


Research Article

An Improved Image Spam Classification Model Based on Deep Learning Techniques

A. Buboo Singh , Kh Manglem Singh, Y. Jina Chanu, Khelchandra Thongam, and Kh Johnson Singh

National Institute of Technology, Imphal, Manipur, India

Correspondence should be addressed to A. Buboo Singh; angom102@gmail.com

Received 14 May 2022; Accepted 10 July 2022; Published 2 August 2022

Academic Editor: Vincenzo Conti

Copyright © 2022 A. Buboo Singh et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Image Spam is a type of spam that has embedded text in an image. Classification of Image Spam is done using various machine learning approaches based on a broad set of features extracted from the image. For its remarkable results, the convolutional neural networks (CNN) are widely used in image classification as well as feature extraction tasks. In this research, we analyze image spam using a CNN model based on deep learning techniques. The proposed model is fine-tuned and optimized for both feature extraction as well as for classification tasks. We also compared our proposed model to different “Improved” and “Challenge” image spam datasets, which were developed for increasing the difficulty of the classification task. Our model significantly improves the accuracy of the classification task as compared to other approaches on the same datasets.

1. Introduction

Spam can be defined in a simple term as unsolicited bulk e-mail (UBE) in short and is not only annoying but may also contain links to phishing websites or malware attached as executable files. The number of spam is increasing and according to Shcherbakova et al. [1], during 2019, spam accounted for more than half of all inbound e-mails. One of the techniques commonly used by a spammer to evade text-based spam filters is to embed messages inside an image. To further prevent easy extraction of the embedded text from the image using OCR techniques, the messages embedded are subjected to various forms of alteration [2] such as multiframe animated GIF, by adding noise to the image, using a hand written style of image, by using patchy fonts and randomization.

The most common approaches in image spam filtering consist of firstly extracting the image features such as those that are based on file properties, metadata, low-level or global image features, or those related to image textures. Secondly, the extracted features are then used as input to machine learning models to classify the images as either

spam or nonspam. Among the machine learning techniques, some require manually selected input image features and the accuracy and complexity of the approaches depend on the number and types of features used.

Alternatively, other approaches based on deep learning techniques use raw images as input as they have the capabilities for automatic feature extraction from the raw images. Among the deep learning techniques, convolutional neural network stands out when used in the area of image classification, leading to numerous improvements to deep network training [3].

Training a deep learning model from scratch requires many data because it contains millions of trainable parameters, and a small dataset would be insufficient to get a good generalization of the model. Therefore, we propose the use of a pretrained model of CNN that uses the transfer learning (TL) technique and use it as a feature extractor from image spam. The extracted features are then fed into our fine-tuned and optimized custom ReDense layer, which finally classifies the input image as either spam or nonspam.

In this paper, we analyzed our proposed model on “improved” [4] as well as the “challenge” [5] datasets, which

are specially hand-crafted by the respective authors to make the classification task difficult by making the spam images look similar to that of nonspam images. We also compared our proposed model with other approaches on various other image spam datasets. Our proposed model outperformed the other approaches significantly in terms of accuracy and compute complexity.

The remainder of the paper is organized as follows. In Section 2, we give a brief review of image spam classification and related works, along with a brief overview of convolutional neural networks and transfer learning. In Section 3, we discuss the materials and methods used in our research work, including a detailed explanation of the various datasets used. In this section, we also present the base CNN model and also highlighted the calculation of performance measures. Section 4 gives our proposed CNN model. In Section 5, we present our detection results, while Section 6 gives our conclusions and suggestions for future work.

2. Related Works

2.1. Image Spam Classifications. Image spam detectors can be broadly categorized into two types. The first type is based on extracting the textual content embedded in the image using some form of optical character recognition techniques and then uses text-based filters to classify the input as either spam or nonspam. Many works [6–9] are based on using such an approach to recover text from the spam images and also different types of text-filtering techniques.

The second type of image spam classification approach uses various image features and uses various machine learning techniques in the classification process. Some of the works use image features that are based on file properties and metadata [10], global image features including color and gradient histograms [11–17], low-level image features [18–22], image texture-based features related to a histogram, gradient, run-length matrix, co-occurrence matrix, autoregressive model, and wavelet transform [23–25]. Other works use image features such as Speeded Up Robust Feature (SURF) [26] and n-gram after converting the image to a string of its Base64 format [27].

In Ref. [28], the author uses multiple features fusion techniques using HOG, gradient, and color features from the images which were analyzed and filtering was carried out using a KNN classifier.

The work presented in Ref. [29] uses a fusion model to filter spam by processing the image and text part separately using a CNN and an LSTM, respectively, and finally combining the resulting classification probabilities to identify whether the e-mail is spam or not.

Recent work presented in Ref. [30] uses deep convolution neural network (DCNN) and transfer learning based CNN models and claims to achieve very high accuracy of 99% in some of the proposed models with zero false-positive rates in the best case. However, the model could achieve an accuracy of 97.3% on the “improved” [4] dataset.

The main purpose of this research is to improve the accuracy of the classification of the “Improved” and “Challenge” datasets created by Refs. [4, 5], respectively. The

datasets are developed to benchmark the accuracy of the various machine learning approaches adopted in the area of image spam classification and are hand-crafted to make the spam images look similar to that of nonspam images.

In Refs. [4, 5], the authors use a broad set of image features consisting of 21 and 38 features, respectively, and conducted various experiments primarily involving feature selection and feature reduction. The number of features is then reduced to an optimal number by using recursive features elimination techniques which reduces the features with the smallest weights. Further, they develop a new spam image dataset that cannot be detected using their PCA or SVM approach. The author reasserts that this new dataset should prove valuable for improving image spam detection capabilities. The same datasets are being used in our experiments.

2.2. Convolutional Neural Networks. The ImageNet Very Large Scale Visual Recognition Challenge (ILSVRC) [31], is one of the reasons for the recent improvement in the area of computer vision tasks. A large number of models based on convolutional neural network [32, 33] are being released which are pretrained in ILSVRC and which can be reused as a baseline model. Example of such models are VGG-16/19 network [34], Inception-v3 [35], residual Network (ResNet) [36], depthwise separable convolution networks (Xception) [37], and densely connected networks (DensNet) [38]. With the availability of a framework that allows us to develop our models for any specific tasks [39], recent state-of-the-art CNN models such as Big Transfer (BiT) are gaining popularity in various image analysis works [40].

2.3. Transfer Learning. Training a deep CNN model such as VGG16, VGG19, ResNet, Xception, or BiT from scratch requires a lot of data because they contain millions of trainable parameters [41], in which a small dataset would be insufficient to get a good generalization of the model. On the contrary, the mentioned baseline models can be reused using their pretrained weights employing a transfer learning technique.

Transfer learning has been a useful machine learning method in which a pretrained model of CNN is reused to take advantage of its weights to take them into account as initialization for a new CNN model for a different purpose [42]. There exist two primary ways to use transfer learning from a model:

- (i) Reuse a model as a feature extractor and use a new different classifier.
- (ii) Reuse the model to perform fine-tuning (FT). FT is a technique that uses some unfrozen layers of a full model to slightly adjust both the new fully connected (FC) layers of the classifier and specific layers of the CNN-like convolution layers [43].

In our experiments, we used a CNN model and TL for the extraction of features from the input images, and the features vector thus obtained is then fed into our binary classifier for classification of the given input image as spam and nonspam.

3. Materials and Methods

In this section, we will introduce the datasets used for this research. Details of the mechanism for the generation of the “improved” and “Challenge” datasets will also be discussed along with the base convolution neural network from which we developed our proposed model. In addition, some performance measures will be explained, as shown in Figure 1

3.1. Datasets Used in the Experiments

- (1) Dataset 1 (Dredze dataset [10]): this dataset was created by the authors of Ref. [10] and contains 2550 personal nonspam images, 3239 personal spam images, and 9503 SpamArchive spam images, out of which we have used only the cleaned personal images and retained only 1089 spam and 1029 nonspam images. The dataset is available at https://www.cs.jhu.edu/~mdredze/datasets/image_spam/.
- (2) Dataset 2 (Image Spam Hunter (ISH) [11]): this dataset was created by the authors of Ref. [11] and contains 928 spam images along with 810 nonspam images, in JPEG format, out of which we retained 920 spam and 810 nonspam images. The dataset is available at https://users.cs.northwestern.edu/~yga751/ML/ISH.htm#dataset_.
- (3) Dataset 3 (improved dataset [4]): this dataset was developed by the authors of Ref. [4] and contains 1029 generated “improved” images, from the perspective of the spammer, since these images are likely to be much more difficult to detect along with 810 nonspam images. To make the dataset more difficult to detect, they added background layers, modified the color elements, introduced noise, and also modified the metadata. Figure 1 gives two randomly selected examples from their improved dataset.
- (4) Dataset 4 (challenge datasets A and B [5]): this dataset is created by the authors of Ref. [5] by extracting the content of an existing spam image and then overlaying it on a nonspam image. It consists of 810 spam and 810 nonspam images. The author applied various image processing techniques to actual spam images to make the images look more like a nonspam image. They used the Dredze dataset for their spam corpus and overlaid nonspam images from the ISH dataset.

The challenge spam image generation approach is shown in Figure 2, where a text from a spam mail, as shown in Figure 2(b), is overlaid on a nonspam image, as shown in Figure 2(a), to generate a challenge spam image, as shown in Figure 2(c).

Figure 3 shows the scatterplots of the compression ratio and color entropy for nonspam (ham) and the challenge dataset images, which clearly show that the nonspam (ham) and challenge dataset images are more closely aligned, as compared to those of nonspam (ham) and existing spam images.

3.2. CNN Model. Our proposed model uses a TL method in conjunction with the base CNN model BiT-M R50×1 network shown in Figure 4. The model is a state-of-the-art model which is pretrained on ImageNet-21 K, a dataset with 14 million images labeled with 21,843 classes. The input to the model is a 224×224 color image and its output is the 2048-dimensional features vector, before a multilabel classification head. The hidden layers are a combination of convolution blocks, as shown in Figure 5 and identity blocks, as shown in Figure 6, of various dimensions with a couple of pooling applied for dimensionality reduction.

Big Transfer (BiT) is not a new model but a recipe for pretraining image classification models on large supervised datasets. They are based on ResNet 50 model and are efficiently fine-tuned on a given target task. The recipe achieves excellent performance gain on a wide variety of tasks, even when using very few labeled examples from the target dataset. Contrary to the original ResNet architecture, the performance improvement is due to the use of group normalization instead of batch normalization and weight standardization of the convolution kernels.

For pretraining on large scale and stabilizing the training by normalizing the activation, group normalization (GN) is used in place of batch normalization (BN). Some of the benefits are as follows: first, BN’s state (mean and variance of neural activations) needs adjustment between pretraining and transfer, whereas GN is stateless, thus side-stepping this difficulty. Second, BN uses batch-level statistics, which become unreliable with small per-device batch sizes that are inevitable for large models. Since GN does not compute batch-level statistics, it also side-steps this issue.

3.3. Performance Measure. In order to assess the effectiveness of the proposed method, different evaluation indicators have been used, such as Accuracy, Recall, Precision, and F1-score, which are defined as

$$\begin{aligned}
 \text{Accuracy} &= \frac{TP + TN}{TP + TN + FP + FN}, \\
 \text{Recall} &= \frac{TP}{TP + FN}, \\
 \text{Precision} &= \frac{TP}{TP + FP}, \\
 \text{F1 - score} &= \frac{2 \times (\text{Precision} + \text{Recall})}{\text{Precision} + \text{Recall}},
 \end{aligned} \tag{1}$$

where false positive (FP) is the no. of legitimate e-mails that are misclassified; false negative (FN) is the no. of misclassified spam; true positive (TP) is the no. of spam that is correctly classified; and true negative (TN) is the no. of legitimate e-mails that are correctly classified.

For spam detection, the evaluation metrics about the accuracy, recall, precision, and F1-score are mainly based on the confusion matrix (as shown in Table 1).

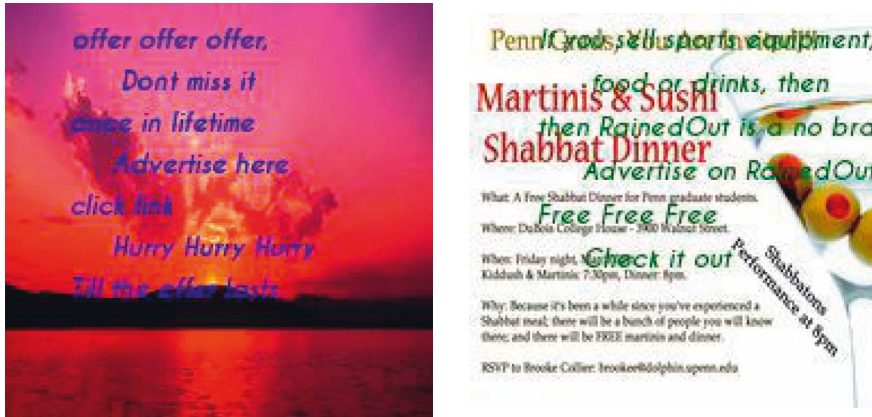


FIGURE 1: Examples of improved spam images (image source: [4]).

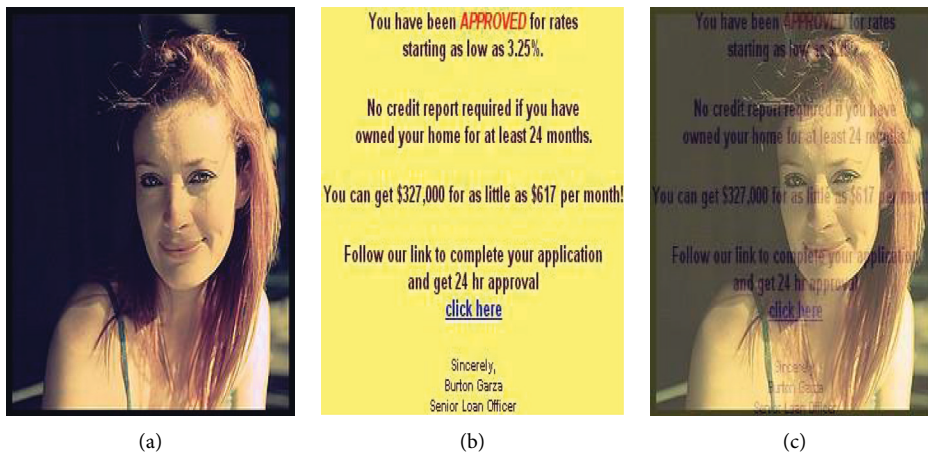


FIGURE 2: Challenge dataset (image source: [5]): (a) nonspam image, (b) spam text, and (c) challenge spam image.

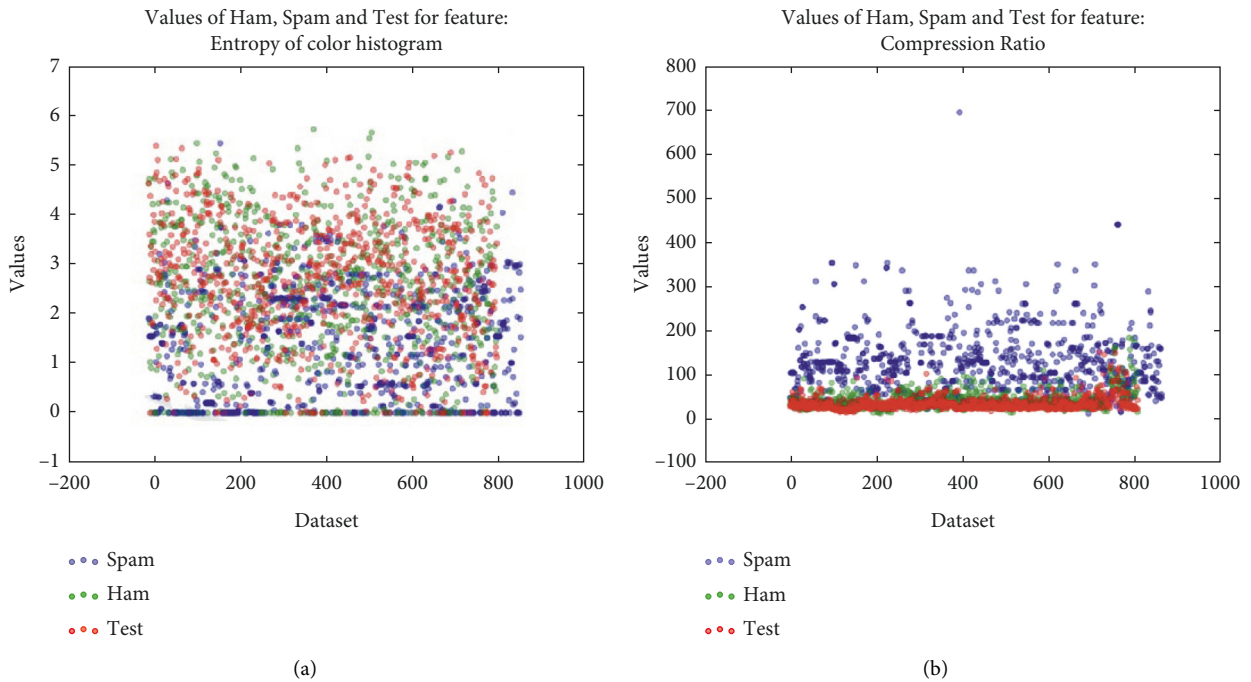


FIGURE 3: Feature value comparison scatterplot (Image source: [5]). (a) Entropy of color histogram. (b) Compression ratio.

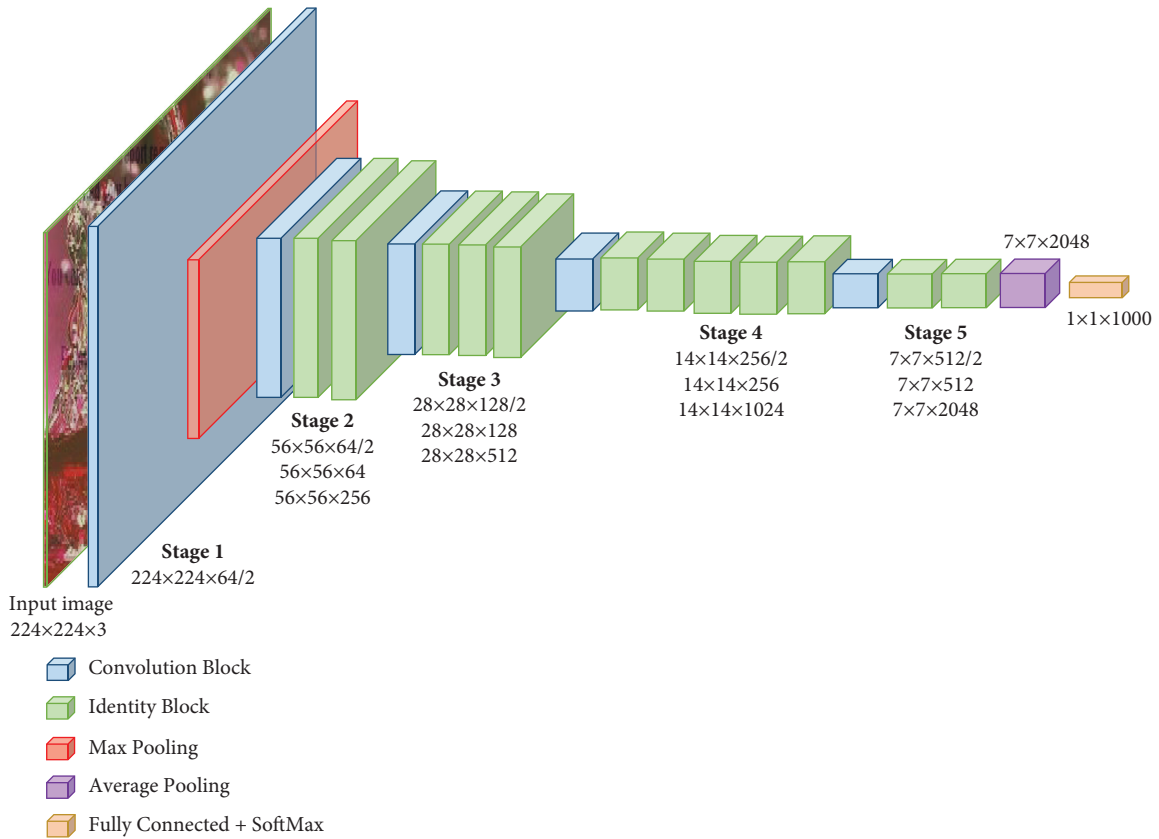


FIGURE 4: BiT-M R50 \times 1 architecture based on ResNet 50 (image source: author).

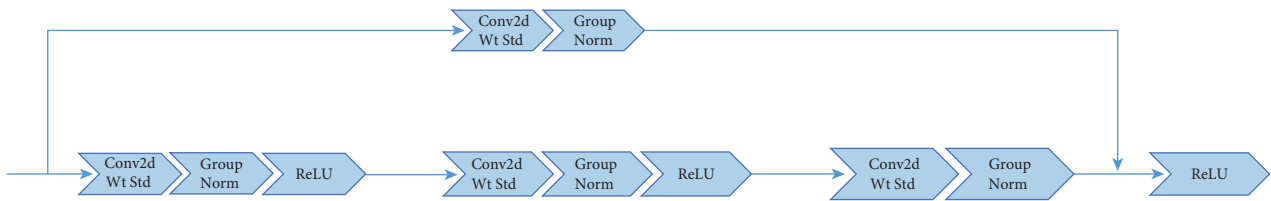


FIGURE 5: Convolution block with GN and kernel eight standardization (image source: author).

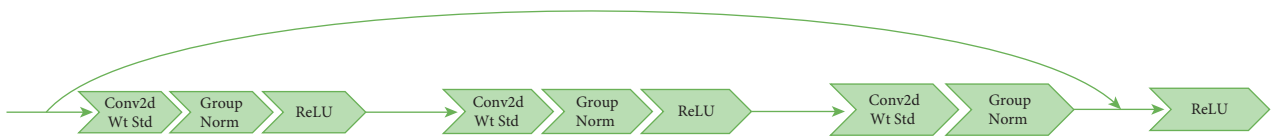


FIGURE 6: Identity block (image source: author).

4. Proposed Model

The proposed model uses two main components:

- (1) A feature extractor for extraction of image features from the input images
- (2) A binary classifier for classification of the input image as either spam or nonspam

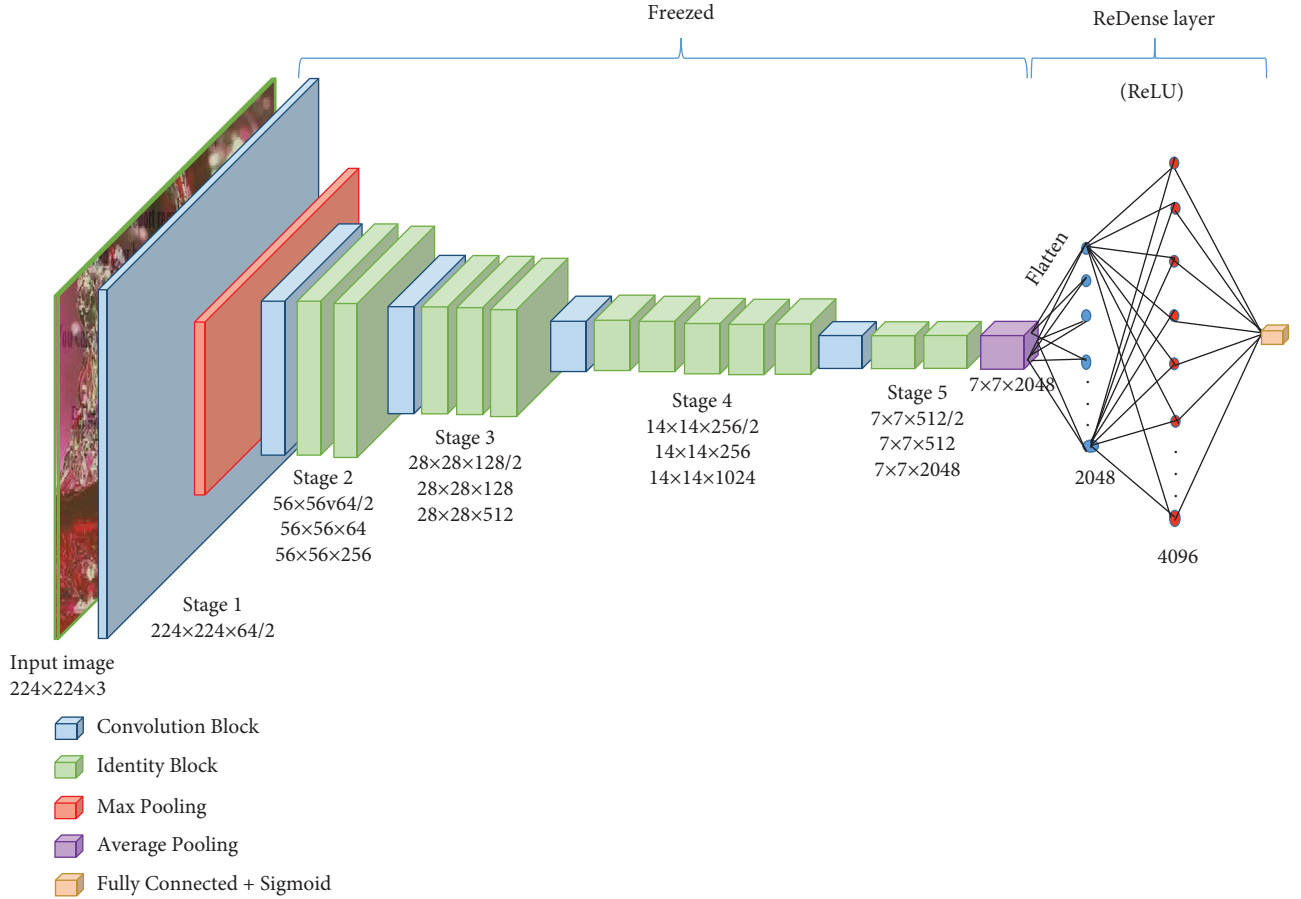
The feature extractor is based on the BiT-M R50 \times 1 CNN model, as shown in Figure 7, where the main convolution, identity, and pooling blocks from stage 1 to stage 5 are frozen; therefore, they are no longer used in the

training again. Preprocessing is performed on the dataset such that all the images are resized to 224×224 dimensions. Moreover, we also normalize the image data such that the value is between 0 and 1. This helps to make sure that the data has a similar distribution and hence helps the model converge faster. It also helps improve the stability of the model.

The stages are used to transform the input dimension of $224 \times 224 \times 3$ to $7 \times 7 \times 2048$ using various combinations of convolution layers and pooling layers at different stages with different strides. The output after the last stage is flattened to get a 2048-dimensional feature vector.

TABLE 1: Confusion matrix.

Prediction	Actual	
	Spam	Nonspam
Spam	TP	FN
Nonspam	FP	TN

FIGURE 7: Proposed CNN model with the added $1 \times 1 \times 4096$ ReDense layer and final dense layer with sigmoid activation (image source: author).

The final layer is replaced by two new layers, namely, a $1 \times 1 \times 4096$ ReDense layer and an output Dense layer with a sigmoid activation function, which is used for the binary classification purpose. The ReDense layer [44] is a $2 \times m$ dense layer with a ReLU activation function, where m is the number of dimensions in the flattened output. The addition of the ReDense layer helps to improve the accuracy of the classification task. The only training required is the two dense layers added at the end; therefore, the computational requirement is hugely reduced compared to training the whole 50 layers, if TL was not used.

The feature vector, generated in the previous block, is then fed into a ReDense layer consisting of a Dense layer with 4096 neurons and ReLU activation function followed by a single neuron output layer with a sigmoid activation function. Only the ReDense layer is trained using the feature vector during the training phase.

We experimented with a different set of network hyperparameters and found that the values given in Table 2 result in the highest accuracy.

5. Experiment and Results

In this section, the conducted experiments will be explained and the implementation details will be mentioned. We include the explanation for the experimental framework used, as well as the validation and test results obtained.

5.1. Experimental Framework. The image preprocessing techniques were implemented in Python 3.6 using OpenCV [45] as the main image processing library. All experiments were conducted on an Intel Xeon Quad-Core processor Workstation running Windows 10 Pro

TABLE 2: Network hyperparameters.

Batch size	Learning rate	No. of epochs	Optimizer	Loss function
64	1×10^{-3}	25	SGD with momentum	Sparse categorical loss

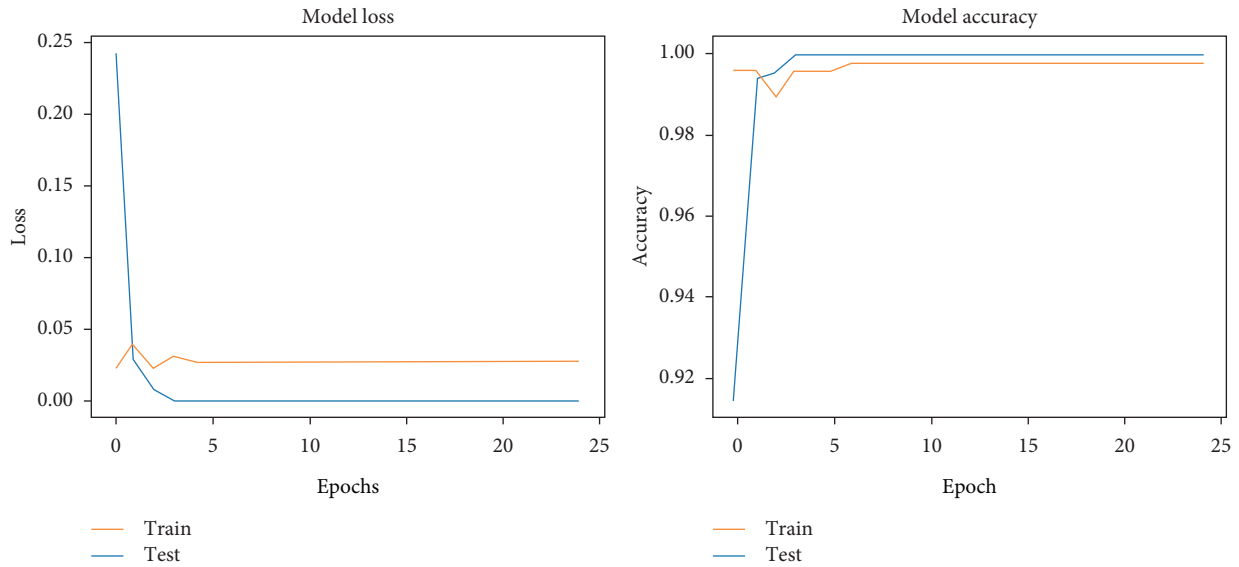


FIGURE 8: Validation loss and ROC curve of the proposed CNN model on the improved dataset [4].

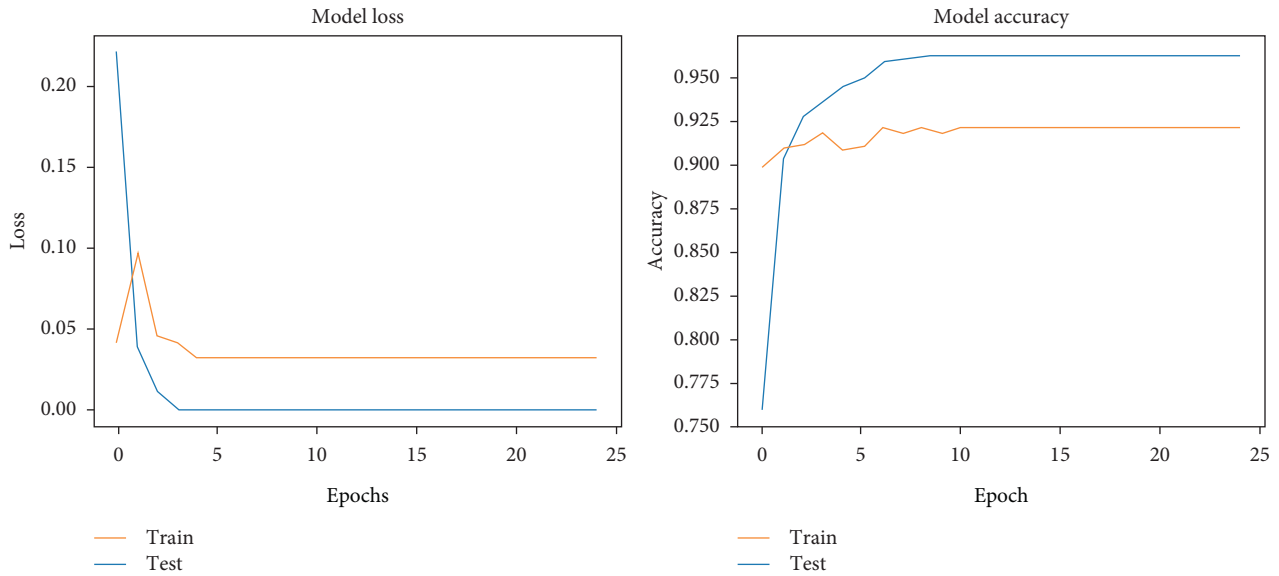


FIGURE 9: Validation loss and ROC curve of the proposed CNN model on the challenge-A dataset [5].

64-bit, with 32 GB of RAM along with an Nvidia P1000 GPU with 4 GB VRAM. The deep learning framework Keras [46] was used in the implementation of the transfer learning model.

5.2. Results. We performed some experiments by training our proposed CNN model on the training sets of the five different datasets, namely, “Improved” [4], “Challenge-A” [5], “Challenge-B” [5], “Dredze” [9], and “ISH” [10], and

then we validate the model by employing the validation sets. Figures 8–12 show the validation loss along with the ROC curve of our proposed CNN model on different datasets.

Our proposed CNN model achieved a near-perfect accuracy of 99% on the improved dataset while getting an excellent result on the two challenge datasets A and B, with an accuracy of 93% and 98%, respectively. The accuracy achieved by our proposed model far exceeds the accuracy obtained by the respective authors using the SVM classifier as shown in Table 3.

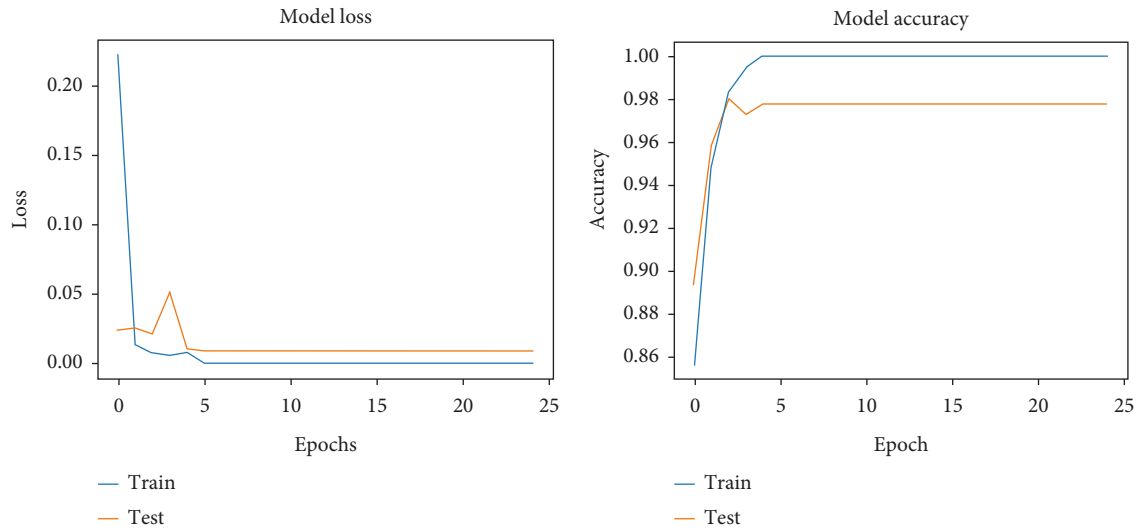


FIGURE 10: Validation loss and ROC curve of the proposed CNN model on the challenge-B dataset [5].

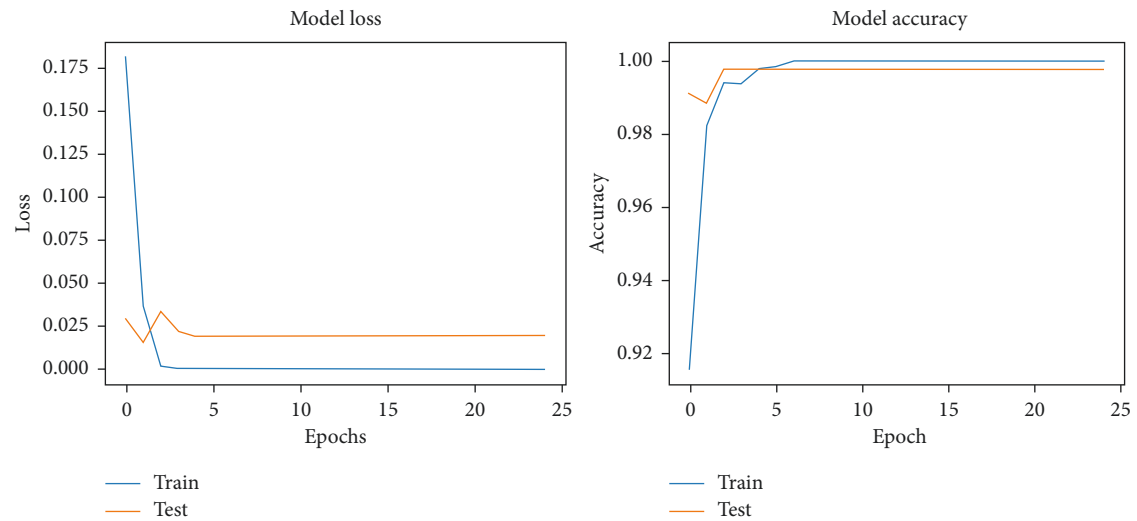


FIGURE 11: Validation loss and ROC curve of the proposed CNN model on the Dredze dataset [9].

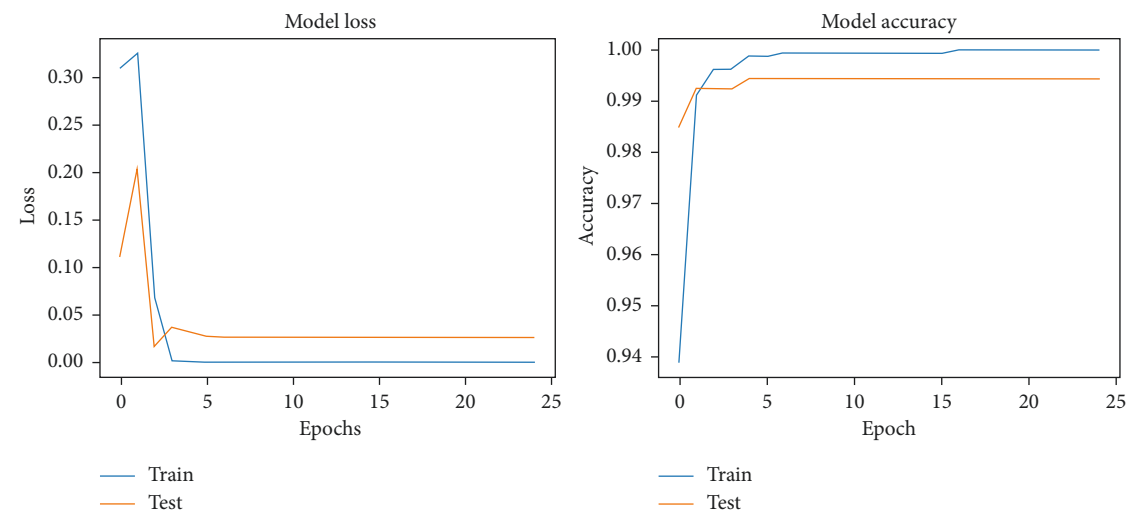


FIGURE 12: Validation loss and ROC curve of the proposed CNN model on the ISH dataset [10].

TABLE 3: Accuracy comparison for improved dataset and challenge datasets.

	Dataset 3 Improved dataset	Dataset 4A Challenge dataset	Dataset 4B Challenge dataset
Our proposed CNN model	99.78	93.75	97.83
Annadatha and Stamp (SVM) [4]	70.00	—	—
Chavda et al. (SVM) [5]	—	69.32	69.32
Sriram et al. (DCNN) [30]	97.30	—	—

TABLE 4: Accuracy comparison on Dredze et al. [10] and ISH datasets [11].

	Dataset 1 [10]	Dataset 2 [11]
Our proposed CNN model	99.44	99.77
Sriram et al. (DCNN) [30]	97.30	99.80
Annadatha and Stamp (SVM) [4]	—	97.00
Chavda et al. (SVM) [5]	98.00	97.00
Qian et al. (SVM with Gaussian kernel) [26]	97.90	98.30
Gao et al. (ISH) [11]	—	94.94
Yuan and Zhang (multifeatures fusion method) [28]	95.00	—
Das et al. (multiple classifier) [9]	98.00	—
Kumar and Biswas (image texture features) [25]	98.50	98.56
Dredze and Gevaryahu [10]	98.00	—
Shen et al. (comprehensive visual modeling) [47]	96.80	—
Wang et al. (low-level image feature) [19]	97.00	—
Al-Duwairi et al. (Base64 encoding) [23]	99.00	—
Al-Duwairi et al. (texture analysis) [24]	98.55	98.10
Liu et al. (multilayer spam filter) [22]	94.30	94.30
Gupta et al. (low-level and metadata features) [20]	93.30	—
Xu et al. (K-labels propagation model) [27]	90.00	—
Soranamageswari and Meena (ANN) [16]	92.82	—
Kumaresan et al. (SVM and PSO) [21]	90.00	—
Yang et al. (multimodal fusion) [29]	92.64	—

We also experimented on other commonly available public spam image datasets, namely, the popular Dredze [10] and Image Spam Hunter [11] datasets. The results of our experiments are then compared with other approaches based on a variety of machine learning methods and features, which are ranging from low level to metadata and OCR. Here, also our proposed CNN model obtained an excellent accuracy result and showed improvement in the already near-perfect results obtained by other authors using various ML techniques, as shown in Table 4.

6. Conclusions

Image Spam classification is a type of machine learning problem where features from the images are extracted and trained using machine learning models. The support vector machine technique offers a model with excellent results. However, when carefully hand-crafted datasets of image spam were given to such a model based on the SVM [4, 5], the results were not up to the mark as compared with normal image spam datasets. We showed that such improved image spam that cannot be reliably detected using

the image processing-based features earlier could be reliably detected using our proposed CNN model based on deep learning techniques, with significant improvement in detection accuracy.

We showed that by optimizing the ReDense layer with hypertuning of various network parameters, the classification accuracy of the CNN model could be improved. Moreover, our experiment once again reiterates that deep learning techniques, using TL, can extract features from raw input images, even though these images were not part of the training data, and perform classification with significantly higher accuracy. We showed that using a pretrained model of CNN that uses the TL technique proves highly cost-effective in terms of computing requirements and at the same time gives high accuracy in the classification task, even in a small dataset. The achieved accuracy indicates that the proposed approach is not only viable and robust but also has the potential to be applied to other areas of image classification.

In future work, we would want to try using an automatic parameter-tuning method and apply the proposed algorithm to several other image datasets, to make a statistical analysis of its performance.

Data Availability

Dredze Dataset 1 is available at https://www.cs.jhu.edu/~mdredze/datasets/image_spam/, https://www.cs.jhu.edu/~mdredze/datasets/image_spam/personal_image_spam.tar.gz, and https://www.cs.jhu.edu/~mdredze/datasets/image_spam/personal_image_ham.tar.gz. Image Spam Hunter Dataset 2 is available at <https://users.cs.northwestern.edu/~yga751/ML/ISH.htm#datas> et. Datasets 2 and 3 are available from the respective authors on request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors gratefully acknowledge the Ministry of Electronics & Information Technology, Government of India for their economic support to develop this work, under the AA No. 12(3)/2016-ESD Dated 22nd March 2016.

References

- [1] T. Shcherbakova, T. Sidorina, and T. Kulikova, "Spam and phishing in Q1 2020," 2020, <https://securelist.com/spam-and-phishing-in-q1-2020/97091/>.
- [2] A. Attar, R. M. Rad, and R. E. Atani, "A survey of image spamming and filtering techniques," *Artificial Intelligence Review*, vol. 40, no. 1, pp. 71–105, 2013.
- [3] X. Wang, Y. Zhao, and F. Pourpanah, "Recent advances in deep learning," *International Journal of Machine Learning and Cybernetics*, vol. 11, no. 4, pp. 747–750, 2020.
- [4] A. Annadatha and M. Stamp, "Image spam analysis and detection," *Journal of Computer Virology and Hacking Techniques*, vol. 14, no. 1, pp. 39–52, 2018.
- [5] A. Chavda, K. Potika, F. D. Troia, and M. Stamp, "Support vector machines for image spam analysis," in *Proceedings of the 15th International Joint Conference on e-Business and Telecommunications - Volume 1: BASS*, pp. 431–441, Porto, Portugal, July 2018.
- [6] P. Wan and M. Uehara, "Spam detection using Sobel operators and OCR," in *Proceedings of the 2012 26th International Conference on Advanced Information Networking and Applications Workshops*, pp. 1017–1022, 2012 Mar.
- [7] P. Wan and M. Uehara, "Multiple filters of spam using sobel operators and OCR," in *Proceedings of the 2012 Sixth International Conference on Complex, Intelligent, and Software Intensive Systems*, pp. 164–169, IEEE, Palermo, Italy, 2012 July.
- [8] D. Yamakawa and N. Yoshiura, "Applying tesseract-OCR to detection of image spam mails," in *Proceedings of the 2012 14th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, vol. 25, pp. 1–4, IEEE, Seoul, Korea (South), September 2012.
- [9] M. Das, A. Bhomick, Y. J. Singh, and V. Prasad, "A modular approach towards image spam filtering using multiple classifiers," in *Proceedings of the 2014 IEEE International Conference on Computational Intelligence and Computing Research*, pp. 1–5, IEEE, Coimbatore, India, December 2014.
- [10] M. Dredze, R. Gevayahu, and A. Elias-Bachrach, "Learning fast classifiers for image spam," in *Proceedings of the CEAS 2007 The Fourth Conference on Email and Anti-Spam*, pp. 2007–2487, Mountain View, CA, USA, August 2007.
- [11] Y. Gao, M. Yang, X. Zhao et al., "Image spam hunter," in *Proceedings of the 2008 IEEE international conference on acoustics, speech and signal processing*, pp. 1765–1768, IEEE, Las Vegas, NV, USA, April 2008.
- [12] Y. Gao and A. Choudhary, "Active learning image spam hunter," in *International Symposium on Visual Computing*, pp. 293–302, Springer, Berlin, Heidelberg, 2009.
- [13] Y. Gao, M. Yang, and A. Choudhary, "Semi supervised image spam hunter: a regularized discriminant em approach," in *Proceedings of the International Conference on Advanced Data Mining and Applications*, pp. 152–164, Beijing, China, August 2009.
- [14] P. He, X. Wen, and W. Zheng, "A simple method for filtering image spam," in *Proceedings of the 2009 Eighth IEEE/ACIS International Conference on Computer and Information Science*, pp. 910–913, IEEE, Shanghai, China, June 2009.
- [15] Z. Wang, W. K. Josephson, Q. Lv, M. Charikar, and K. Li, "Filtering image spam with near-duplicate detection," in *Proceedings of the CEAS The Fourth Conference on Email and Anti-Spam*, Mountain View, CA, USA, August 2007.
- [16] M. Soranamageswari and C. Meena, "Statistical feature extraction for classification of image spam using artificial neural networks," in *Proceedings of the 2010 Second International Conference On Machine Learning and Computing*, pp. 101–105, IEEE, Bangalore, India, 2010 February.
- [17] Y. Gao, A. Choudhary, and G. Hua, "A comprehensive approach to image spam detection: from server to client solution," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 4, pp. 826–836, 2010.
- [18] Z. M. Win and N. Aye, "Detecting image spam based on file properties, histogram and hough transform," *Journal of Advances in Computer Networks*, vol. 2, no. 4, pp. 287–292, 2014.
- [19] C. Wang, F. Zhang, F. Li, and Q. Liu, "Image spam classification based on low-level image features," in *Proceedings of the 2010 International Conference on Communications, Circuits and Systems (ICCCAS)*, pp. 290–293, IEEE, Chengdu, China, July 2010.
- [20] A. Gupta, C. Singhal, and S. Aggarwal, "Identification of image spam by using low level & metadata features," *International Journal of Network Security & Its Applications*, vol. 4, no. 2, pp. 163–178, 2012.
- [21] T. Kumaresan, S. Sanjushree, K. Suhasini, and C. Palanisamy, "Image spam filtering using support vector machine and particle swarm optimization," *International Journal of Computer Application*, vol. 1, pp. 17–21, 2015.
- [22] T. J. Liu, W. L. Tsao, and C. L. Lee, "A high performance image-spam filtering system," in *Proceedings of the 2010 Ninth International Symposium on Distributed Computing and Applications to Business Engineering and Science*, pp. 445–449, IEEE, Hong Kong, China, August 2010.
- [23] B. Al-Duwairi, I. Khater, and O. Al-Jarrah, "Texture analysis-based image spam filtering," in *Proceedings of the 2011 International Conference for Internet Technology and Secured Transactions*, pp. 288–293, IEEE, Abu Dhabi, UAE, December 2011.
- [24] B. Al-Duwairi, I. Khater, and O. Al-Jarrah, "Detecting image spam using image texture features," *International Journal for Information Security Research (IJISR)*, vol. 2, no. 3/4, pp. 344–353, 2012.
- [25] P. Kumar and M. Biswas, "SVM with Gaussian kernel-based image spam detection on textual features," in *Proceedings of*

- the 2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT)*, pp. 1–6, IEEE, Ghaziabad, India, February 2017.
- [26] X. Qian, W. Zhang, Y. Zhang, G. Zhou, and Z. Wang, “Detecting image spam based on k-labels propagation model,” in *Proceedings of the 2013 10th Web Information System and Application Conference*, vol. 10, pp. 170–175, IEEE, Yangzhou, China, November 2013.
- [27] C. Xu, Y. Chen, and K. Chiew, “An approach to image spam filtering based on Base64 encoding and N-gram feature extraction,” in *Proceedings of the 2010 22nd IEEE International Conference on Tools with Artificial Intelligence*, pp. 171–177, IEEE, Arras, France, October 2010.
- [28] S. Yuan and C. Zhang, “An improved multiple features fusion method for image spam filtering,” in *Proceedings of the 2016 3rd International Conference on Information Science and Control Engineering (ICISCE)*, pp. 200–203, IEEE, Beijing, July 2016.
- [29] H. Yang, Q. Liu, S. Zhou, and Y. Luo, “A spam filtering method based on multi-modal fusion,” *Applied Sciences*, vol. 9, no. 6, p. 1152, 2019.
- [30] S. Sriram, R. Vinayakumar, V. Sowmya et al., “Deep convolutional neural networks for image spam classification,” 2020, <https://hal.archives-ouvertes.fr/hal-02510594/>.
- [31] O. Russakovsky, J. Deng, H. Su et al., “ImageNet large scale visual recognition challenge,” *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.
- [32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in *Proceedings of the Advances in Neural Information Processing Systems (NIPS’12)*, Lake Tahoe, NV, USA, December 2012.
- [33] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [34] L. Geng, S. Zhang, J. Tong, and Z. Xiao, “Lung segmentation method with dilated convolution based on VGG-16 network,” *Computer Assisted Surgery*, vol. 24, no. sup2, pp. 27–33, 2019.
- [35] M. Jung and S. Chi, “Human activity classification based on sound recognition and residual convolutional neural network,” *Automation in Construction*, vol. 114, Article ID 103177, 2020.
- [36] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition CVPR*, pp. 770–778, IEEE, Las Vegas, NV, USA, July 2016.
- [37] F. Chollet, “Xception.: Deep learning with depthwise separable convolutions,” in *Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition CVPR 2017*, pp. 1800–1807, IEEE, Honolulu, HI, USA, July 2017.
- [38] Z. Yao, J. Li, Z. Guan, Y. Ye, and Y. Chen, “Liver disease screening based on densely connected deep neural networks,” *Neural Networks*, vol. 123, pp. 299–304, 2020.
- [39] A. Kolesnikov, L. Beyer, X. Zhai et al., “Big Transfer (bit): general visual representation learning,” in *Computer Vision - ECCV 2020*, pp. 491–507, Springer, Cham, 2020.
- [40] F. Chollet, “Keras: the Python deep learning library,” 2020, <https://keras.io/>.
- [41] J. Luján-García, C. Yáñez-Márquez, Y. Villuendas-Rey, and O. Camacho-Nieto, “A transfer learning method for pneumonia classification and visualization,” *Applied Sciences*, vol. 10, no. 8, p. 2908, 2020.
- [42] M. Tsiakmaki, G. Kostopoulos, S. Kotsiantis, and O. Ragos, “Transfer learning from deep neural networks for predicting student performance,” *Applied Sciences*, vol. 10, no. 6, p. 2145, 2020.
- [43] F. Chollet, *Deep Learning with Python*, pp. 287–295, Manning Publications Co, Shelter Island, NY, USA, 1st Ed edition, 2018.
- [44] A. M. Javid, S. Das, M. Skoglund, and S. Chatterjee, “A relu dense layer to improve the performance of neural networks,” in *Proceedings of the ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2810–2814, IEEE, June 2021.
- [45] OpenCV, “Open CV,” 2020, <https://opencv.org/>.
- [46] F. Chollet, “Keras: the Python deep learning library,” 2020, <https://keras.io/>.
- [47] J. Shen, R. H. Deng, Z. Cheng, L. Nie, and S. Yan, “On robust image spam filtering via comprehensive visual modeling,” *Pattern Recognition*, vol. 48, no. 10, pp. 3227–3238, 2015.