

Automatic Characterisation of Dye Decolorisation in Fungal Strains using Expert, Traditional, and Deep Features

Marina Arredondo-Santoyo · César Domínguez · Jónathan Heras* · Eloy Mata · Vico Pascual · M^a Soledad Vázquez-Garcidueñas · Gerardo Vázquez-Marrufo*

Received: date / Accepted: date

Abstract Fungi have diverse biotechnological applications in, among others, agriculture, bioenergy generation, or remediation of polluted soil and water. In this context, culture media based on color change in response to degradation of dyes are particularly relevant; but measuring dye decolorisation of fungal strains mainly relies on a visual and semiquantitative classification of color intensity changes. Such a classification is a subjective, time-consuming and difficult to reproduce process. In order to deal with these problems, we have performed a systematic evaluation of different image-classification approaches considering ad-hoc expert features, traditional computer vision features, and transfer-learning features obtained from deep neural networks. Our results favor the transfer learning approach reaching an accuracy of 96.5% in the evaluated dataset. In this paper, we provide the first, at least up to the best of our knowledge, method to automatically characterise dye decolorisation level of fungal strains from images of inoculated plates.

Marina Arredondo-Santoyo and Gerardo Vázquez-Marrufo
Multidisciplinary Center of Biotechnology Studies (CMEB)
Faculty of Veterinary Medicine, Universidad Michoacana de San Nicolás de Hidalgo, Mexico.

César Domínguez, Jónathan Heras, Eloy Mata, Vico Pascual
Department of Mathematics and Computer Science,
University of La Rioja, Logroño, Spain.

M^a Soledad Vázquez-Garcidueñas
Division of Postgraduate Studies,
Faculty of Medical and Biological Sciences “Dr. Ignacio Chávez”,
Universidad Michoacana de San Nicolás de Hidalgo, Mexico.

*Corresponding authors: (J. Heras) jonathan.heras@unirioja.es, and (G. Vázquez-Marrufo) gvazquezmarrufo@yahoo.com.mx.

Keywords Fungal decolorisation · Image classification · Computer vision · Deep learning · Transfer learning

1 Introduction

Fungi are important sources of metabolites and enzymes which have diverse biotechnological applications in agriculture; the food, paper, and textile industries; the synthesis of organic compounds and metabolites with pharmaceutical activities; cosmetic production; bioenergy generation; and remediation of polluted soil and water [26,10]. Because of the considerable diversity of fungal species, that are distributed in all ecosystems of the planet and occupy diverse niches as biotrophs or saprophytes [53], the isolation and characterisation of new strains with potential for biotechnological applications remains to be a dynamic field of mycological research [76,49,39].

Despite the revolution of fungal biotechnology that happened during the past two decades due to development of omic sciences [3,22] and massive data analysis [37,64]; isolation of fungal strains with biotechnological relevance, their identification, and their morphological and physiological characterisation continue to be relevant, for which selective media are routinely used for strain isolation and for detection of their extracellular metabolites or enzymes [39].

In that context, culture media based on color change, in response to degradation of dyes or evidencing production of extracellular hydrolytic or oxidative enzymes, are particularly relevant. Most color-change assays rely on a visual and semiquantitative classification of color intensity changes, using an arbitrary scale for making comparative analyses between the differ-

ent assayed fungal strains [4,9,76]. This approach implies that the results from assays are subjective, time-consuming, and unreproducible within the same laboratory and also across laboratories, even when assays are made under the same experimental conditions. Therefore, automatic and reliable tools for the selection and characterisation of fungal strains are needed for avoiding the dependence on the experimenter's interpretation that is commonly present when assessing fungal capacity for dye decolorisation [57,4,62], for degradation of xenobiotics [45,42] or for reduction of Fe [51], as well as by assays aimed at production of oxidative lignolytic enzymes [56,25,79] and a large variety of hydrolytic enzymes — such as cellulases [58,36,40], xylanases [77], proteases [2], chitinases [69], amylases [2,69], and lipases [2], among other examples.

and

Up to the best of our knowledge, there is not any method in the literature to automatically characterise the dye decolorisation level of fungal strains. In this work, we tackle this problem by analysing fungal-strain images using computer vision techniques. In particular, the characterisation of the dye decolorisation level of fungal-strain images can be seen as an image-classification problem. In this kind of problem, it is important to make two important decisions: how do we describe the images? and what machine-learning algorithm is employed to construct the classification model? To answer the first question, we analyse three different approaches to describe the images: (1) using the information provided by the expert biologists, (2) using traditional computer vision descriptors, and (3) using transfer learning — a successful deep learning technique. In order to answer the latter question, we perform a systematic statistical analysis to compare the most widely employed classification algorithms.

The contribution of this paper is threefold. First of all, we provide the first annotated dataset of images to characterise dye decolorisation of fungal strains. In addition, we perform a thorough comparison of different methods to automatise such a task. Finally, we provide a model to classify dye decolorisation of fungal strains with an outstanding accuracy of 96.5%.

2 Materials and methods

2.1 Fungal strains

Strains obtained from the Michoacan University Culture Collection (Cepario Michoacano Universitario, CMU) of several species of basidiomycetes (unpublished data) and of *Trichoderma* spp. [23] were used for de-

colorisation assays. Assays of extracellular enzymatic activity were made only in strains of *Trichoderma* spp.

2.2 Solid media used

Most strains were maintained in potato dextrose agar (PDA, DifcoTM, USA), also used for production of inocula. Decolorisation of phenolic substances assays were carried out both in PDA and in malt extract agar (MEA, DifcoTM, USA). Both media were prepared as indicated by suppliers.

Isolates of *Trichoderma* spp. were cultured and maintained in Vogel's medium N (VMN), and inocula were generated in the same medium. VMN 50X stock solution was prepared with: 150 g/L $Na_3C_6H_5O_7 \cdot 5H_2O$, 250 g/L KH_2PO_4 , 100 g/L NH_4NO_3 , 10 g/L $MgSO_4 \cdot 7H_2O$, 5 g/L $CaCl_2 \cdot 2H_2O$, 5 ml of biotin solution (5 mg biotin in 100 mL 50% ethanol), and 5 ml of trace element solution (5 g/L citric acid $\cdot 2H_2O$, 5 g/L $ZnSO_4 \cdot H_2O$, 1 g/L $FeCl_3 \cdot 6H_2O$, 0.25 g/L $CuSO_4 \cdot 5H_2O$, 0.072 g/L $MnCl_2 \cdot 4H_2O$, 0.05 g/L H_3BO_3 , 0.05 g/L $Na_2MoO_4 \cdot 2H_2O$). For its use, the VMN stock solution was diluted in distilled water to a final concentration of 1X and 1.5% glucose and 15g/L of bacteriological agar were added.

Detection of xylanases was made in xylanolysis basal medium (XBM) prepared with: 1 g/L $C_4H_{12}N_2O_6$, 1 g/L KH_2PO_4 , 0.1 g/L yeast extract, 0.5 g/L $MgSO_4 \cdot 7H_2O$, 0.001 g/L $CaCl_2 \cdot 2H_2O$, 0.1 g/L peptone, and 1.6% (w/v) bacteriological agar supplemented with 0.4% glucose. For detection of xylanases 4% (W/V) xylan (Sigma, USA) and 1.6% agar were added to the XBM. Protease activity was detected in 0.4% gelatin from porcine skin (Sigma, USA) and 1.6% agar at pH 6.

All media were sterilized by autoclaving at 120 °C and 15 pounds/in².

2.3 Fungal inoculum generation

All tested strains were stored in PDA at 4 °C until used. Inocula were generated using 6 mm cylindrical plugs removed with a cork borer from the margin of mycelia colony in log growth phase and inoculated in the center of a 90 mm PDA Petri dish. The dishes were incubated in darkness at 28 °C until mid-log phase, and then 6 mm of inoculum was taken from the margin of the colony as described. These mycelial plugs were used for assays.

The inocula for xylanases assays were prepared from 6 mm cylindrical plugs taken as described above from several *Trichoderma* spp. isolates cultivated in XBM,

and for protease assays, from several isolates of *Trichoderma* spp. cultivated in VMN.

2.4 Decolorisation and enzymatic assays

Decolorisation assays were carried out in 90 mm Petri dishes with PDA or MEA in independent experiments supplemented with the phenolic compound guaiacol and the dyes Methyl Orange (MO), Direct Blue 71 (DB71), Acid Blue 1 (AB1), Chicago Sky Blue 6B (CSB-6B) from the azo dye group; Carmine Indigo (CI) from the indigoid group, Remazol Brilliant Blue-R (RBBR) from the anthraquinone group; and the triphenylmethane type dye Acid Fuchsin (AF). The guaiacol substrate was used at a final concentration of 0.01% (w/v) [41] and the dyes at a final concentration of 250 $\mu\text{g}/\text{mL}$ [50]. All chemicals were purchased from Sigma-Aldrich (USA). The media with phenolic compounds were inoculated at the center of the plate with a 6 mm mycelium plug obtained as previously described and incubated in darkness at 28 °C. A control plate with media, but without fungal inoculum, was incubated to confirm that the color change was not induced by physicochemical factors during fungal growth. The decolorisation of phenolic dyes was determined visually by the fading and loss of color of the media, while guaiacol oxidation was registered as the formation of a reddish-brown halo in the media [50,41]. All experiments were performed in triplicate. Detection of xylanase activity was made following the method of Pointing [56] by development the plates by flooding them with lugol solution (I2/KI). Proteolytic activity was detected by the method of Hankins and Anagnostakis [30], by precipitation of undegraded proteins by ammonium sulfate.

2.5 Image acquisition

Color change results were documented photographically from Petri plates in a transilluminator with white light from an 8 W lamp. Photographs were taken with a Sony Full HD 1080 camera with a Zeiss lens and a resolution of 12 Megapixels, but any camera with the same or higher resolution can be used. Photographs were taken without flash to avoid light reflection. The Petri plates were preferentially photographed by the bottom surface, but in some cases top surface provided a better resolution for color analysis. In this last case, the lid was removed before capturing the image to prevent light reflection and easing the focusing of the surface. In any case, it is recommendable to photograph both the top and bottom surfaces in order to select the image that better records color changes and colony diameter.

3 Protocol

The protocol employed in this work to construct an image-classifier for dye decolorisation of fungal assays is summarised in Figure 1 — such a protocol is commonly used in the context of image classification [59]. The rest of this section is devoted to explain each step of the protocol.

3.1 Dataset

Following the procedure presented in the previous section, a total of 235 images of dye decolorisation assays were acquired. The images of the dataset were annotated by the biological experts with one of the following four labels indicating the decolorisation level: “-” (no decolorisation), “+”, “++”, and “+++” (completely decolorised). The initial dataset contained 123 “-” images, 47 “+” images, 27 “++” images, and 38 “+++” images. Examples of the images of the dataset are provided in Figure 2.

The dataset of images was clearly imbalanced; and, it is a well-known result that classification algorithms might be negatively impacted by imbalanced datasets [12,7]. To deal with this problem, we employed the synthetic minority oversampling technique (SMOTE) [11]; a data augmentation [68] method that not only deals with the imbalanced problem but also improves the generalisation of the models. Namely, we augmented the dataset by applying vertical and horizontal flips to the image, and random rotations and translations — the number of applied transformations for each class depended on the number of initial images for that class. The final dataset consists of 1204 images: 306 “-” images, 313 “+” images, 297 “++” images, and 288 “+++” images. Augmenting the dataset of images not only deals with the problem of imbalanced data, but also handles the overfitting phenomenon [59]. The final dataset of images is freely available at <https://github.com/joheras/DecolorisationImages>.

3.2 Feature extraction methods

Once the dataset of images is created, the next step in the protocol consists in determining which features will be used to describe the images of the dataset. Different approaches can be employed to generate the feature vector associated with an image. In this study, we consider three different alternatives.

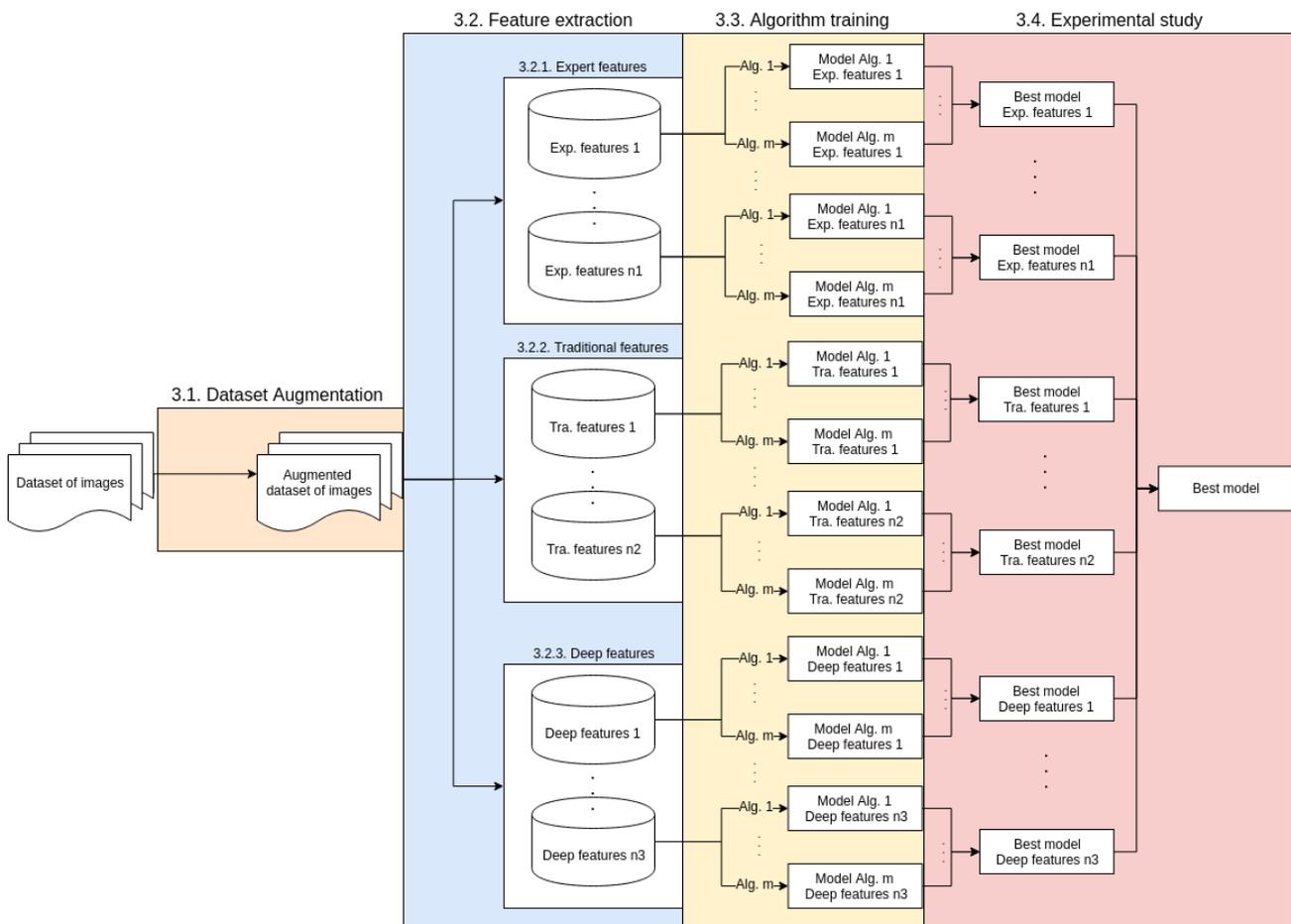


Fig. 1 Protocol employed in this work to select the best model for classifying dye decolorisation images.

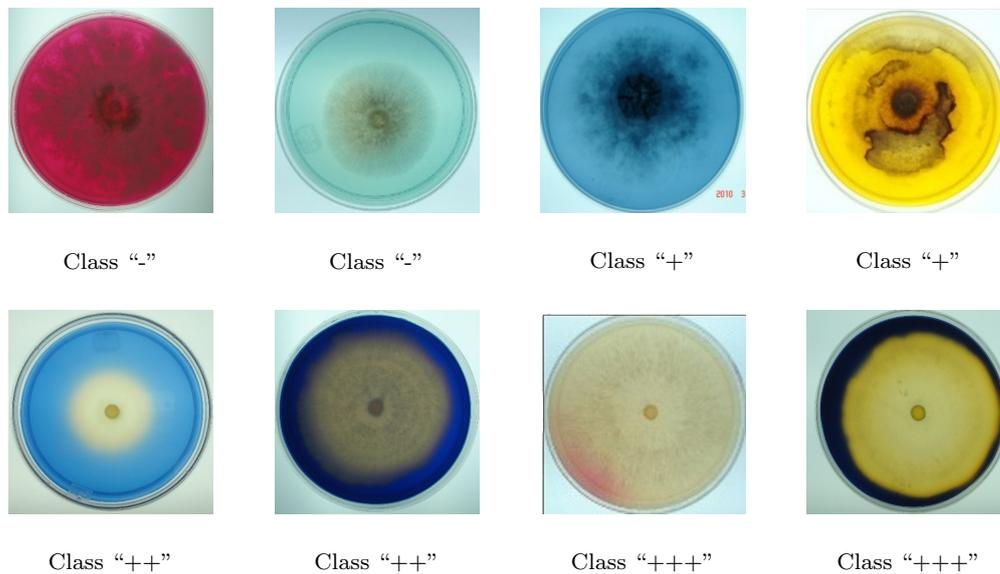


Fig. 2 Examples of the dataset of images and their classes.

3.2.1 Extracted features from experts' knowledge

The first approach that we study to create the feature vector associated with a dye decolorisation image tries to capture the experts' knowledge. The intuitive idea behind the proposed algorithm consists in measuring the "distance" from different regions of the image to an image that has been completely decolorised and to a control image that only contains the dye. Given an image I , the control image C , and the decolorised image D , the procedure to generate a feature vector from I is described as follows:

1. Convert the images I , C and D to the colour space L^*a^*b and rescale them to the same size. The L^*a^*b colour space mimics the methodology in which humans see and interpret colour, and in this colour space the Euclidean distance has an actual perceptual meaning [35].
2. Generate the images C_0, C_1, \dots, C_{100} such that C_i is the linear blending [75] of $(1-i/100)C + (i/100)D$. This process generates a decolorisation scale for each dye ranging from no decolorisation ($i = 0$) to completely decolorised ($i = 100$).
3. From $j = 1$ to 10, extract the annulus A_j of image I with centre the centre of the image I , radius of the inner circle $w \times (j - 1)/10$ (where w is the width of the image), and radius of the outer circle $w \times j/10$.
4. For each annulus A_j , split it into 2^p sectors (with $p \geq 2$) of the same size $A_j^1, \dots, A_j^{2^p}$.
5. For each A_j^k , with $1 \leq j \leq 10$ and $1 \leq k \leq 2^p$, obtain the index i ($0 \leq i \leq 100$) such that the distance $d(A_j^k, C_i)$ is minimised. The distance $d(E, F)$ between two images E and F is defined as the Euclidean distance between the normalised histograms of E and F .
6. Return the indexes obtained in the previous step as feature vector.

The above algorithm not only takes as input the images I, C and D but also the exponent of the number of sectors p (see Step 4) and the number of bins that are employed to generate the histograms of Step 5. In this work, we will consider three common values for the numbers of bins that are 4 bins per channel, 8 bins per channel and 16 bins per channel (we will call each variant of the algorithm A444, A888, and A161616 respectively); and p will take values from 2 to 6. In addition, we will consider a variant of these algorithms where the feature vectors are normalised, since this can have a positive impact in some classification algorithms [44].

3.2.2 Traditional computer vision features

In the second approach to describe dye decolorisation images, we use two traditional computer vision feature descriptors that characterise the texture of an image: *Haralick* [31] and *Histograms of local gradients* (HOG) [24].

The Haralick features are derived from the grey level co-occurrence matrix. This matrix records how many times two grey-level pixels adjacent to each other appear in an image. Then, based on this matrix, 13 values are extracted from the co-occurrence matrix to quantify the texture.

The HOG features are computed over a grid of overlapping rectangular blocks in the image. For each block, its histogram describes the frequency of the occurring gradient directions inside that block.

Both for the Haralick and the HOG features, we consider two approaches to generate the feature vector that describes an image. In the former, we extract either the Haralick or the HOG features from the image, and that is its feature vector. In the latter, we stack the image with a control image of the dye (see Figure 3); and, subsequently, either the Haralick or the HOG features are computed from the stacked image, and used as feature vector of the original image.

3.2.3 Deep learning features

Finally, the last approach that is considered in this work to generate the feature vectors employs deep learning. Deep Neural Networks (DNNs) [43] are the state-of-the-art technique for image classification; but its use may be unfeasible in many situations since they require very large training sets (from thousands up to several million images) [59]. To overcome this difficulty, the common approach in the literature consists in applying *transfer learning* [59]. This technique consists in partially re-using a model trained in a source task in a new target task [52, 15, 29, 48].

As explained in [59], transfer learning can be used in different ways; the approach that we follow in this work consists in using the output of the source trained network as "off-the-shelf" features that are employed to train a complete new classifier for the target task [52]. In order to apply transfer learning in our context, we consider 8 publicly available trained DNNs: DenseNet [34], GoogleNet [72], Inception v3 [73], OverFeat [65], Resnet 50 [32], VGG16 [70], VGG19 [70], and Xception v1 [13]. All these networks were initially trained for the image classification task of the ImageNet ILSVRC challenge [63], a dataset of 1.2 million images which are hand labelled with the presence/absence of 1000 cate-

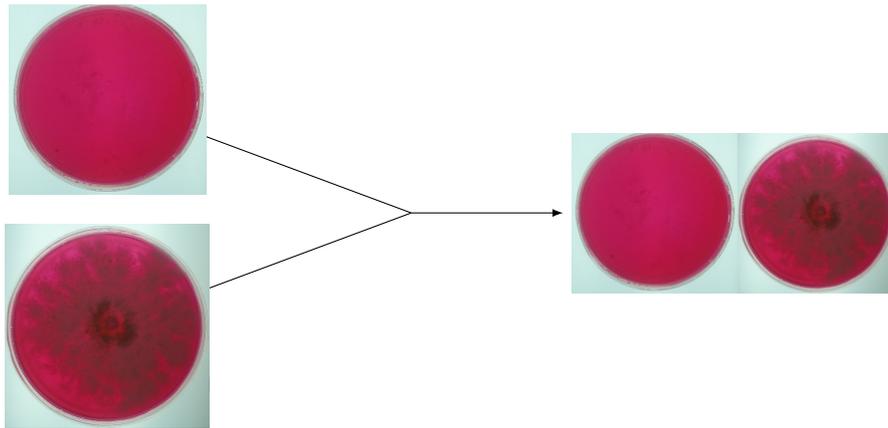


Fig. 3 Stack of an image with a control image of the dye. *Top-Left.* Control image. *Bottom-Left.* Analysed image. *Right.* Stack of images.

gories. We strip the last layer of these networks, and use the output produced by their last layers as feature vectors — information about the networks can be found in Table 1.

Using each one of the aforementioned networks, and analogously to the case of the traditional features, we consider two approaches to generate the feature vector that describes an image. In the former, we input the image to the network and use the output produced by the last layer of the network as the feature vector. In the latter, we stack the image with a control image of the dye, then, the resulting image is inputted to the network, and the output produced by the last layer of the network is used as feature vector.

3.3 Classification algorithms

From the dataset of images, the feature vectors obtained using one of the previously mentioned approaches are fed to a classifier that is trained with them. The 6 classifiers that are considered in this work are Extremely Randomised Trees (from now on ERT) [28], KNN [21], Logistic Regression (from now on LR) [47], Multilayer Perceptron (from now on MLP) [6], Random Forest (from now on RF) [8], and SVM [20]. The classification models produced by each combination of descriptor and classification algorithm are systematically evaluated by means of a statistical study.

3.4 Experimental study

In order to validate the classification models obtained with the protocol previously explained, a stratified 10-fold cross-validation approach was employed. To evaluate the performance of the classifiers, we measured

their accuracy (i.e. the proportion of samples for which the model produces the correct output), the results are taken as the mean and standard deviation of the accuracy for the 10 test sets. The hyper parameters of each classification algorithm were chosen using a 10-fold nested validation with each of the training sets, and using a randomised search on the parameters distributions.

In order to determine whether the results obtained were statistically significant, several null hypothesis tests were performed using the methodology presented in [27,67]. In order to choose between a parametric or a non-parametric test to compare the models, we check three conditions: independence, normality and heteroscedasticity — the use of a parametric test is only appropriate when the three conditions are satisfied [27].

The independence condition is fulfilled because we perform different runs following a stratified 10-fold cross-validation approach for separating the data with a prior random reshuffling of the samples. A stratified cross-validation approach splits the dataset into ten random equal-size subsets preserving the percentage of samples for each class. Nine of such subsets are chosen ten times to train the model and the remaining set is used to test them — in each iteration, one of the subsets is chosen to be the test set. We use the Shapiro-Wilk test [66] to check normality — with the null hypothesis being that the data follow a normal distribution — and, a Levene test [46] to check heteroscedasticity — with the null hypothesis being that the results are heteroscedastic.

When comparing two models, we use Student’s t-test [27] when the parametric conditions are satisfied, and Wilcoxon’s test [27] otherwise. In both cases, the null hypothesis is that the two models have the same performance.

Network	Input image size	Number of layers	Size of output feature vector
DenseNet 40 12	32×32	40	12928
GoogleNet	231×231	22	1000
Inception v3	299×299	48	131072
OverFeat	231×231	7	3072
Resnet 50	224×224	50	2048
VGG16	224×224	16	25088
VGG19	224×224	19	25088
Xception v1	299×299	36	204800

Table 1 Information about the networks employed in this study.

If we compare more than two models, we employ an ANOVA test [67] when the parametric conditions are fulfilled, and a Friedman test [67] otherwise. In both cases, the null hypothesis is that all the models have the same performance. Once the test for checking whether a model is statistically better than the others has been conducted, a post-hoc procedure is employed to address the multiple hypothesis testing among the different models. A Holm post-hoc procedure [33], in the non-parametric case, or a Bonferroni-Dunn post-hoc procedure [67], in the parametric case, has to be used for detecting significance of the multiple comparisons [27,67] and the p values should be corrected and adjusted. We perform our experimental analysis with a level of confidence equal to 0.05.

In addition, the size effect is measured using Cohen’s d [18] and Eta Squared [19].

3.5 Software and hardware

The code employed in this work was implemented in Python using several libraries: OpenCV [38] (library for image processing and computer vision), the Keras framework [14] with a Tensorflow back-end [1] (provides the DNNs), the sklearn library [54] (library for machine learning), mahotas [17] (library for computer vision), and STAC [61] (library for statistical analysis). All the source code for this paper is freely available at <https://github.com/joheras/ObjectClassificationByTransferLearning>.

All experiments were performed under Linux OS on a machine with CPU Intel Core i7-4790 3.60GHz, GPU NVIDIA Quadro K1100M.

4 Results

4.1 Expert approach

As we have explained in Section 3.2.1, the algorithm to extract features from experts’ knowledge is parametrised by the exponent p of the number of sectors 2^p , and the number of bins. Since such a parametri-

sation produces several combinations, here we only include the results considering p with value 2 (i.e. we take the 4 quadrants of the annulus) and considering 4, 8 and 16 bins per channel — those combinations produce the best results. The results for the other combinations are available at the project webpage.

The results are presented in Tables 2 and 3, and Figure 4. From the results presented in those tables and figure, we can notice that decreasing the number of bins have a positive impact in all the classifiers. When we compare the different classifiers, we obtain that the ERT classifier produces the models with the best accuracy independently of the number of bins that are employed and whether normalisation is used. Besides, and although there are not significant differences with respect to the second best classifier (RF in all the cases but when 4 bins are employed and the features are further normalised, in that case the second best classifier is KNN), the Cohen’s d size effect of the differences between the first and the second best classifiers are large in all the cases (ranging from 0.69 to 1.70) except in the case of A888 without normalisation wherein the size effect is small (0.15). **We can observe that all the classifiers built using expert features obtain an accuracy over 50% but just a few of them get an accuracy over 90%.**

Now, we compare the best models for each number of bins with and without normalisation of the feature vectors. We first consider the case where the features are not normalised; in such a case, the models to compare are A444-ERT, A888-ERT and A161616-ERT. Although, there are not found significant differences in the accuracy of the models (ANOVA $F = -0.244$; $p > 0.05$), we obtain an eta squared = 0.75 with large size effect. If we consider the model with the best mean (A444-ERT), and take into account the size effect with respect to the other two models, we obtain that in both cases the size effect is large (0.70 and 2.52, respectively). Secondly, when the feature vectors are not normalised, the models to compare are A444-ERT, A888-ERT and A161616-ERT. Again, although there are not found significant differences in the accuracy of the models (ANOVA $F = -0.234$; $p > 0.05$), we obtain an eta squared = 0.45 with large size effect. If we consider the

Bins	ERT	KNN	LR	MLP	RF	SVM	Test (Anova or Friedman)	After post-hoc procedure
A161616	<i>83.1(2.6)</i>	62.8(3.7)	50.7(3.7)	61.4(3.1)	80.6(3.2)	69.9(3.7)	203.8***	ERT \simeq RF, SVM; ERT > KNN, MLP, LR
A888	<i>91.4(2.1)</i>	77.1(5.4)	58.6(4.4)	73.6(3.7)	91.0(2.7)	83.5(4.1)	132.2***	ERT \simeq RF, SVM; ERT > KNN, MLP, LR
A444	93.2(2.7)	83.1(3.3)	63.2(5.2)	76.2(5.7)	91.3(2.5)	86.9(3.0)	563.7***	ERT \simeq RF; ERT > SVM, KNN, MLP, LR

Table 2 Mean (and standard deviation) for the different studied models without normalising the feature vectors. The best result for each feature extraction method is in *italics*, and the best result in **bold face**. *** $p < 0.001$; >: there are significant differences; \simeq : there are not significant differences.

model with the best mean (A444-ERT), and take into account the size effect with respect to the other two models, we obtain that in the size effects are medium-large and large (0.60 and 2.50, respectively).

4.2 Traditional computer vision features

The results obtained for the traditional computer vision features are presented in Tables 4 and 5, and Figure 5. In general, the use of a control image seems to outperform the case when that image is not employed. At first sight, there is not a clear feature extractor or classifier with better accuracy than the rest. In the case of models where the control image is taken into account, the classifiers constructed using the HOG features outperform those constructed using Haralick features; however, when the control image is not considered, several classifiers constructed using Haralick features have a better accuracy than those built using HOG features. Moreover, the ERT classifier is the best method in three out of the four cases; and, it is only outperformed by SVM and LR working with HOG features and without using a control image.

Now, we compare the best models for each traditional computer vision features with the different classifiers. We first consider the case where the features are extracted without considering the control image. In such a case, the models to compare are Haralick-ERT and HOG-MLP. The Wilcoxon’s test is employed to compare these 2 models since the normality condition is not satisfied ($W = 0.89, p = 0.032$). We obtain that the Haralick-ERT method is significantly better than HOG-MLP ($t=0.000$; $p=0.005$) with a large size effect (Cohen’s $d = 4.83$). When the features are extracted using not only the image, but also the control image; the models to compare are Haralick-ERT and HOG-ERT. We obtain that the HOG-ERT method is significantly better than Haralick-ERT (Student’s $t = -2.71$, $p = 0.014$ with a large size effect (Cohen’s $d = 1.21$).

4.3 Deep learning features

The results obtained for each combination of DNN, classifier, and approach to generate feature vectors are included in Tables 6 and 7, and Figure 6. As it can be seen, there is not a classifier that always produces the best results for all the cases — this is a case of the no free lunch theorem for machine learning [78]. Although, the use of a control image seems to have a positive impact; that is not true in all the cases (see, for instance, OverFeat-ERT or OverFeat-KNN). In addition, no network is able to obtain an accuracy higher than 90% with all the classifiers; as exception, we can cite Resnet 50 using the control image that almost reaches that milestone. Moreover, the LR classifier seems to have a good and stable performance independently of the network, obtaining an accuracy higher than 90% with all the networks when the control image is used. **However, we can notice that, on the contrary to the use of expert and traditional features, using deep features we can create several models with an accuracy over 90% (compare Figures 4, 5 and 6).** Finally, some classifiers seem to have very different behaviour depending on the network used. For instance, the accuracy of the SVM classifier when the image control is taken into account ranges from 33% (using features from the VGG16 network) to 96% (using the features extracted from the Resnet 50 network).

Now, we compare the best models that can be obtained for each DNN with the different classifiers. We first consider the case where the features are extracted without considering the control image; in such a case, the models to compare are DenseNet-ERT, GoogleNet-ERT, Inception-LR, OverFeat-SVM, Resnet-LR, VGG16-LR, VGG19-LR, and Xception-LR. Although, there are not found significant differences in the accuracy between the models (ANOVA $F = 0.05$; $p = 0.99$), a large size effect (eta squared = 0.18) is obtained. If we consider the model with the best mean (Resnet-LR), and take into account the size ef-

Bins	ERT	KNN	LR	MLP	RF	SVM	Test (Anova or Friedman)	After post-hoc procedure
A161616	<i>75.6(3.5)</i>	61.8(2.8)	50.6(2.8)	63.3(4.2)	67.7(5.1)	65.5(4.0)	0.38	ERT \simeq RF, SVM, MLP, KNN, LR
A888	<i>81.7(5.1)</i>	77.4(4.4)	58.1(3.9)	72.0(3.6)	77.8(5.5)	72.0(3.6)	0.48	ERT \simeq RF, KNN, MLP, SVM, LR
A444	<i>84.4(3.1)</i>	80.1(2.5)	63.0(4.9)	76.1(3.8)	77.2(5.0)	73.9(5.2)	27.6***	ERT \simeq KNN; ERT > RF, MLP, SVM, LR

Table 3 Mean (and standard deviation) for the different studied models normalising the feature vectors. The best result for each feature extraction method is in *italics*, and the best result in **bold face**. *** $p < 0.001$; >: there are significant differences; \simeq : there are not significant differences.

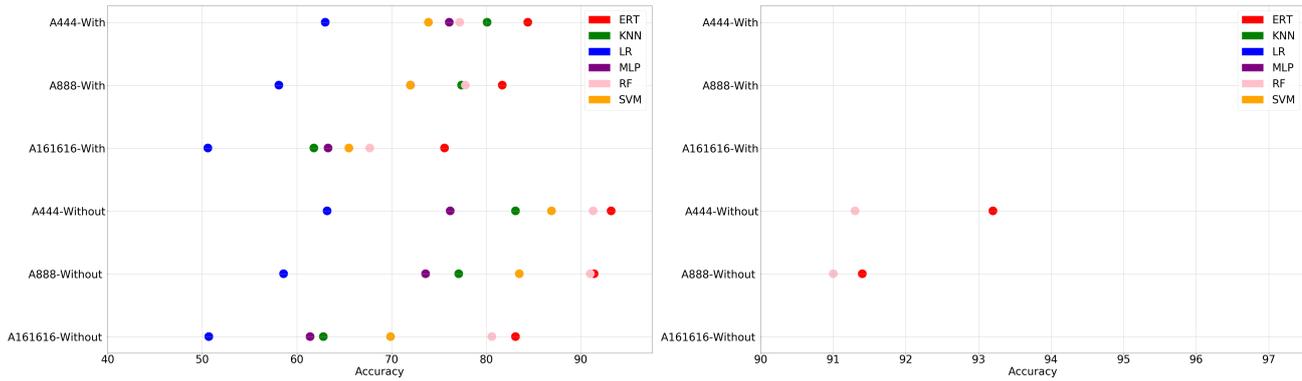


Fig. 4 **Left.** Scatter plot showing the accuracy of all the combinations of expert features and classifiers. **Right.** Zoom of the scatter plot on the combinations of expert features and classifiers that obtain an accuracy higher than 90%.

CV features	ERT	KNN	LR	MLP	RF	SVM	Test (Anova or Friedman)	After post-hoc procedure
Haralick	<i>90.0(2.0)</i>	50.4(4.6)	48.8(4.6)	40.1(5.3)	86.3(3.0)	62.8(3.5)	203.8***	ERT \simeq RF; ERT > SVM, KNN, LR, MLP
HOG	72.5(4.1)	60.0(3.9)	73.0(4.0)	75.3(3.4)	60.5(4.9)	<i>75.9(5.2)</i>	34.2***	SVM \simeq MLP, ERT, LR; SVM > RF, KNN

Table 4 Mean (and standard deviation) for the different studied models without considering the control image to generate the feature vectors. The best result for each feature extraction method is in *italics*, and the best result in **bold face**. *** $p < 0.001$; >: there are significant differences; \simeq : there are not significant differences.

CV features	ERT	KNN	LR	MLP	RF	SVM	Test (ANOVA or Friedman)	After post-hoc procedure
Haralick	<i>92.7(2.5)</i>	55.6(3.7)	47.0(4.3)	33.2(0.5)	88.8(3.3)	65.2(4.4)	1648.8***	ERT \simeq RF; ERT > SVM, KNN, LR, MLP
HOG	<i>95.4(1.5)</i>	81.3(2.9)	92.2(1.7)	92.4(1.9)	92.7(1.9)	92.4(1.4)	20.3***	ERT > RF, MLP, SVM, LR, KNN

Table 5 Mean (and standard deviation) for the different studied models considering the control image to generate the feature vectors. The best result for each feature extraction method is in *italics*, and the best result in **bold face**. *** $p < 0.001$; >: there are significant differences; \simeq : there are not significant differences.

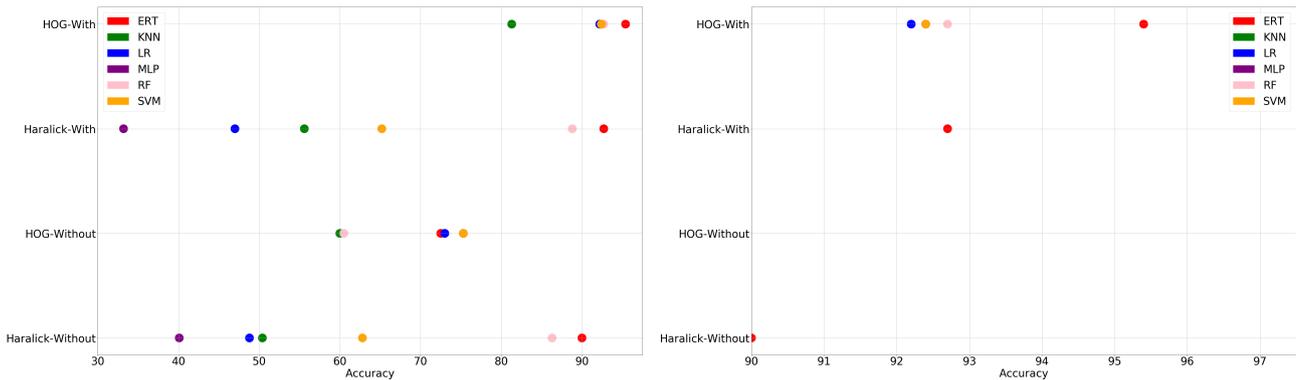


Fig. 5 **Left.** Scatter plot showing the accuracy of all the combinations of traditional features and classifiers. **Right.** Zoom of the scatter plot on the combinations of traditional features and classifiers that obtain an accuracy higher than 90%.

Network	ERT	KNN	LR	MLP	RF	SVM	Test (Anova or Friedman)	After post-hoc procedure
DenseNet	91.4(1.7)	84.0(3.3)	90.1(2.5)	57.3(10.4)	87.3(2.1)	33.3(4.7)	64.42***	ERT \simeq LR, RF; ERT $>$ MLP, SVM, KNN
GoogleNet	92.4(2.1)	49.2(3.8)	89.4 (2.4)	85.7 (5.9)	89.4 (2.1)	60.5(4.8)	50.2***	ERT \simeq LR; ERT $>$ RF, MLP, SVM, KNN
Inception v3	88.6(2.8)	83.1(3.5)	<i>92.6(1.2)</i>	91.1(2.1)	80.0(2.7)	34.6(4.8)	68.0***	LR \simeq MLP, ERT; LR $>$ KNN, RF, SVM
OverFeat	89.5(2.5)	85.8(4.0)	91.2(2.5)	91.7(2.3)	85.8(4.0)	<i>92.5(2.3)</i>	22.25***	SVM \simeq MLP, LR, ERT; SVM $>$ KNN, RF
Resnet 50	93.5(1.9)	46.4(4.9)	<i>94.5(1.7)</i>	93.3(2.7)	89.9(2.1)	73.1(6.1)	75.4***	LR \simeq ERT, MLP; LR $>$ RF, SVM, KNN
VGG16	89.9(2.3)	79.1(3.1)	<i>91.7(1.8)</i>	89.8(2.8)	82.5(2.2)	31.3(4.9)	81.7***	LR \simeq ERT, MLP; LR $>$ RF, KNN, SVM
VGG19	90.1(2.1)	84.4(3.1)	<i>92.7(2.3)</i>	90.9(2.4)	78.7(4.3)	33.1(4.7)	85.8***	LR \simeq MLP, ERT; LR $>$ KNN, RF, SVM
Xception v1	90.1(2.7)	87.8(2.9)	<i>93.5(1.6)</i>	92.2(2.0)	82.1(3.7)	91.9(1.3)	21.7***	LR \simeq MLP, SVM, ERT; LR $>$ KNN, RF

Table 6 Mean (and standard deviation) for the different studied models without considering the control image to generate the feature vectors. The best result for each DNN in *italics*, the best result in **bold** face. *** $p < 0.001$; $>$: there are significant differences; \simeq : there are not significant differences.

fect with the rest of the models using Cohen’s d, we obtain a medium size effect (0.57) when compared with Xception-LR, and a large one (ranging from 0.84 to 2.47) when compared with the rest of the models.

When the features are extracted using not only the image, but also the control image; the models to compare are DenseNet-ERT, GoogleNet-SVM, Inception-LR, OverFeat-SVM, Resnet-SVM, VGG16-ERT, VGG19-LR, and Xception-LR. In this case, there are found significant differences in the accuracy between the models (Friedman $F = 6.24$; $p = 1.42 \times 10^{-5}$) — Resnet-SVM \simeq DenseNet-ERT, GoogleNet-SVM, Inception-LR, VGG16-ERT, VGG19-LR, Xception-LR; Resnet-SVM $>$ OverFeat-SVM— and a large size effect (eta squared = 0.17) is obtained. If we consider the model with the best mean (Resnet-SVM), and take into account the size effect with the rest of the models us-

ing Cohen’s d, we obtain a small size effect (0.14) when compared with DenseNet-ERT, and medium-large size effects (ranging from 0.53 to 1.58) when compared with the rest of the models.

4.4 Comparing the best methods

Finally, we compare the best methods obtained with each approach using the methodology presented in Section 3.4. Namely, we compare the following methods: from the expert approach, A444-W-ERT (A444 without normalisation and using ERT) and A444-N-ERT (A444 with normalisation and using ERT); from the traditional computer vision approach, Haralick-W-ERT (Haralick without control image and using ERT) and HOG-C-ERT (HOG with control image and using

Network	ERT	KNN	LR	MLP	RF	SVM	Test (ANOVA or Friedman)	After post-hoc procedure
DenseNet	<i>96.2(2.3)</i>	85.5(4.3)	94.3(2.8)	62.2(18.6)	93.9(2.9)	42.5(4.6)	119.57***	ERT \simeq LR, RF; ERT > MLP, SVM, KNN
GoogleNet	92.5(3.1)	88.6(2.5)	92.4(2.8)	92.0(3.1)	88.6(4.2)	<i>95.4(2.2)</i>	0.35	SVM \simeq ERT, LR, MLP, KNN, RF
Inception v3	93.0(2.7)	87.6(3.1)	<i>95.5(1.6)</i>	94.3(2.1)	86.8(2.0)	46.4(4.8)	79.2***	LR \simeq MLP, ERT; LR > KNN, RF, SVM
OverFeat	87.2(2.4)	82.6(4.5)	92.7(2.1)	92.2(2.6)	82.0(3.5)	<i>93.0(2.4)</i>	58.8***	SVM \simeq LR, MLP; SVM > ERT, KNN, RF
Resnet 50	92.6(2.8)	90.1(3.2)	95.2(2.3)	94.7(2.3)	89.6(1.8)	<i>96.5(1.6)</i>	-0.74	SVM \simeq LR, MLP, ERT, KNN, RF
VGG16	<i>95.0(2.0)</i>	86.4(2.3)	94.7(1.7)	92.4(1.7)	89.2(3.5)	33.1(4.3)	78.7***	ERT \simeq LR, MLP; ERT > RF, KNN, SVM
VGG19	94.4(1.6)	84.5(2.9)	<i>94.6(2.3)</i>	92.4(2.7)	87.1(2.3)	33.7(4.4)	60.6***	LR \simeq ERT, MLP; LR > RF, KNN, SVM
Xception v1	93.5(2.7)	89.9(4.4)	<i>95.2(2.3)</i>	94.8(1.7)	86.8(3.0)	94.8(1.9)	20.3	LR \simeq SVM, MLP; ERT; LR > KNN, RF

Table 7 Mean (and standard deviation) for the different studied models considering the control image to generate the feature vectors. The best result for each DNN in *italics*, the best result in **bold face**. *** $p < 0.001$; >: there are significant differences; \simeq : there are not significant differences.

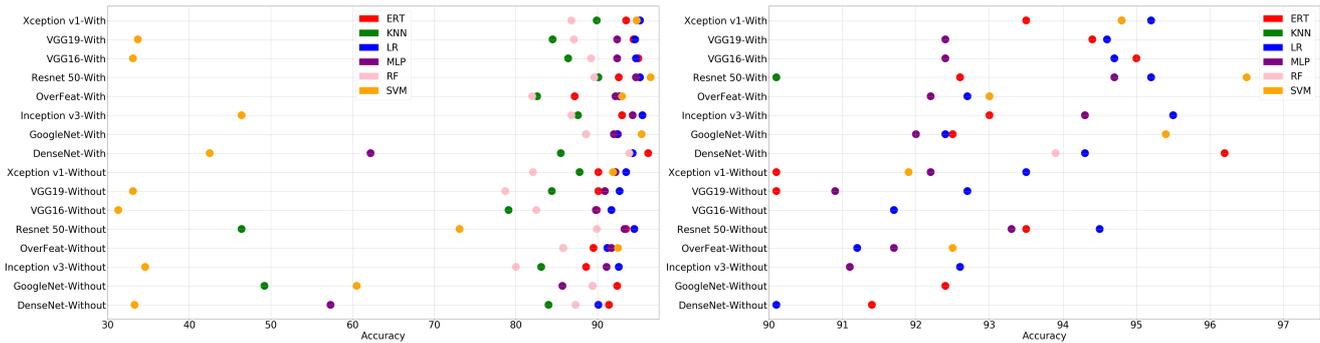


Fig. 6 Left. Scatter plot showing the accuracy of all the combinations of deep features and classifiers. Right. Zoom of the scatter plot on the combinations of deep features and classifiers that obtain an accuracy higher than 90%.

ERT); and, from the deep learning approach, Resnet-W-LR (Resnet without control image and using LR) and Resnet-C-SVM (Resnet with control image and using SVM). We repeat in Table 8 the accuracy obtained by each of those methods, and introduce a graphical comparison of those methods in Figure 7.

In order to compare these methods, the non-parametric Friedman’s test is employed since the normality condition is not fulfilled (Shapiro-Wilk’s test $W = 0.924496$; $p = 0.001169$). The Friedman’s test performs a ranking of the models compared (see Table 8), assuming as null hypothesis that all the models have the same performance. We obtain significant differences ($F = 29.55$; $p < 3.61 \times 10^{-13}$), with a large size effect $\eta^2 = 0.77$.

The Holm algorithm was employed to compare the control model (winner) with all the other models ad-

justing the p value, results are shown in Table 9. As it can be observed in Table 9, there are four techniques with no significant differences as we failed to reject the null hypothesis. The size effect is also taken into account using Cohen’s d , and as it is shown in Table 9, it is medium or large when we compare the winning model with the rest of the models.

As it can be seen in those tables, the winner model is Resnet-C-SVM with an extremely good accuracy of 96.5%. Although, the use of traditional computer vision features losses in both cases (with control image and without it) with respect to the use of a DNN, the networks trained using traditional features only obtain a 1% less accuracy that the networks trained using deep features; and, indeed, the use of traditional computer vision features that use the control images outperforms the use of DNN features that do not use the control

Technique	Accuracy	Friedman's test average ranking
Resnet-C-SVM	96.5 (1.6)	5.25
HOG-C-ERT	95.4 (1.5)	4.65
Resnet-W-LR	94.5 (1.7)	4.4
A444-W-ERT	93.2 (2.7)	3.6
Haralick-W-ERT	90.0 (2.0)	2.1
A444-N-ERT	84.4 (3.1)	1

Table 8 Best methods obtained with each approach.

Technique	Z value	p value	adjusted p value	Cohen's d
HOG-C-ERT	0.71	0.47	0.61	0.66
Resnet-W-LR	1.01	0.30	0.61	1.11
A444-W-ERT	1.97	0.048	0.14	1.34
Haralick-W-ERT	3.76	1.6×10^{-4}	6.66×10^{-4}	3.48
A444-N-ERT	5.07	3.77×10^{-7}	1.88×10^{-6}	4.56

Table 9 Adjusted p -values with Holm, and Cohen's d. Control technique: Resnet-C-SVM.

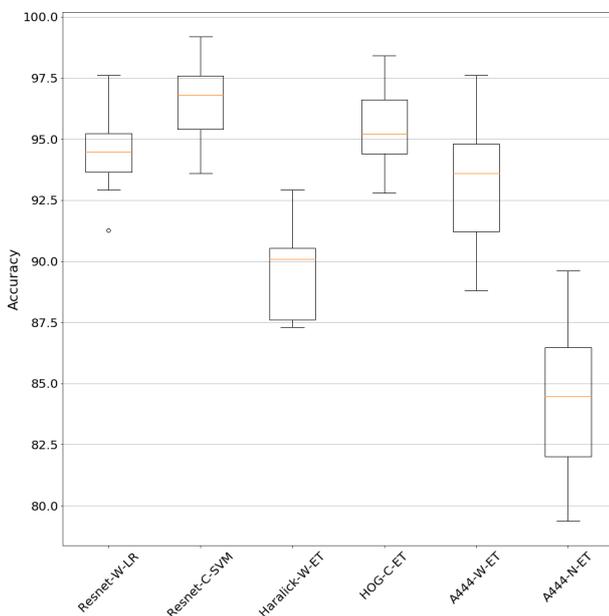


Fig. 7 Results from 10 independent runs in accuracy for the best method in each approach.

image. Moreover, the expert method approach also obtains a remarkable accuracy of 93.2% which outperforms traditional computer vision features when control images are not used. Finally, independently of the approach that is employed to build the model, we can create models which accuracy is over 90%, except in the case of bins with normalisation.

5 Discussion and conclusions

Automated digital plate reading is increasingly adopted as a mean to improve the quality, efficiency and reproducibility withing laboratories. However, current automated digital plate reading systems and tools are designed to analyse bacterial culture plates or yeasts, but

not fungal strains [60]. Moreover, the research devoted to analyse images of fungal strains has been focused on measuring the diameter of colored halos using semi-automated techniques [74,55,5]. Hence, and up to the best of our knowledge, the work presented in this paper is the first time that the problem of automatically characterising the dye decolorisation level of fungal strains has been tackled.

Such a problem, that fits in the category of image-classification problems, can be undertaken by employing different computer vision and machine learning techniques. In this paper, we have conducted a thorough study, using statistical methods, of different alternatives that combine features extracted from the experts' knowledge, traditional computer vision methods, and deep learning techniques with the most common classification algorithms.

In the literature, DNNs have shown an outstanding performance in image-classification tasks by greatly improving traditional and ad-hoc methods [59,71,16]. However, DNNs are greedy, and training them from scratch might be challenging due to the huge amount of data and time that they need [59]. Instead, the use of transfer learning takes advantage of DNNs but without the prohibitive costs associated with them, and also obtaining good accuracies. This result is also obtained in this paper where the best model is based on the transfer learning approach, achieving an accuracy of 96.5%. Nevertheless, in this particular scenario, a similar accuracy can be obtained either using traditional features or features coming from the experts' knowledge.

One of the greatest benefits of DNNs is that they remove the need for most feature engineering, since they are capable of extracting useful features from raw data. However, this does not mean that feature engineering is no longer necessary since good features can allow us to solve problems in a better way. This can be seen in the

results obtained in this work, the best model is achieved when DNNs do not only work with raw data, but also take into account some extra information: the control image. Even more, DNNs working only with raw data are outperformed by traditional computer vision features combined with some feature engineering obtained from the control image. Therefore, we can conclude that applying blindly DNNs will probably produce good results, but these results can be improved adding some expert knowledge to those networks.

As conclusion of this work, we have presented the first model that automatically characterises the dye decolorisation level of fungal strains. Such a model has a high accuracy and has been selected after applying an exhaustive statistical study of different alternatives. Hence, this model can greatly reduce the burden and subjectivity of visually classifying the dye decolorisation level by providing a standard and reproducible method.

The main task that remains as further work consists in developing a tool that, using the best classification model found in this paper, can be easily employed by researchers to measure the dye decolorisation level of their fungal strains. In order to facilitate its use, we are planning to create a web application that will be freely accessible.

Compliance with Ethical Standards

Funding: This work was partially supported by the Ministerio de Economía y Competitividad [MTM2014-54151-P, MTM2017-88804-P], and Agencia de Desarrollo Económico de La Rioja [2017-I-IDD-00018].

Conflict of Interest: All the authors declare that they have no conflict of interest.

Ethical approval: This article does not contain any studies with human participants or animals performed by any of the authors.

References

- Abadi, M., et al.: TensorFlow: Large-scale machine learning on heterogeneous systems (2015). URL <http://tensorflow.org/>. Software available from tensorflow.org
- Abdel-Raheem, A., Shearer, C.A.: Extracellular enzyme production by freshwater ascomycetes. *Fungal Diversity* **11**, 1–19 (2002)
- Aguilar-Pontes, M.W., et al.: (Post-) Genomics approaches in fungal research. *Briefings in Functional Genomics* **13**(6), 424–439 (2014)
- Anastasi, A., et al.: Decolourisation of model and industrial dyes by mitosporic fungi in different culture conditions. *World Journal of Microbiology and Biotechnology* **25**(8), 1363–1374 (2009)
- Andrews, M.Y., et al.: Digital image quantification of siderophores on agar plates. *Data in Brief* **6**, 890–898 (2016)
- Bishop, C.M.: *Neural Networks for Pattern Recognition*. Oxford University Press, UK (1995)
- Branco, P., Torgo, L., Ribeiro, R.: A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys* **49**(2), 31:1–31:50 (2016)
- Breiman, L.: Random Forests. *Machine Learning* **45**(1), 5–32 (2001)
- Casieri, L., et al.: Survey of ectomycorrhizal, litter-degrading, and wood-degrading basidiomycetes for dye decolorization and ligninolytic enzyme activity. *Antonie van Leeuwenhoek* **98**(4), 483–504 (2010)
- Chambergo, F.S., Valencia, E.Y.: Fungal biodiversity to biotechnology. *Applied Microbiology and Biotechnology* **100**(6), 2567–2577 (2016)
- Chawla, N.V., Bowyer, K.W., Hall, L., Kegelmeyer, W.: Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research* **16**(1), 321–357 (2002)
- Chawla, N.V., Japkowicz, N., Kotcz, A.: Editorial: Special issue on learning from imbalanced datasets. *ACM SIGKDD Explorations Newsletter* **6**(1), 1–6 (2004)
- Chollet, F.: Xception: Deep Learning with Depthwise Separable Convolutions. *CoRR* **abs/1610.02357** (2016). URL <http://arxiv.org/abs/1610.02357>
- Chollet, F., et al.: Keras (2015). <https://github.com/fchollet/keras>
- Christodoulidis, S., et al.: Multisource Transfer Learning With Convolutional Neural Networks for Lung Pattern Analysis. *IEEE Journal of Biomedical and Health Informatics* **21**(1), 76–84 (2017)
- Codella, N., et al.: Deep Learning, Sparse Coding, and SVM for Melanoma Recognition in Dermoscopy Images. In: *Proceedings of International Workshop on Machine Learning in Medical Imaging (MICCAI 2015)*, Lecture Notes in Computer Science, pp. 118–126. Springer, Germany (2015)
- Coelho, L.P.: Mahotas: Open source software for scriptable computer vision. *Journal of Open Research Software* **1**(1), e3 (2013)
- Cohen, J.: *Statistical Power Analysis for the Behavioral Sciences*. Academic Press, USA (1969)
- Cohen, J.: Eta-squared and partial eta-squared in fixed factor anova designs. *Educational and Psychological Measurement* **33**, 107–112 (1973)
- Cortes, C., Vapnik, V.: Support-Vector Networks. *Machine Learning* **20**(3), 273–297 (1995)
- Cover, T., Hart, P.: Nearest Neighbor Pattern Classification. *IEEE Trans. Inf. Theor.* **13**(1), 21–27 (2006)
- Culibrk, L., et al.: Systems biology approaches for host–fungal interactions: An expanding multi-omics frontier. *Omics: a Journal of Integrative Biology* **20**(3), 127–138 (2016)
- Cázares-García, S.V., et al.: Typing and selection of wild strains of trichoderma spp. producers of extracellular laccase. *Biotechnology Progress* **32**(3), 787–798 (2016)
- Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01, CVPR '05*, pp. 886–893. IEEE Computer Society, San Diego, CA, USA (2005)

25. Dhouib, A., et al.: Screening for ligninolytic enzyme production by diverse fungi from Tunisia. *World Journal of Microbiology and Biotechnology* **21**(8), 1415–1423 (2005)
26. Gao, D., et al.: A critical review of the application of white rot fungus to environmental pollution control. *Critical Reviews in Biotechnology* **30**(1), 70–77 (2010)
27. Garcia, S., et al.: Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Information Sciences* **180**, 2044–2064 (2010)
28. Geurts, P., Ernst, D., Wehenkel, L.: Extremely randomized trees. *Machine Learning* **63**(1), 3–42 (2006)
29. Ghafoorian, M., et al.: Transfer Learning for Domain Adaptation in MRI: Application in Brain Lesion Segmentation. *CoRR abs/1702.07841* (2017). URL <http://arxiv.org/abs/1702.07841>
30. Hanking, L., Anagnostakis, S.L.: The use of solid media for detection of enzyme production by fungi. *Mycology* **67**(3), 597–607 (1975)
31. Haralick, R.M., Shanmugam, K., Dinstein, I.: Textural features for image classification. *IEEE Transactions on Systems, Man and Cybernetics SMC-3*(6), 610–621 (1973)
32. He, K., et al.: Deep Residual Learning for Image Recognition. In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'16)*, IEEE Computer Society, pp. 770–778. IEEE, Las Vegas, USA (2016)
33. Holm, O.S.: A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics* **6**, 65–70 (1979)
34. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'17)* (2017)
35. Hunter, R.S.: Photoelectric Color-Difference Meter. *Journal of the Optical Society of America* **38**(7), 661 (1948)
36. Hyun, M.W., et al.: Detection of cellulolytic activity in *Ophiostoma* and *Leptographium* species by chromogenic reaction. *Mycobiology* **34**(2), 108–110 (2006)
37. Jayasiri, S.C., et al.: The faces of fungi database: fungal names linked with morphology, phylogeny and human impacts. *Fungal Diversity* **74**(1), 3–18 (2015)
38. Kaehler, A., Bradski, G.: *Learning OpenCV 3*. O'Reilly Media, USA (2015)
39. Kameshwar, A.K.S., Qin, W.: Qualitative and quantitative methods for isolation and characterization of lignin-modifying enzymes secreted by microorganisms. *BioEnergy Research* **10**(1), 248–266 (2017)
40. Kasana, R.C., et al.: A rapid and easy method for the detection of microbial cellulases on agar plates using Gram's iodine. *Current Microbiology* **57**(5), 503–507 (2008)
41. Kiiskinen, L.L., et al.: Screening for novel laccase producing microbes. *Journal of Applied Microbiology* **97**, 640–646 (2004)
42. Kornilłowicz-Kowalska, T., Rybczyńska, K.: Screening of microscopic fungi and their enzyme activities for decolorization and biotransformation of some aromatic compounds. *International Journal of Environmental Science and Technology* **12**(8), 2673–2686 (2015)
43. Krizhevsky, A., et al.: ImageNet Classification with Deep Convolutional Neural Networks. In: F. Pereira, C.J.C. Burges, L. Bottou, K.Q. Weinberger (eds.) *Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Curran Associates, Inc., USA (2012)
44. LeCun, Y., et al.: *Neural Networks: Tricks of the Trade*, *Lecture Notes in Computer Science*, vol. 1524, chap. Efficient BackProp, pp. 9–50. Springer, Berlin (2002)
45. Lee, H., et al.: Biotechnological procedures to select white rot fungi for the degradation of PAHs. *Journal of Microbiological Methods* **97**, 56–62 (2014)
46. Levene, H.: Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling, chap. Robust tests for equality of variances, pp. 278–292. *Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling*, Stanford University Press, USA (1960)
47. McCullagh, P., Nelder, J.A.: *Generalized Linear Models*. Chapman & Hall, London (1989)
48. Menegola, A., et al.: Knowledge Transfer for Melanoma Screening with Deep Learning. *CoRR abs/1703.07479* (2017). URL <http://arxiv.org/abs/1703.07479>
49. Mouhamadou, B., et al.: Molecular screening of xerophilic *Aspergillus* strains producing mycophenolic acid. *Fungal Biology* **121**(2), 103–111 (2017)
50. Nyanhongo, G.S., et al.: Decolorization of textile dyes by laccases from a newly isolated strain of *Trametes modesta*. *Water Research* **36**, 1449–1456 (2002)
51. Oses, R., et al.: Evaluation of fungal endophytes for lignocellulolytic enzyme production and wood biodegradation. *International Biodeterioration & Biodegradation* **57**(2), 129–135 (2006)
52. Pan, S.J., Yang, Q.: A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering* **22**(10), 1345–1359 (2010)
53. Peay, K.G., et al.: Dimensions of biodiversity in the Earth mycobiome. *Nature Reviews Microbiology* **14**(7), 434–447 (2016)
54. Pedregosa, F., et al.: Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* **12**, 2825–2830 (2011)
55. Pedrini, N., et al.: Control of pyrethroid-resistant chagas disease vectors with entomopathogenic fungi. *PLOS Neglected Tropical Diseases* **3**, 1–11 (2009)
56. Pointing, S.B.: Qualitative methods for the determination of lignocellulolytic enzyme production by tropical fungi. *Fungal Diversity* **2**, 17–33 (1999)
57. Pointing, S.B., et al.: Dye decolorization by sub-tropical basidiomycetous fungi and the effect of metals on decolorizing ability. *World Journal of Microbiology and Biotechnology* **16**(2), 199–205 (2000)
58. Pointing, S.B., et al.: Production of wood-decay enzymes, mass loss and lignin solubilization in wood by tropical xylariaceae. *Mycological Research* **107**(2), 231–235 (2003)
59. Razavian, A.S., et al.: CNN features off-the-shelf: An astounding baseline for recognition. In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW'14)*, IEEE Computer Society, pp. 512–519. IEEE, Columbus, Ohio, USA (2014)
60. Rhoads, D.D., et al.: A review of the current state of digital plate reading of cultures in clinical microbiology. *Journal of Pathology Informatics* **6**(23), 1–8 (2015)
61. Rodríguez-Fdez, I., et al.: STAC: a web platform for the comparison of algorithms using statistical tests. In: *Proceedings of the 2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)* (2015)
62. Rovati, J.I., et al.: Polyphenolic substrates and dyes degradation by yeasts from 25 de Mayo/King George Island (Antarctica). *Yeast* **30**(11), 459–470 (2013)
63. Russakovsky, O., et al.: ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision* **115**(3), 211–252 (2015)
64. Schoch, C.L., et al.: Finding needles in haystacks: linking scientific names, reference specimens and molecular data for fungi. *Database* (2014). DOI 10.1093/database/bau061

65. Sermanet, P., et al.: OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. CoRR [abs/1312.6229](https://arxiv.org/abs/1312.6229) (2013). URL <http://arxiv.org/abs/1312.6229>
66. Shapiro, S.S., Wilk, M.B.: An analysis for variance test for normality (complete samples). *Information Sciences* **180**, 2044–2064 (1965)
67. Sheskin, D.: *Handbook of Parametric and Nonparametric Statistical Procedures*. CRC Press, London (2011)
68. Simard, P., Steinkraus, D., Platt, J.C.: Best practices for convolutional neural networks applied to visual document analysis. In: I.C. Society (ed.) *Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR'03)*, vol. 2, pp. 958–964 (2003)
69. Simonis, J.L., et al.: Extracellular enzymes and soft rot decay: are ascomycetes important degraders in fresh water? *Fungal Diversity* **31**(1), 135–146 (2008)
70. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. CoRR [abs/1409.1556](https://arxiv.org/abs/1409.1556) (2014). URL <http://arxiv.org/abs/1409.1556>
71. Szegedy, C., et al.: DeepFace: Closing the Gap to Human-Level Performance in Face Verification. In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'14)*, IEEE Computer Society, pp. 1701–1708. IEEE, USA (2014)
72. Szegedy, C., et al.: Going deeper with convolutions. In: *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'15)*, IEEE Computer Society, pp. 1–9. IEEE, Boston, USA (2015)
73. Szegedy, C., et al.: Rethinking the Inception Architecture for Computer Vision. CoRR [abs/1512.00567](https://arxiv.org/abs/1512.00567) (2015). URL <http://arxiv.org/abs/1512.00567>
74. Szekeres, A., et al.: A novel, image analysis-based method for the evaluation of in vitro antagonism. *Journal of Microbiological Methods* **65**(3), 619–622 (2006)
75. Szeliski, R.: *Computer Vision: Algorithms and Applications*. Springer, London (2010)
76. Sørensen, A., et al.: Onsite enzyme production during bioethanol production from biomass: screening for suitable fungal strains. *Applied Biochemistry and Biotechnology* **164**(7), 1058–1070 (2011)
77. Tortella, G.R., et al.: Enzymatic characterization of Chilean native wood-rotting fungi for potential use in the bioremediation of polluted environments with chlorophenols. *World Journal of Microbiology and Biotechnology* **24**(12), 2805 (2008)
78. Wolpert, D.H.: The lack of a priori distinction between learning algorithms. *Neural Computation* **8**(7), 1341–1390 (1996)
79. Xu, C., et al.: Screening of ligninolytic fungi for biological pretreatment of lignocellulosic biomass. *Canadian Journal of Microbiology* **61**(10), 745–752 (2015)