




Image-Based Animal Recognition based on Transfer Learning

Reconocimiento de animales desde imágenes utilizando aprendizaje por transferencia

G. S. Gomez-Gomez  ; D. Collazos-Huertas  ; A. M. Álvarez-Meza 
 DOI: <https://doi.org/10.22517/23447214.24538>
 Artículo de investigación científica y tecnológica

Abstract— Automatic image-based recognition systems have been widely used to solve different computer vision tasks. In particular, animals' identification in farms is a research field of interest for computer vision and the agriculture community. It is then necessary to develop robust and precise algorithms to support detection, recognition, and monitoring tasks to enhance farm management. Traditionally, deep learning approaches have been proposed to solve image-based detection tasks. Nonetheless, databases holding many instances are required to achieve competitive performances, not to mention the hyperparameter tuning, the noisy images, and the low-resolution issues. In this paper, we propose a transfer learning approach for image-based animal recognition. We enhance a pre-trained Convolutional Neural Network based on the ResNet101 model for animal classification from noisy and low-quality images, holding few samples. First, a dog vs. cat task is tested from the well-known CIFAR database. Further, a cow vs. no cow database is built to test our transfer learning approach. The achieved results show competitive classification performance using different types of architectures compared to state-of-the-art methodologies.

Index Terms— Animal recognition, Computer vision, deep learning, image processing, transfer learning.

Resumen— Los sistemas de reconocimiento automático basados en imágenes se han utilizado ampliamente para resolver diferentes tareas de visión por computador. En particular, la identificación de animales en granjas es un campo de investigación de interés para comunidad relacionada con visión artificial y agricultura. En este sentido, es necesario desarrollar algoritmos robustos y precisos para respaldar las tareas de detección, reconocimiento y monitoreo, en aras de apoyar la gestión de granjas en agricultura. Tradicionalmente, se han propuesto enfoques de aprendizaje profundo para resolver tareas de detección basadas en imágenes. No obstante, se requieren de bases de datos con muchas instancias para lograr un rendimiento competitivo, sin mencionar los problemas de ruido y baja resolución en las imágenes y el ajuste de hiperparámetros. En este artículo, proponemos un enfoque de aprendizaje por transferencia para el reconocimiento de animales basado en imágenes. En particular, mejoramos un modelo de red neuronal convolucional basado en la arquitectura ResNet101,

previamente entrenado para la clasificación de animales a partir de imágenes ruidosas y de baja calidad, con pocas muestras. Primero, se prueba una tarea de perro contra gato a partir de la conocida base de datos CIFAR. Además, se crea una base de datos de vaca versus no vaca para probar nuestro enfoque de aprendizaje por transferencia. Los resultados obtenidos muestran un rendimiento de clasificación competitivo utilizando diferentes tipos de arquitecturas, en comparación con las metodologías actuales.

Palabras claves— Reconocimiento de animales, visión por computador, aprendizaje profundo, procesamiento de imágenes, aprendizaje por transferencia.

I. INTRODUCTION

IN recent years, advances in deep learning have encouraged the generation of novelty learning algorithms based on Convolutional Neural Networks--(CNN) to solve a wide variety of problems focusing on computer vision tasks such as visual tracking, segmentation, and image classification [1-4]. Indeed, relevant applications such as smart farming and medicine require robust computer vision systems to support diagnosis and monitoring tasks [5-8].

However, the amount of data, the storage, and the long training time are some limitations that CNN-based models provide. Namely, the volume of information is directly related to the overfitting and performance decaying when evaluating new sample sets [9,10]. Several approaches solve the overfitting issues by creating synthetic samples, penalizing the loss function, and regularizing the architectures [12-13]. Still, many samples are required, not to mention the hyperparameter tuning drawbacks [14,15].

Transfer learning has recently emerged as an alternative to reuse pre-trained models (on large databases) to improve a specific task's performance and robustness [16-18]. Such a strategy aims to exploit the ability to directly assign the knowledge acquired by a network model to solve similar problems as an alternative to lead machine learning problems,

A. Álvarez-Meza, is with is a full-time professor, Signal Processing and Recognition Group, Universidad Nacional de Colombia Sede Manizales, Colombia (e-mail: amalvarezme@unal.edu.co)

This manuscript was sent on October 21, 2020 and accepted on June 30, 2021.

G. S. Gomez-Gomez, is with is a student of Engineering Program, Signal Processing and Recognition Group, Universidad Nacional de Colombia Sede Manizales, Colombia (e-mail: gsgomezg@unal.edu.co)

D. Collazos-Huertas, is with is a PhD student, Signal Processing and Recognition Group Universidad Nacional de Colombia Sede Manizales, Colombia (e-mail: dfcollazosh@unal.edu.co)



i.e., image-based recognition, from small datasets. Notwithstanding, the main problem relies on identifying an effective use of this approach by enhancing a pre-trained network's parameters based on the new instances [19]. In particular, for image-based object recognition problems, some CNN-based approaches based on pre-trained architectures are proposed. AlexNet, GoogleNet, VGG-16, and ResNet-based networks are commonly applied on the well-known CIFAR and ImageNet datasets. Nonetheless, low accuracy is obtained, and complex hyperparameter tuning is required, besides the low resolution and noisy data challenges [21,22].

Here, we propose an image-based animal recognition approach based on transfer learning. We aim to classify cat vs. dog (from the well-known CIFAR database) and cow vs. no-cow (from a custom-built database) for concrete testing. In short, our strategy aims to: i) attain competitive accuracy results regarding image-based animal recognition tasks, even for small databases, ii) properly tune the required hyperparameters, and iii) deal with the low resolution and noisy sample issues. In this sense, our approach employed the ResNet101 architecture and compared the achieved performance against other well-known state-of-the-art architectures, e.g., GoogleNet-2014, Vgg16, and ResNet50. Achieved results demonstrate how ResNet101 coupled with transfer learning favors the discrimination of images in small datasets. Besides, obtained performances show that the transfer learning method is more effective in classifying pixelated images. Our proposal could be an alternative to support computer vision tasks, i.e., medical image processing and intelligent farming systems, from databases with few samples.

The rest of the manuscript is organized as follows. Sections II and III presents the methods. Sections IV and V describe the experimental setup and the obtained results. Finally, Section VI outlines the conclusions and future work.

II. IMAGE-BASED RECOGNITION USING DEEP LEARNING

Let $\{X_i \in R^{W \times H \times C}, y_i \in N\}_{i=1}^I$ be an input image set, holding I samples in C color channels, sizing WH pixels, and equipped with output labels y_i . The training of a Deep Learning-based image recognition model is twofold: feature mapping learning based on convolutional neural networks (CNN) and multilayer perceptron-based classification.

The first stage exploits the local spatial correlation of input images through a convolutional filter arrangement $\{K_{n,l}\}^{N_l}$, for which a square-shaped layer kernel, sizing $P \times P \times C$, explores the spatial relationships between pixels. Note that the number of kernels depends on the number of layers, $n \in \{1, 2, \dots, N_l\}$. Thus, the convolutional operation projects stepwise a given image X , as follows:

$$\hat{X}_L = (\phi_L \circ \dots \circ \phi_1)(X), \quad (1)$$

where:

$$\phi_l(\hat{X}_{l-1}) = \gamma_l(K_{n,l} \otimes \hat{X}_{l-1} + B_{n,l}). \quad (2)$$

The convolutional layer in Eq. (2) holds the nonlinear activation $\gamma_l: R^{W_l \times H_l \times C} \rightarrow R^{W_l \times H_l \times C}$, $\hat{X}_l \in R^{W_l \times H_l \times C}$ is the l -th CNN feature map (being $\hat{X}_0 = X$), and $B_{n,l} \in R^{W_l \times H_l \times C}$ is the bias matrix ($l \in \{0, 1, \dots, L\}$). Notations \otimes and \circ stand for convolution operator and function composition, respectively. Besides, $W_l < W$ and $H_l < H$. Overall, the feature maps allow extracting relevant patterns concerning the spatial relationships among pixels.

In turn, a multi-layer perceptron-based classifier is applied on the L CNN-based feature map, yielding:

$$\hat{y} = (\phi_D \circ \dots \circ \phi_1)(u_0), \quad (3)$$

where $\phi_d(u_{d-1}) = \eta_d(A_d u_{d-1} + \alpha_d)$ is a dense layer ruled by the non-linear activation function $\eta_d: R^{P'_d} \rightarrow R^{P'_d}$, $P'_d \in N$ is the number of neurons at the d -th layer, $d \in \{0, \dots, D\}$, ($d = 0$ is the initial concatenation before the classification layer), $A_d \in R^{P'_d \times P'_{d-1}}$ is a weighting matrix that contains all connection weights between the preceding neurons, $\alpha_d \in R^{P'_d}$ is a bias vector, and $u_d \in R^{P'_d}$ is the d -th hidden layer vector that is iteratively updated as $u_d = \phi(u_{d-1})$, from the input flattened vector $u_0 = [\text{vec}(\hat{X}_L)]$, sizing $\xi = W'H'C \sum_{l \in L} N_l$, with $W' < W$, $H' < H$, after concatenating all matrix rows across the C color channels.

Note that the Deep Learning-based image recognition model estimates the predicted labels under the optimized trainable parameters $\theta = \{K_{n,l}, A_d, b_{n,l}, \alpha_d\}$, as seen in Eqs. (1-3), which are minimized in terms of the output labels as follows:

$$\theta^* = \arg \min_{\{K_{n,l}, A_d, b_{n,l}, \alpha_d\}} (L(\hat{y}_i, y_i | \theta); \forall i \in I), \quad (4)$$

where $L: N \times N \rightarrow R$ is a given loss function, i.e., mean square error or cross-entropy, that is solved through a mini-batch based gradient descend procedure using back-propagation and automatic differentiation [22,23].

III. IMAGE-BASED ANIMAL RECOGNITION USING TRANSFER LEARNING

Transfer learning aims to reuse trained models across similar tasks. It is recently a popular approach in deep methods, where pre-trained models on large databases are used as the starting point on similar problems. In particular, such a process repeats the architecture, and fixing the weights of low-level layers, demonstrates exciting results for computer vision and natural language processing tasks. Concerning the image-based recognition problem, the most common CNN-based architectures include the VGG-16, the GoogleNet-2014, and the ResNet. Namely, the VGG-16 holds a low-complex architecture compared to the remaining networks since it contains a relatively small number of sequential convolutional layers. The GoogleNet-2014, also known as InceptionVn, where n refers to the Google updated version, presents a startup module that acts as extractors of multi-level features. At last, the ResNet, with its two versions ResNet50 and ResNet101, contains a more in-depth architecture than VGG-16 and

GoogleNet-2014, combining convolutional layers with residual modules. Table 1 summarizes the essential properties of the networks above for image-based recognition.

This study uses the pre-trained parameters of the networks exposed in Table I, obtained from the reduced version of the ImageNet collection (including 1000 categories, each with approximately 1000 images), as a common reference point for evaluating large-scale image classification models [4]. The proposed methodology consists of using the learned parameter from the ImageNet set to initialize the parameters of a particular network. Namely, we fix the lower layers parameters to initialize the feature mapping generation stage. Then, we solve the following optimization problem:

$$\theta^\dagger = \arg \min_{\{K_{n,l}, A_d, b_{n,l}, \alpha_d\}} (L(\hat{y}_i, y_i | \theta); \forall i \in I), \quad (5)$$

where $\theta^\dagger = \{K_{n,l}\} \cup \{K_{n,l}, A_d, b_{n,l}, \alpha_d\}$, holds the fixed low-level CNN kernels $\{K_{n,l}\}$, the high-level CNN kernels $\{K_{n,l'}\}$ ($l' < l$); and the multi-layer perceptron parameters (fully connected layers) $\{A_d, b_{n,l}, \alpha_d\}$. It is worth mentioning that the transfer learning is encoded in the low-level CNN kernels, which are trained for each architecture in Table I from the ImageNet dataset. Meanwhile, the remaining parameters (high-level CNN kernels and multi-layer perceptron variables) must be optimized on the animal image-based recognition dataset of interest through Eq. (5). Therefore, the recognition model will not have to learn from scratch all the low-level structures in most pictures; it will only have to know the higher-level structures. Additionally, it will not only speed up training considerably but will also require much less training data. Finally, note that the output layer must be updated concerning the number of considered classes.

TABLE I
DEEP LEARNING MODELS FOR IMAGE-BASED RECOGNITION

Architecture	Trained layers	Description
VGG-16	16	It holds thirteen convolutional layers, each one with a set of 3x3 filters and a ReLu activation, followed by three fully connected layers, the first two with 4096 neurons and the third one with 1000 neurons.
GoogleNet-2014	22	It has a sequential block of five convolutional layers and eleven starter modules, and one fully connected layer of 1000 neurons. Also, it uses the global average for the fully connected layer.
ResNet-50	50	It holds 50 convolutional layers and 16 residual units.
ResNet-101	101	It holds 33 start modules and 99 convolutional layers with ReLu-based normalization and activation. Besides, a global average is applied to the fully connected layer.

IV. EXPERIMENTAL SETUP

The proposed image-based animal recognition scheme is presented in Fig. 1. First, a preprocessing stage is carried out to adjust the image dimension required by each studied architecture. We fix the input dimension as 224x224 for VGG16, ResNet50, ResNet101 architectures, and 229x229 for the GoogleNet-2014. Then, we extract the spatial features selecting trainable and non-trainable CNN-based layers for each network, as presented in Table II. Further, as explained in Section III, we carry out the training and validation procedure, optimizing the trainable parameters based on the loss function chosen. Finally, the classification performance is measured.

Our approach is tested on two databases. Firstly, we use the widely known CIFAR10² dataset as a benchmark for image-based object recognition from noisy and low-resolution samples [1,3,11]. CIFAR10 collects images of size 32x32 pixels, holding ten classes (airplane, automobile, bird, cat, deer, dog, frog, horse, boat, and truck), for a total of 60.000 images, divided into 50.000 samples for training and 10.000 for testing. We built a data subset composed of 2.000 images belonging to two classes: cat and dog. Then, we split this subset, 80% for training, and 20% for testing [24,25]. In turn, we collect a small cow database, termed IMG COW, holding cow and no cow instances captured from farm-based scenarios. Chiefly, the IMG COW dataset is composed of 1.500 images, as follows: cow (500 samples), chicken (200 samples), and horse (200 samples), all capture from the Animals-10³ dataset. The remaining 600 illustrations were taken from the training set of the CIFAR10 collection. Again, 80% of the samples are randomly selected as the training set, and the remaining 20% for testing. Fig. 2 illustrates some examples of the studied datasets.

The binary cross-entropy (BCE) is used to solve the optimization problem in Eq. (5). Then, the loss in logistic between the real probability and the predicted one is computed as follows:

$$L(\hat{y}_i, y_i) = \frac{-1}{N} \sum_{i=1}^I y_i \log(p(\hat{y}_i)) + (1 - y_i) \log(1 - p(\hat{y}_i)) \quad (6)$$

To evaluate our transfer learning proposal, we carried out three experiments for each architecture described in Table 1. The first experiment aims to compare the model's performance for the CIFAR10 data subset. The second experiment tests each model in the IMG COW dataset. Finally, a method comparison is presented with state-of-the-art approaches [20,21].

In all experiments, we set the following hyperparameters: The optimizer is fixed based on the adaptive estimation of moments, called the Adaptive Moment Estimation (ADAM) algorithm [4], using a learning rate value of 0.001 and 250 epochs. Moreover, the pooling size is fixed as 64. The experimental codes were developed using TensorFlow 2 and are publicly available on Github⁴. Concerning the performance criteria, the following metrics are considered:

² <https://www.cs.toronto.edu/~kriz/cifar.html>

³ <https://www.kaggle.com/alessiocorrado99/animals10>

⁴ <https://github.com/gloria256/EXPERIMENTOS-ARTICULO-1>

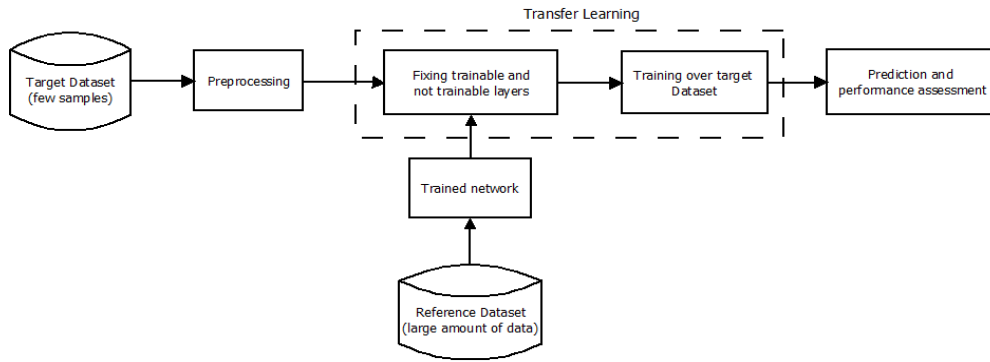


Fig. 1. Image-based animal recognition sketch using transfer learning. The VGG-16, GoogleNet-2014, ResNet-50, and ResNet-101 architectures are tested as trained networks. As a Reference Dataset, ImageNet collection is employed, and as Target Dataset, a dog vs. cat and a cow vs. no cow databases are built.

$$Acc = \frac{100(TP + TN)}{TP + FP + TN + FN} \quad [\%] \quad (7)$$

$$Acc_c = \frac{100 TP_c}{N_c} \quad [\%] \quad (8)$$

$$Re = \frac{100 TP}{TP + FN} \quad [\%] \quad (9)$$

$$Pr = \frac{100 TP}{TP + FP} \quad [\%] \quad (10)$$

$$F1 = \frac{200(Pr * Re)}{Pr + Re} \quad [\%] \quad (11)$$

where $Acc, Acc_c, Re, Pr, F1 \in [0,100][\%]$ stand for Accuracy, Accuracy per class, Recall, Precision, and F1-score, respectively. Namely, TP, TN, FN , and FP represent the True Positive, True Negative, False Negative, and False Positive rates. Specifically, the Pr metric gives us the quality of the prediction. But this would not be very useful since the classifier would ignore all but one positive instance. But this would not be very useful, since the classifier would ignore all but one positive instance. So Pr is typically used along with the Re metric, also called *sensitivity* or the *true positive rate*, which is the ratio of positive instances that are correctly detected by the classifier and gives us the quantity about what percentage of the positive class have we been able to identify. Finally, the $F1$ -score combines precision and recall in a single measure and consists of the harmonic mean between them. Whereas the regular mean treats all values equally, the harmonic mean gives much more weight to low values. As a result, the classifier will only get a high $F1$ if both recall and precision are high.

TABLE II
DEEP LEARNING MODELS FOR IMAGE-BASED RECOGNITION

Method	Not trainable Layers
VGG-16	1 to 13
GoogleNet-2014	1 to 81
ResNet-50	1 to 49
ResNet-101	1 to 100

V. RESULTS AND DISCUSSION

Table III shows the results of the first experiment (cat vs. dog from CIFAR10 subset). The ResNet101 network obtains a performance of 76.2% and outperforms in 1.2%, 3%, and 16.2% the other ResNet50, Vgg16, and GoogleNet-2014 architectures, respectively. Besides, the obtained results report that the transfer learning approach generates better performance in architectures with residual units: ResNet101 and ResNet50, whose difference in all the evaluation metrics does not exceed 1.3%. The overall ResNet101's account (our approach) exceeds the remaining architectures by 6.8% on average.

Table IV shows the achieved performances of the second experiment (IMGCOW dataset). As seen, the GoogleNet-2014 network obtains the lowest classification accuracy. Again, models composed of residual units, i.e., ResNet50 and ResNet101, achieve the highest performance. However, both achieve similar accuracy, indicating that the number of residual units does not drastically influence the IMGCOW data's recognition. Despite the networks Vgg16, ResNet50, and ResNet101 being competitive, the ResNet101 is ranked as the best performing, with an achieved accuracy of 98.3%.

The transfer learning effectiveness is presented in Tables III and IV. These results show that transfer learning provides better performance in classifying images that do not offer visual distortion, such as pixelation. Specifically, Table III reports the obtained results dealing with fully pixelated images (CIFAR10 subset), whose classification accuracy ranges from 60% to 70.2% in all considered evaluation metrics for each model. While Table IV shows the achieved performance using 60% of images without distortion (IMGCOW). In this case, the models yield a classification performance fluctuating between 78% and 98.3%. Of note, ResNet-based transfer learning allows dealing with imbalance issues and noisy scenarios, as reported in the Precision, Recall, and F1 scores, mitigating false positive and false negative predictions. Then, ResNet-based representation favors the model generalization to code each class's relevant patterns.

Finally, Table V shows the results of the third experiment. In this case, we carried out a method comparison between the architecture that yields the highest performance in the guidelines of Table III, e.g., ResNet101, and deep learning-based state-of-the-art classifiers. Specifically, we consider the

results for cat vs. dog reported in [1,3]. The attained results demonstrate that our ResNet101-based transfer learning approach can obtain comparable performance compared to ResNet50 and GoogleNet architectures without transfer learning. It is worth mentioning that our method employs a much lower number of samples than those used in the studied state-of-the-art.

TABLE III
CIFAR10 DATA SUBSET RESULTS (CAT VS. DOG) USING TRANSFER LEARNING (TL). RESULTS ARE DISPLAYED IN [%].

Architecture+TL	Precision	Recall	F1	Accuracy
GoogleNet-2014	60	60	60	60
Vgg16	73.5	72.5	73.5	73.2
ResNet50	75	75	75	75
ResNet101	76	76.5	76	76.2

this end, we couple a ResNet101-based scheme within a transfer learning framework and compare our approach with widely known architectures, such as GoogleNet-2014, Vgg16, and ResNet50. We assess all involved models into two datasets: a subset of samples extracted from the CIFAR10 database (cat vs. dog classes) and a database composed of cow and no cow images (IMGCOW). The results show that our transfer learning approach can fulfill the following aspects: i) achieve reasonable classification accuracy from small datasets, ii) properly fit the required hyperparameters without overfitting, and iii) deal with the low resolution and noisy sample issues. Indeed, our strategy yields competitive performance against state-of-the-art methods concerning the number of input samples used. Furthermore, in the subset formed from CIFAR10, we employ a low number of instances concerning the compared architectures without transfer learning. Then, residual units favor transferring the knowledge between similar tasks, e.g., ResNet models. Of note, the experimental codes were developed using TensorFlow 2, and the codes for method comparison are publicly available.

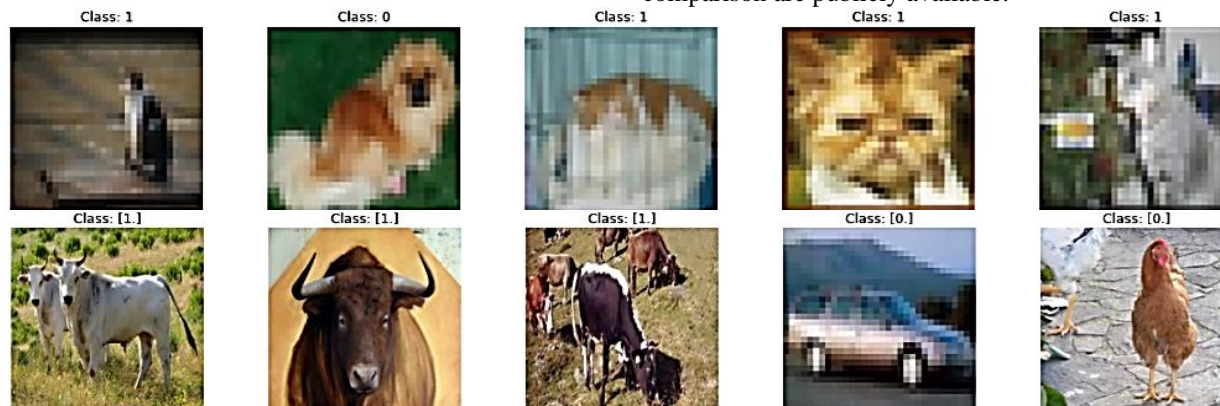


Figure 2. Studied databases. First row: CIFAR10 subset (dog vs cat). Second row: IMGCOW (cow vs no cow).

TABLE IV
IMGCOW DATASET RESULTS (COW VS. NO COW) USING TRANSFER LEARNING (TL). RESULTS ARE DISPLAYED IN [%].

Architecture+TL	Precision	Recall	F1	Accuracy
GoogleNet-2014	78	80.5	77.5	78.3
Vgg16	97	96	97	97
ResNet50	98	97	97.5	98
ResNet101	98	98	98.5	98.3

TABLE V
METHOD COMPARISON RESULTS FOR CIFAR10 DATA SUBSET (CAT VS. DOG). RESULTS ARE DISPLAYED IN [%]. TL: TRANSFER LEARNING. -: NOT PROVIDED.

Architecture	Precision		Acc _c		Acc
	Dog	Cat	Dog	Cat	
GoogleNet-2014 [20]	-	-	57.5	78.8	68.1
ResNet50 [20]	-	-	82.1	61.9	72
ResNet50 [21]	84	68	76.8	77.7	77.2
Proposed TL (ResNet101)	76	76	76	77.0	76.2

VI. CONCLUSIONS AND FUTURE WORK

In this study, we introduce an image-based animal recognition approach based on a transfer learning strategy. To

As future work, we plan to test our approach to smart farming environments to support a real-time vision system for cow counting. Also, medical imaging tasks from few samples could be tested. In turn, trying different loss functions and architectures is a research field of interest.

ACKNOWLEDGMENTS

A. Álvarez-Meza thanks to the project “Sistema de visión por computador para el monitoreo automático de variables de productividad en plantas industriales” (HERMES 46185 - FIA - Universidad Nacional de Colombia - Manizales).

REFERENCES

- [1] Qun Liu and S. Mukhopadhyay, “Unsupervised Learning using Pretrained CNN and Associative Memory Bank.” 2018. doi: 10.1109/IJCNN.2018.8489408
- [2] Cai, Z., & Vasconcelos, N. (2019). Cascade R-CNN: high quality object detection and instance segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence. doi: 10.1109/TPAMI.2019.2956516
- [3] Liu, C., Liu, P., Zhao, W., & Tang, X. (2019). Visual Tracking by Structurally Optimizing Pre-trained CNN. IEEE Transactions on Circuits and Systems for Video Technology. doi: 10.1109/TCSVT.2019.2938038

- [4] Yao, G., Lei, T., & Zhong, J. (2019). A review of Convolutional-Neural-Network-based action recognition. *Pattern Recognition Letters*, 118, 14-22. doi: 10.1016/j.patrec.2018.05.018
- [5] Su, J., Yi, D., Su, B., Mi, Z., Liu, C., Hu, X., ... & Chen, W. H. (2020). Aerial Visual Perception in Smart Farming: Field Study of Wheat Yellow Rust Monitoring. *IEEE Transactions on Industrial Informatics*. doi: 10.1109/TII.2020.2979237
- [6] Mukherjee, A., Misra, S., Sukrutha, A., & Raghuvanshi, N. S. (2020). Distributed aerial processing for IoT-based edge UAV swarms in smart farming. *Computer Networks*, 167, 107038. doi: 10.1016/j.comnet.2019.107038
- [7] Bullock, J., Cuesta-Lázaro, C., & Quera-Bofarull, A. (2019, March). XNet: A convolutional neural network (CNN) implementation for medical X-Ray image segmentation suitable for small datasets. In *Medical Imaging 2019: Biomedical Applications in Molecular, Structural, and Functional Imaging* (Vol. 10953, p. 109531Z). International Society for Optics and Photonics. doi: 10.1117/12.2512451
- [8] Zhou, T., Ruan, S., & Canu, S. (2019). A review: Deep learning for medical image segmentation using multi-modality fusion. *Array*, 3, 100004. doi: 10.1016/j.array.2019.100004
- [9] Xu, Q., Zhang, M., Gu, Z., & Pan, G. (2019). Overfitting remedy by sparsifying regularization on fully-connected layers of CNNs. *Neurocomputing*, 328, 69-74. doi: 10.1016/j.neucom.2018.03.080
- [10] Webster, R., Rabin, J., Simon, L., & Jurie, F. (2019). Detecting overfitting of deep generative networks via latent recovery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 11273-11282).
- [11] Lu, N., Zhang, T., Niu, G., & Sugiyama, M. (2020, June). Mitigating overfitting in supervised classification from two unlabeled datasets: A consistent risk correction approach. In *International Conference on Artificial Intelligence and Statistics* (pp. 1115-1125).
- [12] Li, Z., Kamnitsas, K., & Glocker, B. (2019, October). Overfitting of neural nets under class imbalance: Analysis and improvements for segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* (pp. 402-410). Springer, Cham. doi: 10.1007/978-3-030-32248-9_98
- [13] Bejani, M. M., & Ghatee, M. (2020). Theory of adaptive SVD regularization for deep neural networks. *Neural Networks*. doi: 10.1016/j.neunet.2020.04.021
- [14] Ranjit, M. P., Ganapathy, G., Sridhar, K., & Arumugham, V. (2019, July). Efficient deep learning hyperparameter tuning using cloud infrastructure: intelligent distributed hyperparameter tuning with Bayesian optimization in the cloud. In *2019 IEEE 12th International Conference on Cloud Computing (CLOUD)* (pp. 520-522). IEEE. doi: 10.1109/CLOUD.2019.00097
- [15] Choi, D., Shallue, C. J., Nado, Z., Lee, J., Maddison, C. J., & Dahl, G. E. (2019). On empirical comparisons of optimizers for deep learning. *arXiv preprint arXiv:1910.05446*.
- [16] Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., ... & He, Q. (2020). A comprehensive survey on transfer learning. *Proceedings of the IEEE*. doi: 10.1109/JPROC.2020.3004555
- [17] Zheng, H., Wang, R., Yang, Y., Yin, J., Li, Y., Li, Y., & Xu, M. (2019). Cross-domain fault diagnosis using knowledge transfer strategy: A review. *IEEE Access*, 7, 129260-129290. doi: 10.1109/ACCESS.2019.2939876
- [18] Khan, S., Islam, N., Jan, Z., Din, I. U., & Rodrigues, J. J. C. (2019). A novel deep learning-based framework for the detection and classification of breast cancer using transfer learning. *Pattern Recognition Letters*, 125, 1-6. doi: 10.1016/j.patrec.2019.03.022
- [19] Sun, Q., Liu, Y., Chua, T. S., & Schiele, B. (2019). Meta-transfer learning for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 403-412).
- [20] Sharma, N., Jain, V., & Mishra, A. (2018). An analysis of convolutional neural networks for image classification. *Procedia computer science*, 132, 377-384. 10.1016/j.procs.2018.05.198
- [21] Liu, Q., & Mukhopadhyay, S. (2018, July). Unsupervised learning using pretrained CNN and associative memory bank. In *2018 International Joint Conference on Neural Networks (IJCNN)* (pp. 01-08). IEEE. 10.1109/IJCNN.2018.8489408
- [22] Géron, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. O'Reilly Media.
- [23] Heaton, J. (2018). *Ian goodfellow, yoshua bengio, and aaron courville: Deep learning*. doi: 10.1007/s10710-017-9314-z
- [24] Trnovszký, T., Kamencay, P., Orješek, R., Benčo, M., & Sýkora, P. (2017). Animal recognition system based on convolutional neural network. doi: 10.15598/aece.v15i3.2202
- [25] Nguyen, H., Maclagan, S. J., Nguyen, T. D., Nguyen, T., Flemons, P., Andrews, K., ... & Phung, D. (2017, October). Animal recognition and identification with deep convolutional neural networks for automated wildlife monitoring. In *2017 IEEE international conference on data science and advanced Analytics (DSAA)* (pp. 40-49). IEEE. doi: 10.1109/DSAA.2017.31



Gloria Stephany Gómez Gómez

received his undergraduate degree in electronic engineering (2020) from the Universidad Nacional de Colombia. Research interests: deep learning.

ORCID: <https://orcid.org/0000-0001-5895-0496>



Diego Fabian Collazos Huertas

received his undergraduate degree in electronic engineering (2014), and his M.Sc. (2016) from the Universidad Nacional de Colombia. Currently, he is a Ph.D. (c) at the same university. Research interests: deep learning and signal processing.

ORCID: <https://orcid.org/0000-0002-0434-3444>



Andrés Marino Álvarez Meza

received his undergraduate degree in electronic engineering (2009), his M.Sc. (2011), and his Ph.D. in automatics from the Universidad Nacional de Colombia. Currently, he is a Professor in the Department of Electrical, Electronic and Computation Engineering at the same university. Research interests: machine learning and signal processing.

ORCID: <https://orcid.org/0000-0003-0308-9576>