3D Solid Texture Classification Using Locally-Oriented Wavelet Transforms

Yashin Dicente Cid, Henning Müller, Alexandra Platon, Pierre-Alexandre Poletti, and Adrien Depeursinge

***Abstract*— Many image acquisition techniques used in biomedical imaging, material analysis, and structural geology are capable of acquiring 3D solid images. Computational analysis of these images is complex but necessary, since it is difficult for humans to visualize and quantify their detailed 3D content. One of the most common methods to analyze 3D data is to characterize the volumetric texture patterns. Texture analysis generally consists of encoding the local organization of image scales and directions, which can be extremely diverse in 3D. Current state-of-the-art techniques face many challenges when working with 3D solid texture, where most approaches are not able to consistently characterize both scale and directional information. 3D Riesz–wavelets can deal with both properties. One key property of Riesz filterbanks is steerability, which can be used to locally align the filters and compare textures with arbitrary (local) orientations. This paper proposes and compares three novel local alignment criteria for higher-order 3D Riesz–wavelet transforms. The estimations of local texture orientations are based on higher-order extensions of regularized structure tensors. An experimental evaluation of the proposed methods for the classification of synthetic 3D solid textures with alterations (such as rotations and noise) demonstrated the importance of local directional information for robust and accurate solid texture recognition. These alignment methods achieved an accuracy of 0.95 in the rotated data, three times more than the unaligned Riesz descriptor that achieved 0.32. The accuracy obtained is better than all other techniques that are published and tested on the same database.**

***Index Terms*— 3D solid texture analysis, 3D texture classification, Riesz–wavelets steerability, aligned higher–order Riesz–wavelet transform, rotation–covariance.**

I. Introduction

IMAGING techniques in areas such as biomedical imaging [1], material analysis [2], and structural geology [3], allow the acquisition of 3D solid texture (3DST) images. These images consist of textured dense 3D volumes, containing 2D textures in all planes of $\mathbb{R}^3$. Most 3D data are analyzed in 2D slices, as this is the easiest for volumetric navigation. Computational approaches are required because humans are not able to fully visualize, easily interpret and quantify 3DST properties. The automatic analysis of 3DST remains a challenging topic. Spatial scales and directions in images are fundamental for texture discrimination [4], [5]. In most cases, systems capable of acquiring 3D images also provide voxel sizes in physical dimensions and controlled viewpoints (such as X–ray computed tomography, CT or magnetic resonance imaging, MRI). In this case, the analysis of image scales requires describing 3DST at multiple scales and not using scale–invariant descriptors, the latter entailing the risk of regrouping patterns of different nature. Scale is itself a powerful discriminative property.

Much work has been done on 2D texture analysis since the 1970s and many approaches exist, as it is less complex than in 3D [1]. Popular approaches are gray–level matrices (co–occurrence [6], run–length [7], and size–zone [8]), local binary patterns (LBP) [9], wavelets and filterbanks (Laplacian of Gaussian [10], Gabor [11], maximum–response eight [12], Riesz [13], [14]), fractals [15], and learned representations (dictionary learning [16], deep convolutional neural networks [17]). In natural images, 2D analysis also often includes visual descriptors such as color, shape, and appearance [18] that complement texture. 3D extensions of color descriptors are relatively straightforward. Many 3D shape descriptors exist and are usually based on 3D surface models but can also be based on solid 3D information [19]. For image classification or object recognition these features are aggregated in the form of visual words or Fisher vectors. This article aims only at describing texture properties in 3DST, and it does not cover color or shape features that can be combined with texture.

First attempts for describing 3DST were based on extending 2D texture descriptors to 3D. LBPs [9] were extended to 3D in [20]. 2D LBPs encode the local organizations of image directions by constructing binary codes over circular neighborhoods. Histograms of pattern occurrences can be used as a texture descriptor. Image rotations result in circular shifts of the LBPs and rotation–invariance can be obtained by using the magnitude of the discrete Fourier transform of the LBPs [21]. When extended to 3D, defining ordered spherical neighborhoods is not trivial, making it difficult to obtain rotation–invariance. Solutions were proposed using spherical harmonics [22] and cylindrical neighborhoods [23]. However, the radius of the neighborhood (scale) required by LBPs is controlled by a parameter that needs to be optimized for every application. Another set of descriptors

Manuscript received December 22, 2015; revised June 15, 2016 and December 16, 2016; accepted January 20, 2017. Date of publication February 5, 2017; date of current version March 9, 2017. This work was supported in part by the Swiss National Science Foundation under Grant 320030-146804 and Grant PZ00P2 154891 and in part by CIBM. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Peter Tay.

Y. Dicente Cid and H. Müller are with the MedGIFT Group, Institute of Information Systems, University of Applied Sciences Western Switzerland (HES-SO), 3960 Sierre, Switzerland, and also with the University Hospital and University of Geneva, 1211 Geneva, Switzerland (e-mail: yashin.dicente@hevs.ch).

A. Platon and P.-A. Poletti are with the Department of Radiology and Medical Informatics, Geneva University Hospital, 1211 Geneva, Switzerland.

A. Depeursinge is with the MedGIFT Group, Institute of Information Systems, University of Applied Sciences Western Switzerland (HES-SO), 3960 Sierre, Switzerland, and also with the Biomedical Imaging Group, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2017.2665041

1057-7149 © 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

used to describe 3DST is based on gray level co–occurrence matrices (GLCMs) [24], [25]. A GLCM contains the counts of all co–occurrences between two voxel values separated by a distance r and in a specific direction. In 3D, the directions are defined for every surface point of a spherical neighborhood centered at each voxel. The radius r of the neighborhood defines the scale. Texture descriptors can be obtained from each matrix by calculating, for example, the homogeneity, entropy or energy of a matrix in a specific direction and scale. Rotation–invariant features can be obtained by summing the descriptors obtained from GLCMs in all directions, so a subset of 13 uniformly distributed directions. This procedure discards the directionality of the descriptor. Similarly to 3D LBPs, the scales need to be defined manually and require optimization. GLCMs usually require a reduction of the bit depth used to describe voxel values: the number of possible co–occurrences is extremely large, leading to unstable values or extremely sparsely populated matrices. Run–length encoding (RLE) [7] was extended to 3D in [26]. Similarly to GLCMs, this method requires reducing gray–levels and choosing directions. The requirement of making arbitrary choices entails the risk of losing important information. However, in RLE, the scale is not fixed and the descriptor can encode several dominant scales (run–lengths).

While these methods presented simple solutions to deal with texture directionality and rotation–invariance, they do not encode image scales and directions in a systematic fashion. Wavelets constitute an attractive solution to fully exploit texture information at multiple scales through multi–resolution image representations, where their Fourier transform is systematically split into a series of dyadic bands [27]. To that end, wavelets were successfully used for texture analysis in 2D [11], [14], [28]–[30] and 3D [31]–[33]. Fully leveraging the directional information of textures is a challenging problem, which is even more difficult in 3D. In most cases, local rotation–invariance is required because class–specific texture primitives or textons as well as the essential matter "stiches" have random local orientations [34]. In addition, the local organization of image directions (LOID), or how directional structures intersect, is fundamental for texture segregation [35]. Therefore, an inherent dilemma of directional texture analysis is to either use (i) texture operators that are insensitive to directions and invariant to rotations (such as measurements based on gray–level matrices averaged over multiple directions or isotropic filters and wavelets) or (ii) use directional operators that are able to characterize the LOIDs but not in a rotation–invariant fashion. In particular, isotropic–wavelets are rotation–invariant but they are not able to encode the directionality of the texture. The latter can be recovered by coupling each isotropic bandpass filter with directional filters, resulting in a loss of rotation–invariance.

The d–dimensional Riesz–wavelet transform combines isotropic–wavelets and directional all–pass filters [36], [37]. The filters behave like local higher–order partial image derivatives. Several papers have already shown the benefits of the 3D Riesz–wavelet transform when analyzing 3DST [33], [38]–[41]. The order of the Riesz transform controls the richness of the directional filterbank, where increasingly complex structures can be described with higher orders. A key property of Riesz–wavelets is steerability [42]–[44]. This means that the local response of each Riesz component to an image rotated by an arbitrary angle can be derived analytically from a linear combination of the responses of all components of the filterbank. Therefore, the descriptor can be locally aligned analytically (meaning that it does not require convolving the image with the rotated filter) without losing the directional information. This property is known as rotation–covariance and it allows comparing textures with arbitrary local orientations [14], [39]. The steerability of the 3D Riesz–wavelet transform is described in the literature [13], [37] in which case the local image directions are known. The estimation of the latter in the context of complex 3D texture patterns is challenging. Chenouard *et al.* proposed to estimate local image orientations when using 1st–order Riesz–wavelets [45] that encode simple image gradients. In most cases, the characterization of complex structures containing subtle directional information requires using higher–order Riesz filters.

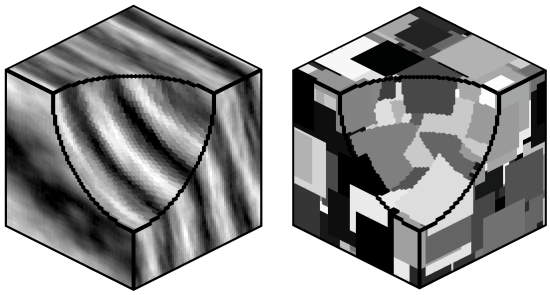

Fig. 1. Two examples of 3DST images where a circular section is removed to show the solidity of the interior.

The main contribution of this paper is to develop and compare three methods for estimating local texture orientations from Riesz transforms of any order in the context of 3DST analysis. The paper is structured as follows: Section II defines 3DST information and the database used in the experiments. Section III introduces the formulation and definitions required to understand the alignment methods presented in Section IV. The experimental setup is then detailed in Section V, and the results are presented in Section VI. The discussion of the results takes place in Section VII, followed by conclusions and perspectives (Section VIII).

II. 3D Solid Texture (3DST)

A 3D solidly textured image corresponds to a uniformly textured volume in 3D. Textures existing in more than two dimensions cannot be fully visualized by humans. Only virtual navigation in multi–planar rendering or semi–transparent visualization allows visualizing subregions in 2D projections or slices. Figure 1 shows two examples of 3DST images.

The methods developed in this work were evaluated using the RFAI database[1] (Reconnaissance de Formes, Analyse

[1] http://www.rfai.li.univ-tours.fr/fr/ressources/3Dsynthetic_images_database.html, as of June 13, 2016.

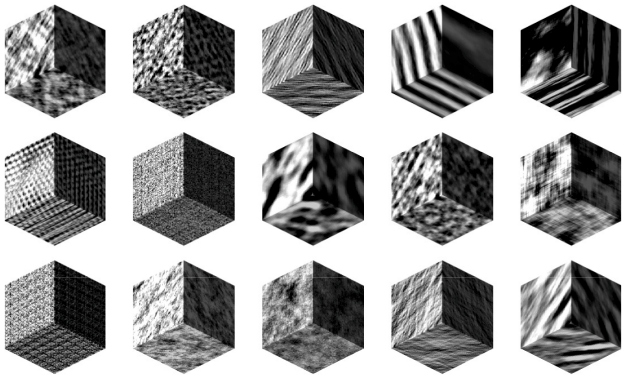

Fig. 2. A total of 15 classes compose the *Fourier* dataset of the RFAI database. A three–dimensional visualization of one instance of every class is shown. Each instance is a 3D image of $64 \times 64 \times 64$ voxels.

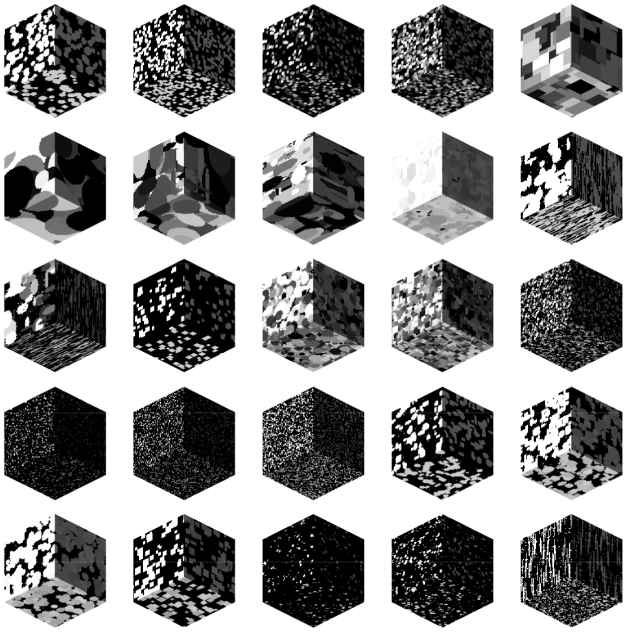

Fig. 3. The *Geometric* dataset of the RFAI database is composed of 25 classes. Each instance is a 3D image of $64 \times 64 \times 64$ voxels.

d'Images) of 3D synthetic textured images [46]. This is one of the few 3DST databases available with clear ground truth. Two datasets of this database containing images of size $64 \times 64 \times 64$ voxels of different nature were chosen. The first dataset is called *Fourier* and contains 15 classes of texture built from synthetic distributions in the Fourier domain. The second is called *Geometric* and contains 25 classes of texture constructed using random positioning of geometric shapes such as spheres, cubes, and ellipses. A detailed description of the two datasets can be found in [46]. Figures 2 and 3 show one instance of every class for the *Fourier* and *Geometric* datasets respectively. Each class contains ten instances generated with the same random process. Figure 4 shows the ten instances of one class of the *Geometric* dataset. Four additional subsets for each dataset were created by applying various alterations

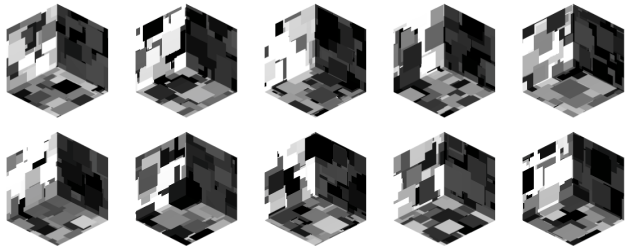

Fig. 4. Each class in both subsets (*Fourier* and *Geometric*) contains ten instances. The ten images of one class of the *Geometric* dataset are shown. They were generated with random positioning of the same basic shape, that in this example corresponds to cubes with a uniform gray level value.

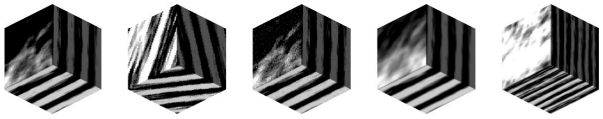

Fig. 5. Every image in the database is provided together with four alterations: *Normal* (no alteration), *Noise* (Gaussian noise), *Smooth* (Gaussian blur), *Rotate* (3D rotation) and *Subsampling*. This figure shows one image of one class of the *Geometric* dataset and its four alterations.

to the initial images. In total, five subsets are available per dataset corresponding to the following alterations applied to each image: *Normal* (no alteration), *Noise* (Gaussian noise), *Smooth* (Gaussian blur), *Rotate* (3D rotation) and *Subsampling*. An example of each of the alterations applied to one image is shown in Figure 5.

The RFAI database was built to provide reference 3DST data for comparing approaches of 3D texture segmentation, retrieval and classification. For the segmentation task, the database was used to evaluate several state–of–the–art descriptors in [47] and [48] and to design new features capable to encode the human perception of texture [49]–[51]. Suzuki *et al.* used the database to evaluate a texture retrieval technique developed previously [52], [53], as well as a novel key–point detector [54]. For the classification task, Maani *et al.* used this database to compare existing 3DST descriptors in [55]. These were: 3D GLCM [25], 3D LBP [56], second orientation pyramid (SOP) filtering [57], [58] and a novel approach based on the local frequency descriptor (LFD) [59]. Unfortunately, only the *Normal* subset of the *Fourier* dataset was used and can thus be compared to our approach. However, these classification results are used as a baseline for this subset. The accuracies of these methods on the chosen subset were: LFD 0.99 ± 0.01, 3D GLCM 0.73 ± 0.04, 3D LBP 0.80 ± 0.04, and SOP 0.97 ± 0.02. Wagner *et al.* selected nine classes of this database with a similar appearance to biological textures to test their classification method in [60]. A direct comparison with the performance achieved by our method is difficult, since the aforementioned studies did not specify the validation strategy used to estimate the classification performance, so how the data was partitioned between training and test data. The rotation subset was excluded from their experiments, which limits the evaluation of the rotation–invariance of the methods used.

To summarize, this database was used for testing descriptors with several designs but the evaluation was most often reduced to artificially chosen subsets and classes. In this work, all texture alterations provided by the authors of the RFAI database were used to evaluate the robustness of the proposed methods. The two datasets chosen, *Fourier* and *Geometric*, are complementary in the sense that the former contains textures with well defined properties in the Fourier domain, while the latter includes textures based on geometric spatial shapes. There is a lack of real 3DST in this database as it only contains synthetic textures. Nonetheless, the two selected datasets together include a broad range of texture patterns that can be found in areas such as biomedical imaging [1], and material composition [2].

III. Higher–Order Riesz–Wavelet Transform

This section presents all the preliminary knowledge required to introduce the methods explained in Section IV. We first introduce the notation used in this paper that is common to the state–of–the–art. Then, the higher–order Riesz transform is defined for a 3D image. The following subsection shows how to steer the Riesz components when the orientation of the original image is known. The coupling of the Riesz transform with isotropic–wavelets is explained in the next subsection. It also explains how the steering of the Riesz components commutes with the isotropic–wavelet framework. The approach proposed by Chenouard and Unser [45] to estimate the local orientation of 1st–order Riesz components is then introduced. The formal definition of rotation–covariance follows. Finally, a differentiation of the Riesz components based on their profile is defined. This distinction is required to highlight the importance of the methods proposed in this work.

A. Notation

1) The image as a 3D Signal: A 3D image is considered a 3D signal f indexed by the continuous–domain space variable $\boldsymbol{x} = (x_1, x_2, x_3) \in \mathbb{R}^3$. The 3–dimensional Fourier transform $\mathcal{F}$ of f is noted as:

$$f(\boldsymbol{x}) \overset{\mathcal{F}}{\longleftrightarrow} \hat{f}(\boldsymbol{\omega}) = \int_{\mathbb{R}^3} f(\boldsymbol{x}) e^{-j\langle \boldsymbol{\omega}, \boldsymbol{x}\rangle} \mathrm{d}x_1 \mathrm{d}x_2 \mathrm{d}x_3, \qquad (1)$$

with the pulsation vector $\boldsymbol{\omega} = (\omega_1, \omega_2, \omega_3) \in \mathbb{R}^3$.

2) Multi–Index Notation: Following the notation used in [37], we consider 3D index vectors of the form $\boldsymbol{n} = (n_1, n_2, n_3) \in \mathbb{N}^3$. The following multi–index notations and operators are used:

- Sum of components: $|\boldsymbol{n}| = n_1 + n_2 + n_3$,
- max of components: $\max(\boldsymbol{n}) = \max(n_1, n_2, n_3)$,
- factorial: $\boldsymbol{n}! = n_1! n_2! n_3!$,
- exponentiation of a vector $\boldsymbol{v} = (v_1, v_2, v_3) \in \mathbb{R}^3$: $\boldsymbol{v}^{\boldsymbol{n}} = v_1^{n_1} v_2^{n_2} v_3^{n_3}$.

B. Higher–Order Riesz Transforms of 3D Signals

Unser and Van de Ville presented in [43] the Nth–order Riesz transform of a d–dimensional signal $\boldsymbol{\mathcal{R}}^{(N)}\{f\}$. In the case of 3–dimensional signals, the Riesz transform of order $N \in \mathbb{N}$ is composed of $M = \frac{(N+2)(N+1)}{2}$ components that are denoted by $\mathcal{R}^{\boldsymbol{n}}$, with $\boldsymbol{n} = (n_1, n_2, n_3) \in \mathbb{N}^3$ such that $|\boldsymbol{n}| = N$. Given $\boldsymbol{\omega} = (\omega_1, \omega_2, \omega_3) \in \mathbb{R}^3$, each of these operators is an all–pass filter with the directional frequency response defined in Fourier as

$$\widehat{\mathcal{R}^{\boldsymbol{n}}}(\boldsymbol{\omega}) = (-j)^N \sqrt{\frac{N!}{n_1! n_2! n_3!}} \frac{\omega_1^{n_1} \omega_2^{n_2} \omega_3^{n_3}}{(\omega_1^{n_1} + \omega_2^{n_2} + \omega_3^{n_3})^{\frac{N}{2}}}. \qquad (2)$$

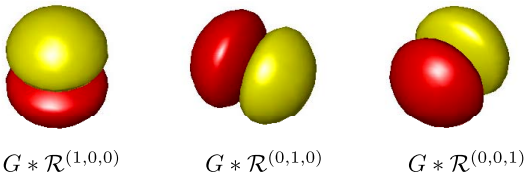

Fig. 6. 1st–order Riesz filters $\mathcal{R}^{\boldsymbol{n}}$ convolved with Gaussian kernels $G(\boldsymbol{x})$. The three filters have exactly the same profile but different orientations. They are oriented along each of the three directions x_1, x_2 and x_3 of $\mathbb{R}^3$.

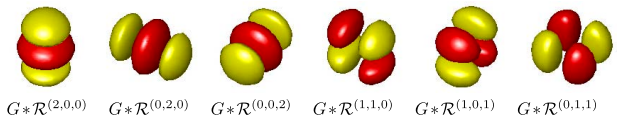

Fig. 7. 2nd–order Riesz filters $\mathcal{R}^{\boldsymbol{n}}$ convolved with Gaussian kernels $G(\boldsymbol{x})$. There are two groups of three filters with the same profile but different orientations.

Figures 6 and 7 show the 1st– and 2nd–order Riesz components. They are presented convolved with Gaussian kernels in order to visualize the profile since the support of the components is all $\mathbb{R}^3$. The 2nd–order Riesz filters shown in Figure 7 are used to illustrate properties of the higher–order components in further sections.

C. Steerability

The Riesz filterbanks are steerable [43], [44], which means that the local response of each component $\mathcal{R}^{\boldsymbol{n}}$ of an image $f(\boldsymbol{x})$ rotated by an arbitrary rotation matrix $\mathbf{R} \in \mathbb{R}^{3\times3}$ (represented by $f_{\mathbf{R}}$) can be derived analytically from a linear combination of the responses of all components of the filterbank, using a steering matrix $\boldsymbol{S}_{\mathbf{R}}$ as

$$\boldsymbol{\mathcal{R}}\{f_{\mathbf{R}}\} = \boldsymbol{S}_{\mathbf{R}} \boldsymbol{\mathcal{R}}\{f\}, \qquad (3)$$

where $\boldsymbol{\mathcal{R}}$ denotes the Riesz transform of any order.

Unser *et al.* demonstrated in [13, Th. 1] that for a given 3D rotation matrix $\mathbf{R} = (\boldsymbol{r}_1, \boldsymbol{r}_2, \boldsymbol{r}_3)^T$ with $\boldsymbol{r}_i \in \mathbb{R}^3$, the steering matrix $\mathbf{S}_{\mathbf{R}}$ with elements $s_{\boldsymbol{n},\boldsymbol{m}}$ (in multi–index notation) is defined as

$$s_{\boldsymbol{n},\boldsymbol{m}} = \sqrt{\frac{\boldsymbol{m}!}{\boldsymbol{n}!}} \sum_{|\boldsymbol{k}_1|=n_1} \sum_{|\boldsymbol{k}_2|=n_2} \sum_{|\boldsymbol{k}_3|=n_3} \delta_{\boldsymbol{k}_1+\boldsymbol{k}_2+\boldsymbol{k}_3,\boldsymbol{m}} \cdot \frac{\boldsymbol{n}!}{\boldsymbol{k}_1! \boldsymbol{k}_2! \boldsymbol{k}_3!} \boldsymbol{r}_1^{\boldsymbol{k}_1} \boldsymbol{r}_2^{\boldsymbol{k}_2} \boldsymbol{r}_3^{\boldsymbol{k}_3}, \qquad (4)$$

where $\boldsymbol{k}_i \in \mathbb{N}^3$ and $\delta_{\boldsymbol{k}_1+\boldsymbol{k}_2+\boldsymbol{k}_3,\boldsymbol{m}}$ is the Kronecker symbol used to exclude the summation terms with $\boldsymbol{k}_1 + \boldsymbol{k}_2 + \boldsymbol{k}_3 \neq \boldsymbol{m}$.

This steering matrix preserves the inner–structure of the Riesz filters, i.e., rotating their coefficients in a coherent way by preserving the angles between the elements of the filterbank.

A consequence of this theorem is that, if the rotation matrix $\mathbf{R}$ is known, it is possible to analytically align the Riesz coefficients applying the steering matrix $\mathbf{S_R}$, where the Riesz transform of $f(\boldsymbol{x})$ is computed only once.

D. 3D Riesz–Wavelet Pyramid

The Riesz transform has the property to map any frame of $L_2(\mathbb{R}^3)$ (including wavelet frames) into $L_2(\mathbb{R}^3)$ since it preserves the inner product of $L_2(\mathbb{R}^3)$ [36]. In this work, we used the Riesz–wavelet construction proposed by Chenouard and Unser in [45]. The latter consists in applying the 3D Riesz transform to the coefficients of an isotropic–wavelet pyramid[2] to build a steerable wavelet transform in 3D. In this case, the Riesz and wavelet transforms can be commuted. The Riesz transform is applied to each scale of the isotropic pyramid defined by the wavelet function ψ. When using an isotropic primary wavelet, the directionality information is encoded at each scale of the pyramid by the Riesz transform only. The Riesz–wavelet coefficients can thus be steered using the same steerability matrix $\mathbf{S_R}$ as the Riesz coefficients (see Eq. (3)). Specifically, the coefficients $q_{k,\mathbf{U}}(x)$ corresponding to a rotation of the Riesz–wavelet atoms by the unitary matrix $\mathbf{U}$ are computed as:

$$q_{k,\mathbf{U}}(\boldsymbol{x}) = \boldsymbol{\mathcal{R}}_{\mathbf{U}}\{\psi_k * f\}(\boldsymbol{x}) = \mathbf{U}\boldsymbol{\mathcal{R}}\{\psi_k * f\}(\boldsymbol{x}) = \mathbf{U}q_k(\boldsymbol{x}), \tag{5}$$

where ψ_k is the ψ isotropic wavelet at scale k.

E. Tensor–Based Estimation of Local Orientations

In [45], Chenouard and Unser presented a tensor–based estimation of the local orientation for the 1st–order Riesz transform at a position $\boldsymbol{x}_0$. This estimation relies on computing the eigenvectors of the tensor matrix $\mathbf{J}(\boldsymbol{x}_0)$, with

$$\mathbf{J}(\boldsymbol{x}_0) = \begin{pmatrix} \mathcal{R}_1^2\{g*f\}(\boldsymbol{x}_0) & \mathcal{R}_1\mathcal{R}_2\{g*f\}(\boldsymbol{x}_0) & \mathcal{R}_1\mathcal{R}_3\{g*f\}(\boldsymbol{x}_0) \\ \mathcal{R}_2\mathcal{R}_1\{g*f\}(\boldsymbol{x}_0) & \mathcal{R}_2^2\{g*f\}(\boldsymbol{x}_0) & \mathcal{R}_2\mathcal{R}_3\{g*f\}(\boldsymbol{x}_0) \\ \mathcal{R}_3\mathcal{R}_1\{g*f\}(\boldsymbol{x}_0) & \mathcal{R}_3\mathcal{R}_2\{g*f\}(\boldsymbol{x}_0) & \mathcal{R}_3^2\{g*f\}(\boldsymbol{x}_0) \end{pmatrix}, \tag{6}$$

where $g(\boldsymbol{x})$ is the regularization function of the orientation map, in our case a 3D Gaussian window, and $\boldsymbol{\mathcal{R}} = (\mathcal{R}^{(1,0,0)}, \mathcal{R}^{(0,1,0)}, \mathcal{R}^{(0,0,1)}) = (\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3)$.

The collection of eigenvectors of $\mathbf{J}(\boldsymbol{x}_0)$ sorted by eigenvalue, defines a rotation matrix $\mathbf{U}_g$. For each location $\boldsymbol{x}_0$, the resulting matrix $\mathbf{U}_g$ maximizes the energy of the first component of the rotated Riesz transform $\mathbf{U}_g\boldsymbol{\mathcal{R}}\{f\}$. It then maximizes the residual energy for the second component, and then for the third.

[2]A redundant pyramid is used to ensure translation–invariance of the wavelet transform.

F. Rotation–Covariance

Orienting the Riesz operators $\mathcal{R}_i$ locally following the approach explained in Section III-E, is referred to as *rotation–covariance* (represented by $\boldsymbol{\mathcal{R}}^{\mathrm{RC}}\{f\}(\boldsymbol{x})$), where the organization of image directions is characterized independently from its local orientation [14], [39]. The latter differs from the monogenic signal [45], since all the scales of the wavelet frame are aligned using the same rotation matrix $\mathbf{U}_g$ derived from the highest image resolution regularized by $g(\boldsymbol{x})$. It also differs from rotation–invariant operators since $\boldsymbol{\mathcal{R}}^{\mathrm{RC}}\{f\}(\boldsymbol{x})$ is directional. Rotation–covariance is warranted while the Riesz operators are steered coherently. In the case of the 2nd–order Riesz transform, if after steering the components the response corresponding to the component $G * \mathcal{R}^{(0,2,0)}$ is moved to the component $G * \mathcal{R}^{(2,0,0)}$, then, to preserve the relation between the components, the response to the component $G * \mathcal{R}^{(0,1,1)}$ needs to be moved to the component $G*\mathcal{R}^{(1,0,1)}$ (see Figure 7). The steering matrix $\mathbf{S_R}$ defined in Eq. (4) preserves the coherence between the components. In the case of the 1st–order Riesz transform, as defined in Section III-E, given a rotation matrix $\mathbf{U}_g$, it results that $\mathbf{S}_{\mathbf{U}_g} = \mathbf{P}\mathbf{U}_g$, where $\mathbf{P} \in \mathbb{R}^{3\times3}$ is a permutation matrix. $\mathbf{P}$ does not affect the property of rotation–covariance, thanks to the equivalence of the profiles of the Riesz components of order 1 (see Figure 6).

G. Uni–Directional and Multi–Directional Riesz Components

In the following sections, we distinguish uni– from multi–directional components. This distinction is based on the directional profiles of the Riesz components.

Definition 1: We define as uni–directional Riesz components *the components* $\mathcal{R}^{\boldsymbol{n}}$ *with* $\max(\boldsymbol{n}) = N$. Multi–directional Riesz components *are then those with* $\max(\boldsymbol{n}) < N$.

As an example, Figure 7 contains three uni–directional components ($G*\mathcal{R}^{(2,0,0)}$, $G*\mathcal{R}^{(0,2,0)}$, and $G*\mathcal{R}^{(0,0,2)}$) and three multi–directional components ($G*\mathcal{R}^{(1,1,0)}$, $G*\mathcal{R}^{(1,0,1)}$, and $G*\mathcal{R}^{(0,1,1)}$). For a 3–dimensional signal there are always three uni–directional Riesz components. These filters are encoding the variations of a signal only along one of the three main directions of the image $f(\boldsymbol{x})$. In the particular case of the 1st–order Riesz transform, all components are uni–directional (see Figure 6).

IV. Methods

Based on these points, the open questions are the following: "How can we estimate local image orientations based on higher–order Riesz transforms?" and, "Can we apply the same technique as defined in Section III-E?". In this section, three approaches for estimating local image orientations based on 3D Riesz transforms of any order are proposed and qualitatively compared.

A. Local Image Orientation Estimation Based on M–D Riesz Representations

This section presents the straightforward extension of the technique explained in Section III-E from the 1st– to the

Nth–order. The Nth–order Riesz transform is composed of M components. This extension consists in building a matrix $\mathbf{J}(\boldsymbol{x}_0) \in \mathbb{R}^{M\times M}$ with all the Riesz components $(\mathcal{R}_1, \ldots, \mathcal{R}_M)$ as

$$\mathbf{J}(\boldsymbol{x}_0) = \begin{pmatrix} \mathcal{R}_1^2\{g * f\}(\boldsymbol{x}_0) & \cdots \mathcal{R}_1\mathcal{R}_M\{g * f\}(\boldsymbol{x}_0) \\ \vdots & \vdots \\ \mathcal{R}_M\mathcal{R}_1\{g * f\}(\boldsymbol{x}_0) \cdots & \mathcal{R}_M^2\{g * f\}(\boldsymbol{x}_0) \end{pmatrix}. \quad (7)$$

It is worth to notice that the Riesz filters $(\mathcal{R}_1, \ldots, \mathcal{R}_M)$ do not have the same profile when $N > 1$. This was not the case in Eq. (6), where the three filters only differed in their orientation and yet had the same profile (see Figures 6 and 7). Following Eq. (6) in Section III-E, an M–dimensional unitary matrix $\mathbf{U}_g$ can be built from a sorted collection of the eigenvectors of $\mathbf{J}(\boldsymbol{x}_0)$. $\mathbf{U}_g$ can be used as a rotation matrix in $\mathbb{R}^M$ to align the Riesz transform. Eq. (8) shows the final form of this aligned transform $\boldsymbol{\mathcal{R}}^{M\mathrm{D}}$.

$$\boldsymbol{\mathcal{R}}^{M\mathrm{D}}\{f\} = \mathbf{U}_g\boldsymbol{\mathcal{R}}\{f\} \quad (8)$$

This procedure yields rotation–invariant texture descriptors but it does not rotate the Riesz components coherently, breaking then their directional inner structure. In this case, $\mathbf{U}_g$ cannot be used as a steering matrix following Eq. (4) (see Section III-F). This method reorders the responses of the Riesz filters independently from the directional pattern that they are encoding, discarding the property of rotation–covariance as well as the ability to characterize the local organization of image directions. We demonstrate why the inner–structure of the Riesz filterbanks is not preserved with the application of $\mathbf{U}_g$ and what is its inconvenience. Proposition 1 demonstrates that the result of applying $\mathbf{U}_g$ to a given Riesz vector $\boldsymbol{\mathcal{R}}$ always produces the same aligned vector, independently of the initial order of the components.

Proposition 1: Let $\boldsymbol{\mathcal{R}}(\boldsymbol{x}_0) \in \mathbb{R}^M$ be the Nth–order Riesz transform at the position $\boldsymbol{x}_0 \in \mathbb{R}^3$ of $f(\boldsymbol{x}_0)$, and let $\mathbf{J}(\boldsymbol{x}_0)$ be the matrix defined in Eq. (7). Let $\mathbf{U}$ be the matrix composed of the sorted collection of eigenvectors of $\mathbf{J}(\boldsymbol{x}_0)$. Given a permutation matrix $\mathbf{P} \in \mathbb{R}^{M\times M}$, we define $\tilde{\boldsymbol{\mathcal{R}}}(\boldsymbol{x}) = \mathbf{P}\boldsymbol{\mathcal{R}}(\boldsymbol{x}), \forall \boldsymbol{x} \in \mathbb{R}^3$. We name $\tilde{\mathbf{J}}(\boldsymbol{x}_0)$ the matrix defined in Eq. (7) for $\tilde{\boldsymbol{\mathcal{R}}}$. Then, $\forall\, \mathbf{P}, \mathbf{U}\boldsymbol{\mathcal{R}}(\boldsymbol{x}_0) = \tilde{\mathbf{U}}\tilde{\boldsymbol{\mathcal{R}}}(\boldsymbol{x}_0)$, where $\tilde{\mathbf{U}}$ is the sorted collection of eigenvectors of $\tilde{\mathbf{J}}(\boldsymbol{x}_0)$.

Proof: To simplify the notation we omit specifying the position $\boldsymbol{x}_0$.

It is straightforward that $\tilde{\mathbf{J}} = \mathbf{P}\mathbf{J}\mathbf{P}^{-1}$. This expression corresponds to a change of the basis and the eigenvalues of the matrices $\tilde{\mathbf{J}}$ and $\mathbf{J}$ are the same. Let Λ be the diagonal matrix containing the eigenvalues of $\mathbf{J}$ and $\tilde{\mathbf{J}}$ ordered in descending order. Then,

$$\tilde{\mathbf{J}} = \mathbf{P}\mathbf{J}\mathbf{P}^{-1} = \mathbf{P}\mathbf{U}^T\Lambda(\mathbf{U}^T)^{-1}\mathbf{P}^{-1} = \mathbf{P}\mathbf{U}^T\Lambda(\mathbf{P}\mathbf{U}^T)^{-1}. \quad (9)$$

Since $\tilde{\mathbf{J}} = \tilde{\mathbf{U}}^T\Lambda(\tilde{\mathbf{U}}^T)^{-1}$, then $\tilde{\mathbf{U}}^T = \mathbf{P}\mathbf{U}^T$, i.e., $\tilde{\mathbf{U}} = \mathbf{U}\mathbf{P}^T$. Hence,

$$\tilde{\mathbf{U}}\tilde{\boldsymbol{\mathcal{R}}} = \mathbf{U}\mathbf{P}^T\mathbf{P}\boldsymbol{\mathcal{R}} = \mathbf{U}\boldsymbol{\mathcal{R}}, \quad (10)$$

because by definition $\mathbf{P}^T = \mathbf{P}^{-1}$ as $\mathbf{P}$ is a permutation matrix. ∎

The limitation of this approach is exemplified when two different patterns yield identical overall filterbank responses but are distributed in components with different profiles. In this case, aligning the Riesz components with the M–dimensional matrix $\mathbf{U}_g$ produces the same vector $\tilde{\boldsymbol{\mathcal{R}}}$ for both texture patterns, even if they differ. This method entails the risk of mixing image responses to Riesz components with different profiles, creating a rotation–invariant but not rotation–covariant descriptor, i.e., not preserving the directional relations between filters. The method only assigns the response of the filter with the highest energy to the first component of the Riesz transform, no matter which is the first component. The resulting feature vector is always the same for any sorting of the Riesz components. This technique is referred to as *M–dimensional* alignment in the following sections.

B. Local Image Orientation Estimation Based on Uni–Directional Riesz Components

This method only uses the Riesz coefficients corresponding to the uni–directional Riesz components to estimate local image orientations (see Def. 1). These three components are orthogonal and encode the Nth–order image derivatives along each 3D direction $(\partial^N/\partial x_1^N, \partial^N/\partial x_2^N, \partial^N/\partial x_3^N)$. Their profiles are rotated versions of each other and encode the same type of texture patterns but along different directions. Every Nth–order Riesz transform contains exactly three uni–directional Riesz components.

For this alignment, the tensor $\mathbf{J}(\boldsymbol{x_0})$ proposed in Eq. (6) is constructed using the response at the position $\boldsymbol{x}_0$ to these three filters. We refer to this tensor as $\mathbf{J}^{\mathrm{Ud}}(\boldsymbol{x_0})$. The rotation matrix $\mathbf{U}_g^{\mathrm{Ud}}$ can be computed from the eigenvectors of $\mathbf{J}^{\mathrm{Ud}}(\boldsymbol{x_0})$ (see Section III-E). Eq. (4) yields the steering matrix $\mathbf{S}_{\mathbf{U}_g^{\mathrm{Ud}}}$, that is $M \times M$. This steering matrix is used to align the Riesz components as shown in Eq. (11).

$$\boldsymbol{\mathcal{R}}^{\mathrm{Ud}}\{f\} = \mathbf{S}_{\mathbf{U}_g^{\mathrm{Ud}}}\boldsymbol{\mathcal{R}}\{f\} \quad (11)$$

In this particular case, the estimation of the local orientation is solely based on the uni–directional components of the Nth–order Riesz transform. All Riesz components are subsequently aligned with the M–dimensional steering matrix $\mathbf{S}_{\mathbf{U}_g^{\mathrm{Ud}}}$. This technique is referred to as *uni–directional* alignment in the following sections.

C. Local Orientation Estimation Based on 1st–Order Riesz Components

The estimation of the local orientation described in Section III-E can be used to align higher–order filters. Using the 1st–order filters, the local orientations of the image are estimated by the technique explained in Section III-E, yielding the matrix $\mathbf{U}_g$. In this case, we refer to this unitary matrix as $\mathbf{U}_g^{\mathrm{O1}}$ since only the 1st–order filters are used. Applying then Eq. (4), the M–dimensional steering matrix $\mathbf{S}_{\mathbf{U}_g^{\mathrm{O1}}}$ for the Nth–order Riesz transform is computed. Then, the M–dimensional vector of Riesz components can be aligned applying this steering matrix (see Eq. (12)).

$$\boldsymbol{\mathcal{R}}^{\mathrm{O1}}\{f\} = \mathbf{S}_{\mathbf{U}_g^{\mathrm{O1}}}\boldsymbol{\mathcal{R}}\{f\} \quad (12)$$

This technique is referred to as *1st–order* alignment in the following sections. In this case, the local orientation is computed based on image gradients. Since the estimation of the local orientations is always based on the 1st–order Riesz components, it remains stable when the order of the transform increases. Thanks to steerability, the three uni–directional components of the 1st–order Riesz transform cover all 3D image directions. However, this coverage is not complete when only using uni–directional components of the higher–order Riesz transform. In addition, the 1st–order Riesz transform of the image is required for this alignment, but since higher–order Riesz transforms can be obtained from recursions of the 1st–order Riesz transforms, the latter can be kept for further estimation of the local orientations.

V. Experimental Setup

Our methods were evaluated and compared across five orders of the Riesz transform (from 0 to 4). Order 0 means that only the wavelet framework was applied, resulting in only one feature per spatial point and scale. To maintain the property of translation–invariance, the undecimated wavelet transform was used. The wavelet framework used is based on isotropic–wavelets with optimal bandwidth [33], [61]. The dimension M of the Riesz transform of each scale for the orders 1 to 4 are 3, 6, 10 and 15 respectively. The number of scales was set up with respect to the side length (c_v) of the cubic volumes as $\log_2(c_v) - 1$. Which results in five scales, as every image in the selected datasets contains $64 \times 64 \times 64$ voxels. The steering matrix $\mathbf{S_R}$ presented in Eq. (4) was only computed for the first scale and was then applied to all the scales. One important parameter when aligning the Riesz filters is the σ of the regularization Gaussian window applied to compute the neighborhood of the point $\boldsymbol{x}_0$ (see Sections III-E, IV-A, IV-B and IV-C).

The final feature space consisted in the concatenation of the energies of the Riesz–wavelet coefficients averaged over the entire solid volumes. One–versus–all support vector machine (SVM) [62] classifiers with a linear kernel were used to learn and predict the class labels. The LIBSVM library [63] was used. When using linear SVMs, the only parameter to optimize is the cost (C) of errors. Values from 10^{-4} to 10^4 with a logarithmic step of 1 were tested (9 values in total). A leave–one–out (LOO) cross–validation (CV) was used to optimize C and to estimate the testing performance. For all image alterations (including *Normal*) and each fold of the LOO CV, the training of the classifier was performed using exclusively the unaltered images (*Normal* subset).

The overall effectiveness of the methods proposed is compared using the average accuracy (Acc_{av}) of the classification. Acc_{av} is a trustworthy measure when evaluating classifiers on balanced classes with similar distributions [64]. The representation of classes in all subsets is strictly balanced, and the different elements inside a class were generated with a similar random process. The resulting classes have a similar distribution in the aligned and unaligned M–dimensional Riesz space. For a multi–class classification problem with l classes $\{Class_1, \ldots, Class_l\}$, the average accuracy is

$$Acc_{av} = \frac{\sum_{i=1}^{l} \frac{tp_i + tn_i}{tp_i + fn_i + fp_i + tn_i}}{l}, \tag{13}$$

where tp_i, fp_i, tn_i and fn_i are the true positives, false positives, true negatives and false negatives of $Class_i$ respectively. Acc_{av} is referred to as *accuracy*.

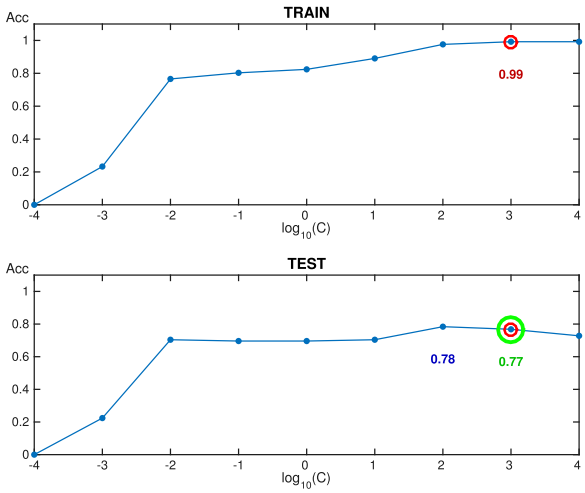

Fig. 8. The cost C is optimized for each fold of the LOO CV with the *Normal* subset. The accuracy of our method is defined as the accuracy obtained when using the value of C that performed best on the training set. This example is based on an M–dimensional alignment method, Riesz order 1, $\sigma = 3$, dataset *Geometric*, and testing with the *Rotate* subset. The maximum accuracy in the training phase (using the *Normal* dataset) was 0.99 and was achieved when $C = 10^3$ (marked in red). The accuracy on the test set (using the *Rotate* data) for this C was 0.77 and is marked in green. The selected accuracy on the test set was not the best achieved, which is 0.78 and is marked in blue). This method prevents data overfitting when selecting the value of C.

VI. Results

The accuracy shown in this section was obtained on the test sets of the LOO CV (see Section V). The cost parameter C that performed best in the training set, i.e., *Normal* subset, was used. Figure 8 details the procedure to select the best C from the training set.

This procedure is repeated for each Riesz order tested (from 0 to 4) and each alignment method explained in the previous sections. We recall that order 0 means using the isotropic–wavelet framework only, which does not require aligning the Riesz components. The influence of the σ value of the regularization Gaussian window used to compute the local orientations was investigated for each alignment method and Riesz order. 8 σ values were chosen in [0.5; 4] with a step of 0.5. These bounds were chosen based on preliminary experiments but did not guarantee to find the optimal value of σ in all cases. Figure 9 shows a detailed example with the accuracy obtained for each σ. The σ did not affect the unaligned Riesz transform (dark blue).

Figures 10 and 11 show the accuracies obtained for the datasets *Fourier* and *Geometric* respectively. In order to summarize the results achieved (1,600 accuracies obtained in total after the parameter selection), only two values are shown

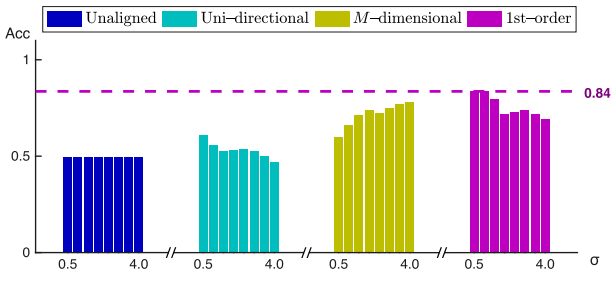

Fig. 9. Influence of tensor regularization σ on the classification accuracy. These accuracies were obtained with the subset *Rotate* of the *Geometric* dataset when using the 4th–order Riesz transforms. Each color corresponds to a different alignment technique. Dark blue corresponds to unaligned Riesz transforms. Each bar corresponds to the accuracy obtained for a single value of $\sigma \in [0.5; 4]$. The dashed line shows the maximum accuracy achieved and its color corresponds to the method achieving this maximum.

TABLE I

BEST CLASSIFICATION ACCURACY OBTAINED BY EACH ALIGNMENT METHOD FOR THE *Fourier* SUBSETS. THE LAST FOUR METHODS CORRESPOND TO THOSE PRESENTED IN [59] (SEE SECTION II)

	***Fourier* subsets**				
Method	*Normal*	*Rotate*	*Noise*	*Smooth*	*Subsampling*
Unaligned	0.99	0.87	0.99	0.83	0.27
Uni–directional	**1.00**	0.89	**1.00**	0.85	0.42
M–dimensional	**1.00**	**0.97**	**1.00**	0.85	**0.45**
1st–order	**1.00**	0.96	**1.00**	**0.87**	0.41
LFD	0.99 ± 0.01	–	–	–	–
3–D GLCM	0.73 ± 0.04	–	–	–	–
3–D LBP	0.80 ± 0.04	–	–	–	–
SOP	0.97 ± 0.02	–	–	–	–

TABLE II

BEST CLASSIFICATION ACCURACY OBTAINED BY EACH ALIGNMENT METHOD FOR THE *Geometric* SUBSETS

	***Geometric* subsets**				
Method	*Normal*	*Rotate*	*Noise*	*Smooth*	*Subsampling*
Unaligned	0.75	0.64	0.75	0.20	0.12
Uni–directional	0.99	0.77	0.95	0.34	0.20
M–dimensional	**1.00**	0.80	0.95	0.36	**0.21**
1st–order	0.99	**0.86**	**0.96**	**0.42**	0.20

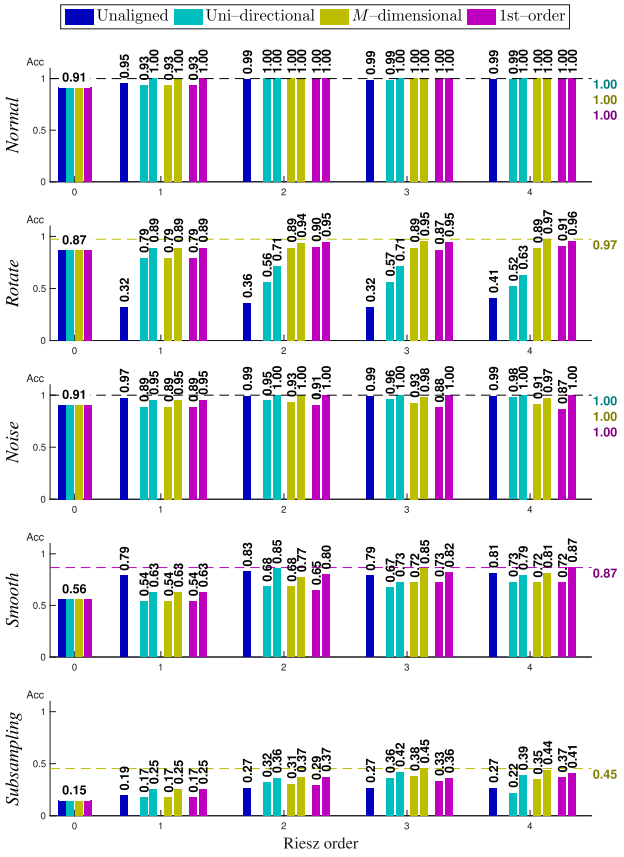

Fig. 10. Accuracies obtained for the *Fourier* dataset. Maximum and minimum accuracies with respect to σ are shown for each combination of alignment methods, Riesz order and texture alteration. For each subset, the maximum accuracy obtained is highlighted with a dashed line in the color corresponding to the method that achieved it. In the case of the *Normal* and *Noise* subsets, most aligning strategies achieved the maximum accuracy of 1.00 for orders > 1. The dashed line is then shown in black and one colored–value per method is shown at the right of the graph.

per alignment strategy (M–dimensional, uni–directional and 1st–order), Riesz order, and subset. These values correspond to the worst and best accuracies obtained when varying the σ value. For order 1, the three alignment methods are equivalent. In the case of order 0, only one bar is presented, as σ is not used. Only one bar is also shown for the unaligned method. Tables I and II summarize the results obtained for each subset, where the best performance achieved by each alignment method is shown. Moreover, Table I contains the performance of the methods obtained in [59] on the *Normal* subset.

a) Fourier *dataset:* The average accuracy of a random multi–class classifier is $1/l$, where l is the number of classes. In the case of the *Fourier* dataset that contains 15 classes this is 0.07. Figure 10 summarizes the results obtained for the *Fourier* dataset. For the subsets *Normal*, *Rotate* and *Noise*, the isotropic–wavelet framework (with Riesz order 0) achieved an accuracy above 0.87 in all cases. It did not perform as good for *Smooth* and *Subsampling*. It is worth noticing that for *Rotate*, it performed better than using the unaligned Riesz transforms of any order > 0. The advantage of using the aligned Riesz transform is highlighted in this subset. It can also be observed that the uni–directional alignment is not as good as any of the others. Moreover, this method performed worse when the Riesz order was increased. The M–dimensional and the 1st–order alignment strategies performed similarly for this dataset. Both had an overall performance better than the uni–directional and much better than the unaligned Riesz transform. Furthermore, their performance increased with the order of the Riesz transform.

b) Geometric *dataset:* The average accuracy of a random multi–class classifier for the *Geometric* dataset containing 25 classes is of 0.04. Figure 11 illustrates the results obtained for this dataset and shows that the accuracy of the unaligned Riesz transforms were below this threshold in some cases. This happens in the *Subsampling* subset for orders 2 to 4. Moreover, it only slightly reached this value for order 1. The aligned

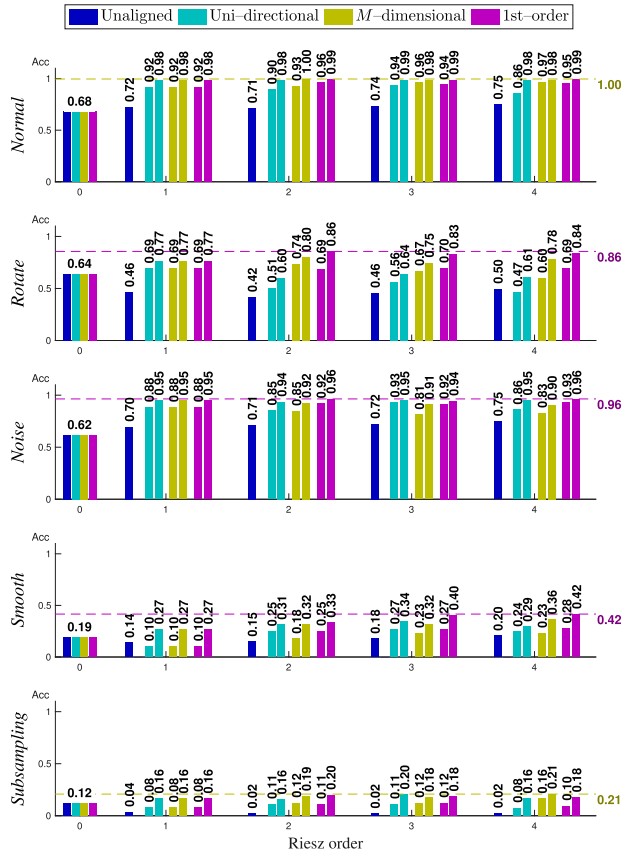

Fig. 11. Accuracies obtained for the *Geometric* dataset. Max and min accuracy with respect to σ are shown for each combination of alignment method, Riesz order and subset. For each subset, the maximum accuracy obtained is highlighted with a dashed line in the color of the corresponding method and its value is shown at the right of the graph.

methods performed much better on this dataset in comparison with the unaligned Riesz transform. The isotropic–wavelet framework alone (Riesz order 0) obtained similar performance in subsets *Normal*, *Rotate*, and *Noise* (between 0.64 and 0.68), but lower than in the *Fourier* dataset, which was above 0.87. The improvement when using any of the alignment strategies is very strong in the *Geometric* dataset. The 1st–order alignment method outperformed all others for *Rotate*, *Noise*, and *Smooth*, while the *M*–dimensional method performed similarly on the other two (*Normal* and *Subsampling*).

VII. Discussion

The results presented in Section VI show the benefits of both increasing the order of the Riesz transform and the necessity of aligning the Riesz-wavelet coefficients. The unaligned Riesz transform generally performed better than the isotropic–wavelet itself (Riesz order 0). However, in the case of rotated data, the latter clearly outperformed the unaligned Riesz transform.

We selected a linear SVM classifier to highlight the strength of the aligned descriptors, independently of the classifier used. Although non–linear classifiers could potentially perform better than the linear SVM, they require more parameters to tune, resulting in problem–specific approaches and with high risk of overfitting. In our experiments, the linear classifier can not handle the dispersion of the Riesz descriptors in the feature space caused by the 3D rotations applied to the 3DST. On the other hand, the isotropic–wavelet framework is rotation–invariant, and its performance is not affected by this alteration, showing similar performance in the *Normal* and the *Rotate* subsets of both datasets. This highlights the necessity of an aligning method, which regroups texture patterns with similar local organization of image directions in consistent regions of the feature space. The linear classifier can therefore better learn and perform with this cleaned feature space. In this case, all aligning strategies presented outperform the isotropic–wavelet framework (even with worst σ). Moreover, all best accuracies were obtained when using orders superior to 1, and most were obtained with the highest order tested. This supports the initial assumption that locally–oriented higher–order Riesz–wavelets are needed to characterize complex three–dimensional morphological properties of 3DST. However, increasing the order of the Riesz transform was not always beneficial, for example in the subset *Rotate* of the *Geometric* dataset. In particular, in the case of the uni–directional alignment applied to the *Rotate* subsets, augmenting the order of the Riesz transform was found to be a disadvantage. This is due to the use of only a subset of the components to compute the local orientation, which does not cover all possible local 3D image directions. The Riesz filters have the property to cover all directions but the range of directions covered by each component is inversely proportional to the order of the transform (i.e., the filters are increasingly selective in terms of directionality). This problem did not exist when using the *M*–dimensional or the 1st–order alignments. In both cases, all the components of the Riesz–wavelet transform are used, i.e., the M components of the Nth–order and the three components of the 1st–order respectively, covering all image directions. It is important to notice the stable behavior of the unaligned Riesz transforms when increasing the order. The performance is strictly related to the distribution of the descriptors in the feature space. In this case, increasing the order varies the dimension of the feature space but the feature vectors preserve a similar distribution. Therefore, the linear classifier performs similarly.

As expected, the alignment methods presented the highest improvement in the subset *Rotate* of both datasets when considering the worst and the best accuracies obtained. Particularly, the unaligned Riesz transform performed worse than the isotropic–wavelets in these sets. The benefits of the alignments were observed for most texture alterations of both datasets. However, all three alignment strategies proposed used the σ parameter (see Sections III-E, IV-A, IV-B and IV-C). This parameter that was already introduced in the tensor–based alignment for the 1st–order Riesz transform in [45], has shown to have a strong influence on the classification performance. Surprisingly, the *M*–dimensional alignment achieved one of the best overall results. This technique provides rotation–invariant descriptors but does not preserve the directional inner relations of the Riesz components (see Section IV-A). In the *Fourier* dataset, the benefits

of maintaining this inner–structure are not clear and both M–dimensional and 1st–order alignments behaved similarly. This is not the case for the *Geometric* dataset, where the sharp textures with strong directional gradients require these directional relations to be well described (see Figure 3). The 1st–order alignment preserves these inner–relations, achieving the highest performance on this dataset.

The overall best results are achieved by the 1st–order alignment. The reason behind this is that this alignment is the only one that uses all directions for all Riesz orders when estimating the local orientation of the texture, while keeping the rotation–covariance. Uni–directional alignment only uses all of the directions for Riesz order 1. Otherwise, it uses a subset of directions. On the other hand, the M–dimensional alignment always uses all directions. Nonetheless, as demonstrated in Proposition 1, it does not preserve the directional inner relations of the Riesz filters, therefore not preserving rotation–covariance.

As we mentioned before, a full comparison with other techniques was difficult, as only a part of the dataset had been used in publications previously. Considering the results found in the literature (see Table I), we can see that our techniques achieved what is considered a perfect classification in the subset where other techniques were evaluated. Only LFD achieved a similar accuracy.

The Riesz–wavelet framework used in this work relies on the frequency analysis of the texture. Therefore, this descriptor is more affected by changes in texture sharpness. The *Geometric* subset contains by construction very sharp transitions at the boundaries of the geometric shapes. Thus, it is consistent to observe an important degradation of performance for the subsets altered by Gaussian blur (*Smooth*) and subsampling. These two alterations result in frequency shifts towards the center of the spectrum. The *Fourier* subset contains fewer sharp spatial transitions and these alterations have a lower impact when using the Riesz–wavelet descriptor. Trying to classify blurred sharp patterns using well–defined sharp models in the training set is not common (the *Normal* subset was always used for training). This particular case could be addressed by applying a Gaussian smoothing before training the classifiers.

VIII. Conclusions and Future Work

In this work, we proposed a novel method for extracting rotation–covariant features from solid textures commonly found in biomedical imaging, material analysis, and structural geology. Three methods are based on locally–oriented wavelet transforms and extend the state–of–the–art techniques to use higher–order Riesz–wavelet transforms. These extensions were developed for any order of the Riesz transform but we limited our experiments to order 4. The three alignment strategies are equivalent for order 1, and also equivalent to the existing tensor–based approach for order 1 proposed by Chenouard and Unser in [45] (see Section III-E). A complete experimental evaluation of the classification capabilities of the aligned descriptors was performed using the RFAI dataset. The results show the benefits of increasing the order of the Riesz transforms and support the appropriate local alignment of the Riesz components. The performance of the three alignment strategies outperforms all unaligned Riesz–wavelet transforms. The comparison with state–of–the–art descriptors is difficult since there is no equivalent evaluation in the literature that covers all aspects of the same database. However, LFD, 3D GLCM, 3D LBP, and SOP descriptors were tested on the *Normal* subset of *Fourier* dataset (see Section II). The methods presented here outperform these techniques achieving an accuracy of 1.00 for all orders in this subset.

Next steps include a complete comparison between the methods presented and the state–of–the–art 3DST descriptors. Ideally, a database with real 3D data should be used for such an analysis. The RFAI database provides images with four alterations, but these alterations were not mixed. An intermediate step is to create a database combining possible alterations that can be found in real 3DST data. Moreover, these methods provide the opportunity to learn steerable models of class–specific local organization of image directions to estimate the local orientations, which showed to provide excellent results for 2D texture classification in [14].

References

[1] A. Depeursinge, A. Foncubierta-Rodríguez, D. Van De Ville, and H. Müller, "Three-dimensional solid texture analysis in biomedical imaging: Review and opportunities," *Med. Image Anal.*, vol. 18, no. 1, pp. 176–196, Jan. 2014.

[2] F. Hild, S. Roux, D. Bernard, G. Hauss, and M. Rebai, "On the use of 3D images and 3D displacement measurements for the analysis of damage mechanisms in concrete-like materials," in *Proc. Int. Conf. Fracture Mech. Concrete Concrete Struct.*, 2013, pp. 1–12.

[3] M. Zucali, M. Voltolini, B. Ouladdiaf, L. Mancini, and D. Chateigner, "The 3D quantitative lattice and shape preferred orientation of a mylonitised metagranite from Monte Rosa (Western Alps): Combining neutron diffraction texture analysis and synchrotron X-ray microtomography," *J. Struct. Geol.*, vol. 63, pp. 91–105, Jun. 2014.

[4] C. Blakemore and F. W. Campbell, "On the existence of neurones in the human visual system selectively sensitive to the orientation and size of retinal images," *J. Physiol.*, vol. 203, no. 1, pp. 237–260, 1969.

[5] B. M. ter Haar Romeny, "Multi-scale and multi-orientation medical image analysis," in *Biomedical Image Processing*. Heidelberg, Germany: Springer, 2011, pp. 177–196.

[6] R. M. Haralick, "Statistical and structural approaches to texture," *Proc. IEEE*, vol. 67, no. 5, pp. 786–804, May 1979.

[7] M. M. Galloway, "Texture analysis using gray level run lengths," *Comput. Graph. Image Process.*, vol. 4, no. 2, pp. 172–179, Jun. 1975.

[8] G. Thibault *et al.*, "Shape and texture indexes application to cell nuclei classification," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 27, no. 1, pp. 140–145, Feb. 2009.

[9] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[10] D. Marr and E. Hildreth, "Theory of edge detection," *Proc. Roy. Soc. London. B, Biol. Sci.*, vol. 207, pp. 187–217, Feb. 1980.

[11] T. S. Lee, "Image representation using 2D Gabor wavelets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 10, pp. 959–971, Oct. 1996.

[12] M. Varma and A. Zisserman, "A statistical approach to texture classification from single images," *Int. J. Comput. Vis.*, vol. 62, nos. 1–2, pp. 61–81, 2005.

[13] M. Unser, N. Chenouard, and D. Van De Ville, "Steerable pyramids and tight wavelet frames in $L_2(\mathbb{R}^d)$," *IEEE Trans. Image Process.*, vol. 20, no. 10, pp. 2705–2721, Oct. 2011.

[14] A. Depeursinge, A. Foncubierta-Rodríguez, D. Van de Ville, and H. Müller, "Rotation–covariant texture learning using steerable riesz wavelets," *IEEE Trans. Image Process.*, vol. 23, no. 2, pp. 898–908, Feb. 2014.

[15] A. P. Pentland, "Fractal-based description of natural scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 6, no. 6, pp. 661–674, Jun. 1984.

[16] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Supervised dictionary learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, pp. 1033–1040.

[17] M. Cimpoi, S. Maji, and A. Vedaldi. (2014). "Deep convolutional filter banks for texture recognition and segmentation." [Online]. Available: https://arxiv.org/abs/1411.6836

[18] H. Müller, N. Michoux, D. Bandon, and A. Geissbuhler, "A review of content-based image retrieval systems in medical applications—Clinical benefits and future directions," *Int. J. Med. Informat.*, vol. 73, no. 1, pp. 1–23, Feb. 2004.

[19] B. Bustos, D. Keim, D. Saupe, and T. Schreck, "Content-based 3D object retrieval," *IEEE Comput. Graph. Appl.*, vol. 27, no. 4, pp. 22–27, Jul. 2007.

[20] J. Fehr, "Rotational invariant uniform local binary patterns for full 3D volume texture analysis," in *Proc. Finnish Signal Process. Symp. (FINSIG)*, Oulu, Finland, 2007, pp. 1–6.

[21] T. Ahonen, J. Matas, C. He, and M. Pietikäinen, "Rotation invariant image description with local binary pattern histogram Fourier features," in *Image Analysis*. Heidelberg, Germany: Springer, 2009, pp. 61–70.

[22] J. Fehr and H. Burkhardt, "3D rotation invariant local binary patterns," in *Proc. 19th Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2008, pp. 1–4.

[23] A. Burner, R. Donner, M. Mayerhoefer, M. Holzer, F. Kainberger, and G. Langs, "Texture bags: Anomaly retrieval in medical images based on local 3D-texture similarity," in *Medical Content-Based Retrieval for Clinical Decision Support* (Lecture Notes in Computer Science), vol. 7075, H. Greenspan, H. Müller, and T. Syeda-Mahmood, Eds., Sep. 2012, vol. 7075. Heidelberg, Germany: Springer, pp. 116–127.

[24] A. S. Kurani, D.-H. Xu, J. Furst, and D. S. Raicu, "Co-occurrence matrices for volumetric data," in *Proc. 7th IASTED Int. Conf. Comput. Graphics Imag.*, Kauai, HI, USA, Aug. 2004, pp. 447–452.

[25] V. A. Kovalev, F. Kruggel, H.-J. Gertz, and D. Y. von Cramon, "Three-dimensional texture analysis of MRI brain datasets," *IEEE Trans. Med. Imag.*, vol. 20, no. 5, pp. 424–433, May 2001.

[26] D.-H. Xu, A. S. Kurani, J. D. Furst, and D. S. Raicu, "Run-length encoding for volumetric texture," in *Proc. 4th IASTED Int. Conf. Vis., Imag. Image Process. (VIIP)*, Marbella, Spain, Sep. 2004, pp. 1–6.

[27] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 7, pp. 674–693, Jul. 1989.

[28] S. Arivazhagan and L. Ganesan, "Texture classification using wavelet transform," *Pattern Recognit. Lett.*, vol. 24, nos. 9–10, pp. 1513–1521, Jun. 2003.

[29] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Trans. Image Process.*, vol. 4, no. 11, pp. 1549–1560, Nov. 1995.

[30] J. Portilla and E. P. Simoncelli, "A parametric texture model based on joint statistics of complex wavelet coefficients," *Int. J. Comput. Vis.*, vol. 40, no. 1, pp. 49–70, Oct. 2000.

[31] V. Barra and J.-Y. Boire, "Tissue segmentation on MR images of the brain by possibilistic clustering on a 3D wavelet representation," *J. Magn. Reson. Imag.*, vol. 11, no. 3, pp. 267–278, Mar. 2000.

[32] P. Cirujeda *et al.*, "A 3-D riesz-covariance texture model for prediction of nodule recurrence in lung CT," *IEEE Trans. Med. Imag.*, vol. 35, no. 12, pp. 2620–2630, Dec. 2016.

[33] A. Depeursinge *et al.*, "Optimized steerable wavelets for texture analysis of lung tissue in 3-D CT: Classification of usual interstitial pneumonia," in *Proc. IEEE 12th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2015, pp. 403–406.

[34] B. Julesz, "Textons, the elements of texture perception, and their interactions," *Nature*, vol. 290, pp. 91–97, Mar. 1981.

[35] J. R. Bergen and M. S. Landy, "Computational modeling of visual texture segregation," in *Computational Models of Visual Processing*. Cambridge, MA, USA: MIT Press, 1991, pp. 253–271.

[36] M. Unser, D. Sage, and D. Van De Ville, "Multiresolution monogenic signal analysis using the Riesz–Laplace wavelet transform," *IEEE Trans. Image Process.*, vol. 18, no. 11, pp. 2402–2418, Nov. 2009.

[37] N. Chenouard and M. Unser, "3D steerable wavelets in practice," *IEEE Trans. Image Process.*, vol. 21, no. 11, pp. 4522–4533, Nov. 2012.

[38] O. A. J. del Toro, A. Foncubierta–Rodríguez, M. I. V. Gómez, H. Müller, and A. Depeursinge, "Epileptogenic lesion quantification in MRI using contralateral 3D texture comparisons," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI* (Lecture Notes in Computer Science), vol. 16. Heidelberg, Germany: Springer, 2013, pp. 353–360.

[39] A. Depeursinge *et al.*, "Rotation-covariant texture analysis of 4D dual-energy CT as an indicator of local pulmonary perfusion," in *Proc. IEEE 10th Int. Symp. Biomed. Imags (ISBI)*, Apr. 2013, pp. 149–152.

[40] O. A. Jiménez–del–Toro, P. Cirujeda, Y. D. Cid, and H. Müller, "RadLex terms and local texture features for multimodal medical case retrieval," in *Multimodal Retrieval in the Medical Domain* (Lecture Notes in Computer Science), vol. 9059. Heidelberg, Germany: Springer, 2015.

[41] P. Cirujeda *et al*, "3D Riesz-wavelet based Covariance descriptors for texture classification of lung nodule tissue in CT," in *Proc. 37th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBC)*, Aug. 2015, pp. 7909–7912.

[42] M. Unser and D. Van De Ville, "Higher-order riesz transforms and steerablewavelet frames," in *Proc. 16th IEEE Int. Conf. Image Process. (ICIP)*, Nov. 2009, pp. 3801–3804.

[43] M. Unser and D. Van De Ville, "Wavelet steerability and the higher-order riesz transform," *IEEE Trans. Image Process.*, vol. 19, pp. 636–652, Mar. 2010.

[44] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 9, no. 3, pp. 891–906, Sep. 1991.

[45] N. Chenouard and M. Unser, "3D steerable wavelets and monogenic analysis for bioimaging," in *Proc. IEEE Int. Symp. Biomed. Imag., Nano Macro*, Mar. 2011, pp. 2132–2135.

[46] L. Paulhac, P. Makris, and J.-Y. Ramel, "A solid texture database for segmentation and classification experiments," in *Proc. 4th Int. Conf. Comput. Vis. Theory Appl. (VISAPP)*, Feb. 2009, pp. 135–141.

[47] J. Olivier and L. Paulhac, *3D Ultrasound Image Segmentation: Interactive Texture-Based Approaches*. Rijeka, Croatia: INTECH, 2011.

[48] L. Paulhac, J.-Y. Ramel, and P. Makris, "A combined topological and statistical approach for interactive segmentation of 3D images," *Mach. Vis. Appl.*, vol. 24, no. 6, pp. 1239–1253, Apr. 2013.

[49] L. Paulhac, P. Makris, J.-M. Gregoire, and J.-Y. Ramel, "Human understandable features for segmentation of solid texture," in *Proc. 5th Int. Symp. Adv. Vis. Comput. (ISVC)*, 2009, pp. 379–390.

[50] L. Paulhac, "Outils et méthodes d'analyse d'images 3D texturées: Application à la segmentation des images échographiques," Ph.D. dissertation, François Rabelais Univ., Tours, France, Nov. 2009.

[51] L. Paulhac, P. Makris, J.-Y. Ramel, and J.-M. Gregoire, "A framework of perceptual features for the characterisation of 3D textured images," *Signal, Image Video Process.*, vol. 9, no. 2, pp. 305–329, 2013.

[52] M. T. Suzuki, Y. Yaginuma, and H. Kodama, "A statistical method for selecting pattern descriptors of textured 3D models," in *Proc. World Congr. Eng. Comput. Sci.*, vol. 1. Oct. 2010, pp. 1–7.

[53] M. T. Suzuki, T. Shibata, Y. Yaginuma, and H. Kodama, "Extended 3D HLAC pattern features for solid textures," in *Proc. IEEE Int. Conf. Signal Image Process. Appl. (ICSIPA)*, Nov. 2011, pp. 541–546.

[54] M. T. Suzuki, "A fractal-based keypoint computation method for solid textures," in *Proc. Int. Conf. Inf. Retr. Knowl. Manage. (CAMP)*, Mar. 2012, pp. 312–316.

[55] R. Maani, S. Kalra, and Y.-H. Yang, "Robust volumetric texture classification of magnetic resonance images of the brain using local frequency descriptor," *IEEE Trans. Image Process.*, vol. 23, no. 10, pp. 4625–4636, Oct. 2014.

[56] L. Paulhac, P. Makris, and J.-Y. Ramel, "Comparison between 2D and 3D local binary pattern methods for characterisation of three-dimensional textures," in *Proc. 5th Int. Conf. Image Anal. Recognit. (ICIAR)*, Jun. 2008, pp. 670–679.

[57] C. C. Reyes-Aldasoro and A. Bhalerao, "Volumetric texture segmentation by discriminant feature selection and multiresolution classification," *IEEE Trans. Med. Imag.*, vol. 26, no. 1, pp. 1–14, Jan. 2007.

[58] C. C. Reyes-Aldasoro and A. Bhalerao, "Volumetric texture description and discriminant feature selection for MRI," in *Proc. Biennial Int. Conf. Inf. Process. Med. Imag.*, vol. 2732. Jul. 2003, pp. 282–293.

[59] R. Maani, S. Kalra, and Y.-H. Yang, "Rotation invariant local frequency descriptors for texture classification," *IEEE Trans. Image Process.*, vol. 22, no. 6, pp. 2409–2419, Jun. 2013.

[60] F. Wagner, A. Gryanik, R. Schulz-Wendtland, P. A. Fasching, and T. Wittenberg, "3D characterization of texture: Evaluation for the potential application in mammographic mass diagnosis," *Biomed. Eng./Biomed. Tech.*, vol. 57, no. SI-1, pp. 490–493, 2012.

[61] J. P. Ward, P. Pad, and M. Unser, "Optimal isotropic wavelets for localized tight frame representations," *IEEE Signal Process. Lett.*, vol. 22, no. 11, pp. 1918–1921, Nov. 2015.

[62] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995.

[63] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, 2011.

[64] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, 2009.

Yashin Dicente Cid received the B.Sc. degree in mathematics from the University of Barcelona, Spain, in 2008, and the M.Sc. degree in computer vision and artificial intelligence from the Autonomous University of Barcelona, Spain, in 2012. He started his Ph.D. in medical imaging at University of Geneva, Switzerland, in 2014, and he joined the MedGIFT Group lead by H. Müller with the University of Applied Sciences Western Switzerland (HES-SO), Sierre. His Ph.D. focuses on the analysis of 4–D CT images, and it is partially founded by the Swiss National Foundation. His interests include texture analysis and graph-based comprehension of N–dimensional images.

Henning Müller received the master's degree in medical informatics from Heidelberg University, Germany, in 1997, the Ph.D. degree in multimedia information retrieval from Geneva University, Switzerland, in 2002, and the Habilitation degree in medical informatics from the University and University hospitals of Geneva, Switzerland, in 2008. He was named a Professor in radiology in 2014. He was also with Daimler Benz Research and Technology North America, Portland, OR, USA, and with Monash University, Melbourne, Australia. Since 2007, he has been a Professor in computer science with the University of Applied Sciences Western Switzerland (HES-SO), Sierre, Switzerland. He has lead the ImageCLEF benchmark on multilingual and multimodal information retrieval for ten years. He has published over 400 scientific articles on visual information analysis and retrieval. He has participated in several EU projects and has initiated several national projects. From 2015 to 2016, he was with the Martinos Center for Biomedical Imaging, Harvard Medical School, and the Massachusetts General Hospital, Boston, MA, USA.

Alexandra Platon was born in Iasi, Romania, in 1966. She currently is an attending Radiologist with the Emergency Radiology Unit, Geneva University Hospital, Switzerland. Her area of research is emergency radiology (trauma and non-trauma).

Pierre-Alexandre Poletti was born in 1964. He completed his medical studies with the Faculty of Medicine, University of Geneva. After a board in medical radiology at the University Hospital of Geneva, he received a fellowship in emergency radiology with the University of Maryland Medical System, Baltimore. He is currently responsible for the emergency radiology Unit, Geneva University Hospital (HUG). He is also an Expert for the Radiation Safety Committee of HUG and the Advisory Committee of atomic-biological-chemical protection coordinates of the Geneva State Council. His areas of research include the development and improvement of emergency medical imaging, especially using CT-imaging.

Adrien Depeursinge received the B.Sc. and M.Sc. degrees in electrical engineering from the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in 2003 and 2005, respectively, with a specialization in signal and image processing, and the Ph.D. degree from the University Hospitals of Geneva in 2010. His Ph.D. thesis was on medical image analysis with a focus on texture analysis and content-based image retrieval. He then spent two years as a Post-Doctoral Fellow with the Department of Radiology, School of Medicine, Stanford University. He has currently a joint position as a Professor of Computer Science with the Institute of Information Systems, University of Applied Sciences Western Switzerland (HES-SO) and as a Senior Research Scientist with the Biomedical Imaging Group, EPFL.