Patch-based CNN evaluation for bark classification

Debaleena Misra, Carlos F Crispim-Junior, Laure Tougne

Patch-based CNN evaluation for bark classification

Debaleena Misra[1], Carlos Crispim-Junior[1][0000−0002−5577−5335], and Laure Tougne[1][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].

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

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

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

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

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

2 RELATED WORK

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

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

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

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

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

3 METHODOLOGY

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

3.1 Dataset

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

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

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

3.2 Patch preparation

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

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

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

3.3 Image re-scaling

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

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

Super-resolution using sub-pixel convolutional neural network Supervised machine learning methods can learn mapping functions from low-resolution images to their high-resolution representations. Recent advances in deep neural networks called Super-resolution CNN (SRCNN) [12] have shown promising improvements in terms of computational performances and reconstruction accuracy. These models are trained with low-resolution images as inputs and their high-resolution counterparts are the targets. The first convolutional layers of such neural networks perform feature extraction from the low-resolution images. The next layer maps these feature maps non-linearly to their corresponding high-resolution patches. The final layer combines the predictions within a spatial neighbourhood to produce the final high-resolution image.

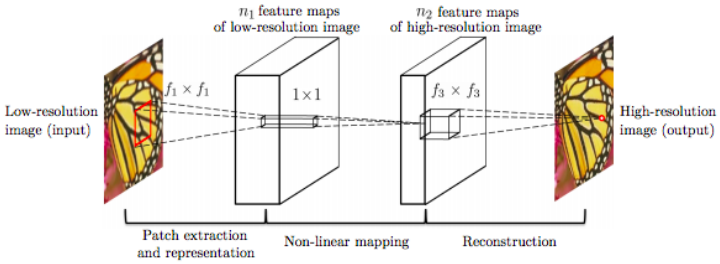

Fig. 2. SRCNN architecture [12].

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

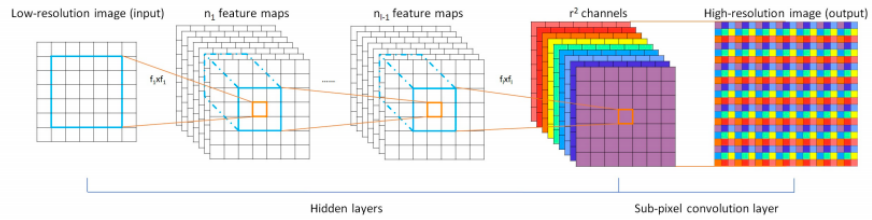

Fig. 3. Efficient sub-pixel convolutional neural network (ESPCN) [29].

3.4 CNN classification

The CNN models used for image recognition in this work are 3 recent architectures: Squeezenet [20], MobileNetV2 [28] and VGG16 [30]. Since Bark-101 is a small dataset, we apply transfer-learning and use the pre-trained weights of the models trained on the large-scale image data of ImageNet [11]. The convolutional 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

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

3.5 Evaluation metrics

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

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

3.6 Majority Voting

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

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

4 EXPERIMENTS

4.1 Dataset

Pre-processing As we used pre-trained models for fine-tuning, we resized and normalised all our images to the same format the network was originally trained on, i.e. ImageNet standards. For patch-based CNN experiments, no size transformations were done during training as the patches were already of size 224x224. For the benchmark experiments with whole images, the images were randomly re-sized and cropped to the size of 224x224 during image-loading. For data augmentation purposes, torchvision transforms were used to obtain random horizontal flips, rotations and color jittering, on all training images.

Patch details Non-overlapping patches of size 224x224 were extracted in a sliding window manner from non-scaled original and upscaled Bark-101 images (upscale factor of 4). Bark-101 originally has 1292 training images and 1295 test images. After patch-extraction by the two methods, we obtain a higher count of 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

4.2 Training details

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

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

5 RESULTS AND DISCUSSIONS

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

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

5.1 Using whole images

We begin by comparing the classification accuracy of different pre-trained CNNs fine-tuned with the non-scaled original Bark-101 data. Whole images were used 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	**43.7**%
VGG16	42.3%
MobilenetV2	34.2%

5.2 Using patches

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

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

Patches from Non-Scaled Original Images Patches of size 224x224 were extracted from Bark-101 data, without any re-sizing of the original images. The wide variation in the sizes of Bark-101 images resulted in a minimum of zero and a maximum of 9 patches possible per image. We kept all possible crops, resulting in a total of 3156 train, 1051 validation and 4164 test patches. Although the number of training samples is higher than that for training with whole images (section 5.1), only a total of 1169 original training images (belonging to 96 classes) allowed at least one square patch of size 224x224. Thus, there is data 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 proportions	Class F1-scores
Squeezenet	47.84	43.47	**44.09**	43.17	43.63
VGG16	47.48	44.40	**45.25**	44.32	44.09
MobilenetV2	41.83	37.61	**38.69**	36.68	37.22

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

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

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

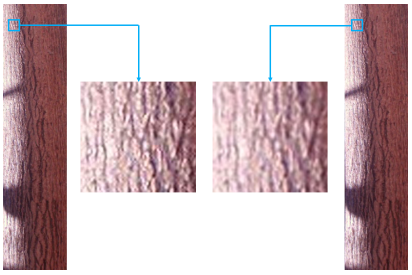

Fig. 4. An example pair of upscaled images and sample patches from them. The left-most image has been upscaled by ESPCN and the right-most one by bicubic interpolation. Between them, sample extracted patches are shown where the differences between the two methods of upscaling become visible.

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

CNN model	Patch-level accuracy(%)	Absolute accuracy(%) by Majority Voting			
		Random	Max confidence	Class proportions	Class F1-scores
Squeezenet	35.69	48.32	**48.57**	48.11	48.19
VGG16	41.04	57.21	56.99	**57.22**	57.14
MobilenetV2	33.36	43.73	43.60	**43.83**	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 confidence	Class proportions	Class F1-scores
Squeezenet	34.85	49.40	49.38	**49.45**	49.38
VGG16	39.27	**55.86**	55.75	55.76	55.75
MobilenetV2	32.12	**42.19**	41.78	41.93	41.85

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

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

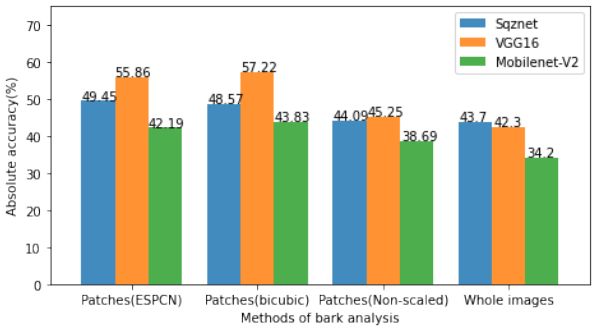

Fig. 5. Comparison of Bark-101 classification accuracy (absolute) using CNNs in this work. Best accuracy of 41.9% was obtained in the original work [25] using Late Statistics on LBP-like filters and SVM classifiers.

when several ties occur (table 3), than when fewer ties are found (tables 4 and 5). However, since the total number of test images in Bark-101 is 1295, the overall count of ties can still be considered low in our study. Nevertheless, we decided to include this comparison to demonstrate the difficulties of encountering ties in majority voting for patch-based CNN and investigate existing strategies to overcome this. It is interesting to observe (in table 3) that for patches extracted from non-scaled original Bark-101 (where there is a higher number of ties), the best tie-breaking strategy is the *maximum confidence sum*, as affirmed in [22] 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

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

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

6 CONCLUSION AND FUTURE WORK

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

ACKNOWLEDGEMENTS

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

References

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

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