance. In 23rd Computer Vision Winter Workshop.

[21] Qiu, L., Wu, X. and Yu, Z., 2019. A high-efficiency fully convolutional networks for pixel-wise surface defect detection. IEEE Access, 7, pp.15884-15893.

[22] Yu, Z., Wu, X. and Gu, X., 2017, July. Fully convolutional networks for surface defect inspection in industrial environment. In International Conference on Computer Vision Systems (pp. 417-426). Springer, Cham.

[23] Mei, S., Yang, H. and Yin, Z., 2018. An unsupervised-learning-based approach for automated defect inspection on textured surfaces. IEEE Transactions on Instrumentation and Measurement, 67(6), pp.1266-1277.

[24] Yang, H., Chen, Y., Song, K. and Yin, Z., 2019. Multiscale feature-clustering-based fully convolutional autoencoder for fast accurate visual inspection of texture surface defects. IEEE Transactions on Automation Science and Engineering, 16(3), pp.1450-1467.

[25] Springenberg, J.T., Dosovitskiy, A., Brox, T. and Riedmiller, M., 2014. Striving for simplicity: The all convolutional net. arXiv preprint arXiv:1412.6806.

[26] Ioffe, S. and Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.