Original Article
ISSN (Online): 2582-7472

ShodhKosh: Journal of Visual and Performing Arts
June 2024 5(6), 2697–2706

INTEGRATED OPTIMIZED DEEP LEARNING AND REINFORCEMENT LEARNING FOR FIBER FLAWS DETECTION

Dr. B. Vinothini [1]

[1] Assistant Professor and Head, Department of Computer Applications, St. Joseph's College for Women, Tirupur

ABSTRACT

The most challenging task in the cotton business is finding Fabric Faults (FFs) and refining material durability appropriately. To alleviate this, an Enhanced Pairwise-Potential Activation Layer in Optimized Multi-Criteria Convolutional Neural Network (EPPAL-OMCCNN) model was created, which considers a multi-objective active sampling strategy for annotation and tuning CNN for FF detection. But, it needs to predict historical and new kinds of unknown FF patterns accurately. So, this article introduces a deep Reinforcement Learning (RL) scheme into the EPPAL-OMCCNN model to predict new unknown FFs with the help of prior knowledge. At first, the multi-objective sampling strategy is applied to the fabric image database to label more influential images. Then, these images are used to construct the Optimized CNN (OCNN) with the RL model, which is trained by the fabric defect characteristics to predict the new unknown fabric pattern defects precisely. Finally, the experimental results exhibit that the EPPAL-OMCCNN-RL model on the TILDA set accomplishes 97.58% accuracy contrasted with the different deep learning-based FF detection models.

Keywords: Fabric Faults, Defect detection, EPPAL-OMCCNN, Multi-Objective Sampling, Deep Reinforcement Learning

Corresponding Author

Dr. B. Vinothini,
vinothini.prabhu@gmail.com

DOI
10.29121/shodhkosh.v5.i6.2024.3319

Funding: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

1. INTRODUCTION

The cotton business has become a well-recognized international business. By eliminating 45–65% of defects, operational costs can be reduced. Because a fabric fault is prevented or corrected after it has been noticed, weavers will often check the cotton material for extreme design flaws in modern looms [1-3]. The apparel trade has advanced toward highly autonomous material testing. The only way to increase dependability for textiles is often through product testing, which aids in the speedy and effective resolution of very minor flaws [4]. Comparing the ordinary emerging fiber screening to the traditional analysis, the identification rate will increase by nearly 80% [5-6]. There are several different recognition methods for FFs, including probabilistic, experiential, structured, composite, training-based, and design-based. These methods may be costly and prone to mistakes, connected with specific flaws, and incompatible with differences in fabric structure and efficiency. An improved model aims to achieve high durability in handling irregularities in textile structures and deformity classes [7].

In the early 20th century, deep learning models, like CNN, were used to successfully separate cloth themes [8]. Compared to a textured fabric's recurrent module, they have proved less effective in spotting flaws. Fully Convolutional Networks (FCN), U-Net, and other CNN genres control crucial modules such as convolution, pooling, and activation phases, where the pooling can prevent overfitting issues and minimize dimensionality. It extracts features along a whole contextual correlation and linguistic information that are inefficient for mapping appropriate image attributes [9].

Since they are shown by fewer pixel intensities, some FFs are thought of as restricted patterns. Some cotton analogies, such as overlap, impairment, coarse sets, etc., frequently only comprise inadequate pixels and create a very imbalanced dataset [10]. To enhance restricted pattern localization, CNN's performance was reconfigured by alternative practice. To address insufficient information on fiber pictures, CNN does not have many convolutional units, hence pooling is required to maintain image preservation. From this perspective, the PPAL-CNN model [11] was presented to recognize FFs with the aid of probabilistic loss values. The CRF must be given a previous probability in place of being learned. It was difficult to establish intricate connections among FF labels if there were numerous or manifested associations.

So, an EPPAL-CNN technique [12] was established to address this issue and resolves the challenging pattern correlation of FFs. The CRF was first enhanced using external memory rules that were supported by the memory channels, enabling CRFs to comprehend the regional features of FFs. The linkages between cotemporal classes are included in the DCRF's factorial structure, which also creates restricted probability correlations between different classes. Using external memory, a higher-order Markov correlation across classes was created. On the other hand, the gradient-based optimization approaches used to train the weights of CNN have an unusual tendency toward convergence, which leads to ineffective categorization. To address CNN's unwanted convergence, an EPPAL-OCNN model was proposed [13], which contains an individual weight optimization approach based on NWM-Adam. The fact that past gradients were given more memory than present gradients served as the inspiration for the development of this exponential smoothing average version. The detective model for each flaw must be created more slowly even as the rate of identification improves.

Additionally, acquiring a large number of annotated examples necessary for training the OCNN model takes additional effort. An EPPAL-OMCCNN model was subsequently suggested to reduce the time needed to produce more annotated samples for FF detection. Multi-objective active deep learning was used in this novel model to reduce the cost of manual labeling to a predetermined range [14]. The OCNN structure was first constructed using some randomly chosen data. The OCNN structure was then modified by suggesting more influential examples for user labeling and updating the learning set. The dependability of unannotated samples is assessed across all epochs using two separate criteria: annotated samples, and the current structure. To avoid data redundancy, the initial aim makes use of density and relevance to assess the trustworthiness of unannotated samples. For the second aim, the ambiguity and tag-based factors are used to assess the dependability of unannotated pictures, which in turn speeds up the convergence of the OCNN and reduces the efficiency variance between labels. Conversely, this model must adapt to predict the historical and novel category of defect patterns. Therefore, in this paper, a deep Reinforcement Learning (RL) scheme is integrated with the EPPAL-OMCCNN model to predict various kinds of fabric defect patterns simultaneously. In this study, the RL network is incorporated with the OMCNN, which is modified by the samples annotated by the multi-objective sampling strategy. An important feature of RL is that an agent captures desirable attributes. This indicates that it gradually adjusts previous knowledge and skills or learns new characteristics. RL also makes utilization of trial-and-error learning, which is a crucial component. As a result, full knowledge or environment management is not needed by the RL, whereas it merely requires interacting with the environment and obtaining data. During learning, the samples are provided sequentially and utilized to progressively modify the agent behavior. Thus, this RL with the EPPAL-OMCCNN model can enhance the accuracy of detecting new FF patterns according to prior experience.

The rest of this paper can be outlined: The recent FF identification studies are discussed in Section II. The EPPAL-OMCCNN-RL model is described in Section III, and its performance is shown in Section IV. Section V summarizes the results and suggests improvements.

2. LITERATURE SURVEY

Mo et al. [15] developed a Weighted Double-Low-Rank Decomposition (WDLRD) scheme to identify fabric defects by allocating various weights to the matrix singular values. At first, the background matrix and defect matrix were

normalized by the weighted nuclear norm to preserve the most relevant features. A defect prior map was created to identify fabric defects. But, its accuracy was detecting extremely small faults was not effective. Because the defect matrix was low rank, a few real-fault areas were lost.

Peng et al. [16] designed a robust Priori Anchor CNN called (PRAN-Net) to identify fabric defects. Initially, Feature Pyramid Network (FPN) was utilized by the decided multi-scale attribute maps to reserve more detailed data on small defects. Then, a trick was applied to create sparse Priori anchors depending on fabric faults ground truth boxes rather than predetermined anchors to locate complex defects precisely. Moreover, a categorization network was utilized to categorize and refine the location of the fabric defects. However, it needs DRL networks to identify many small defects.

Liu et al. [17] developed an effective weakly supervised shallow network, namely DLSE-Net with Link-SE (L-SE) and Dilation Up-Weight Class Activation Map (DUW-CAM) to identify fabric defects. A 3-branch framework was adopted to lessen the semantic gap created by the link of various layers and the L-SE unit was used to enhance the interpretation ability. Also, dilated convolution and attention strategies were introduced to suppress the background and enrich the defect areas. But, the accuracy was not efficient to identify tiny regions of fabric defects.

Jun et al. [18] developed a training-based model for automatically identifying fabric faults. Initially, a predetermined-size square slider was used to crop the actual image and an enhanced histogram normalization was applied to enrich all cropped images. Moreover, the Inception-V1 structure was used to predict the appearance of defects in the local region and the LeNet-5 framework was applied as a voting system to identify the category of the fabric defects. But, it was only approximately detected fabric defects, therefore it was hard to precisely find the cause of the defects.

Almeida et al. [19] developed a rapid and automated defect identification method using a custom CNN. Initially, fabric images were collected and pre-processed by histogram normalization to strengthen the fault area. After that, a modified CNN was applied to identify the fabric faults. But, the database was limited and less effective for identifying more complex defects.

Hu et al. [20] designed an unsupervised printing defect identification scheme by processing the disparity map between the test and the reference images. To achieve this, a Content-Based Image Retrieval (CBIR) scheme was introduced to restore the reference sample, which involves an image corpus, a Convolutional Denoising Auto-Encoder (CDAE) and a Hash Encoder (HE). The CDAE was used to extract a reliable feature interpretation of the patterns and the HE was used to index the feature vectors to binary code when preserving their similarity. Using the restored reference image, the fault was detected by employing the Tsallis entropy thresholding and opening process on the disparity map. However, the accuracy and stability of the detection were not satisfactory.

Wu et al. [21] designed a Wide-And-Light Network (WALNet) model depending on Faster Residual-CNN (FR-CNN) to identify usual fabric faults. In this model, the feature extraction network was enhanced by the dilated convolution unit, which utilizes a multi-scale convolution kernel to adapt to faults of various dimensions and learn the desired traits. Also, high-level semantic traits were merged with low-level detail characteristics using a skip-path and a sequence of anchor frames was aimed to identify the multi-scale fabric defects. However, it requires more annotated samples, which takes more time.

Liu et al. [22] designed an improved YOLOv4 model, which adopts a novel Spatial Pyramid Pooling (SPP) structure to identify fabric defects. Initially, the anchor was split based on the defect features and the CLAHE was applied to process the image to eliminate inappropriate color data and enrich the image contrast. Moreover, the SPP structure was enhanced by soft pooling and applied to detect fabric defects. However, it was not effective in detecting tiny faults.

Xiang et al. [23] presented an online identification of fabric defects depending on the improved CenterNet with deformable convolution. Also, an implicit Feature Pyramid Network (i-FPN) was adopted to improve the identification efficiency for tiny faults and accelerate the process. However, false identifications exist since wrinkles and imperfections were extremely analogous. As well, a few faults merely contain a minimal quantity of learning samples so it was complex for the framework to localize and detect them.

Huang et al. [24] developed a semantic segmentation model called Repeated Pattern Defect Network (RPDNet) utilizing a continuous pattern evaluation scheme for pixel-level identification of fabric defects. First, the periodic features related to possible defect regions were captured by obtaining continuous pattern details and appropriate guidance of the network in a high-level semantic space. After that, semi-supervised training was applied to learn those features and identify fabric defects. However, this model fails to sufficiently handle continuous pattern-free images.

Jing et al. [25] designed a very effective CNN called Mobile-UNet to establish an end-to-end fabric defect identification. The mid-frequency harmonizing error value was utilized to solve the data imbalance problem. Also, a depth-wise separable convolution was adopted to decrease the complexity cost and network dimension. Since a supervised scheme, this model necessitates a huge quantity of manually labeled samples, resulting in a longer detection time.

3. PROPOSED METHODOLOGY

In this section, the EPPAL-OMCCNN-RL model is described briefly. Figure 1 illustrates the block diagram of the presented study to identify FFs.

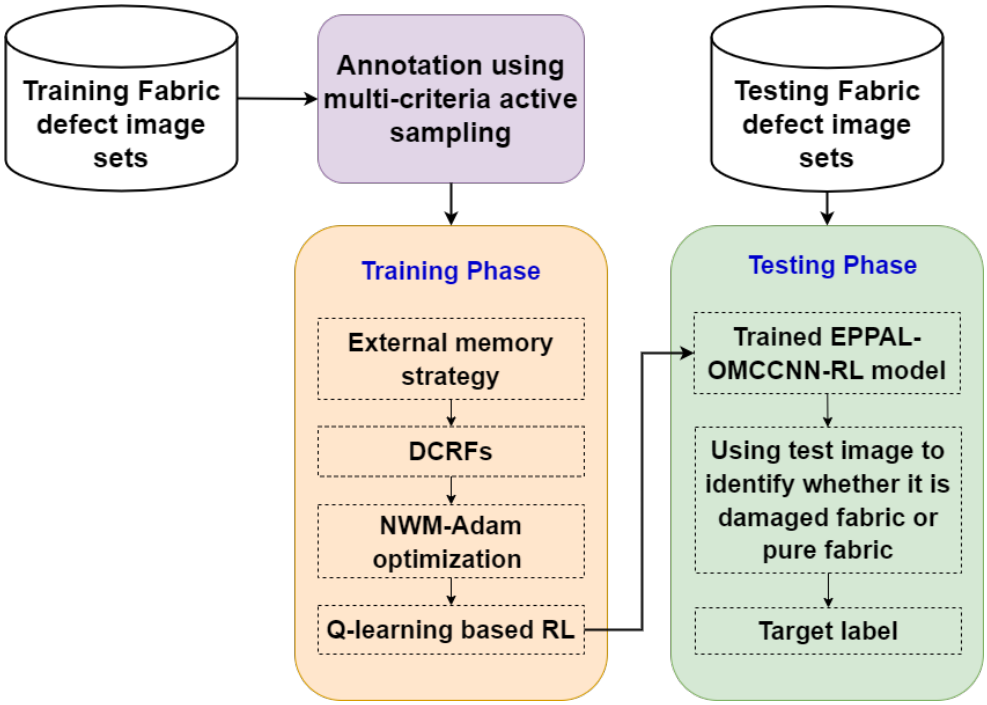

Figure 1. Block Diagram of Presented Study for FFs Detection

Consider the learning fabric image set $\mathcal{D} = \{x_i, y_i\}_{i=1}^{N}$, where x_i is i^{th} example in $\mathcal{D}$ and contains a string: $\{x_{i1}, \dots, x_{iT}\}$ and y_i is their corresponding tags $\{y_{i1}, \dots, y_{iT}\}$. Then, the multi-criteria active sampling strategy is performed to label the new unknown fabric images and find the most influential image sets. Those are further used to construct the EPPAL-OMCCNN-RL model and in the training phase of EPPAL-OMCCNN-RL, each x_t represents the temporal traits in the input example with their corresponding y_t.

In this EPPAL-OMCCNN-RL, external memory strategies are initially combined with the CRF to capture the local features. Also, the NWA-Adam optimization is applied to drive the rapid convergence of RL-based OCNN to predict the new FFs by considering prior knowledge. The description of RL is presented below.

3.1 Reinforcement Learning Model

The components of the typical RL learning model are an agent, an environment, a limited state space S, a set of activities A the agent may access, and a bonus variable: $S \times A \to R$. The main idea behind RL is to teach the agent to make better judgments through trial-and-error interactions with the environment. According to the analysis of the current condition s_t in the environment, the agent takes into consideration an action a_t during each outcome interval t. The condition of the atmosphere will change to the new condition s_{t+1} when the action is completed. The agent will simultaneously receive a bonus r_t that represents the variety of state transitions. This kind of agent-atmosphere process occurs often.

The RL agent is a long-term decision-maker; thus, it intends to boost its projected group bonuses over an interval: $\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t]$, where $\gamma \in (0, 1]$ defines a variable lowering prospective bonus. It achieves its goal by instructing the agent on how to choose acceptable actions under different conditions. Q-learning is one of the most often used RL models

without modeling. It does not require prior knowledge of the network, such as the likelihood of state transition. It learns how to make defensible judgments based on encounters.

The agent preserves a Q-factor $\big(Q(s,a)\big)$ for every condition-action set that represent the projected long-term bonus when taking a at s. The agent may comprehend the anticipated Q-factor for all actions at the current condition relying on this Q-factor. The agent also decides which actions should be taken into account to achieve the best long-term group bonuses. Each time an event occurs, the Q-factor is repeatedly modified by

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha * \left[r(s_t,a_t) + \gamma * \max_a Q(s_{t+1},a) - Q(s_t,a_t)\right] \qquad (1)$$

In Eq. (1), $\alpha \in (0,1]$ is the learning rate, $r(s_t,a_t)$ is the bonus obtained by a_t in s_t and γ is the deduction percentage. The Q-learning strategy generally presents a statistical formulation of the Q-factor. However, managing complex control concerns involving several states and actions are not appropriate. DRL uses the OCNN to build the relationship between these state-activity pairings and their associated Q-factor to address this issue.

The deep Q-learning engine, which is based on the OCNN, is also created to determine the agent's optional activities. The RL's role in the agent-atmosphere interface network is shown in Figure 2.

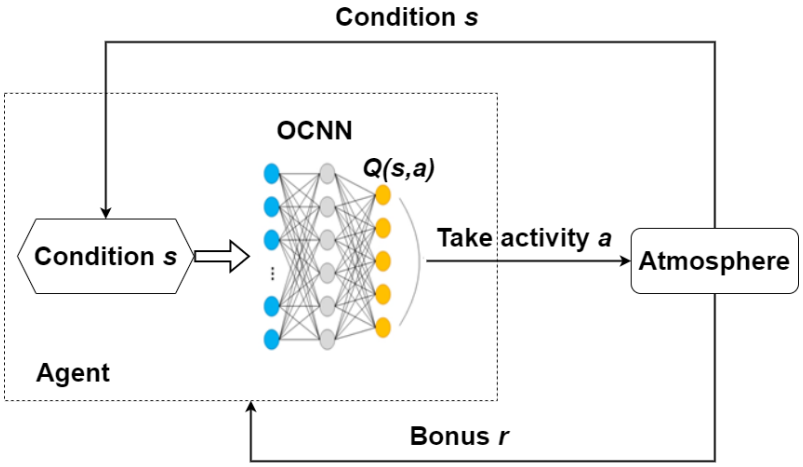

Figure 2. Schematic Representation of RL with OCNN Model

Training process

For all decision intervals t_j for j^{th} characteristic, the RL agent completes a_j relying on s_j. After obtaining the immediate bonus r_j and s_{j+1}, the changing profiles (s_j, a_j, r_j, s_{j+1}) is added to the repetition memory Δ with capability N_Δ. In all instants when Q-learning modification occurs, weight and bias variables (w,θ) of the OCNN are updated using the mini-batch that comprises constant arbitrary examples S_Δ from Δ. To minimize the computation period, this modification might occur all U decision intervals $(U \geq 1)$. The link between training observation and sequence interactions is disrupted by the interaction replay rule, allowing the agent to adapt from random transition examples rather than sequence interactions and decreasing the disparity of fine-tuned parameters. Because each example may be chosen several times to adjust parameters, training details are also used extensively.

Two CNN—the target and analysis networks—are used simultaneously to further reduce the divergence and oscillations of OCNN variables during learning phases. These two networks have a comparable design but different variables. To create the necessary Q-factors, the target network is used when refining Q-learning. Target networks have the notable characteristic of being temporarily frozen networks, whereas the analysis network keeps the new variables and is used to forecast Q-factors. Thus, based on the Q-factors, the FFs are predicted properly from the new unknown examples.

Algorithm:

Input: Training dataset $\mathcal{D} = \{x_i, y_i\}_{i=1}^{N}$

Output: FF and pure fabrics images

Begin

Initialize the maximum epoch T, and the number of chosen images in all epochs N;

Split $\mathcal{D}$ into the annotated image collection $\mathcal{D}_L^t$ and unannotated image collection $\mathcal{D}_{UL}^t$;
Obtain the new annotated dataset and train the EPPAL-OMCCNN-RL model;
for$(each\ training\ example)$
Initialize α, γ, learning rate β, initial training period τ, mini-batch S_Δ, replay time η;
Initialize Δ with N_Δ;
Initialize analysis and desired activity-Q with random variables, w and θ;
for$\left(each\ new\ example\ j\ at\ t_j\right)$
Randomly select an activity; or else, $a_j = \underset{a}{\operatorname{argmax}}\, Q\left(s_j, a; w, \theta\right)$;
Train j according to a_j, obtain r_j and condition change at consecutive decision period t_{j+1} with s_{j+1};
Add change $\left(s_j, a_j, r_j, s_{j+1}\right)$ to Δ;
if$(j \geq \tau\ and\ j \equiv 0 \bmod \beta)$
if$(j \equiv 0 \bmod \eta)$
Set $\hat{Q} = Q$;
end if
Randomly choose examples S_Δ from Δ;
for$(each\ change\ (s_k, a_k, r_k, s_{k+1})\ in\ S_\Delta)$
$target_k = r_k + \gamma * \underset{a'}{max}\, \hat{Q}(s_{k+1}, a'; w', \theta')$;
Adjust OCNN variables w, θ with a loss value of $target_k - Q(s_k, a_k; w, \theta)^2$;
end for
end if
end for
Predict the new FF patterns;
end for
Validate the classifier efficiency;
End

IV. EXPERIMENTAL RESULTS

The effectiveness of the EPPAL-OMCCNN-RL is examined in this section by using MATLAB 2017b to design it and contrast it with the current models: EPPAL-OMCCNN [14], PRAN-Net [16], DLSE-Net [17], WALNet [21], improved CenterNet [23] and Mobile-UNet [25]. The Irish Longitudinal Study on Ageing (TILDA) picture corpus [26], which includes 7 labels of fibres with faults and 1 label of fibres without flaws, is taken into consideration in this assessment. The observation is that a whole corpus has 3200 TIF visuals that are 1.2GB in size. The TILDA collection includes detailed descriptions of each image's defects. 1100 visuals are employed for testing, and 2100 images are employed for training. The confusion matrix of EPPAL-OMCCNN-RL on test set for recognizing FFs.

Table 1. Confusion Matrix for EPPAL-OMCCNN-RL on Test Set

	Detected label		
Real label		Pure	Damaged
	Pure (550 for each class)	TP **538**	FN **12**
	Damaged (550 for other class)	FP **12**	TN **538**

4.1 Accuracy

It measures the ratio of correctly identified damaged to pure fiber samples.

$$Acc = \frac{True\ Positive\ (TP)+True\ Negative(TN)}{TP+TN+False\ Positive\ (FP)+False\ Negative(FN)} \times 100\%$$

The number of damaged fibre samples that are accurately identified as damaged is defined by TP, whereas the number of pure fibre samples that are precisely identified as healthy is defined by TN. Additionally, FP defines the number of damaged fibre samples that were mistakenly identified as pure, whereas FN defines the number of pure fibre samples that were mistakenly identified as damaged.

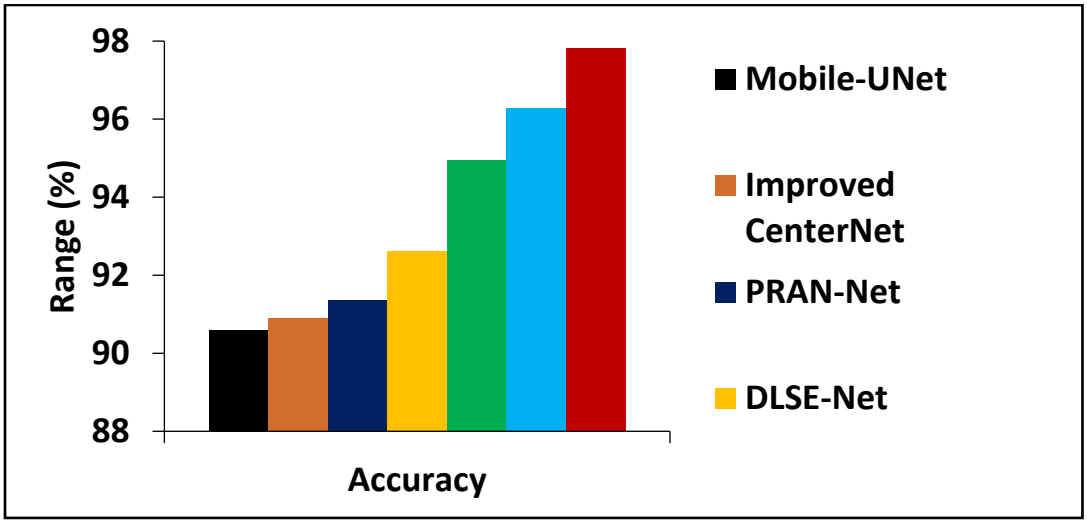

Figure 3. Comparison of Accuracy

Figure 3 depicts the accuracy (in %) values for various existing and proposed FF detection models. It scrutinizes that the accuracy of EPPAL-OMCCNN-RL is 7.99% better than the Mobile-UNet, 7.6% better than the improved CenterNet, 7.08% better than the PRAN-Net, 5.61% better than the DLSE-Net, 3.02% better than the WALNet and 1.61% better than the EPPAL-OMCCNN models. Accordingly, this EPPAL-OMCCNN-RL model will enhance the accuracy of detecting the FFs compared to the other models.

4.2 Precision

It is the percentage of properly declared damaged fiber samples.

$$Pre = \frac{No.of\ properly\ recognized\ defective\ fibers}{No.of\ properly\ recognized\ defective\ fibers+No.of\ improperly\ recognized\ defective\ fibers}$$

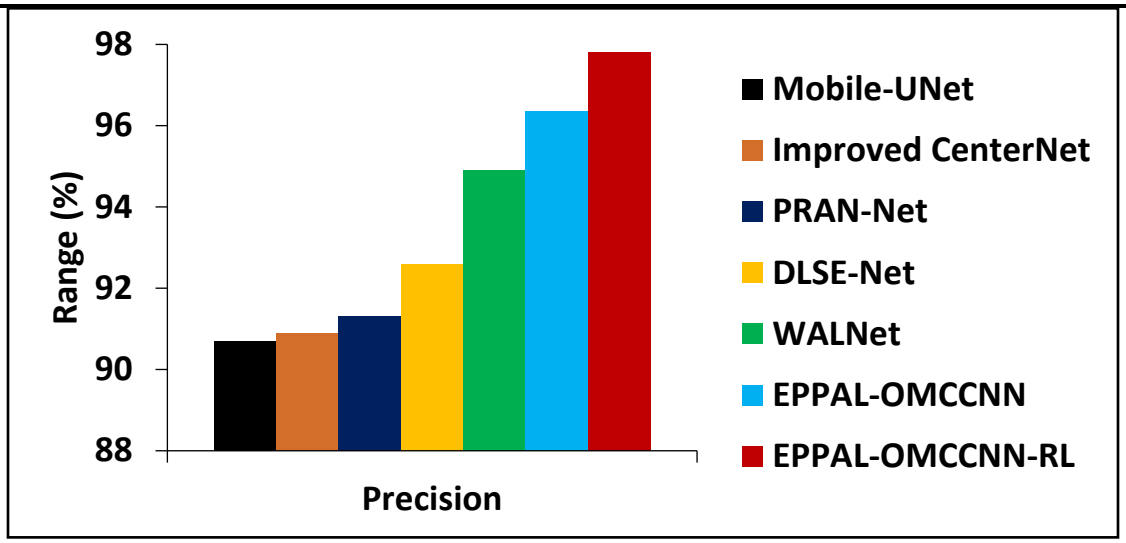

Figure 4. Comparison of Precision

Figure 4 shows the precision (in %) values for various FF detection models. It indicates that the precision of EPPAL-OMCCNN-RL is 7.84% superior to the Mobile-UNet, 7.59% superior to the improved CenterNet, 7.12% superior to the PRAN-Net, 5.62% superior to the DLSE-Net, 3.06% superior to the WALNet and 1.49% superior to the EPPAL-OMCCNN. So, this EPPAL-OMCCNN-RL increases the precision of detecting the FFs than the other models.

4.3 Recall

It is the percentage of damaged fiber images that are correctly identified as damaged.

$$Rc = \frac{No.of\ properly\ recognized\ defective\ fibers}{No.of\ properly\ recognized\ defective\ fibers + No.of\ improperly\ recognized\ non-defective\ fibers}$$

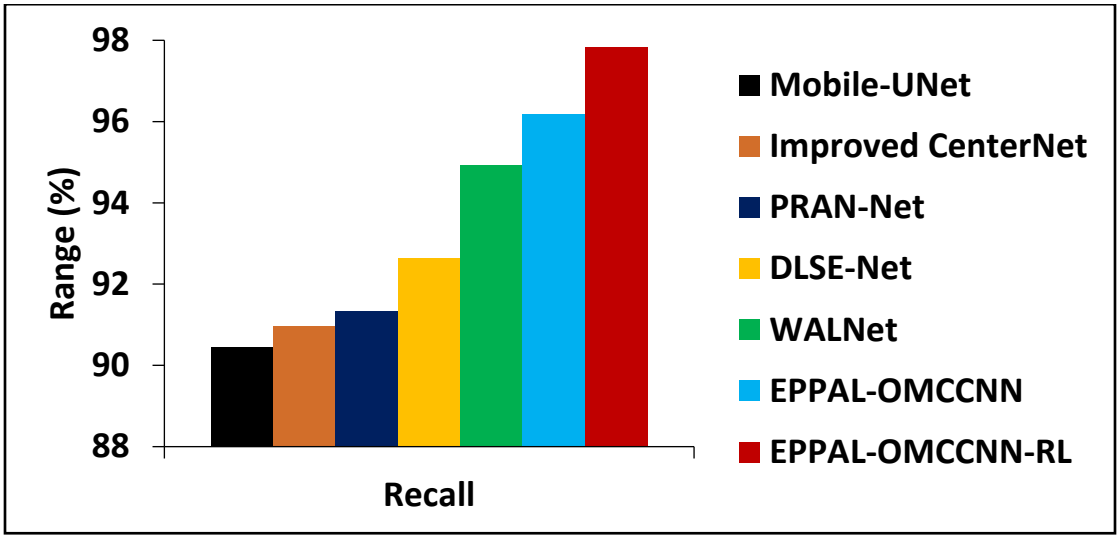

Figure 5. Comparison of Recall

Figure 5 demonstrates the recall (in %) values for various existing and proposed FF detection models. It observes that the recall of EPPAL-OMCCNN-RL is 8.15% higher than the Mobile-UNet, 7.54% higher than the improved CenterNet, 7.11% higher than the PRAN-Net, 5.59% higher than the DLSE-Net, 3.06% higher than the WALNet and 1.69% higher than the EPPAL-OMCCNN models. Thus, this EPPAL-OMCCNN-RL model can maximize the recall of detecting the FFs compared to the other models.

4.4 F-measure

It is determined by

$$F - measure = 2 \times \frac{Pr \cdot Rc}{Pr + Rc}$$

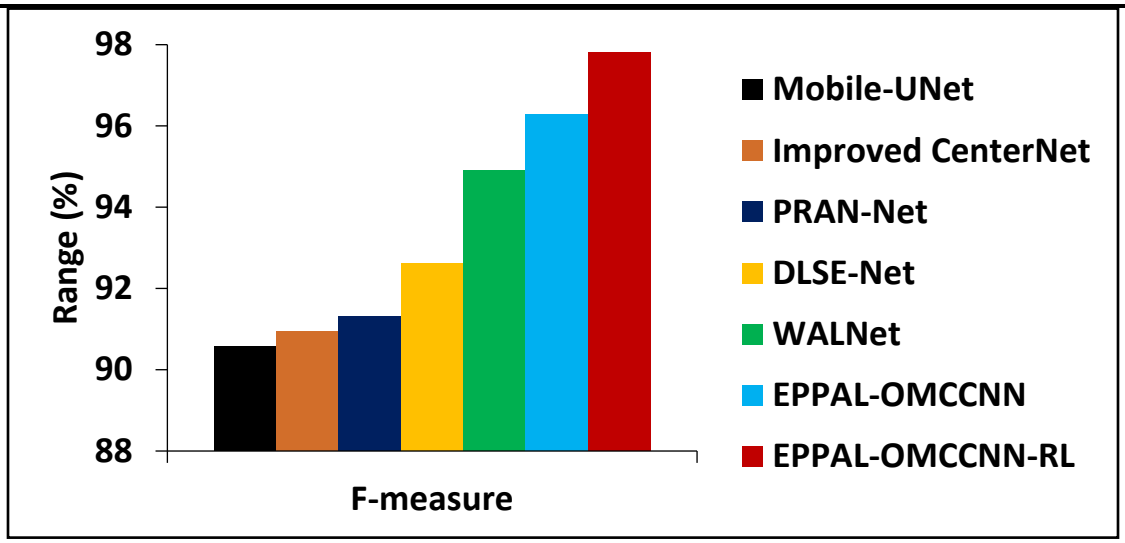

Figure 6. Comparison of F-measure

Figure 6 displays the f-measure (in %) values for various existing and proposed FF detection models. It indicates that the f-measure of EPPAL-OMCCNN-RL is 7.99% higher than the Mobile-UNet, 7.57% higher than the improved CenterNet, 7.11% higher than the PRAN-Net, 5.6% higher than the DLSE-Net, 3.06% higher than the WALNet and 1.59% higher than the EPPAL-OMCCNN models. Thus, this EPPAL-OMCCNN-RL model can maximize the f-measure of detecting the FFs compared to the other models.

4. CONCLUSION

In this paper, the EPPAL-OMCCNN-RL model was designed by integrating the RL model into the EPPAL-OMCCNN for FF detection. Primarily, the multi-objective active training was used to alleviate the class imbalance issue during the learning phase. Then, the EPPAL-OCNN-RL model was constructed using the annotated influential images for FF recognition. This model can predict new FF patterns based on prior knowledge regarding FFs with high accuracy. Finally, the experimental findings using the TILDA corpus accomplished that the EPPAL-OMCCNN-RL has 97.82% accuracy which was 5.43% greater than all the existing FF detection models.

CONFLICT OF INTERESTS

None.

ACKNOWLEDGMENTS

None.

REFERENCES

Wu, R., Zhang, J. X., Leaf, J., Hua, X., Qu, A., Harvey, C., ... & Marschner, S. (2020). Weavecraft: an interactive design and simulation tool for 3D weaving. *ACM Transactions on Graphs*, *39*(6), 210-216.

Negm, M., & Sanad, S. (2020). Cotton fibres, picking, ginning, spinning and weaving. In *Handbook of Natural Fibres*, Woodhead Publishing, pp. 3-48.

Atkar, A., Pabba, M., Sekhar, S. C., & Sridhar, S. (2021). Current limitations and challenges in the global textile sector. In *Fundamentals of Natural Fibres and Textiles*, Woodhead Publishing, pp. 741-764.

Dils, C., Kalas, D., Reboun, J., Suchy, S., Soukup, R., Moravcova, D., ... & Schneider-Ramelow, M. (2022). Interconnecting embroidered hybrid conductive yarns by ultrasonic plastic welding for e-textiles. *Textile Research Journal*, 1-20.

Rubino, F., Nisticò, A., Tucci, F., & Carlone, P. (2020). Marine application of fiber reinforced composites: a review. *Journal of Marine Science and Engineering*, *8*(1), 1-28.

Karuppannan Gopalraj, S., & Kärki, T. (2020). A review on the recycling of waste carbon fibre/glass fibre-reinforced composites: fibre recovery, properties and life-cycle analysis. *SN Applied Sciences*, *2*(3), 1-21.

Amor, N., Noman, M. T., & Petru, M. (2021). Classification of textile polymer composites: recent trends and challenges. *Polymers*, *13*(16), 1-27.

Gadri, S., & Neuhold, E. (2020). Building best predictive models using ML and DL approaches to categorize fashion clothes. In *International Conference on Artificial Intelligence and Soft Computing*, Springer, Cham, pp. 90-102.

Ghosh, A., Sufian, A., Sultana, F., Chakrabarti, A., & De, D. (2020). Fundamental concepts of convolutional neural network. In *Recent trends and advances in artificial intelligence and Internet of Things*, Springer, Cham, pp. 519-567.

Hu, Y., Long, Z., Sundaresan, A., Alfarraj, M., AlRegib, G., Park, S., & Jayaraman, S. (2021). Fabric surface characterization: assessment of deep learning-based texture representations using a challenging dataset. *The Journal of the Textile Institute*, *112*(2), 293-305.

Ouyang, W., Xu, B., Hou, J., & Yuan, X. (2019). Fabric defect detection using activation layer embedded convolutional neural network. *IEEE Access*, *7*, 70130-70140.

Vinothini, B., & Sheeja, S. (2021). Memory enhanced dynamic conditional random fields embedded pairwise potential CNN for fabric defects identification. *International Journal of Engineering Trends and Technology*, *69*, 227-234.

Vinothini, B., & Sheeja, S. (2022). Optimizing gradients weight of enhanced pairwise-potential activation layer in CNN for fabric defect detection. *Indian Journal of Computer Science and Engineering*, *13*(3), 688-696.

Vinothini, B., & Sheeja, S. (). Optimized multi-objective deep learning with enhanced pairwise-potential activation layer for fiber faults identification.

Mo, D., Wong, W. K., Lai, Z., & Zhou, J. (2020). Weighted double-low-rank decomposition with application to fabric defect detection. *IEEE Transactions on Automation Science and Engineering*, *18*(3), 1170-1190.

Peng, P., Wang, Y., Hao, C., Zhu, Z., Liu, T., & Zhou, W. (2020). Automatic fabric defect detection method using PRAN-net. *Applied Sciences*, *10*(23), 1-13.

Liu, Z., Huo, Z., Li, C., Dong, Y., & Li, B. (2021). DLSE-Net: a robust weakly supervised network for fabric defect detection. *Displays*, *68*, 1-10.

Jun, X., Wang, J., Zhou, J., Meng, S., Pan, R., & Gao, W. (2021). Fabric defect detection based on a deep convolutional neural network using a two-stage strategy. *Textile Research Journal*, *91*(1-2), 130-142.

Almeida, T., Moutinho, F., & Matos-Carvalho, J. P. (2021). Fabric defect detection with deep learning and false negative reduction. *IEEE Access*, *9*, 81936-81945.

Hu, X., Fu, M., Zhu, Z., Xiang, Z., Qian, M., & Wang, J. (2021). Unsupervised defect detection algorithm for printed fabrics using content-based image retrieval techniques. *Textile Research Journal*, *91*(21-22), 2551-2566.

Wu, J., Le, J., Xiao, Z., Zhang, F., Geng, L., Liu, Y., & Wang, W. (2021). Automatic fabric defect detection using a wide-and-light network. *Applied Intelligence*, *51*(7), 4945-4961.

Liu, Q., Wang, C., Li, Y., Gao, M., & Li, J. (2022). A fabric defect detection method based on deep learning. *IEEE Access*, *10*, 4284-4296.

Xiang, J., Pan, R., & Gao, W. (2022). Online detection of fabric defects based on improved CenterNet with deformable convolution. *Sensors*, *22*(13), 1-18.

Huang, Y., & Xiang, Z. (2022). RPDNet: automatic fabric defect detection based on a convolutional neural network and repeated pattern Analysis. *Sensors*, *22*(16), 1-17.

Jing, J., Wang, Z., Rätsch, M., & Zhang, H. (2022). Mobile-Unet: An efficient convolutional neural network for fabric defect detection. *Textile Research Journal*, *92*(1-2), 30-42.

Workgroup on texture analysis of DFG's. TILDA textile texture database. Available online: https://lmb.informatik.unifreiburg.de/resources/datasets/tilda.en.html (accessed on 15 September 2022)