2017 30th SIBGRAPI Conference on Graphics, Patterns and Images Tutorials

A Review of Texture Classification Methods and Databases

Paulo Cavalin
IBM Research
International Business Machines (IBM)
Rio de Janeiro, RJ, Brazil
Email: pcavalin@br.ibm.com

Luiz S. Oliveira
Departamento de Informática (DInf)
Universidade Federal do Paraná (UFPR)
Curitiba, PR, Brazil
Email: lesoliveira@inf.ufpr.br

***Abstract*—In this survey, we present a review of methods and resources for texture recognition, presenting the most common techniques that have been used in the recent decades, along with current tendencies. That said, this paper covers since the most traditional approaches, for instance texture descriptors such as gray-level co-occurence matrices (GLCM) and Local Binary Patterns (LBP), to more recent approaches such as Convolutional Neural Networks (CNN) and multi-scale patch-based recognition based on encoding approaches such as Fisher Vectors. In addition, we point out relevant references for benchmark datasets, which can help the reader develop and evaluate new methods.**

***Keywords*-Texture recognition, Image recognition, Deep Learning**

I. Introduction

Texture classification consists of an image processing and computer vision task, which can be applied to numerous fields and industries such as computer-aided medical diagnosis [1], [2], [3], [4], classification of forest species [5], [6], [7], [8], geo-processing [9] writer identification and verification [10], oil and gas [11], [12], [13], [14], and agriculture [15], [16], [17]. A definition of textures has been presented in [18], where the authors state that a textured image or a textured area in the image can be characterized by a non-uniform or varying spatial distribution of intensity or color. As a result, even though the texture classification task presents some similarities to object classification, such as the strong correlation of pixel intensities in the 2D space, and some approaches for image recognition can be applied to both, some unique features of textures, such as the ability to perform the classification using only a relatively small fragment of a texture, different approaches have emerged over the years in order to deal with this problem more efficiently.

For a better illustration of the problem, we present samples of objects in Figure 1, and examples of textures in Figure 2. Note that for objects, shape may be the most important feature to differ one object from another. For instance, the shape of the chair in Figure 1(a) is totally different from that of the airplane in Figure 1(b). In addition, the entire object (or at least most of it) should appear in the image for the appropriate identification of such. For texture recognition, on the other hand, the way the pixels are arranged tend to indicate the class to which the image belongs. The pixels in the chequered texture presented in Figure 2(a) are disposed in a completely different way from those of the sample of a marbled texture in Figure 2(b). And the pixels in Figure 2(c) and Figure 2(d) are also organized in ways that differ from the other two textures. For this reason, texture recognition generally depends more on methods that capture patterns of such arrangements, instead of detecting salient points as in object recognition. Furthermore, given that textures are represented by these patterns of arrangements of pixels, such patterns can be captured from smaller portions of the images. As illustrated in Figure 3(a) and Figure 3(b), small patches may represent the same pattern of texture as the entire image. This feature has resulted in the proposal of different methods that are specific for texture classification, aiming at both reducing cost of classification and increasing the perfomance of such systems.

((a)) Chair

((b)) Airplane

Fig. 1. Samples of objects

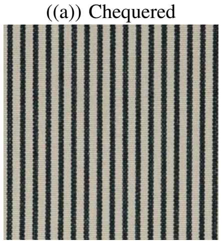
Given these standpoints, in this paper we present a review of the most relevant techniques for texture classification investigated in the recent decades. The idea is to provide to the reader an understanding of the most common methods for this task, covering from the more standard techniques to more recent tendencies. In addition, we present relevant references that cover the most relevant publicly available datasets, which can be used to benchmark systems.

The remainder of this paper is organized as follows. We start by presenting, in Section II, some of the most relevant texture descriptors that have been used as feature set, such as

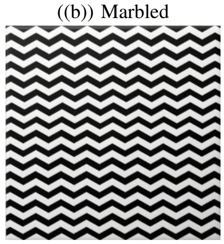
2474-0705 /17 $31.00 © 2017 IEEE
DOI 10.1109/SIBGRAPI-T.2017.10

((a)) Chequered

((b)) Marbled

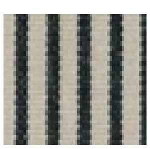


((c)) Lined



((d)) Zigzagged

Fig. 2. Samples of Textures



((a)) Lined

((b)) Zigzagged

Fig. 3. Samples of patches cropped from the lined and zigzagged textures

Gray-level Co-occurence Matrices (GLCM) and Local Binary Patterns (LBP). Next, in SectionIII, we discuss patch-based methods, which exploit one the fundamental properties of textures: the multiplicity of the similar patterns over the image or a piece of image. Then, methods that have been employing deep learning, specially Convolutional Neural Networks (CNNs), are examined in Section IV. In Section V, we provide relevant references related to publicly available benchmark texture datasets. Finally, in Section VI, we deliberate about current and future trends and applications of the area.

II. Texture descriptors

Although deep learning approaches have become more and more popular in numerous image recognition application in the past decade, the use of visual descriptors is still of great importance in this field, mainly when there is not sufficient data or computing resources to train a complex model such as a CNN. In a recent study, it has been demonstrated that the use of more traditional texture descriptors such as LBP can be better than CNNs in different scenarios [19], showing that such descriptors can be useful depending on the application. For this reason, in this section we cover in greater detail some texture descriptors that we judge as the most relevant in the literature. For the sake of completeness, a texture descriptor consists of encoding a given image or piece of image into a N-dimensional feature vector, aiming at capturing the main properties of the texture contained in the image [20]

A. Gray-Level Co-occurrence Matrices (GLCM)

This is one of the most well-known texture descriptors in the literature [9], [20], [21], [22]. It consists of computing statistical experiments on the matrix (or matrices) containing the co-occurrences of the pixel intensities at given angles and distance. Such statistics experiments intuitively provide measures of properties such as smoothness, coarseness, and regularity, for instance, on the distribution of pixels on the texture.

By definition, a GLCM is the probability of the joint occurrence of gray-levels i and j, where $i \leq G$ and $j \leq G$ and G denoted the gray-level depth of the image, within a defined spatial relation in an image. That spatial relation is defined in terms of a distance D and an angle θ. From this θ and D-dependent GLCM, statistical information can be extracted to define a feature vector. That is, assuming that $M'(i,j,\theta,d)$ is the normalized co-occurrence of gray-level i and j observed for consecutive pixels at distance D and angle θ, we can use a GLCM to describe texture by computing a set of statistical measures from the GLCM, such as:

$$Contrast(\theta, D) = \sum_{i=1}^{G}\sum_{i=1}^{G}(i-j)^2 M'(i,j,\theta,D) \tag{1}$$

$$Energy(\theta, D) = \sum_{i=1}^{G}\sum_{i=1}^{G} M'(i,j,\theta,D)^2 \tag{2}$$

$$Entropy(\theta, D) = \sum_{i=1}^{G}\sum_{i=1}^{G} M'(i,j,\theta,D) \log M'(i,j,\theta,D). \tag{3}$$

More examples of such statistical measures, such as Correlation, Homogeneity, Maximum Likelihood, and 3rd Order Moment, can be found in [23]. In that paper, Haralick suggests a set of 14 statistical measures. Nonetheless, an optimum set of descriptors has proven to be most effective in many applications [21], [24]. For instance, in [21], different values of D and θ have been evaluated. The best setup found was the combination of $D = 5$ and $\theta = \{0, 45, 90, 135\}$. Considering also that a set of six statistical descriptors presented the best results, i.e. Energy, Contrast, Entropy, Homogeneity, Maximum Likelihood, and 3rd Order Momentum, a feature vector with 24 components has been used (1 distance times 4 angles times for measures).

B. Local Binary Patterns (LBP)

Along with GLCM, LBP is likely the most use texture descriptor, which first emerged in the 1990s. As stated in [25], at first LBP was introduced as a local contrast descriptor [26] and a further development of the texture spectra introduced in [27]. Shortly afterwards, it was shown to be interesting as a texture descriptor [28] and has become widely used since then [29], [18], [24], [30], [25].

In a more simplistic way, LBP works as follows. A histogram is computed with the distribution of the binary configurations of the pixels of the image, based on thresholding the surrounding window of each pixel when the intensity of the neighborhood pixel is below or above the center value. An illustration of this process is depicted in Figure 4.

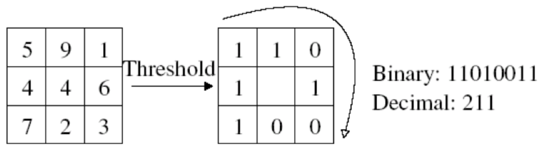


Fig. 4. The original LBP operator.

In the most simple implementation of LBP, 3×3 windows can be considered and a 256 bin histogram can be generated given the 2^8 possible combinations of the binary windows. This is equivalent to the implementation with $P = 8$ and $R = 1.0$ in Figure 5, i.e. 8 neighborhood pixels at distance equals to 1, which can be denoted as $\text{LBP}_{8,1}$ for the sake of simplicity (where $\text{LBP}_{P,R}$ represents the general notation). Nevertheless, the parameters P and R can be set to different values in accordance to the application, as depicted in the same figure. More details on the generalized LBP can be found in [29].

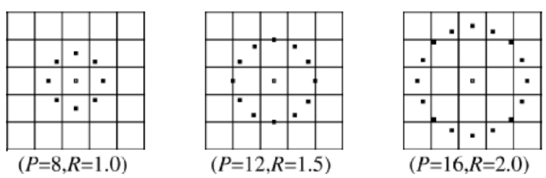


Fig. 5. Examples of different configurations for LBP.

Over the years, though, several variations of LBP have been proposed, with several purposes such different sampling techniques to effectively capture characteristics of certain features, or improving robustness to noise [31], [32], [25], [33].

Some variations focus on post-processing the original LBP, such as the Uniform LBP. In this case, only patterns that are considered as uniform are taken into account in the histogram, while the remaining ones account only for a single position of such. In greater detail, as stated in [29], uniform patterns, which can be represented by 58 binary patterns out of the original 256 set, might account for nearly 90% of all patterns in the neighborhood in texture images. A binary pattern is called uniform if it contains at most two 0/1 transitions when the binary string is considered circular. Given that uniform binary patterns can represent a reasonable quantity of information by a reduced set of binary patterns, the Uniform LBP is generally implemented by considering a 59-position normalized histogram, where the first 58 positions take into account the occurrence of an uniform pattern, and the 59-th position represents the occurrence of all other non-uniform ones. Following this trend, the Robust LBP (RLBP) consists of a variation of Uniform LBP [34]. By considering three-bit string of the neighbor pixels, the 010 and 101 sequences are considered as noisy and changed to 000 and 111, respectively.

Other variations work by changing the way the local binary patters are computed [33], [35], [36]. In the case of the Median Robust Extended LBP (MRLEBP) [36], instead of considering the raw image intensities, a local approach is used by making used of image medians. And LBP is extended to a multiscale descriptor, where the image medians are compared with a sampling scheme aiming at capturing both micro and macro structure in the textures. In a recent review, this method has show to be competitive even against CNNs [19] (discussed later in Section IV), which have become very popular in image recognition research in recent years. By comparing LBP-based systems against CNNs, on classification problems with different challenges, the best overall accuracy has been achieved by MRELBP. Nonetheless, on problems with textures with large appearance variation, CNNs have clearly outperformed the other methods.

A more complete review of the variations LBP can be found in [32], [19]).

C. Other texture descriptors

1) Gabor Filters Banks: Gabor filters banks have been widely used as texture descriptors [37], [38]. The main idea is to represent an image by taking into account both frequency and orientation aspects, by making use of a set of classes of Gabor functions. Such classes of functions can be generated from a main Gabor function [39].

The functions used by Gabor filters are complex and bi-dimensional sine functions, modeled also by a bi-dimensional Gaussian function. The frequency and orientation defined by the sine functions are the key component to describe the different types of texture that can appear in an image. Given that Gabor filters allow for varying a high set of parameters, such as frequency, orientation, eccentricity and symmetry, the set composed of the such combination of parameters are named Gabor Filter Banks [39]. Such characteristic is also one of the main drawbacks of the approach. Given that it may difficult to define the best set of parameters, filter bank design has emerged with method to define the optimal set of filter banks for a problem [38].

Basically, to use it in practice, this approach consists of transforming the input image by taking into account the entire set of filter banks. Suppose the total number of Gabor filters N_F, this will result also in a set with N_F transformed images, after applied the Gabor filter G_k, where $1 \geq k \leq N_F$, on the input image I. A feature vector can be then computed, for instance, by concatenating the mean and standard deviation the magnitude of each of the I'_k transformed images. For greater details, see [38].

2) Local Phase Quantization (LPQ): Proposed by Ojansivu et Heikkilä [40], LPQ is based on quantized phase information of the Discrete Fourier Transform (DFT). It uses the local phase information extracted using the 2-D DFT or, more precisely, a Short-Term Fourier Transform (STFT) computed over a rectangular $M \times M$ neighborhood N_x at each pixel position x of the image $i(x)$ defined by Equation 4:

$$F(u,x) = \sum_{y \in N_x} i(x-y)e^{-2\pi j u^T y} = w_u^T f_x \quad (4)$$

where w_u is the basis vector of the 2-D DFT at frequency u, and f_x is another vector containing all M^2 image samples from N_x.

The STFT can be implemented using a 2-D convolution $f(x)e^{-2\pi j u^T x}$ for all u. In LPQ only four complex coefficients are considered, corresponding to 2-D frequencies $u_1 = [a, 0]^T$, $u_2 = [0, a]^T$, $u_3 = [a, a]^T$, and $u_4 = [a, -a]^T$, where a is a scalar frequency below the first zero crossing of the DFT $H(u)$. $H(u)$ is DFT of the point spread function of the blur, and u is a vector of coordinates $[u, v]^T$. More details of the formal definition of LPQ can be found in [40], where those authors introduced all mathematical formalism.

At the end, we will have a 8-position resulting vector G_x for each pixel in the original image. These vectors G_x are quantized using a simple scalar quantizer (see Equation 5 and Equation 6), where g_j is the j-th components of G_x.

$$q_j = \begin{cases} 1, & \text{if } g_j \geq 0 \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

$$b = \sum_{j=1}^{8} q_j 2^{j-1} \quad (6)$$

The quantized coefficients are represented as integer values between 0-255 using binary coding (Equantion 6). These binary codes will be generated and accumulated in a 256-histogram, similar to the LBP method. The accumulated values in the histogram will be used as the LPQ 256-dimensional feature vector.

III. Patch- and Multiscale-based approaches

As we previously stated, textures present some properties that allowed researchers to proposed methods that exploit specific features of this problem in order to improve accuracy. One of these methods is patch-based classification. Given that an image may contain repetitions of the same texture pattern, instead of extracting features from the entire image as a whole, the idea is to divide the original image into several smaller images, i.e. the patches, where each patch can be considered as a different "observation" of the input image, and both train and perform recognition by considering these multiple "observations". For training, the training set can be increased by considering the multiple patches, so that the parameters of the classifier can be better estimated. And for the recognition phase, the final classification can be computed by considering multiple points of view or opinions pointed out by the resulting classification considering the multiple patches.

Before getting into the details of state-of-the-art patch-based methods, lets first define patches. Consider an image denoted $I_{W,H}$, where W denotes the width of the image and H its height. A patch consists of a sub-image denoted $I'_{W',H'}$ extracted from $I_{W,H}$, where $W' < W$ or $H' < H$. Considering also that a patch can be captured from any location in the image, a patch can be defined as $\psi_{(x,y)^0,W',H'}$, where $(x,y)^0$ correspond to the upper left corner of the patch and $W' - x = W$ and $H' - y = H$. By varying the values of $(x,y)^0$, W' and H', the set of patches $\Psi = \{\ldots, \psi_{(x,y)^0_k,W'_k,H'_k} \ldots\}$ can be created, where $(x,y)^0_k, W'_k, H'_k$ represents a different configuration for a patch.

Given the set of patches Ψ extracted from $I_{W,H}$, two types of systems can be built to conduct the texture classification: 1) multiple patches combined at classification level and 2) multiple patches combined at feature extraction level.

In the first type system, which is illustrated in Figure 6, patches are combined by the fusion of the classification outputs of the patches classified individually. That is, after the extraction of a texture descriptor has been done for a patch, the resulting feature vector is processed by classifier producing a patch-level classification result. The patch-level classifications of all patches in Ψ are combined by means of some classifier fusion approach, such as majority voting, sum rule, etc [41], and the final classification result for the original image is computed [7], [22], [42], [43], [44], [45], [4]. In essence, this approach in similar to an ensemble-based classification method. However, instead of making use of multiple classifiers, the classification is done by a combination of multiple classifications provided by the same classifier. The use of multiple classifiers has also been investigated, but in this case, a multiple classifier approach is combined at patch level, for instance, by combining multiple texture descriptors [22], [46].

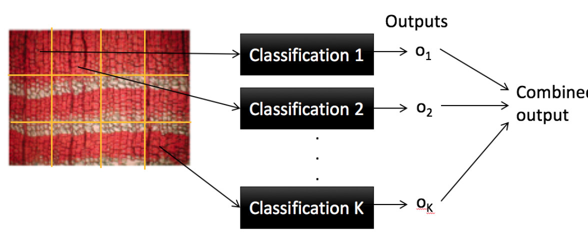


Fig. 6. Patches combined at classification level. The features extracted from each patch is processed by a classifier, and the combination of the classification of the individual patches is used for the final output.

In the second type of system, which is depicted in Figure 7, the patches are combined at feature extraction level. In this case, the feature extraction process is done similarly to the first type of system, i.e. one or more feature vectors are extracted from each patch. But, instead of classifying the feature vector of each patch individually, the feature vectors are combined by a multiple feature vector encoding method, such as bag of visual words [47], Fisher Vectors [48] or Dense Encoding [49]. Such encoding results in a single feature vector which is afterwards processed by a standard classification approach [47], [50], [48].

The extraction of patches can be done either without overlapping, in an method similar to quad-trees [22], or considering overlapping [43]. Furthermore, both single- and multi-scale

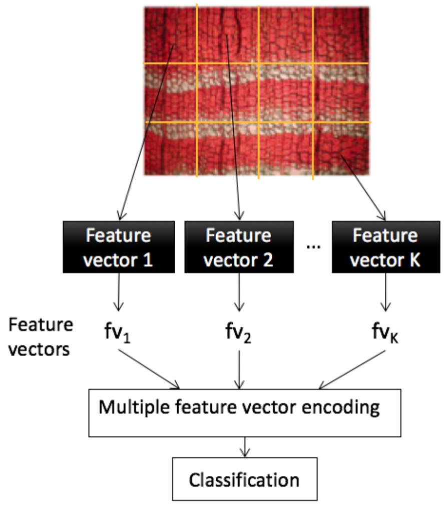


Fig. 7. Patches combined at feature extraction level. The features extracted from each patch are encoded into a single feature vector, which is the processed by a classifier for the final output.

approaches have been investigated [22], [44], [4], [51]. While in the former a single size for the patches is considered, in the latter, patches with different sizes can extracted and, analogously, can be combined at classification and feature extraction levels [46], [48].

In spite of the higher accuracy that patch-based methods can achieved, a negative drawback is the increase in cost for the recognition step. For example, in the case that the combination of is done at classification level, several classification steps must be performed, that is, at least one for each patch. As the number of patches increases, the cost for computing the classification increases as well. With this problem in mind, approaches to deal with the higher complexity of patch-based methods have been presented [44], [52]. In [44], the adaptive multiple feature vector framework has been proposed, the goal of which is to combine different multiple-patch system with different costs, to reduce the overall cost for a set of samples. That is, less difficult samples are recognized by less costly systems, while more difficult samples are recognized by more costly systems. The author demonstrated that such approach is able to reduce to 1/5 the overall cost of the most complex system, while retaining a comparable accuracy level.

IV. Deep learning approaches

With the prominent recent advances on deep learning, specially Convolution Neural Networks for image recognition [53], the application of CNNs on texture recognition problems has drawn the attention of various researchers. In some works, the focus has been mainly at evaluation standard CNN architectures that had been previously used for objected recognition [54] or combinations of CNNs with other classifiers such as Support Vector Machines (SVM) trained with textural descriptors [55].

In other front, though, approaches specifically tailored for textures have emerged [43], [56], [57], [58]. In our view, we observe that these approaches mainly differ in two aspects. The first is related to the way the classification approach is organized. In other words, we can find methods that present texture-specific CNN architectures that processed the whole image at once, and the parameters of such CNNs are learned end-to-end [56], [59], [55], [58]. And there are some approaches that basically consist of extensions of patch-based methods, but using standard CNN architectures originally designed for objected recognition [43], [45], [51], [60]. The second aspect in which the approaches may differ lies in the way the CNN is trained, which can either from scratch [43], [4], [61], [56], [57] or by making use of pre-trained models or transfer learning [45], [48], [51].

Some of the CNN architectures tailored for texture recognition are:

- Texture CNN (T-CNN) [56]: this architecture aims at designing learnable filter banks, which are embedded in the architecture of the CNN. The approach makes use of an energy layer. In this layer, the feature maps are pooled by averaging their activated output. This approach is considered to be similar to an energy response to a filter bank, since such pooling results in a single value for each feature map. Despite not presenting improvements in classification accuracy, this architecture has demonstrated to be less costly owing to the reduced number of parameters.
- Wavelet CNN [57]: this architecture consists of a combination of a more standard object-recognition CNN with multiresolution resolution analysis and an energy layer. The authors propose to formulate convolution and pooling in CNNs as filtering and downsampling, similar to previous works considering image analysis in the frequency domain. The authors have implemented this architecture with an energy layer based on Haar wavelets, whose parameters are optimized during the learning of the CNN. Although the proposed architecture resulted in better recognition performance than a standard VGG-16 CNN, it has not been able to outperform other approaches such as Fisher Vector-CNN (FV-CNN), discussed later in this section.
- Deep TEN [58]: in this approach, the authors have designed an architecture in which a patch-based feature extraction approach, by making use of an encoding similar to VLAD that is embedded in the CNN's architecture. Both the parameters of the VLAD encoding layer and the CNN filters are learned in an end-to-end fashion. The authors claim that the results outperforms the state-of-the-art, but no texture-tailored CNN method has been considered in such comparison, making it difficult to have a better understanding of the performance of such architecture.

Regarding CNNs that extend patch-based methods, one example is the method presented in [43]. The way the method works is straight-forward, and similar to previous works on patch-based classification with texture descriptors. In the training phase, which is depicted in Figure 8, a standard CNN model is trained on patches that are randomly extracted from the image, using a gradient descent algorithm. After the CNN has been trained, then the recognition of a texture image is carried out with the combination of the recognition results of the patches in the image, very similarly to the idea presented in Figure 6. The authors claim that the method has been originally proposed to cope with the high resolution of the input images.

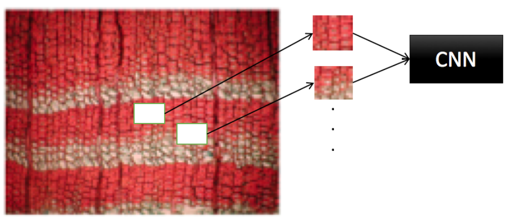


Fig. 8. CNN training with random patches. During each training iteration, a randomly-selected patch is cropped from an image, and used as input to train the CNN's parameters.

In the aforementioned approaches, the parameters of the CNNs are generally learned from scratch, i.e. the parameters are randomly or uniformly initialized and then trained with some algorithm such as gradient descent. Nevertheless, the use of pre-trained CNN models has become very popular in the recent years [62], [63]. This method consists of reusing a CNN trained on another dataset, commonly the ImageNet database [53], and using one of the top layers of the neural network as a feature extractor to train a new classifier applied on another dataset [48], [51]. The method FV-CNN presented in [48], which combined patch-based classification using the fisher vectors encoding method, on feature extracted with pre-trained CNNs, has been able to outperform state-of-the-art methods, such as standard CNNs and SIFT features. In [51], the comparison of pre-trained CNNs on patches combined at classification level has achieved competitive accuracies when compared with the method proposed in [43]. Related to these methods, the transfer learning evaluation presented in [45] has shown that a good feature extractor can be learned for textures recognition even if two datasets from different domains are used. That is, a CNN trained on the larger forest species recognition problem and transfered to a smaller materials recognition problem has achieved higher recognition that a CNN trained from scratch for the problem.

V. Datasets

Along with the methods that we previously described, many datasets have appeared over the years and contributed to developing the area. Many of these datasets have been firstly proposed to solve some application-specific problem and later became benchmarks for texture recognition systems [24], [42], [49]. More recently, though, datasets targeted specifically for texture recognition have been proposed [50].

A comprehensive work describing several datasets have been published recently [64], [65], in which the reader can find the most popular datasets used for benchmark (for instance, the Brodatz database). It is worth mentioning one interesting contribution of the work of [64] is the classification of texture databases into three distinct types: 1) Bio-medical textures, such as in the Computed-tomography emphysema database and the IICBU biological image repository; 2) Natural textures, for instance the Salzburg texture image database (STex) and the Brodatz dataset; and 3) Materials textures, as found in the KTH-TIPS and Kylberg texture datasets. In [65], the same classification is used for the texture dababases, but the authors complement that work with another set of texture datasets, where both papers together present a relevant compilation of existing databases.

A more recent database is the Describable Texture Dataset (DTD) [50]. The authors proposed the dataset to investigate the problem of texture description, i.e. some type of texture such as veined or marbled which can be presented by different types of materials, with textures captured in the wild, i.e. texture images extracted from the web. Not only such dataset can be used as an interesting benchmark for texture recognition systems, where current recognition rates are only at about 60 to 69% [48], [57], but also the texture descriptions can be used to train models to be later used as feature extractors. For instance, in [48], the classification output of a CNN trained on the dataset has been used to complement the feature extraction done with other texture descriptors. We believe that further investigation in this topic is a promising direction.

Another dataset that presents interesting features in the BreakHis dataset [49]. This dataset contains samples of malignant and benign breast cancer, which can also be classification into one of the classes of tumor. In addition, images have been captured at different magnification factors, to simulate the procedure that is generally followed by the human expert. This dataset differs from other texture recognition problems in one interesting aspect: just a small portion of the image is relavant for the classification, that is, only the region where the tumor appears matters for pointing out whether the tumor is benign or malignant. Approach based on selecting relevant patches, such as the method in [61], to improve the performance of the methods on datasets such as this one.

VI. Conclusion and Future Perspectives

Research in texture recognition has developed as a branch in image recognition. Different methods have been proposed to exploit the specific characteristic of textures, starting by the proposal of texture descriptors then moving to novel classification schemes. Among the texture descriptors, LBP and its variants tend to be most popular choice currently. And regarding classification schemes, patch-based classification seems to be a standard method in the area. Given that deep learning approaches are more recent, we can observe investigations in different directions. Nevertheless, the evaluation of CNNs in various benchmark databases has indicated that such neural networks are promising for textures to, and in the future might become a standard approach such as it has happened in object recognition.

Texture recognition methods can be used for many applications, such as industrial inspection, material selection, medical imaging, and thus forth. But some recent works have demonstrated that texture recognition methods can also be applied in problems where the notion of texture is not as clear, such as demonstrated in [10], [60]. In [10], texture recognition methods are used for author identification in handwritten letters. And in [60], texture classification methods are applied on acoustic spectrograms for music classification. We believe, thus, that other relations between might appear in the future, and texture recognition method will be applied to many other problems beyond the more common image recognition problems.

Finally, we observe that current efforts in making publicly available datasets is valuable to advance the area. Specific datasets such as DTD, that aim at defining some taxonomy for textures, might be combined with transfer learning and in the future may produce the same type of collaborative work in object recognition that has been defined with the ImageNet and pre-trained CNN models. With these trained models freely available on the web, researchers and developers are able to create systems quickly.

References

[1] H. Harms, U. Gunzer, and H. M. Aus, "Combined local color and texture analysis of stained cells," *Comput. Vision Graph. Image Process.*, vol. 33, no. 3, pp. 364–376, Mar. 1986. [Online]. Available: http://dx.doi.org/10.1016/0734-189X(86)90183-0

[2] A. Khademi and S. Krishnan, "Medical image texture analysis: A case study with small bowel, retinal and mammogram images," in *2008 Canadian Conference on Electrical and Computer Engineering*, May 2008, pp. 001 949–001 954.

[3] R. N. Sutton and E. L. Hall, "Texture measures for automatic classification of pulmonary disease," *IEEE Trans. Comput.*, vol. 21, no. 7, pp. 667–676, Jul. 1972. [Online]. Available: http://dx.doi.org/10.1109/T-C.1972.223572

[4] F. Spanhol, L. S. Oliveira, C. Petitjean, and L. Heutte, "Breast cancer histopathological image classification using convolutional neural network," in *Proc. of the International Joint Conference on Neural Networks (IJCNN 2016)*, Vancouver, Canada, 2016.

[5] J. Y. Tou, P. Y. Lau, and P. o. Y. H. Tay, "Computer vision-based wood recognition system," in *Int. Workshop on Advanced Image Technology*, 2007.

[6] J. Y. Tou, Y. H. Tay, and P. Y. Lau, "One-dimensional grey-level co-occurrence matrices for texture classification," *International Symposium on Information Technology (ITSim 2008)*, pp. 1–6, 2008.

[7] P. L. Paula Filho, L. S. Oliveira, A. S. Britto, and R. Sabourin, "Forest species recognition using color-based features," in *Proceedings of the 20th International Conference on Pattern Recognition*, Instanbul, Turkey, 2010, pp. 4178–4181.

[8] M. Nasirzadeh, A. A. Khazael, and M. b. Khalid, "Woods recognition system based on local binary pattern," in *2010 2nd International Conference on Computational Intelligence, Communication Systems and Networks*, July 2010, pp. 308–313.

[9] R. Haralick, K. Shanmugam, and I. Dinstein, "Texture features for image classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 3, no. 6, 1973.

[10] D. Bertolini, L. Oliveira, E. Justino, and R. Sabourin, "Texture-based descriptors for writer identification and verification," *Expert Syst. Appl.*, vol. 40, no. 6, pp. 2069–2080, May 2013. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2012.10.016

[11] M. A. Simaan, "Texture-based techniques for interpretation of seismic images," in *OCEANS '98 Conference Proceedings*, vol. 1, Sep 1998, pp. 429–432 vol.1.

[12] D. Gao, "Latest developments in seismic texture analysis for subsurface structure, facies, and reservoir characterization: A review," *GEOPHYSICS*, vol. 76, no. 2, pp. W1–W13, 2011. [Online]. Available: http://dx.doi.org/10.1190/1.3553479

[13] A. Amin and M. Deriche, "A hybrid approach for salt dome detection in 2d and 3d seismic data," in *2015 IEEE International Conference on Image Processing (ICIP)*, Sept 2015, pp. 2537–2541.

[14] R. D. S. Ferreira, A. B. Mattos, E. V. Brazil, R. Cerqueira, M. Ferraz, and S. Cersosimo, "Multi-scale evaluation of texture features for salt dome detection," in *2016 IEEE International Symposium on Multimedia (ISM)*, Dec 2016, pp. 632–635.

[15] M. Jhuria, A. Kumar, and R. Borse, "Image processing for smart farming: Detection of disease and fruit grading," in *2013 IEEE Second International Conference on Image Information Processing (ICIIP-2013)*, Dec 2013, pp. 521–526.

[16] S. Mohana and C. Prabhakar, "A novel technique for grading of dates using shape and texture features," *Machine Learning and Applications: An International Journal (MLAIJ)*, vol. 1, no. 2.

[17] J. D. Pujari, R. Yakkundimath, and A. S. Byadgi, "Image processing based detection of fungal diseases in plants," *Procedia Computer Science*, vol. 46, pp. 1802 – 1808, 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S187705091500201X

[18] G. Z. T. A. M Pietikäinen, A Hadid, *Computer Vision Using Local Binary Patterns*, ser. 10. London: Springer, 2011, vol. 40.

[19] L. Liu, P. Fieguth, Y. Guo, X. Wang, and M. Pietikinen, "Local binary features for texture classification," *Pattern Recogn.*, vol. 62, no. C, pp. 135–160, Feb. 2017. [Online]. Available: https://doi.org/10.1016/j.patcog.2016.08.032

[20] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using gabor filters," in *1990 IEEE International Conference on Systems, Man, and Cybernetics Conference Proceedings*, Nov 1990, pp. 14–19.

[21] P. Cavalin, L. S. Oliveira, A. L. Koerich, and A. S. Britto, "Wood defect detection using grayscale images and an optimized feature set," in *IECON 2006 - 32nd Annual Conference on IEEE Industrial Electronics*, Nov 2006, pp. 3408–3412.

[22] P. R. Cavalin, J. Martins, M. N. Kapp, and L. E. S. Oliveira, "A multiple feature vector framework for forest species recognition," Coimbra, Portugal, pp. 16–20, 2013.

[23] R. M. Haralick, "Statistical and structural approach to texture," *Proceedings of IEEE*, vol. 65, no. 5, 1979.

[24] J. Martins, L. S. Oliveira, S. Nisgoski, and R. Sabourin, "A database for automatic classification of forest species," *Machine Vision and Applications*, vol. 24, no. 3, pp. 567–578, 2013. [Online]. Available: http://dx.doi.org/10.1007/s00138-012-0417-5

[25] G. Kylberg and I.-M. Sintorn, "Evaluation of noise robustness for local binary pattern descriptors in texture classification," *EURASIP Journal on Image and Video Processing*, vol. 2013, no. 1, p. 17, 2013. [Online]. Available: http://dx.doi.org/10.1186/1687-5281-2013-17

[26] D. Harwood, T. Ojala, M. Pietikäinen, S. Kelman, and L. Davis, "Texture classification by center-symmetric auto-correlation, using Kullback discrimination of distributions," *Pattern Recognition Letters*, vol. 16, no. 1, pp. 1–10, 1995.

[27] L. Wang and D.-C. He, "Texture classification using texture spectrum," *Pattern Recogn.*, vol. 23, no. 8, pp. 905–910, Aug. 1990. [Online]. Available: http://dx.doi.org/10.1016/0031-3203(90)90135-8

[28] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions." *Pattern Recognition*, vol. 29, no. 1, pp. 51–59, 1996. [Online]. Available: http://dblp.uni-trier.de/db/journals/pr/pr29.htmlOjalaPH96

[29] T. Ojala, M. Pietikãinen, and T. Mãenpãã, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[30] B. Yang and S. Chen, "A comparative study on local binary pattern (lbp) based face recognition: Lbp histogram versus lbp image," *Neurocomputing*, vol. 120, pp. 365 – 379, 2013, image Feature Detection and Description. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231213003068

[31] L. Nanni, A. Lumini, and S. Brahnam, "Local binary patterns variants as texture descriptors for medical image analysis," *Artif. Intell.*

Med., vol. 49, no. 2, pp. 117–125, Jun. 2010. [Online]. Available: http://dx.doi.org/10.1016/j.artmed.2010.02.006

[32] ——, "Survey on lbp based texture descriptors for image classification," *Expert Syst. Appl.*, vol. 39, no. 3, pp. 3634–3641, Feb. 2012. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2011.09.054

[33] R. Mehta and K. Egiazarian, "Dominant rotated local binary patterns (drlbp) for texture classification," *Pattern Recogn. Lett.*, vol. 71, no. C, pp. 16–22, Feb. 2016. [Online]. Available: https://doi.org/10.1016/j.patrec.2015.11.019

[34] C. J., K. V., Z. G., and P. M., "Rlbp: Robust local binary pattern." in *Proc. the British Machine Vision Conference (BMVC 2013), Bristol, UK*, 2013, p. 10 p.

[35] M. Cote and A. B. Albu, "Robust texture classification by aggregating pixel-based lbp statistics," *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 2102–2106, Nov 2015.

[36] L. Liu, S. Lao, P. W. Fieguth, Y. Guo, X. Wang, and M. Pietikäinen, "Median robust extended local binary pattern for texture classification," *IEEE Transactions on Image Processing*, vol. 25, no. 3, pp. 1368–1381, March 2016.

[37] M. Idrissa and M. Acheroy, "Texture classification using gabor filters," *Pattern Recognition Letters*, vol. 23, no. 9, pp. 1095 – 1102, 2002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167865502000569

[38] F. Bianconi and A. Fernández, "Evaluation of the effects of gabor filter parameters on texture classification," *Pattern Recogn.*, vol. 40, no. 12, pp. 3325–3335, Dec. 2007. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2007.04.023

[39] Y. Ji, K. H. Chang, and C.-C. Hung, "Efficient edge detection and object segmentation using gabor filters," in *Proceedings of the 42Nd Annual Southeast Regional Conference*, ser. ACM-SE 42. New York, NY, USA: ACM, 2004, pp. 454–459. [Online]. Available: http://doi.acm.org/10.1145/986537.986651

[40] V. Ojansivu and J. Heikkilä, "Blunr insensitive texture classification using local phase quantization," in *Proceedings of the 3rd International Conference on Image and Signal Processing (ICISP '08)*, 2008, pp. 236–243.

[41] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 3, pp. 226–239, Mar. 1998. [Online]. Available: http://dx.doi.org/10.1109/34.667881

[42] P. L. Paula Filho, L. S. Oliveira, S. Nisgoski, and A. S. Britto, Jr., "Forest species recognition using macroscopic images," *Mach. Vision Appl.*, vol. 25, no. 4, pp. 1019–1031, May 2014. [Online]. Available: http://dx.doi.org/10.1007/s00138-014-0592-7

[43] L. G. Hafemann, L. S. Oliveira, and P. R. Cavalin, "Forest species recognition using deep convolutional neural networks," in *22nd International Conference on Pattern Recognition, ICPR 2014, Stockholm, Sweden, August 24-28, 2014*, 2014, pp. 1103–1107. [Online]. Available: http://dx.doi.org/10.1109/ICPR.2014.199

[44] P. R. Cavalin, M. N. Kapp, and L. E. S. Oliveira, "An adaptive multi-level framework for forest species recognition," in *Brazilian Conference on Intelligent Systems*, Natal, Brazil, 2015.

[45] L. G. Hafemann, L. S. Oliveira, P. R. Cavalin, and R. Sabourin, "Transfer learning between texture classification tasks using convolutional neural networks," in *Neural Networks (IJCNN), 2015 International Joint Conference on*. IEEE, 2015, pp. 1–7.

[46] M. Kapp, R. Bloot, P. R. Cavalin, and L. S. Oliveira, "Automatic forest species recognition based on multiple feature sets," in *International Joint Conference ou Neural Networks*, 2014, pp. 1296–1303.

[47] W. Roderick, C. Anderson, and A. Manheim, "Remote surface classification for robotic platforms," Stanford University, Tech. Rep., 2016.

[48] M. Cimpoi, S. Maji, I. Kokkinos, and A. Vedaldi, "Deep filter banks for texture recognition, description, and segmentation," *International Journal of Computer Vision*, vol. 118, no. 1, pp. 65–94, 2016. [Online]. Available: http://dx.doi.org/10.1007/s11263-015-0872-3

[49] F. Spanhol, L. S. Oliveira, C. Petitjean, and L. Heutte, "A dataset for breast cancer histopathological image classification," *IEEE Transactions on Biomedical Engineering (TBME)*, 2015.

[50] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi, "Describing textures in the wild," in *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[51] F. Spanhol, P. Cavalin, L. S. Oliveira, C. Petitjean, and L. Heutte, "Deep features for breast cancer histopathological image classification (accepted for publication)," in *International Conference on Systems, Man and Cybernetics (SMC 2017)*, Banff, Canada, 2017.

[52] P. R. Cavalin, M. N. Kapp, and L. E. S. Oliveira, "Multi-scale texture recognition systems with reduced cost: a case study on forest species (accepted for publication)," in *International Conference on Systems, Man and Cybernetics (SMC 2017)*, Banff, Canada, 2017.

[53] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[54] F. H. C. Tivive and A. Bouzerdoum, "Texture classification using convolutional neural networks," in *2006 IEEE Region 10 Conference (TENCON 2006)*, Hong Kong, China, 2006, pp. 1–14.

[55] H. Adly and M. Moustafa, "A hybrid deep learning approach for texture analysis," 03 2017.

[56] V. Andrearczyk and P. F. Whelan, "Using filter banks in convolutional neural networks for texture classification," *Pattern Recognition Letters*, vol. 84, pp. 63 – 69, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167865516302185

[57] S. Fujieda, K. Takayama, and T. Hachisuka, "Wavelet Convolutional Neural Networks for Texture Classification," *ArXiv e-prints*, Jul. 2017.

[58] H. Zhang, J. Xue, and K. Dana, "Deep TEN: Texture Encoding Network," in *Conference on Computer Vision and Pattern Recognition (CVPR 2017)*, 2017.

[59] A. Radhakrishnan, C. Durham, A. Soylemezoglu, and C. Uhler, "Patchnet: Interpretable neural networks for image classification," *CoRR*, vol. abs/1705.08078, 2017. [Online]. Available: http://arxiv.org/abs/1705.08078

[60] Y. M. Costa, L. S. Oliveira, and C. N. Silla, "An evaluation of convolutional neural networks for music classification using spectrograms," *Applied Soft Computing*, vol. 52, no. Complete, pp. 28–38, 2017.

[61] L. Hou, D. Samaras, T. M. Kurc, Y. Gao, J. E. Davis, and J. H. Saltz, "Patch-based convolutional neural network for whole slide tissue image classification," in *Proceedings. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2016)*, 2016.

[62] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky, *Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I.* Cham: Springer International Publishing, 2014, ch. Neural Codes for Image Retrieval, pp. 584–599. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-10590-1$_{3}$8

[63] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, 2014, pp. 647–655. [Online]. Available: http://jmlr.org/proceedings/papers/v32/donahue14.html

[64] S. Hossain and S. Serikawa, "Texture databases – a comprehensive survey," *Pattern Recognition Letters*, vol. 34, no. 15, pp. 2007 – 2022, 2013, smart Approaches for Human Action Recognition. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167865513000536

[65] F. Bianconi and A. Fernández, "An appendix to "texture databases – a comprehensive survey"," *Pattern Recognition Letters*, vol. 45, pp. 33 – 38, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167865514000671