ICIC Express Letters
Volume **16**, Number **9**, September **2022**

ICIC International ©2022 ISSN 1881-803X
pp. **943–950**

AN ENSEMBLE OF CNN-ELM MODELS FOR TRASH CLASSIFICATION

John Paul Tan Yusiong

Division of Natural Sciences and Mathematics
University of the Philippines Visayas Tacloban College
Magsaysay Boulevard, Tacloban City 6500, Leyte, Philippines
jtyusiong@up.edu.ph

Received October 2021; accepted December 2021

Abstract. *Solid waste management has become a concern in urban areas due to the increasing amount of garbage generated daily. To ensure the proper handling of a variety of wastes, waste disposal necessitates segregation. As a result, a reliable and accurate classification method is critical for resolving this problem. Transfer learning and different data augmentation techniques are used to train existing CNN-based waste classification models. However, the existing methodologies involve a lengthy training process. This paper proposes LitterNets, a model that achieves similar performance to existing models but requires much less training time. LitterNets is an ensemble of heterogeneous CNN-ELM models. The CNN-ELM models are distinct in terms of the pre-trained CNNs used as feature extractors, but the classification layer employs the same extreme learning machine network architecture. The experiments showed that the proposed approach significantly lowers training time and achieves results equivalent to existing state-of-the-art waste classification models, even though no data augmentation techniques were utilized during training. Further analysis revealed that using an ensemble of CNN-ELM models to increase accuracy is a simple yet effective strategy. For example, on the TrashNet test set, a LitterNets model consisting of three ResNet-ELM variants and two DenseNet-ELM variants achieved a classification accuracy of 93.97%, while training these five CNN-ELM models took only a total of 380 seconds, or 6 minutes and 20 seconds.*

Keywords: CNN-ELM model, Ensemble method, Transfer learning, Extreme learning machine, Waste classification

1. Introduction.

The quantity of garbage generated each day is growing as more people move to urban areas. As they go about their daily lives, urban residents create a variety of trash, including food waste, paper, cardboard, plastics, glass, metals, and bulky objects. As a result, solid waste management has become a priority in urban areas since garbage poses a threat to the environment as well as the health and safety of those who live there if it is not collected and disposed of promptly and properly. Furthermore, waste segregation is critical for the proper handling of various wastes [1]. Waste segregation necessitates a reliable and fast classification method, and it is the first step in establishing advanced waste disposal and segregation systems.

Conventional waste segregation and disposal methods rely on manual screening by human sorters, which is time-consuming, labor-intensive, and detrimental to the health of the sorters. An alternative approach is to utilize automatic waste sorting systems, as demonstrated by current CNN-based waste classification models trained through transfer learning and various data augmentation methods [2, 3, 4, 5]. Although this framework yields excellent accuracy, it takes longer to develop a trained model due to the additional steps. For example, Bircanoglu et al. [2] reported that their proposed model's longest

DOI: 10.24507/icicel.16.09.943

training session took three hours to complete the three hundred epochs using a GTX 1080 Ti GPU.

In contrast, LitterNets, the proposed ensemble of CNN-ELM models, can achieve equivalent performance to current models while taking significantly less training time. LitterNets is an ensemble of heterogeneous CNN-ELM models, where each CNN-ELM model differs in terms of the pre-trained CNN used as a feature extractor, but the classification layer is created using the same extreme learning machine network architecture. Extreme learning machine (ELM) is a single hidden layer feedforward neural network (SLFN) training method that converges considerably quicker than conventional methods and produces remarkable results without the need for iteratively tuning the hidden weight parameters [6]. As a result, each CNN-ELM model is trained in a single epoch, and no data augmentation techniques were applied to the training images.

Essentially, this work makes the following contributions.

1) It demonstrates the reliability and effectiveness of a heterogeneous ensemble of CNN-ELM models for automatic waste classification by utilizing transfer learning and extreme learning machines.
2) It trains the proposed hybrid architecture in a single epoch in a supervised manner without using any data augmentation methods. In this network model, the lower layers of a pre-trained CNN are retained and used as a feature extractor, while the classification layer is built using an extreme learning machine.
3) It reveals that the CNN-ELM framework significantly reduces training time based on the number of epochs and produces results comparable to those obtained by existing state-of-the-art waste classification models, despite the absence of data augmentation techniques during training, and LitterNets is a simple yet effective method for increasing accuracy.

The remainder of the paper is divided into the following sections. Section 2 examines existing methods for automatically classifying waste, whereas Section 3 describes the proposed methodology. Section 4 discusses the experiments and results. Finally, Section 5 presents the conclusions of this study.

2. **Related Work.** This section examines some of the current methods for automatically classifying waste. These existing techniques for automatic waste classification vary in their choice of CNN architectures, their use of data augmentation techniques to increase the size of the training set, and their use of pre-trained CNNs for transfer learning, with the lower layers preserved and used as a feature extractor while the classification layer is trained to correctly classify wastes. Additionally, they are distinct in terms of the datasets used and the number of waste classes considered.

For example, some research studies selected multiple well-known CNN models and performed other steps such as enhancing the training data through data augmentation techniques, training each model from scratch and fine-tuning each of the trained models, and implementing other transfer learning approaches. Afterwards, these models were evaluated and a comparison of their performance was conducted [2, 4, 5, 7]. Moreover, several researchers also presented new waste datasets [3, 8, 9, 10] rather than just training their models on the well-known TrashNet dataset [11]. However, the majority of the newly created datasets are not publicly available.

Nevertheless, the existing automatic waste classification models have several limitations. First, training must be conducted in many epochs. Second, the training period is lengthy. Third, data augmentation techniques are used to improve accuracy. Finally, even though they trained several CNN models, they did not examine how using the ensemble method can improve performance.

To address these limitations, some researchers proposed using a hybrid architecture that combines CNN and ELM for image classification. The CNN-ELM framework has been used to solve real-world problems such as DNA damage classification [12], age and gender classification [13], cervical cancer classification [14], electrocardiogram (ECG) signal classification [15], and accident image classification [16]. In these studies, CNN-ELM models outperform CNN-only models.

LitterNets is proposed as an alternative model that addresses these limitations and employs the ensemble method to achieve better performance.

3. **LitterNets: An Ensemble of CNN-ELM Models.** This section describes the proposed ensemble of CNN-ELM models for automatic waste classification and each of the CNN-ELM models. Also, it is necessary to have a dataset consisting of various waste images with the associated ground-truth class labels. An overview of the LitterNets model for automatic waste classification is shown in Figure 1 while each CNN-ELM architecture in the LitterNets model follows the same framework depicted in Figure 2.

3.1. **LitterNets for automatic waste classification.** The LitterNets, as depicted in Figure 1, combines several trained CNN-ELM models and each of these trained models makes a prediction given an input image. Afterwards, a majority voting scheme is employed where the class label that receives the highest number of votes will be considered the ensemble model's final class label.

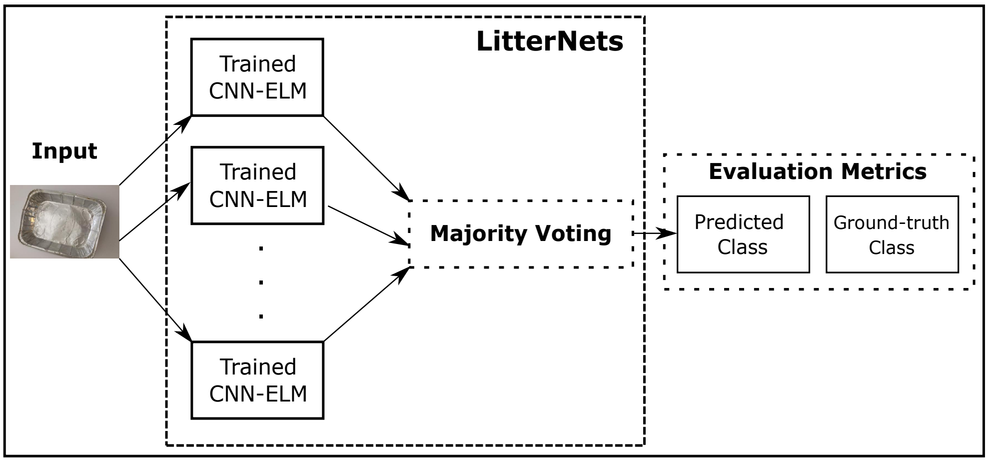

Figure 1. LitterNets: An ensemble of CNN-ELM models

3.2. **The CNN-ELM framework.** As shown in Figure 2, the training and test phases of each CNN-ELM model are pretty straightforward. Initially, the lower layers of a pre-trained CNN are preserved and used as a feature extractor, but the classification layer is discarded because, in the CNN-ELM model, the classification layer is built using the extreme learning machine. Thus, during the training phase, only the ELM component is trained.

Training the CNN-ELM is fast since it can be done in a single epoch, and no data augmentation methods were applied on the training set. On the other hand, in the test phase, evaluation metrics were used to quantify the performance of the trained CNN-ELM model using the test set.

For the single-layer ELM network, the number of input neurons is the same as the number of parameters in the last layer of the pre-trained CNN's feature extraction layer, the hidden layer has 100 neurons, and the number of output neurons in the output layer is the same as the number of class labels. In addition, the activation function of all neurons in the ELM is the Gaussian error linear unit or GELU [17].

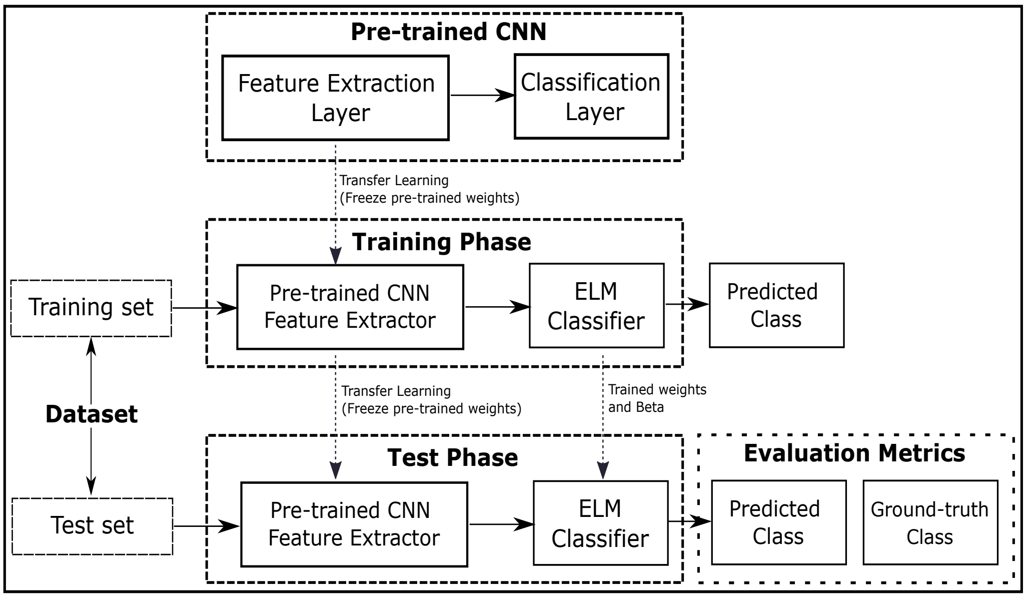

FIGURE 2. The CNN-ELM framework for LitterNets

3.3. **Evaluation metrics.** This section delves into the evaluation metrics that were employed in this research work. Evaluation metrics are used to measure the performance of trained models, or how effectively they predict unknown occurrences. For classification problems, there are four common evaluation metrics: accuracy, recall, precision, and F1-score [18]. The proportion of correct outcomes to the total number of instances examined is defined as accuracy in Equation (1). Precision is defined as the fraction of true positives among all positive predictions made by the classifier, as described in Equation (2). Recall is the fraction of true positives among all samples that should have been classified as positive, as defined in Equation (3). Finally, the F1-score is a precision-recall metric that strikes a good balance between these two metrics. A low rate of false positives and false negatives is necessary for an excellent F1-score. The F1-score metric is defined by Equation (4). In the following equations, TP signifies true positive, TN denotes true negative, FP means false positive, and FN denotes false negative.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1\text{-}score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4}$$

4. **Experimental Results and Discussion.** The dataset utilized in this study and the experiments conducted to evaluate the performance of the proposed model, LitterNets, are summarized in this section.

4.1. **The TrashNet dataset.** Yang and Thung of Stanford University produced the TrashNet dataset [11]. This dataset contains colored images of six different types of waste, each exclusively belonging to one type of waste. As shown in Figure 3, the six types of waste included in the TrashNet dataset are glass, paper, cardboard, plastic, metal, and general trash. All images in the dataset have an image resolution of 512×384. The total number of images in the dataset is 2,527, with the distribution of images by class given in Table 1.

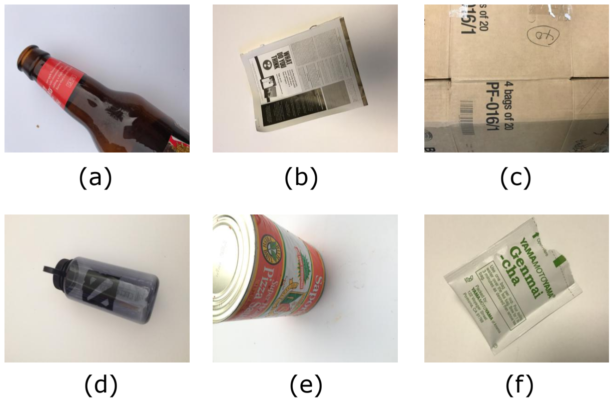

Figure 3. Sample images from the TrashNet dataset: (a) Glass; (b) paper; (c) cardboard; (d) plastic; (e) metal; and (f) trash

Table 1. TrashNet dataset: Distribution of images by class

Type	**Training set**	**Test set**	**Validation set**	**Total**
Glass	354	82	65	**501**
Paper	403	108	83	**594**
Cardboard	287	70	46	**403**
Plastic	347	74	61	**482**
Metal	286	68	56	**410**
Trash	91	29	17	**137**
Total	**1,768**	**431**	**328**	**2,527**

4.2. **Experiments and results.** LitterNets, the proposed ensemble of CNN-ELM models for automatic waste classification, and the models themselves were implemented using TensorFlow [19] and the Keras library [20]. The TrashNet test set was used to evaluate the accuracy of LitterNets and CNN-ELM models in identifying the six different waste classes. The pre-trained CNN's lower layers were frozen and used a feature extractor, but the classification layer was removed and replaced with an extreme learning machine having six output neurons in the output layer. The feature extraction layers did not get any additional training or fine-tuning. The dimension of all images in the TrashNet dataset was reduced to 224×224 to match the usual input image size of models trained on the ImageNet dataset [21]. To assess the performance of LitterNets, two experimental setups were conducted. The first setup entails training seven different CNN-ELMs and evaluating the effect of the pre-trained CNNs on the hybrid network's performance. The second setup aims to evaluate the impact of the different design choices in creating an ensemble of CNN-ELM models.

Table 2 shows the seven pre-trained CNNs used in this study, the network architecture of the single-layer ELMs, the time it took to train each CNN-ELM model using 1,768 images in the train set, and the time it took to evaluate each trained CNN-ELM using the test set containing 431 images. Each CNN-ELM model was trained in about two minutes, and each of the trained models was evaluated within twenty-one seconds using the test images. These findings show that the CNN-ELM architecture, which combines transfer learning with the extreme learning machine training algorithm, can significantly lower training time.

The performance of the seven CNN-ELM models and quantitative data obtained from the initial experimental setup using the evaluation metrics are presented in Table 3. The ResNet101-ELM model was by far the most effective, while the VGG16-ELM model was

Table 2. The different CNN-ELM models, together with their associated training and test times

CNN	ELM			Training time (in seconds)	Test time (in seconds)
	Input	Hidden	Output		
VGG16	25,088	100	6	54	13
VGG19	25,088	100	6	56	14
ResNet50	100,352	100	6	63	15
ResNet101	100,352	100	6	75	18
ResNet152	100,352	100	6	85	21
DenseNet121	50,176	100	6	72	18
DenseNet169	81,536	100	6	85	21

Table 3. The performance of the different CNN-ELM models using the evaluation metrics

CNN-ELM model	Accuracy	Precision	Recall	F1-score	Performance ranking
VGG16-ELM	79.81%	80.83%	77.11%	78.93%	7
VGG19-ELM	80.28%	79.37%	76.42%	77.86%	6
ResNet50-ELM	91.42%	91.37%	89.28%	90.31%	3
ResNet101-ELM	92.11%	92.19%	89.50%	90.83%	1
ResNet152-ELM	90.26%	90.30%	88.98%	89.63%	4
DensetNet121-ELM	86.54%	86.12%	84.04%	85.07%	5
DensetNet169-ELM	91.42%	91.48%	90.40%	90.94%	2

Table 4. The LitterNets models

3 CNN-ELMs	Training time	5 CNN-ELMs	Training time	7 CNN-ELMs	Training time
ResNet101-ELM	*75*	ResNet101-ELM	*75*	ResNet101-ELM	*75*
DenseNet169-ELM	*85*	DenseNet169-ELM	*85*	DenseNet169-ELM	*85*
ResNet50-ELM	*63*	ResNet50-ELM	*63*	ResNet50-ELM	*63*
		ResNet152-ELM	*85*	ResNet152-ELM	*85*
		DenseNet121-ELM	*72*	DenseNet121-ELM	*72*
				VGG19-ELM	*56*
				VGG16-ELM	*54*
Total training time	**223**		**380**		**490**

the least accurate. Despite having the same number of hidden and output neurons in their ELM network architecture, the performance of the CNN-ELM models differs. This performance variation can be attributed to the number of features generated by the feature extraction layer, with more features resulting in better performance. For example, the lowest three CNN-ELM models have fewer input features than the top three CNN-ELM models. Based on these results, the CNN-ELM model's performance is dependent on the number of input features used, and the more features, the more accurate the prediction, but each additional feature increases the training time and computational cost. It can also be observed that the hybrid model is effective even though the dataset is imbalanced, given that there are significantly fewer trash samples.

The three variants of LitterNets are shown in Table 4. The LitterNets model with three CNN-ELMs combined the top three most accurate trained models. The LitterNets model with five CNN-ELMs was a combination of the first five CNN-ELM models, whereas the LitterNets model with seven CNN-ELMs included all of the trained CNN-ELM models.

It can be observed that the total training time increases as more CNN-ELM models are added to the LitterNets model.

The results of the second experimental setup, which aimed to validate the performance of LitterNets, an ensemble of CNN-ELM models, are shown in Table 5. The quantitative data obtained in the experiments reveal that all of the LitterNets models outperformed individual CNN-ELM models in terms of accuracy. Among the three LitterNets variants tested, the LitterNets variant with five CNN-ELMs achieved the best accuracy rate of 93.97%. Furthermore, the LitterNets models with three CNN-ELM models and five CNN-ELM models outperformed the LitterNets variant with seven CNN-ELM models. Based on these results, finding a good combination of heterogeneous CNN-ELM models is tricky because adding more models to the ensemble does not always improve performance but it increases the training time of the ensemble.

Table 5. The performance of the LitterNets models using the evaluation metrics

LitterNets model	**Accuracy**	**Precision**	**Recall**	**F1-score**	**Test time per image (in seconds)**
3 CNN-ELMs	93.50%	93.96%	91.39%	92.66%	0.125
5 CNN-ELMs	**93.97%**	**94.37%**	**92.23%**	**93.29%**	0.211
7 CNN-ELMs	93.27%	93.74%	91.28%	92.49%	0.269

Finally, Table 6 compares the results of several models that used the TrashNet dataset. Except for LitterNets, previous research works require many epochs to train and fine-tune their models and each epoch requires several seconds to complete. On the other hand, each CNN-ELM model in LitterNets is trained in a single epoch, and it still achieved excellent performance.

Table 6. A comparison of the different models that used the TrashNet dataset

Author	**Model used**	**Epochs**	**Accuracy**
Ruiz et al. [5]	Inception-ResNet model	~ 55	88.66%
Bircanoglu et al. [2]	DenseNet121	$10 + 200$	95.00%
Bircanoglu et al. [2]	InceptionResNetV2	$10 + 200$	87.00%
Bircanoglu et al. [2]	RecycleNet	200	81.00%
Aral et al. [7]	InceptionV4	$7 + 120$	94.00%
Aral et al. [7]	DenseNet169	$7 + 120$	95.00%
Aral et al. [7]	DenseNet121	$10 + 100$	95.00%
Aral et al. [7]	InceptionV4	$10 + 200$	89.00%
Aral et al. [7]	MobileNet	$10 + 200$	84.00%
Best proposed model	LitterNets: 5 CNN-ELMs	$1 + 1 + 1 + 1 + 1$	93.97%

5. **Conclusions.** This paper introduced LitterNets, an ensemble of CNN-ELM models for automatic waste classification. LitterNets was trained on a single epoch, which resulted in a substantial reduction in training time. Furthermore, even though no data augmentation techniques were utilized during the training process, the proposed model had a similar performance to those obtained by existing state-of-the-art waste classification models. Most significantly, this paper demonstrated a simple and effective technique for increasing accuracy that leverages on the pre-trained CNNs as feature extractors, employs extreme learning machines for the classification layer, and combines multiple trained CNN-ELM models. Future research directions include extending the LitterNets framework to classify

and recognize other objects, as well as investigating different methodologies and establishing a criterion for model selection when constructing an ensemble architecture in order to combine the base models more effectively.

REFERENCES

[1] N. J. G. J. Bandara, J. P. A. Hettiaratchi, S. C. Wirasinghe and S. Pilapiiya, Relation of waste generation and composition to socio-economic factors: A case study, *Environmental Monitoring and Assessment*, vol.135, nos.1-3, pp.31-39, 2007.
[2] C. Bircanoglu, M. Atay, F. Beser, O. Genc and M. A. Kizrak, RecycleNet: Intelligent waste sorting using deep neural networks, *2018 Innovations in Intelligent Systems and Applications (INISTA)*, 2018.
[3] G. Huang, J. He, Z. Xu and G. Huang, A combination model based on transfer learning for waste classification, *Concurrency and Computation: Practice and Experience*, vol.32, no.19, 2020.
[4] W.-L. Mao, W.-C. Chen, C.-T. Wang and Y.-H. Lin, Recycling waste classification using optimized convolutional neural network, *Resources, Conservation and Recycling*, vol.164, DOI: 10.1016/j.resconrec.2020.105132, 2021.
[5] V. Ruiz, A. Sanchez, J. F. Velez and B. Raducanu, Automatic image-based waste classification, in *Bioinspired Systems and Biomedical Applications to Machine Learning. IWINAC 2019. Lecture Notes in Computer Science*, J. F. Vicente, J. Alvarez-Sanchez, F. de la Paz Lopez, J. T. Moreo and H. Adeli (eds.), Cham, Springer, 2019.
[6] J. Wang, S. Lu, S. H. Wang and Y. D. Zhang, A review on extreme learning machine, *Multimedia Tools and Applications*, 2021.
[7] R. A. Aral, S. R. Keskin, M. Kaya and M. Haciomeroglu, Classification of TrashNet dataset based on deep learning models, *2018 IEEE International Conference on Big Data*, 2018.
[8] H. Wang, Y. Li, L. M. Dang, J. Ko, D. Han and H. Moon, Smartphone-based bulky waste classification using convolutional neural networks, *Multimedia Tools and Applications*, vol.79, nos.39-40, pp.29411-29431, 2020.
[9] A. H. Vo, L. H. Son, M. T. Vo and T. Le, A novel framework for trash classification using deep transfer learning, *IEEE Access*, vol.7, pp.178631-178639, 2019.
[10] S. Frost, B. Tor, R. Agrawal and A. G. Forbes, CompostNet: An image classifier for meal waste, *2019 IEEE Global Humanitarian Technology Conference (GHTC)*, 2019.
[11] M. Yang and G. Thung, *Classification of Trash for Recyclability Status*, CS229 Project Report 2016, https://github.com/garythung/trashnet, Accessed on 1 June 2021.
[12] Y. T. Hafiyan, Afiahayati, R. D. Yanuaryska, E. Anarossi, V. M. Sutanto, J. Triyanto and Y. Sakakibara, A hybrid convolutional neural network-extreme learning machine with augmented dataset for DNA damage classification using comet assay from Buccal Mucosa sample, *International Journal of Innovative Computing, Information and Control*, vol.17, no.4, pp.1191-1201, 2021.
[13] M. Duan, K. Li, C. Yang and K. Li, A hybrid deep learning CNN-ELM for age and gender classification, *Neurocomputing*, vol.275, pp.448-461, 2018.
[14] 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.
[15] S. Zhou and B. Tan, Electrocardiogram soft computing using hybrid deep learning CNN-ELM, *Applied Soft Computing*, vol.86, DOI: 10.1016/j.asoc.2019.105778, 2020.
[16] A. Pashaei, M. Ghatee and H. Sajedi, Convolution neural network joint with mixture of extreme learning machines for feature extraction and classification of accident images, *Journal of Real-Time Image Processing*, vol.17, no.4, pp.1051-1066, 2019.
[17] D. Hendrycks and K. Gimpel, Gaussian error linear units (GELUs), *arXiv Preprint*, arXiv: 1606.08415, 2016.
[18] N. Japkowicz and M. Shah, *Evaluating Learning Algorithms: A Classification Perspective*, Cambridge University Press, Cambridge, 2011.
[19] M. Abadi, P. Barham, J. Chen, Z. Chen, Z. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard et al., TensorFlow: A system for large-scale machine learning, *The 12th USENIX Conference on Operating Systems Design and Implementation*, pp.265-283, 2016.
[20] T. P. Trappenberg, Neural networks and Keras, *Fundamentals of Machine Learning*, pp.66-90, 2019.
[21] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and F.-F. Li, ImageNet: A large-scale hierarchical image database, *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp.248-255, DOI: 10.1109/CVPR.2009.5206848, 2009.