



**HAL**  
open science

## Patch-based CNN evaluation for bark classification

Debaleena Misra, Carlos F Crispim-Junior, Laure Tougne

► **To cite this version:**

Debaleena Misra, Carlos F Crispim-Junior, Laure Tougne. Patch-based CNN evaluation for bark classification. Workshop on Computer Vision Problems in Plant Phenotyping, Aug 2020, Edinburgh, United Kingdom. hal-02969811v2

**HAL Id: hal-02969811**

**<https://hal.science/hal-02969811v2>**

Submitted on 19 Oct 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Patch-based CNN evaluation for bark classification

Debaleena Misra<sup>1</sup>, Carlos Crispim-Junior<sup>1</sup>[0000-0002-5577-5335], and Laure Tougne<sup>1</sup>[0000-0001-9208-6275]

Univ Lyon, Lyon 2, LIRIS, F-69676 Lyon, France  
{debaleena.misra, carlos.crispim-junior, laure.tougne}@liris.cnrs.fr

**Abstract.** The identification of tree species from bark images is a challenging computer vision problem. However, even in the era of deep learning today, bark recognition continues to be explored by traditional methods using time-consuming handcrafted features, mainly due to the problem of limited data. In this work, we implement a patch-based convolutional neural network alternative for analyzing a challenging bark dataset Bark-101, comprising of 2587 images from 101 classes. We propose to apply image re-scaling during the patch extraction process to compensate for the lack of sufficient data. Individual patch-level predictions from fine-tuned CNNs are then combined by classical majority voting to obtain image-level decisions. Since ties can often occur in the voting process, we investigate various tie-breaking strategies from ensemble-based classifiers. Our study outperforms the classification accuracy achieved by traditional methods applied to Bark-101, thus demonstrating the feasibility of applying patch-based CNNs to such challenging datasets.

**Keywords:** Bark classification, convolutional neural networks, transfer learning, patch-based CNNs, image re-scaling, bicubic interpolation, super-resolution networks, majority voting

## 1 Introduction

Automatic identification of tree species from images is an interesting and challenging problem in the computer vision community. As urbanization grows, our relationship with plants is fast evolving and plant recognition through digital tools provide an improved understanding of the natural environment around us. Reliable and automatic plant identification brings major benefits to many sectors, for example, in forestry inventory, agricultural automation [32], botany [27], taxonomy, medicinal plant research [3] and to the public, in general. In recent years, vision-based monitoring systems have gained importance in agricultural operations for improved productivity and efficiency [17]. Automated crop harvesting using agricultural robotics [2] for example, relies heavily on visual identification of crops from their images. Knowledge of trees can also provide landmarks in localization and mapping algorithms [31].

39 Although plants have various distinguishable physical features such as leaves,  
 40 fruits or flowers, bark is the most consistent one. It is available round the year,  
 41 with no seasonal dependencies. The aging process is also a slow one, with vi-  
 42 sual features changing over longer periods of time while being consistent during  
 43 shorter time frames. Even after trees have been felled, their bark remains an  
 44 important identifier, which can be helpful for example, in autonomous timber  
 45 assessment. Barks are also more easily visually accessible, contrary to higher-  
 46 level leaves, fruits or flowers. However, due to the low inter-class variance and  
 47 high intra-class variance for bark data, the differences are very subtle. Besides,  
 48 bark texture properties are also impacted by the environment and plant diseases.  
 49 Uncontrolled illumination alterations and branch shadow clutter can addition-  
 50 ally affect image quality. Hence, tree identification from only bark images is a  
 51 challenging task not only for machine learning approaches [5][7][8][25] but also  
 52 for human experts [13].

53  
 54 Recent developments in deep neural networks have shown great progress in image  
 55 recognition tasks, which can help automate manual recognition methods that are  
 56 often laborious and time consuming. However, a major limitation of deep learn-  
 57 ing algorithms is that a huge amount of training data is required for attaining  
 58 good performance. For example, the ImageNet dataset [11] has over 14 million  
 59 images. Unfortunately, the publicly available bark datasets are very limited in  
 60 size and variety. Recently released *BarkNet 1.0* dataset [8] with 23,000 images  
 61 for 23 different species, is the largest in terms of number of instances, while  
 62 *Bark-101* dataset [25] with 2587 images and 101 different classes, is the largest  
 63 in terms of number of classes. The data deficiency of reliable bark datasets in  
 64 literature presumably explains why majority of bark identification research has  
 65 revolved around hand-crafted features and filters such as Gabor [4][18], SIFT  
 66 [9][13], Local Binary Pattern (LBP) [6][7][25] and histogram analysis [5], which  
 67 can be learned from lesser data.

68  
 69 In this context, we study the challenges of applying deep learning in bark recog-  
 70 nition from limited data. The objective of this paper is to investigate patch-based  
 71 convolutional neural networks for classifying the challenging Bark-101 dataset  
 72 that has low inter-class variance, high intra-class variance and some classes with  
 73 very few samples. To tackle the problem of insufficient data, we enlarge the train-  
 74 ing data by using patches cropped from original Bark-101 images. We propose  
 75 a patch-extraction approach with image re-scaling prior to training, to avoid  
 76 random re-sizing during image-loading. For re-scaling, we compare traditional  
 77 bicubic interpolation [10] with a more recent advance of re-scaling by super-  
 78 resolution convolutional neural networks [12]. After image re-scale and patch-  
 79 extraction, we fine-tune pre-trained CNN models with these patches. We obtain  
 80 patch-level predictions which are then combined in a majority voting fashion to  
 81 attain image-level results. However, there can be ties, i.e. more than one class  
 82 could get the largest count of votes, and it can be challenging when a considerable  
 83 number of ties occur. In our study, we apply concepts of ensemble-based classi-

84 fiers and investigate various tie-breaking strategies [22][23][26][34][35] of major-  
 85 ity voting. We validated our approach on three pre-trained CNNs - Squeezenet  
 86 [20], MobileNetV2 [28] and VGG16 [30], of which the first two are compact and  
 87 light-weight models that could be used for applications on mobile devices in the  
 88 future. Our study demonstrates the feasibility of using deep neural networks for  
 89 challenging datasets and outperforms the classification accuracy achieved using  
 90 traditional hand-crafted methods on Bark-101 in the original work [25].

91  
 92 The rest of the paper is organised as follows. Section 2 reviews existing ap-  
 93 proaches in bark classification. Then, section 3 explains our methodology for  
 94 patch-based CNNs. Section 4 describes the experimentation details. Our results  
 95 and insights are presented in section 5. Finally, section 6 concludes the study  
 96 with discussions on possible future work.

## 97 2 RELATED WORK

98 Traditionally, bark recognition has been studied as a texture classification prob-  
 99 lem using statistical methods and hand-crafted features. Bark features from 160  
 100 images were extracted in [33] using textual analysis methods such as gray level  
 101 run-length method (RLM), concurrence matrices (COMM) and histogram in-  
 102 spection. Additionally, the authors captured the color information by applying  
 103 the grayscale methods individually to each of the 3 RGB channels and the overall  
 104 performance significantly improved. Spectral methods using Gabor filters [4] and  
 105 descriptors of points of interests like SURF or SIFT [9][13][16] have also been  
 106 used for bark feature extraction. The AFF bark dataset, having 11 classes and  
 107 1082 bark images, was analysed by a bag of words model with an SVM classifier  
 108 constructed from SIFT feature points achieving around 70% accuracy [13].

109  
 110 An earlier study [5] proposed a fusion of color hue and texture analysis for  
 111 bark identification. First the bark structure and distribution of contours (scales,  
 112 straps, cracks etc) were described by two descriptive feature vectors computed  
 113 from a map of Canny extracted edges intersected by a regular grid. Next, the  
 114 color characteristics were captured by the hue histogram in HSV color space as  
 115 it is indifferent to illumination conditions and covers the whole range of possi-  
 116 ble bark colors with a single channel. Finally, image filtering by Gabor wavelets  
 117 was used to extract the orientation feature vector. An extended study on the  
 118 resultant descriptor from concatenation of these four feature vectors, showed  
 119 improved performance in tree identification when combined with leaves [4]. Sev-  
 120 eral works have also been based on descriptors such as Local Binary Patterns  
 121 (LBP) and LBP-like filters [6][7][25]. Late Statistics (LS) with two state-of-art  
 122 LBP-like filters - Light Combination of Local Binary Patterns (LCoLBP) and  
 123 Completed Local Binary Pattern (CLBP) were defined, along with bark priors on  
 124 reduced histograms in the HSV space to capture color information [25]. This ap-  
 125 proach created computationally efficient, compact feature vectors and achieved  
 126 state-of-art performance on 3 challenging datasets (BarkTex, AFF12, Bark-101)

127 with SVM and KNN classifiers. Another LBP-inspired texture descriptor called  
 128 Statistical Macro Binary Pattern (SMBP) attained improved performance in  
 129 classifying 3 datasets (BarkTex, Trunk12, AFF) [7]. SMBP encodes macrostruc-  
 130 ture information with statistical description of intensity distribution which is  
 131 rotation-invariant and applies an LBP-like encoding scheme, thus being invari-  
 132 ant to monotonic gray scale changes.

133  
 134 Some early works [18][19] in bark research have interestingly been attempted  
 135 using artificial neural networks (ANN) as classifiers. In 2006, Gabor wavelets  
 136 were used to extract bark texture features and applied to a radial basis proba-  
 137 bilistic neural network (RBPNN) for classification [18]. It achieved around 80%  
 138 accuracy on a dataset of 300 bark images. GLCM features have also been used  
 139 in combination with fractal dimension features to describe the complexity and  
 140 self-similarity of varied scaled texture [19]. They used a 3-layer ANN classifier on  
 141 a dataset of 360 images having 24 classes and obtained an accuracy of 91.67%.  
 142 However, this was before the emergence of deep learning convolutional neural  
 143 networks for image recognition.

144  
 145 Recently, there have been few attempts to identify trees from only bark informa-  
 146 tion using deep-learning. LIDAR scans created depth images from point clouds,  
 147 which were applied to AlexNet resulting in 90% accuracy, using two species only  
 148 - Japanese Cedar and Japanese Cypress [24]. Closer to our study with RGB  
 149 images, patches of bark images have been used to fine-tune pre-trained deep  
 150 learning models [15]. With constraints on the minimum number of crops and  
 151 projected size of tree on plane, they attained 96.7% accuracy, using more than  
 152 10,000 patches for 221 different species. However, the report lacked clarity on the  
 153 CNN architecture used and the experiments were performed on private data pro-  
 154 vided by a company, therefore inaccessible for comparisons. Image patches were  
 155 also used for transfer-learning with ResNets to identify species from the BarkNet  
 156 dataset [8]. This work obtained an impressive accuracy of 93.88% for single crop  
 157 and 97.81% using majority voting on multiple crops. However, BarkNet is a large  
 158 dataset having 23,000 high-resolution images for 23 classes, which significantly  
 159 reduces the challenges involved. We draw inspiration from these works and build  
 160 on them to study an even more challenging dataset - Bark-101 [25].

## 161 3 METHODOLOGY

162 Our methodology presents a plan of actions consisting of four main components:  
 163 *Image re-scaling, patch-extraction, fine-tuning pre-trained CNNs* and *majority*  
 164 *voting analysis*. The following sections discuss our work-flow in detail.

### 165 3.1 Dataset

166 In our study, we chose the Bark-101 dataset. It was developed from PlantCLEF  
 167 database, which is part of the ImageCLEF challenges for plant recognition [21].

168 Bark-101 consists of a total of 2587 images (split into 1292 train and 1295 test  
 169 images) belonging to 101 different species. Two observations about Bark-101 ex-  
 170 plain the difficulty level of this dataset. Firstly, these images simulate real world  
 171 conditions as PlantCLEF forms their database through crowdsourced initiatives  
 172 (for example from mobile applications as Pl@ntnet [1]). Although the images  
 173 have been manually segmented to remove unwanted background, Bark-101 still  
 174 contains a lot of noise in form of mosses, shadows or due to lighting conditions.  
 175 Moreover, no constraints were given for image sizes during Bark-101 preparation  
 176 leading to a huge variability of size. This is expected in practical outdoor set-  
 177 tings where tree trunk diameters fluctuate and users take pictures from varying  
 178 distances. Secondly, Bark-101 has high intra-class variability and low inter-class  
 179 variability which makes classification difficult. High intra-class variability can  
 180 be attributed to high diversity in bark textures during the lifespan of a tree.  
 181 Low inter-class variability is explained by the large number of classes (101) in  
 182 the dataset, as a higher number of species imply higher number of visually alike  
 183 textures. Therefore, Bark-101 can be considered a challenging dataset for bark  
 recognition.



Fig. 1. Example images from Bark-101 dataset.

184

### 185 3.2 Patch preparation

186 In texture recognition, local features can offer useful information to the clas-  
 187 sifier. Such local information can be obtained through extraction of patches,  
 188 which means decomposing the original image into smaller crops or segments.  
 189 Compared to semantic segmentation techniques that use single pixels as input,  
 190 patches allow to capture neighbourhood local information as well as reduces ex-  
 191 ecution time. These patches are then used for fine-tuning a pre-trained CNN  
 192 model. Thus, the patch extraction process also helps to augment the available  
 193 data for training CNNs and is particularly useful when the number of images  
 194 per class is low, as is the case with Bark-101 dataset.

195

196 Our study focused on using a patch size of 224x224 pixels, following the de-  
 197 fault ImageNet size standards used in most CNN image recognition tasks today.  
 198 However, when the range of image dimensions vary greatly within a dataset, it

199 is difficult to extract a useful number of non-overlapping patches from all im-  
 200 ages. For example, in Bark-101, around 10 percent of the data is found to have  
 201 insufficient pixels to allow even a single square patch of 224x224 size. Contrary  
 202 to common data pre-processing for CNNs where images are randomly re-sized  
 203 and cropped during data loading, we propose to prepare the patches beforehand  
 204 to have better control in the patch extraction process. This also removes the risk  
 205 of extracting highly deformed patches from low-dimension images. The original  
 206 images are first upscaled by a given factor and then patches are extracted from  
 207 them. In our experiments, we applied two different image re-scaling algorithms  
 208 - traditional bicubic interpolation method [10] and a variant of super-resolution  
 209 network, called efficient sub-pixel convolutional neural network (ESPCN) [29].

### 210 3.3 Image re-scaling

211 Image re-scaling refers to creating a new version of an image with new dimen-  
 212 sions by changing its pixel information. In our study, we apply *upsampling* which  
 213 is the process of obtaining images with increased size. However, these operations  
 214 are not loss-less and have a trade-off between efficiency, complexity, smoothness,  
 215 sharpness and speed. We tested two different algorithms to obtain high-resolution  
 216 representation of the corresponding low-resolution image (in our context, reso-  
 217 lution referring to spatial resolution, i.e. size).

218 **Bicubic interpolation** This is a classical image upsampling algorithm involv-  
 219 ing geometrical transformation of 2D images with Lagrange polynomials, cubic  
 220 splines or cubic convolutions [10]. In order to preserve sharpness and image  
 221 details, new pixel values are approximated from the surrounding pixels in the  
 222 original image. The output pixel value is computed as a weighted sum of pixels  
 223 in the 4-by-4 pixel neighborhood. The convolution kernel is composed of piece-  
 224 wise cubic polynomials. Compared to bilinear or nearest-neighbor interpolation,  
 225 bicubic takes longer time to process as more surrounding pixels are compared  
 226 but the resampled images are smoother with fewer distortions. As the destina-  
 227 tion high-resolution image pixels are estimated using only local information in  
 228 the corresponding low-resolution image, some details could be lost.

229 **Super-resolution using sub-pixel convolutional neural network** Super-  
 230 vised machine learning methods can learn mapping functions from low-resolution  
 231 images to their high-resolution representations. Recent advances in deep neu-  
 232 ral networks called Super-resolution CNN (SRCNN) [12] have shown promising  
 233 improvements in terms of computational performances and reconstruction accu-  
 234 racy. These models are trained with low-resolution images as inputs and their  
 235 high-resolution counterparts are the targets. The first convolutional layers of  
 236 such neural networks perform feature extraction from the low-resolution images.  
 237 The next layer maps these feature maps non-linearly to their corresponding  
 238 high-resolution patches. The final layer combines the predictions within a spa-  
 239 tial neighbourhood to produce the final high-resolution image.

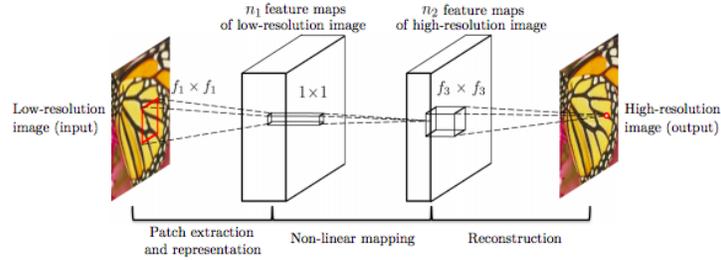


Fig. 2. SRCNN architecture [12].

240

241 In our study, we focus on the efficient sub-pixel convolutional neural network  
 242 (ESPCN) [29]. In this CNN architecture, feature maps are extracted from low-  
 243 resolution space, instead of performing the super-resolution operation in the  
 244 high-resolution space that has been upscaled by bicubic interpolation. Addition-  
 245 ally, an efficient sub-pixel convolution layer is included that learns more complex  
 246 upscaling filters (trained for each feature map) to the final low-resolution feature  
 247 maps into the high-resolution output. This architecture is shown in figure 3, with  
 248 two convolution layers for feature extraction, and one sub-pixel convolution layer  
 249 which accumulates the feature maps from low-resolution space and creates the  
 250 super-resolution image in a single step.

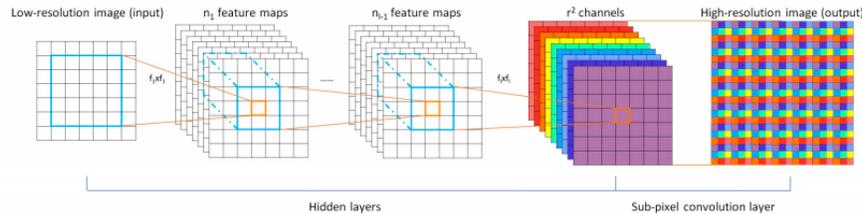


Fig. 3. Efficient sub-pixel convolutional neural network (ESPCN) [29].

251

### 3.4 CNN classification

252

253 The CNN models used for image recognition in this work are 3 recent architec-  
 254 tures: Squeezenet [20], MobileNetV2 [28] and VGG16 [30]. Since Bark-101 is a  
 255 small dataset, we apply transfer-learning and use the pre-trained weights of the  
 256 models trained on the large-scale image data of ImageNet [11]. The convolutional  
 257 layers are kept frozen and only the last fully connected layer is replaced with a  
 new one having random weights. We only train this layer with the dataset to

258 make predictions specific to the bark data. We skip the detailed discussion of  
 259 the architectures, since its beyond the scope of our study and can be found in  
 260 the references for the original works.

### 261 3.5 Evaluation metrics

262 In our study, we report two kinds of accuracy: *patch-level* and *absolute*.

- 263 – **Patch-level accuracy** - Describes performance at a patch-level, i.e. among  
 264 all the patches possible (taken from 1295 test images), we record what per-  
 265 centage of patches has been correctly classified.
- 266 – **Absolute accuracy** - Refers to overall accuracy in the test data, i.e. how  
 267 many among 1295 test images of Bark-101 could be correctly identified. In  
 268 our patch-based CNN classification, we computed this by majority voting  
 269 using 4 variants for resolving ties, described in the following section 3.6.

### 270 3.6 Majority Voting

271 Majority voting [23][35] is a popular label-fusion strategy used in ensemble-based  
 272 classifiers [26]. We use this for combining the independent patch-level predictions  
 273 into image-level results for our bark classification problem. In simple majority  
 274 voting, the class that gets the largest number of votes among all patches is con-  
 275 sidered the overall class label of the image. Let us assume that an image  $I$  can be  
 276 cropped into  $x$  parent patches and gets  $x_1, x_2, x_3$  number of patches classified  
 277 as the first, second and third classes. The final prediction class of the image is  
 278 taken as the class that has  $\max(x_1, x_2, x_3)$  votes, i.e. the majority voted class.

279 However, there may be cases when more than one class gets the largest num-  
 280 ber of votes. There is no one major class and ties can be found, i.e. multiple  
 281 classes can have the highest count of votes. In our study, we examine few tie-  
 282 breaking strategies from existing literature in majority voting [22][23][26][34][35].  
 283 The most common one is *Random Selection* [23] of one of the tied classes, all of  
 284 which are assumed to be equi-probable. Another trend of tie-breaking approaches  
 285 relies on class priors and we tested two variants of class prior probability in our  
 286 study. First, by *Class Proportions* strategy [34] that chooses the class having the  
 287 higher number of training samples among the tied classes. Second, using *Class*  
 288 *Accuracy* (given by F1-score), the tie goes in favor of the class having a better  
 289 F1-score. Outside of these standard methods, more particular to neural network  
 290 classifiers is the breaking of ties in ensembles by using Soft Max Accumulations  
 291 [22]. Neural network classifiers output, by default, the class label prediction ac-  
 292 companied by a confidence of prediction (by using soft max function) and this  
 293 information is leveraged to resolve ties in [22]. In case of a tie in the voting pro-  
 294 cess, the confidences for the received votes of the tied classes, are summed up.  
 295 Finally, the class that accumulates the *Maximum Confidence sum* is selected.  
 296

## 297 4 EXPERIMENTS

### 298 4.1 Dataset

299 **Pre-processing** As we used pre-trained models for fine-tuning, we resized and  
 300 normalised all our images to the same format the network was originally trained  
 301 on, i.e. ImageNet standards. For patch-based CNN experiments, no size transfor-  
 302 mations were done during training as the patches were already of size 224x224.  
 303 For the benchmark experiments with whole images, the images were randomly  
 304 re-sized and cropped to the size of 224x224 during image-loading. For data aug-  
 305 mentation purposes, torchvision transforms were used to obtain random hori-  
 306 zontal flips, rotations and color jittering, on all training images.

307 **Patch details** Non-overlapping patches of size 224x224 were extracted in a  
 308 sliding window manner from non-scaled original and upscaled Bark-101 images  
 309 (upscale factor of 4). Bark-101 originally has 1292 training images and 1295 test  
 310 images. After patch-extraction by the two methods, we obtain a higher count of  
 311 samples as shown in table 1. In our study, 25% of the training data was kept for  
 validation.

**Table 1.** Count of extracted unique patches from Bark-101.

Source Image	Train Validation Test		
Non-Scaled Bark-101	3156	1051	4164
Upscaled Bark-101	74799	24932	99107

312

### 313 4.2 Training details

314 We used CNNs that have been pre-trained on ImageNet. Three architectures  
 315 were selected - Squeezenet, MobileNetV2 and VGG16. We used Pytorch to fine-  
 316 tune these networks with the Bark-101 dataset, starting with an initial learning  
 317 rate of 0.001. Training was performed over 50 epochs with the Stochastic gradi-  
 318 ent descent (SGD) optimizer, reducing the learning rate by a factor of 0.1 every  
 319 7th epoch.

320

321 ESPCN was trained from scratch for a factor of 4, on the 1292 original training  
 322 images of Bark-101, with a learning rate of 0.001 for 30 epochs.

## 323 5 RESULTS AND DISCUSSIONS

324 We present our findings with the two kinds of accuracy described in section 3.5.  
 325 Compared to absolute accuracy, patch-level accuracy provides a more precise

326 measure of how good the classifier model is. However, for our study, it is the  
 327 absolute accuracy that is of greater importance as the final objective is to improve  
 328 identification of bark species. It is important to note that Bark-101 is a challeng-  
 329 ing dataset and the highest accuracy obtained in the original work on Bark-101  
 330 [25] was 41.9% using Late Statistics on LBP-like filters and SVM classifiers. In  
 331 our study, we note this as a benchmark value for comparing our performance  
 332 using CNNs.

### 333 5.1 Using whole images

334 We begin by comparing the classification accuracy of different pre-trained CNNs  
 335 fine-tuned with the non-scaled original Bark-101 data. Whole images were used  
 336 and no explicit patches were formed prior to training. Thus, training was carried  
 out on 1292 images and testing on 1295. Table 2 presents the results.

**Table 2.** Classification of whole images from Bark-101.

CNN	Absolute accuracy
Squeezenet	<b>43.7%</b>
VGG16	42.3%
MobilenetV2	34.2%

337

### 338 5.2 Using patches

339 In this section, we compare patch-based CNN classification using patches ob-  
 340 tained by the image re-scaling methods explained in section 3.3.

341

342 In the following tables (3, 4 and 5), the column for *Patch-level accuracy* gives  
 343 the local performance, i.e how many of the test patches are correctly classified.  
 344 This number of test patches vary across the two methods - patches from non-  
 345 scaled original and those from upscaled Bark-101 (see table 1). For *Absolute*  
 346 *accuracy*, we calculate how many of the original 1295 Bark-101 test images are  
 347 correctly identified, by majority voting (with 4 tie-breaking strategies) on patch-  
 348 level predictions. Column *Random selection* gives results of arbitrarily breaking  
 349 ties by randomly selecting one of the tied classes (averaged over 5 trials). In *Max*  
 350 *confidence* column, the tied class having the highest soft-max accumulations is  
 351 selected. The last two columns use class priors for tie-breaking. *Class proportions*  
 352 selects the tied class that appears most frequently among the training samples  
 353 (i.e. having highest proportion) while *Class F1-scores* resolves ties by select-  
 354 ing the tied class which has higher prediction accuracy (metric chosen here is  
 355 F1-score). The best absolute accuracy among different tie-breaking methods for  
 356 each CNN model, is highlighted in bold.

357

358 **Patches from Non-Scaled Original Images** Patches of size 224x224 were  
 359 extracted from Bark-101 data, without any re-sizing of the original images. The  
 360 wide variation in the sizes of Bark-101 images resulted in a minimum of zero  
 361 and a maximum of 9 patches possible per image. We kept all possible crops, re-  
 362 sulting in a total of 3156 train, 1051 validation and 4164 test patches. Although  
 363 the number of training samples is higher than that for training with whole im-  
 364 ages (section 5.1), only a total of 1169 original training images (belonging to 96  
 365 classes) allowed at least one square patch of size 224x224. Thus, there is data  
 366 loss as the images from which not even a single patch could be extracted were  
 excluded. Results obtained by this strategy are listed in Table 3.

**Table 3.** Classification of patches from non-scaled original Bark-101.

CNN model	Patch-level accuracy(%)	Absolute accuracy(%) by Majority Voting			
		Random selection	Max confidence	Class pro- portions	Class F1- scores
Squeezenet	47.84	43.47	<b>44.09</b>	43.17	43.63
VGG16	47.48	44.40	<b>45.25</b>	44.32	44.09
MobilenetV2	41.83	37.61	<b>38.69</b>	36.68	37.22

367

368

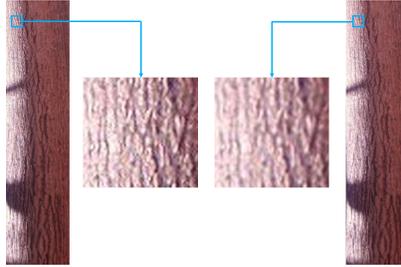
369 We observe that the patch-level accuracy is higher than absolute accuracy,  
 370 which can possibly be explained due to the data-loss. From the original test data,  
 371 only 1168 images (belonging to 96 classes) had dimensions that allowed at least  
 372 one single patch of 224x224, resulting in a total of 4164 test patches. Around  
 373 127 test images were excluded and by default, classified as incorrect, therefore  
 374 reducing absolute accuracy. For patch-level accuracy, we reported how many of  
 375 the 4164 test patches were correctly classified.

376

377 **Patches from Upscaled Images** The previous sub-section highlights the need  
 378 for upscaling original images, so that none is excluded from patch-extraction.  
 379 Here, we first upscaled all the original Bark-101 images by a factor of 4 and then  
 380 extracted square patches of size 224x224 from them. Figure 4 shows an example  
 381 pair of upscaled images and their corresponding patch samples.

382

383 We observe that among all our experiments, better absolute accuracy is ob-  
 384 tained when patch-based CNN classification is performed on upscaled Bark-101  
 385 images and shows comparable performance between both methods of upscaling  
 386 (bicubic or ESPCN). The best classifier performance in our study is **57.22%**  
 387 from VGG16 fine-tuned by patches from Bark-101 upscaled by bicubic interpo-  
 388 lation (table 4). This is a promising improvement from both the original work  
 389 [25] on Bark-101 (best accuracy of 41.9%) as well as the experiments using whole



**Fig. 4.** An example pair of upscaled images and sample patches from them. The left-most image has been upscaled by ESPCN and the right-most one by bicubic interpolation. Between them, sample extracted patches are shown where the differences between the two methods of upscaling become visible.

**Table 4.** Classification of patches from Bark-101 upscaled by bicubic interpolation.

CNN model	Patch-level accuracy(%)	Absolute accuracy(%) by Majority Voting			
		Random	Max con-fidence	Class pro-portions	Class F1-scores
Squeezenet	35.69	48.32	<b>48.57</b>	48.11	48.19
VGG16	41.04	57.21	56.99	<b>57.22</b>	57.14
MobilenetV2	33.36	43.73	43.60	<b>43.83</b>	43.60

**Table 5.** Classification of patches from Bark-101 upscaled by ESPCN.

CNN model	Patch-level accuracy(%)	Absolute accuracy(%) by Majority Voting			
		Random	Max con-fidence	Class pro-portions	Class F1-scores
Squeezenet	34.85	49.40	49.38	<b>49.45</b>	49.38
VGG16	39.27	<b>55.86</b>	55.75	55.76	55.75
MobilenetV2	32.12	<b>42.19</b>	41.78	41.93	41.85

390 images (best accuracy of 43.7% by Squeezenet, from table 2).

391

392

393

394

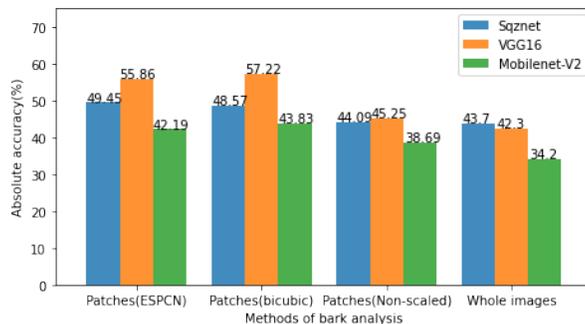
395

396

397

398

The comparison of tie-majority strategies shows that the differences are not substantial. This is because the variations can only be visible when many ties are encountered, which was not always the case for us. Table 6 lists the count of test images (whole) where ties were encountered. We observe that test images in the patch method with non-scaled original Bark-101, encounter 4-5 times more ties than when using upscaled images (bicubic or EPSCN). Our study thus corroborates that the differences among tie-breaking strategies are more considerable



**Fig. 5.** Comparison of Bark-101 classification accuracy (absolute) using CNNs in this work. Best accuracy of 41.9% was obtained in the original work [25] using Late Statistics on LBP-like filters and SVM classifiers.

399 when several ties occur (table 3), than when fewer ties are found (tables 4 and 5).  
 400 However, since the total number of test images in Bark-101 is 1295, the overall  
 401 count of ties can still be considered low in our study. Nevertheless, we decided  
 402 to include this comparison to demonstrate the difficulties of encountering ties  
 403 in majority voting for patch-based CNN and investigate existing strategies to  
 404 overcome this. It is interesting to observe (in table 3) that for patches extracted  
 405 from non-scaled original Bark-101 (where there is a higher number of ties), the  
 406 best tie-breaking strategy is the *maximum confidence sum*, as affirmed in [22]  
 407 where the authors had tested it on simpler datasets (having a maximum of 26  
 classes in the *Letter* dataset) taken from the UCI repository [14].

**Table 6.** Count of test images showing tied classes in majority voting.

Patch Method	Squeezenet	VGG16	MobileNetV2
Non-Scaled Original	217	283	274
Upscaled by Bicubic	52	45	45
Upscaled by ESPCN	50	46	63

408  
 409 To summarise, we present few important insights. First, when the total count of  
 410 training samples is low, patch-based image analysis can improve accuracy due  
 411 to better learning of local information and also since the total count of training  
 412 samples increases. Second, image re-scaling invariably introduces distortion and  
 413 reduces the image quality, hence patches from upscaled images have a loss of fea-  
 414 ture information. As expected, patch-level accuracy is lower when using patches  
 415 from upscaled images (tables 4 and 5), compared to that of patches from non-  
 416 scaled original images having more intact features (table 3). However, we also

417 observe that absolute accuracy falls sharply for patches taken from non-scaled  
418 original Bark-101. This is because several of the original images have such low  
419 image dimensions, that no patch formation was possible at all. Therefore, all  
420 such images (belonging to 5 classes, see section 5.2 for details) were by default  
421 excluded from our consideration, resulting in low absolute accuracy across all  
422 the CNN models tested. Thus, we infer that for datasets having high diversity  
423 and variation of image dimensions, upscaling before patch-extraction can ensure  
424 better retention and representation of data. Finally, we also observe that it is  
425 useful to examine tie-breaking strategies in majority voting compared to rely-  
426 ing on simple random selection. These strategies are particularly significant if a  
427 considerable number of ties are encountered.

## 428 6 CONCLUSION AND FUTURE WORK

429 Our study demonstrates the potential of using deep learning for studying chal-  
430 lenging datasets such as Bark-101. For a long time, bark recognition has been  
431 treated as a texture classification problem and traditionally solved using hand-  
432 crafted features and statistical analysis. A patch-based CNN classification ap-  
433 proach can automate bark recognition greatly and reduce the efforts required  
434 by time-consuming traditional methods. Our study shows its effectiveness by  
435 outperforming accuracy on Bark-101 from traditional methods. An objective  
436 of our work was also to incorporate current trends in image re-scaling and  
437 ensemble-based classifiers in this bark analysis, to broaden perspectives in the  
438 plant vision community. Thus, we presented recent approaches in re-scaling by  
439 super-resolution networks and several tie-breaking strategies for majority voting  
440 and demonstrated their impact on performance. Super-resolution networks have  
441 promising characteristics to counter-balance the degradation introduced due to  
442 re-scaling. Although for our study with texture data as bark, its performance  
443 was comparable to traditional bicubic interpolation, we hope to investigate its  
444 effects on other plant data in future works. It would also be interesting to de-  
445 rive inspiration from patch-based image analysis in medical image segmentation  
446 where new label fusion methods are explored to integrate location information of  
447 patches for image-level decisions. In future works, we intend to accumulate new  
448 state-of-art methods and extend the proposed methodology to other plant or-  
449 gans and develop a multi-modal plant recognition tool for effectively identifying  
450 tree and shrub species. We will also examine its feasibility on mobile platforms,  
451 such as smart-phones, for use in real-world conditions.

## 452 ACKNOWLEDGEMENTS

453 This work has been conducted under the framework of the ReVeRIES project  
454 (Reconnaissance de Végétaux Récréative, Interactive et Educative sur Smart-  
455 phone) supported by the French National Agency for Research with the reference  
456 ANR15-CE38-004-01.

457 **References**

- 458 1. Affouard, A., Goëau, H., Bonnet, P., Lombardo, J.C., Joly, A.: Pl@ntNet app in  
459 the era of deep learning. In: ICLR: International Conference on Learning Repre-  
460 sentations. Toulon, France (Apr 2017)
- 461 2. Barnea, E., Mairon, R., Ben-Shahar, O.: Colour-agnostic shape-based 3d fruit de-  
462 tection for crop harvesting robots. *Biosystems Engineering* **146**, 57–70 (2016)
- 463 3. Begue, A., Kowlessur, V., Singh, U., Mahomoodally, F., Pudaruth, S.: Automatic  
464 recognition of medicinal plants using machine learning techniques. *International*  
465 *Journal of Advanced Computer Science and Applications* **8**(4), 166–175 (2017)
- 466 4. Bertrand, S., Ameer, R.B., Cerutti, G., Coquin, D., Valet, L., Tougne, L.: Bark  
467 and leaf fusion systems to improve automatic tree species recognition. *Ecological*  
468 *Informatics* **46**, 57–73 (2018)
- 469 5. Bertrand, S., Cerutti, G., Tougne, L.: Bark Recognition to Improve Leaf-based  
470 Classification in Didactic Tree Species Identification. In: VISAPP 2017 - 12th  
471 International Conference on Computer Vision Theory and Applications. Porto,  
472 Portugal (Feb 2017)
- 473 6. Boudra, S., Yahiaoui, I., Behloul, A.: A comparison of multi-scale local binary pat-  
474 tern variants for bark image retrieval. In: Battiato, S., Blanc-Talon, J., Gallo, G.,  
475 Philips, W., Popescu, D., Scheunders, P. (eds.) *Advanced Concepts for Intelligent*  
476 *Vision Systems*. pp. 764–775. Springer International Publishing, Cham (2015)
- 477 7. Boudra, S., Yahiaoui, I., Behloul, A.: Plant identification from bark: A texture  
478 description based on statistical macro binary pattern. In: 2018 24th International  
479 Conference on Pattern Recognition (ICPR). pp. 1530–1535. IEEE (2018)
- 480 8. Carpentier, M., Giguère, P., Gaudreault, J.: Tree species identification from bark  
481 images using convolutional neural networks. In: 2018 IEEE/RSJ International Con-  
482 ference on Intelligent Robots and Systems (IROS). pp. 1075–1081. IEEE (2018)
- 483 9. Cerutti, G., Tougne, L., Sacca, C., Joliveau, T., Mazagol, P.O., Coquin, D., Vaca-  
484 vant, A.: Late Information Fusion for Multi-modality Plant Species Identification.  
485 In: *Conference and Labs of the Evaluation Forum*. p. Working Notes. Valencia,  
486 Spain (Sep 2013)
- 487 10. De Boor, C.: Bicubic spline interpolation. *Journal of mathematics and physics*  
488 **41**(1-4), 212–218 (1962)
- 489 11. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-  
490 scale hierarchical image database. In: 2009 IEEE conference on computer vision  
491 and pattern recognition. pp. 248–255 (2009)
- 492 12. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for  
493 image super-resolution. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.)  
494 *Computer Vision – ECCV 2014*. pp. 184–199. Springer International Publishing,  
495 Cham (2014)
- 496 13. Fiel, S., Sablatnig, R.: Automated identification of tree species from images of the  
497 bark, leaves or needles. In: 16th Computer Vision Winter Workshop. Mitterberg,  
498 Austria (Feb 2011)
- 499 14. Frank, A.: Uci machine learning repository. <http://archive.ics.uci.edu/ml> (2010)
- 500 15. Ganschow, L., Thiele, T., Deckers, N., Reulke, R.: Classification of tree species  
501 on the basis of tree bark texture. *International Archives of the Photogrammetry,*  
502 *Remote Sensing and Spatial Information Sciences-ISPRS Archives* **42**(W13) (2019)
- 503 16. Goëau, H., Joly, A., Bonnet, P., Selmi, S., Molino, J.F., Barthélémy, D., Boujemaa,  
504 N.: LifeCLEF Plant Identification Task 2014. In: Cappellato, L., Ferro, N., Halvey,  
505 M., Kraaij, W. (eds.) *CLEF: Conference and Labs of the Evaluation Forum*. vol.  
506 *CEUR Workshop Proceedings*, pp. 598–615. Sheffield, United Kingdom (Sep 2014)

- 507 17. Hemming, J., Rath, T.: Pa—precision agriculture: computer-vision-based weed  
508 identification under field conditions using controlled lighting. *Journal of agricul-*  
509 *tural engineering research* **78**(3), 233–243 (2001)
- 510 18. Huang, Z.K., Huang, D.S., Du, J.X., Quan, Z.H., Guo, S.B.: Bark classification  
511 based on gabor filter features using rbpnn neural network. In: *International con-*  
512 *ference on neural information processing*. pp. 80–87. Springer (2006)
- 513 19. Huang, Z.K., Huang, D.S., Du, J.X., Quan, Z.H., Guo, S.B.: Bark classification  
514 based on gabor filter features using rbpnn neural network. In: *International con-*  
515 *ference on neural information processing*. pp. 80–87. Springer (2006)
- 516 20. Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K.:  
517 Squeezenet: Alexnet-level accuracy with 50x fewer parameters and j 0.5 mb model  
518 size. arXiv preprint arXiv:1602.07360 (2016)
- 519 21. ImageCLEF: Plantclef 2017 (accessed on 2020-04-15), [https://www.imageclef.](https://www.imageclef.org/lifeclef/2017/plant)  
520 [org/lifeclef/2017/plant](https://www.imageclef.org/lifeclef/2017/plant)
- 521 22. Kokkinos, Y., Margaritis, K.G.: Breaking ties of plurality voting in ensembles of  
522 distributed neural network classifiers using soft max accumulations. In: *IFIP In-*  
523 *ternational Conference on Artificial Intelligence Applications and Innovations*. pp.  
524 20–28. Springer (2014)
- 525 23. Malmasi, S., Dras, M.: Native language identification using stacked generalization.  
526 arXiv preprint arXiv:1703.06541 (2017)
- 527 24. Mizoguchi, T., Ishii, A., Nakamura, H., Inoue, T., Takamatsu, H.: Lidar-based indi-  
528 vidual tree species classification using convolutional neural network. In: *Videomet-*  
529 *rics, Range Imaging, and Applications XIV*. vol. 10332, p. 103320O. International  
530 Society for Optics and Photonics (2017)
- 531 25. Ratajczak, R., Bertrand, S., Crispim-Junior, C.F., Tougne, L.: Efficient Bark  
532 Recognition in the Wild. In: *International Conference on Computer Vision Theory*  
533 *and Applications (VISAPP 2019)*. Prague, Czech Republic (Feb 2019)
- 534 26. Rokach, L.: Ensemble-based classifiers. *Artificial Intelligence Review* **33**(1-2), 1–39  
535 (2010)
- 536 27. Sa Junior, J.J.d.M., Backes, A.R., Rossatto, D.R., Kolb, R.M., Bruno, O.M.: Mea-  
537 suring and analyzing color and texture information in anatomical leaf cross sec-  
538 tions: an approach using computer vision to aid plant species identification. *Botany*  
539 **89**(7), 467–479 (2011)
- 540 28. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: In-  
541 verted residuals and linear bottlenecks. In: *Proceedings of the IEEE conference on*  
542 *computer vision and pattern recognition*. pp. 4510–4520 (2018)
- 543 29. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert,  
544 D., Wang, Z.: Real-time single image and video super-resolution using an efficient  
545 sub-pixel convolutional neural network. In: *Proceedings of the IEEE conference on*  
546 *computer vision and pattern recognition*. pp. 1874–1883 (2016)
- 547 30. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale  
548 image recognition. In: Bengio, Y., LeCun, Y. (eds.) *3rd International Conference*  
549 *on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015,*  
550 *Conference Track Proceedings* (2015)
- 551 31. Smolyanskiy, N., Kamenev, A., Smith, J., Birchfield, S.: Toward low-flying au-  
552 tonomous mav trail navigation using deep neural networks for environmental  
553 awareness. In: *2017 IEEE/RSJ International Conference on Intelligent Robots and*  
554 *Systems (IROS)*. pp. 4241–4247. IEEE (2017)
- 555 32. Tian, H., Wang, T., Liu, Y., Qiao, X., Li, Y.: Computer vision technology in  
556 agricultural automation—a review. *Information Processing in Agriculture* **7**(1),  
557 1–19 (2020)

- 558 33. Wan, Y.Y., Du, J.X., Huang, D.S., Chi, Z., Cheung, Y.M., Wang, X.F., Zhang,  
559 G.J.: Bark texture feature extraction based on statistical texture analysis. In: Pro-  
560 ceedings of 2004 International Symposium on Intelligent Multimedia, Video and  
561 Speech Processing, 2004. pp. 482–485. IEEE (2004)
- 562 34. Woods, K., Kegelmeyer, W.P., Bowyer, K.: Combination of multiple classifiers  
563 using local accuracy estimates. *IEEE transactions on pattern analysis and machine*  
564 *intelligence* **19**(4), 405–410 (1997)
- 565 35. Xu, L., Krzyzak, A., Suen, C.Y.: Methods of combining multiple classifiers and  
566 their applications to handwriting recognition. *IEEE transactions on systems, man,*  
567 *and cybernetics* **22**(3), 418–435 (1992)