CHAPTER 3

Biomedical Texture Operators and Aggregation Functions

A Methodological Review and User's Guide

Adrien Depeursinge[*,†], **Julien Fageot**[*]

[*]École Polytechnique Fédérale de Lausanne (EPFL), Biomedical Imaging Group, Lausanne, Switzerland
[†]University of Applied Sciences Western Switzerland (HES-SO), Institute of Information Systems, Sierre, Switzerland

Abstract

This chapter reviews most popular texture analysis approaches under novel comparison axes that are specific to biomedical imaging. A concise checklist is proposed as a user guide to assess the relevance of each approach for a particular medical or biological task in hand. We revealed that few approaches are regrouping most of the desirable properties for achieving optimal performance. In particular, moving frames texture representations based on learned steerable operators showed to enable data-specific and rigid-transformation-invariant characterization of local directional patterns, the latter being a fundamental property of biomedical textures. Potential limitations of having recourse to data augmentation and transfer learning for deep convolutional neural networks and dictionary learning approaches to palliate the lack of large annotated training collections in biomedical imaging are mentioned. We conclude by summarizing the strengths and limitations of current approaches, providing insights on key aspects required to build the next generation of biomedical texture analysis approaches.

Keywords

Biomedical texture analysis, Convolutional neural networks, Deep learning, Dictionary learning, Gabor filters, Gray-level cooccurrence matrices, Gray-level run-length matrices, Gray-level size-zone matrices, Local binary patterns, Steerable wavelets

3.1 INTRODUCTION

When starting from scratch to enable quantitative image analysis in a biomedical research project or a clinical setting, the number of available approaches can be disconcerting. Whereas implementing intensity-based features already raises several challenges in terms of reproducibility and relevance to a particular medical or biological applicative context, finding adequate texture measures requires extensive expertise as well as time-consuming iterative validation processes. Knowing precisely the type of textural information sought and further opting for the appropriate analysis technique is as much challenging as crucial for success. To that end, Section 1.2 of Chapter 1 proposed a formal definition of Biomedical Texture (BT) from both perceptual and mathematical

DOI: 10.1016/B978-0-12-812133-7.00003-X

perspectives. It was suggested that BTs are realizations of intricate and nonstationary spatial stochastic processes, and that spatial scales and directions are fundamental properties of BTs. The relation between scale and directions was further developed in Section 2.4.1 of Chapter 2, suggesting that the Local Organization of Image Directions (LOID) is a fundamental property of BT. The LOIDs relate to the texton theory (see Section 1.2.4 of Chapter 1), where textons are crucial elements of preattentive vision [1]. Section 1.3.1 of Chapter 1 introduced a general framework of texture analysis methods where any BTA approach can be decomposed into a succession of local texture operators and regional aggregation functions. In addition, Section 1.3.3 of Chapter 1 and Chapter 7 provided evidence that the type of geometric invariances required for BTA are invariances to nonrigid transformations,[1] which considerably differs from general purpose texture analysis traditionally used in computer vision for photographic image analysis. These aspects were further clarified and exemplified in Chapter 2, where a set of comparison dimensions between BTA methods was introduced. In particular, nonexclusive categories of texture operators were presented, including directionally insensitive/sensitive, aligned, Moving Frames (MF), and learned. The discrimination abilities of each category was evaluated, where the destructive effects of integrative aggregation were demonstrated. In particular, it was found that some groups of operators lose their ability to characterize the LOIDs when integrated over a Region Of Interest (ROI), the latter being required to obtain scalar-valued texture measurements. The ultimate challenging requirement of combining the ability to describe the LOIDs with robustness to rigid transformations was only fulfilled by the groups of MF operators.

In this work, we compare most popular BTA approaches under the light of the novel comparison dimensions introduced in Chapter 2. A concise checklist is proposed to assess the relevance of each BTA approach for a particular medical or biological task in hand. The main groups of methods considered are (i) convolutional (Section 3.2, further information on CNNs can be found in Chapters 4 and 9), (ii) Gray-Level Matrices (GLM, Section 3.3), (iii) Local Binary Patterns (LBP, Section 3.4), and (iv) fractals (Section 3.5, further information can be found in Chapter 5). The main categories of BTA methods are summarized in Fig. 3.1. Our choice to review these categories is based on their popularity and diversity. The strengths and weaknesses of the methods are summarized and discussed in Section 3.6. The notation used in this chapter are based on Section 1.2.2 of Chapter 1.

[1] Optimal operators should be equivariant to translations and locally equivariant to rotations.

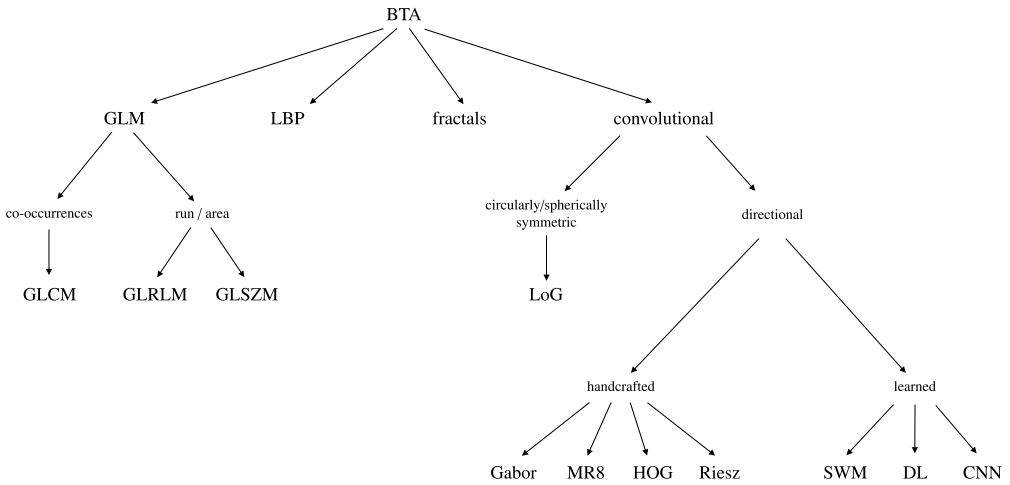

Figure 3.1 Overview and dependence of the BTA approaches reviewed in this chapter. These categories were chosen based on their popularity and diversity.

3.2 CONVOLUTIONAL APPROACHES

A large group of approaches called *convolutional* are based on linear texture operators. In spatial domain, the application of a linear operator $\mathcal{G}_n$ to $f(\boldsymbol{x})$[2] at the position $\boldsymbol{x}_0$ is characterized by the *operator function* $g_n(\boldsymbol{x})$ as

$$\mathcal{G}\{f\}(\boldsymbol{x}_0) = (g_n * f)(\boldsymbol{x}) = \int_{\boldsymbol{x}\in\mathbb{R}^D} f(\boldsymbol{x}) g_n(\boldsymbol{x}_0 - \boldsymbol{x})\, \mathrm{d}\boldsymbol{x}. \tag{3.1}$$

The operation $(g_n * f)(\boldsymbol{x}_0)$ is called a *convolution*: we slide the operator function $g_n(\boldsymbol{x})$ to the input texture function $f(\boldsymbol{x})$ over all positions $\boldsymbol{x}_0$. When the position $\boldsymbol{x}_0$ is fixed, we remark that the value of $\mathcal{G}\{f\}(\boldsymbol{x}_0)$ is simply the scalar product

$$\mathcal{G}_n\{f\}(\boldsymbol{x}_0) = \langle f(\cdot), g_n(\boldsymbol{x}_0 - \cdot)\rangle. \tag{3.2}$$

A convolution in the spatial domain corresponds to a multiplication in the Fourier domain as

$$(g_n * f)(\boldsymbol{x}) \overset{\mathcal{F}}{\longleftrightarrow} \hat{g}_n(\boldsymbol{\omega}) \cdot \hat{f}(\boldsymbol{\omega}).$$

Convolution operators are the ones for which the response map $h_n(\boldsymbol{x})$ depends linearly on the input texture function $f(\boldsymbol{x})$. The definition of the function $g_n(\boldsymbol{x})$ is a priori

[2] In this section, except as otherwise stipulated, we will consider continuous operators and functions indexed by the coordinate vector $\boldsymbol{x}$. Discretized versions can be obtained following the notions introduced in Section 1.2.2 of Chapter 1.

free of constraints. For the convolution, the spatial support of the operator is exactly the domain on which the operator function $g_n(\boldsymbol{x})$ is nonzero. Operators with finite spatial supports are desirable to study local texture properties of nonstationary processes (see Section 2.3.1 of Chapter 2), hence we shall consider localized operator functions. Also, band-pass filters for which $\hat{g}_n(\mathbf{0}) = 0$ (*i.e.*, zero gain for the null frequency $\boldsymbol{\omega} = \mathbf{0}$ corresponding to the average) are able to focus on texture alone and do not include any intensity information. This ensures improved robustness of the texture operator responses to variations in illumination.

As will be exploited thereafter, it is often interesting to define new texture operators based on convolution operators, with the goal of locally aligning the orientation (see Section 2.4.3 of Chapter 2). We develop this framework in the 2D setting. The rotation matrix with angle θ_0 is noted

$$\mathbf{R}_{\theta_0} = \begin{pmatrix} \cos\theta_0 & -\sin\theta_0 \\ \sin\theta_0 & \cos\theta_0 \end{pmatrix}.$$

Consider a convolution operator $\mathcal{G}_n$ with operator function $g_n(\boldsymbol{x})$. For each location $\boldsymbol{x}_0$, we set

$$\theta_{\boldsymbol{x}_0} = \arg \max_{\theta_0 \in [0, 2\pi)} \left\langle f(\cdot), g_n(\boldsymbol{x}_0 - \mathbf{R}_{\theta_0} \cdot) \right\rangle \tag{3.3}$$

Here, $\theta_{\boldsymbol{x}_0}$ is the angle that maximizes the scalar product between the texture image $f(\boldsymbol{x})$ and the rotated version of $g_n(\boldsymbol{x})$ around the location $\boldsymbol{x}_0$. It is therefore the angle for which the operator function is the most aligned with $f(\boldsymbol{x})$ at position $\boldsymbol{x}_0$. We then define

$$\begin{aligned} \mathcal{H}_n\{f\}(\boldsymbol{x}_0) &= \langle f(\cdot), g_n(\boldsymbol{x}_0 - \mathbf{R}_{\theta_{\boldsymbol{x}_0}} \cdot) \rangle \\ &= \int_{\boldsymbol{x} \in \mathbb{R}^D} f(\boldsymbol{x}) g_n(\boldsymbol{x}_0 - \mathbf{R}_{\theta_{\boldsymbol{x}_0}} \boldsymbol{x}) \mathrm{d}\boldsymbol{x}. \end{aligned} \tag{3.4}$$

The response map $h_n(\boldsymbol{x}) = \mathcal{H}_n\{f\}(\boldsymbol{x})$ is no longer linear in $f(\boldsymbol{x})$, therefore the texture operator $\mathcal{H}_n$ is not a convolution operator in the sense of Eq. (3.1). We include the angle alignment in this section since it is based on the convolutional framework and characterized by the operator function $g_n(\boldsymbol{x})$ according to Eq. (3.4). Convolution operators and local angle alignment together allow to design texture operators equivariant to local rotations that can be used to study the local orientation in the texture image (see Section 1.3.3 of Chapter 1).

The most popular convolutional approaches in texture analysis are detailed in the following subsections while distinguishing three main categories of convolutional operator functions (see Fig. 3.1): circularly/spherically symmetric filters (Section 3.2.1), directional filters (Section 3.2.2), and learned filters (Section 3.2.3). It is worth noting that directional and learned filters are nonexclusive categories.

3.2.1 Circularly/spherically symmetric filters

Circularly/spherically symmetric filters are convolutional texture operators with functions that only depend on the radial polar coordinate r (see Section 2.2 of Chapter 2): $g_n(\boldsymbol{x}) = g_n(||\boldsymbol{x}||) = g_n(r)$. A direct consequence of this is their complete lack of directional sensitivity and their invariance to local rotations.

A simple example of such convolutional texture processing is based on Laplacian of Gaussian (LoG) filters.[3] Their handcrafted operator function $g_\sigma(\boldsymbol{x})$ is a radial second-order derivative of a D-dimensional Gaussian filter as

$$g_\sigma(\boldsymbol{x}) = -\frac{1}{\pi\sigma^2}\left(1 - \frac{||\boldsymbol{x}||^2}{2\sigma^2}\right) e^{-\frac{||\boldsymbol{x}||^2}{2\sigma^2}}, \tag{3.5}$$

where the standard deviation of the Gaussian σ controls the scale of the operator. LoGs are band-pass and circularly/spherically symmetric. It is straightforward to notice in Eq. (3.5) that $g_\sigma(\boldsymbol{x})$ only depends on the norm of $\boldsymbol{x}$ and is therefore circularly symmetric: we have $g_\sigma(\boldsymbol{x}) = g_\sigma(r)$. Their 2D profiles and response maps obtained with synthetic tumors are depicted in Fig. 1.12 of Chapter 1. LoGs can also be approximated by a difference of two Gaussians.[4] LoGs were mentioned to be important in biological visual processing by Marr in [2].

Multiscale texture measurements $\boldsymbol{\eta}$ can be obtained by averaging the absolute values or the energies of the response maps (*i.e.*, $|h_{\sigma_n}(\boldsymbol{x})|$ or $h^2_{\sigma_n}(\boldsymbol{x})$, respectively) of a series of operators with increasing values of $\sigma_1 < \sigma_2 < \cdots < \sigma_N$ (see Fig. 1.12 of Chapter 1). The average is computed over a ROI $\boldsymbol{M}$. They are implemented in 2D in the TexRAD[5] commercial medical research software [4]. The properties of LoG filters are summarized in Table 3.1.

3.2.2 Directional filters

Several handcrafted approaches were proposed for texture analysis using directional filters. The latter are sensitive to image directions (see Section 2.4.2 of Chapter 3). The most important and popular methods are discussed in this section, including Gabor wavelets (Section 3.2.2.1), Maximum Response 8 (MR8, Section 3.2.2.2), Histogram of Oriented Gradients (HOG, Section 3.2.2.3), and the Riesz transform (Section 3.2.2.4).

[3] They are also called *Mexican hat* filters due their 2D shape.

[4] The Difference of Gaussians (DoG) best approximates LoG when the ratio of the two variances of the Gaussians are $\sigma_1 = \frac{\sigma_2}{\sqrt{2}}$.

[5] http://texrad.com, University of Sussex, Brighton, UK, as of November 21, 2016.

Table 3.1 Properties LoG filters

Operator linearity	Linear.
Handcrafted	Yes.
3D extension	Trivial as the second-order derivative of a 3D Gaussian function. The LoGs are circularly/spherically symmetric functions which only depend on the radial coordinate r.
Coverage of image directions	Complete: the angular part is constant for a fixed radius r.
Directionality and local rotation-invariance	Directionally insensitive and locally rotation-invariant.
Characterization of the LOIDs	No.
Coverage of image scales	Incomplete for a single value of σ. However, their wavelet extension allows a full coverage of the spatial spectrum [3].
Band-pass	Yes.
Gray-level reduction	Not required.
Illumination-invariance	No. However, they are robust to changes in illumination because LoGs are band-pass filters.
Aggregation function	Integrative (typically): computes the average of the absolute values or the energies of the response maps over $\boldsymbol{M}$.

3.2.2.1 Gabor wavelets

A popular example of directional texture operators is the family of Gabor wavelets [5–7], which consists in a systematic parcellation of the Fourier domain with elliptic Gaussian windows (see Fig. 3.2). The D-dimensional filter function of the Gabor operator $g_{\rho,\boldsymbol{\sigma},\theta}(\boldsymbol{x})$ is the most general function that minimizes the uncertainty principle [8] (see Section 2.3.1 of Chapter 2). Moreover, it was suggested in [9] that 2D Gabor filters are appropriate models of the transfer function of simple cells in the visual cortex of mammalian brains and thus mimicking the early layers of human visual perception. In the spatial domain, they correspond to Gaussian-windowed oscillatory functions as

$$g_{\rho,\boldsymbol{\sigma},\theta}(\boldsymbol{x}) = \frac{\sigma_1\sigma_2}{2\pi}\, e^{-\sigma_1^2\tilde{x}_1^2+\sigma_2^2\tilde{x}_2^2}\, e^{\mathrm{j}2\pi\rho\tilde{x}_1}, \tag{3.6}$$

where ρ is the center of the Gaussian window in Fourier, $\boldsymbol{\sigma} = (\sigma_1, \sigma_2)$ contains the radial and orthoradial standard deviations of the Gaussian window in Fourier, $\tilde{\boldsymbol{x}} = (\tilde{x}_1, \tilde{x}_2) = \mathbf{R}_\theta \boldsymbol{x}$, and $\tilde{\boldsymbol{\omega}} = (\tilde{\omega}_1, \tilde{\omega}_2) = \mathbf{R}_\theta \boldsymbol{\omega}$ (see Fig. 3.2).

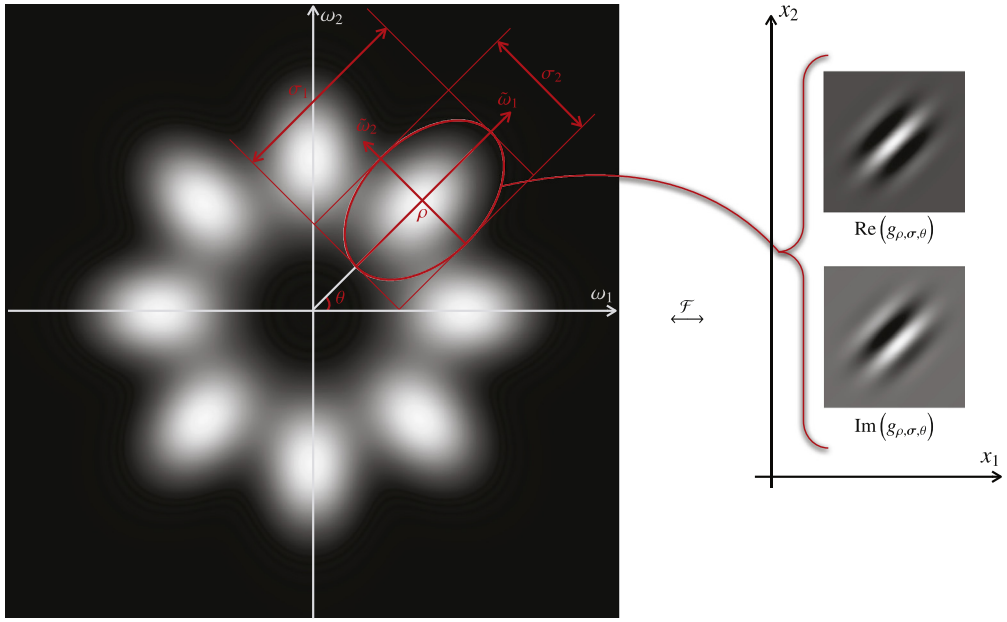

Figure 3.2 One isolated scale ρ of Gabor wavelets with 4 orientations θ and a spectral dispersion of $\boldsymbol{\sigma} = (\sigma_1, \sigma_2)$. Note that the spatial angular polar coordinate θ is equivalent to the Fourier angular coordinate ϑ.

Multiscale and multiorientation texture measurements $\boldsymbol{\eta}$ are obtained by averaging the absolute values or the energies of the response maps (*i.e.*, $|h_{\rho,\boldsymbol{\sigma},\theta}(\boldsymbol{x})|$ or $h^2_{\rho,\boldsymbol{\sigma},\theta}(\boldsymbol{x})$, respectively) for various values of ρ, $\boldsymbol{\sigma}$, and θ. The properties of Gabor wavelets are summarized in Table 3.2.

3.2.2.2 Maximum Response 8 (MR8)

Another popular example of handcrafted directional filterbank is the MR8 approach [11]. The latter includes a filterbank with a collection of 38 operator functions $g_n(\boldsymbol{x})$ (see Fig. 3.3). Two of them are circularly symmetric (one Gaussian and one LoG). Eighteen are directional multiscale edge detectors based on oriented first-order Gaussian derivatives. Another 18 are directional multiscale ridge detectors based on oriented second-order Gaussian derivatives. However, the approach yields a total of eight response maps. The first two come from the convolution of the image with the circularly symmetric Gaussian and LoG filters. The remaining six are computed as follows. For each directional detector types (*i.e.*, first- and second-order Gaussian derivatives), only one detector per scale is kept. Among the six different filter orientations per scale, the detector kept at the position $\boldsymbol{x}_0$ is the one that maximizes the detection in the sense of Eq. (3.3), where θ_0 is coarsely discretized. This results in six non-linear texture operators

Table 3.2 Properties of Gabor wavelets

Operator linearity	Yes.
Handcrafted	Yes.
3D extension	Requires using 3D elliptic Gaussian windows in the volumetric Fourier domain and systematically indexing their orientations with angles θ, ϕ (corresponding to angles ϑ, φ in spherical Fourier, see Section 2.2 of Chapter 2).
Coverage of image directions	Complete with appropriate choices of directions θ and orthoradial standard deviations σ_2 respecting Parseval's identity.
Directionality and local rotation-invariance	Directional and not locally rotation-invariant. A local rotation of the input image $f(\mathbf{R}_{\theta_0,\boldsymbol{x}_0} \cdot \boldsymbol{x})$ will swap the responses of the operators with various orientations θ.
Characterization of the LOIDs	No. Gabor filters/wavelets are unidirectional operators that are not able to characterize the LOIDs when used with an integrative aggregation function (*e.g.*, average, see Section 2.4.2 and Fig. 2.13 of Chapter 2). MF representations based on a consistent alignment criteria (*e.g.*, Hessian-based structure tensor, see Section 2.4.3 and Fig. 2.14 of Chapter 2) can be used to locally align Gabor operators (using, *e.g.*, steerability [10]) and allow characterizations of the LOIDs.
Coverage of image scales	Complete for filters with appropriate choices of radial frequencies ρ and standard deviation σ_1 respecting Parseval's identity. This is the case for Gabor wavelets.
Band-pass	Yes.
Gray-level reduction	Not required.
Illumination-invariance	No. However, they are robust to changes in illumination because Gabor filters are band-pass.
Aggregation function	Integrative (typically): computes the average of the absolute values or the energies of the response maps over $\boldsymbol{M}$.

$\mathcal{G}_{n=1,\ldots,6}$ that are "aligned" at every position and achieve approximated local rotation-invariance. This process is illustrated in Fig. 3.3.

The aggregation function usually consists in pixel-wise clustering of the maximum filter responses to create a texton dictionary (see Section 1.2.4 of Chapter 1). Class-wise models can then be created as texton occurrence histograms, which can be used to compare texture instances by measuring distances between them (*e.g.*, Euclidean, χ^2). The vector of texture measurements $\boldsymbol{\eta}$ contains the bin values of the texton occurrences. The properties of the MR8 approach are summarized in Table 3.3.

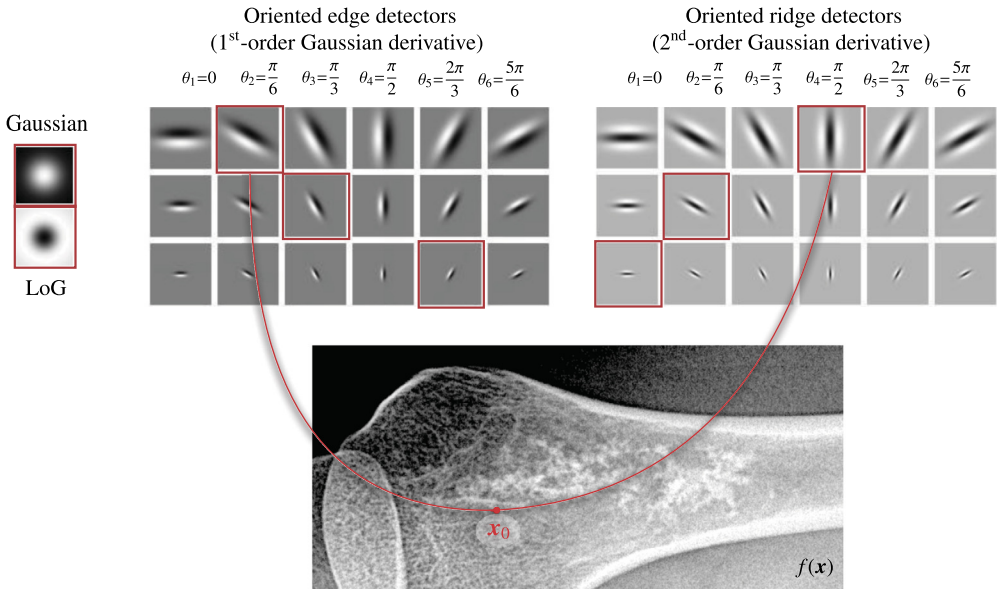

Figure 3.3 Top row: the 38 operators of the MR8 filterbank. At a given position $\boldsymbol{x}_0$, the collection of operators $\mathcal{G}_{n=1,\ldots,8}\{f\}(\boldsymbol{x}_0)$ consists of a subset of the six directional operators with maximum response over all orientations $\theta_{1,\ldots,6}$ plus the two circularly symmetric operators. This results in eight responses for each position (hence the name of the approach), which are marked in red in the example above.

3.2.2.3 Histogram of Oriented Gradients (HOG)

An efficient gradient-based MF representation is the local image descriptor used in the Scale Invariant Feature Transform (SIFT) approach [12] called the Histogram of Oriented Gradients (HOG). The texture function $f(\boldsymbol{x})$ is first filtered with circularly symmetric multiscale DoG filters $g_{\sigma_i}(\boldsymbol{x})$ (see Section 3.2.1) using a dyadic scale progression (*i.e.*, $\sigma_{i+1} = 2\sigma_i$), which yields a collection of response maps $h_{\sigma_i}(\boldsymbol{x})$. At a fixed scale i, corresponding to the standard deviation $\sigma_i = \sigma$, the gradient orientation map $h^{\theta}_{\sigma}(\boldsymbol{x})$ is computed from the response map $h_{\sigma}(\boldsymbol{x})$ as

$$h^{\theta}_{\sigma}(\boldsymbol{x}) = \arctan\left(\frac{\nabla_{x_2}\{h_\sigma\}(\boldsymbol{x})}{\nabla_{x_1}\{h_\sigma\}(\boldsymbol{x})}\right), \tag{3.7}$$

where $\nabla_{x_1}\{h_\sigma\}(\boldsymbol{x})$ and $\nabla_{x_2}\{h_\sigma\}(\boldsymbol{x})$ yield gradients maps. Equation (3.7) provides local angle values maximizing the gradient magnitude in the spirit of Eq. (3.3). For a fixed orientation θ, we set $\boldsymbol{u}(\theta) = \mathbf{R}_\theta(1,0) = (\cos\theta, \sin\theta)$ and $\nabla_{\boldsymbol{u}(\theta)} = \cos\theta \cdot \nabla_{x_1} + \sin\theta \cdot \nabla_{x_2}$ the oriented derivative with direction $\boldsymbol{u}(\theta)$. We define the oriented response map as $h_{\sigma,\theta}\{\boldsymbol{x}_0\} = \nabla_{\boldsymbol{u}(\theta)}\{h_\sigma\}(\boldsymbol{x}_0)$. When using discretized image functions, operators and response maps $h_\sigma(\boldsymbol{\xi})$ (see Section 1.2.2 of Chapter 1), the latter can be estimated using

Table 3.3 MR8 properties

Operator linearity	No. The filtering operations are linear, but keeping the maximum value among the six orientations is a nonlinear operation.
Handcrafted	The operators are handcrafted.[a]
3D extension	Requires extending the filter orientations to 3D with systematic sampling of angles θ, ϕ.
Coverage of image directions	Near to complete, but not strictly respecting Parseval's identity.
Directionality and local rotation-invariance	Directional and approximate local rotation-invariance.
Characterization of the LOIDs	No, all MR8 operators are unidirectional. In addition, they are "aligned" independently at a position $\boldsymbol{x}_0$ and do not yield MF representations.
Coverage of image scales	Incomplete.
Band-pass	Yes, with the exception of the circularly symmetric Gaussian filter (low-pass).
Gray-level reduction	Not required.
Illumination-invariance	No, mostly because of the circularly symmetric Gaussian filter.
Aggregation function	Consists of two consecutive aggregation functions: a first piece-wise integrative function[b] is used to construct the texton dictionary and second integrative function counts the texton occurrences and organizes them in a histogram.

[a] The texton dictionary is derived from the data, but the operators are handcrafted.
[b] Clustering results in averaging within local homogeneous regions in the feature space.

pixel differences as

$$\begin{aligned} \nabla_{x_1}\{h_\sigma\}(\boldsymbol{\xi}) &= h_\sigma(\xi_1 + \Delta\xi_1, \xi_2) - h_\sigma(\xi_1 - \Delta\xi_1, \xi_2), \\ \nabla_{x_2}\{h_\sigma\}(\boldsymbol{\xi}) &= h_\sigma(\xi_1, \xi_2 + \Delta\xi_2) - h_\sigma(\xi_1, \xi_2 - \Delta\xi_2). \end{aligned} \tag{3.8}$$

For each position $\boldsymbol{x}_0$, the dominant gradient direction $\theta_{\boldsymbol{x}_0}$ is obtained by maximizing Eq. (3.7). To define the HOG operators, we align and sample the orientation $\theta = \theta_q - \theta_{\boldsymbol{x}_0} = \frac{2\pi}{q} - \theta_{\boldsymbol{x}_0}$ with $q = 1, \ldots, 8$. The final collection of HOG operators $\mathcal{G}_{\sigma,q}$ is given at location $\boldsymbol{x}_0$ by

$$\mathcal{G}_{\sigma,q}\{f\}(\boldsymbol{x}_0) = h_{\sigma,q}(\boldsymbol{x}_0).$$

The value of $h_{\sigma,q}(\boldsymbol{x}_0)$ can be efficiently evaluated using the steerability of the gradient operator [13] as

$$h_{\sigma,q}(\boldsymbol{x}_0) = \cos(\theta_q - \theta_{\boldsymbol{x}_0}) \cdot \nabla_{x_1}\{h_\sigma\}(\boldsymbol{x}_0) + \sin(\theta_q - \theta_{\boldsymbol{x}_0}) \cdot \nabla_{x_2}\{h_\sigma\}(\boldsymbol{x}_0). \tag{3.9}$$

The collection of HOG operators $\mathcal{G}_{\sigma,q}$ provides a MF representation oriented with $\theta_{\boldsymbol{x}_0}$ and containing eight redundant frame components (see Section 2.4.3 of Chapter 2). Whereas the response maps $h_\sigma(\boldsymbol{x})$ depend linearly on the texture image $f(\boldsymbol{x})$, it is not anymore the case for the HOG operators due to the local alignment of the angle: the HOG operators are not convolution operators themselves, but take advantage of the convolution framework since they are based on the DoG filters.

For obtaining texture measurements $\boldsymbol{\eta}$, the responses of HOG operators can be aggregated over regions $\boldsymbol{M}$ using component-wise averages (see Fig. 2.15 of Chapter 2). This allows building scale-wise histograms of oriented gradients, where each bin q corresponds to the average response of the response map $h_{\sigma,q}(\boldsymbol{x})$ of its corresponding operator $\mathcal{G}_{\sigma,q}$ over $\boldsymbol{M}$. The properties of HOGs are summarized in Table 3.4.

3.2.2.4 Riesz transform

A more elegant approach to compute directional transitions between pixel values is to compute them in the Fourier domain instead of using pixel differences in Gaussian-smoothed images as presented in Eq. (3.8). This also provides the opportunity to easily compute higher-order image derivatives of order l as

$$\frac{\partial^l}{\partial x_d^l} f(\boldsymbol{x}) \overset{\mathcal{F}}{\longleftrightarrow} (\mathrm{j}\omega_d)^l \hat{f}(\boldsymbol{\omega}), \tag{3.10}$$

where $1 \leq d \leq D$. It can be noticed that differentiating an image along the direction x_d only requires multiplying its Fourier transform by $\mathrm{j}\omega_d$. Computing lth-order derivatives has an intuitive interpretation (*e.g.*, texture gradient for $l = 1$, curvature for $l = 2$), which makes them attractive for understanding the meaning of the texture measures in a particular medical or biological applicative context. However, a pure image derivative filter as computed in Eq. (3.10) is high-pass (because multiplied by ω_d) and accentuates high frequencies along x_d. Therefore it is desirable to implement image derivatives as all-pass filters, which is provided with the real Riesz transform[6] $\mathcal{R}\{f\}(\boldsymbol{x})$ as [15]

$$\boldsymbol{\mathcal{R}}\{f\}(\boldsymbol{x}) = \begin{pmatrix} \mathcal{R}_1\{f\}(\boldsymbol{x}) \\ \vdots \\ \mathcal{R}_D\{f\}(\boldsymbol{x}) \end{pmatrix} \overset{\mathcal{F}}{\longleftrightarrow} -\mathrm{j}\frac{\boldsymbol{\omega}}{||\boldsymbol{\omega}||}\hat{f}(\boldsymbol{\omega}). \tag{3.11}$$

[6] The Riesz transform is the multidimensional extension of the Hilbert transform.

Table 3.4 HOG properties

Operator linearity	No. The initial filtering operations are linear, but "aligning" the HOGs with the dominant directions $\theta_{\boldsymbol{x}_0}$ obtained from Eq. (3.9) is a nonlinear operation.
Handcrafted	Yes.
3D extension	Requires extending the gradient orientations to 3D with systematic sampling of angles θ, ϕ, which is proposed in [14].
Coverage of image directions	Complete.
Directionality and local rotation-invariance	Directional and locally rotation-invariant.
Characterization of the LOIDs	Yes. The HOGs provide MF representations oriented with $\theta_{\boldsymbol{x}_0}$ (see Eq. (3.7)).
Coverage of image scales	Incomplete and depends on the choices of DoG scales σ.
Band-pass	Yes.
Gray-level reduction	Not required.
Illumination-invariance	No. However, they are robust to changes in illumination because gradient operators computed on top of DoGs coefficients are band-pass.
Aggregation function	The average of the response maps of each HOG operator's responses can be used to build the scale-wise histogram of oriented gradients. This yields gradient-based MF representations (see Section 2.4.3 of Chapter 2).

It can be noticed by dividing the Fourier representation with the norm of $\boldsymbol{\omega}$ transforms Eq. (3.10) in D all-pass operators $\mathcal{R}_d$. For a fixed order L, the collection of higher-order all-pass image derivatives are defined in Fourier as

$$\widehat{\mathcal{R}^{\boldsymbol{l}}}\{f\}(\boldsymbol{\omega}) = (-\mathrm{j})^L \sqrt{\frac{L!}{l_1! \cdots l_D!}} \frac{\omega_1^{l_1} \cdots \omega_D^{l_D}}{\left(\omega_1^2 + \cdots + \omega_D^2\right)^{L/2}} \hat{f}(\boldsymbol{\omega}), \tag{3.12}$$

which yields a total of $\binom{L+D-1}{D-1} = \frac{(L+D-1)!}{L!(D-1)!}$ all-pass filters for all combinations of the elements l_d of the vector $\boldsymbol{l}$ as $|\boldsymbol{l}| = l_1 + \cdots + l_D = L$. The collection of Riesz operators of order L is denoted by $\boldsymbol{\mathcal{R}}^L$. A set of band-pass, multiscale, and multiorientation operator functions $g_{\sigma,\boldsymbol{l}}(\boldsymbol{x})$ can be obtained by simply applying the Riesz transform to circularly

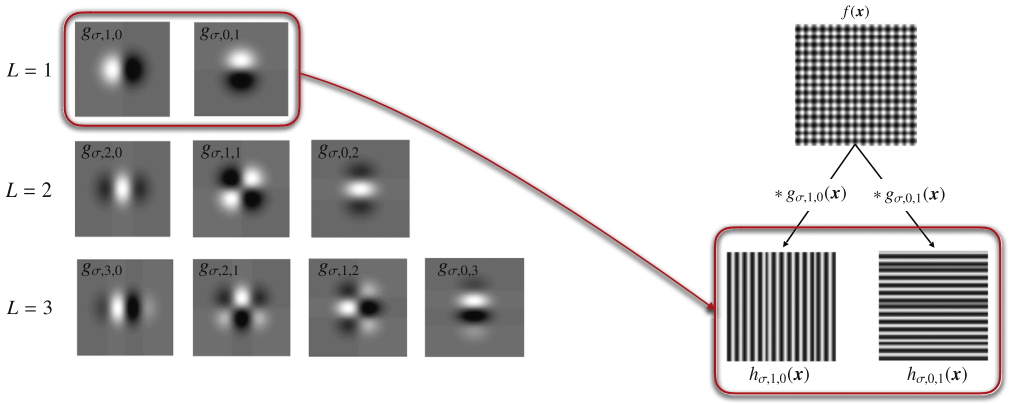

Figure 3.4 Two-dimensional Lth-order Riesz texture operators with functions $g_{\sigma,\boldsymbol{l}}(\boldsymbol{x})$ providing Lth-order image derivatives at a fixed scale σ. Applying $L = 1$ texture operators with functions $g_{\sigma,1,0}$ and $g_{\sigma,0,1}$ to the input texture f with convolution yields response maps $h_{\sigma,1,0}(\boldsymbol{x})$ and $h_{\sigma,0,1}(\boldsymbol{x})$ decomposing f into its vertical and horizontal directions, respectively. Qualitatively, $L = 1$ corresponds to gradient estimation whereas $L = 2$ estimates the Hessian. They both have intuitive interpretations (*e.g.*, texture slope for $L = 1$, curvature for $L = 2$), which makes them attractive for understanding the meaning of the texture measures in a particular medical or biological applicative context.

symmetric wavelets or multiscale filters, *e.g.*, the LoG filter g_σ (see Eq. (3.5)) as

$$g_{\sigma,\boldsymbol{l}}(\boldsymbol{x}) = \mathcal{R}^{\boldsymbol{l}}\{g_\sigma\}(\boldsymbol{x}).$$

Examples of 2D real Riesz operators and their application to a directional texture are shown in Fig. 3.4. They are implemented as a plugin for quantitative image analysis on the ePAD radiology platform (see Chapter 13). 3D real Riesz filters are depicted in Fig. 2.12 of Chapter 2 and are implemented in the QuantImage radiomics web platform[7] (see Chapter 12).

Riesz operators as defined in Eq. (3.12) are not locally rotation-invariant/equivariant. However, local rotation-invariance/equivariance and rich MF representations can be achieved in a convenient fashion through the most interesting property of Riesz texture operators, which is steerability. The Riesz operator functions $g_{\sigma,\boldsymbol{l}}(\boldsymbol{x})$ are steerable, which will be detailed in 2D in the following text. 2D steerability means that the responses of $g_{\sigma,\boldsymbol{l}}$ rotated by an angle θ_0 can be very efficiently computed with a linear combination of a finite number of basis elements, which is parameterized by a steering matrix $\mathbf{A}^L_{\theta_0}$ as

$$\boldsymbol{\mathcal{R}}^L\{g_\sigma\}(\mathbf{R}_{\theta_0}\boldsymbol{x}) = \mathbf{A}^L_{\theta_0}\boldsymbol{\mathcal{R}}^L\{g_\sigma\}(\boldsymbol{x}), \tag{3.13}$$

[7] https://radiomics.hevs.ch, as of March 1, 2017.

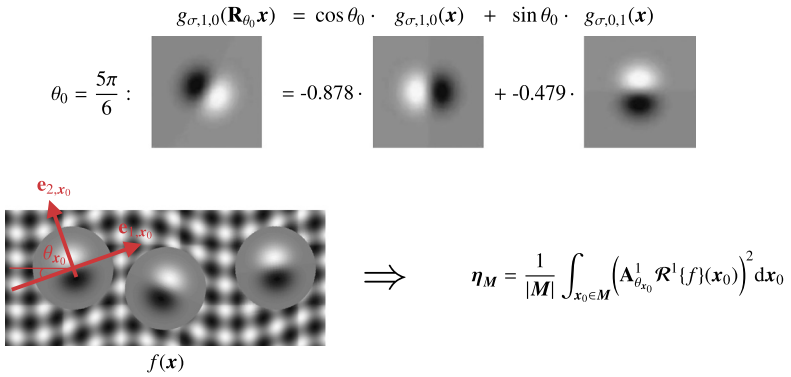

Figure 3.5 Top row: example of the steering of the first-order Riesz operator $g_{\sigma,1,0}$ at the position $\boldsymbol{x}_0 = \mathbf{0}$ with an angle $\theta_0 = \frac{5\pi}{6}$. Bottom row: rich MF representations of $f(\boldsymbol{x})$ can be obtained from locally steered Riesz filterbanks (exemplified for $L = 1$).

where g_σ is the circularly symmetric function used to control the spatial support of the operators. For order 1 (*i.e.*, $L = 1$), $\mathbf{A}^1_{\theta_0}$ is equal to the 2D rotation matrix $\mathbf{R}_{\theta_0}$. For each scale σ, rich D-dimensional MF texture representations can be obtained by finding the local angle $\theta_{\boldsymbol{x}_0}$ maximizing the response of one chosen Riesz operator function $g_{\sigma,l_1,\ldots,l_D}$ for each position $\boldsymbol{x}_0$ using steerability [16–20]. For instance, 2D angle maps corresponding to a texture function f can be obtained for $L = 1$ as

$$\theta_{\boldsymbol{x}_0} := \underset{\theta_0\in[0,2\pi)}{\arg\max}\Big(\cos\theta_0\cdot(g_{\sigma,1,0} * f)(\boldsymbol{x}_0) + \sin\theta_0\cdot(g_{\sigma,0,1} * f)(\boldsymbol{x}_0)\Big). \tag{3.14}$$

This is a particular case of the angle alignment framework introduced in Eqs. (3.3) and (3.4).

A collection of MF texture measurements $\boldsymbol{\eta}$ can be obtained from component-wise averages of the absolute values (or energies) of the Riesz wavelet coefficients steered with angle maps obtained from, *e.g.*, Eq. (3.14). Steerability of the real Riesz transform and its use for the construction of MF representations is illustrated in Fig. 3.5. The properties of real Riesz-based texture analysis are summarized in Table 3.5.

By slightly modifying the definition of the Riesz transform (see Eq. (3.12)) into its complex form, it is also possible to obtain two-dimensional texture descriptors that can linearly quantify the amount of local circular frequencies. The latter are implicitly modeled by LBP operators, which unfortunately require a binarization and discretization of the circular neighborhoods resulting in a potentially large loss of information (see Section 3.4). This is not the case when using the complex Riesz transform. The 2D lth order complex Riesz transform $\mathcal{R}^l_{\mathbb{C}}$ is defined in polar coordinates in Fourier as [27]

$$\widehat{\mathcal{R}^l_{\mathbb{C}}\{f\}}(\rho,\vartheta) = e^{\mathrm{j}l\vartheta}\hat{f}(\rho,\vartheta). \tag{3.15}$$

Table 3.5 Properties of real and complex Riesz texture representations, as well as Steerable Wavelet Machines (SWM)

Operator linearity	No. The filtering operations are linear, but "aligning" the filters either using operator steering or the complex magnitudes of CHs are nonlinear operations.
Handcrafted	Yes for real and complex Riesz. No for learned SWMs operators (see Section 3.2.3.1).
3D extension	Real Riesz representations are extended to 3D by considering the subspace of filters spanned by all combinations of partial derivatives relatively to $\{x_1, x_2, x_3\}$ [16,17] (see Eq. (3.12)). The extension of complex Riesz transforms (*i.e.*, CH) to three dimensions is not straightforward. Spherical harmonics-based representations can be considered [21,22].
Coverage of image directions	Complete.
Directionality and local rotation-invariance	Directional and not locally rotation-invariant in their initial form. However, rich directional and locally rotation-invariant representations can be obtained at a low computational cost by either using the complex magnitudes (complex Riesz) or steerability (real and complex Riesz as well as SWMs).
Characterization of the LOIDs	Yes. Moreover, rich and learned MF representations can be obtained with steerability and SWMs.
Coverage of image scales	Complete when used with circularly symmetric wavelet representations [18] (*e.g.*, Simoncelli [23], Meyer [24], Shannon [25]).
Band-pass	Yes.
Gray-level reduction	Not required.
Illumination-invariance	No. However, they are robust to changes in illumination when Riesz operators are based on band-pass circularly symmetric primal functions.
Aggregation function	The average or covariances of the absolute values (or energies) of each Riesz operator response can be used [19,26].

Using a process similar to construct real Riesz filters, applying the complex Riesz transform to circularly symmetric filters or wavelets (*e.g.*, LoG filters, Eq. (3.5), or circularly symmetric Simoncelli wavelets, Eq. (2.2) of Chapter 2) yields steerable, complex, and band-pass Circular Harmonic (CH) filters or wavelets $g_{\sigma,l}(\boldsymbol{x}) = \mathcal{R}^l_{\mathbb{C}}\{g_\sigma\}(\boldsymbol{x}) \in \mathbb{C}$ with $l = 0, \ldots, L$ [28,29]. CHs stand out as the canonical representation of steerability [27],

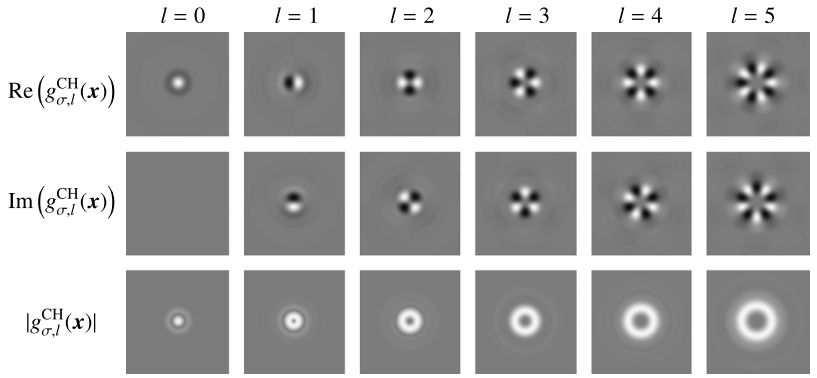

Figure 3.6 Impulse responses of Circular Harmonics (CH) of order $l = 0, \ldots, 5$.

which implies that every steerable representation $g_{\sigma,L,c}$ with $c = 1, \ldots, C$ components per scale can be obtained with a $C \times (L+1)$ complex shaping matrix $\mathbf{U}$ as

$$\begin{pmatrix} g_{\sigma,L,1} \\ \vdots \\ g_{\sigma,L,C} \end{pmatrix} = \mathbf{U} \begin{pmatrix} g_{\sigma,0} \\ \vdots \\ g_{\sigma,L} \end{pmatrix}. \tag{3.16}$$

As a consequence, any steerable representation can be obtained with a specific shaping matrix $\mathbf{U}$ (*e.g.*, gradient and Hessian real Riesz, Simoncelli's steerable pyramid). Examples of CH filters are shown in Fig. 3.6. Rich and locally rotation-invariant descriptions of the LOIDs can be obtained with a very cheap computational cost by computing the complex magnitudes of the response maps $h_{\sigma,l=0,\ldots,L}(\boldsymbol{x})$. The collection of CH with various orders $l = 0, 1, \ldots, L$ defines an orthonormal system corresponding to a Fourier basis for circular frequencies up to a maximum order L. The process is similar to LBP as presented in [30] but with the desirable properties of a fully linear approach (*e.g.*, no gray-level transformation required) besides the final step computing the complex magnitudes. However, absolute values of CH filters do not define MF representations since the interharmonic phase is lost by taking the magnitude of the responses of the operators. Rich MF representations can be obtained by steering all harmonics with a unique local orientation $\theta_{\boldsymbol{x}_0}$, as proposed in [26]. Although more computationally expensive than using the magnitudes of the operators, steering them is relatively cheap because CH are self-steerable, resulting in block-diagonal steering matrices $\mathbf{A}^L_{\theta_0}$ for multiorder representations.

Texture measures $\boldsymbol{\eta}$ can be obtained from the averages or covariances of absolute values (or energies) of steered or nonsteered Riesz response maps over a ROI $\boldsymbol{M}$ [19]. The properties of complex Riesz-based texture analysis are summarized in Table 3.5.

3.2.3 Learned filters

All texture operators described and discussed in the previous sections are handcrafted. This means that the type of texture information extracted by theses operators is assumed to be relevant for the texture analysis task in hand. Therefore the design of these operators in terms of the coverage of image scales and directions was based on prior assumptions (*e.g.*, ad hoc or based on theoretic guidelines, see Section 1.3.1 of Chapter 1). Whereas classical approaches use machine learning on top of handcrafted representations, more recent approaches proposed to derive the design of the operators from data to identify the combinations of scales and directions that are optimal for the texture analysis task in hand. When compared to handcrafted operators, learned ones reduce the risk of unnecessary modeling of texture properties that are not related to the targeted application (see Fig. 2.5 of Chapter 2). They eliminate human bias to include or exclude arbitrary operator scales and directions. It is worth noting that learned approaches still predominantly use handcrafted components of the aggregation function (*e.g.*, ReLU, sigmoid, pooling, see Section 3.2.3.3). Three important approaches for learning convolutional texture filters are discussed in this section, including Steerable Wavelet Machines (SWM, Section 3.2.3.1), Dictionary Learning (DL, Section 3.2.3.2), and deep Convolutional Neural Networks (CNNs, Section 3.2.3.3).

3.2.3.1 Steerable Wavelet Machines (SWM)

In [31–33] and [26], we proposed to use SVMs to learn optimally discriminant linear combinations of real or complex Riesz operators. The most interesting property of this approach called Steerable Wavelet Machines (SWM) is to combine the flexibility of learned representations with steerability,[8] *i.e.*, we learned the lines $\boldsymbol{u}_c$ of the shaping matrix $\mathbf{U}$ (see Eq. (3.16)) using one-versus-all classification configurations. This yields class-specific steerable signatures of the essential stitches of biomedical tissue (allowing locally rotation-invariant/equivariant descriptions of the LOIDs) that can be used for building data-specific MF representations. This approach is based on handcrafted representations of image scales based on dyadic circularly symmetric wavelet functions and we are currently extending the learning to image scales as well. The learned operators are limited to the span of Riesz representations. However, it was observed that a limited number of circular harmonics (*e.g.*, $L = 3, \ldots, 10$) yields optimal texture representations for classification. In addition, the span of CH becomes less and less restricted with large values of L and can represent any function for $L \to \infty$ since it can be interpreted as a Fourier transform for circular frequencies. The properties of SWMs are summarized in Table 3.5.

[8] Linear combinations of steerable subspaces are themselves steerable.

3.2.3.2 Dictionary Learning (DL)

Recent popular approaches propose to fully learn operators from data (*e.g.*, a collection of bounded discretized texture functions $\mathcal{I} = \{f_1(\boldsymbol{\xi}), f_2(\boldsymbol{\xi}), \ldots, f_I(\boldsymbol{\xi})\}$) with very little constraints besides their maximal spatial support $G_1 \times \cdots \times G_D$. A first notable example is unsupervised Dictionary Learning (DL). The learned set of operators will depend on the learning criteria used [34]. Basic dimensionality reduction methods such as Principal Component Analysis (PCA), Independent Component Analysis (ICA), or K-means clustering based on the $|\mathbf{G}|$ pixel values of $G_1 \times \cdots \times G_D$ patches $\gamma_p(\boldsymbol{\xi})$ can be used to derive N essential *atoms* $g_{n,\mathcal{I}}(\boldsymbol{\xi})$, which can be further used as operator functions [35]. PCA requires that the atoms are orthogonal to each other (*i.e.*, it removes correlation between them). ICA minimizes the correlation as well as higher-order dependence between atoms, which are not necessarily orthogonal. K-means finds N prototype atoms $g_{n,\mathcal{I}}(\boldsymbol{\xi})$ from clustering of the $|\mathbf{G}|$-dimensional space spanned by the pixel values of the patches $\gamma_p(\boldsymbol{\xi})$. The atoms are also called *textons* [36,37], which were identified as the elementary units of preattentive human texture perception [1]. Textons relate to texture primitives (see Section 1.2.4 of Chapter 1) and to the LOIDs (see Section 2.4.1 of Chapter 2). The process for extracting 3×3 textons from a texture function $f(\boldsymbol{\xi})$ is illustrated in Fig. 3.7.

Another successful unsupervised approach was based on creating a collection of atoms from which we can reconstruct the patches $\gamma_p(\boldsymbol{\xi})$ with a vector of coefficients

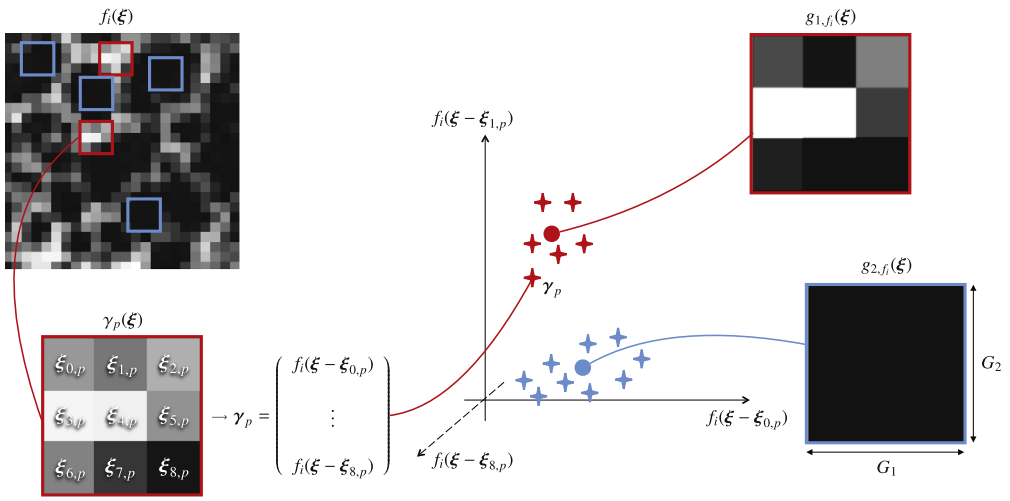

Figure 3.7 Dictionary learning (DL) process from one single image $f_i(\boldsymbol{\xi})$. A collection of 3×3 patches $\gamma_p(\boldsymbol{\xi})$ are extracted from f_i. The patch matrices are vectorized to create vectors $\boldsymbol{\gamma}_p$. K-means clustering is used in the space spanned by instance vectors $\boldsymbol{\gamma}_p$ to find N atoms $g_{n,f_i}(\boldsymbol{\xi})$ that are representative of f_i. These atoms are called *textons* and can be further used as texture operator functions.

$\boldsymbol{\alpha}_p$ containing a minimum number of nonzero elements [38,39]. Every atoms $g_{n,\mathcal{I},\lambda}$ are vectorized, transposed and piled up to create the $|\boldsymbol{G}| \times N$ dictionary matrix $\mathbf{D}$ with N atoms. Solving the following optimization problem for a collection of P patches can be used to compute both $\mathbf{D}$ and α as

$$\mathbf{D} := \underset{\mathbf{D},\boldsymbol{\alpha}}{\arg\min} \sum_{p=1}^{P} ||\boldsymbol{\gamma}_p - \mathbf{D}\boldsymbol{\alpha}_p||_2^2 + \lambda||\boldsymbol{\alpha}_p||_1, \tag{3.17}$$

where $||\cdot||_1$ is the ℓ_1-norm[9] and λ is controlling the sparsity of the atom coefficients $\boldsymbol{\alpha}$. Additional topological constraints were added in Topographic Independant Component Analysis (TICA) to take into account the important knowledge that patches that are spatially close to each other have similar statistical properties [40]. All texture operators dictionaries $\mathbf{D}$ learned with unsupervised approaches (*e.g.*, K-means, PCA, ICA, TICA, reconstruction, see Eq. (3.17)) are not directly optimizing a discriminative criteria, which means that they are not necessarily optimal for biomedical texture classification of a set of considered tissue types. To tackle this issue, supervised dictionary learning was proposed [38], where $\mathbf{D}$ is obtained as

$$\mathbf{D} := \underset{\mathbf{D},b,\boldsymbol{\alpha}}{\arg\min} \sum_{p=1}^{P} \mathcal{C}\left(\gamma_p,\ \boldsymbol{\gamma}_p^T \mathbf{D}\boldsymbol{\alpha}_p + b\right) + \lambda||\mathbf{D}||_F^2, \tag{3.18}$$

where $\boldsymbol{\gamma}_p$ are the vectorized training patches and their labels $\left(\gamma_p \in \{-1, +1\}\right)_{p=1,\ldots,P}$, $\mathcal{C}(\gamma_p, \check{\gamma}_p)$ is the cost function,[10] b the bias, and $||\cdot||_F$ the Frobenius matrix norm.[11] An inherent challenge of DL methods is the large number of free parameters (*i.e.*, features or variables) to learn which is equal the size (*i.g.*, measure) of the dictionary $|\boldsymbol{G}| \times N$, resulting in a very high-dimensional feature space. This typically requires a very large collection of training patches to respect the recommended ratio between feature dimensionality and number of training instances equal to ten [41]. As an order of magnitude, 1,210,000 training patches are required to learn 1000 11×11 atoms while respecting this ratio. A collection of N texture measurements $\boldsymbol{\eta}$ can be obtained by computing the averages of the absolute values or energies of the response maps $h_n(\boldsymbol{\xi}) = (g_n * f)(\boldsymbol{\xi})$ within a ROI $\boldsymbol{M}$. The properties of DL-based texture analysis are summarized in Table 3.6.

[9] The ℓ_1-norm of a vector $\boldsymbol{x}$ is $||\boldsymbol{x}||_1 = \sum_{d=1}^{D} |x_d|$, whereas the ℓ_2-norm is $||\boldsymbol{x}||_2 = \sqrt{\sum_{d=1}^{D} x_d^2}$.

[10] The cost function defines how label prediction errors of the estimations $\check{\gamma}_p$ are penalized. The logistic loss function is used in [38] and is defined as $\mathcal{C}(\gamma_p, \check{\gamma}_p) = \log(1 + e^{-\gamma_p \check{\gamma}_p})$.

[11] The Frobenius matrix norm is defined as $||\mathbf{A}||_F = \sqrt{\sum_{i=1}^{I} \sum_{j=1}^{J} a_{i,j}^2}$, where $a_{i,j}$ are the elements of $\mathbf{A}$.

Table 3.6 Properties DL texture operators

Operator linearity	Yes.
Handcrafted	No. The texture operators functions $g_n(\boldsymbol{\xi})$ (*i.e.*, the dictionary atoms) are fully derived from a collection of training texture functions $\mathcal{I}$.
3D extension	Straightforward. It requires vectorizing 3D image patches $\gamma_p(\boldsymbol{\xi})$. However, the number of free parameters will grow cubically.
Coverage of image directions	Complete, but only over the restricted spatial support $\boldsymbol{G}$ of the atoms.
Directionality and local rotation-invariance	Directional and not locally rotation-invariant. Data augmentation can be used to improve robustness to input rotation, but it has undesirable effects (see Section 3.2.3.4).
Characterization of the LOIDs	Yes, but not with local rotation-invariance.
Coverage of image scales	Incomplete. Typical spatial supports $\boldsymbol{G}$ of the atoms are 3×3, 5×5, 11×11, which is much smaller than the spatial supports $\boldsymbol{F}$ of biomedical texture functions.
Band-pass	No.
Gray-level reduction	Not required.
Illumination-invariance	No. The texture operators functions are not band-pass filters.
Aggregation function	The average of the absolute values (or energies) of each atom response can be used.

3.2.3.3 Deep Convolutional Neural Networks (CNN)

One major and very successful work on filter learning are CNNs and their deep architectures[12] [43–47]. Review and applications of deep learning in texture analysis for tissue image classification are further detailed in Chapters 4, 9, and 10. In a nutshell, deep CNNs consist of a cascade of Q convolutional layers, where each of the latter typically contains:

- **(i)** a multichannel convolution of the input $f_{n',q}(\boldsymbol{\xi})$[13] with a set of N $G_1 \times \cdots \times G_D$ multichannel operator functions $g_{n,n',q}(\boldsymbol{\xi})$, called *receptive fields*,
- **(ii)** a simple pointwise nonlinear gating $\mathcal{V}(x)$ of the response map $h_{n,q}(\boldsymbol{\xi})$,
- **(iii)** an optional cross-channel normalization [45] and,

[12] A notable exception is the Scattering Transform (ST) [42], which is a handcrafted deep CNN.

[13] The first layer $q = 1$ will use the original image $f_{n',1}(\boldsymbol{\xi})$ as input, whereas the next layers $q = 2, \ldots, Q$ will use the $n' = 0, \ldots, N'$ outputs (*i.e.*, *channels*) $f_{n',q-1}(\boldsymbol{\xi})$ of the previous layer as input for the convolution.

(iv) a pooling operation resulting in a down- or up-sampling of $h_{n,q}(\boldsymbol{\xi})$.

In 2D, the multichannel convolution in a layer q between the input $f_{n',q-1}(\boldsymbol{\xi})$ and the set of N multichannel filters $g_{n,n',q}(\boldsymbol{\xi})$ is

$$h_{n,q}(\boldsymbol{\xi}) = \sum_{n'=1}^{N'} (g_{n,n',q} * f_{n',q-1})(\boldsymbol{\xi}). \tag{3.19}$$

(3.19) yields a multichannel response map $h_{n,q}(\boldsymbol{\xi})$ with a dimensionality of $F_1 \times F_2 \times N'$ for a 2D input function of domain $\boldsymbol{F}$. The convolution is two-dimensional, where no cross-channel convolution is carried out. Popular examples of nonlinear gating functions $\mathcal{V}(x)$ are the Rectified Linear Unit (ReLU)

$$\mathcal{V}_{\mathrm{ReLU}}(x) = \max(0, x),$$

the sigmoid function

$$\mathcal{V}_{\mathrm{sig}}(x) = \frac{1}{1 + e^{-x}},$$

the absolute value and the energy. The aim of the pooling operation is to down- or up-sample feature maps to achieve a multiscale analysis throughout the cascade of convolutional layers. Whereas the upsampling operation is often straightforward (no specific superresolution approach [46]), the downsampling requires using a criteria on which value to keep over local patches (*e.g.*, maximum, minimum, median, sum, average). The structure of one convolutional layer is depicted in Fig. 3.8.

Deep CNN architectures consist of a cascade of convolutional layers containing a large collection of operator functions $g_{n,n',q}(\boldsymbol{\xi})$ learned to minimize a cost function $\mathcal{C}(y_p, \breve{y}_p)$. The total number of free parameters is equal to the product of the number of channels per operator N', the number of operator per layer N, the size of the spatial support $|\boldsymbol{G}|$ of the operators, and the number of layers Q, if we assume that N, N', and $|\boldsymbol{G}|$ are identical for all layers. It is very common for CNN architectures designed for classification to have a final *Fully Connected* (FC) layer that computes the decision value based on a linear combination of all elements of the final feature maps $h_{n,Q}(\boldsymbol{\xi})$. As an example, the complete forward function $u_{\mathrm{forward}}(f(\boldsymbol{\xi}))$ of a classification CNN transforms the input texture $f_p(\boldsymbol{\xi})$ into an estimated class label $\breve{y}_p$[14] is a composition of all functions u_q from all layers as

$$u_{\mathrm{forward}}(f(\boldsymbol{\xi}); \boldsymbol{g}): \quad \boldsymbol{F} \to \mathbb{R}, \tag{3.20}$$

$$\breve{y}_p = u_{\mathrm{forward}}(f_p(\boldsymbol{\xi}); \boldsymbol{g}) = u_{\mathrm{FC}}(\cdots u_2(u_1(f_p(\boldsymbol{\xi}); \boldsymbol{g}_1); \boldsymbol{g}_2)\cdots); \boldsymbol{g}_{\mathrm{FC}}), \tag{3.21}$$

[14] $\breve{y}_p$ can be binary (*e.g.*, $\breve{y}_p \in \{-1, +1\}$) or continuous (*e.g.*, probability $\breve{y}_p \in [0, 1]$, or $\breve{y}_p \in \mathbb{R}$). We will assume that $\breve{y}_p \in \mathbb{R}$ in our example.

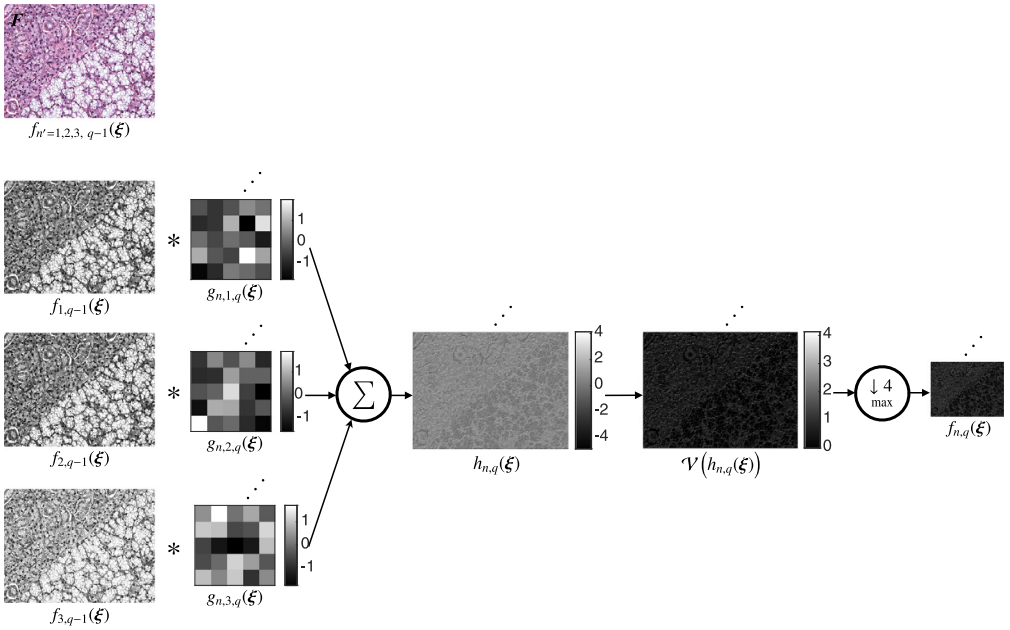

Figure 3.8 Structure of one typical convolutional layer of CNNs. The multichannel output of the previous layer $q-1$ constitutes the input $f_{n',q-1}(\boldsymbol{\xi})$ of the layer q. A multichannel convolution is carried out between the channels of $f_{n',q-1}$ and the collection of $n = 1, \ldots, N$ operator functions $g_{n,n',q}$ and yields a collection of response maps $h_{n,q}(\boldsymbol{\xi})$ as detailed in Eq. (3.19). The latter undergo a simple pointwise nonlinear gating $\mathcal{V}(x)$, followed by a pooling operation to output the final feature maps $f_{n,q}(\boldsymbol{\xi})$. In the example above the 5×5 operator functions $g_{n,n',q}$ are initialized with random values following a normal distribution with zero mean. The nonlinear gating function $\mathcal{V}$ is a ReLU, and the pooling operation is a $4\times$ downsampling process (noted as $\downarrow 4$) where the maximum value is kept over 4×4 patches and a distance (or *stride*) of 4 between their centers.

where $\boldsymbol{g}_q$ is the collection of free parameters of the operator functions $g_{n,n',q}(\boldsymbol{\xi})$ in the layer q, and $\boldsymbol{g}$ is the total collection of free parameters of the model. When CNN architectures are designed for segmentation (*e.g.*, U-Net [46]), the output of the forward function is an estimated segmentation map $\breve{\gamma}_p(\boldsymbol{\xi})$. Based on a set of training textures from which the true labels γ_p are known, it is possible to compute the prediction errors between $\breve{\gamma}_p$ and γ_p. The cost of this error is determined by the cost function $\mathcal{C}(\gamma_p, \breve{\gamma}_p)$. Typical $\mathcal{C}(\gamma_p, \breve{\gamma}_p)$ (also called *loss* function) are the logistic loss function for classification, or Euclidean loss for regression. The total cost function $\mathcal{C}_{\text{tot}}$ over the entire training set (called *epoch*) is accumulating errors over instances p as

$$\mathcal{C}_{\text{tot}}(\boldsymbol{g}) = \frac{1}{P}\sum_{p=1}^{P} \mathcal{C}\left(\gamma_p, u_{\text{forward}}(f_p(\boldsymbol{\xi}); \boldsymbol{g})\right).$$

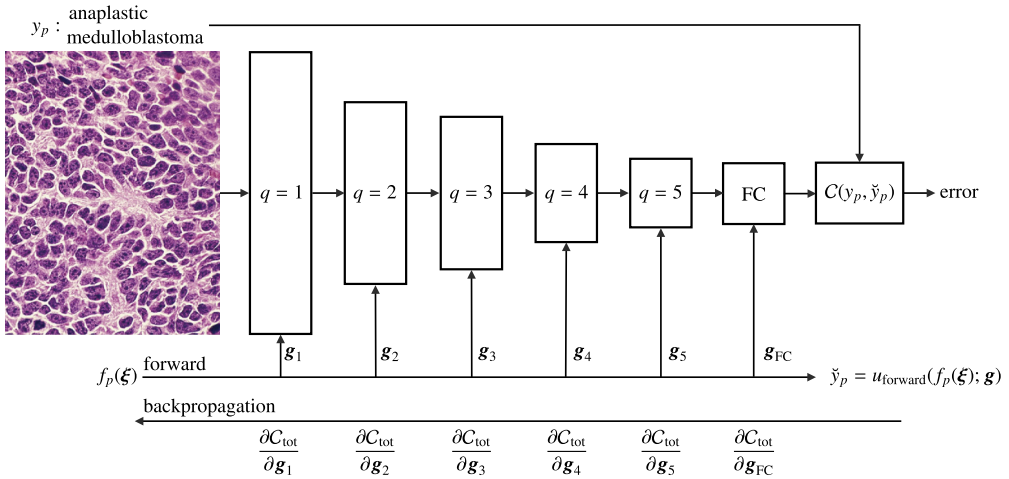

Figure 3.9 CNN architecture with a cascade of $Q = 5$ convolutional and one fully connected layer. The forward function $u_{\mathrm{forward}}(f(\boldsymbol{\xi}); \boldsymbol{g})$ is a composition of layer-wise functions u_g and transforms an input image $f_p(\boldsymbol{\xi})$ into a predicted label $\check{y}_p$. The cost function C_{tot} of accumulated errors over the entire training set is minimized through several epochs using backpropagation to update the free parameters $\boldsymbol{g}$. The latter allows learning the optimal profile of the convolutional operator functions $g_{n,n',q}$.

After computing the total cost for one epoch t, the parameters of the model $\boldsymbol{g}_t$ are modified to minimize the total loss as

$$\boldsymbol{g}_{t+1} = \boldsymbol{g}_t - \beta_t \frac{\partial C_{\mathrm{tot}}}{\partial \boldsymbol{g}}(\boldsymbol{g}_t), \tag{3.22}$$

where β_t is the learning rate at the epoch t. Solving Eq. (3.22) learns the operator functions $g_{n,n',q}(\boldsymbol{\xi})$ and is called *backpropagation*. $\frac{\partial C_{\mathrm{tot}}}{\partial \boldsymbol{g}}$ is computed using the chain rule, which allows combining the derivatives of the loss with respect to the parameters $\boldsymbol{g}_q$ of each consecutive layer [43]. Solving Eq. (3.22) through a total number T of epoch can be efficiently carried out with gradient descent optimization techniques. An example of a CNN architecture, its forward function and backpropagation is illustrated in Fig. 3.9. CNN-based texture analysis do not yield texture measurements *per se* as it directly outputs a tissue class probability or segmentation map. The aggregation function results from a cascade of pooling operations and, if applicable, the FC layer. The number of free parameters can be extremely large. For instance, a network with ten layers, ten 11×11 operators per layer with ten channels and with a 50×50 FC layer leads to more than 10^8 free parameters. It therefore requires very large training sets to correctly learn the model, which is most often difficult to get in biomedical imaging. Approaches called *transfer learning* or *fine tuning* are often

Table 3.7 Properties of deep CNNs for biomedical texture analysis

Operator linearity	Yes. The operators $\mathcal{G}_{n,n',q}$ are linear, but the full forward function is not.
Handcrafted	No.
3D extension	Requires using 3D operators, which was recently proposed in [51–53]. Training 3D CNNs is challenging because the number of parameters grows cubically.
Coverage of image directions	Complete, but only over the discrete spatial support $\mathbf{G}$ of the operators.
Directionality and local rotation-invariance	Directional and not locally rotation-invariant. Data augmentation can be used to improve robustness to input rotation, but it has undesired effects (see Section 3.2.3.4).
Characterization of the LOIDs	Yes, but not in a locally rotation-invariant fashion.
Coverage of image scales	Near-to-complete with the spatial support of the operators and the cascade of down- or up-sampling pooling operations. The coverage is not systematic though and do not respect Parseval's identity. A truly multiscale CNNs was proposed in [54].
Band-pass	No.
Gray-level reduction	Not required.
Illumination-invariance	No. The texture operator functions are not band-pass filters.
Aggregation function	The aggregation function results from a cascade of pooling operations and, if applicable, the FC layer.

used is to tackle this problem [48,49] (see Section 4.4.3.3 of Chapter 4). It consists of reusing and slightly adapting models trained with millions of images from other domains (*e.g.*, photographs in ImageNet [50]). In the particular case of biomedical texture analysis, reusing these models is risky because the types of invariances learned by networks based on photographic imagery resulting from scene captures obtained with varying viewpoints is very different from the ones desirable in BTA (see Section 1.3.3 of Chapter 1). An interesting observation is that when large datasets are available, the type of operators learned in the first layers of deep CNNs share very similar properties with handcrafted convolutional filters such as LoG, Gabor, MR8, and Riesz. The properties of CNNs for biomedical texture analysis are summarized in Table 3.7.

3.2.3.4 Data augmentation

Data augmentation aims to tackle two important limitations of filter learning approaches relying on a large number of free parameters: DL and CNNs [45,48,49,46]. First, both approaches require very large training sets to respect an acceptable ratio between the number of free parameters and training instances and allow suitable generalization performance (*i.e.*, limiting the risk of overfitting the training data). Second, neither of the methods are invariant to local scalings or local rotations. Whereas invariance to image scale is most often not desirable for biomedical texture analysis, invariance to local rotations is fundamental (see Section 1.3.3 of Chapter 1). Data augmentation consists of generating additional training instances from geometric transformations (including nonrigid) of the available ones, and to further train the models with these new instances. It provided very important performance gain in various applications [45,48], including biomedical [46]. However, it is obvious that forcing convolutional operators to recognize sheared, scaled, and rotated versions of the texture classes without augmenting the degrees of freedom will result in a strong decrease of the specificity of the model. In particular, forcing invariance to local rotations will make operators insensitive to directions, which was illustrated in [55]. A quantitative comparison between circularly symmetric and locally aligned operators (*e.g.*, MF representations) is detailed in Section 2.4.4 of Chapter 2. In this context, learning steerable filters allows adapting the texture representation while keeping local rotation-invariance and high specificity of the operators [31,26].

3.3 GRAY-LEVEL MATRICES

Gray-Level Matrices (GLM) constitute a large and popular group of non-linear texture operators. There are mainly three approaches based on GLMs: Gray-Level Cooccurrence Matrices (GLCM), Gray-Level Run-Length Matrices (GLRLM), and Gray-Level Size Zone Matrices (GLSZM). Their simplicity is at the origin of their popularity and many implementations can be found (*e.g.*, MaZda,[15] see Chapter 11, LIFEx,[16] Matlab,[17] Scikit-image,[18] the QuantImage web platform,[19] see Chapter 12, ePAD's Quantitative Feature Explore (QFExplore) texture plugins,[20] see Chapter 13). They are often used together. They are based on discrete texture functions $f(\boldsymbol{\xi})$ and non-linear operators $\mathcal{G}_n$. Their extensions to 3D are straightforward. However, they suffer from several imperfections. A major one is the nonsystematic coverage and poor preservation of image scales

[15] http://www.eletel.p.lodz.pl/programy/mazda/, as of March 4, 2017.
[16] http://www.lifexsoft.org, as of November 20, 2016.
[17] http://www.mathworks.com/help/images/ref/graycomatrix.html, as of November 20, 2016.
[18] http://scikit-image.org/docs/stable/api/skimage.feature.html, as of February 26, 2017.
[19] https://radiomics.hevs.ch, as of March 4, 2017.
[20] https://epad.stanford.edu/plugins, as of June 20, 2017.

and directions, especially when the spatial supports $G_{1,n} \times \cdots \times G_{D,n}$ of their operators are large. Therefore, they are best suited for applications where the size of the ROIs are small (*e.g.*, small lesions in low-resolution medical images). They also require drastic reductions of gray levels, where the values of $f(\boldsymbol{\xi}) \in \mathbb{R}$ are quantized with Λ values as $f_\Lambda(\boldsymbol{\xi}) \in \{1, \ldots, \Lambda\}$. The quantization is typically based on $\Lambda = 8, 16, 32$, which results in an important potential loss of information when analyzing rich image contents coded with 12 to 16 bits.[21] Their properties are reviewed and detailed in the three following Subsections 3.3.1, 3.3.2, and 3.3.3.

3.3.1 Gray-Level Cooccurrence Matrices (GLCM)

GLCMs [56] can be seen as a collection of operators mapping the discretized and quantized image function $f_\Lambda(\boldsymbol{\xi})$ to binary output values at a position $\boldsymbol{\xi}_0$ as

$$\mathcal{G}_{\Delta \boldsymbol{k}}^{\lambda_i,\lambda_j}\{f_\Lambda\}(\boldsymbol{\xi}_0) = \begin{cases} 1 & \text{if } f_\Lambda(\boldsymbol{\xi}_0) = \lambda_i \text{ and } f_\Lambda(\boldsymbol{\xi}_0 + \Delta \boldsymbol{k} \circ \Delta \boldsymbol{\xi}) = \lambda_j, \\ 0 & \text{otherwise,} \end{cases} \tag{3.23}$$

where λ_i and λ_j are the pixel values at positions $\boldsymbol{\xi}_0$ and $(\boldsymbol{\xi}_0 + \Delta \boldsymbol{k} \circ \Delta \boldsymbol{\xi})$, respectively. $\Delta \boldsymbol{k}$ and $\Delta \boldsymbol{\xi}$ contains the dimension-wise shifts and sampling steps, respectively (see Section 1.2.2 of Chapter 1). $\Delta \boldsymbol{k} \circ \Delta \boldsymbol{\xi}$ denotes the element-wise product[22] between the two vectors as

$$\Delta \boldsymbol{k} \circ \Delta \boldsymbol{\xi} = \begin{pmatrix} \Delta k_1 \cdot \Delta \xi_1 \\ \vdots \\ \Delta k_D \cdot \Delta \xi_D \end{pmatrix}.$$

The aggregation function is integrative. It counts the responses of the operators and organizes them in square *cooccurrence* matrices of dimensions Λ^2 indexed by (λ_i, λ_j), where Λ is the number of gray-levels. A series of scalar texture measurements[23] $\boldsymbol{\eta}$ is obtained by computing statistics (*e.g.*, contrast, correlation, entropy) from the co-occurrence matrices. Their properties are summarized in Table 3.8. GLCMs are not invariant to local rotations, but the later is often approximated by either regrouping the counts of operators over all directions in a shared matrix, or by averaging scalar texture measurements from cooccurrences matrices obtained with different directions (see Fig. 3.10).

[21] Image pixels encoded with 16 bits can take more than 65,000 possible values.
[22] It is also called the *Hadamard* product.
[23] They are often called *Haralick features* [56].

Table 3.8 GLCM properties

Operator linearity	Nonlinear.
Handcrafted	Yes.
3D extension	Straightforward: displacements $\Delta\boldsymbol{k} \circ \Delta\boldsymbol{\xi}$ between pixels can live in subsets of either $\mathbb{R}^2$ or $\mathbb{R}^3$.
Coverage of image directions	Incomplete: typical directions used are $\theta_{1,\ldots,4} = 0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}$ in 2D.
Directionality and local rotation-invariance	Unidirectional and not rotation-invariant. However, local rotation-invariance is often approximated by either regrouping the counts of operators over all directions in a shared matrix, or by averaging scalar texture measurements from cooccurrences matrices obtained with different directions (see Fig. 3.10).
Characterization of the LOIDs	No.
Coverage of image scales	Incomplete: typical displacements values are $\|\|\Delta\boldsymbol{k}\|\| \approx 1, 2, 3$. Moreover, displacements along image diagonals (*e.g.*, $\theta_2 = \frac{\pi}{4}$ and $\theta_4 = \frac{3\pi}{4}$) are often considered to have integer norms (*e.g.*, $1, 2, 3$) instead of their actual values (*e.g.*, $\sqrt{2}, 2\sqrt{2}, 3\sqrt{2}$), resulting in anisotropic descriptions of image scales.
Band-pass	Qualitatively similar in the sense that the mean value of the image is not influencing the output value of the operator. It is worth noting that the transfer function is not defined in the Fourier domain because the operator is nonlinear.
Gray-level reduction	A reduction is required to avoid obtaining very large and sparse cooccurrence matrices. Typical gray-level reductions are $\Lambda = 8, 16, 32$.
Illumination-invariance	No, although it is approximated by reducing the number of gray-levels Λ.
Aggregation function	Integrative: counts the binary responses of each GLCM operator $\mathcal{G}^{\lambda_i,\lambda_j}_{\Delta\boldsymbol{k}}$ over $\boldsymbol{M}$ and organizes them in a corresponding cooccurrence matrix indexed by λ_i and λ_j. Scalar texture measurements $\boldsymbol{\eta}$ called *Haralick features* are obtained from statistics of the cooccurrence matrices.

3.3.2 Gray-Level Run-Length Matrices (GLRLM)

GLRLMs [57] count the number of aligned pixels (called stride) with equal gray-level value λ, length $\gamma \in \mathbb{N}^*$ and direction θ. Their operators $\mathcal{G}_{\lambda,\gamma,\theta}\{f_\Lambda\}(\boldsymbol{\xi}_0)$ yield a value of 1

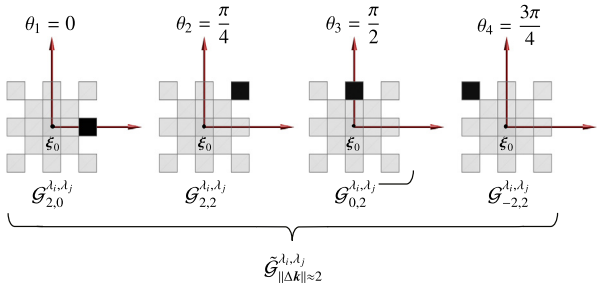

Figure 3.10 Approximate local rotation-invariance with GLCMs: the responses of GLCM operators $\mathcal{G}^{\lambda_i,\lambda_j}_{\Delta k_1,\Delta k_2}$ are combined over four directions $\theta_{1,\ldots,4}$ to reduce the directional sensitivity of the operators and approximate locally rotation-invariant texture analysis. In 2D, the angle between the vectors $\Delta\boldsymbol{k}$ and $\boldsymbol{e}_1 = (1, 0)$ is noted θ (see Section 2.2 of Chapter 2).

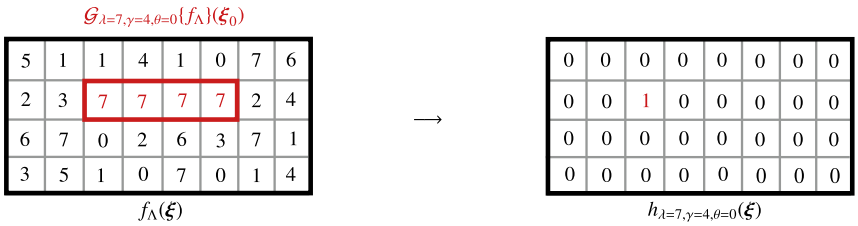

Figure 3.11 Application of a GLRLM operator $\mathcal{G}_{\lambda=7,\gamma=4,\theta=0}$ to an input image f_Λ. Its response map $h_{\lambda=7,\gamma=4,\theta=0}(\boldsymbol{\xi})$ highlights the presence of the sought stride with gray-level 7, length 4 and orientation 0 at the position $\boldsymbol{\xi}_0$.

if a stride starting at $\boldsymbol{\xi}_0$, of length γ, and direction θ is detected, and 0 otherwise. An example of a GLRLM operator and its response map is depicted in Fig. 3.11.

The aggregation function counts the number of strides detected with the corresponding operator and organizes them in a *run-length* matrix indexed by λ and γ. The size of this matrix is the number Λ of gray-levels considered, times the number of stride lengths tested. The texture measures $\boldsymbol{\eta}$ are obtained from statistics of the run-length matrices (*e.g.*, short run emphasis, run-length nonuniformity, see [58]). Their properties are summarized in Table 3.9.

3.3.3 Gray-Level Size Zone Matrices (GLSZM)

GLSZMs [59] are extending the concept of GLRLM to zone areas or volumes. Their operators $\mathcal{G}_{\lambda,\zeta}\{f_\Lambda\}(\boldsymbol{\xi}_0)$ yield a binary value of 1 if $\boldsymbol{\xi}_0$ belongs to a uniform zone with gray-level λ, *i.e.*, any zone with area equal to ζ containing $\boldsymbol{\xi}_0$, and area $\zeta \in \mathbb{N}^*$, and $\boldsymbol{\xi}_0$ does not belong to a uniform zone with larger area.

The aggregation function counts the number of zones detected with the corresponding operator and organizes them in a *size-zone* matrix indexed by λ and ζ. The size of

Table 3.9 GLRLM properties

Operator linearity	Nonlinear.
Handcrafted	Yes.
3D extension	Straightforward: run directions can be extended to 3D and indexed with angles θ, ϕ.
Coverage of image directions	Incomplete: typical directions used are $\theta_{1,\ldots,4} = 0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}$ in 2D.
Directionality and local rotation-invariance	Unidirectional and not rotation-invariant. However, local rotation-invariance is often approximated by either regrouping the counts of operators over all directions, or by averaging scalar texture measurements from run-length matrices obtained with different directions.
Characterization of the LOIDs	No.
Coverage of image scales	Complete if the maximum run-length γ is equal to the size of the image. However, run length along image diagonals (*e.g.*, $\theta_2 = \frac{\pi}{4}$ and $\theta_4 = \frac{3\pi}{4}$) are often considered to have integer γ values (*e.g.*, $1, 2, 3$) instead of their actual values (*e.g.*, $\sqrt{2}, 2\sqrt{2}, 3\sqrt{2}$), resulting in anisotropic descriptions of image scales.
Band-pass	Qualitatively similar in the sense that the mean value of the image is not influencing the output value of the operator. It is worth noting that the transfer function is not defined in the Fourier domain because the operator is nonlinear.
Gray-level reduction	A reduction is required to avoid obtaining very large and sparse run-length matrices. Typical gray-level reductions are $\Lambda = 8, 16, 32$.
Illumination-invariance	No, although it is approximated by reducing the number of gray-levels Λ.
Aggregation function	Integrative: counts the binary responses of each GLRLM operator $\mathcal{G}_{\lambda,\gamma,\theta}$ over $\boldsymbol{M}$ and organizes them in a corresponding run-length matrix indexed by integer values of λ and γ. Scalar texture measurements $\boldsymbol{\eta}$ are obtained from statistics of the run-length matrices.

this matrix is the number Λ of gray-levels considered, times the number of zone areas tested. The texture measures $\boldsymbol{\eta}$ are obtained from statistics of the matrices, which are

the same as for the GLRLM (see [58]) plus two additional measures introduced in [59]. Their properties are summarized in Table 3.10.

Table 3.10 GLSZM properties

Operator linearity	Nonlinear.
Handcrafted	Yes.
3D extension	Straightforward: the search for contiguous pixels with identical gray-level values can be extended from 8-connected 2D neighborhoods to their 26-connected 3D counterparts.
Coverage of image directions	Complete.
Directionality and local rotation-invariance	Insensitive to image directions. Elongated and circular zones are mixed.
Characterization of the LOIDs	No.
Coverage of image scales	Complete if the maximum zone surface ζ is equal to the area of the image. However, the notion of scale is ill-defined because a fixed zone area ζ can correspond to both elongated or circular regions.
Band-pass	Qualitatively equivalent in the sense that the mean value of the image is not influencing the output value of the operator. It is worth noting that the transfer function is not defined in the Fourier domain because the operator is nonlinear.
Gray-level reduction	A reduction is required to avoid obtaining very large and sparse size-zone matrices. Typical gray-level reductions are $\Lambda = 8, 16, 32$.
Illumination-invariance	No, although it is approximated by reducing the number of gray-levels Λ.
Aggregation function	Integrative: counts the binary responses of each GLSZM operator $\mathcal{G}_{\lambda,\zeta}$ over $\boldsymbol{M}$ and organizes them in a corresponding size-zone matrix indexed by integer values of λ and ζ. Scalar texture measurements $\boldsymbol{\eta}$ are obtained from statistics of the size-zone matrices.

3.4 LOCAL BINARY PATTERNS (LBP)

Rotation-invariant Local Binary Patterns (LBP) were first introduced by Ojala et al. in 2002 [60] and many extensions were proposed later on (*e.g.*, [30,61,62]). At a po-

sition $\boldsymbol{\xi}_0$ of the image f, the LBP operator $\mathcal{G}_{\gamma,r}\{f\}(\boldsymbol{\xi}_0)$ describes the organization of binarized pixels over circular neighborhoods $\Upsilon(\gamma, r, \boldsymbol{\xi}_0)$ of radius r containing γ equally spaced points. The gray value of points that do not fall exactly in the center of pixels are estimated by interpolation (see Fig. 3.12, top). The decimal value of the operator is given by the binary sequence of the circular neighborhood (*e.g.*, $10101010 = 170$, see Fig. 3.12, middle). The binary value of a point $\boldsymbol{p} \in \Upsilon(\gamma, r, \boldsymbol{\xi}_0)$ is 1 if $f(\boldsymbol{\xi}_1) > f(\boldsymbol{\xi}_0)$, and 0 otherwise, with $\boldsymbol{\xi}_1$ the pixel location where $\boldsymbol{p}$ is located. An example of approximately locally rotation-invariant LBP texture analysis is depicted in Fig. 3.12 middle and bottom.

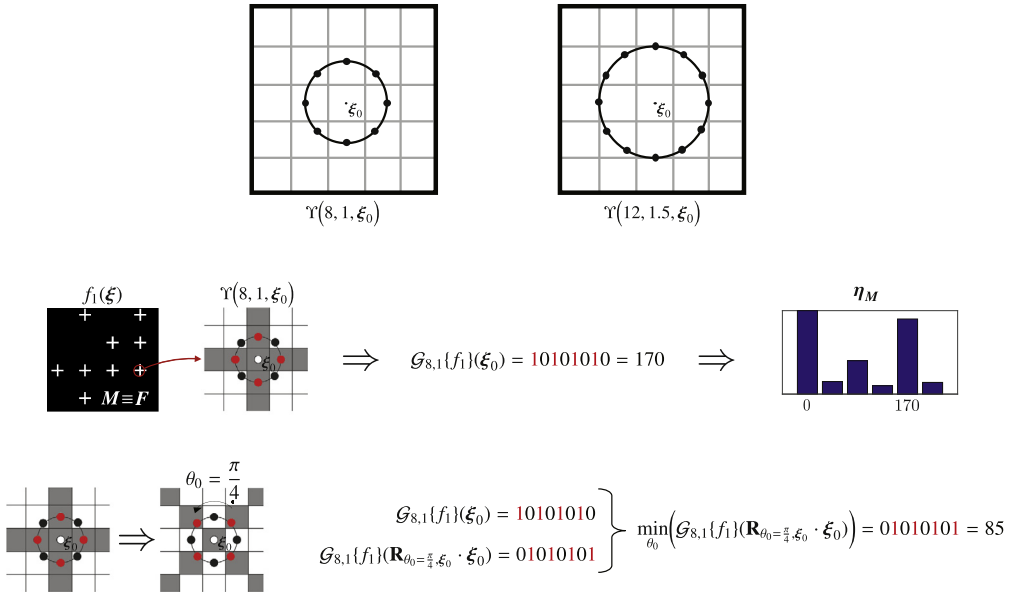

Figure 3.12 Extraction of LBP texture measures on $f_1(\xi)$. Top row: example of two different circular neighborhoods $\Upsilon(\gamma, r, \xi_0)$. Middle row: a LBP operator $\mathcal{G}_{8,1}\{f_1\}(\xi_0)$ encodes the LOIDs at the position ξ_0 over a circular neighborhood $\Upsilon(8, 1, \xi_0)$ with $\gamma = 8$ equally-spaced points and a radius $r = 1$ pixel. Its binary response for a +-shaped primitive is 10101010, which correspond to a decimal value of 170. The response maps of the operators are aggregated over the region $\boldsymbol{M}$ by counting the binary codes and organizing them into a histogram of texture measurements $\boldsymbol{\eta}_{\boldsymbol{M}}$. Bottom row: local rotation-invariance with LBPs. Local image rotations $\mathbf{R}_{\theta_0,\xi_0} \cdot \xi$ correspond to bit-wise circular shifts of the binary codes. Local rotation-invariance can be achieved by minimizing the decimal values of the binary codes over all possible discrete circular shifts $\theta_0 = \frac{q\pi}{4}, \forall q \in \{0, 1, \ldots, 7\}$.

The responses of the operators are aggregated over a ROI $\boldsymbol{M}$ by counting the binary sequences (or decimal values) and organizing them in a histogram. The latter can be used for extracting texture measures $\boldsymbol{\eta}$. The properties of LBPs are summarized in Table 3.11.

Table 3.11 LBP properties

Operator linearity	Nonlinear.
Handcrafted	Yes.
3D extension	Not trivial: The ordering of points is straightforward in 2D on circular neighborhoods, but is undefined in 3D for spherical neighborhoods [63]. Approaches were proposed to either define an arbitrary ordering for each (γ, r) over (θ, ϕ) and to use it for all positions [64], or to use cylindrical neighborhoods by concatenating the responses of 2D LBP operators along a given axis ξ_d [65].
Coverage of image directions	Complete if γ considers all pixels/voxels touching the 2D/3D perimeter of Υ for a fixed radius r.
Directionality and local rotation-invariance	Directional and locally rotation-equivariant. Local rotation-invariance can be obtained either by performing circular bit-wise right shifts of the binary codes and keeping the minimum value [60] (see Fig. 3.12), or by computing the modulus of the discrete 1D Fourier transforms of the binary code [30]. In the former case, local rotation-invariance is achieved by locally aligning the operators (see Section 2.4.3 of Chapter 2).
Characterization of the LOIDs	Yes, with invariance/equivariance to local rotations. However, they do not define MF representations since each operator $\mathcal{G}_{\gamma,r}$ is aligned independently at the position $\boldsymbol{\xi}_0$.
Coverage of image scales	Incomplete: typical radius values are $r = 1, 2, 3$. Multiscale LBPs were proposed in [66] when extracted on top of wavelet coefficients.
Band-pass	Qualitatively similar in the sense that the mean value of the image is not influencing the output value of the operator. It is worth noting that the transfer function is not defined in the Fourier domain because the operator is nonlinear.
Gray-level reduction	Not required. However, the local binarization operation results in an important reduction of the values analyzed.
Illumination-invariance	Yes.
Aggregation function	Integrative: counts the responses of each LBP operator $\mathcal{G}_{\gamma,r}$ over $\boldsymbol{M}$ and organizes them in a histogram. The collection of scalar texture measures $\boldsymbol{\eta}$ contains the bins of the histogram.

3.5 FRACTALS

Another popular method in BTA is to estimate the fractal properties of biomedical tissue, *e.g.*, how structures are similar across a series of monotonously increasing scales [67,68]. The latter is measured with the *Fractal Dimension* (FD) η_{FD}. The larger the FD, the most regular structures are through multiple scales, which corresponds closely to our intuitive notion of roughness [69]. The maximum FD is equal to the dimensionality D of the texture function. For 2D texture functions indexed by the spatial coordinates $(x_1, x_2) \in \mathbb{R}^2$, we typically consider the set $S = \{(x_1, x_2) \in \mathbb{R}^2 : f(x_1, x_2) = \gamma\} \subset \mathbb{R}^2$ for a fixed value $\gamma \in \mathbb{R}$. Note that S is a level set of f. Then the fractal dimension can take values $0 \leq \eta_{\text{FD}} \leq 2$. A comprehensive description and review of fractal analysis for BTA is presented in Chapter 5.

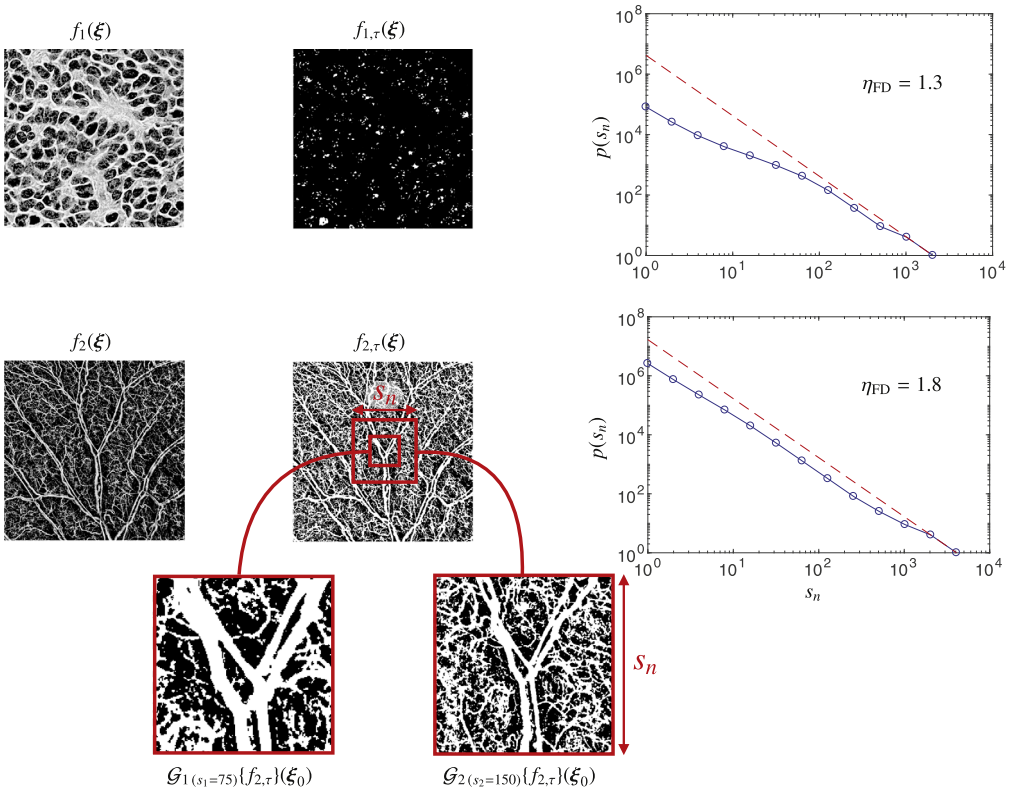

Figure 3.13 2D box-counting method used to estimate the fractal dimension η_{FD} of a texture. Although not recommended in general, a binarization of the texture images is used in this example. The latter is carried out with a thresholding operation, where the threshold τ was chosen to highlight cell nucleus in f_1 and the vascular structure in f_2. The visualization of the profiles $p(s_n)$ reveals the fractal nature of the vascular structure with a FD of 1.8. The maximum FD bound of 2 corresponding to a fully white image corresponds to the dashed red line in the profiles $p(s_n)$.

Table 3.12 Properties of fractal-based texture analysis using the box-counting method

Operator linearity	In principle yes, but it depends on the type of transformation of pixel values. In the example, the binarization of f is a nonlinear operation.
Handcrafted	Yes.
3D extension	Straightforward by using multiscale 3D boxes as operators' supports.
Coverage of image directions	Complete.
Directionality and local rotation-invariance	Insensitive to image directions.
Characterization of the LOIDs	No.
Coverage of image scales	Complete. However, the FD is invariant to image scale, which entails the risk of regrouping tissue structures of different natures.
Band-pass	Not qualitatively equivalent in the sense that the mean value of the image has an influence on the output value of the operator. It is worth noting that the transfer function is not defined in the Fourier domain because the operator is nonlinear.
Gray-level reduction	No. A binarization step or gray-level reduction is usually not recommended as this would both arbitrarily discard important texture information and degrade the stability of the calculation of η_{FD} (see [68,70]).
Illumination-invariance	No.
Aggregation function	Constructs the log-log profile of the averages of operator response maps h_n obtained with operators of spatial supports with varying sizes s_n (see Fig. 3.13).

A popular method for computing the FD is the *box-counting* approach, while several other methods exist (*e.g.*, box counting and probabilities [71], based on Fourier analysis [69] or wavelets [72], see Chapter 5 for detailed descriptions and more approaches). Box-counting relies on a collection of multiscale operators $\mathcal{G}_n$. For simplification, let us assume that the latter have square or cubic spatial support $G_{1,n} = G_{2,n} = \cdots = G_{D,n} = s_n$. The collection of operators have monotonously increasing spatial supports $\mathbf{G}_n \subset \mathbf{G}_{n+1}$. In order to simply exemplify the box-counting algorithm, let us consider binary images $f_\tau(\boldsymbol{\xi}) \in \{0, 1\}$ obtained from the binarization of an input image $f(\boldsymbol{\xi})$ with a thresh-

old τ[24]. At a position $\boldsymbol{\xi}_0$, each fractal operator $\mathcal{G}_n$ counts the number p of pixels with value equals to 1 (*e.g.*, $f_\tau(\boldsymbol{\xi}_0) = 1$) over its support $\boldsymbol{G}_n$. This yields N response maps $h_n(\boldsymbol{\xi}_0) = \mathcal{G}_n\{f_\tau\}(\boldsymbol{\xi}_0)$.

The aggregation function consists of fitting a log-log profile of the averages of h_n through the responses of the collection of operators with spatial supports of varying sizes s_n, where $|\boldsymbol{G}_n| = {s_n}^D$. The average number p of pixels with value equals to 1 over a ROI $\boldsymbol{M}$ is computed as

$$p(s_n) = \frac{1}{|\boldsymbol{M}|} \sum_{\boldsymbol{\xi} \in \boldsymbol{M}} h_n(\boldsymbol{\xi}).$$

The FD is obtained with the value of η_{FD} that best fits the following relation:

$$\log\left(p(s_n)\right) \simeq \eta_{\mathrm{FD}} \log\left(s_n\right). \tag{3.24}$$

In this particular case, $\eta_{\mathrm{FD}} = 2$ corresponds to a perfectly regular image filled with ones. The box counting method is illustrated in Fig. 3.13, where two biomedical textures with different FD are compared. The properties of fractal-based texture analysis are summarized in Table 3.12 and are further developed in Chapter 5. The FD is invariant to image scale, which entails the risk of regrouping tissue structures of different natures (see Section 1.3.3 of Chapter 1).

3.6 DISCUSSIONS AND CONCLUSIONS

A qualitative comparison of most popular BTA approaches was proposed under the light of the general framework introduced in Section 1.3.1 of Chapter 1, and based on the comparison dimensions presented in Chapter 2. Our aim is to both provide a user guide for choosing a BTA method that is relevant to the problem in hand, as well as to provide insights on key aspects required to build the next generation of BTA approaches. The review focused on most popular group of methods and was not exhaustive. It included (i) convolutional methods and their operator subtypes: circularly/spherically symmetric, directional and learned, (ii) gray-level matrices and their subtypes: GLCMs, GLRLMs, GLSZMs, (iii) LBPs, and (iv) fractals based on the box-counting method. All approaches' operators were found to be equivariant to translations, which is an inherited property from the general framework introduced in Section 1.3.1 of Chapter 1. However, very few methods were combining both the ability to characterize the LOIDs and with invariance to local rotations. In particular, most approaches are either invariant to local rotations because they are insensitive to directions (*e.g.*, combined GLCM or

[24] The binarization step or gray-level reduction is usually not recommended for box-counting as this would both arbitrarily discard important texture information and degrade the stability of the calculation of η_{FD} (see [68,70] and Chapter 5).

GLRLM operators, LoGs, fractals) and they cannot characterize the LOIDs, or they are directional but not locally rotation-invariant (*e.g.*, unidirectional GLCM or GLRLM operators, Gabor wavelets, unaligned real Riesz, DL, CNNs). Within this dilemma, one may favor directional versus directionally insensitive approaches depending on the expected importance of the LOIDs versus invariance to local rotations, respectively. The only approaches that are able to characterize the LOIDs with invariance to (local) rigid transformations are LBPs as well as CH, steered real Riesz wavelets and SWMs. Other factors must be taken into account to evaluate the relevance of the method. An important property that is responsible for the success of approaches such as DL and CNNs is the ability to derive the texture operators[25] from the data in a supervised or unsupervised fashion. A challenge for the success of the learning-based methods is to have sufficient representation of each intra-class variants, which are most often not available in focused and innovative biomedical applications (see Chapter 6). Quick fixes have been extensively used to tackle this challenge. A first one is to introduce implicit handcrafted invariances with data augmentation, which has the undesirable effects of both increasing the training computational load (minor issue), and more importantly to decrease the specificity of the model [55] (see Section 3.2.3.4). A second quick fix is to use transfer learning and fine tuning in order to recycle models trained on other image types. This is not exempt of risks as borrowing models from general computer vision with strong invariance to scale (*e.g.*, models trained on ImageNet) are not fulfilling the requirement of BTA (see Section 3.2.3). An inherent risk of methods requiring important gray-level reductions or binarization (*e.g.*, GLM, LBP, fractals based on box-counting) is to miss or mix important texture properties. It was found that methods based on learned steerable wavelets (*i.e.*, SWMs [26,31]) were regrouping several desirable properties such as the ability to learn optimally discriminant sets of LOIDs with invariance to rigid transformation, with a small amount of training data, and without requiring neither data augmentation nor gray-level reductions. Few methods were found to provide easily interpretable texture measurements. Notable examples among them are the contrast and energy measures of GLCMs, the fractal dimension and roughness [69] (see Chapter 5), as well as 2D and 3D steered real Riesz wavelets that are measuring image derivatives of various orders in a locally rotation-invariant fashion [18,20,17,16] (see Section 12.2.4.4 of Chapter 12).

We recognize several limitations of the current review, including the absence of quantitative comparison of performance, computational complexity and time of the approaches. Detailed reviews of specific BTA properties, approaches or applications are presented in Chapter 7 (invariance to rigid transformations), Chapters 4 and 9 (deep learning), Chapter 6 (machine learning), and Chapter 10 (digital histopathology).

[25] CNNs with FC layers are also learning the aggregation function.

ACKNOWLEDGMENTS

This work was supported by the Swiss National Science Foundation (under grant PZ00P2_154891) and the CIBM.

REFERENCES

[1] B. Julesz, Textons, the elements of texture perception, and their interactions, Nature 290 (5802) (mar 1981) 91–97.

[2] D. Marr, E. Hildreth, Theory of edge detection, Proc. R. Soc. Lond. B, Biol. Sci. 207 (1167) (feb 1980) 187–217.

[3] D. Van De Ville, M. Unser, Complex wavelet bases, steerability, and the Marr-like pyramid, IEEE Trans. Image Process. 17 (11) (November 2008) 2063–2080.

[4] B. Ganeshan, E. Panayiotou, K. Burnand, S. Dizdarevic, K.A. Miles, Tumour heterogeneity in non-small cell lung carcinoma assessed by CT texture analysis: a potential marker of survival, Eur. Radiol. 22 (4) (2012) 796–802.

[5] T. Sing Lee, Image representation using 2D Gabor wavelets, IEEE Trans. Pattern Anal. Mach. Intell. 18 (10) (oct 1996) 959–971.

[6] A.G. Ramakrishnan, S.K. Raja, H.V.R. Ram, Neural network-based segmentation of textures using Gabor features, in: Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing, 2002, pp. 365–374.

[7] M. Haghighat, S. Zonouz, M. Abdel-Mottaleb, CloudID: trustworthy cloud-based and cross-enterprise biometric identification, Expert Syst. Appl. 42 (21) (2015) 7905–7916.

[8] M. Petrou, P. García Sevilla, Non-Stationary Grey Texture Images, John Wiley & Sons, Ltd, 2006, pp. 297–606.

[9] J.G. Daugman, Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters, J. Opt. Soc. Am. A, Opt. Image Sci. Vis. 2 (7) (1985) 1160–1169.

[10] W. Pan, T.D. Bui, C.Y. Suen, Rotation-invariant texture classification using steerable Gabor filter bank, in: M. Kamel, A. Campilho (Eds.), Image Analysis and Recognition: Second International Conference, Proceedings, ICIAR 2005, Toronto, Canada, September 28–30, 2005, Springer, Berlin, Heidelberg, ISBN 978-3-540-31938-2, 2005, pp. 746–753.

[11] M. Varma, A. Zisserman, A statistical approach to texture classification from single images, Int. J. Comput. Vis. 62 (1–2) (2005) 61–81.

[12] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.

[13] W.T. Freeman, E.H. Adelson, The design and use of steerable filters, IEEE Trans. Pattern Anal. Mach. Intell. 13 (9) (sep 1991) 891–906.

[14] P. Scovanner, S. Ali, M. Shah, A 3-dimensional SIFT descriptor and its application to action recognition, in: Proceedings of the 15th ACM International Conference on Multimedia, MM'07, ACM, New York, NY, USA, 2007, pp. 357–360.

[15] M. Unser, D. Van De Ville, Wavelet steerability and the higher-order Riesz transform, IEEE Trans. Image Process. 19 (3) (March 2010) 636–652.

[16] N. Chenouard, M. Unser, 3D steerable wavelets in practice, IEEE Trans. Image Process. 21 (11) (2012) 4522–4533.

[17] Y. Dicente Cid, H. Müller, A. Platon, P.-A. Poletti, A. Depeursinge, 3-D solid texture classification using locally-oriented wavelet transforms, IEEE Trans. Image Process. 26 (4) (2017) 1899–1910, http://dx.doi.org/10.1109/TIP.2017.2665041.

[18] A. Depeursinge, P. Pad, A.C. Chin, A.N. Leung, D.L. Rubin, H. Müller, M. Unser, Optimized steerable wavelets for texture analysis of lung tissue in 3-D CT: classification of usual interstitial pneumonia, in: IEEE 12th International Symposium on Biomedical Imaging, ISBI 2015, IEEE, apr 2015, pp. 403–406.
[19] P. Cirujeda, Y. Dicente Cid, H. Müller, D.L. Rubin, T.A. Aguilera, B.W. Loo Jr., M. Diehn, X. Binefa, A. Depeursinge, A 3-D Riesz-covariance texture model for prediction of nodule recurrence in lung CT, IEEE Trans. Med. Imaging 35 (12) (2016) 2620–2630.
[20] A. Depeursinge, A. Foncubierta-Rodríguez, D. Van De Ville, H. Müller, Lung texture classification using locally-oriented Riesz components, in: G. Fichtinger, A. Martel, T. Peters (Eds.), Medical Image Computing and Computer Assisted Intervention, MICCAI 2011, in: Lecture Notes in Computer Science, vol. 6893, Springer, Berlin/Heidelberg, sep 2011, pp. 231–238.
[21] J.P. Ward, M. Unser, Harmonic singular integrals and steerable wavelets in $L_2(\mathbb{R}^d)$, Appl. Comput. Harmon. Anal. 36 (2) (mar 2014) 183–197.
[22] H. Skibbe, M. Reisert, T. Schmidt, T. Brox, O. Ronneberger, H. Burkhardt, Fast rotation invariant 3D feature computation utilizing efficient local neighborhood operators, IEEE Trans. Pattern Anal. Mach. Intell. 34 (8) (aug 2012) 1563–1575.
[23] J. Portilla, E.P. Simoncelli, A parametric texture model based on joint statistics of complex wavelet coefficients, Int. J. Comput. Vis. 40 (1) (2000) 49–70.
[24] I. Daubechies, Ten Lectures on Wavelets, vol. 61, SIAM, 1992.
[25] M. Unser, N. Chenouard, D. Van De Ville, Steerable pyramids and tight wavelet frames in $L_2(\mathbb{R}^d)$, IEEE Trans. Image Process. 20 (10) (oct 2011) 2705–2721.
[26] A. Depeursinge, Z. Püspöki, J.-P. Ward, M. Unser, Steerable wavelet machines (SWM): learning moving frames for texture classification, IEEE Trans. Image Process. 26 (4) (2017) 1626–1636.
[27] M. Unser, N. Chenouard, A unifying parametric framework for 2D steerable wavelet transforms, SIAM J. Imaging Sci. 6 (1) (2013) 102–135.
[28] G. Jacovitti, A. Neri, Multiresolution circular harmonic decomposition, IEEE Trans. Signal Process. 48 (11) (2000) 3242–3247.
[29] K. Liu, H. Skibbe, T. Schmidt, T. Blein, K. Palme, T. Brox, O. Ronneberger, Rotation-invariant HOG descriptors using Fourier analysis in polar and spherical coordinates, Int. J. Comput. Vis. 106 (3) (2014) 342–364.
[30] T. Ahonen, J. Matas, C. He, M. Pietikäinen, Rotation invariant image description with local binary pattern histogram Fourier features, in: A.-B. Salberg, J. Hardeberg, R. Jenssen (Eds.), Image Analysis, in: Lecture Notes in Computer Science, vol. 5575, Springer, Berlin, Heidelberg, 2009, pp. 61–70.
[31] A. Depeursinge, A. Foncubierta-Rodríguez, D. Van De Ville, H. Müller, Rotation-covariant texture learning using steerable Riesz wavelets, IEEE Trans. Image Process. 23 (2) (February 2014) 898–908.
[32] A. Depeursinge, M. Yanagawa, A.N. Leung, D.L. Rubin, Predicting adenocarcinoma recurrence using computational texture models of nodule components in lung CT, Med. Phys. 42 (2015) 2054–2063.
[33] A. Depeursinge, C. Kurtz, C.F. Beaulieu, S. Napel, D.L. Rubin, Predicting visual semantic descriptive terms from radiological image data: preliminary results with liver lesions in CT, IEEE Trans. Med. Imaging 33 (8) (August 2014) 1–8.
[34] I. Tosic, P. Frossard, Dictionary learning, IEEE Signal Process. Mag. 28 (2) (mar 2011) 27–38.
[35] R. Rigamonti, V. Lepetit, Accurate and efficient linear structure segmentation by leveraging ad hoc features with learned filters, in: N. Ayache, H. Delingette, P. Golland, K. Mori (Eds.), Medical Image Computing and Computer-Assisted Intervention, MICCAI 2012, in: Lecture Notes in Computer Science, vol. 7510, Springer, Berlin, Heidelberg, 2012, pp. 189–197.
[36] Y.-J. Lee, O.L. Mangasarian, SSVM: a smooth support vector machine for classification, Comput. Optim. Appl. 20 (1) (2001) 5–22.
[37] S.-C. Zhu, C.-E. Guo, Y. Wang, Z. Xu, What are textons?, in: Special Issue on Texture Analysis and Synthesis, Int. J. Comput. Vis. 62 (1–2) (apr 2005) 121–143.

[38] J. Mairal, F. Bach, J. Ponce, G. Sapiro, A. Zisserman, Supervised dictionary learning, Adv. Neural Inf. Process. Syst. (2008) 1033–1040.

[39] M.J. Gangeh, A. Ghodsi, M.S. Kamel, Dictionary learning in texture classification, in: Proceedings of the 8th International Conference on Image Analysis and Recognition, Part I, 2011, pp. 335–343.

[40] A. Hyvärinen, J. Hurri, P.O. Hoyer, Energy correlations and topographic organization, in: Natural Image Statistics: A Probabilistic Approach to Early Computational Vision, Springer London, London, 2009, pp. 239–261.

[41] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (mar 2003) 1157–1182.

[42] J. Bruna, S.G. Mallat, Invariant scattering convolution networks, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1872–1886.

[43] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016, http://www.deeplearningbook.org.

[44] A. Vedaldi, K. Lenc, MatConvNet: convolutional neural networks for MATLAB, in: Proceedings of the 23rd ACM International Conference on Multimedia, MM '15, ACM, New York, NY, USA, 2015, pp. 689–692.

[45] A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classification with deep convolutional neural networks, in: F. Pereira, C.J.C. Burges, L. Bottou, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems, vol. 25, Curran Associates, Inc., 2012, pp. 1097–1105.

[46] O. Ronneberger, P. Fischer, T. Brox, U-Net: convolutional networks for biomedical image segmentation, in: N. Navab, J. Hornegger, W.M. Wells, A.F. Frangi (Eds.), Medical Image Computing and Computer-Assisted Intervention, MICCAI 2015, in: Lecture Notes in Computer Science, vol. 9351, Springer International Publishing, 2015, pp. 234–241.

[47] B. Sahiner, H.-P. Chan, N. Petrick, D. Wei, M.A. Helvie, D.D. Adler, M.M. Goodsitt, Classification of mass and normal breast tissue: a convolution neural network classifier with spatial domain and texture images, IEEE Trans. Med. Imaging 15 (5) (oct 1996) 598–610.

[48] A.S. Razavian, H. Azizpour, J. Sullivan, S. Carlsson, CNN features off-the-shelf: an astounding baseline for recognition, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 512–519.

[49] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, jun 2014, pp. 1717–1724.

[50] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: a large-scale hierarchical image database, in: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, 2009, pp. 248–255.

[51] Q. Dou, H. Chen, Y. Jin, L. Yu, J. Qin, P.-A. Heng, 3D Deeply Supervised Network for Automatic Liver Segmentation from CT Volumes, Springer International Publishing, Cham, 2016, pp. 149–157.

[52] F. Milletari, N. Navab, S.-A. Ahmadi, V-Net: fully convolutional neural networks for volumetric medical image segmentation, CoRR, abs/1606.04797, 2016.

[53] Ö. Çiçek, A. Abdulkadir, S.S. Lienkamp, T. Brox, O. Ronneberger, 3D U-Net: learning dense volumetric segmentation from sparse annotation, in: Medical Image Computing and Computer Assisted Intervention, MICCAI 2016, in: Lecture Notes in Computer Science, vol. 9901, Springer International Publishing, 2016, pp. 424–432.

[54] J. Wang, J.D. MacKenzie, R. Ramachandran, D.Z. Chen, A deep learning approach for semantic segmentation in histology tissue images, in: S. Ourselin, L. Joskowicz, M.R. Sabuncu, G. Unal, W. Wells (Eds.), Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016: 19th International Conference, Proceedings, Part II, Athens, Greece, October 17–21, 2016, Springer International Publishing, Cham, ISBN 978-3-319-46723-8, 2016, pp. 176–184.

[55] D. Marcos Gonzalez, M. Volpi, D. Tuia, Learning rotation invariant convolutional filters for texture classification, CoRR, abs/1604.06720, 2016.
[56] R.M. Haralick, Statistical and structural approaches to texture, Proc. IEEE 67 (5) (may 1979) 786–804.
[57] M.M. Galloway, Texture analysis using gray level run lengths, Comput. Graph. Image Process. 4 (2) (1975) 172–179.
[58] D.-H. Xu, A.S. Kurani, J. Furst, D.S. Raicu, Run-length encoding for volumetric texture, in: The 4th IASTED International Conference on Visualization, Imaging, and Image Processing, VIIP 2004, Marbella, Spain, sep 2004.
[59] G. Thibault, B. Fertil, C. Navarro, S. Pereira, P. Cau, N. Levy, J. Sequeira, J.-L. Mari, Texture indexes and gray level size zone matrix application to cell nuclei classification, in: Pattern Recognition and Information Processing, 2009, pp. 140–145.
[60] T. Ojala, M. Pietikäinen, T. Mäenpää, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Anal. Mach. Intell. 24 (7) (jul 2002) 971–987.
[61] Z. Guo, L. Zhang, D. Zhang, A completed modeling of local binary pattern operator for texture classification, IEEE Trans. Image Process. 19 (6) (jun 2010) 1657–1663.
[62] L. Liu, S. Lao, P.W. Fieguth, Y. Guo, X. Wang, M. Pietikäinen, Median robust extended local binary pattern for texture classification, IEEE Trans. Image Process. 25 (3) (mar 2016) 1368–1381.
[63] A. Depeursinge, A. Foncubierta-Rodríguez, D. Van De Ville, H. Müller, Three-dimensional solid texture analysis and retrieval in biomedical imaging: review and opportunities, Med. Image Anal. 18 (1) (2014) 176–196.
[64] J. Fehr, H. Burkhardt, 3D rotation invariant local binary patterns, in: 19th International Conference on Pattern Recognition, 2008, ICPR 2008, December 2008, pp. 1–4.
[65] A. Burner, R. Donner, M. Mayerhoefer, M. Holzer, F. Kainberger, G. Langs, Texture bags: anomaly retrieval in medical images based on local 3D-texture similarity, in: H. Greenspan, H. Müller, T. Syeda-Mahmood (Eds.), Medical Content-Based Retrieval for Clinical Decision Support, MCBR-CDS 2011, in: Lecture Notes in Computer Sciences (LNCS), vol. 7075, September 2012, pp. 116–127.
[66] Y.D. Wang, Q.Y. Yan, K.F. Li, Hand vein recognition based on multi-scale LBP and wavelet, in: 2011 International Conference on Wavelet Analysis and Pattern Recognition, jul 2011, pp. 214–218.
[67] B.B. Mandelbrot, The Fractal Geometry of Nature, W.H. Freeman and Company, 1977.
[68] O.S. Al-Kadi, D. Watson, Texture analysis of aggressive and nonaggressive lung tumor CE CT images, IEEE Trans. Biomed. Eng. 55 (7) (July 2008) 1822–1830.
[69] A.P. Pentland, Fractal-based description of natural scenes, IEEE Trans. Pattern Anal. Mach. Intell., PAMI-6 (6) (nov 1984) 661–674.
[70] N. Sarkar, B.B. Chaudhuri, An efficient differential box-counting approach to compute fractal dimension of image, IEEE Trans. Syst. Man Cybern. 24 (1) (jan 1994) 115–120.
[71] J.M. Keller, S. Chen, R.M. Crownover, Texture description and segmentation through fractal geometry, Comput. Vis. Graph. Image Process. 45 (2) (1989) 150–166.
[72] B. Pesquet-Popescu, J.L. Vehel, Stochastic fractal models for image processing, IEEE Signal Process. Mag. 19 (5) (sep 2002) 48–62.