IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 13, No. 4, December 2024, pp. 4711~4721
ISSN: 2252-8938, DOI: 10.11591/ijai.v13.i4.pp4711-4721

Anchor selection based deep learning two stage fabric defect localization

Hattarki Pooja[1], Shrideva Soma[2]
[1]Department of Computer Science and Engineering, Visvesvaraya Technological University, Belagavi, India
[2]Department of Computer Science and Engineering, PDA College of Engineering, Gulbarga, India

Article Info

Article history:

Received Nov 22, 2023
Revised Apr 2, 2024
Accepted Apr 17, 2024

Keywords:

Anchor selection approach
Deep learning regression
Fabric defect detection
Quaternion Fourier transform
Super pixel segmentation

ABSTRACT

Localizing and classifying fabric defects is a crucial step in the quality control process used in the production of textiles. Recently, fabric defect classification and detection have made use of deep learning approaches based on anchor selection. But due to in effectiveness in anchor selection, the computational overhead and localization error are higher in these solutions. As a solution to this problem, this work proposes a two-stage improvised anchor selection deep learning technique. In first stage, quaternion fourier transform frequency domain analysis along with super pixel segmentation is done over the fabric image to select probable defect regions. In the second stage deep learning based regression along with super pixel segment comparison is done over the probable defect regions localize and categorize the defect. Due to effectiveness in selection of probable defect regions and categorization of regions, the defect localization accuracy is increased at a comparative low computational overhead in the proposed two stage improvise anchor selection deep learning technique. Testing against the irish longitudinal study on ageing (TILDA) fabric defect detection dataset, the proposed solution is found to provide 1.2% higher fabric defect localization accuracy at a 3% lower computation overhead compared to most recent existing works.

Corresponding Author:

Hattarki Pooja
Department of Computer Science and Engineering, Visvesvaraya Technological University
Belagavi, India
Email: poojahattarki11@rediffmail.com

1. INTRODUCTION

During the textile production process, a variety of elements influence fabric quality, including material quality, mechanical considerations, dye type, yarn size, and human factors. One of the most important quality control phases in the textile manufacturing process is the detection and classification of fabric faults. The traditional strategy of using human visual examination to find faults is costly, time-consuming, and slow (12 meters per minute). It is consequently unsuited for mass production. Fast assembly is necessary in large-scale manufacturing processes, and faults must be identified early on to avoid resource waste and future phases of defect development. Early defect discovery boosts product value and reduces corporate losses [1]. Many automatic strategies for detecting fabric problems have been investigated in attempt to attain this objective. Existing automatic fabric flaw detection systems can be divided into two categories: classical approaches and machine learning techniques. Traditional methods detect problems by extracting statistical, structural, spectral, and model-based information and thresholding them. They are not adaptable to fabric textural changes, and thresholds or characteristics must be fine-tuned or tailored to each fabric. In short, these strategies lack generality. As a result, machine learning approaches are presented as solutions to this problem. They fall into two categories: conventional machine learning methods and deep learning methods. In order to detect damaged fabrics, classical machine learning techniques take a range of manually created visual signals and train machine learning algorithms like support vector machine (SVM), artificial neural networks (ANN) as well as others. Recently, fabric defect detection has made use of deep learning [2], [3]. By using convolution operations, deep learning classifiers may learn complicated features without the requirement for hand-crafted features.

Over 70 distinct categories exist for fabric defects. The majority of faults in photos are quite small and have an odd aspect ratio. One stage and two stage deep learning algorithms are the two types of algorithms utilized for microscopic fault detection. One-stage algorithms are more accurate but have a faster inspection speed [4]. Two stage algorithms have higher accuracy but have higher runtime. Anchor selection based deep learning approaches reduce the computational overhead of two stage algorithms by selecting anchor points or probable defective regions around which defect detection should work. But these methods can be further improved by selecting the anchor with higher probability of being defect and reducing the number of redundant anchors. Selecting the anchors with higher probability of turning defect reduces the computation complexity and increases the defect localization accuracy. This work proposes a two stage improvised anchor selection deep learning technique addressing the problem in defect localization and categorization at a comparatively lower computational complexity [5]. The proposed two stage solution, select the most probable regions for defect analysis in the first stage so that computational complexity is reduced. In the second stage, deep learning based regression analysis using an optimized convolutional neural network (CNN) is done to localize and categorize the defect accurately with the regions selected in the first stage. Following are the novel contributions of the proposed solution:

- A novel integration of quaternion fourier transform based frequency domain analysis with super pixel segmentation to select the most probable defect regions in the fabric at the first stage is proposed. By selecting accurate defect regions, redundant and irrelevant regions are isolated from analysis contributing to lower false positives and reduced computational complexity.
- A novel CNN architecture with sequence of full-connection layers which are then finally branched to Softmax classification and bounding box regression to localize and categorize the defect in the regions selected in first stage is proposed. The proposed CNN architecture is able to learn more intricate features necessary for defect localization and categorization. As the result, the accuracy of defect localization/categorization is increased.

The structure of this document is as follows: the current technique for localizing fabric defects is described in section 2. The two-stage deep learning solution based on anchoring is described in full in section 3. The outcomes of the suggested solution are given in section 4. The conclusion and areas for additional investigation are presented in section 5.

2. RELATED WORK

Almeida *et al.* [6] proposed a customized CNN architecture and trained with fabric net dataset to detect fabric defects. The architecture had four convolutional and max pool layers followed by two fully connected layers. Rectified linear unit (ReLU) was used as activation function. The input images are pre-processed using histogram equalization to improve the contrast. The pre-processed image is then passed to CNN. The method works well for only uniform colors and cannot work well for complex textures. Li *et al.* [7] integrated both deep features and handcrafted features to solve the problem of detecting defects in fabric with complex textures. Hand crafter features and deep features are fused and dimension reduction is done using principal component analysis (PCA). Low rank decomposition model is proposed to decompose the features to low rank matrix and sparse matrix representing the background region and defect region. The method has higher false positives and some patters are not recognized for defects. Yapi *et al.* [8] proposed a signature-based method to detect fabric defects. Image is first pre-processed to detect basic pattern size for image decomposition into blocks. Signature for each block is formed using redundant contour let transform. Bayesian classifier is trained to classify the signature to defective or non-defective based on training dataset. The method is based on non-overlapping block decomposition and minor defects at boundary of decomposed blocks are not detected. Susan and Sharma [9] proposed an unsupervised method for fabric defect detection. The image is split to textural patches through a sliding window approach. For each of the patches, a regularity index based on Gaussian gain is calculated. The regularity index deviating from the standard deviation all patches are detected as defective patches. The method has higher false positives and works only for certain distribution of defects. Tong *et al.* [10] proposed a defect detection model based on non-locally centralized sparse representation.

The methods build a dictionary from non defective samples. Using the dictionary, a non defective fabric pattern is constructed and compared to a fabric pattern image to spot the defect. The method has high overhead and works only for certain patterns. Qu *et al.* [11] proposed a dictionary-based fabric defect method using dual scales. Dictionary is leant from training images in two Gaussian scales. Matching is done for test image in two scales to isolate the defect. Like other dictionary-based methods, this method works only for certain patterns. Shi *et al.* [12] proposed a low-rank decomposition modelto segment the defective region in the fabric. Gradient weight matrix is first constructed from the original image. This matrix is decomposed into low rank part, sparse part and noise part. Saliency map is constructed from sparse part and thresholded to locate the fabric defect. The method has higher false ratio for dot patterned fabric. Huangpeng *et al.* [13] proposed an unsupervised method using texture prior for detection of defects. A texture prior map is constructed with higher value in map indicating probable abnormal regions. From the prior map, low rank decomposition is done to identify the defective regions. The method works only for the case of small defects and as defect size increases, the accuracy of detection reduces. Wang *et al.* [14] integrated manual and deep learning local features to detect fabric flaws. Local features are combined with deep features that were extracted using a CNN. The sparse matrix is found by executing non-convex total variation. A saliency map is produced from it. Defect regions are found by segmenting the saliency map. This strategy involves a larger computational burden. Several researchers [15], [16] employed a multiscale CNN to identify fabric flaws. To find the faults', clustering analysis is performed using pre-known defect size information. The approach is not universal for all patterns, but it does work for minute flaws. Mei *et al.* [17] suggested a method for fault detection and localization based on unsupervised learning. Convolutional denoising autoencoder networks investigate flaws at each resolution while reconstructing the picture at several Gaussian pyramid layers. This technique requires a small training set, but the computational complexity is relatively high. Wei *et al.* [18] used a CNN and compressive sensing in tandem to identify fabric flaws. Compressive sensing is a technique for expanding the training set's volume and augmenting data with tiny sample numbers. This technique raises CNN's training volume, which increases accuracy. However, the accuracy of the CNN will be impacted if it is overloaded with training examples of various patterns. Ouyang *et al.* [19] used CNN for fabric defect detection. Based on random distribution of motif in defective fabric, defect probability map is generated and this is passed as input to CNN for segmentation of defects. This method is applicable only a bigger size defect.

Jing *et al.* [20] used deep CNN for defect detection. The fabric image is decomposed into patches and each local patch is labeled. CNN is trained with labeled patches. Sliding over the image, each patch was classified to defect or not using the CNN. The method works only for big size defects. Wei *et al.* [21] proposed a faster regional-based convolutional network for fabric defect detection. Visual geometry group (VGG16) was used for feature extraction. VGG16 features are extracted at various layers and pooling layer is adapted to fuse the features. The method works only for uniform patterns. Zhang *et al.* [22] used K means algorithm to classify defective fabrics. Based on histogram, the fabrics are clustered to two class of defective and non-defective. The method works only for big defects. Priori anchor convolutional neural network (PRAN-Net) was employed by [23] to detect fabric defects. The author reserved more in-depth information about minute faults by using the feature pyramid network (FPN) to pick multi-scale feature maps. In order to find extreme flaws more precisely and effectively, the authors suggested a method for generating sparse priori anchors using ground truth boxes for fabric defects rather than permanent anchors. To find fabric flaws, Cheng *et al.* [24] employed an enhanced version of the you only look once version 3 (YOLOv3) algorithm. Here are two crucial actions to take: initially, the k-means algorithm is used to identify the size and quantity of previous frames, and the fabric defect size is combined with it to carry out the dimension clustering of target frames based on YOLOv3. Secondly, the high-level data is integrated with the low-level features, and the feature maps of various sizes are supplemented with the YOLO detection layer to detect defects. The technique is limited to lattice and gray cloth. Sandhya *et al.* [25] trained Alexnet to classify fabric image to defective or non defective. But the approach works only for bigger defects. Several researchers in [26]–[28] optimized CNN for fabric defect detection. VGG16 model was customized for number of channels to improve the accuracy of defect detection. The method requires large volume of training dataset. Luo *et al.* [29] proposed an attention mechanism over YOLO deep learning network for fabric defect detection. In this fabric image is processed as whole using improved YOLO deep learning models. The summary of the done is presented in Table 1.

Though the network can detect defect, it cannot categorize the defect. Also, the computation complexity increases exponentially as the fabric dimension increases. Huang and Xiang [30] proposed a CNN architecture based on repeated pattern detection to localize fabric defects. Though the model is able to localize defect with higher accuracy, it can localize only for certain fabric types and the computation complexity increase exponentially with increase in fabric dimension. Luo *et al.* [31] combined YOLOV3 with deformable CNN to localize fabric defects. The image is processed as whole YOLO to extract features. The features are then processed by deformable CNN to localize the defect. But as the fabric is processed as whole, the computational complexity is higher. Fang *et al.* [32] proposed a CNN architecture with attention mechanism to detect fabric defects. The fabric image is first filleted using frequency domain filtering and then processed with CNN architecture to localize the defects. The training volume need for defect localization needs to be very high to achieve better accuracy. Liu *et al.* [33] modified the YOLOV4 architecture by replacing maxpool with softpool and series of convolutional layers for fabric defect detection. Though the modification yielded

6% higher accuracy compared to YOLOV4, the computational complexity is higher as image is processed as whole. From the survey, it can be seen that most important problems in existing works are computational complexity, higher false positives and lacking generic defect detection capability. The proposed solution in this work addresses these issues.

Table 1. Survey summary

Works	Solution	Issue	Difference to proposed work
Almeida *et al.* [6]	Fabric image processed as whole with modified CNN architecture	The method works well for only uniform colors and cannot work well for complex textures	The proposed work selects most probable defect regions and process each region instead of whole image. So computational complexity is lower compared to existing work
Li *et al.* [7]	Integrated both deep and handcrafted features to detect fabric defects	Has higher false positives and pattern specific	The proposed solution works for any patterns
Yapi *et al*. [8]	Block based comparison using Bayesian classifier	minor defects at boundary of decomposed blocks are not detected	Proposed solution does dynamic region-based analysis instead of fixed blocks. This reduces the computational complexity and reduces false positives
Susan and Sharma [9]	Image split to fixed size patches and patches compared using Gaussian gain	Has higher false positives and works only for certain distribution of defects	Proposed solution is independent of defect distribution
Tong *et al.* [10]	Built dictionary from non defective samples and comparison is done	Has higher false positives and suffers from zero-day problem	The proposed solution is not affected from zero-day problem.
Qu *et al.* [11]	Dictionary based fabric defect method using dual scales.	It works only for certain patterns	The proposed solution is pattern independent
Huangpeng *et al*. [13]	Texture based analysis to detect defect	Works only for certain defect dimensions	Frequency based analysis and works for any defect dimensions
Ouyang *et al.* [19]	Motiff based defect map construction and processing with CNN to localize defect	Works only for certain defect dimensions	Frequency based defect map construction and processing with CNN to work generic for any defect dimensions
Solutions [29]–[33]	YOLO based processing of whole image at once to localize the defect	Computational complexity and false positives are higher	Defect analysis is done only on probable defect regions reducing the computational complexity and false positives

3. PROPOSED METHODOLOGY

From the survey, it can be seen that deep learning methods provide better localization of fabric defects but their computation complexity is higher. Anchor based approaches reduced the computation complexity of deep learning approaches. But the effectiveness of anchor-based approaches can be further improved by selecting the best anchors. Motivated by this observation, this work proposes a deep learning-based anchor selection approach (ASA) and integrates with deep learning-based fabric defect localization for improved localization accuracy and reduced computational overhead. The proposed work adopts the fabric defect detection process of PRAN-Net. In PRAN-Net, features were extracted at various scales using Resnet-101 where about 12 texture features are extracted and then the Priori anchors are generated in each scale of feature map as defect proposals. The defect proposals are then classified to defect or no defect by the classification network as shown in Figure 1. Multi scale feature map generation using Resnet-101 proposed in PRAN-net is used in this work. The three feature maps of Class 1, Class 2, and Class3 which has enough information for tiny defect detection is used in this work. Algorithm 1 shows the detect defect using ASA.

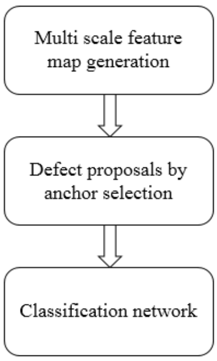

Figure.1 Stages in PRAN-Net

Algorithm 1. Detect defect using ASA
Input Fabric images (Defective and normal fabric)
Output Defect type (Tiny and Big defects)
Step 1 ti HSV transform(image)
Step 2 qv QDCT (ti)
Step 3 Gqv Gaussian_Smooth(qv);
Step 4 si Inverse QDCT(Gqv);
Step 5 for each segment in si
Calculate for each segment
If > 0.7
Keep segment
Else
Mask segment
Step 6 fsi Extract PRNet features(si)
Step 7 defecttype Classify fsi using softmax
Step 8 return defecttype

PRAN-Net selected anchors using a location prediction network which operates at level of pixels. A 1×1 convolutional layer is used to create the faulty score map by semantic segmentation. The defect probability map is then created by transforming the defective scores using an element-wised sigmoid function, and it has the same dimensions as the feature map. Defect pixels are those whose values above the threshold value ΔL, where ΔL is the preset threshold of 100 for determining the value of a pixel in a faulty region. This results in large number of anchor regions in the PRAN-Net and also defective proposals are made based on fixed size where size information for each defect must be known priori.

Deviating from PRAN-Net, this work proposes an anchor point and defective proposal region selection based on frequency domain analysis combined with super pixel-based selection. The suggested approach uses input photographs that have been subsampled to half of their original size because processing the original size of the images will result in higher computational costs. The subsampled red, green, and blue (RGB) image is subjected to the HSI transform to produce hue, saturation, and intensity (HSI) images. Use of HSI in frequency domain analysis is preferred over RGB color space because of it is greater conformity with the human visual system. Prior to being converted to HSI, the colors in RGB are normalized within the interval 0 to 1. where the normalized numbers for the pixel are R', G', and B'. Using RGB color images, the HSI transform is applied as per (5) and (6).

$$Total = R + G + B \tag{1}$$

$$R' = \frac{R}{total*255} \tag{2}$$

$$G' = \frac{G}{total*255} \tag{3}$$

$$B' = \frac{B}{total*255} \tag{4}$$

$$H = \begin{cases} \theta \\ 360 - \theta, \end{cases} \quad B \leq G, B > G \tag{5}$$

$$S = 1 - \frac{3}{(R+G+B}[\min(R, G, B)] \tag{6}$$

$$l = \frac{1}{3}(R + G + B) \tag{7}$$

$$\theta = arcos\left\{\left(\left(\frac{1}{2}\right)\right)[(R-G)+(R-B)]/([(R-B)(G-B)^{\frac{1}{2}})\right\} \tag{8}$$

$$f(n, m) = H(n, m)_{\mu 1} + S(n, m)_{\mu 2} + l(n,$$
$$f(n, m) = H(n, m)_{\mu 1} + S(n, m)_{\mu 2}$$
$$f(n, m) = H(n, m)_{\mu 1} + S(n, m)_{\mu 2}$$
$$f(n, m) = H(n, m)_{\mu 1} + S(n, m)_{\mu 2} + l(n, \tag{9}$$

where R, G, and B represent the image's red, green, and blue component's pixel values. In (9) provides a quaternion representation of the HSI image. where the pixel's position is (n, m) are the pixel's hue, saturation, and intensity at (n, m). The condition determines the value of μ. The pixel at (u, v) undergoes the quaternion Fourier transform, with F1 and F2 representing the pixel's fourier derivative function, as indicated by (10).

$$F(u,v) = F_1(u,v) + F_1(u,v)_{\mu 2} \tag{10}$$

$$F_1(u,v)\frac{1}{\sqrt{MN}}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} e - \mu_1 2\pi(\left(\frac{mu}{M}\right) + \frac{nu}{N} f_1(n,m) \tag{11}$$

An image's borders and other abrupt pixel changes greatly increase the high-frequency component of the fourier transform. Therefore, attenuating the designated range of high-frequency elements in the image's quaternion fourier transform allows for frequency domain smoothing of the image. The quaternion fourier converted data are then sent via a gaussian quaternion high pass filter. In (12) provides the two-dimensional Gaussian high pass filter transfer function with the cut-off frequency at a distance D0. where D (u, v) is the distance between point (u, v) and the origin of the frequency rectangle (M/2, N/2) where M and N are the rectangle's dimensions and M and N stand for the rectangle's length and breadth, respectively, and σ is a measure of Gaussian dispersion. It is computed using the formula in (13).

$$H_q(u,v) \tag{12}$$

$$D(u,v) = [\left(u - \frac{M}{2}\right)^2 \tag{13}$$

$$\begin{aligned} f_i(n,m) &= \frac{1}{\sqrt{MN}}\sum_{v=0}^{M-1}\sum_{u=0}^{N-1} e - \mu_1 2\pi(\left(\frac{mu}{M}\right) + \frac{nu}{N} f_1(l \\ f_i(n,m) &= \frac{1}{\sqrt{MN}}\sum_{v=0}^{M-1}\sum_{u=0}^{N-1} e - \mu_1 2\pi(\left(\frac{mu}{M}\right) + \\ f_i(n,m) &= \frac{1}{\sqrt{MN}}\sum_{v=0}^{M-1}\sum_{u=0}^{N-1} e - \mu_1 2\pi(\left(\frac{mu}{M}\right) + \end{aligned} \tag{14}$$

Compared to ideal high pass filters, the outcomes of Gaussian high pass filtering are smoother. An output image is produced using the inverse quaternion transform after Gaussian high pass filtering. In (14) describes how to perform the inverse transform. The frequency domain map with highlighted salient regions is the output image that is produced following the inverse transform. Super pixels are collections of spatially related pixels with comparable color or intensity characteristics. Images are divided into segments with their natural borders retained using super pixel segmentation. The most widely used technique for super pixel segmentation in this study is simple linear iterative clustering (SLIC). Salient area segments' super pixel segments have a discernible color contrast with other segments, and their spatial distribution is sparser than that of other segments. Based on these two observations, each of salient region identified in frequency domain analysis, are evaluated in terms of contract with other segments to decide probability of it being a defective region. The probability of a salient region to an anchor region or defect proposal is found as given in (15).

$$P_R = \frac{1}{WC} \tag{15}$$

Where

$$WC = \sum_{j=1}^{n} W(i,j).\left|\left|mc_i - mc_j\right|\right|$$

$$WC = \sum_{j=1}^{n} W(i,j).\left|\left|mc_i - \right|\right| \tag{16}$$

$$W(i,j) = \left[SP_j\right| Sim_d(i,j) \tag{17}$$

The weight W is calculated based on the area of super pixel segment and mc is the salient region the spatial similarity is given as shown in (17). When is greater than the threshold, the salient region is chosen as the anchor region after a comparison is made using a threshold of 0.8 for resemblance of the region to the probable region. A ROI align layer resizes defect suggestions to the fixed size and the feature maps that go with them. Following that, the features are passed via a series of full-connection layers before branching out to bounding box regression and classification. The regression branch predicts the position of the defect after the secondary location, while the classification branch predicts the type of the defect. In this work, the classifier is the Softmax classifier. The likelihood of each fault as well as the absence of any defects is estimated by the softmax classifier. In a regression situation, the Softmax classifier is employed. Assume that the softmax classifier needs to estimate the probability for each of the K values under the following scenarios: $\{1,2,\ldots K\}$. The K dimensional vector containing the K estimated probabilities is the result of the softmax classifier. In (18) provides the loss function that will be used to train the softmax regression classifier.

$$L = -\left[\sum_{i=1}^{m}\sum_{k-0}^{1} 1\left\{y^{(i)} = k\right\}\log P\left(y^{(i)} = k|z^{(i)};\theta\right)\right] \quad (18)$$

$$P\left(y^{(i)} = k\middle|z^{(i)};\theta\right) = \frac{\exp\left(\theta^{(k)}z^{(i)}\right)}{\sum_{j=1}^{K} exp\left(\theta^{(k)}z^{(i)}\right)} \quad (19)$$

4. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed solution is implemented in MATLAB 2019b and the performance is tested in machine with configuration of: Intel i5 central processing unit (CPU), 8 GB random access memory (RAM) and Windows 10 OS. A dataset of fabric defect defection is constructed with two datasets of textile defect detection [34] and the standard TILDA dataset. The dataset has images in total six classes: good, color, cut, hole, thread, metal_contamination (MC). The dataset has mix of both tiny and bigger defects. The performance is measured in terms of accuracy, precision, recall, localization error, and defect detection time. The performance is measured for each class of defects and for both tiny and bigger defects. The performance of proposed ASA is compared against PRAN-Net [23], multi scale CNN [16] and faster region-based convolutional neural networks (RCNN) [21]. The comparison of existing three solutions with the proposed work is presented in Table 2. Figure 2 shows the comparison of performance in terms of accuracy, precision, and recall.

Table 2. Performance comparison

Method	Accuracy	Precision	Recall
ASA	93.1	64.5	55.3
PRAN-Net	91.9	62.3	53.3
Multiscale-CNN	89.4	58.4	51
Faster RCNN	87.1	60.2	50.9

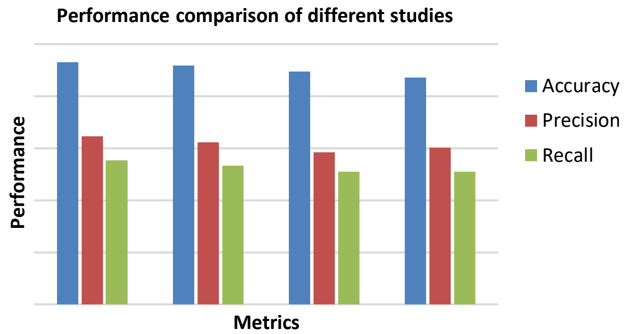

Figure 2. Comparison of performance in terms of accuracy, precision, and recall

The proposed solution has atleast 1.2% higher accuracy compared to PRAN-Net, 3.7% higher accuracy compared to multiscale CNN, 6% higher accuracy compared to faster RCNN as shown in Figure 2. ASA performs better when compared to the existing three methods because of the feature maps that are extracted at various scales using Resnet-101. The accuracy has increased in the proposed solution due to selection of anchor regions with higher probability of defect localized near it. This reduced the false positive positives in the proposed solution. The number of anchor regions found PRAN-Net is very higher compared to proposed solution and many of redundant or irrelevant anchor points contributed to higher false positives. Due to this PRAN-Net accuracy is lower compared to proposed solution. Faster RCNN misses out many small defects to speed up in defect detection stage and this has reduced the accuracy. Multiscale approach does analysis at various scales but the small defects (less than 1% of total images) get missed out. The average time to detect defect is measured across the four solutions and the result is given in Table 3. Figure 3 shows the comparison of defect detection in terms of time.

Table 3. Computation time comparison

Method	Defect detection time (ms)					
	Color	Cut	Hole	Thread	MC	Avg
ASA	120	127	119	131	123	124
PRAN-Net	132	135	132	142	130	134.2
Multiscale- CNN	161	167	162	172	164	165.2
Faster RCNN	123	130	124	137	127	128.2

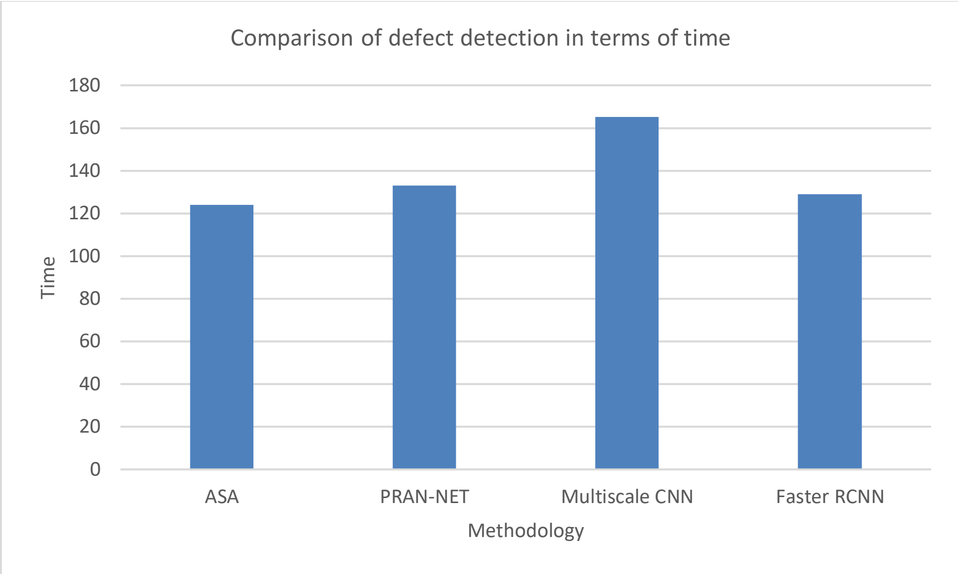

Figure 3. Comparison of defect detection in terms of time

The average time taken for defect detection in proposed solution is 8% lower compared to PRAN-Net, 33% lower compared to multiscale CNN and 3% lower compared to Faster RCNN as shown in Figure 3. The time has reduced in proposed solution due to selection of few anchor regions where defect detection is focused. PRAN-Net has higher time due to more anchor regions compared to proposed solution. Faster RCNN reduced the executed by comprising on the accuracy. The test images are split to two categories namely: tiny (< 1% of total image) and big (>1% of total image) and accuracy is compared across 4 solution for both image categories. The results are given in Table 4.

Table 4. Comparison of accuracy for tiny and big defects

Method	Accuracy	
	Tiny defects	Big defects
ASA	92	94.2
PRAN-Net	90.9	92.9
Multiscale-CNN	87.8	88.1
Faster RCNN	86.2	88

The average accuracy for tiny defects in proposed solution is 1.1% higher compared to PRAN-Net, 4.2% higher compared to Multiscale CNN and 5.8% higher compared to Faster RCNN. The average accuracy for big defects in proposed solution is 1.3% higher compared to PRAN-Net, 6.1% higher compared to Multiscale CNN and 6.2% higher compared to Faster RCNN. The proposed solution performed well for tiny defects due to its unique selection of anchor points. These anchor points were able to cover even a tiny defect region. The accuracy for different types of defects is measured in proposed solution and given in Table 5. The highest accuracy is achieved for Color defect in proposed solution followed by cut defect. The receiver operating characteristic (ROC) plot comparing four solutions is given in Figure 4.

Table 5. Comparison of accuracy for various defects

Defect	Accuracy
Color	94.5
Cut	93.2
Hole	91.1
Thread	92.4
MC	90.3

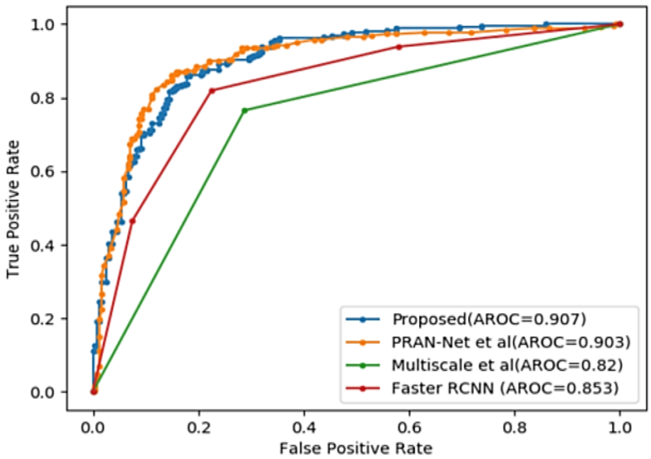

Figure 4. AROC curve comparison for ASA, PRAN-Net, multiscale CNN, faster RCNN

The ROC area is higher in proposed solution compared to existing works. The proposed solution demonstrates better trade-off between sensitivity and specificity compared to existing works. The defect localization error is measured in terms of Hausdorff distance (). This is measured by calculating the difference of contours of predicted and actual defect region as given in (20). Lower the value of lower is the defect localization error. The average value is measured for tiny and big detects and the result is given in Table 6.

$$d_H(P,A) = max\begin{cases}\sup\inf d(P,A), p\varepsilon P^{t\varepsilon T}\\ \sup\inf d(P,A), a\varepsilon A^{r\varepsilon R}\end{cases} \quad (20)$$

Table 6. Defect localization error

Method	Tiny	Big
ASA	18	43
PRAN-Net	24	57
Multiscale-CNN	29	62
Faster RCNN	30	64

The defect localization error in terms of is at least 33% lower for tiny defects and 32% lower for big defects. The defect localization error has reduced in the proposed solution due to selection of most probable regions using combined frequency domain analysis with super pixel segmentation at first stage and deep learning regression on the selected region in the second stage. As summarized from Table 1, the computing complexity in approach processing image as whole is higher compared to anchor selection-based approaches. ASA reducing the computing complexity by narrowing the defect analysis to selective regions. But with improper selection of probable regions, they increased the false positives and also increased the computing complexity though it is lower than deep learning approaches processing image as whole. This work solved this problem by selecting the best probable regions and doing deep learning regression analysis on those regions. From the results of defect localization error, accuracy and defect detection time, it can be seen that proposed solution has effectively solved the problems. Also, the proposed solution performed consistently for different types of defects and defect dimensions.

5. CONCLUSION

This work proposed an ASA based two stage deep learning fabric defect localization technique. The anchors were selected with machine learning approach using saliency region detection and scoring the salient region for a probability of defect occurrence. Deep learning localizes the faults in the selected anchor regions. The proposed solution is able to reduce the computational overhead by 3% and also increases the fabric defection localization by 1.2%. Comparing the suggested approach to the current methods, it is found that the latter performs worse overall in terms of accuracy, precision, and recall, and ASA can handle both large and little flaws. Future studies will assess the suggested solution's resilience against a range of intricate textures.

REFERENCES

[1] J. Bullon, A. G. Arrieta, A. H. Encinas, and A. Q. Dios, "Manufacturing processes in the textile industry. Expert systems for fabrics production," *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal*, vol. 6, no. 1, pp. 41–50, Jan. 2017, doi: 10.14201/ADCAIJ2017614150.

[2] T. Mahmud, J. Sikder, R. J. Chakma, and J. Fardoush, "Fabric defect detection system," in *IntelLigent Computing and Optimization: Proceedings of the 3rd International Conference on Intelligent Computing and Optimization 2020 (ICO 2020)*, 2021, pp. 788–800, doi: 10.1007/978-3-030-68154-8_68.

[3] C. Li, J. Li, Y. Li, L. He, X. Fu, and J. Chen, "Fabric defect detection in textile manufacturing: a survey of the state of the art," *Security and Communication Networks*, pp. 1–13, 2021, doi: 10.1155/2021/9948808.

[4] C. C. -Ho and G. K. H. Pang, "Fabric defect detection by Fourier analysis," *IEEE Transactions on Industry Applications*, vol. 36, no. 5, pp. 1267–1276, 2000, doi: 10.1109/28.871274.

[5] J. Wang, K. Chen, S. Yang, C. C. Loy, and D. Lin, "Region proposal by guided anchoring," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2960–2969, doi: 10.1109/CVPR.2019.00308.

[6] T. Almeida, F. Moutinho, and J. P. M. -Carvalho, "Fabric defect detection with deep learning and false negative reduction," *IEEE Access*, vol. 9, pp. 81936–81945, 2021, doi: 10.1109/ACCESS.2021.3086028.

[7] C. Li, B. Jiang, Z. Liu, Y. Dong, S. Tang, and D. Weng, "Fabric defect detection based on deep-handcrafted feature and weighted low-rank matrix representation," *Journal of Engineered Fibers and Fabrics*, vol. 16, 2021, doi: 10.1177/15589250211008453.

[8] D. Yapi, M. S. Allili, and N. Baaziz, "Automatic fabric defect detection using learning-based local textural distributions in the contourlet domain," *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 3, pp. 1014–1026, Jul. 2018, doi: 10.1109/TASE.2017.2696748.

[9] S. Susan and M. Sharma, "Automatic texture defect detection using Gaussian mixture entropy modeling," *Neurocomputing*, vol. 239, pp. 232–237, 2017, doi: 10.1016/j.neucom.2017.02.021.

[10] L. Tong, W. K. Wong, and C. K. Kwong, "Fabric defect detection for apparel industry: a nonlocal sparse representation approach," *IEEE Access*, pp. 1–1, 2017, doi: 10.1109/ACCESS.2017.2667890.

[11] T. Qu, L. Zou, Q. Zhang, X. Chen, and C. Fan, "Defect detection on the fabric with complex texture via dual-scale over-complete dictionary," *The Journal of The Textile Institute*, vol. 107, no. 6, pp. 743–756, Jun. 2016, doi: 10.1080/00405000.2015.1061760.

[12] B. Shi, J. Liang, L. Di, C. Chen, and Z. Hou, "Fabric defect detection via low-rank decomposition with gradient information," *IEEE Access*, vol. 7, pp. 130423–130437, 2019, doi: 10.1109/ACCESS.2019.2939843.

[13] Q. Huangpeng, H. Zhang, X. Zeng, and W. Huang, "Automatic visual defect detection using texture prior and low-rank representation," *IEEE Access*, vol. 6, pp. 37965–37976, 2018, doi: 10.1109/ACCESS.2018.2852663.

[14] J. Wang, C. Li, Z. Liu, Y. Dong, and Y. Huang, "Combing deep and handcrafted features for NTV-NRPCA based fabric defect detection," in *Pattern Recognition and Computer Vision*, 2019, pp. 479–490, doi: 10.1007/978-3-030-31726-3_41.

[15] Y. Li, W. Zhao, and J. Pan, "Deformable patterned fabric defect detection with fisher criterion-based deep learning," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 2, pp. 1256–1264, Apr. 2017, doi: 10.1109/TASE.2016.2520955.

[16] S. Zhao, L. Yin, J. Zhang, J. Wang, and R. Zhong, "Real-time fabric defect detection based on multi-scale convolutional neural network," *IET Collaborative Intelligent Manufacturing*, vol. 2, no. 4, pp. 189–196, Dec. 2020, doi: 10.1049/iet-cim.2020.0062.

[17] S. Mei, Y. Wang, and G. Wen, "Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model," *Sensors*, vol. 18, no. 4, Apr. 2018, doi: 10.3390/s18041064.

[18] B. Wei, K. Hao, X. Tang, and Y. Ding, "A new method using the convolutional neural network with compressive sensing for fabric defect classification based on small sample sizes," *Textile Research Journal*, vol. 89, no. 17, pp. 3539–3555, Sep. 2019, doi: 10.1177/0040517518813656.

[19] W. Ouyang, B. Xu, J. Hou, and X. Yuan, "Fabric defect detection using activation layer embedded convolutional neural network," *IEEE Access*, vol. 7, pp. 70130–70140, 2019, doi: 10.1109/ACCESS.2019.2913620.

[20] J. Jing, H. Ma, and H. Zhang, "Automatic fabric defect detection using a deep convolutional neural network," *Coloration Technology*, vol. 135, no. 3, pp. 213–223, Jun. 2019, doi: 10.1111/cote.12394.

[21] B. Wei, K. Hao, X. S. Tang, and L. Ren, "Fabric defect detection based on faster RCNN," *Advances in Intelligent Systems and Computing*, vol. 849, pp. 45–51, 2019, doi: 10.1007/978-3-319-99695-0_6.

[22] H. Zhang, J. Ma, J. Jing, and P. Li, "Fabric defect detection method based on improved fast weighted median filtering and K-means," *Fangzhi Xuebao/Journal of Textile Research*, vol. 40, no. 12, pp. 50–56, 2019, doi: 10.13475/j.fzxb.20181200407.

[23] P. Peng, Y. Wang, C. Hao, Z. Zhu, T. Liu, and W. Zhou, "Automatic fabric defect detection method using pran-net," *Applied Sciences*, vol. 10, no. 23, pp. 1–13, 2020, doi: 10.3390/app10238434.

[24] Z. Cheng, X. Luo, Y. Shi, and K. Kita, "Fabric defect detection algorithm based on YOLOv3 Transfer learning," in *International Conference on Frontiers of Electronics, Information and Computation Technologies*, New York, USA: ACM, May 2021, pp. 1–7, doi: 10.1145/3474198.3478149.

[25] N. Sandhya, N. M. Sashikumar, M. Priyanka, S. M. Wenisch, and K. Kumarasamy, "Automated fabric defect detection and classification: a deep learning approach," *Textile and Leather Review*, vol. 4, pp. 315–335, Dec. 2021, doi: 10.31881/TLR.2021.24.

[26] Z. Liu, C. Zhang, C. Li, S. Ding, Y. Dong, and Y. Huang, "Fabric defect recognition using optimized neural networks," *Journal of Engineered Fibers and Fabrics*, vol. 14, Jan. 2019, doi: 10.1177/1558925019897396.
[27] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," *Proceedings-30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, pp. 936–944, 2017, doi: 10.1109/CVPR.2017.106.
[28] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *arXiv-Computer Science*, Mar. 2017, doi: 10.48550/arXiv.1703.06870.
[29] X. Luo, Q. Ni, R. Tao, and Y. Shi, "A lightweight detector based on attention mechanism for fabric defect detection," *IEEE Access*, vol. 11, pp. 33554–33569, 2023, doi: 10.1109/ACCESS.2023.3264262.
[30] Y. Huang and Z. Xiang, "RPDNet: automatic fabric defect detection based on a convolutional neural network and repeated pattern analysis," *Sensors*, vol. 22, no. 16, Aug. 2022, doi: 10.3390/s22166226.
[31] X. Luo, Z. Cheng, Q. Ni, R. Tao, and Y. Shi, "Defect detection algorithm for fabric based on deformable convolutional network," *Textile Research Journal*, vol. 93, no. 9–10, pp. 2342–2354, May 2023, doi: 10.1177/00405175221143742.
[32] B. Fang, X. Long, F. Sun, H. Liu, S. Zhang, and C. Fang, "Tactile-based fabric defect detection using convolutional neural network with attention mechanism," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–9, 2022, doi: 10.1109/TIM.2022.3165254.
[33] Q. Liu, C. Wang, Y. Li, M. Gao, and J. Li, "A fabric defect detection method based on deep learning," *IEEE Access*, vol. 10, pp. 4284–4296, 2022, doi: 10.1109/ACCESS.2021.3140118.
[34] N. Belkhir, "Textile defect detection," *Kaggle*, 2019. Accessed: Feb. 11, 2023. [Online]. Available: https://www.kaggle.com/datasets/belkhirnacim/textiledefectdetection

BIOGRAPHIES OF AUTHORS

Hattarki Pooja earned her B.E. in Information Science and Engineering and M.Tech. in Computer Science and Engineerig from Visvesvaraya Technological University Belagavi in 2014 and 2016 respectively. She worked as guest faculty at Department of Computer Science at Gulbarga University, Kalaburagi for 1 year in 2017-2018 and currently working as Asst. professor at Department of Computer Science and Engineering, Veerappa Nisty Engineering College, Shorapur from 2021. She can be contacted at email: poojahattarki11@rediffmail.com.

Shridevi Soma working presently as H.O.D and Professor in Department of Computer Science and Engineering, Poojya Doddappa Appa College of Engineering, Kalaburagi. She has 18 years of teaching and 10 years of research experience. She completed her B.E., M.Tech., and Ph.D. in Computer Science and Engineering. Her research area includes digital image processing, pattern recognition, cloud computing, internet of things, and big data analytics. She published more than 30 research papers in above mentioned areas, also guiding research students. She can be contacted at email: shridevisoma@pdaengg.com.