Logo Recognition by Combining Deep Convolutional Models in a Parallel Structure

Mohammad Karimi
Electrical Enginering Department
Shahed University
Tehran, Iran
mhkarimi1@outlook.com

Alireza Behrad
Electrical Enginering Department
Shahed University
Tehran, Iran
behrad@shahed.ac.ir

***Abstract*— In this paper, a new approach is proposed for logo recognition using deep convolutional neural networks. Precise recognition of logos is of high importance in several applications like intelligent traffic control systems and copyright infringement. To enhance the efficiency of logo recognition, we have employed several strategies. In the first strategy, pre-trained deep models are employed for deep feature extraction and classification using the Support Vector Machine (SVM) classifiers. In the second strategy, existing pre-trained deep models are modified for logo recognition after transfer learning and fine-tuning. Finally, fine-tuned models are employed in a parallel structure to enhance the efficiency of logo recognition. We tested the proposed structures with Logos32-plus dataset and the results showed that combining fine-tuned deep models using a voting algorithm gives rise to the best recognition rate of 98.4%. The comparison of results of the proposed structure with a state-of-art deep approach for logo recognition shows the efficiency of the proposed approach.**

Keywords*— *logo recognition; deep learning; convolutional neural network

I. INTRODUCTION

Logo recognition is one of the most important tasks in machine vision. It spans a wide range of application like intelligent traffic control, copyright infringement and brand recognition in marketing and social media on the Internet. Traditionally logo recognition has been addressed by the handcrafted features and shallow classifiers. For instance, Kalantidis et al. introduced a triangulation-based scalable logo recognition algorithm [1]. They used a bag-of-words model that groups features in triples using multi-scale Delaunay triangulation. They extracted a simple signature for each triangle that incorporates both visual appearance and local geometry. They used Speeded Up Robust Features (SURF) algorithm [2] to extract scale invariant local features. In [3] an algorithm was proposed to recognize vehicle logos that is based on an optimized Scale-Invariant Feature Transform (SIFT)-based feature-matching scheme. It was shown that the enhanced matching approach increases the recognition accuracy in comparison with the standard SIFT-based feature-matching method. Pornpanomchai et al. proposed a logo recognition system to identify company logos [4]. This method uses simple features based on average RGB values and the histogram of edge points. Then a pattern matching algorithm is used for logo recognition. Romberg et al. leveraged the spatial layout of local features detected in the logo images for scalable logo recognition in the real-world images [5]. They extracted local descriptors using SIFT algorithm [6] derived from hessian-affine interest points [7]. Lei et al. introduced an approach for the classification of merchandise logos [8]. They leveraged a combination of local edge-based DAISY descriptor, spatial histogram, and salient region detection algorithm to enhance the efficiency of logo recognition.

Logo recognition using handcrafted feature needs feature engineering and the proper selection of features. Furthermore, scale and rotation invariant features should be extracted to bring scale and rotation invariant recognition property. However; extraction and matching scale and rotation invariant features are computationally expensive in general.

Machine vision algorithms are in transition from traditional approaches to modern methods. Recently deep learning models have revolutionized progress in machine vision. Deep Convolutional Neural Networks (DCNNs) use hierarchical learning strategy. Every layer in a deep network learns a feature of the image. Bottom layers in deep network learn low-level features like edge and color features while top layers represent high-level features. The fully connected layer at the end of network generates the feature map and works as a classifier. Finally, a softmax layer produces the score of each class in classification.

Recently, several deep learning approaches have been proposed to solve existing machine vision problems. For instance, deep models have been widely employed in various applications of machine vision, such as image classification [9], object detection [10, 11], image retrieval [12], graph mining [13], fingerprint spoofing detection [14], semantic segmentation [15], and translating video and images to natural language [16, 17].

Recently deep models have been successfully adopted for logo recognition and classification. Bianco et al. adopted a deep model for logo recognition [18]. Their model pipeline comprises a logo region proposal that is followed by a Convolutional Neural Network (CNN). Eggert et al. leveraged pre-trained deep CNNs for feature extraction and used a set of SVMs for the classification of the extracted features [19]. In

[20] the pre-trained DCNNs are employed for logo recognition after fine-tuning using logo images.

In this study, new algorithms are proposed for logo recognition using DCNNs. In the proposed algorithm three different strategies are used for efficient logo recognition. In the first strategy, the pre-trained deep models are used for deep feature extraction. Then logos are recognized by a trained Support Vector Machine (SVM) classifier. In the second strategy, transfer learning concept is used to fine-tune some pre-trained deep models for logo recognition. Finally, the fine-tuned deep models are used in a parallel structure to obtain more efficient logo recognition deep model.

This paper is organized as follows. In the next section, the proposed algorithms for logo recognition are described. Section III presents the results of experiments and the comparison of the proposed method with an existing state-of-the-art algorithm. Finally, we conclude the paper in Section IV.

II. Proposed Methods

In this paper, new structures based on deep models are proposed for logo recognition. Deep models need a large number of images for training. To counter the problem, we adopted the well-known pre-trained deep models for logo recognition. We use three strategies for logo recognition: 1- logo recognition using deep features, 2- fine-tuning the pre-trained DCNNs, and 3- combining fine-tuned DCNNs in a parallel structure.

A. Logo Recognition using Deep Features

Although DCNNs are used for combined feature extraction and classification, the outputs of various layers can be used as the features of the input image. Generally, the last layer of DCNN is a softmax layer that produces the score of each class of classification. The last fully connected layer before the softmax layer also generates a feature map that works as a classifier. Other layers of a DCNN can be used for feature extraction. Generally, the top layers of a DCNN are responsible for the high-level classification outputs while the bottom layers generate low-level features. Therefore; we adopted the output of the last layer before the classification fully connected layer for the feature extraction. We used five well-known and pre-trained DCNNs for feature extraction: 1- AlexNet [9], 2- VGG-16 [21], 3- VGG-19 [21], 4- GoogLeNet [22], and 5- ResNet [23]. Table I describes the layer number and the dimension of features that are used logo recognition in the proposed algorithm.

We use SVM classifiers to categorize the extracted deep features and recognize logos. SVM is a binary classifier that categorizes the deep features by finding a decision plane or hyperplane. SVM classifiers adopt different kernel functions such as linear, polynomial, radial basis function (RBF) and sigmoid. We experimentally use the linear kernel to categorize deep features and recognize logos. SVM is inherently a binary classifier; multi-class logo recognition can be solved by using two basic strategies:

- One-against-all approach: for an m-class logo recognition m SVMs are trained. Each SVM separates a single class from all other classes.
- One-against-one approach: for an m-class logo recognition m(m-1)/2 SVMs are trained. Each of the SVMs separates two of the classes from each other.

One-against-one approach needs a high number of SVM classifiers for training. Therefore one-against-all approach is used in this study for logo recognition.

TABLE I. The Layer Number and the Dimension of Features That Are Used for Logo Recognition.

Pre-trained DCNN	Layer number for feature extraction	Feature dimension
AlexNet	Layer 20	4096
VGG-16	Layer 37	4096
VGG-19	Layer 42	4096
GoogLeNet	Layer 152	1024
ResNet	Layer 174	2048

B. Fine-tuning Pre-trained DCNNs

Training a DCNN with millions of parameters needs a large number of images. DCNNs like AlexNet are trained with subsets of ImageNet database [24]. However, existing logo databases do not include enough number of logo images for training a DCNN. To counter the problem, the concept of transfer learning can be adopted to transfer the knowledge of a pre-trained DCNN to a different but related DCNN. To this intent, we fined-tuned well-known and pre-trained DCNNs for logo recognition. The pre-trained DCNNs like AlexNet are generally trained by 1000-class subsets of ImageNet database. To fine-tune this DCNN for logo recognition, we remove the final 1000-way softmax and fully connected layers and replace them with new N-way softmax and fully connected layers, where N is the number of logo classes. The weights for the new fully connected layer should be initialized randomly. We initialize new weights by Xavier approach [25]. Xavier initialization causes the variance of feature maps to remain the same while passing through the various layers of a DCNN. Finally, the modified DCNN is trained with logo database using Stochastic Gradient Descent (SGD) approach. We fine-tuned five pre-trained DCNNs for logo recognition. comprising 1- AlexNet [9], 2- VGG-16 [21], 3- VGG-19 [21], 4- GoogLeNet [22], and ResNet [23].

It is important to note that the GoogLeNet DCNN employs two auxiliary classifiers that are connected to intermediate layers to handle the problem of vanishing gradient. To fine-tune GoogLeNet for logo recognition the softmax and last fully connected layers of these two layers must be redesigned for the logo recognition task. During the retraining process, the losses of two auxiliary classifiers are added to the main loss of the network with a discount weight of 0.3.

C. Combining Fine-tuned DCNNs

The aim of combining fine-tuned DCNNs is to enhance the efficiency of logo recognition by merging the feature maps of two or several DCNNs. Fig. 1 illustrates the proposed parallel structure for combining GoogLeNet and ResNet DCNNs. To parallelize GoogLeNet and ResNet DCNNs, the last fully connected and softmax layers of fine-tuned GoogLeNet and ResNet DCNNs are removed. Then the feature maps of modified GoogLeNet and ResNet models are merged by a concatenate layer to produce a feature map with a dimension of

3072. The concatenate layer is followed by Rectified Linear Unit (ReLU), fully connected, and softmax layers. Here N-way fully connected and softmax layers are adopted where N is the numbers logo classes. We initialize new weights by Xavier approach and retrain the network with logo database and SGD approach.

In the second approach, the three fine-tuned DCNNs are combined using a voting algorithm. Fig. 2 shows the structure of the combined network using a voting algorithm. In this structure, the output of three fine-tuned DCNNs including ResNet, GoogLeNet, and VGG-16 are combined using a voting algorithm as follows:

- If two or three of DCNNs produce the same label, this label is considered as the output label.
- When the output labels of three DCNNs are different, the output label is determined based on the maximum score approach.

III. Experimental Results

We tested the proposed structures in MATLAB environment using MatConvNet library [26] and NVIDIA Quadro P5000 GPU. To test the proposed structures and compare the results with a state-of-the-art approach, Logos-32plus database [18] is used. Logos-32plus dataset is an extension of FlickrLogos-32 dataset [5] with 32 different logo brands and 12312 logo instances. We use 50% of logo instances for train and the remaining logos for validation.

Table II shows the accuracy of logo recognition using deep features and SVM classifiers. We experimentally use SVM classifiers with linear kernels and one-against-all approach. The results of Table II show that the deep features extracted using ResNet DCNN achieve the best recognition accuracy of 97.20%.

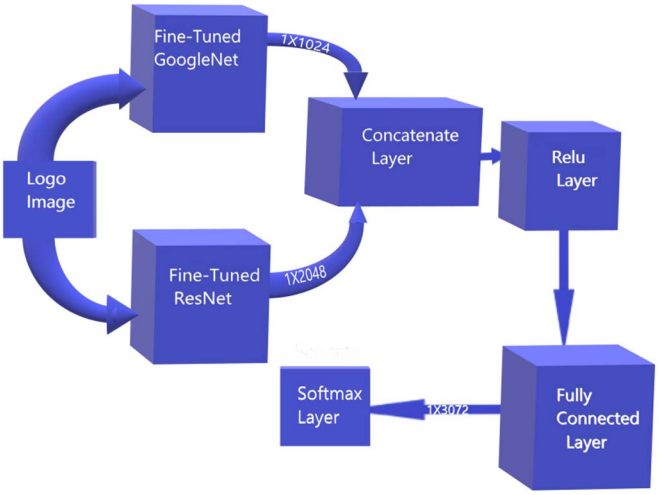

Fig. 1. The proposed parallel structure for combining GoogLeNet and ResNet DCNNs

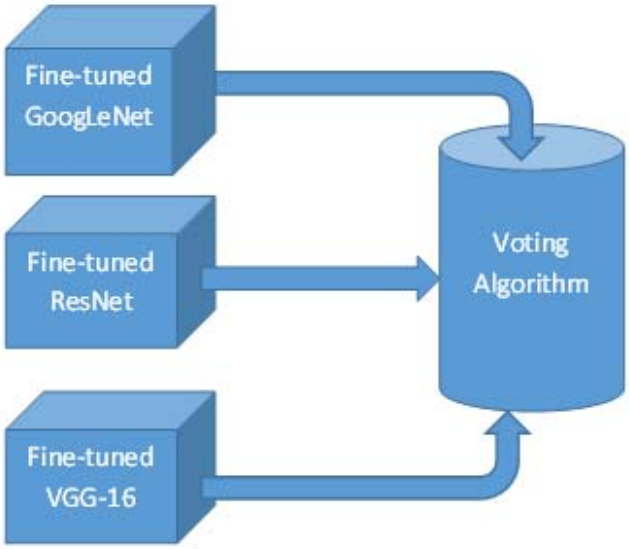

Fig. 2. Combining modified ResNet, GoogLeNet, and VGG-16 DCNNs using a voting algorithm.

TABLE II. The accuracy of logo recognition using deep features and SVM classifiers.

Pre-trained DCNN	Feature dimension	Accuracy (%)
AlexNet	4096	94.14%
VGG-16	4096	91.88%
VGG-19	4096	91.15%
GoogLeNet	1024	95.63%
ResNet	2048	97.20%

Fig. 3 shows the error of logo recognition versus the training epochs for the train and validation data. This figure shows the error of logo recognition for retraining modified DCNNs comprising AlexNet, VGG-16, VGG-19, GoogLeNet, ResNet and the parallel structure of Fig. 1. The results of Fig. 3 illustrate that the modified VGG-16 model has a good capability for logo recognition.

Table III shows various parameters for the training of modified DCNNs and the parallel structure of Fig. 1. Table IV illustrates the accuracy of logo recognition using the proposed approaches and the method of Bianco et al. [18]. The results of Table IV show that combining modified ResNet, GoogLeNet, and VGG-16 DCNNs using voting algorithm results in the highest accuracy among the proposed methods. The results of Table 4 indicate that the accuracy of 98.40% can be obtained by using the proposed voting structure. Furthermore, the comparison of the results of the proposed voting structure with Bianco et al. method [18] shows that the proposed voting structure enhances the accuracy up to 2.6%.

Fig. 4 shows sample logo images that are correctly recognized by the proposed algorithm. Fig. 5 illustrates sample falsely recognized logo images by the proposed algorithm. The results of Fig. 5 reveal that the main sources of errors are related to unclear and rotated logo images.

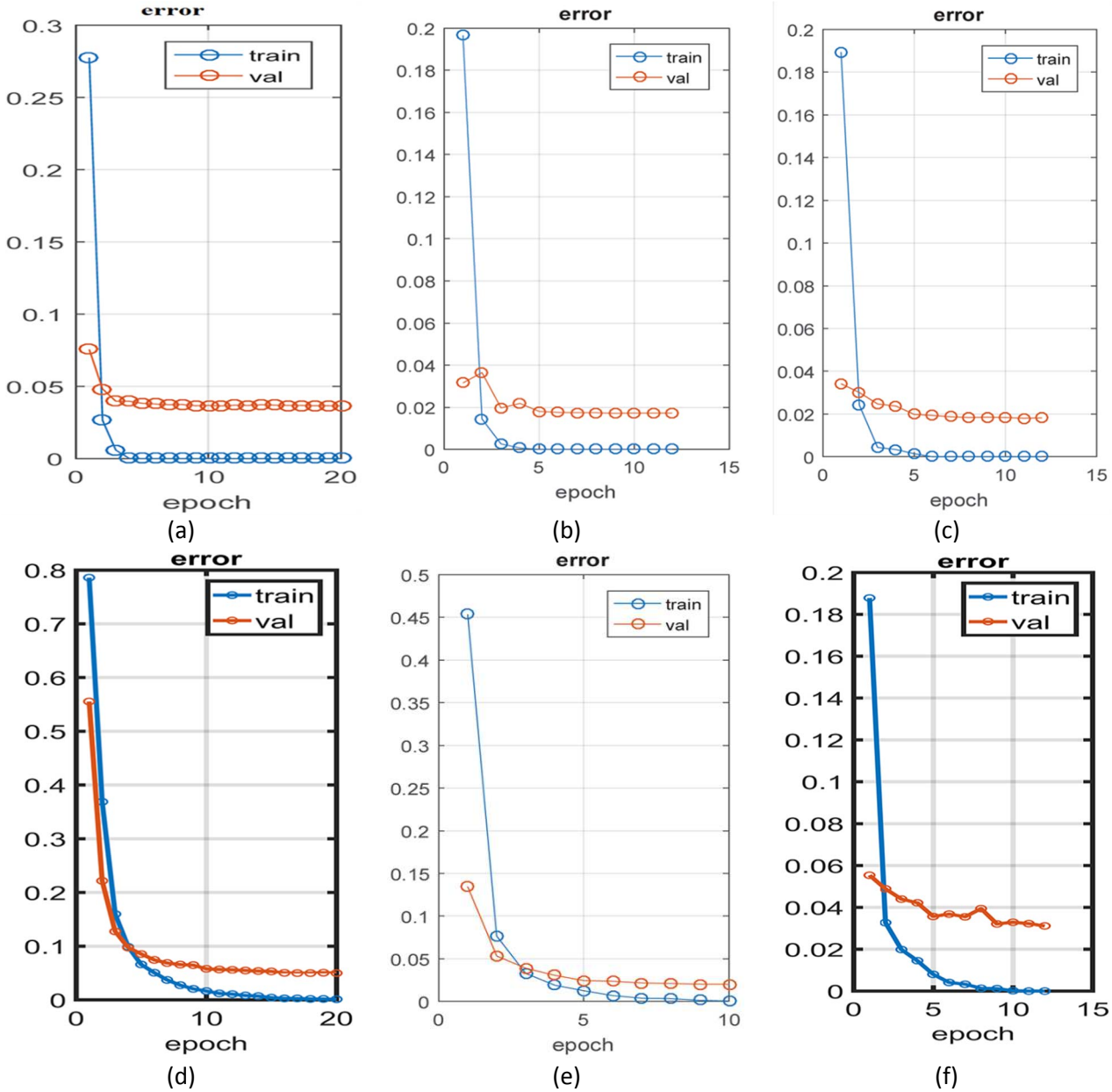

Fig. 3. The error of logo recognition versus training epochs for the train and validation data: (a), modified AlexNet structure, (b) modified VGG-16, (c) modified VGG-19, (d) modified GoogLeNet, (e) modified ResNet, and (f) the parallel structure of Fig. 1.

IV. Conclusions

In this study, three different structures were proposed for logo recognition. In the first strategy, deep features extracted by pre-trained deep models and SVM classifiers were employed for logo recognition. Our results showed that the deep features extracted using ResNet DCNN achieve the best recognition accuracy of 97.20%. In the second strategy, existing pre-trained deep models were modified for logo recognition. The results of logo recognition using Logos32-plus dataset showed that the modified and fine-tuned VGG-16 model achieves the highest accuracy of 98.30%. Finally, fine-tuned models are employed in a parallel structure to enhance the efficiency of logo recognition. The experimental results showed that that the accuracy of 98.40% can be obtained by using the proposed voting structure. Furthermore, the comparison of the results of the proposed voting structure with Bianco et al. method [18] showed that the proposed voting structure enhances the accuracy up to 2.6%.

TABLE III. VARIOUS PARAMETERS FOR THE TRAINING OF MODIFIED DCNNS AND THE PARALLEL STRUCTURE OF FIG. 1.

DCNN	Training approach	Weight Initialization	Learning rate	Weight Decay
Modified AlexNet structure	SGD	Xavier	0.05	0.05
Modified VGG-16 structure	SGD	Xavier	0.5	0.1
Modified VGG-19 structure	SGD	Xavier	0.1	0.1
Modified GoogLeNet structure	SGD	Xavier	0.0005	0.1
Modified ResNet structure	SGD	Xavier	0.1	0.1
The parallel structure of Fig. 1	SGD	Xavier	0.0005	0.1

TABLE IV. THE ACCURACY OF LOGO RECOGNITION USING THE PROPOSED APPROACHES AND THE METHOD OF BIANCO ET AL. [18].

DCNN	Accuracy (%)	Objective function
Modified AlexNet structure	96.40	0.18
Modified VGG-16 structure	98.30	0.08
Modified VGG-19 structure	98.20	0.1
Modified GoogLeNet structure	95	0.01
Modified ResNet structure	98	0.08
The parallel structure of Fig. 1	97.30	0.09
Combined structure of Fig. 2 (Voting structure)	98.40	-
Bianco et al. method [18]	95.8	-

Fig. 4. Sample correctly recognized logo images

Fig. 5. Some falsely recognized logo images.

REFERENCES

[1] Y. Kalantidis, L. G. Pueyo, M. Trevisiol, R. van Zwol, and Y. Avrithis, "Scalable triangulation-based logo recognition," in *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, 2011, p. 20.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Computer Vision and Image Understanding,* vol. 110, pp. 346-359, 2008.

[3] A. P. Psyllos, C. N. E. Anagnostopoulos, and E. Kayafas, "Vehicle Logo Recognition Using a SIFT-Based Enhanced Matching Scheme," *IEEE Transactions on Intelligent Transportation Systems,* vol. 11, pp. 322-328, 2010.

[4] C. Pornpanomchai, P. Boonsripornchai, P. Puttong, and C. Rattananirundorn, "Logo recognition system," in *2015 International Computer Science and Engineering Conference (ICSEC)*, 2015, pp. 1-6.

[5] S. Romberg, L. G. Pueyo, R. Lienhart, and R. Van Zwol, "Scalable logo recognition in real-world images," in *Proceedings of the 1st ACM International Conference on Multimedia Retrieval*, 2011, p. 25.

[6] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision,* vol. 60, pp. 91-110, 2004.

[7] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, *et al.*, "A comparison of affine region detectors," *International Journal of Computer Vision,* vol. 65, pp. 43-72, 2005.

[8] B. Lei, V. L. L. Thing, Y. Chen, and W. Y. Lim, "Logo Classification with Edge-Based DAISY Descriptor," in *2012 IEEE International Symposium on Multimedia*, 2012, pp. 222-228.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems* 2012, pp. 1097-1105.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 39, pp. 1137-1149, 2017.

[11] S. Ren, K. He, R. Girshick, X. Zhang, and J. Sun, "Object detection networks on convolutional feature maps," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 39, pp. 1476-1481, 2017.

[12] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "Deep image retrieval: Learning global representations for image search," in *European Conference on Computer Vision*, 2016, pp. 241-257.

[13] Y. Hou and L. B. Holder, "On Graph Mining With Deep Learning: Introducing Model R for Link Weight Prediction," *Journal of Artificial Intelligence and Soft Computing Research,* vol. 9, pp. 21-40, 2019.

[14] G. B. de Souza, D. F. da Silva Santos, R. G. Pires, A. N. Marana, and J. P. Papa, "Deep Features Extraction for Robust Fingerprint Spoofing

Attack Detection," *Journal of Artificial Intelligence and Soft Computing Research,* vol. 9, pp. 41-49, 2019.

[15] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 39, pp. 2481-2495, 2017.

[16] Y. Pan, T. Yao, H. Li, and T. Mei, "Video captioning with transferred semantic attributes," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 984-992.

[17] X. He and L. Deng, "Deep Learning for Image-to-Text Generation: A Technical Overview," *IEEE Signal Processing Magazine,* vol. 34, pp. 109-116, 2017.

[18] S. Bianco, M. Buzzelli, D. Mazzini, and R. Schettini, "Deep learning for logo recognition," *Neurocomputing,* vol. 245, pp. 23-30, 2017/07/05/ 2017.

[19] C. Eggert, A. Winschel, and R. Lienhart, "On the benefit of synthetic data for company logo detection," in *Proceedings of the 23rd ACM International Conference on Multimedia*, 2015, pp. 1283-1286.

[20] F. N. Iandola, A. Shen, P. Gao, and K. Keutzer, "Deeplogo: Hitting logo recognition with the deep neural network hammer," *arXiv preprint arXiv:1510.02131,* 2015.

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

[22] C. Szegedy, L. Wei, J. Yangqing, P. Sermanet, S. Reed, D. Anguelov, *et al.*, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1-9.

[23] 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.

[24] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, 2009, pp. 248-255.

[25] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 2010, pp. 249-256.

[26] A. Vedaldi and K. Lenc, "Matconvnet: Convolutional neural networks for matlab," in *Proceedings of the 23rd ACM International Conference on Multimedia*, 2015, pp. 689-692.