Explainable Contrastive and Cost-Sensitive Learning for Cervical Cancer Classification

Abstract—**This paper proposes an efficient system for classifying cervical cancer cells using pre-trained convolutional neural networks (CNNs). We first fine-tune five pre-trained CNNs to minimize the overall cost of misclassification by prioritizing accuracy for certain classes that have higher associated costs or importance. To further enhance the performance of the models, supervised contrastive learning is included to make the models more adept at capturing important features and patterns. Extensive experimentations are conducted to evaluate the proposed system on the SIPaKMeD dataset. The experimental results demonstrate the effectiveness of the developed system, achieving an accuracy of 97.29%. To make our system more trustworthy, we have employed several explainable AI techniques to interpret how the models reached a specific decision. The implementation of the system can be found at - https://github.com/isha-67/CervicalCancerStudy.**

Index Terms—**Cervical Cancer, Cost-Sensitive Learning, Contrastive Learning, SIPaKMeD, XAI, LIME, GradCAM**

I. Introduction

Cervical cancer, the world's third-most common type of cancer, is the leading cause of cancer-related deaths in women [1]. However, unlike other cancers, cervical cancer can be prevented. Many cytology-based screening programs can detect cervical cell abnormalities before they become cancer. One of the most popular screening tests for cervical cancer is the Papanicolaou test [2], also known as the Pap test. However, a challenge in detecting cancerous cells from pap smear images is that it requires highly qualified pathologists to analyze the results, which is hard to find, especially in developing countries. This is where computer-supported tools, such as deep learning, particularly CNNs, can be used to identify patterns relevant to medical diagnosis and perform image classification to make up for these limitations.

Convolutional Neural Networks (CNNs) automatically extract important features from input images, eliminating the need for manual feature extraction, which is highly effective in medical image classification for improving accuracy compared to traditional methods. Several existing works [3], [4] have used convolutional neural networks (CNNs) to classify and detect cervical cancer. But, a great concern associated with these studies is that how the deep learning models make such decisions to classify the images cannot be explained. Some studies on medical images [5] [6] have employed convolutional neural networks (CNNs) for image classification, where they have also utilized explainable AI techniques to explain how their models reached a specific conclusion.

In our study, we developed a deep learning-based system for classifying cancer cell types using the SIPaKMeD Dataset [7]. We fine-tuned five different classifiers, refining their ability to classify cancer cell types. In order to make the classifiers more attuned to real-world classification errors, we included cost-sensitive learning. We also applied the concept of supervised contrastive learning [8] so that our models extract more discriminative and representative features and enhance the classification accuracy. In addition, we implemented explainable AI techniques to provide insights into our models' decision-making processes, aiming to build trust in automated medical image classification.

II. Related Works

A. Ghoneim, G. Muhammad, and M. S. Hossain [9] developed a cervical cancer cell detection and classification system using pre-trained convolutional neural networks (VGG16 and CaffeNet) utilizing the Herlev database that achieved 99.7% accuracy in the binary classification problem and 97.2% in the multi-class classification problem. Rahaman et al. [10] developed a hybrid deep feature fusion (HDFF) technique to enhance cervical cell image classification performance using four pre-trained models: VGG16, VGG19, ResNet-50, and XceptionNet. They achieved maximum classification accuracies of 99.85%, 99.38%, and 99.14% on binary and 7-class tasks on the SIPaKMeD dataset. Pramanik et al. [4] developed an ensemble method that minimizes the error between observations and ground truth, outperforming Inception V3, Inception ResNet V2, and MobileNetV2 models with an accuracy of 96.96%. Manna, R. Kundu, D. Kaplun, A. Sinitca, and R. Sarkar [11] introduced a classification model using ensemble methods, combining three CNN architectures: Inception v3, DenseNet-169, and Xception. The model achieved high accuracy rates of 98.55% for a binary classification and, 95.43% for a 5-class classifier on the SIPaKMeD dataset and 99.23% on the Mendeley LBC dataset. A. Tripathi, A. Arora, and A. Bhan [3] classified cancer cell growth stages using ResNet-50, ResNet-152, VGG16, and VGG19 pre-trained models, with ResNet-152 achieving 94.89% highest accuracy. Hsieh et al. [12] proposed detecting bone metastases on bone scans using image classification and contrastive learning. The study showed that contrastive learning enhances the accuracy of deep learning models, with the ResNet-50 model having the highest accuracy of 94.30%. Ravi [13] developed an attention-cost-sensitive deep learning-based feature fusion ensemble meta-classifier technique for skin cancer classification using the HAM10000 dataset. He used cost weights to address data imbalance. The EfficientNetV2 model outperformed other models with a 96% accuracy rate.

III. Methodology

Figure 1 presents our proposed methodology that comprises fivr distinct steps: input data, image preprocessing, model training, performance evaluation, and the interpretation of the model performance. We describe each of the steps below in detail.

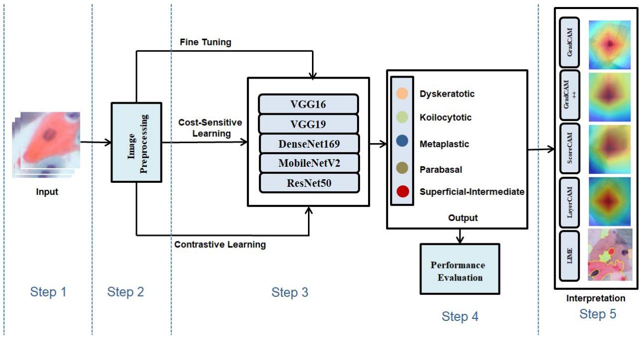

Fig. 1. Proposed Methodology

A. Input

The input to the system is the SIPaKMeD Dataset [7] consisting of pap smear images, where medical experts have manually defined the nucleus and the area of cytoplasm in each image and have labeled the images with expertise. It includes 4049 precisely cropped photos of isolated cells and 966 images of Pap smear slide cluster cells. The images were captured with an optical microscope CCD camera (OLYMPUS BX53F) [7]. There are five different class labels: *Superficial-Intermediate* (813 images), *Parabasal* (727 images), *Koilocytotic* (825 images), *Metaplastic* (793 images) and *Dyskeratotic* (813 images). The dataset was split into three sets: train, test, and validation. The train set comprises 80% of the total images. The remaining 20% of the images were further divided into test (80%) and validation (20%) sets, resulting in 2589 train images, 648 validation images, and 812 test images. We augmented only the original training set with various techniques, resulting in 18,123 images for a more diverse training dataset. Augmentations include affine transformations (rotation, translation, scaling, shearing, zooming, flipping, padding), noise injection, contrast adjustment, brightness modification, and pixel value changes.

B. Image Preprocessing

For consistency and compatibility, all dataset images were standardized to a uniform resolution. The dataset contained images ranging from 62 × 48 pixels to 531 × 553 pixels. Downsizing large images can hinder feature learning, while up-scaled and zero-padded small images add complexity. We determined the optimal resolution through scatter and density plots, considering computational resources and feature recognition. We calculated the dimensions of each image and created a 2D histogram to find the most common size, which was 98.5 × 108.9 pixels. We selected 110 × 110 pixels for uniformity, maintaining aspect ratios, facilitating further processing, and enhancing machine learning efficiency, ultimately improving performance and results.

C. Training

We performed three types of training on five different pre-trained CNN models: standard fine-tuning, fine-tuning with cost-sensitive learning, and fine-tuning with supervised contrastive learning. This section provides a brief overview of the models, cost-sensitive and contrastive learning.

1) Models: For our experiment purposes, we have utilized five pre-trained CNN architectures: ResNet-50 [14], MobileNetV2 [15], DenseNet-169 [16], VGG16 [17] and VGG19. We chose ResNet-50 as it is ideal for deep networks and has high accuracy due to skip connections. MobileNetV2 provides a balance between size and accuracy. DenseNet-169 promotes feature reuse, efficient parameter usage, and good accuracy. VGG16 and VGG19 are Simple, widely used architectures suitable for transfer learning and various image classification tasks. During training, we froze the first 86 layers of ResNet-50, the first 100 layers of MobileNetV2, the first 249 layers of DenseNet-169, the first 13 layers of VGG16, and the initial 17 layers of VGG19.

2) Cost-Sensitive Learning: Cost-Sensitive learning is an area of learning where the costs associated with class-imbalanced data are taken into account [18], [19]. Depending on the application, there are different ways to create the cost matrix. One way is to put class weights according to the distribution of class labels. The default log loss function, as represented by Equation 1, assigns equal weight to all classes in classification tasks, regardless of their distribution. This can introduce bias in cases of imbalanced data.

$$LogLoss = 1/N\sum_{i=1}^{N}[-(y_i log(\bar{y_i}) + (1-y_i) log(1-\bar{y_i}))] \quad (1)$$

To address this limitation, Equation 2 calculates the Weighted Log Loss by assigning proper weights to each class based on their distribution in unbalanced data.

$$\begin{aligned}\text{WeightedLogLoss} = \frac{1}{N}\sum_{i=1}^{N} & [-w_0(y_i \log(\bar{y_i})) \\ & + w_1((1-y_i)\log(1-\bar{y_i}))]\end{aligned} \quad (2)$$

As the SIPaKMeD dataset is slightly imbalanced, we assign the costs of *Dyskeratotic, Koilocytotic, Metaplastic, Parabasal* and *Superficial-Intermediate* class with values 0.996319018404908, 0.9842424242424243, 1.0213836477987421, 1.0278481012658227, and 0.9724550898203593 respectively in all the experiments related to cost-sensitive learning.

3) Supervised Contrastive Learning: Supervised contrastive learning [8] focuses on learning the representation of the instances in the dataset by minimizing similarity between negative pairs and maximizing similarity between positive pairs in the feature space. The contrastive loss function is designed to pull together similar instances in the feature space while pushing apart dissimilar instances. The steps involved in supervised contrastive learning are depicted in Figure 2. At first, we divide the augmented training dataset into batches of size 32. In a batch, images with the same class labels are considered positive pairs, and images with different class labels are considered negative pairs. The images of each batch are then passed through a base encoder, which is a pre-trained CNN model. The base encoder extracts high-level features from the input images and maps them into a continuous representation space. The representations obtained from the base encoder are then fed into a projection head. The projection head consists of fully connected layers that project the representations into a different space. This projection step helps to maximize the similarity between the representations of positive pairs while minimizing the similarity between representations of negative pairs. After the projection, the representations are passed through a frozen encoder. The frozen encoder is a copy of the base encoder with its weights fixed. It serves as a feature extractor, aiming to capture additional information from the representations. Finally, the frozen encoder's output is used to predict the labeled data.

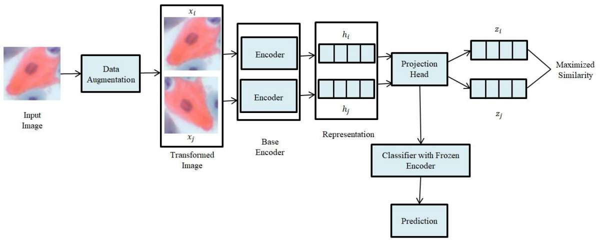

Fig. 2. Steps of Supervised Contrastive Learning

D. Performance Evaluation

To evaluate our system, we have used standard evaluation metrics for classification tasks such as weighted accuracy, weighted precision, weighted recall, and weighted F1 score. The weighted metrics were measured by considering the instance occurrences of the classes in the dataset.

E. Interpretation

In the final step, we leverage gradient-based and perturbation-based explainable AI techniques to make our system transparent and reliable. Gradient-based Explainability Analysis (XAI) uses gradients to understand how input features contribute to model predictions. GradCAM [20] is a class-discriminative localization technique that computes the importance score in the final convolutional layer of a CNN using gradients. GradCAM++ [21] takes both forward and backward gradients into account, whereas ScoreCAM [22] employs a global average pooling operation to calculate a single importance score for each feature map. LayerCAM [23] generates class activation maps by analyzing different layers of a deep neural network and identifying regions in an input image that contribute significantly to the model's classification decision. Perturbation-based XAI methods involve altering input data to observe the model's predictions. LIME, or Local Interpretable Model Agnostic Explanation, generates random perturbations to determine feature significance in classification [24]. It is model-agnostic, highlights positive superpixels, and uses color-coded visualizations to provide interpretable insights into the model's decision-making process.

IV. Evaluation

A. Hyper-Parameter Setting

All the experiments were performed on Google Colaboratory and Kaggle Notebook. The implementations were written using Python language, Tensorflow framework, and Keras Library. In all the experiments, we used *Softmax* as an activation function, *Adam* optimizer, a learning rate of $1e-3$ and a dropout rate of 0.50. We ran all the experiments for 50 epochs. While we incorporated *Categorical cross-entropy* for standard fine-tuning and fine-tuning with cost-sensitive learning, we utilized multi-class *n-pairs loss* for fine-tuning with contrastive learning. We also experimented with supervised *NT-Xent loss*, *triplet margin loss*, but multi-class *n-pairs loss* yields the best performance.

B. Experimental Results

TABLE I
Performance Comparison using Fine-Tuning

Classifier	Class	Weighted Precision	Weighted Recall	Weighted F1 Score	Accuracy
VGG16	Dyskeratotic	0.94	0.96	0.95	95.57%
	Koilocytotic	0.93	0.88	0.90	
	Metaplastic	0.95	0.96	0.96	
	Parabasal	0.99	0.99	0.99	
	Superficial-Intermediate	0.97	0.99	0.98	
VGG19	Dyskeratotic	0.92	0.94	0.93	94.21%
	Koilocytotic	0.90	0.87	0.88	
	Metaplastic	0.94	0.94	0.94	
	Parabasal	0.97	0.98	0.98	
	Superficial-Intermediate	0.98	0.99	0.99	
DenseNet-169	**Dyskeratotic**	**0.95**	**0.97**	**0.96**	**97.17%**
	Koilocytotic	**0.98**	**0.93**	**0.96**	
	Metaplastic	**0.95**	**0.98**	**0.97**	
	Parabasal	**0.99**	**0.99**	**0.99**	
	Superficial-Intermediate	**0.99**	**0.99**	**0.99**	
MobileNet-V2	Dyskeratotic	0.98	0.96	0.97	96.55%
	Koilocytotic	0.93	0.94	0.93	
	Metaplastic	0.98	0.95	0.97	
	Parabasal	0.97	1.00	0.98	
	Superficial-Intermediate	0.98	0.98	0.98	
ResNet-50	Dyskeratotic	0.94	0.96	0.95	94.58%
	Koilocytotic	0.89	0.90	0.89	
	Metaplastic	0.92	0.93	0.93	
	Parabasal	1.00	0.98	0.99	
	Superficial-Intermediate	0.99	0.96	0.98	

TABLE II
PERFORMANCE COMPARISON USING COST-SENSITIVE LEARNING

Classifier	Class	Weighted Precision	Weighted Recall	Weighted F1 Score	Accuracy
VGG16	Dyskeratotic	0.96	0.98	0.97	95.94%
	Koilocytotic	0.94	0.90	0.92	
	Metaplastic	0.95	0.95	0.95	
	Parabasal	0.99	0.99	0.99	
	Superficial-Intermediate	0.97	0.98	0.97	
VGG19	Dyskeratotic	0.96	0.93	0.94	94.09%
	Koilocytotic	0.88	0.90	0.89	
	Metaplastic	0.93	0.91	0.92	
	Parabasal	0.97	0.98	0.98	
	Superficial-Intermediate	0.97	0.99	0.98	
DenseNet-169	**Dyskeratotic**	**0.96**	**0.96**	**0.96**	**97.29%**
	Koilocytotic	**0.96**	**0.94**	**0.95**	
	Metaplastic	**0.96**	**0.98**	**0.97**	
	Parabasal	**1.00**	**0.99**	**0.99**	
	Superficial-Intermediate	**0.99**	**0.99**	**0.99**	
MobileNet-V2	Dyskeratotic	0.98	0.94	0.96	95.57%
	Koilocytotic	0.90	0.94	0.92	
	Metaplastic	0.97	0.93	0.95	
	Parabasal	0.97	1.00	0.98	
	Superficial-Intermediate	0.96	0.97	0.97	
ResNet-50	Dyskeratotic	0.91	0.98	0.94	94.21%
	Koilocytotic	0.91	0.87	0.89	
	Metaplastic	0.92	0.92	0.92	
	Parabasal	0.99	0.97	0.98	
	Superficial-Intermediate	0.98	0.97	0.97	

TABLE III
PERFORMANCE COMPARISON USING CONTRASTIVE LEARNING

Classifier	Class	Weighted Precision	Weighted Recall	Weighted F1 Score	Accuracy
VGG16	**Dyskeratotic**	**0.95**	**0.98**	**0.97**	**97.29%**
	Koilocytotic	**0.98**	**0.94**	**0.96**	
	Metaplastic	**0.96**	**0.97**	**0.97**	
	Parabasal	**1.00**	**0.99**	**0.99**	
	Superficial-Intermediate	**0.98**	**0.99**	**0.98**	
VGG19	Dyskeratotic	0.98	0.97	0.97	96.68%
	Koilocytotic	0.95	0.95	0.95	
	Metaplastic	0.94	0.96	0.95	
	Parabasal	0.99	0.97	0.98	
	Superficial-Intermediate	0.99	0.98	0.99	
DenseNet-169	Dyskeratotic	0.96	0.95	0.95	93.47%
	Koilocytotic	0.90	0.86	0.88	
	Metaplastic	0.92	0.93	0.92	
	Parabasal	0.93	0.98	0.95	
	Superficial-Intermediate	0.96	0.96	0.96	
MobileNet-V2	Dyskeratotic	0.95	0.96	0.95	95.81%
	Koilocytotic	0.95	0.93	0.94	
	Metaplastic	0.95	0.97	0.96	
	Parabasal	0.97	0.99	0.98	
	Superficial-Intermediate	0.98	0.96	0.97	
ResNet-50	Dyskeratotic	0.95	0.96	0.95	96.43%
	Koilocytotic	0.95	0.93	0.94	
	Metaplastic	0.96	0.98	0.97	
	Parabasal	0.98	0.99	0.99	
	Superficial-Intermediate	0.98	0.96	0.97	

We conducted three training experiments: standard fine-tuning, fine-tuning with cost-sensitive learning, and fine-tuning with supervised contrastive learning. The results are summarized in Tables I,II, and III. In the initial fine-tuning phase, as indicated in Table I, DenseNet-169 emerged as the top-performing model with an impressive accuracy of 97.17%. Conversely, VGG19 and ResNet-50 exhibited relatively lower performance, while VGG16 and MobileNetV2 achieved intermediate results. Moving on to fine-tuning with cost-sensitive learning, as shown in II, DenseNet-169 maintained its superiority but with only a slight improvement in its performance. However, in the case of fine-tuning with supervised contrastive learning, as shown in III, DenseNet-169's performance took a downturn, with VGG-16 emerging as the best classifier with a remarkable accuracy of 97.29%.

C. *Comparison with Existing Works*

Comparing our study to existing works (Table IV), it is evident that our approach outperforms several prior studies. Pramanik et al. achieved 96.96% accuracy using an ensemble approach, A. Manna, R. Kundu, D. Kaplun, A. Sinitca, and R. Sarkar reached 95.43% with another ensemble technique, and A. Tripathi, A. Arora, and A. Bhan obtained 94.89% through ResNet-152 fine-tuning. Our study surpassed these benchmarks, achieving 97.17% accuracy with DenseNet-169 fine-tuning. DenseNet-169 with cost-sensitive learning further improved the accuracy to 97.29%, while in contrastive learning, VGG16 excelled with an accuracy of 97.29% as well, showcasing superior performance compared to prior work.

TABLE IV
COMPARISON WITH PREVIOUS WORKS

Author	Methods	Accuracy
Pramanik et al. [4]	Fuzzy distance-based ensemble approach using Inception V3, Inception ResNet V2, and MobileNet V2	96.96%
A. Manna, R. Kundu, D. Kaplun, A. Sinitca, and R. Sarkar [11]	Ensemble technique using Inception v3, DenseNet-169, and Xception	95.43%
A. Tripathi, A. Arora, and A. Bhan [3]	Fine-tuning (ResNet-152)	94.89%
Our study	Fine-tuning (DenseNet-169) **Cost-Sensitive learning (DenseNet-169)** **Supervised Contrastive learning (VGG16)**	97.17% **97.29%** **97.29%**

D. *Interpretation using XAI*

We used neural network visualization toolkit[1] to use gradient-based visualization techniques like GradCAM, Grad-

[1] https://github.com/keisen/tf-keras-vis

CAM++, ScoreCAM, LayerCAM, and perturbation-based visualization technique LIME to visualize classifications.

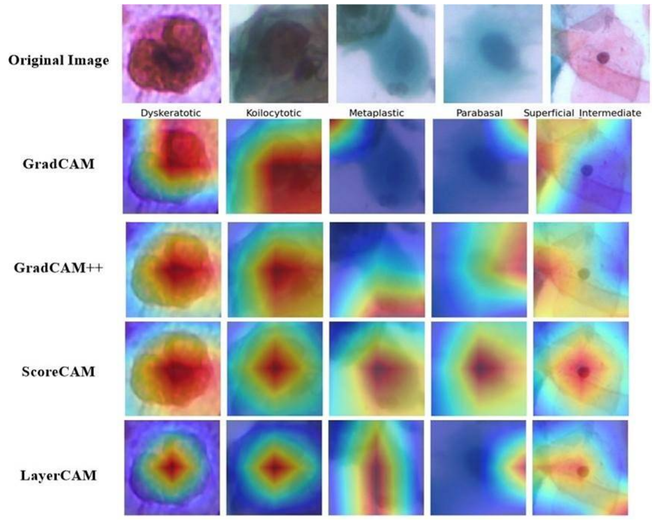

Fig. 3. Five sample outputs of correctly classified instances of DenseNet-169 using gradient-based XAI techniques

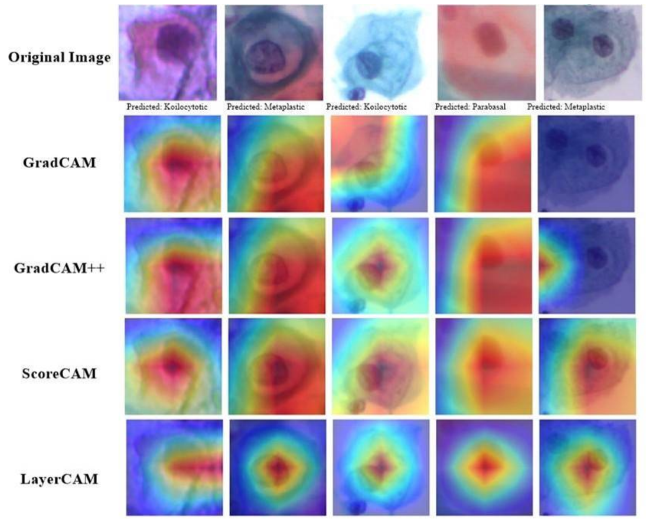

Fig. 4. Five sample outputs of misclassified instances of DenseNet-169 using gradient-based XAI techniques

Figure 3 represents five correctly classified images from five different classes, and 4 represents five misclassified images where advanced gradient-based visualization techniques were employed, including GradCAM, GradCAM++, ScoreCAM, and LayerCAM. By applying these methods, crucial areas within the images were highlighted, shedding light on the regions that influenced the classification. In 3, GradCAM and GradCAM++ were unable to generate satisfactory heatmaps for the *Metaplastic* and *Parabasal* classes. LayerCAM also struggled to produce an accurate heatmap for the *Parabasal* class. However, ScoreCAM proved to be effective in generating a reliable heatmap for the *Parabasal* class. The limitations observed in the performance of GradCAM, GradCAM++, and LayerCAM emphasize their difficulty in capturing the essential features and influential regions within the *Metaplastic* and *Parabasal* class images. By using a combination of techniques like GradCAM, GradCAM++, LayerCAM, and ScoreCAM, we were able to gain a comprehensive understanding of the model's behavior across different classes, identifying both strengths and weaknesses in its classification capabilities.

E. Perturbation-based Visualization

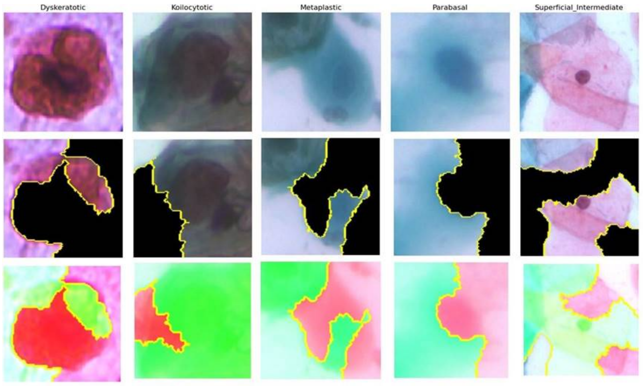

Fig. 5. Five sample outputs of correctly classified instances of DenseNet-169 using Perturbation-based Visualization technique LIME

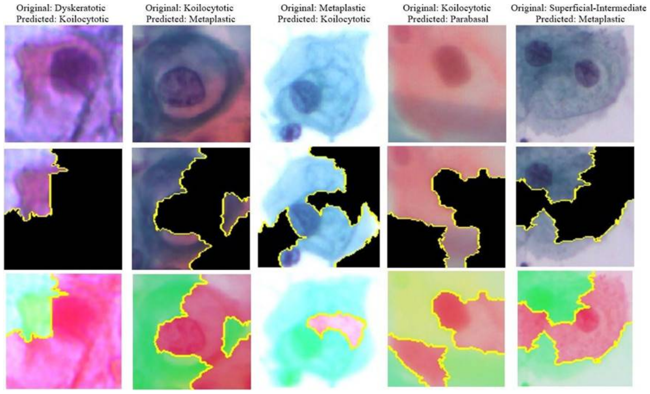

Fig. 6. Five sample outputs of misclassified instances of DenseNet-169 using Perturbation-based Visualization technique LIME

Figure 5 displays a set of five correctly classified images representing five different classes, while Figure 6 presents five misclassified images where the perturbation-based visualization technique, Local Interpretable Model-Agnostic Explanations (LIME) was used. LIME helped identify and highlight the important regions and features within these images that contributed to the incorrect classifications. By generating random perturbations of the input data, LIME identified the top superpixels that positively influence the predicted class. LIME generated two types of visualizations of the images: one that focuses solely on positive contributions and another that incorporates both positive and negative contributions. These visualizations allowed us to gain insights into potential weaknesses or biases present in the model's decision-making process. By highlighting the influential regions while preserving the rest of the image, LIME provides interpretable insights into the factors influencing the model's output. Moreover, LIME employed color-coded visualizations, with green representing the "pros" (positive contributions) and red representing the "cons" (negative contributions). This color scheme facilitates a clear understanding of the model's decision by visualizing the positive and negative aspects of the prediction. LIME's approach also gives interpretable insights into the model's decision-making process by showcasing the pixels that influenced the misclassifications.

V. Conclusion

This study utilizes deep learning techniques, including VGG16, VGG19, ResNet-50, MobileNetV2, and DenseNet-169, with a focus on cost sensitivity and supervised contrastive learning to improve cervical cancer cell classification. Evaluation metrics like precision, recall, F1 score, and accuracy are used to assess the system performance. We enhance interpretability with gradient and perturbation-based visualization, aiding trust in automated decisions. Our research showcases the potential of automated systems in cervical cancer detection, contributing to early prevention. Future work could involve broader dataset testing for generalization performance.

References

[1] M. Arbyn, X. Castellsagué, S. de Sanjosé, L. Bruni, M. Saraiya, F. Bray, and J. Ferlay, "Worldwide burden of cervical cancer in 2008," *Annals of oncology*, vol. 22, no. 12, pp. 2675–2686, 2011.

[2] K. Duraisamy, K. Jaganathan, and J. C. Bose, "Methods of detecting cervical cancer," *Advances in Biological Research*, vol. 5, no. 4, pp. 226–232, 2011.

[3] A. Tripathi, A. Arora, and A. Bhan, "Classification of cervical cancer using deep learning algorithm," in *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE, 2021, pp. 1210–1218.

[4] R. Pramanik, M. Biswas, S. Sen, L. A. de Souza Júnior, J. P. Papa, and R. Sarkar, "A fuzzy distance-based ensemble of deep models for cervical cancer detection," *Computer Methods and Programs in Biomedicine*, vol. 219, p. 106776, 2022.

[5] H. Panwar, P. Gupta, M. K. Siddiqui, R. Morales-Menendez, P. Bhardwaj, and V. Singh, "A deep learning and grad-cam based color visualization approach for fast detection of covid-19 cases using chest x-ray and ct-scan images," *Chaos, Solitons & Fractals*, vol. 140, p. 110190, 2020.

[6] M. Esmaeili, R. Vettukattil, H. Banitalebi, N. R. Krogh, and J. T. Geitung, "Explainable artificial intelligence for human-machine interaction in brain tumor localization," *Journal of Personalized Medicine*, vol. 11, no. 11, p. 1213, 2021.

[7] M. E. Plissiti, P. Dimitrakopoulos, G. Sfikas, C. Nikou, O. Krikoni, and A. Charchanti, "Sipakmed: A new dataset for feature and image based classification of normal and pathological cervical cells in pap smear images," in *2018 25th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2018, pp. 3144–3148.

[8] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *Advances in neural information processing systems*, vol. 33, pp. 18 661–18 673, 2020.

[9] A. Ghoneim, G. Muhammad, and M. S. Hossain, "Cervical cancer classification using convolutional neural networks and extreme learning machines," *Future Generation Computer Systems*, vol. 102, pp. 643–649, 2020.

[10] M. M. Rahaman, C. Li, Y. Yao, F. Kulwa, X. Wu, X. Li, and Q. Wang, "Deepcervix: A deep learning-based framework for the classification of cervical cells using hybrid deep feature fusion techniques," *Computers in Biology and Medicine*, vol. 136, p. 104649, 2021.

[11] A. Manna, R. Kundu, D. Kaplun, A. Sinitca, and R. Sarkar, "A fuzzy rank-based ensemble of cnn models for classification of cervical cytology," *Scientific Reports*, vol. 11, no. 1, p. 14538, 2021.

[12] T.-C. Hsieh, C.-W. Liao, Y.-C. Lai, K.-M. Law, P.-K. Chan, and C.-H. Kao, "Detection of bone metastases on bone scans through image classification with contrastive learning," *Journal of Personalized Medicine*, vol. 11, no. 12, p. 1248, 2021.

[13] V. Ravi, "Attention cost-sensitive deep learning-based approach for skin cancer detection and classification," *Cancers*, vol. 14, no. 23, p. 5872, 2022.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[15] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[16] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[18] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, and R. Togneri, "Cost-sensitive learning of deep feature representations from imbalanced data," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 8, pp. 3573–3587, 2017.

[19] V. Ravi, H. Narasimhan, and T. D. Pham, "A cost-sensitive deep learning-based meta-classifier for pediatric pneumonia classification using chest x-rays," *Expert Systems*, vol. 39, no. 7, p. e12966, 2022.

[20] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[21] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," in *2018 IEEE winter conference on applications of computer vision (WACV)*. IEEE, 2018, pp. 839–847.

[22] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu, "Score-cam: Score-weighted visual explanations for convolutional neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2020, pp. 24–25.

[23] P.-T. Jiang, C.-B. Zhang, Q. Hou, M.-M. Cheng, and Y. Wei, "Layercam: Exploring hierarchical class activation maps for localization," *IEEE Transactions on Image Processing*, vol. 30, pp. 5875–5888, 2021.

[24] M. T. Ribeiro, S. Singh, and C. Guestrin, "" why should i trust you?" explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 1135–1144.