

Received March 18, 2019, accepted April 13, 2019, date of publication April 29, 2019, date of current version June 10, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2913620

Fabric Defect Detection Using Activation Layer Embedded Convolutional Neural Network

WENBIN OUYANG, BUGAO XU^{ID}, JUE HOU, AND XIAOHUI YUAN^{ID}

Department of Computer Science and Engineering, University of North Texas, Denton, TX 76203, USA

Corresponding author: Bugao Xu (bugao.xu@unt.edu)

This work was supported by the Walmart U.S. Manufacturing Innovation Fund under Grant GP20013.

ABSTRACT Loom malfunctions are the main cause of faulty fabric production. A fabric inspection system is a specialized computer vision system used to detect fabric defects for quality assurance. In this paper, a deep-learning algorithm was developed for an on-loom fabric defect inspection system by combining the techniques of image pre-processing, fabric motif determination, candidate defect map generation, and convolutional neural networks (CNNs). A novel pairwise-potential activation layer was introduced to a CNN, leading to high accuracy of defect segmentation on fabrics with intricate features and imbalanced dataset. The average precision and recall of detecting defects in the existing images reached, respectively, over 90% and 80% at the pixel level and the accuracy on counting the number of defects from a publicly available dataset exceeded 98%.

INDEX TERMS Convolutional neural network, activation function, fabric defects, imbalanced dataset.

I. INTRODUCTION

A faulty mechanical motion or a yarn breakage on a loom can cause the weave structure to differ from the design, yielding a warp, weft, or point defect, such as harness misdraw, end-out, mispick, and slub. Defects can reduce fabric price by 45% to 65% [1]. In the modern weaving factories, weavers are required to check the fabric in weaving for intricate defects by cruising a number of looms periodically because some of the fabric defects are preventable or correctable if detected on time. Hence, the textile industry has been moving toward automating fabric inspection for consistent evaluation of fabric quality. Compared to the 60–75% defect detection accuracy of human visual judgment [2], a typical state-of-art automatic fabric inspection system can achieve a detection rate up to 90% [3]–[5].

Most auto-fabric inspection systems are based on computer vision techniques including image acquisition and defect segmentation algorithms. The fabric defect detection algorithms can be categorized into statistical, spectral, model-based, learning, structural, and hybrid approaches. The first five approaches for defect segmentation were reported to be sensitive to noise [6], [7], computationally intensive [7]–[9], limited to certain types of defects [7], [10]–[13], and

The associate editor coordinating the review of this manuscript and approving it for publication was Li He.

inconsistent to changes in fabric structures and background [4]. For the past decade, the hybrid approach has been adopted for higher robustness in handling variations in weave structures and defect types. Concepts from other fields, i.e., Bollinger Bands (BB) – a statistical chart of a financial instrument, and Elo rating (ER) – an evaluation method of player performance, have also been introduced to the fabric defect detection. Although BB and ER achieved a detection rate above 96% on patterned fabrics, they failed to detect defects smaller than the repetitive unit of a patterned fabric [14], [15].

Recently, convolutional neural networks (CNN) have been demonstrated for effective image semantic segmentation [16]. The CNNs, e.g. FCN [16], U-Net [17], SegNet [18], and their successors, all share the basic components—convolution, pooling, and activation functions, in which the pooling layer plays a role of avoiding overfitting and reducing the spatial dimensions. A deep network could reach over a hundred convolutional layers. For example, the VGGNet [19] has 16 layers, while the ResNet [20] possesses 152 layers. However, deep networks provide features with a global semantic meaning and abstract details that are not suitable for fine structure segmentation in an image, mainly because (1) the traditional convolutional filters have large receptive fields, and (2) fine structures are further reduced by pooling layers [21]. From the perspective of

computer vision, most fabric defects are considered as fine structures since they are indicated only by a small number of pixels in an image. To improve the fine structure segmentation, further processing is needed to be embedded in a CNN to fine tune the coarse outputs of the CNN.

Another common problem in the neural network learning process is that samples from a real-life application may not always evenly distribute among classes. Applications, such as medical diagnosis, credit fraud detection, computer vision, etc., are brimming with imbalanced datasets. The methods of handling an imbalanced dataset can be categorized into data-level, algorithmic-level and hybrid approaches [22]. As of the data-level approach, over-sampling and under-sampling are the common strategies to adjust the class distribution of a dataset. A synthetic minority oversampling technique (SMOTE) allows samples to be randomly created based on the density distribution [23]. The algorithmic-level approach assigns different cost values to minority and majority samples [24]. It is difficult, however, for such a cost-sensitive approach to unify a general framework since it is often specific to a paradigm. AdaBoost algorithms [25] bridged the sampling and cost-sensitive approaches together by iteratively updating cost weights. Yuan et al. introduced a regularization term in the SAMME algorithm [26] to penalize the weight of a classifier that misclassifies the second-round-misclassified examples [27].

A defect appearance on a fabric image, such as double pick, mispick, coarse pick, etc., usually represents a fine/detail surface structure and often occupies less than 35% of the pixels of the image, yielding an imbalanced fabric defect dataset. Therefore, in the construction of a CNN for fabric defect detection, two factors need to be considered: preservation of fine structures and treatment of imbalanced dataset. A desired CNN should not involve too many convolutional layers to prevent details on a fabric image from missing, and should avoid using pooling layers to retain the image resolution in the feature maps.

In this paper, we adopt a hybrid approach that utilizes statistical defect information and a CNN for fabric defect detection. The motif of a fabric is firstly calculated using the autocorrelation of a fabric image to represent the repeated texture in the fabric. A motif-center-point map, namely node map, is then generated by normalized cross-correlation (NCC) taking the motif as the template. The distributions of node points can indicate the regularity of fabric textures in the image, from which a statistical rule can be derived to relate the node point count in a motif region to the defect judgment. The statistical rule can be utilized as an activation reference, called pairwise potential activation layer, in a newly designed CNN to improve the fabric defect detection performance.

II. RELATED WORK

A. ACTIVATION LAYER IN CNN

To identify an object, a human brain tries to pick up useful information, such as shape, color, smell, feeling, and prior knowledge. Among these features, prior knowledge can

significantly influence the success rate and the response speed of object recognition [28]. In a similar way, a convolutional neural network selects only suitable ones from a tremendous amount of generated features for object identification.

Previous research utilized general activation functions to regulate the contributions of different features. The binary activation function is the primitive way to filter features, but that the zero gradient function does not improve the network evolution. The linear activation function has a similar problem because the constant value of its derivative function cannot reduce the error during the backpropagation. Non-linear activation functions, i.e. Sigmoid and Tanh, achieve breakthroughs from linear functions. However, learning updates can be trapped in the $>+3$ or <-3 regions when the gradients curves of Sigmoid and Tanh functions approach to zero. The ReLU activation function makes the network sparse and efficient because the coefficient of a negative neuron is always zero, yielding dead neurons which are never activated. If too many dead neurons are generated by the ReLU activation, the surviving neurons in the network may not be powerful enough to recognize objects correctly. An alternative way is to use the Leak ReLU or PReLU to avoid dead neurons. Since our research is only associated with binary detection (defect or non-defect), the multi-group classification activation functions, i.e. Softmax, are not discussed in this work.

The training process of a neural network can be easily entered in a gradient vanishing situation if Sigmoid or Tanh activation function is used in a CNN, because the expression of the gradient is,

$$\Delta W_1 = \frac{\partial Loss}{\partial w_1} = \delta'(z_1) \delta'(z_2) \dots \delta'(z_n) \frac{\partial Loss}{\partial w_n} \quad (1)$$

where ΔW is the weight offset, and δ' is the derivative of the activation function. Since the Sigmoid derivative reaches a maximum at $\delta'(0) = 0.25$, according to the derivative chain rule, with each subsequent layer the magnitude of gradient vanishes exponentially. Tanh activation scales the maximum δ' up to one, but the vanishing gradient problem still exists because a δ' is highly possible below one. It is also possible that an imbalanced dataset causes δ' to cluster in either right or left side of the gradient curve where δ' is always less than 1. A large weight may help to avoid the gradient vanishing problem.

When ReLU or its improved version is chosen, the fluctuating loss phenomena can cause the learning process to converge more difficultly on an imbalanced dataset. Therefore, a specially designed activation function is needed in the CNN.

B. FINE STRUCTURE SEGMENTATION ON CNNS

Different approaches were proposed to improve the segmentation of fine structures. Firstly, the method of retrieving the features from earlier layers was presented in [29] and [30] to better estimate fine structures, such as boundary, hollow

area, etc. An alternative way was to conduct a super-pixel representation of an original image to enhance the localized details [31]. The main drawback of this strategy is that it leads to a poor prediction if wrong features are retrieved in the very beginning.

Secondly, a nonlinear model was applied to produce accurate semantic segmentation based on a label map. The nonlinear model can be a support vector machine (SVM) [32], a random forest [33], or a Conditional Random Field (CRF) [34]. The DeepLab is the pioneer that utilized the CRF as a post-processing procedure after a CNN. The DeepLab treated the prediction of a CNN as the unary potential and took the generated energy map as the pairwise potential to form a CRF presentation. Because the potentials of a CRF integrate the prior probability, the pairwise potential, and the Gaussian smooth term that encourages similar pixels having similar posterior [35], [36], a CRF is able to assist recovering the details of the CNN output. Although the CRF post-processing significantly improves the fine structure segmentation in an image, it does not fully take advantage of the strength of a CRF since it is isolated from the CNN learning process. It needs to be noted that a simple combination of a CNN and a CRF may not be an optimal solution since it wastes many features generated by the CNN because of limited latent features in CRF. Zheng et al. formulated dense CRF as a Recurrent Neural Network (RNN) so that the CRF energy could be calculated during the CNN training [21], but its performance relied on the recurrently computed forward pass, which was time-consuming [37].

III. METHODS

A. DEFECT PROBABILITY MAP GENERATION

The autocorrelation measurement of an image $I(i, j)$, $i \in (0, N), j \in (0, M)$, is to shift the image in the vertical and horizontal direction with different scales, which the function is described as,

$$\rho(x, y) = \frac{\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} I(i, j)I(i+x, j+y)}{\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} I(i, j)^2} \quad (2)$$

where x and y represent the image shifting scales in the horizontal and vertical directions. The efficiency of this calculation can be improved by the Fast Fourier Transform (FFT) in (3),

$$\rho(x, y) = \frac{ABS(IFT2(FFT2(I) . * Conj(FFT2(I))))}{\sum_{i=0}^{N-1} \sum_{j=0}^{M-1} I(i, j)^2} \quad (3)$$

Fig.1a is a fabric image containing defects, Fig. 1c illustrates its autocorrelation map, and Figs. 1d and 1e are the side views of the autocorrelation map in the warp and the filling directions. The average period intervals are about 7 pixels in the warp direction, and about 16 pixels in the filling direction, which define the size of the weave repeating unit (fabric motif). The fabric motif can be generated by averaging pixels in every 7×16 non-defective area. In this

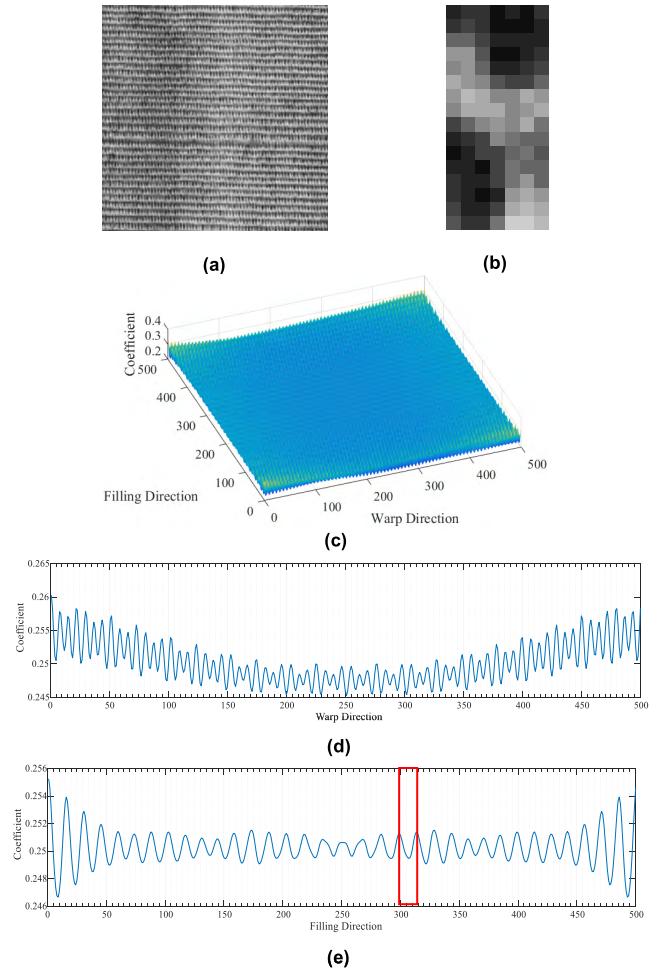


FIGURE 1. (a) A fabric image containing defects. (b) The generated fabric motif. (c) 3D view of the autocorrelation coefficient map. (d) Side view of the autocorrelation map in the warp direction. (e) Side view of the autocorrelation map in the filling direction.

research, if either the horizontal peak interval or the vertical interval is 3 pixels above the corresponding averaged interval, this area will be excluded from the fabric motif calculation. The calculated fabric motif of Fig. 1a is shown in Fig. 1b. The red -box in Fig. 1e indicates a defect region in Fig. 1a. However, the approximate 16-pixel peak-to-peak interval of the defect region is not distinguishable with other regular fabric texture. Therefore, further processing is needed.

Using the calculated motif image (Fig.1b) as a template, a fabric motif map of the image (Fig.1a) can be generated by calculating its localized correlations with the template. Compared to the sum of absolute difference (SAD), the sum of squared differences (SSD), and the hamming distance (SHD), the normalized cross-correlation (NCC) proves to be more robust for calculating motif center points. To eliminate the intensity differences between the template and the image, the mean values should be subtracted. Therefore, the zero-mean NCC (ZNCC) is used for the fabric motif map generation. The calculation of the ZNCC on an image

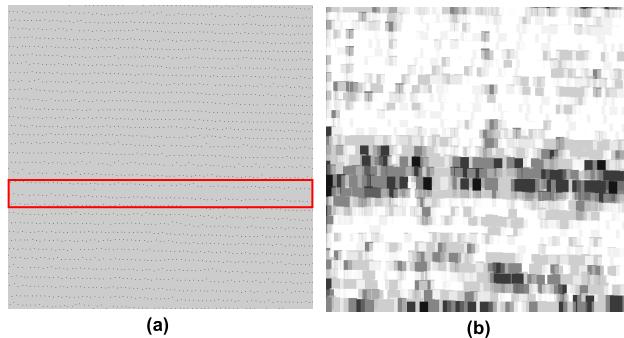


FIGURE 2. (a) The node point map of Fig. 1a. (b) The generated defect probability map.

can be presented as the following equation,

$$\begin{aligned} f(u, v) &= \frac{n(u, v)}{d(u, v)} \\ &= \frac{\sum_{i,j} \{[I(i, j) - \bar{I}] [t(i - u, j - v) - \bar{t}]\}}{\sqrt{\sum_{i,j} [I(i, j) - \bar{I}]^2} \sum_{i,j} [t(i - u, j - v) - \bar{t}]^2} \end{aligned} \quad (4)$$

where $t(i, j)$ is a pixel in the template, $I(i, j)$ is a pixel in the image. The nominator in (4) can be rearranged according to Lewis [38],

$$n(u, v) = \sum_{i,j} I(i, j) t'(i - u, j - v) \quad (5)$$

Equation (5) is a convolutional expression in the time domain. In the frequency domain, it can be expressed by more efficient dot multiplication:

$$n(u, v) = \text{ABS}(\text{IFFT2}(\text{FFT2}(I) . * \text{Conj}(\text{FFT2}(t')))) \quad (6)$$

The denominator can be rewritten as

$$\sqrt{\left[\sum_{i,j} I(i, j)^2 - \frac{1}{N} \left(\sum_{i,j} I(i, j)\right)^2\right] \cdot \sigma \cdot \sqrt{N}} \quad (7)$$

where N is the total pixel count in an image and σ is the standard deviation of the calculated template. By combining (6) and (7), the correlation coefficient can be rearranged as follows,

$$\begin{aligned} f(u, v) &= \frac{n(u, v)}{d(u, v)} \\ &= \frac{\text{ABS}(\text{IFFT2}(\text{FFT2}(I) . * \text{Conj}(\text{FFT2}(t'))))}{\sqrt{\left[\sum_{i,j} I(i, j)^2 - \frac{1}{N} \left(\sum_{i,j} I(i, j)\right)^2\right] \cdot \sigma \cdot \sqrt{N}}} \end{aligned} \quad (8)$$

Fig. 2a demonstrates the motif-center-point detection result of Fig. 1a, namely fabric node map. The map eliminates the original fabric weave pattern but inherits the fabric defect information, indicated by a wider horizontal gap as shown in the red box in Fig. 2a. According to the fact that the count of node points in a motif area should be constant in a regular fabric, a statistical rule to determine the probability of defect areas can be derived. On a node map, each node

TABLE 1. Defect probability regarding with different numbers of nodes in nodes-searching region.

# of nodes	1	2	3	4	5
Defect probability	0.927	0.788	0.516	0.252	0.059
# of nodes	6	7	8	9	
Defect probability	0	0.122	0.028	0.003	

represents the center of a region whose dimension is the same as the template. Taking the ground truth images as references, we were able to find the probability of a defect area that is related to the number of nodes found in each node-searching region. After investigating the 1160 node maps generated from our fabric image dataset, we created a table showing defect probability Table 1 regarding the number of nodes found in each region. It is found that the probability of being a defect region is high if the number of nodes in the region is fewer than four. Fig. 2b shows the generated defect probability map according to Table 1. In the image, dark regions represent higher chances of defect areas. Despite the edge areas in Fig. 2b, the defect areas in the original image do show higher probabilities. But there are exceptions that some areas with high defect probability in Fig. 1a are actually defect-free areas, meaning that a further step is needed to correct the defect probability map. The map will be used as a pairwise potential activation layer in a convolutional neural network.

B. PAIRWISE POTENTIAL ACTIVATION LAYER

Unlike the prediction on a standalone sample by traditional discrete classifiers, a Conditional Random Field (CRF) considers sample's neighbors to be a random variable distribution in an undirected graphical model. Of an image, a single pixel is meaningful only if its neighbors is taken into account. Therefore, the application of a CRF can be extended to the image segmentation, in which random fields describe the correlations among different pixels sharing similar properties. Let's denote an image as I , which has a pixel vector $X = \{x_1, x_2, \dots, x_n\}$ and a corresponding label set $L = \{l_1, l_2, \dots, l_n\}$. According to the Hammersley-Clifford theorem, CRF obeys the Gibbs distribution [38], that is,

$$P(X = x | L) = \frac{1}{Z(L)} \exp(-E(x | L)) \quad (9)$$

where the $Z(L)$ is the partition function and $E(x | L)$ represents the pixel's energy with the following expression,

$$E(x) = \Psi_u(x_i) + \Psi_p(x_i, x_j) \quad (10)$$

where x_i and x_j are two pixels, Ψ_u is the unary potential outcome of a feature extracting network (i.e. CNN), and Ψ_p is the pairwise potential representing the relationship among pixels in cliques that have similar characteristic, such as gradient,

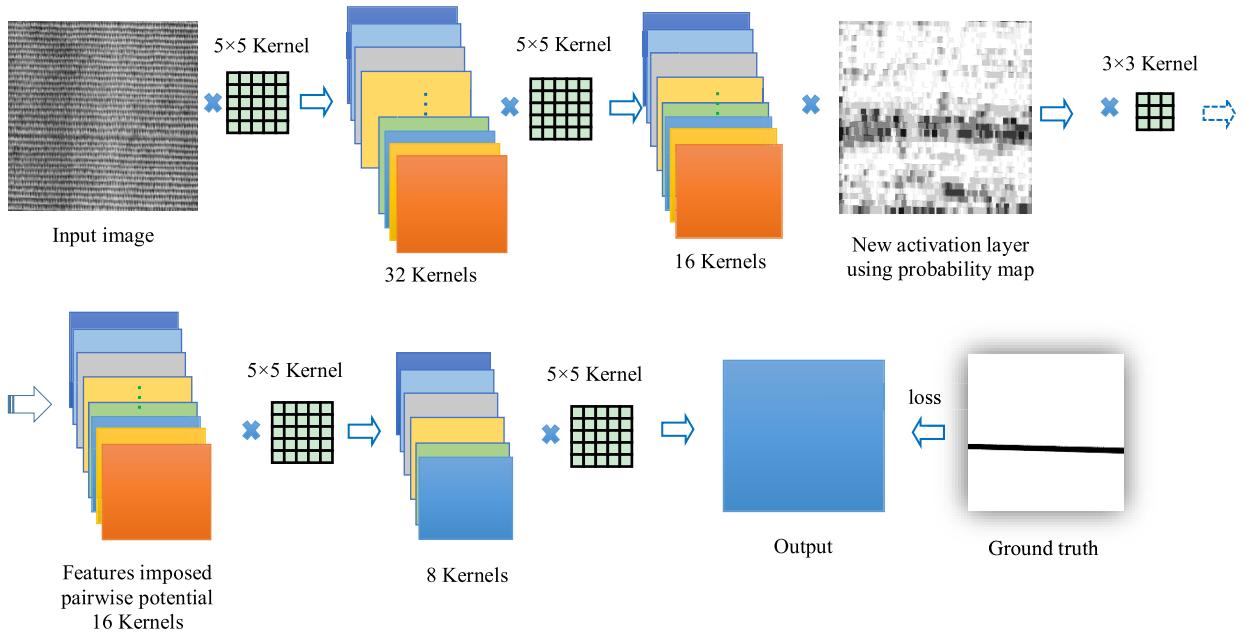


FIGURE 3. The architecture of the new convolutional neural network.

texture, or color, preferably, with the same label. The unary potential, called the rough term in the CRF, describes the CNN's output that is an abstract segmentation result. The pairwise, in contrary, provides the smoothing term that takes the effect by adjusting the pixel labels according to the discovered similar properties among pixels. The CRF has been demonstrated to be effective for improving detail recognition after the CNN segmentation [34].

Pairwise potential Ψ_p can be further expanded to [10],

$$\Psi_p(x_i, x_j) = \mu(x_i, x_j) \sum_{m=1}^K \omega^{(m)} k^{(m)}(f_i, f_j) \quad (11)$$

where f is an image feature derived from pixels information (color, intensity, or gradient), $k^{(m)}$ is a feature filter that may be a Gaussian, high-pass, or low-pass filter, $\mu(\cdot, \cdot)$ is a label function describing the compatibility between pixel labels, and $\omega^{(m)}$ is the regularization term. In the statistical mechanical ensemble, Eq. (11) can be explained as a partition function which attempts to maintain constant temperature and pressure. Here, the pairwise potential also aims to keep cluster cliques with similar properties. By bringing the logarithm to Eq. (11), $\ln(\Psi_p(x_i, x_j))$ is the characteristic state function called the Gibbs free energy. To get a similar logarithm expression on the unary term of Eq. (10), $\Psi_u(x_i)$ can be changed to $\ln(e^{\Psi_u(x_i)})$. A new energy function can be written as,

$$\begin{aligned} \hat{E}(x) &= \ln(e^{\Psi_u(x_i)}) + \ln(\Psi_p(x_i, x_j)) \\ &= \ln(e^{\Psi_u(x_i)} \times \Psi_p(x_i, x_j)) \end{aligned} \quad (12)$$

where $e^{\Psi_u(x_i)} \times \Psi_p(x_i, x_j)$ is a function similar to the activation process in a CNN in which $\Psi_u(x_i)$ and $\Psi_p(x_i, x_j)$

represent the feature map and the activation function, respectively. Unlike a traditional CRF, which is normally attached to the end of a CNN, the pairwise activation function can be inserted in any position in an end-to-end network whose kernels can be updated during the learning process. In this application, the defect probability map can be used as an activation map in the CNN because,

- (1) in the defect probability map, pixels with similar properties (fabric motif) have lower defect probabilities.
- (2) the defect probability map illustrates the defect distribution of an image, which can be used to replace the $k^{(m)}$ filter.
- (3) the defect probabilities are calculated according to the motif node distributions in a larger region, representing a cumulative relationship among different pixels in a clique. This is the same concept as the pairwise potential, $\Psi_p(x_i, x_j)$.

C. NETWORK ARCHITECTURE

Figure presents a CNN for the fabric defect detection. To achieve state-of-art detection result, the fabric defect probability map is introduced to the network as a dynamic activation layer, namely pairwise potential activation layer (PPAL). The probability map, which contains the prior knowledge or defect statistical rules, is critical to the judgment of the probable defect areas. The new 7-layer CNN includes,

- (1) the original image input layer.
- (2) the first hidden layer – the first convolutional layer with $32 \times 5 \times 5$ kernels.
- (3) the second hidden layer – another convolutional layer with $16 \times 5 \times 5$ kernels.

- (4) the pairwise potential activation layer. Instead of using activation function, each feature in the output of the previous convolutional layer is multiplied by this specific activation map with one 3×3 convolutional kernel, which takes more pixels' properties into consideration.
- (5) the fourth hidden layer with $8 \times 5 \times 5$ kernels to convolute the 16 prior knowledge (statistical rules) imposed feature maps.
- (6) the last hidden layer – single 5×5 kernel is used to generate the output image.
- (7) backpropagation with loss calculation is performed by taking ground truth images as references.
- (8) the middle position insertion of the PPAL is to ease off the influence of the probability map. If the PPAL is inserted at the output layer, the output will be similar to the probability map, which weakens the convolutional features.

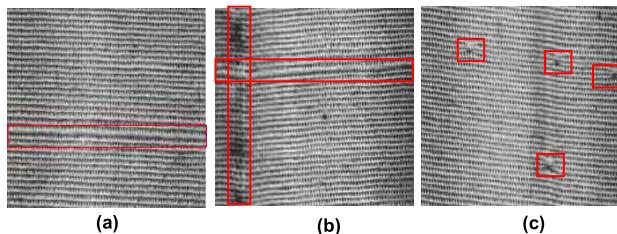


FIGURE 4. Sample fabric defect images. (a) Horizontal defects. (b) Vertical and horizontal defects. (c) Isolated defects.

IV. EXPERIMENTS

A. DATASETS

To evaluate the effectiveness of the proposed PPAL convolutional neural network (PPAL-CNN), we created a fabric defect dataset using our designed on-loom fabric imaging system. The dataset contains 1160 fabric images of 500×500 pixels, which include vertical defects, horizontal defects, isolated defects, and defect-free fabrics, as shown in Fig. 4. In order to provide the ground truth information for the CNN training, each image was manually inspected to mark defects areas. We also used the TILDA—a Textile Texture Database developed by the texture analysis group of the DFG (Deutsche Forschungsgemeinschaft) as a verification set. Each TILDA image has a text description about defect areas in the image.

B. EVALUATION ON DIFFERENT ACTIVATION FUNCTIONS IN NEURAL NETWORKS

A visual comparison on loss curves was performed among four CNNs that have the same structure with different activation functions, i.e. Sigmoid, Tanh, ReLU, and PPAL. In order to avoid small gradient issues in the flat regions of Sigmoid and Tanh activation functions, the cross-entropy loss function was chosen. Two different learning rates, 10^{-10} (slow) and 10^{-4} (fast), were used to check if the learning rate influenced the network learning process. Fig. 5 lists the loss curves

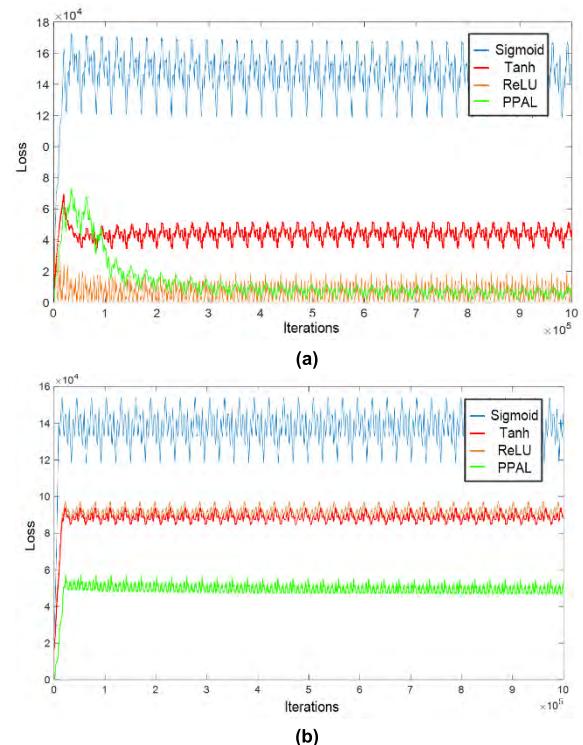


FIGURE 5. Loss curves of Sigmoid, Tanh, ReLU and PPAL activation functions under learning rates of 10^{-4} (a) and 10^{-10} (b).

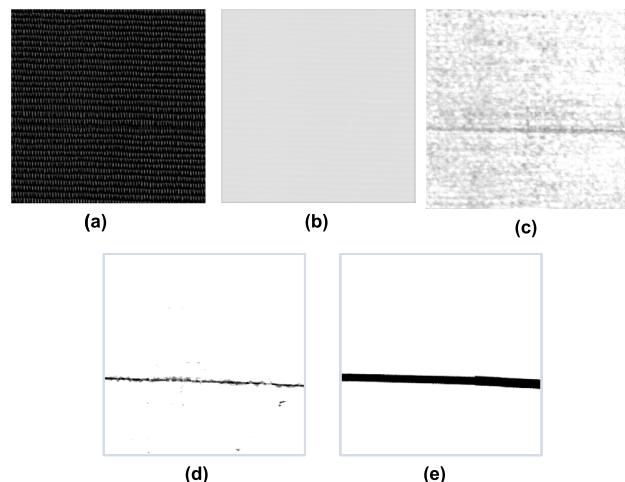


FIGURE 6. Fabric defect detection result on Fig.4a under four activation functions. (a) Sigmoid. (b) Tanh. (c) ReLU. (d) PPAL. (e) Ground Truth.

calculated from the output layer during the 10^6 learning iterations of the networks with the four different activation functions under the two learning rates. Regardless of learning rate, the three activation functions, Sigmoid, Tanh and ReLU, show fluctuation, and the network with Sigmoid or Tanh activation function is not convergent. ReLU loss curve at the 10^{-4} learning rate has a sudden decrease after the first-thousand iterations. This may be because the fast learning rate causes the learning to be trapped in the local minimum. The loss curve of PPAL-CNN at the 10^{-4} learning rate depicts

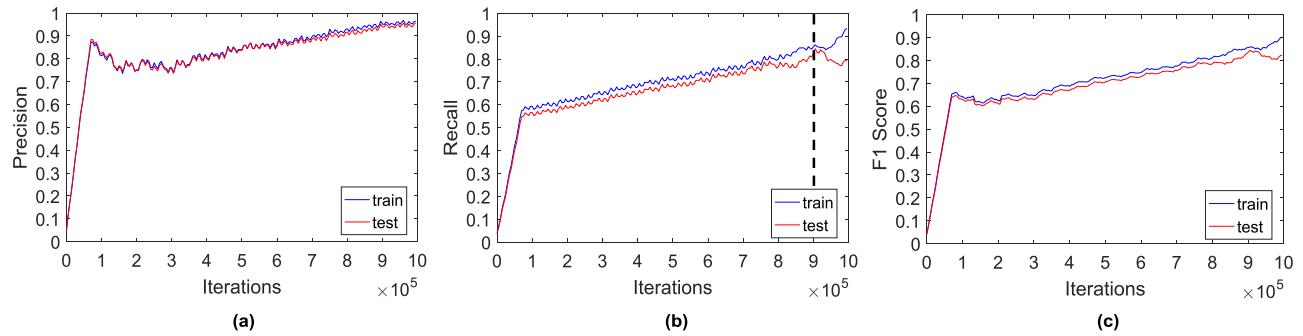


FIGURE 7. Precision, recall, and F1-score evaluating curves during the training epochs (5000 – 10⁶ iterations) using training and testing dataset. (a) Precision. (b) Recall. (c) F1 score.

a normal learning process that starts converging after 2×10^5 iterations (Fig. 5a). However, the same network seems difficult to be convergent at the 10^{-10} learning rate (Fig. 5b). Thus, the 10^{-4} learning rate was chosen for the PPAL-CNN in the fabric defect detection.

Fig. 6 displays the prediction results of Fig. 4a using the CNNs that have the same structures with the four aforementioned activation functions of the 10⁶th training-iteration-model at the 10^{-4} learning rate. In comparison with the ground truth (Fig. 6e), the predicted result of each activation function is consistent with the loss curves depicted in Fig. 5. Among the four activation functions, Sigmoid appears to be the worst because of the highest loss value, and PPAL-CNN demonstrates the best defect detection result.

C. EVALUATION ON FABRIC DEFECT DATASET AT PIXEL LEVEL

A 4-folder cross-validation was performed on the 1160 fabric images. Since the accuracy is measured by true positive and true negative samples, it will not be suitable for our imbalanced fabric defect dataset. Therefore, the 3-metrics, precision, recall and F1-score, that are derived from the confusion matrix are applied to evaluate the detection accuracy at the pixel level. The precision represents the rate of the correctly detected defect pixels over all the predicted defect pixels. The recall is the ratio between correctly detected defect pixels and defect pixels marked in the ground truth image, which represents the integrity of the correctly detected defect region. The F1-score is the harmonic average of precision and recall. Fig. 7 illustrates the curves of the three metrics of the training set (blue) and the testing set (red) from 5000 to 10⁶ iterations. Overall, the three metrics monotonically increase with the training iterations, indicating an ascending defect detection accuracy. In the recall chart (Fig. 7b), the training curve is slightly above the testing curve before the 9×10^5 th iteration. At the 9×10^5 th iteration, the testing curve tends to fall, suggesting a convergent point of the training process. Although the precision of the detected defects keeps improving as the training epoch increases, the small drop of the recall (integrity) at the 9×10^5 th iteration indicates the reductions in both false positive and true positive defect pixels, which is

a sign of overfitting. Therefore, we chose the 9×10^5 th iteration model in the following evaluations.

We trained two sophisticated image segmentation deep neural networks, U-net and SegNet, with the 1160-fabric-image dataset. Without a pre-trained edge energy map of an image, U-net is not suitable for the fabric defect detection.

TABLE 2. Quantitative comparison between the proposed PPAL and SegNet.

	PPAL		SegNet	
	Training	Testing	Training	Testing
Precision	0.9653	0.9292	0.4500	0.3998
Recall	0.8636	0.8513	0.4433	0.4137
F1-Score	0.9117	0.8885	0.4696	0.4235

TABLE 3. Defect detection results of Fig. 4.

	Original image	Ground truth	SegNet	PPAL
1				
2				
3				

Table 2 presents a quantitative comparison between PPAL-CNN and SegNet at the 9×10^5 th iteration. We adopted the basic SegNet network that has 4 encoder and 4 decoder blocks, but the cross-entropy was used to replace the Softmax loss function for the fabric defect detection. The detection results of SegNet suggest that fine defects whose width are

TABLE 4. Quantitative comparison among different proportions of defect pixels.

Proportion of defect pixels	0% (Defect-free)	0-5%	5-10%	10-15%	15-20%	20-25%	25-30%	30-35%	35-40%	40-45%	45-50%
Image count	432	267	166	76	40	36	24	26	11	13	16
Precision	0.9564	0.8568	0.9292	0.9092	0.9370	0.9794	0.9866	0.9884	0.9851	0.9692	0.9544
Recall	0.9564	0.7799	0.7596	0.7726	0.7782	0.7346	0.7954	0.8401	0.9073	0.9193	0.8514
F1-Score	0.9564	0.8166	0.8359	0.8353	0.8503	0.8395	0.8807	0.9082	0.9446	0.9436	0.8993
Proportion of defect pixels	50-55%	55-60%	60-65%	65-70%	70-75%	75-80%	80-85%	85-90%	90-95%	95-100%	
Image count	10	9	4	14	8	3	2	1	2	0	
Precision	0.9767	0.9515	0.9828	0.9841	0.9733	0.9925	0.9959	1	0.9976	N/A	
Recall	0.9071	0.8337	0.9027	0.8738	0.9023	0.9426	0.9658	0.9123	0.9412	N/A	
F1-Score	0.9406	0.8887	0.9410	0.9257	0.9365	0.9669	0.9806	0.9541	0.9686	N/A	

below 20 pixels fail to be detected in both the training and the testing datasets, and defects with sizes above 20 pixels can be partially detected, which result in low recall scores in Table 2. The lower precision scores of SegNet are caused by the false detection of defects in non-defect areas.

Table 3 illustrates the defect detection results of Fig. 4 by SegNet and PPAL-CNN. PPAL-CNN is able to detect the horizontal defects and the isolated defects correctly, but it partially detects vertical defects in image 2. By reexamining the original image 2, the vertical defect area on the ground truth seems to be over-estimated. Visually, the detected defect areas of PPAL-CNN are closer to visual judgments. SegNet is not able to detect the thin or tinny size defects. The vertical defects can be found partially in SegNet, but the white noise appearing in SegNet detection results suggests that SegNet is not a proper neural network for the fabric defect detection.

D. EVALUATION ON THE IMBALANCED DATASET WITH DIFFERENT PROPORTIONS OF DEFECT PIXELS

According to the proportions of the defect pixels in the fabric ground truth images, we clustered the 1160 fabric images into 21 groups. Group #1 indicates defect-free images, and groups #2 - #20 represent the proportion of defect pixels with 5% increment. Among the 1160 images, there are 432 defect-free images and 635 images with defect pixel proportions being less than 35%. In other words, 92% of the fabric images in the dataset have small proportion or none of defect pixels. Those defect regions above 35% could be the fabric selavage regions. The highest proportion (95-100%) of defects does not exist in the dataset.

Table 4 presents the 3-metrics (precision, recall, and F1-score) comparisons among the 21 groups. The precision scores appear stable (all above 0.97) for the defect

proportions above 20% but have a slight decrease (0.9) within the 0-20% range. This reveals that the correctly detected pixels of fine defects are lower than those bulk defects, i.e. the selavage area of the cloth. The recalls of the 0-30% defect proportions are lower than 0.8 which indicate the less defect detection integrity on fine defects. The 3-metrics scores with 0.96 in the defect-free group demonstrate the less false alarm on defect-free fabrics of PPAL-CNN.

E. EVALUATION ON DETECTION ACCURACY ACCORDING TO DEFECT COUNT

In the 1160-fabric-image dataset, totally 1191 fabric defects were found in 728 fabric images. We inspected each predicted defect image according to its corresponding ground truth image. If the predicted defect pixels occupied over 50% in the bounding box of the defect area in the ground truth image, the prediction was considered to be correct. Therefore, according to the defect counts in the dataset, 1177 of 1191 defects were correctly detected, yielding a 98.82% detection accuracy.

It is found that there were 51 over counted defects that should be defect-free areas in the 38 ground truth images. In the 38 false predicted fabric images, 13 of them should be defect-free images. Therefore, at the image level, the detection rates of defect-free images, defect images, and total 1160 fabric images are $(432 - 13)/432 = 96.99\%$, $(728 - 25)/728 = 96.57\%$, $(1160 - 38)/1160 = 96.72\%$, correspondingly.

F. EVALUATION ON TILDA DATASET

The TILDA dataset provides 43 sample images, and each of them represents a type of fabric texture. Table 5 presents the defect detection results of the 11 images from TILDA.

TABLE 5. Defect results on sample images from TILDA dataset.

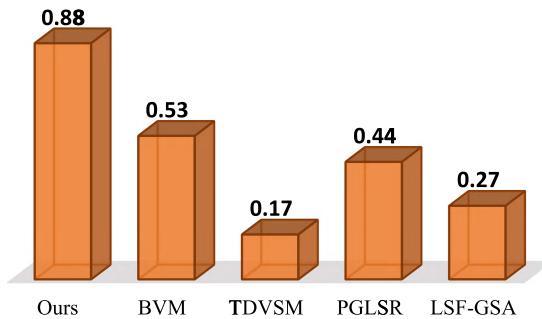
	Original image	Ground truth	PPAL
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			
11			

The 11 sample fabric images have variations in textures, lighting conditions, and defect morphologies. Visually, most detection results show the same defect locations as in the ground truth images, even if the detection integrities of the defects are smaller than the ground truth. Due to the possible fabric texture incompleteness near the image edges, 5-pixel regions around image edges were omitted during the defect segmentation process, which explains the failure defect detection around the bottom area in images 5 and 6 of PPAL-CNN.

TABLE 6. 3-metrics evaluation on TILDA.

Precision	Recall	F1-Score
0.9225	0.8369	0.8776

The results of the 3-metrics evaluation on TILDA are listed in Table 6. The higher precision and the lower recall are in agreements with the visual judgment. Moreover, the recall and precision of the TILDA images are consistent with the previously used 1160 images, which demonstrates the reliability of the proposed algorithm. PPAL-CNN seems able to detect various types of fabric defects on diverse textures under different imaging conditions. Figure 8 illustrates an F1-score comparison among the proposed method and other four state-of-the-art fabric defect detection methods (BVM, TDVSM, PGLSR and LSF-GSA), which demonstrates the superiority of the proposed method. At the image level, the defect detection rate on TILDA was 95.34%.

**FIGURE 8.** F1-score comparison among the proposed method, BVM, TDVSM, PGLSR, and LSF-GSA.

V. DISCUSSION

A. SOURCES OF ERRORS

Since the ground truth images of the two datasets (the 1160-image dataset and the TILDA dataset) were manually generated, possible errors existed when we were tracing the defect areas,

- (1) some tiny defects, such as knots, burl, and short floating thread, may fail to be identified.
- (2) the defect areas may be over-traced or underestimated on some fine vertical or horizontal defects.

- Those inconsistent ground truth tracings may slightly influence the accuracy evaluation.
- (3) to treat fabric selvages as defect areas in this evaluation may decrease the F1-score since it has the chance that some parts of the selvage maintain regular fabric texture.
 - (4) some regular texture areas may be falsely recognized as defect areas because of the image blurring.

B. ANALYSIS OF RESULTS

Analysis on the loss curves with different activation functions is helpful to decide a proper activation function and a better learning rate for the fabric defect detection. The fluctuation on a loss curve or a sudden drop explains the poor defect detection results when only Sigmoid, Tanh, or ReLU activation function was used in a CNN, as shown in Fig. 6. The proper learning rate (10^{-4}) was decided according to the loss curve that presents a normal learning process (green curve in Fig. 5a).

With the increase of training iterations in PPAL-CNN, the precision, recall, and F1-score keep increasing until the 9×10^5 th iteration. However, further training iterations after the 9×10^5 th iteration might not benefit the detection integrity since the possible overfitting would happen. Both quantitative evaluation (Table 2) and visual judgment (Table 3) demonstrate that the performance of PPAL-CNN surpasses those of other CNNs for sophisticated semantic image segmentation.

According to the two dataset evaluation results, PPAL-CNN could detect most of the fabric defects. However, both of the visual and quantitative evaluation results suggest that the defect areas were slightly under predicted when compared with the **ground truth** images, especially for images with defect proportions under 20%. The over-tracing of the defect areas on the ground truth images, such as images 2 and 3 in Table 3 and images 1, 3, 6, 9, and 10 in Table 5, could be the major reason causing the lower recall values. Nonetheless, the proposed PPAL-CNN was able to locate a defect area on a single yarn or in the fabric motif due to the introduction of the fabric defect probability map as a new activation layer of the CNN. The consistency of the 3-metrics (precision, recall, and F1-score) evaluations on both the 1160-images and the TILDA dataset demonstrates that the proposed PPAL-CNN is reliable and robust with various fabric textures and images under different illumination conditions.

VI. CONCLUSION

In this paper, we first applied the autocorrelation to fabric image to determine the fabric motif, and used the motif as a template to generate a fabric motif-center map based on the zero-mean-normalized-cross-correlation. A normal (defect-free) fabric usually has a consistent, uniform motif center arrangement, while a defective fabric has a more random motif-center distribution. According to this observation, a statistical rule to select defect candidates were made by analyzing the motif-center distribution in a fixed

node-searching region, and a defect probability map was generated to be used as prior knowledge of defects in the CNN defect segmentation.

To precisely locate fine structures and handle the imbalanced dataset during the CNN training, we introduced the defect probability map as a dynamic activation layer into the CNN with the pairwise potential function in the CRF.

The proper learning rate (10^{-4}) of the CCN was decided according to the loss learning curves. We evaluated the proposed defect detection algorithm on two fabric datasets—the 1160 images captured in our research lab and the TILDA. The 3-metrics (precision, recall, and F1-score) evaluation results on the two datasets are consistent. The precision values are all above 0.85 with images that have different defect proportions. The recalls, i.e., defect integrity indicator, are slightly lower (about 0.8) when the proportions of the defect pixels are under 30%. An overestimated ground truth might be the major reasons causing the lower recalls. However, if only the presence of defects in an image needs to be judged, the predicted accuracy is above 95%. Regarding the defect counts in the dataset, the proposed PPAL-CNN is able to identify over 98% of the fabric defects correctly.

REFERENCES

- [1] K. Srinivasan, P. H. Dastoor, P. Radhakrishnaiah, and S. Jayaraman, “FDAS: A knowledge-based framework for analysis of defects in woven textile structures,” *J. Textile Inst.*, vol. 83, no. 3, pp. 431–448, 1992.
- [2] M. Patil, S. Verma, and J. Wakode, “A review on fabric defect detection techniques,” *Int. Res. J. Eng. Technol.*, vol. 4, no. 9, pp. 131–136, 2017.
- [3] C.-H. Chan and G. K. H. Pang, “Fabric defect detection by Fourier analysis,” *IEEE Trans. Ind. Appl.*, vol. 36, no. 5, pp. 1267–1276, Sep./Oct. 2000.
- [4] H. Y. T. Ngan, G. K. H. Pang, and N. H. C. Yung, “Automated fabric defect detection—A review,” *Image Vis. Comput.*, vol. 29, no. 7, pp. 442–458, 2011.
- [5] A. Dockery. (2001). *Automated Fabric Inspection: Assessing the Current State of the Art*. Accessed: Jul. 9, 2018. [Online]. Available: www.techexchange.com
- [6] H. Y. T. Ngan and G. K. H. Pang, “Regularity analysis for patterned texture inspection,” *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 1, pp. 131–144, Jan. 2009.
- [7] A. Kumar and G. K. H. Pang, “Defect detection in textured materials using Gabor filters,” *IEEE Trans. Ind. Appl.*, vol. 38, no. 2, pp. 425–440, Mar. 2002.
- [8] H. Sarri-Sarraf and J. S. Goddard, “Vision system for on-loom fabric inspection,” in *Proc. Textile, Fiber Film Ind. Tech. Conf.*, May 1998, pp. 1–8.
- [9] D.-M. Tsai and S.-K. Wu, “Automated surface inspection using Gabor filters,” *Int. J. Adv. Manuf. Technol.*, vol. 16, no. 7, pp. 474–482, 2000.
- [10] H. Y. T. Ngan, G. K. H. Pang, S. P. Yung, and M. K. Ng, “Wavelet based methods on patterned fabric defect detection,” *Pattern Recognit.*, vol. 38, no. 4, pp. 559–576, 2005.
- [11] D. Chetverikov and A. Hanbury, “Finding defects in texture using regularity and local orientation,” *Pattern Recognit.*, vol. 35, no. 10, pp. 2165–2180, 2002.
- [12] D. Chetverikov, “Structural defects: General approach and application to textile inspection,” in *Proc. 15th Int. Conf. Pattern Recognit.*, vol. 1, Sep. 2000, pp. 521–524.
- [13] D. Chetverikov, “Pattern regularity as a visual key,” *Image Vis. Comput.*, vol. 18, no. 12, pp. 975–985, 2000.
- [14] H. Y. T. Ngan and G. K. H. Pang, “Novel method for patterned fabric inspection using Bollinger bands,” *Opt. Eng.*, vol. 45, no. 8, 2006, Art. no. 087202.
- [15] C. S. C. Tsang, H. Y. T. Ngan, and G. K. H. Pang, “Fabric inspection based on the Elo rating method,” *Pattern Recognit.*, vol. 51, pp. 378–394, Mar. 2016.

- [16] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3431–3440.
- [17] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, Cham, Switzerland, 2015, pp. 234–241.
- [18] V. Badrinarayanan, A. Kendall, and R. Cipolla, (2015). "SegNet: A deep convolutional encoder-decoder architecture for image segmentation." [Online]. Available: <https://ieeexplore.ieee.org/document/7803544>
- [19] L. Wang, S. Guo, W. Huang, and Y. Qiao, (2015). "Places205-VGGNet models for scene recognition." [Online]. Available: <https://arxiv.org/abs/1508.01667>
- [20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
- [21] S. Zheng *et al.*, "Conditional random fields as recurrent neural networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1529–1537.
- [22] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 6, pp. 1263–1284, Sep. 2009.
- [23] S. Wang, Z. Li, W. Chao, and Q. Cao, "Applying adaptive over-sampling technique based on data density and cost-sensitive SVM to imbalanced learning," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jun. 2012, pp. 1–8.
- [24] C. Elkan, "The foundations of cost-sensitive learning," in *Proc. Int. Joint Conf. Artif. Intell.*, 2001, vol. 17, no. 1, pp. 973–978.
- [25] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proc. ICML*, vol. 96, 1996, pp. 145–156.
- [26] J. Zhu, H. Zou, S. Rosset, and T. Hastie, "Multi-class AdaBoost," *Statist. Interface*, vol. 2, no. 3, pp. 349–360, 2009.
- [27] X. Yuan, L. Xie, and M. Abouelenien, "A regularized ensemble framework of deep learning for cancer detection from multi-class, imbalanced training data," *Pattern Recognit.*, vol. 77, pp. 160–172, May 2018.
- [28] T. Hailikari, N. Katajavuori, and S. Lindblom-Ylanne, "The relevance of prior knowledge in learning and instructional design," *Amer. J. Pharmaceutical Educ.*, vol. 72, no. 5, p. 113, 2008.
- [29] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 447–456.
- [30] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 2650–2658.
- [31] M. Mostajabi, P. Yadollahpour, and G. Shakhnarovich, "Feedforward semantic segmentation with zoom-out features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3376–3385.
- [32] X.-X. Niu and C. Y. Suen, "A novel hybrid CNN-SVM classifier for recognizing handwritten digits," *Pattern Recognit.*, vol. 45, no. 4, pp. 1318–1325, 2012.
- [33] N. Dhungel, G. Carneiro, and A. P. Bradley, "Automated mass detection in mammograms using cascaded deep learning and random forests," in *Proc. Int. Conf. Digit. Image Comput., Techn. Appl. (DICTA)*, Nov. 2015, pp. 1–8.
- [34] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2017.
- [35] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with Gaussian edge potentials," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 109–117.
- [36] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "TextonBoost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context," *Int. J. Comput. Vis.*, vol. 81, no. 1, pp. 2–23, Dec. 2007.
- [37] Y. Chen, J. Li, H. Xiao, X. Jin, S. Yan, and J. Feng, "Dual path networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4467–4475.
- [38] J. P. Lewis, "Fast template matching," *Vis. Interface*, vol. 95, pp. 15–19, May 1995.
- [39] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 18th Int. Conf. Mach. Learn. (ICML)*, C. E. Brodley and A. P. Danyluk, Eds. San Francisco, CA, USA: Morgan Kaufmann, 2001, pp. 282–289.



WENBIN OUYANG received the B.S. and M.S. degrees in electrical engineering from Donghua University, Shanghai, China, and the Ph.D. degree in computer science and engineering from the University of North Texas, USA, in 2018. His research interests include image processing, deep learning, and 3D technology.



BUGAO XU received the Ph.D. degree from the University of Maryland at College Park, in 1992. He joined the faculty of The University of Texas at Austin, in 1993. Since 2016, he has been a Professor and the Chair with the Department of Merchandising and Digital Retailing, and a Professor with the Department of Computer Science and Engineering, University of North Texas. His research interests include high-speed imaging systems, image and video processing, and 3D imaging and modeling.



JUE HOU is currently pursuing the Ph.D. degree in digital textile technology with Donghua University, Shanghai, China. He is also a Visiting Student with the University of North Texas, USA. His research interests include computer vision, pattern recognition, and deep learning.



XIAOHUI YUAN received the Ph.D. degree in computer science from Tulane University, in 2004. He is currently an Associate Professor with the Department of Computer Science and Engineering, University of North Texas, in 2006. His primary research interests include computer vision and intelligent systems.