Int J Comput Vis
DOI 10.1007/s11263-013-0634-z

Rotation-Invariant HOG Descriptors Using Fourier Analysis in Polar and Spherical Coordinates

Kun Liu · Henrik Skibbe · Thorsten Schmidt · Thomas Blein · Klaus Palme · Thomas Brox · Olaf Ronneberger

Received: 30 September 2012 / Accepted: 21 May 2013

Abstract The histogram of oriented gradients (HOG) is widely used for image description and proves to be very effective. In many vision problems, rotation-invariant analysis is necessary or preferred. Popular solutions are mainly based on pose normalization or learning, neglecting some intrinsic properties of rotations. This paper presents a method to build rotation-invariant HOG descriptors using Fourier analysis in polar/spherical coordinates, which are closely related to the irreducible representation of the 2D/3D rotation groups. This is achieved by considering a gradient histogram as a continuous angular signal which can be well represented by the Fourier basis (2D) or spherical harmonics (3D). As rotation-invariance is established in an analytical way, we can avoid discretization artifacts and create a continuous mapping from the image to the feature space. In the experiments, we first show that our method outperforms the state-of-the-art in a public dataset for a car detection task in aerial images. We further use the Princeton Shape Benchmark and the SHREC 2009 Generic Shape Benchmark to demonstrate the high performance of our method for similarity measures of 3D shapes. Finally, we show an application on microscopic volumetric data.

Keywords Rotation-invariance · Image descriptor · Fourier analysis · Spherical harmonics · Histogram of oriented gradients · Feature design · Volumetric data

1 Introduction

A good image descriptor should be able to capture substantial image patterns and be robust to object deformation or other common transformations. Gradient histogram based features, like HOG (Histogram-of-Oriented-Gradients, Dalal and Triggs 2005), are widely used for 2D image description. They prove to be very robust, and work as a key component of state-of-the-art object recognition frameworks (*e.g.,* Felzenszwalb et al. 2010; Bourdev and Malik 2009). The HOG descriptor employs a histogram binning on the gradient orientation and a spatial aggregation with soft binning. The spatial aggregation cancels out fine details of spatial place-

All authors are part of the BIOSS Centre for Biological Signalling Studies, University of Freiburg.

K. Liu (✉) · H. Skibbe · T. Schmidt · T. Brox · O. Ronneberger
Department of Computer Science, University of Freiburg,
79110 Freiburg, Germany
e-mail: liu@cs.uni-freiburg.de
URL: http://lmb.informatik.uni-freiburg.de/

H. Skibbe
e-mail: skibbe-h@sys.i.kyoto-u.ac.jp

T. Schmidt
e-mail: tschmidt@cs.uni-freiburg.de

T. Brox
e-mail: brox@cs.uni-freiburg.de

O. Ronneberger
e-mail: ronneber@cs.uni-freiburg.de

Present address:
H. Skibbe
Integrated Systems Biology Lab., Department of Systems Science,
Kyoto University, Kyoto 611-0011, Japan

T. Blein · K. Palme
Institute of Biology II (Botany), University of Freiburg,
79104 Freiburg, Germany
e-mail: thomas.blein@versailles.inra.fr

K. Palme
e-mail: klaus.palme@biologie.uni-freiburg.de

Present address:
T. Blein
Institut Jean-Pierre Bourgin, INRA Centre de Versailles-Grignon,
78026 Versailles, France

Published online: 07 June 2013

ment and so allows local deformations when comparing two gradient patterns. Importantly, the gradient orientations are preserved, because they have already been encoded in the histogram bins. [1]

Rotation invariance is useful when objects of the same class can appear in different poses (*e.g.,* the applications in Ronneberger et al. 2002; Flitton et al. 2010; Villamizar et al. 2010; Vedaldi et al. 2011; Ronneberger et al. 2012). HOG features clearly are not rotation-invariant as the orientations in histograms are defined according to a fixed coordinate system. Further, a HOG descriptor usually samples HOG cells on grids to describe an object, and this sampling is not rotation-invariant either.

Common approaches for rotation-invariance in computer vision are based on either pose normalization (*e.g.,* SIFT achieves the invariance on detected interest points by aligning a local coordinate system to the dominant gradient direction (Lowe 2004)) or learning (*e.g.*, Random Ferns (Özuysal et al. 2010), structured SVM (Vedaldi et al. 2011)). The reliability of the orientation assignment is always a concern for pose normalization (Gauglitz 2011), and it becomes even more critical in 3D (Allaire et al. 2008). The learning based methods usually sample objects under artificial rotations and require a highly nonlinear classifier, since the non-invariant features usually change in a complex manner under rotations. On 3D data, simply sampling all possible rotations becomes unattractive. While sampling one object under 2D rotations in steps of $\alpha = \frac{\pi}{18} = 10^\circ$ leads to 36 samples, it leads to approximately 15000 ($= \frac{4\pi}{\alpha^2} \cdot \frac{2\pi}{\alpha}$) samples for 3D rotations, as three angles (*e.g.*, the Euler angles) are required to determine a 3D pose.

In contrast to these methodologies, we base our method on the Fourier analysis in polar and spherical coordinates. A common method to extract rotation-invariant features from Fourier analysis is to use the frequency magnitude in the spectrum of the images. In general this is not a very effective image feature because of the loss of phase information and the global dependency of those frequency terms. In this paper, it will be discussed that Fourier analysis is the natural tool for analyzing rotations in polar/spherical coordinates. It is not only a transform to the frequency domain, but also deeply related to the the irreducible representations and the eigenfunction problem of 2D/3D rotations (Arsenault and Sheng 1986; Lenz 1990). It is also possible to retain more information than the frequency magnitudes in the spectrum. When used in a proper way, rotation-invariant descriptors from Fourier analysis can outperform other techniques for its continuous image-to-feature mapping. As a major improvement of this general methodology, we incorporate the successful concept of HOG into the feature construction.

In this paper, we concentrate on the description task, which can be understood as a procedure to generate feature vectors for objects (or image patches), so that they can be compared with each other using a standard distance measure or be processed by a standard classifier. The rotation invariance means that the descriptor only consists of rotation-invariant features. They are independent of the orientation of the described objects, and hence no further steering is required for their comparison.

Contributions First, we propose a method for building 2D/3D rotation-invariant descriptors from HOG features. We extend the analytical general rotation-invariance from Fourier analysis in polar/spherical coordinates by combining it with one of the most successful image features. This is done by treating the HOG cells as continuous functions defined on circles (spheres), and using the Fourier basis (Spherical harmonics) to represent them (Fig.1).

Second, this paper provides a compact survey of Fourier analysis in polar and spherical coordinates together with a theoretical analysis. We focus on why those techniques are useful and how they can contribute when rotation-invariance is desired.

This paper covers rotation invariance for both 2D and 3D. The 2D part is more intuitive in the explanation and visualization and performs favorably compared with other state-of-the-art methods. The 3D part is more important, because it may have more applications in volumetric data, where more rotation-invariance related problems exist and fewer comparable alternatives are known.

The rest of the paper is organized as follows: First we review some related work for rotation-invariant image analysis. The mathematical background about the rotations and Fourier analysis in polar coordinates is given in Sect.3 and Sect.4. Section5 introduces the Fourier representation of HOG which connects HOG with the Fourier analysis framework. Section6 presents how to create rotation-invariant descriptors from the obtained *Fourier HOG* field. Section7 extends the analysis method and the proposed descriptor into the 3D space. In the experimental part we first demonstrate the rotation invariance and the robustness to noise with toy examples, then we show that our method outperforms the state-of-the-art methods on a public dataset for car detection in aerial images. More experiments for 3D are carried out on the 3D shape retrieval benchmarks, on which we achieve high scores. Finally an application on 3D confocal microscopic images is presented.

2 Related Work

SIFT (Scale Invariant Feature Transform, Lowe (2004)) aligns a local coordinate system to the dominant gradient

[1] In this paper, a quantity that describes certain image content is generally called a *feature*; a single gradient histogram computed in a local patch is referred to as a *HOG cell*; an assembled feature vector that describes a region of multiple cells is referred to as a *HOG descriptor*.

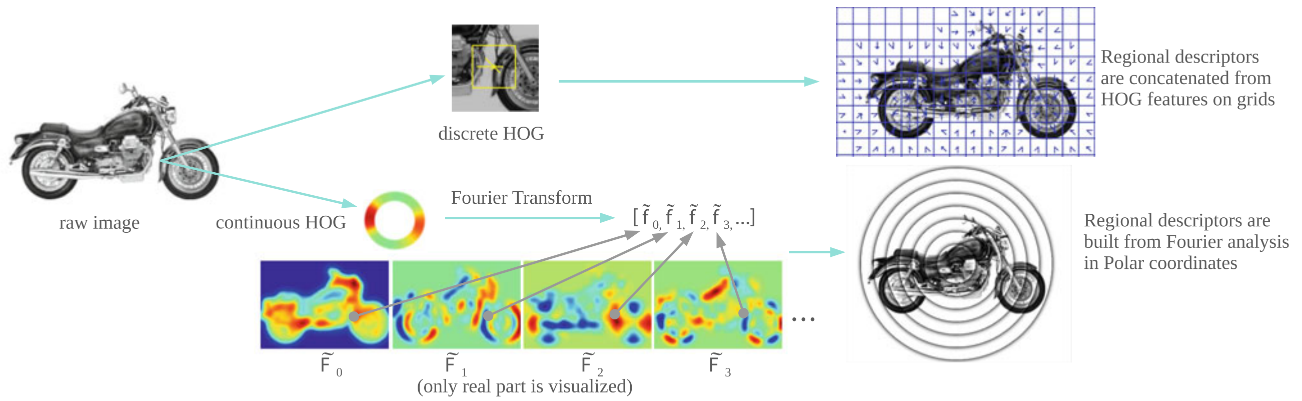

raw image
discrete HOG
Regional descriptors are concatenated from HOG features on grids
continuous HOG
Fourier Transform
$[\tilde{f}_0, \tilde{f}_1, \tilde{f}_2, \tilde{f}_3, ...]$
Regional descriptors are built from Fourier analysis in Polar coordinates
$\tilde{F}_0$
$\tilde{F}_1$
$\tilde{F}_2$
$\tilde{F}_3$
...
(only real part is visualized)

Fig. 1 The standard HOG descriptor and the proposed method. *Top* Standard HOG descriptor computes discrete HOG features on multiple cells and concatenates them into a regional descriptor. *Bottom* The proposed method treats the HOG cells as continuous functions and use the Fourier basis in polar coordinates to represent them (Sect.5), then they can be easily embedded into a Fourier analysis framework (Sect.4) to build a rotation-invariant descriptor (Sect.6)

direction at each detected interest point. This pose normalization relies on the assumption that such a dominant gradient orientation is available. It is a well-known fact that this mechanism does not work well for arbitrary positions or dense feature computation. For instance, the orientation ambiguity is a main source of error in dense image alignment using SIFT features (Lin et al. 2012). Most 2D object recognition approaches skip this step and use the non-invariant dense HOG features with a sliding window classifier (Felzenszwalb et al. 2010).

Pose normalization is particularly unattractive in 3D space as the dimensionality of rotation space grows from one to three. The complexity of 3D rotations is often underestimated. A 3D rotation needs 3 angles to define it, *e.g.*, to change the pose of a globe model, one needs two angles to define the new orientation of the north pole, and a third angle to rotate the globe model around its north-south axis. Many researchers have recently reported their work on extending SIFT or HOG descriptors to 3D data (Kläser et al. 2008; Allaire et al. 2008; Scherer et al. 2010; Flitton et al. 2010). They are all based on pose normalization, except for some applications that do not require rotation-invariance (Kläser et al. 2008). A recent work Knopp et al. (2010a,b) uses SIFT-like features on interest points and a voting framework for the classification of 3D shapes. We will see that these voting methods are suboptimal in comparison with Fourier based methods.

Another popular solution is to solve the rotation problem in a learning framework, where rotation invariance is learned from rotated training samples. Özuysal et al. (2010) use random pixel-difference features for a key point matching task based on sampling the training data under multiple rotations. Vedaldi et al. (2011) use structured learning. Other works on learning rotation-invariant features are Kavukcuoglu et al. (2009); Schmidt and Roth (2012). A big problem of the learning-based methods is the intermingling of rotation invariance with other complex objectives, such as the invariance to some intra-class variations. This makes the overall learning problem much harder. A recent feature learning work that factorize rotation invariance from the remaining learning objectives (Memisevic and Hinton 2010) turns out to generate features similar to the Fourier based features we advocate in this paper (2D functions which are separable in polar coordinates and have the Fourier basis in their angular part). Moreover, the complexity of the learning task grows fast when turning from 2D to 3D rotations.

Analytically deduced rotation-invariance does not depend on pose normalization or sufficient sampling. A fundamental method to compute such invariant features is Group Integration (Burkhardt and Siggelkow 2001; Ronneberger et al. 2002, 2007). Haasdonk and Burkhardt (2007) apply this method to build rotation-invariant kernels. Schmidt and Roth (2012) also use the group integration idea to build the objective function for feature learning. Closely related to the group integration, Fourier analysis in polar and spherical coordinates (Sheng and Arsenault 1986; Wang et al. 2009) provides even more powerful tools for building rotation-invariant image features. Fourier analysis in polar coordinates is a common technique used for image registration (Wolberg and Zokai 2000). It is also the theoretical foundation of steerable filters (Freeman and Adelson 1991).

The advantage of a polar representation has been partially explored in some recent work. Schmidt and Roth (2012); Ahonen et al. (2009) use the permutation of features on polar grids and the Discrete Fourier Transform to achieve rotation-invariance. For HOG-like features, Takacs et al. (2010) propose a rotation-invariant gradient binning, which takes the tangent directions in polar coordinates as the reference directions and then generates rotation-invariant descriptors by averaging on multiple concentric spatial bins. A similar configuration is used in Bendale et al. (2010). However, this gradient binning method is not translation-invariant (the binning is dependent on the relative position to a selected center), and does not generalize well to 3D (a single reference direction is not enough in 3D space). Since we only care about the rotation invariance of the final descriptors, it is not necessary to enforce any invariance on the local feature level, as we will show in Sect.5.

To generalize the Fourier based rotation-invariance from 2D to 3D, the 1D Fourier basis on the unit circle (circular harmonics) just needs to be replaced by the spherical basis on the unit sphere (spherical harmonics). Spherical harmonics are popular in 3D shape description for their related rotation-invariance (Kazhdan 2003; Makadia and Daniilidis 2010). Some important work for introducing the *spherical tensor algebra* in image analysis was done by Reisert and Burkhardt (2008, 2009); Skibbe (2012) further explores the usage of tensor derivatives for fast computation of rotation-invariant descriptions. Besides the descriptors built on intensity values, image gradient and structure tensors have also been investigated in similar frameworks (Fehr 2010; Skibbe et al. 2009). In this paper, we apply the *spherical tensor algebra* to the field of densely computed 3D HOG features. A preliminary version of this paper, which exclusively focuses on the 3D case, has been published in Liu et al. (2011).

What distinguishes this paper from previous works that exploited Fourier analysis to achieve rotation invariance is that we embed the HOG-like features into the systematic Fourier analysis framework. This largely improves the performance of the descriptor. Compared with other rotation-invariant HOG descriptors, our method has a theoretical advantage, namely that the design at the local feature level (i.e., the representation of a single HOG cell) and the descriptor level are based on the same systematic analysis method. Moreover, compared with many other approaches for rotation-invariance, the proposed method avoids quantization artifacts in the gradient binning or pose sampling. Computing the features and descriptors in a fixed coordinate system in a continuous manner (in contrast to the locally estimated coordinates from pose normalization) guarantees that the final descriptor is derived from the image by a smooth continuous mapping. Clearly there is no such guarantee in pose normalized descriptors. This can cause a big difference in the difficulty of succeeding classification tasks, and hence has a big impact on the performance.

3 Rotations in Image Analysis

The purpose of this section is to provide the basic concept of rotations and rotation-invariance, and to introduce the basic formulations for the following discussions.

We write vectors $\mathbf{v} \in \mathbb{C}^n$ and matrices $\mathbf{M} \in \mathbb{C}^{m\times n}$ in bold letters and denote their components by v_i and $M_{i,j}$. We denote the complex conjugate as $\overline{\mathbf{M}}$, the transpose as $\mathbf{M}^\top$, and the conjugate transpose as $\mathbf{M}^\dagger = \overline{\mathbf{M}}^\top$. We use parentheses ( ) to introduce function parameters and use square brackets [ ] to signify the order of operations. The convolution between two functions is $[a * b](\mathbf{x}) = \int_\Omega a(\mathbf{x}-\mathbf{x}')b(\mathbf{x}')d\mathbf{x}'$. The projection of a function a onto another function b, i.e., the inner product of the two functions, is $\langle a, b\rangle = \int_\Omega a(\mathbf{x})\overline{b}(\mathbf{x})d\mathbf{x}$.

The rotations in 2D and 3D Euclidean spaces are characterized by special orthogonal groups SO(2) and SO(3). Elements of these groups can be represented by orthogonal matrices with determinant 1. For instance, the matrix representation for a 2D rotation of angle α (counterclockwise) is $\begin{bmatrix} \cos(\alpha) & -\sin(\alpha) \\ \sin(\alpha) & \cos(\alpha) \end{bmatrix}$.

Rotations of scalar-valued functions A scalar value is a simple quantity that is not changed by rotations or translations, in contrast to vectors and tensors. It can be either real or complex. Given an image $I : \mathbb{R}^2 \to \mathbb{R}; \mathbf{x} \mapsto I(\mathbf{x})$, which is a scalar-valued function, and a rotation $\mathfrak{g} \in \mathrm{SO}(2)$, we first define a coordinate transform $\mathbf{T}_\mathfrak{g} : \mathbb{R}^2 \to \mathbb{R}^2$

$$\mathbf{T}_\mathfrak{g}(\mathbf{x}) := \mathbf{R}_\mathfrak{g}^{-1}\mathbf{x}, \tag{1}$$

where $\mathbf{R}_\mathfrak{g}$ is the matrix representation for the rotation $\mathfrak{g}$. With this coordinate transform we can write the rotated image I_{rot} as

$$I_{\mathrm{rot}}(\mathbf{x}) := I(\mathbf{R}_\mathfrak{g}^{-1}\mathbf{x}) = I(\mathbf{T}_\mathfrak{g}(\mathbf{x})) = [I \circ \mathbf{T}_\mathfrak{g}](\mathbf{x}), \tag{2}$$

where $\circ$ indicates function composition. This means that the rotation $\mathfrak{g}$ on a scalar-valued function is executed by the coordinate transform $\mathbf{T}_\mathfrak{g}$, which maps the original pixel at $\mathbf{R}_\mathfrak{g}^{-1}\mathbf{x}$ to the new position $\mathbf{x}$, or equivalently, from $\mathbf{x}$ to $\mathbf{R}_\mathfrak{g}\mathbf{x}$.

Rotations of vector/tensor-valued functions Higher order quantities (*e.g.*, gradients, structure tensors (Förstner and Gülch 1987), HOG features) are very useful in image analysis, and they have different rotation behaviors compared with the scalar-valued (i.e., the rank-0 tensor) functions.

For instance, we consider a rotation acting on an image gradient field $\mathbf{D} : \mathbb{R}^2 \to \mathbb{R}^2; \mathbf{x} \mapsto \mathbf{D}(\mathbf{x})$, which is a rank-1 tensor field. The rotation cannot be simply computed as $\left[\mathbf{D} \circ \mathbf{T}_\mathfrak{g}\right](\mathbf{x}) = \mathbf{D}(\mathbf{R}_\mathfrak{g}^{-1}\mathbf{x})$. To obtain the correct gradient orientations, this vector field has to transform as:

$$\mathbf{D}_{\mathrm{rot}}(\mathbf{x}) := \mathbf{R}_\mathfrak{g}\,\mathbf{D}(\mathbf{R}_\mathfrak{g}^{-1}\mathbf{x}) = \mathbf{R}_\mathfrak{g}\left[\mathbf{D} \circ \mathbf{T}_\mathfrak{g}\right](\mathbf{x}). \tag{3}$$

See the example in Fig.2 for an illustration. To be more general, *for a rotation* $\mathfrak{g}$ *acting on a tensor field, both the field and the tensor values have to rotate.* The field rotates by a coordinate transform $\mathbf{T}_\mathfrak{g}$, while the tensor values have to rotate according to their definitions (*e.g.*, gradients transform with $\mathbf{R}_\mathfrak{g}$).

In the rest of the paper, for an arbitrary function F, *we use* $\mathfrak{g}F$ *to represent the obtained function after applying the rotation* $\mathfrak{g}$ *on* F. For example, $\mathfrak{g}\mathbf{D} = \mathbf{D}_{\mathrm{rot}} = \mathbf{R}_\mathfrak{g}[\mathbf{D} \circ \mathbf{T}_\mathfrak{g}]$. Distinguishing the proper rotation $\mathfrak{g}$, which keeps the original

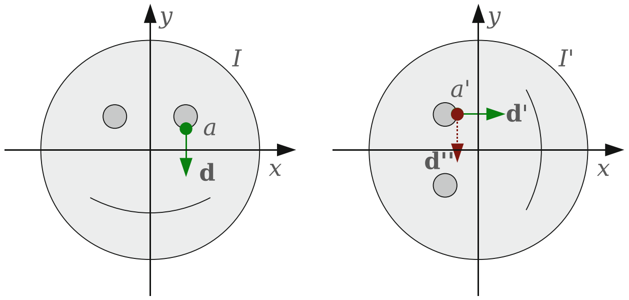


Fig. 2 Rotation behavior of image and gradient. For the 90° rotation $\mathfrak{g}$ in the figure, its rotation matrix is $\mathbf{R}_\mathfrak{g} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$. Assume $a = I(3, 2)$, and the gradient $\mathbf{d} = \mathbf{D}(3, 2)$. It is easy to verify that after the rotation, the pixel value $a' = I'(-2, 3)$ is a copy of $a = I(3, 2) = I(\mathbf{R}_\mathfrak{g}^{-1}[-2, 3]^\top)$. The gradient has a more complicated transform: $\mathbf{d}''$ which is a copy from $\mathbf{d}$ is clearly not the correct gradient. The correct one is $\mathbf{d}' = \mathbf{R}_\mathfrak{g}\mathbf{d} = \mathbf{R}_\mathfrak{g}\mathbf{D}(3, 2) = \mathbf{R}_\mathfrak{g}\mathbf{D}(\mathbf{R}_\mathfrak{g}^{-1}[-2, 3]^\top)$

meaning of the rotated function, from the naive coordinate rotation $\mathbf{T}_\mathfrak{g}$ is essential in our analysis.

Rotations and rotation-invariance of image features We can generalize the above formulations to image features. The feature extraction, for either an intermediate result or the final descriptor, can be abstracted as a functional $h : I \mapsto f$, which takes an image (or an image patch) I as its input, and produces a quantity f as its output. *To keep the meaning of the image feature through a rotation* $\mathfrak{g}$, *it has to transform as if the rotation is acting on the underlying image*, as

$$\begin{aligned} f &:= h(I) \\ \mathfrak{g}f &:= h(\mathfrak{g}I) = h(I \circ \mathbf{T}_\mathfrak{g}). \end{aligned} \tag{4}$$

Rotation-invariance means that the feature remains the "same" when the image rotates. This condition can be formulated as

$$\mathfrak{g}f = f, \quad \text{or equivalently,} \quad h(I \circ \mathbf{T}_\mathfrak{g}) = h(I). \tag{5}$$

However, this formulation is simplified by assuming that the feature is computed at the rotation center (the origin of the reference frame). It is possible to make this assumption when only a single location is to be described (then we can just translate the image before computing the feature).

To be more general, especially for a dense feature computation, we can abstract the computational process as a filter $H : I \mapsto F$, which takes an image I as its input, and produces a function $F : \mathbf{x} \mapsto F(\mathbf{x})$, which comes from computing the image features at all locations. Again, the meaningful rotation has to be defined w.r.t. the processed image as $\mathfrak{g}F := H(\mathfrak{g}I)$. When only regarding the output located at the origin, the invariance is as simple as Eq.(5): $[\mathfrak{g}F](\mathbf{0}) = F(\mathbf{0})$. At other locations, the coordinate transform has to be taken into account, and the rotation-invariance condition should be formulated as

$$\begin{aligned} &[\mathfrak{g}F](\mathbf{x}) = F(\mathbf{T}_\mathfrak{g}(\mathbf{x})) = [F \circ \mathbf{T}_\mathfrak{g}](\mathbf{x}), \\ &\text{or equivalently, } H(I \circ \mathbf{T}_\mathfrak{g}) = H(I) \circ \mathbf{T}_\mathfrak{g}. \end{aligned} \tag{6}$$

Based on Eq.(6), we can point out that *a rotation-invariant feature is a filter output that rotates like a scalar-valued function*, i.e. $\mathfrak{g}F = F \circ \mathbf{T}_\mathfrak{g}$. [2]

4 2D Fourier Analysis in Polar Coordinates

The ideal angular basis A 2D location can be either represented in Cartesian coordinates as $\mathbf{x} = [x, y]^\top \in \mathbb{R}^2$ or in polar coordinates as $[r, \varphi] : r = \|\mathbf{x}\|, \varphi = \Phi(\mathbf{x}) = \text{atan2}(y, x) \in [0, 2\pi)$. For analyzing 2D functions concerning rotations, polar coordinates are ideal, because they separate the angular part, which is involved in rotations, from the radial part, which is naturally invariant to rotations. For the same reason, an ideal basis in polar coordinates should take a separable form $U(r, \varphi) = P(r)\Psi(\varphi)$. While the radial part $P(r)$ can be chosen according to the context, the optimal choice for the angular part is the Fourier basis $\Psi_m(\varphi) = e^{\mathrm{i}m\varphi}$, where m is an integer (Wang et al. 2009). The basis functions $[\Psi_0, \Psi_1, \Psi_2 \ldots]$ form harmonics on a circle, so they are often called *circular harmonics*. To use the Fourier basis functions on $\mathbb{R}^2$, we write $\Psi_m(\mathbf{x})$ as a short notation for $\Psi_m(\Phi(\mathbf{x}))$. In the following, we will explain the optimality of the Fourier basis for rotation analysis.

Simple rotation behavior A function on a circle can be expanded linearly in terms of Fourier basis functions $\Psi_m(\varphi)$ as

$$J(\varphi) = \sum_{m=-\infty}^{\infty} a_m \Psi_m(\varphi) = \sum_{m=-\infty}^{\infty} a_m e^{\mathrm{i}m\varphi}. \tag{7}$$

with coefficients $a_m = \langle J, \Psi_m \rangle = \frac{1}{2\pi} \int_0^{2\pi} J(\varphi) e^{-\mathrm{i}m\varphi} d\varphi$. When the function rotates by an angle α into J', the projection coefficients transform by *a simple multiplication* as

$$J'(\varphi) = J(\varphi - \alpha) = \sum_m a_m e^{\mathrm{i}m(\varphi-\alpha)} = \sum_m (a_m e^{-\mathrm{i}m\alpha}) e^{\mathrm{i}m\varphi}, \tag{8}$$

which is just the "shift property" of the Fourier Transform. The irreducible representations of the 2D rotation group are given by the functions $e^{\mathrm{i}m\alpha}$ $(m \in \mathbb{Z})$, according to the group representation theory (Lenz 1990). The usage of the Fourier basis is optimal in the sense that both the basis and the corresponding expansion coefficients are transformed with the irreducible representations. An example is illustrated in Fig.3.

[2] The property in Eq.(6) has also been referred to as *equivariance* in some works (Reisert and Burkhardt 2008; Vedaldi et al. 2011).

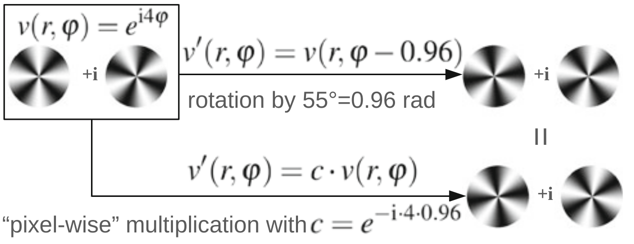

$v(r,\varphi) = e^{i4\varphi}$
+i
$v'(r,\varphi) = v(r,\varphi - 0.96)$
rotation by 55°=0.96 rad
+i
||
$v'(r,\varphi) = c \cdot v(r,\varphi)$
+i
"pixel-wise" multiplication with $c = e^{-i \cdot 4 \cdot 0.96}$

Fig. 3 Rotating a Fourier basis function in polar coordinates. For the illustrated $v(r,\varphi) = e^{i4\varphi}$, $v(r,\varphi-\alpha) = e^{-i4\alpha}v$. That is, for these special functions, the rearrangement (and interpolation) of the pixels caused by a rotation can be substituted by a pixel-wise multiplication with a single complex number

Self-steerable basis and polar tensors A 2D basis function with a Fourier basis in its angular part, i.e., the complex-valued basis function $U(r,\varphi) = P(r)e^{im\varphi}$ ($m \in \mathbb{Z}$, $P(r)$ is an arbitrary radial profile), has the special property

$$\begin{aligned} U(\mathbf{T}_\mathfrak{g}(\mathbf{x})) &= P(r)e^{im(\varphi-\alpha_\mathfrak{g})} = e^{-im\alpha_\mathfrak{g}}U(r,\varphi) \\ \iff U &= e^{im\alpha_\mathfrak{g}}[U \circ \mathbf{T}_\mathfrak{g}]. \end{aligned} \tag{9}$$

Recall that the function $\mathbf{T}_\mathfrak{g}$ is defined as $\mathbf{T}_\mathfrak{g}(\mathbf{x}) := \mathbf{R}_\mathfrak{g}^{-1}\mathbf{x}$ where $\mathbf{R}_\mathfrak{g}$ is the rotation matrix. $\alpha_\mathfrak{g}$ is the rotation angle. This property is called self-steerability (Jacovitti and Neri 2000), as the function itself can be steered to any orientation by a simple multiplicative factor $e^{-im\alpha_\mathfrak{g}}$, or equivalently, the coordinate transform $\mathbf{T}_\mathfrak{g}$ can be compensated by $e^{im\alpha_\mathfrak{g}}$.

Consider a feature from a filtering (convolution) with such a basis function on an image I, $F = H(I) = U * I$. Under a rotation, using Eq.(9) we have

$$\begin{aligned} \mathfrak{g}F &= H(\mathfrak{g}I) = H(I \circ \mathbf{T}_\mathfrak{g}) \\ &= U * [I \circ \mathbf{T}_\mathfrak{g}] = e^{im\alpha_\mathfrak{g}}[U \circ \mathbf{T}_\mathfrak{g}] * [I \circ \mathbf{T}_\mathfrak{g}] \\ &= e^{im\alpha_\mathfrak{g}}[U * I] \circ \mathbf{T}_\mathfrak{g} = e^{im\alpha_\mathfrak{g}}[F \circ \mathbf{T}_\mathfrak{g}]. \end{aligned} \tag{10}$$

Based on this result, we define the *rotation order* of the output function $F = H(I)$ as m, in the sense that its values transform with the irreducible representation $e^{im\alpha_\mathfrak{g}}$, while its coordinates transform with $\mathbf{T}_\mathfrak{g}$. All the complex-valued functions that have the same property as $\mathfrak{g}F = e^{im\alpha_\mathfrak{g}}[F \circ \mathbf{T}_\mathfrak{g}]$ should not be treated as scalar-valued functions since they have different rotation behaviors (recall that $\mathfrak{g}F = F \circ \mathbf{T}_\mathfrak{g}$ if F is a scalar-valued function).

Alternatively, we can call the functions F which have the property $\mathfrak{g}F = e^{im\alpha_\mathfrak{g}}[F \circ \mathbf{T}_\mathfrak{g}]$ as *rank-m polar tensor fields*, because they are the counterparts of the 2D Cartesian tensors in polar coordinates. Schultz et al. (2009) have shown some relations between the tensors in Cartesian coordinates and the Fourier coefficients in polar coordinates[3].

It is of certain interest for us to further investigate the rotation behavior of the basis function U. As a basis function, it is fixed in the whole image analysis process. It is independent of the image rotations, *e.g.*, $\mathfrak{g}[U * I] = U * [\mathfrak{g}I]$ in Eq.(10), thus we have the special rotation behavior for any basis function U w.r.t. *rotations on images*: $\mathfrak{g}U = U$. For those basis functions introduced in Eq.(9), we have

$$\mathfrak{g}U := U = e^{im\alpha_\mathfrak{g}}[U \circ \mathbf{T}_\mathfrak{g}]. \tag{11}$$

This indicates that the basis function U actually has a *rotation order* m, i.e., the basis function is a rank-m polar tensor field. It is therefore rather natural that in Eq.(10) a convolution between the basis function (of rotation order m) and the image (of rotation order 0) produces a filtering result of rotation order m. Later we will present the general additive rules on rotation orders.

The eigenfunction view It is shown above that the Fourier analysis in polar coordinates can factor out the rotations and provide simple rotation behaviors. The Fourier basis in polar coordinates has also been introduced as the eigenfunctions of the Laplace operator (Wang et al. 2009). Here we consider a principle component analysis (PCA) on an image and its rotations. Let the image be sampled on a polar grid as $J(m,n) = I(m \cdot \text{step}_r, n \cdot \text{step}_\varphi)$, where $0 \le m \le M-1$, $0 \le n \le N-1$. For simplicity we only look at the one-dimensional signal $J_m(n)$ at a given radius m, since the computation of PCA is separable in this setting. The rotated samples can be generated with the cyclic shift along n. Considering J_m as a periodical signal (and assume it has a zero mean), the computed covariance matrix $\mathbf{C}$ just consists of the autocorrelation of J_m, which is $C_{n_1,n_2} = \sum_t J_m(t)J_m(t+n_2-n_1)$. $\mathbf{C}$ is symmetric and each column of $\mathbf{C}$ is just a cyclic shift of the other columns. Solving the eigenvalue problem for a matrix like $\mathbf{C}$ yields the eigenvectors

$$\mathbf{C}\mathbf{v} = \lambda\mathbf{v} \Longrightarrow \mathbf{v}_k = [1, e^{-i\frac{2k\pi}{N}}, e^{-i2\frac{2k\pi}{N}}, \ldots, e^{-i(N-1)\frac{2k\pi}{N}}]^\top, \tag{12}$$

where $0 \le k \le N-1$, so $[\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_{N-1}]$ form a discrete Fourier transform matrix (Ponce and Singer 2011; Golub and Loan 1996). The basic principle of PCA tells us that a linear transform with this matrix will diagonalize the covariance matrix, so that the projected data (Fourier coefficients) can be effectively investigated in each dimension separately.

A similar insight can be directly obtained by considering Eq.(9). It simply means that a function $V = P(r)e^{im\varphi}$, is an eigenfunction of the coordinate rotation (Fornasier and Toniolo 2005; Arsenault and Sheng 1986): $V \circ \mathbf{T}_\mathfrak{g} = \lambda V$, i.e., the effect of coordinate rotations on this function is a simple multiplicative factor. When using such functions as basis, the expansion coefficients will transform with these multiplicative factors, since the basis functions stay fixed.

[3] In this paper, we do not rely on this *polar tensor* concept, because we do not need any special mathematical tools for the related analysis of 2D images.

Analytical rotation-invariance For the basis functions or filtering results from the above Fourier analysis (i.e., polar tensor fields), their rotation behaviors can be manipulated by multiplications or convolutions. Consider the functions F_1 of rotation order m_1, and F_2 of rotation order m_2. The individual functions F_1 and F_2 can be either a basis function, an image or a filtering result, as long as they have the rotation behavior in the form as $\mathfrak{g}F = e^{\mathrm{i}m\alpha_\mathfrak{g}}[F \circ \mathbf{T}_\mathfrak{g}]$, $m \in \mathbb{Z}$. We then have the two following equations from coupling the two functions:

$$\mathfrak{g}(F_1 * F_2) = e^{\mathrm{i}(m_1+m_2)\alpha_\mathfrak{g}}[F_1 * F_2] \circ \mathbf{T}_\mathfrak{g}, \qquad (13)$$

$$\mathfrak{g}(F_1 F_2) = e^{\mathrm{i}(m_1+m_2)\alpha_\mathfrak{g}}[F_1 F_2] \circ \mathbf{T}_\mathfrak{g}. \qquad (14)$$

They are easy to verify based on Eq.(10,11).

Obviously the condition for rotation-invariance $\mathfrak{g}F = F \circ \mathbf{T}_\mathfrak{g}$ (Eq.(6)) can be fulfilled by enforcing $m_1 + m_2 = 0$ in the above equations. For example, from Eq.(14) we can derive that, when $\mathfrak{g}F = e^{\mathrm{i}m\alpha_\mathfrak{g}}[F \circ \mathbf{T}_\mathfrak{g}]$,

$$\mathfrak{g}[F\overline{F}] = e^{\mathrm{i}(m-m)\alpha_\mathfrak{g}}[F\overline{F}] \circ \mathbf{T}_\mathfrak{g} = [F\overline{F}] \circ \mathbf{T}_\mathfrak{g}. \qquad (15)$$

This proves the well-known fact that the power spectrum is rotation-invariant.

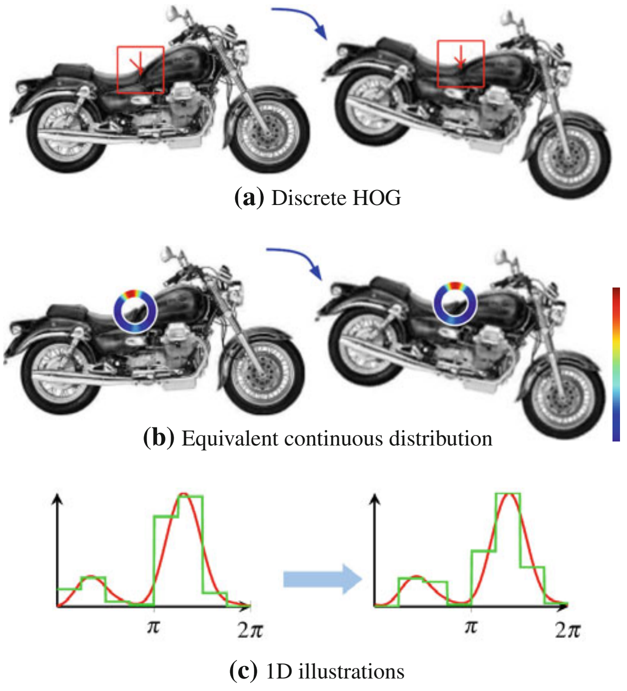
The analysis method discussed in this section allows us to ignore rotation-invariance when we design the feature or descriptor. Establishing rotation invariance can be postponed to the very end when assembling the final descriptor. This allows to achieve rotation invariance with all kinds of features, including HOG.

5 Histogram of Oriented Gradients in Fourier Space

A straightforward way to use the gradient information in polar coordinates is to represent gradient vectors with complex numbers. However, to combine the 2D HOG feature with Fourier analysis, we need more than the representation of gradients.

As proposed by Dalal and Triggs (2005), a HOG cell is always represented by a discrete histogram. When the underlying image rotates by a certain angle, i.e., $\frac{n \cdot 2\pi}{\text{number of bins}}$, the new HOG feature can be obtained by a cyclic permutation. For other rotations, the feature can only be approximated or recomputed.

Fourier representation of continuous HOG A histogram is just a discretized density function. *The original information encoded by a HOG cell is a gradient density function of orientation*, which in the 2D case can be represented by a continuous function $h(\varphi)$. The difference between a discrete and a continuous representation is illustrated in Fig.4. While the discrete representation changes in a complex manner when the image rotates, the continuous representation stays the same up to a cyclic shift. This allows one to use the analytical rotation $h' = h(\varphi - \alpha_\mathfrak{g})$, and quantization errors can be avoided. More importantly, by projecting h onto the Fourier basis, the obtained Fourier representation (illustrated in the middle of Fig.1) transforms with a simple multiplication under rotations, as discussed in Sect.4.



(a) Discrete HOG

(b) Equivalent continuous distribution

(c) 1D illustrations

Fig. 4 Illustration for HOG and its continuous representation for a specific patch, with a rotation of 15°. **(a)** Histogram of oriented gradient. **(b)** A continuous angular signal representing the same distribution. **(c)** 1D illustration of a discrete histogram (*green*) and its corresponding continuous representation (*red*). Note the simple cyclic shift in the continuous representation (Color figure online)

Computing the Fourier representation Normally the HOG features are computed in three steps: gradient orientation binning, spatial aggregation, and normalization (Dalal and Triggs 2005; Felzenszwalb et al. 2010). We explain how these steps should be modified to obtain 2D HOG features in Fourier space that can be used for rotation-invariant image descriptions (according to Step 1 in Fig.5).

In our setting, the orientation quantization for gradients equals creating an orientation distribution function h at each pixel. Let the image gradient computed for one pixel be $\mathbf{d} \in \mathbb{R}^2$, which is oriented to $\Phi(\mathbf{d})$, then the distribution function h for this pixel should be an impulse (Dirac) function with integral $\|\mathbf{d}\|$

$$h(\varphi) := \|\mathbf{d}\|\delta(\varphi - \Phi(\mathbf{d})). \qquad (16)$$

We use the Fourier basis for a continuous and rotation-friendly representation. The Fourier coefficients $\hat{f}_m$ for h read

$$\hat{f}_m = \langle h, e^{\mathrm{i}m\varphi}\rangle = \|\mathbf{d}\|e^{-\mathrm{i}m\Phi(\mathbf{d})} = \|\mathbf{d}\|\overline{\Psi_m(\mathbf{d})}, \qquad (17)$$

where $m \in \mathbb{Z}$, $\Psi_m(\mathbf{d})$ is a short notation for $\Psi_m(\Phi(\mathbf{d}))$.

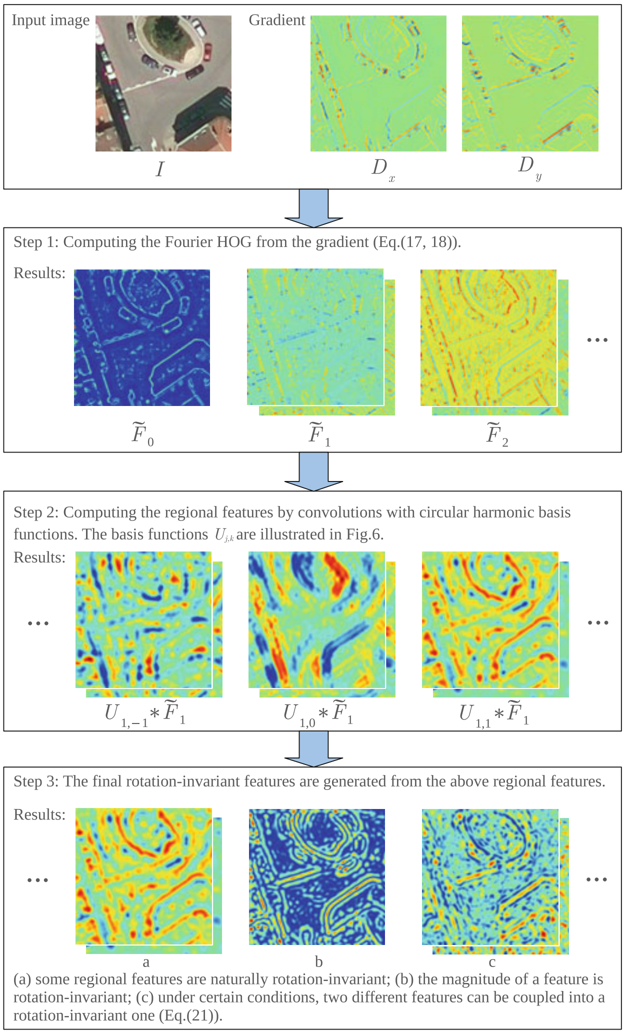

Input image
Gradient
I
D_x
D_y
Step 1: Computing the Fourier HOG from the gradient (Eq.(17, 18)).
Results:
$\widetilde{F}_0$
$\widetilde{F}_1$
$\widetilde{F}_2$
…
Step 2: Computing the regional features by convolutions with circular harmonic basis functions. The basis functions $U_{j,k}$ are illustrated in Fig.6.
Results:
…
$U_{1,-1} * \widetilde{F}_1$
$U_{1,0} * \widetilde{F}_1$
$U_{1,1} * \widetilde{F}_1$
…
Step 3: The final rotation-invariant features are generated from the above regional features.
Results:
…
a
b
c
…
(a) some regional features are naturally rotation-invariant; (b) the magnitude of a feature is rotation-invariant; (c) under certain conditions, two different features can be coupled into a rotation-invariant one (Eq.(21)).

Fig. 5 A flow diagram showing the separate steps of our approach for the 2D case. Complex-valued results are illustrated as two stacked images (real and imaginary parts)

Algorithm 1 2D Fourier HOG Descriptor

```
Input: image $I$
Output: rotation-invariant feature vector field $\mathbf{C} : \mathbf{x} \mapsto \mathbf{C}(\mathbf{x})$

// step 1: compute Fourier HOG (Eq.(17))
$\mathbf{D} = \nabla I$    // compute gradient
for $m = 0 : m_{\max}$ do
    $\hat{F}_m(\mathbf{x}) = \|\mathbf{D}(\mathbf{x})\| e^{-im\Phi(\mathbf{D}(\mathbf{x}))}$    // $\Phi(\mathbf{D}(\mathbf{x}))$ is the gradient direction
    $\widetilde{F}_m = \hat{F}_m / \sqrt{\|\mathbf{D}\|^2 * K}$    // normalization, $K$ is a smoothing convolution kernel
end for
// step 2: compute regional features
$i = 0$
for all $\widetilde{F}_m$ do
    for all basis function $U_{j,k}$ (e.g., the basis in Fig.6) do
        $B_i = U_{j,k} * \widetilde{F}_m$    // regional feature
        $m_i = k - m$    // its rotation order
        $i = i + 1$
    end for
end for
// step 3: generate final rotation-invariant features
$\mathbf{C} = \varnothing$
for all $B_i$ do
    if $m_i = 0$ then
        append $B_i$ to $\mathbf{C}$
    else
        append $\|B_i\|$ to $\mathbf{C}$
    end if
end for
// more features can be generated by coupling (Eq.(21))
for all pairs of features $B_i$ and $B_{i'}$ do
    if $m_i = m_{i'}$ and $m_i \neq 0$ then
        append $\overline{B_i} B_{i'}$ to $\mathbf{C}$
    end if
end for
// split the complex-valued features in $\mathbf{C}$ into real and imaginary part
   to make a real-valued feature vector
```

Even though the projected Fourier coefficients look redundant as a representation for only one gradient, they already encode a "histogram" (density function). It is notable that normally a triangular interpolation is used to soften the orientation quantization for robustness. This can be analytically implemented as a convolution with a 1D triangle kernel, which eventually becomes a simple multiplication in the Fourier space. However, we found this explicit smoothing to be redundant, and it did not improve performance in the experiments. Limiting the maximal frequency degree $|m|$ that is considered for representing the density function is equal to a low-pass filtering in the frequency domain, which already provides the smoothing effect of the classical "soft binning".

The spatial aggregation can be implemented by spatial convolutions on the Fourier coefficients, either with an isotropic triangle or a Gaussian kernel. Finally, also the local contrast normalization can be implemented based on convolutions. A filtering with an isotropic kernel is rotation-invariant, in contrast to some grid-block based operations.

Formally, let $K_1 : \mathbb{R}^2 \to \mathbb{R}$ be the convolution kernel for the spatial aggregation, $K_2 : \mathbb{R}^2 \to \mathbb{R}$ be the convolution kernel for the local normalization (based on gradient energy), $\mathbf{D}$ be the gradient field, and $\hat{F}_m : \mathbb{R}^2 \to \mathbb{C}$ be the densely computed Fourier representation $\hat{f}_m$ from Eq.(17), then we have the *Fourier HOG field* (its degree-m component) as

$$\widetilde{F}_m = \frac{\hat{F}_m * K_1}{\sqrt{\|\mathbf{D}\|^2 * K_2}}. \tag{18}$$

An illustration for these $\widetilde{F}_m$ is shown in Fig.1. The obtained Fourier HOG field $\widetilde{\mathbf{F}} : \mathbb{R}^2 \to \mathbb{C}^M$ (M is the number of coefficients representing a density function) has all the advantages of the normal HOG features. It encodes the local structures in the continuous density functions, hence the gradient orientations are well retained through the spatial smoothing. Because of the desired soft-binning effect in orientation, we only need a few low-frequency Fourier coefficients to encode the useful information (we find $|m| \leq 4$ to be sufficient in our experiments).

Rotation behavior We are most interested in one property of the Fourier HOG field $\hat{F}_m$, that is, how it transforms when the underlying image rotates. To make rotation-invariant features in the end, we need to know the rotation behavior of all intermediate results.

Recall that for a rotation $\mathfrak{g}$, $\mathbf{R}_\mathfrak{g}$ is the rotation matrix, $\mathbf{T}_\mathfrak{g}(\mathbf{x}) := \mathbf{R}_\mathfrak{g}^{-1}\mathbf{x}$, and $\alpha_\mathfrak{g}$ is the rotation angle. From the rotation behavior of a gradient field $\mathfrak{g}\mathbf{D} := \mathbf{R}_\mathfrak{g}\mathbf{D} \circ \mathbf{T}_\mathfrak{g}$ and $\hat{F}_m(\mathbf{x}) = \|\mathbf{D}(\mathbf{x})\|\overline{\Psi_m(\mathbf{D}(\mathbf{x}))}$ (Eq.(17)), we can derive that

$$\begin{aligned}\mathfrak{g}\hat{F}_m &= \left[\|\mathbf{R}_\mathfrak{g}\mathbf{D}\|\overline{\Psi_m(\mathbf{R}_\mathfrak{g}\mathbf{D})}\right] \circ \mathbf{T}_\mathfrak{g} \\ &= \left[\|\mathbf{D}\|e^{-\mathrm{i}m\alpha_\mathfrak{g}}\overline{\Psi_m(\mathbf{D})}\right] \circ \mathbf{T}_\mathfrak{g} = e^{-\mathrm{i}m\alpha_\mathfrak{g}}[\hat{F}_m \circ \mathbf{T}_\mathfrak{g}]. \quad (19)\end{aligned}$$

The conclusion is that $\hat{F}_m$ has a rotation order $-m$. It is the same for $\widetilde{F}_m$, since it is computed from $\hat{F}_m$ just with isotropic filtering.

6 From HOG Cells to HOG Descriptors

A HOG cell can describe the image only locally. To describe a large region with higher level spatial configurations, we need to compute *regional descriptors* (according to Step 2 in Fig.5), on the densely computed HOG cells which in our approach are represented by the Fourier HOG field (see Fig.1 for an illustration). This is equivalent to the SIFT or sliding window techniques, where histograms at neighboring cells are concatenated into a regional descriptor.

Quite a lot of HOG based descriptors have used the polar (or log-polar) spatial binning (*e.g.*, Dalal and Triggs 2005; Takacs et al. 2010; Bendale et al. 2010), which is more rotation-friendly than block-grid based spatial binning. However, without a systematic analysis method and a continuous representation, the power of polar coordinates and Fourier analysis have never been fully explored for making rotation-invariant HOG descriptors.

Computing higher-level features by filtering on the Fourier HOG field As discussed in Sect.4, a basis function in the form $P(r)e^{\mathrm{i}m\varphi}$ has a nice, simple rotation behavior. The Fourier basis is a complete orthogonal basis for the angular part, so we only need to choose a proper radial basis $P(r)$ to build a 2D basis for describing the Fourier HOG field. A natural choice is to sample on the radius, thus the 2D basis functions for computing regional descriptors are

$$U_{j,k}(r,\varphi) = \delta(r - r_j)e^{\mathrm{i}k\varphi} \quad (20)$$

where $j \in \mathbb{N}_0, k \in \mathbb{Z}$. Since the computation of the HOG field includes a smoothed spatial aggregation (like a low-pass filtering), a proper down-sampling in radial direction (according to the scale of the smoothing kernel K_1) can keep all the information.

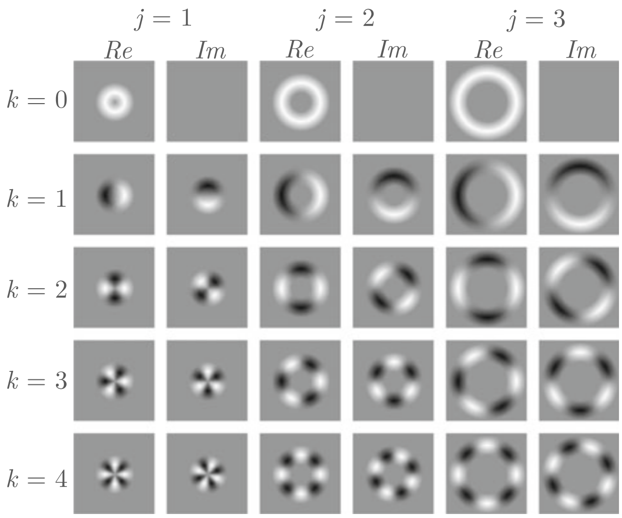


Fig. 6 The basis functions $U_{j,k} = P_j(r)e^{\mathrm{i}k\varphi}$ used for regional descriptions

Computing the convolution between such a basis function $U_{j,k}$ and a component of the Fourier HOG field $\widetilde{F}_m$, generates a feature which describes the configuration of HOG features in the region covered by $U_{j,k}$. According to Eq.(13) the filtering results $U_{j,k} * \widetilde{F}_m$ have the rotation order $k - m$. Using Eq.(14) we can create the rotation-invariant regional descriptors by the following computation

$$\overline{(U_{j_1,k_1} * \widetilde{F}_{m_1})}(U_{j_2,k_2} * \widetilde{F}_{m_2}), \quad \forall k_1 - m_1 = k_2 - m_2. \quad (21)$$

Note $e^{\mathrm{i}n\varphi} = \overline{e^{\mathrm{i}(-n)\varphi}}$, so the above formulation covers all possible rotation-invariant quantities from coupling two of the filtering results. The descriptors created in this way are very effective, as the coupling provides the possibility to create many more invariant features, compared with only taking the magnitude of expansion coefficients. They also keep certain phase information by coupling different features.

Implementation The convolutions of $\hat{F}_m$ with the spatial smoothing kernel K_1 and $U_{j,k}$ can be implemented as one convolution, by combining K_1 into $U_{j,k}$. Therefore we may compose the 2D basis function $U_{j,k}$ from a triangular radial profile and a Fourier basis

$$U_{j,k}(r,\varphi) = \Lambda(r - r_j, \sigma)e^{\mathrm{i}k\varphi}, \quad (22)$$

where Λ is a triangular function of width 2σ defined as $\Lambda(x,\sigma) = \max(\frac{\sigma-|x|}{\sigma}, 0)$. A practical example is visualized in Fig.6, which is the basis used in our experiment. Such basis functions include smoothing both in radial and angular direction. Only positive m are necessary when computing $\hat{F}_m$, because the function h is real-valued. A pseudo code is given in Algorithm 1, according to the flow digram in Fig.5.

Information loss and other limitations The complete HOG information is encoded in $U_{j,k} * \widetilde{F}_m$, since all the previous operations are equivalent to Fourier transform and down-sampling on band-limited signals. We start to lose information when using only the coupling results in Eq.(21) as image descriptors. When only two features from filtering $U_{j,k} * \widetilde{F}_m$ are used to make a rotation-invariant feature, they have to be of the same rotation-order. Similar to the power spectrum which loses all the phase information, the computed rotation-invariant feature in Eq.(21) also loses certain phase information, which is *the relative phase between features of different rotation-orders*. This means, we can only reconstruct the raw features (the filtering results $U_{j,k} * \widetilde{F}_m$) for each rotation-order separately up to an ambiguity of the absolute phase. The raw image pattern cannot be fully reconstructed since there is no phase information for aligning these different parts. However, this does not mean that the signals on different radii can rotate independently, because we can couple two features from different radii into a rotation-invariant feature, as long as they have the same rotation order. To avoid the loss of phase information, the bispectrum (Giannakis 1989), which explores the coupling of three frequencies, could be considered.

Using standard HOG descriptors, the learning process can select the relevant features spatially in a simple linear model (Felzenszwalb et al. 2010) for a given object pose. This simple spatial selectivity is lost as a price for the rotation-invariance, since the regional features are built from convolutions with circular harmonics instead of simple sampling. For the same reason, the visualization of the final features also lacks the intuition of standard HOG.

The normalization in Eq.(18) is a simple one. Some more complicated operations on the histograms (*e.g.*, thresholding) are hard to implement when the histograms are represented in Fourier space. Therefore, if necessary, the desired normalization should be implemented before the Fourier transform.

7 3D Space: Analysis Methods, HOG Features, and Rotation-Invariant Descriptors

In this section, we extend the proposed descriptors to 3D. The 3D descriptors are based on the 3D counterpart of circular harmonics, the so-called spherical harmonics. This makes the definition of 3D features fully analogous to the 2D case. The introduced 3D descriptors are of high value, as existing straightforward extensions of 2D descriptors come with many limitations in practice, *e.g.* 3D SIFT inherently depends on interest point detectors, and pose normalization is even harder in 3D (Flitton et al. 2010).

7.1 Fourier Analysis in Spherical Coordinates

In 3D, to describe a signal on a sphere with 3D rotations, the tools we need are the spherical harmonics (SHs) and the rotation group SO(3). They are widely known from the angular momentum theory for describing the rotational state of physical systems. There are some other references which may be useful for the reader who are not familiar with SHs and the related theories, including Driscoll and Healy (1994); Green (2003), and the physics books Rose (1957); Brink and Satchler (1968).

Spherical harmonics A 3D vector can be either represented in Cartesian coordinates as $\mathbf{x} = [x, y, z]^\top \in \mathbb{R}^3$ or in spherical coordinates as $[r, \theta, \varphi] : r = \|\mathbf{x}\|, \theta = \Theta(\mathbf{x}) = \mathrm{acos}(\frac{z}{\|\mathbf{x}\|}) \in [0, \pi], \varphi = \Phi(\mathbf{x}) = \mathrm{atan2}(y, x) \in [0, 2\pi)$. It is straightforward to extend the coordinate transform $\mathbf{T}_\mathfrak{g}(\mathbf{x}) := \mathbf{R}_\mathfrak{g}^{-1}\mathbf{x}$ to 3D, where $\mathbf{R}_\mathfrak{g} \in \mathbb{R}^{3\times3}$ are the 3D rotation matrices. In analogy to the Fourier basis in polar coordinates, spherical harmonics $Y_m^\ell : S^2 \to \mathbb{C}$ form an orthonormal basis for the angular part in spherical coordinates – the 2-sphere $S^2 = \{\mathbf{x} \in \mathbb{R}^3 : \|\mathbf{x}\| = 1\}$. The SH basis functions can be arranged in a "two-dimensional" structure with two indices ℓ (degree or band) and m (order). The first five degrees are visualized in Fig.7. For the ℓ-th degree, there are $2\ell + 1$ SH basis functions, which are indexed in the range of $-\ell \leq m \leq \ell$. For a more concise notation, we arrange the basis functions of the same degree as a vector $\mathbf{Y}^\ell : S^2 \to \mathbb{C}^{2\ell+1}; [\theta, \varphi] \mapsto [Y_{-\ell}^\ell(\theta, \varphi), Y_{-\ell+1}^\ell(\theta, \varphi), \ldots, Y_\ell^\ell(\theta, \varphi)]^\top$.

Any square-integrable scalar-valued function $J(\theta, \varphi)$ on a sphere, can be expanded into a linear combination of SHs as:

$$J(\theta, \varphi) = \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} \overline{b_m^\ell} Y_m^\ell(\theta, \varphi) = \sum_{\ell=0}^{\infty} \mathbf{b}^{\ell\dagger} \mathbf{Y}^\ell(\theta, \varphi) \tag{23}$$

where $\mathbf{b}^\ell \in \mathbb{C}^{2\ell+1}$ and $(\cdot)^\dagger$ denotes the conjugate transpose.[4] The Schmidt semi-normalized SHs are defined as

$$Y_m^\ell(\theta, \varphi) := \sqrt{\frac{(\ell - m)!}{(\ell + m)!}} P_m^\ell(\cos\theta) e^{im\varphi}, \tag{24}$$

where P_m^ℓ are the associated Legendre polynomials. They can be computed with the given functions in *Matlab* or the *GNU Scientific Library*. These SHs have the normalization relationship $\langle Y_m^\ell, Y_{m'}^{\ell'} \rangle = \int_{S^2} Y_m^\ell \overline{Y_{m'}^{\ell'}} d\mathbf{x} = \frac{4\pi}{2\ell+1} \delta_{\ell,\ell'} \delta_{m,m'}$. Considering the normalization and Eq. 23, the expansion coefficients are computed as $b_m^\ell = \frac{2\ell+1}{4\pi} \langle Y_m^\ell, J \rangle$. We use $\mathbf{Y}^\ell(\mathbf{x})$ as a short notation for $\mathbf{Y}^\ell(\Theta(\mathbf{x}), \Phi(\mathbf{x}))$, which extends the domain of definition to $\mathbb{R}^3$.

The irreducible representations of SO(3) The irreducible representation of 2D rotations, $e^{\mathrm{i}m\alpha_\mathfrak{g}}$, is related to the "shift property" of the Fourier basis functions: $\Psi_m(\mathbf{T}_\mathfrak{g}(\mathbf{x})) = e^{-\mathrm{i}m\alpha_\mathfrak{g}} \Psi_m(\mathbf{x})$, i.e., a coordinate rotation of the basis function is equivalent to a "pixel-wise" multiplication. In analogy to

[4] We purposely define the expansion coefficients with a conjugation, which makes it a standard inner product between the coefficients and SH basis. The same convention is used in Reisert and Burkhardt (2009). The advantage is that this linear expansion can be understood as a coupling between two spherical tensors, which will be explained later.

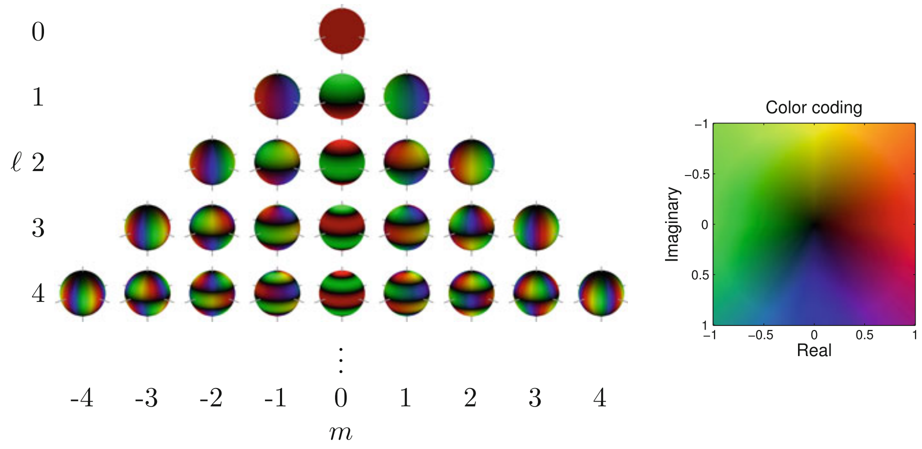


Fig. 7 Visualization of the spherical harmonics in the first 5 frequency degrees

this, in spherical coordinates, the SHs have a similar property

$$\mathbf{Y}^\ell(\mathbf{T}_\mathfrak{g}(\mathbf{x})) = \mathbf{Y}^\ell(\mathbf{R}_\mathfrak{g}^{-1}\mathbf{x}) = \boldsymbol{\mathcal{D}}_\mathfrak{g}^{\ell\,\dagger}\mathbf{Y}^\ell(\mathbf{x})$$
$$\Longleftrightarrow \mathbf{Y}^\ell(\mathbf{x}) = \boldsymbol{\mathcal{D}}_\mathfrak{g}^\ell\mathbf{Y}^\ell(\mathbf{T}_\mathfrak{g}(\mathbf{x})). \tag{25}$$

where $\boldsymbol{\mathcal{D}}_\mathfrak{g}^\ell \in \mathbb{C}^{(2\ell+1)\times(2\ell+1)}$ is an unitary matrix ($\boldsymbol{\mathcal{D}}_\mathfrak{g}^{\ell\,-1} = \boldsymbol{\mathcal{D}}_\mathfrak{g}^{\ell\,\dagger}$), which comes from the irreducible representations of 3D rotation group SO(3). They are the so-called Wigner D-matrices, which are determined by the 3D rotation angles. With $\boldsymbol{\mathcal{D}}_\mathfrak{g}^\ell$, we have a simple multiplicative transform for $\mathbf{Y}^\ell$, in a matrix form. The matrix form also reflects the fact that a single SH basis Y_m^ℓ cannot be investigated individually under rotations.

There is some direct connection between 3D rotations and the 2D case: a 3D rotation $\mathfrak{g}$ around the Z axis by angle α is corresponding to a diagonal Wigner D-matrix, where $\mathcal{D}_{m,m}^\ell = e^{\mathrm{i}m\alpha}$. All Wigner D-matrices are computable given the rotation parameters (3 Euler angles). In this paper, there is no need to compute them since they will finally drop out. Our final rotation-invariant features will be scalar values, i.e., they transform with $\boldsymbol{\mathcal{D}}^0 = 1$. More details about the representation of the 3D rotation group can be found in Brink & Satchler(1968, pp. 13–25) and Rose (1957, pp. 48–73).

Simple rotation behavior To extend the SH expansion to 3D volumes, we can expand a square-integrable scalar-valued 3D function I as

$$I(r,\theta,\varphi) = \sum_\ell \mathbf{b}^\ell(r)^\dagger\mathbf{Y}^\ell(\theta,\varphi), \tag{26}$$

where $\mathbf{b}^\ell : \mathbb{R} \to \mathbb{C}^{2\ell+1}; r \mapsto \mathbf{b}^\ell(r)$ are the SH expansion coefficients of the function on the spherical shell with radius r. Equivalently we can write $I(\mathbf{x}) = \sum_\ell \mathbf{b}^\ell(\|\mathbf{x}\|)^\dagger\mathbf{Y}^\ell(\mathbf{x})$. After a rotation $\mathfrak{g}$, we get

$$[\mathfrak{g}I](\mathbf{x}) = I(\mathbf{T}_\mathfrak{g}(\mathbf{x})) = \sum_\ell \mathbf{b}^\ell(\|\mathbf{x}\|)^\dagger\mathbf{Y}^\ell(\mathbf{T}_\mathfrak{g}(\mathbf{x}))$$
$$= \sum_\ell (\boldsymbol{\mathcal{D}}_\mathfrak{g}^\ell\mathbf{b}^\ell(\|\mathbf{x}\|))^\dagger\mathbf{Y}^\ell(\mathbf{x}). \tag{27}$$

The last conversion uses Eq.(25). The conclusion here is that the SH expansion coefficients for a rotated function just look like $\boldsymbol{\mathcal{D}}_\mathfrak{g}^\ell\mathbf{b}^\ell$.

In analogy to the 2D setting (Eq.(9, 11)), for basis functions $\mathbf{U}^\ell : \mathbb{R}^3 \to \mathbb{C}^{2\ell+1}$ in the form $\mathbf{U}^\ell(r,\theta,\varphi) = P(r)\mathbf{Y}^\ell(\theta,\varphi)$, we can infer the following properties based on Eq.(25),

$$\mathbf{U}^\ell \circ \mathbf{T}_\mathfrak{g} = \boldsymbol{\mathcal{D}}_\mathfrak{g}^{\ell\,\dagger}\mathbf{U}^\ell, \tag{28}$$

$$\mathfrak{g}\mathbf{U}^\ell := \mathbf{U}^\ell = \boldsymbol{\mathcal{D}}_\mathfrak{g}^\ell[\mathbf{U}^\ell \circ \mathbf{T}_\mathfrak{g}]. \tag{29}$$

In analogy to Eq.(10), we can use such a basis for a 3D image description by convolutions. The filtering results $\mathbf{F} : F_m = U_m * I$ have the same rotation behavior as $\mathbf{U}^\ell$,

$$\mathfrak{g}\mathbf{F} = \boldsymbol{\mathcal{D}}_\mathfrak{g}^\ell[\mathbf{F} \circ \mathbf{T}_\mathfrak{g}]. \tag{30}$$

However, this filtering is only valid when applied to scalar-valued functions. It cannot be directly extended to the coupling of two arbitrary functions (as in Eq.(13,14)). In order to deal with higher-order quantities that encode orientations, such as 3D gradients, the approach has to be extended by spherical tensor algebra.

7.2 Spherical Tensor Algebra

Because of the special structure of the SH basis and 3D rotations, it is necessary to introduce the *spherical tensor* concept and the related tensor algebra. Here we recapitulate the most useful computational rules for spherical tensors. We refer to Reisert and Burkhardt (2009) for details and proofs.

A function $\mathbf{F} : \mathbb{R}^3 \to \mathbb{C}^{2\ell+1}$ is called a rank-ℓ *spherical tensor* field, if it transforms under rotations as

$$[\mathfrak{g}\mathbf{F}](\mathbf{x}) = \boldsymbol{\mathcal{D}}_\mathfrak{g}^\ell\mathbf{F}(\mathbf{T}_\mathfrak{g}(\mathbf{x})). \tag{31}$$

The space of all rank-ℓ spherical tensor fields is denoted by $\mathcal{T}_\ell$. The spherical tensors are tensors expressed in spherical coordinates. They are related to the Cartesian tensors, but have simpler rotation behaviors (Rose 1957, pp. 76–77).

According to the definition, the basis $\mathbf{U}^\ell = P(r)\mathbf{Y}^\ell(\theta, \varphi)$ in Eq.(29) and the filtering result in Eq.(30) are both spherical tensor fields.

Spherical tensors have the same element-wise addition rule as Cartesian tensors, but for multiplication, a special operation $\otimes_{(\ell|\ell_1,\ell_2)} : \mathbb{C}^{2\ell_1+1} \times \mathbb{C}^{2\ell_2+1} \rightarrow \mathbb{C}^{2\ell+1}$ is defined to couple two spherical tensors.[5] Let $\mathbf{v} \in \mathbb{C}^{2\ell_1+1}$, $\mathbf{w} \in \mathbb{C}^{2\ell_2+1}$, the coupling operation $\mathbf{v} \otimes_{(\ell|\ell_1,\ell_2)} \mathbf{w} = \mathbf{z} \in \mathbb{C}^{2\ell+1}$ is defined as

$$\begin{aligned} z_m &= [\mathbf{v} \otimes_{(\ell|\ell_1,\ell_2)} \mathbf{w}]_m \\ &= \sum_{\substack{m=m_1+m_2 \\ -\ell_1 \le m_1 \le \ell_1 \\ -\ell_2 \le m_2 \le \ell_2}} C(\ell, m|\ell_1, m_1, \ell_2, m_2) v_{m_1} w_{m_2}, \end{aligned} \tag{32}$$

where $C(\ell, m|\ell_1, m_1, \ell_2, m_2)$ are real-valued coefficients, which depend on those six numbers. They are called Clebsch-Gordan coefficients (Rose 1957, 32–33), which only have non-zero values if $m = m_1+m_2$ and $|\ell_1-\ell_2| \le \ell \le \ell_1+\ell_2$ (a triangle inequality). The defined coupling $\otimes_{(\ell|\ell_1,\ell_2)}$ is therefore only valid if $\ell \ge 0$ and $|\ell_1 - \ell_2| \le \ell \le \ell_1 + \ell_2$. It can be proven that, if $\mathbf{v}$ and $\mathbf{w}$ are both spherical tensors, the coupling result ($\mathbf{z}$) is a spherical tensor (rank-ℓ), too. If $\ell = 0$ (i.e., $\mathbf{z}$ is a scalar value), $\mathbf{v}$ and $\mathbf{w}$ have to be of the same rank due to the triangle inequality.

A normalized variant of this coupling is defined for $\ell \ge 0$ and $|\ell_1 - \ell_2| \le \ell \le \ell_1 + \ell_2$, (Reisert and Burkhardt 2009)

$$\mathbf{v} \ \bullet_{(\ell|\ell_1,\ell_2)} \ \mathbf{w} := \frac{1}{C(\ell, 0|\ell_1, 0, \ell_2, 0)} \mathbf{v} \otimes_{(\ell|\ell_1,\ell_2)} \mathbf{w}. \tag{33}$$

Based on this coupling operation, the *tensorial harmonic expansion* can be introduced, which provides a complete and orthogonal expansion for the angular part of arbitrary spherical tensor fields, that is, for $\mathbf{F} \in \mathcal{T}_\ell$:

$$\mathbf{F}(r, \theta, \varphi) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} \mathbf{a}^{j,k}(r) \otimes_{(\ell|j+k,j)} \mathbf{Y}^j(\theta, \varphi) \tag{34}$$

where $\mathbf{a}^{j,k}(r) \in \mathbb{C}^{2(j+k)+1}$ are the tensorial expansion coefficients of the function on the spherical shell with radius r. They transform as $\mathcal{D}_\mathfrak{g}^{j+k}\mathbf{a}^{j,k}(r)$ under rotations. The index k is due to the fact that, in the tensor coupling defined in Eq.(32), there are multiple possible combinations ℓ_1 and ℓ_2 that result in rank ℓ. Note, if $\ell = 0$, the expansion above degenerates into the scalar expansion in Eq.(26). An efficient way to compute the expansion coefficients $\mathbf{a}^{j,k}$ for a given tensor field is detailed in Appendix.

Finally, for the filtering operation, we need to introduce the convolution based on the tensor product. Let $\mathbf{V} \in \mathcal{T}_{\ell_1}$, $\mathbf{W} \in \mathcal{T}_{\ell_2}$, then the convolution $\tilde{\bullet}_{(\ell|\ell_1,\ell_2)}$, which combines the two tensor fields into a rank-ℓ tensor field ($|\ell_1 - \ell_2| \le \ell \le \ell_1 + \ell_2$), reads

$$[\mathbf{V} \ \tilde{\bullet}_{(\ell|\ell_1,\ell_2)} \mathbf{W}](\mathbf{x}) := \int_{\mathbb{R}^3} \mathbf{V}(\mathbf{x}-\mathbf{x}') \ \bullet_{(\ell|\ell_1,\ell_2)} \mathbf{W}(\mathbf{x}')d\mathbf{x}' \tag{35}$$

Analytical rotation-invariance As in the 2D setting, a scalar (rank-0 tensor) output will be a rotation-invariant output. The manipulation of tensor ranks, which are in correspondence to Eq.(13,14), has already been given in Eq.(32,35). To derive rotation invariant features from spherical tensors, we only need to compute the standard inner-product of two tensors of the same rank. Since the Wigner-D matrices are unitary, their effect will be compensated in the complex inner product. Let $\mathbf{F}_1, \mathbf{F}_2 \in \mathcal{T}_\ell$, then the inner-product $\mathbf{F_1}^\dagger \mathbf{F}_2$ is rotation-invariant. It can be proved as

$$\begin{aligned} \mathfrak{g}[\mathbf{F_1}^\dagger \mathbf{F}_2] &= (\mathfrak{g}\mathbf{F}_1)^\dagger (\mathfrak{g}\mathbf{F}_1) = (\mathcal{D}^\ell_\mathfrak{g}\mathbf{F}_1)^\dagger (\mathcal{D}^\ell_\mathfrak{g}\mathbf{F}_2) \\ &= \mathbf{F_1}^\dagger \mathcal{D}_\mathfrak{g}^{\ell\,\dagger} \mathcal{D}^\ell_\mathfrak{g}\mathbf{F}_2 = \mathbf{F_1}^\dagger \mathbf{F}_2. \end{aligned} \tag{36}$$

The power spectrum on a sphere from the SH expansion is a special case of this formula. In fact, it can be proved that, under certain conditions, the inner-product is just the coupling $\bullet_{(0|\ell,\ell)}$ between two spherical tensors of the same rank, which results in a rank-0 tensor (Reisert and Burkhardt 2009).

7.3 Spherical Harmonic Representation of Continuous HOG

In this section, we explain how to embed the highly descriptive 3D HOG feature into the SH framework.

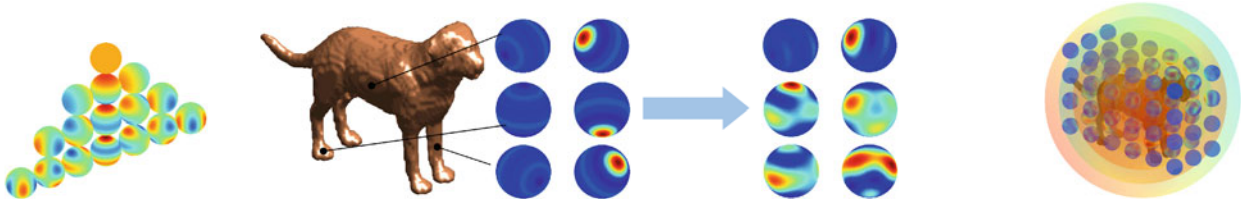
Similar to the 2D case, the concept of HOG is integrated into the SH framework by treating a 3D HOG cell as a continuous distribution defined on the 2-sphere. By using SH coefficients to represent the distribution, rotations can be addressed based on the simple rotation behavior of SH coefficients.

The first step is to create the distribution function h at each voxel. Let a 3D gradient be $\mathbf{d} \in \mathbb{R}^3$ and its spherical coordinate representation be $[\|\mathbf{d}\|, \Theta(\mathbf{d}), \Phi(\mathbf{d})]$. The function which only describes the gradient distribution at one voxel should be a Dirac function $\delta(\theta - \Theta(\mathbf{d}), \varphi - \Phi(\mathbf{d}))$ with integral $\|\mathbf{d}\|$. The SH coefficients, as defined in Eq.(23), are

$$\begin{aligned} \hat{\mathbf{f}}^\ell &= \tfrac{2\ell+1}{4\pi} \langle \mathbf{Y}^\ell, \|\mathbf{d}\|\delta(\theta - \Theta(\mathbf{d}), \varphi - \Phi(\mathbf{d}))\rangle \\ &= \tfrac{2\ell+1}{4\pi} \|\mathbf{d}\| \mathbf{Y}^\ell(\Theta(\mathbf{d}), \Phi(\mathbf{d})) = \tfrac{2\ell+1}{4\pi} \|\mathbf{d}\| \mathbf{Y}^\ell(\mathbf{d}). \end{aligned} \tag{37}$$

In the continuous setting, the "soft binning" for the gradient direction is equivalent to smoothing on the 2-sphere. It can be implemented as multiplication between SH expansion coefficients (Driscoll and Healy 1994; Bülow 2004). Similar to the 2D case, this step can be skipped, since considering

[5] This operator is written as $\circ_\ell$ in Reisert and Burkhardt (2009), since ℓ_1, ℓ_2 can be inferred from the two coupled tensors. In this paper we use the more explicit notation $\otimes_{(\ell|\ell_1,\ell_2)}$.



(a) Spherical harmonics **(b)** Spherical harmonic representation of 3D HOG cells **(c)** Regional descriptor

Fig. 8 3D HOG descriptor. **(a)** *spherical harmonics* as basis functions for representing HOG cells (real part of the first 4 degrees are shown). **(b)** An individual gradient vector is treated as a distribution on a sphere, which is then spatially aggregated into HOG cells (signals on spheres are reconstructed from the SH coefficients, and shown in two views). **(c)** *Regional descriptors* are then computed on the *SH HOG field* by spherical tensor operations

only low-frequency degrees in the SH representation corresponds to low-pass filtering of h.

Including spatial aggregation and normalization, the *SH HOG Field* (its degree-ℓ order-m component) $\widetilde{F}^\ell_m : \mathbb{R}^3 \to \mathbb{C}$ reads

$$\widetilde{F}^\ell_m = \frac{\hat{F}^\ell_m * K_1}{\sqrt{\|\mathbf{D}\|^2 * K_2}}. \tag{38}$$

where $\hat{F}^\ell_m : \mathbb{R}^3 \to \mathbb{C}$ is the densely computed SH representation $\hat{f}^\ell_m$ from Eq.(37), $\mathbf{D}$ is the gradient field, $K_1 : \mathbb{R}^3 \to \mathbb{R}$ is the kernel for the smooth spatial aggregation, $K_2 : \mathbb{R}^3 \to \mathbb{R}$ is the kernel for the local normalization. The process is illustrated in Fig. 8. The signals shown on the spheres are reconstructed from SH expansion in $\ell \leq 5$.

Rotation behavior We can prove that the SH HOG representation in the ℓ^{th} degree is just a rank-ℓ spherical tensor field, so it transforms with the Wigner-D matrices $\mathcal{D}^\ell$ under rotations. Formally, to derive the rotation behavior of $\hat{\mathbf{F}}^\ell : \mathbb{R}^3 \to \mathbb{C}^{2\ell+1}; \mathbf{x} \mapsto [\hat{F}^\ell_{-\ell}(\mathbf{x}), \ldots, \hat{F}^\ell_\ell(\mathbf{x})]^\top$, we need the property $\mathbf{Y}^\ell(\mathbf{x}) = \mathcal{D}^\ell_\mathfrak{g}\mathbf{Y}^\ell(\mathbf{R}_\mathfrak{g}^{-1}\mathbf{x})$ from Eq.(25), and the fact that a gradient field $\mathbf{D}$ transforms as $\mathfrak{g}\mathbf{D} := \mathbf{R}_\mathfrak{g}\mathbf{D} \circ \mathbf{T}_\mathfrak{g}$. Starting from Eq.(37) we can derive

$$\begin{aligned}\mathfrak{g}\hat{\mathbf{F}}^\ell &= [c\|\mathbf{R}_\mathfrak{g}\mathbf{D}\|\mathbf{Y}^\ell(\mathbf{R}_\mathfrak{g}\mathbf{D})] \circ \mathbf{T}_\mathfrak{g} \\ &= [c\|\mathbf{D}\|\mathcal{D}^\ell_\mathfrak{g}\mathbf{Y}^\ell(\mathbf{D})] \circ \mathbf{T}_\mathfrak{g} = \mathcal{D}^\ell_\mathfrak{g}[\hat{\mathbf{F}}^\ell \circ \mathbf{T}_\mathfrak{g}],\end{aligned} \tag{39}$$

where $c = \frac{2\ell+1}{4\pi}$. Hence, $\hat{\mathbf{F}}^\ell$ and the smoothed/normalized $\tilde{\mathbf{F}}^\ell$ are both spherical tensor fields of rank ℓ, and we need the spherical tensor operations to create regional descriptors from the SH HOG field.

7.4 Tensor Operations for Regional Description

Orthogonal expansion Similar to the radial sampling approach discussed in the 2D case (Sect.6), we can compute descriptions on multiple concentric spherical shells around a selected point to describe the surrounding region, *e.g.*, to describe a 3D shape model surrounding its mass center (Kazhdan 2003). In this context, we can use the tensorial harmonic expansion (Eq.(34)) and the analytical rotation-invariance in Eq.(36) to get a rotation-invariant descriptor. Sampling in radial direction is equal to expanding the surrounding HOG field $\widetilde{\mathbf{F}}^\ell$ on the 3D basis $\delta(r - r_n)\mathbf{Y}^j(\theta, \varphi) \in \mathcal{T}_j$ with the tensor product. Sampling on multiple r_n, we can obtain a large group of expansion coefficients $\mathbf{a}^{j,k}_{\ell,n}$ (indices j, k are due to the tensorial expansion, ℓ is the degree of $\widetilde{\mathbf{F}}^\ell$, n comes from the radial sampling) which transform with $\mathcal{D}^{j+k}$. Thus the general formula of the rotation-invariant features is

$$\mathbf{a}^{j,k\dagger}_{\ell,n}\mathbf{a}^{j',k'}_{\ell',n'}, \ \forall j + k = j' + k'. \tag{40}$$

Similar to the 2D case, there is no restriction on n and n', which means we can couple the coefficients across multiple shells (so that the signals on different radii can not rotate independently). There is also no direct dependency on ℓ and ℓ' – the ranks of the coefficients in the SH HOG field, since the tensor rank of $\mathbf{a}^{j,k}_{\ell,n}$ is just $j + k$.

Fast filtering for dense feature computation To compute descriptors densely in the whole volume, we need the convolution between spherical tensor fields $\widetilde{\bullet}_{(\ell|\ell_1,\ell_2)}$ defined in Eq.(35). Similar to the 2D case, we just need some basis functions in the form $P(r)\mathbf{Y}^\ell(\theta, \varphi)$, which are the spherical tensors of order ℓ, to describe the neighborhood of each voxel. The convolution operation guarantees that the filtering results are still spherical tensor fields, so that we can create rotation-invariant features based on their rotation behavior. It has been proposed in Reisert and Burkhardt (2009) to choose specific functions as $P(r)$, by which filtering can be computed much more efficiently than performing one convolution for each feature. Deeper insight on the radial profiles $P(r)$ for fast filtering can be found in Skibbe (2012). Here we make use of such a basis, the so-called *spherical Gaussian derivatives* (SGDs).

We leave the details required for implementation to Appendix. Here we just denote the SGD basis by $\mathbf{G}_n(r, \theta, \varphi) = P_n(r)\mathbf{Y}^{j_n}(\theta, \varphi) \in \mathcal{T}_{j_n}$. The Gaussian in the SGD and the one in the spatial aggregation for HOG can be combined into a single Gaussian convolution. It is also possible to compute multi-scale features by adjusting the scale of the Gaussian.

Finally, we can establish rotation-invariance by coupling the filtering output of the same rank with the inner product. Formally, the rotation-invariant features in the final descriptor can be computed by

$$\left[\mathbf{G}_n \,\widetilde{\bullet}_{(j_n+\ell|j_n,\ell)} \hat{\mathbf{F}}^{\ell}\right]^{\dagger} \left[\mathbf{G}_{n'} \,\widetilde{\bullet}_{(j_{n'}+\ell'|j_{n'},\ell')} \hat{\mathbf{F}}^{\ell'}\right] \quad \forall\, j_n + \ell = j_{n'} + \ell' \tag{41}$$

where only the tensor rank j_n of the SGD basis is relevant for the rotations. The radial profiles $P_n(r)$, which depend on the derivative orders and the scale of the Gaussian, can be chosen freely. The formal definitions and computational rules are given in Appendix. The whole process for computing the 3D SH HOG descriptor densely in a straight-forward implementation (i.e. without the fast filtering using SGDs) is summarized in Algorithm 2 and 3.

Algorithm 2 3D SH HOG Descriptor

Input: 3D volumetric image I
Output: rotation-invariant feature vector field $\mathbf{C} : \mathbf{x} \mapsto \mathbf{C}(\mathbf{x})$

```
$\mathbf{D} = \nabla I$        // compute gradient
// step 1: compute SH HOG (Eq.(37))
for $\ell = 0 : \ell_{\max}$ do
    for $m = -\ell : \ell$ do
        $\hat{F}^{\ell}_{m}(\mathbf{x}) = \|\mathbf{D}(\mathbf{x})\| Y^{\ell}_{m}(\Theta(\mathbf{D}(\mathbf{x})), \Phi(\mathbf{D}(\mathbf{x})))$
        $\widetilde{F}^{\ell}_{m} = \hat{F}^{\ell}_{m} / \sqrt{\|\mathbf{D}\|^2 * K}$        // normalization
    end for
end for
// step 2: compute regional features
$i = 0$
for all $\widetilde{\mathbf{F}}^{\ell}$ do
    for all spherical tensor basis function $\mathbf{G}_n$ (with rank $j_n$) do
        for all selected rank $k$ within $|\ell - j_n| \le k \le \ell + j_n$ do
//          convolution of two spherical tensor fields using the tensor
            product (Algorithm 3)
            $\mathbf{B}_i(\mathbf{x}) = \int_{\mathbb{R}^3}$ tensor_product$(\widetilde{\mathbf{F}}^{\ell}(\mathbf{x}-\mathbf{x}'), \mathbf{G}_n(\mathbf{x}'), k) d\mathbf{x}'$
            $\ell_i = k$        // the rank of the feature
            $i = i + 1$
        end for
    end for
end for
// step 3: generate final rotation-invariant features
$\mathbf{C} = \varnothing$
for all output $\mathbf{B}_i$ do
    if $\ell_i = 0$ then
        append $\mathbf{B}_i$ to $\mathbf{C}$
    else
        append $\|\mathbf{B}_i\|$ to $\mathbf{C}$
    end if
end for
// more features can be generated by coupling (Eq.(36))
for all pairs of features $\mathbf{B}_i$ and $\mathbf{B}_{i'}$ do
    if $\ell_i = \ell_{i'}$ and $\ell_i \neq 0$ then
        append $\mathbf{B}_i{}^{\dagger}\mathbf{B}_{i'}$ to $\mathbf{C}$
    end if
end for
```

Algorithm 3 Tensor product (normalized)

Input: two spherical tensors $\mathbf{v} \in \mathbb{C}^{2\ell_1+1}$, $\mathbf{w} \in \mathbb{C}^{2\ell_2+1}$, and the output tensor rank ℓ
Output: a rank-ℓ spherical tensor $\mathbf{z} \in \mathbb{C}^{2\ell+1}$

```
$\mathbf{z} = 0$
for $m_1 = -\ell_1 : \ell_1$ do
    for $m_2 = -\ell_2 : \ell_2$ do
        $m = m_1 + m_2$
        if $-\ell \le m \le \ell$ then
            $z_m = z_m + \frac{\text{clebsch\_gordan\_coefficient}(\ell,m,\ell_1,m_1,\ell_2,m_2)}{\text{clebsch\_gordan\_coefficient}(\ell,0,\ell_1,0,\ell_2,0)} v_{m_1} w_{m_2}$
        end if
    end for
end for
// refer to Appendix for the computation of the Clebsch-Gordan coef-
   ficients
```

8 Experiments

In this section, 2D/3D experiments on public datasets and one application on biological images are presented, in addition to some basic validation experiments. We aim to show that the proposed descriptor inherits the good description ability from HOG and can facilitate rotation-invariant recognition tasks in different scenarios.

The Matlab code for the 2D experiment and a demo for the 3D dense description are available on our website[6].

8.1 Basic Validation Experiments

We first show some simple experiments to demonstrate how the analytically derived rotation invariance actually holds in practice.

Validation for rotation-invariance We computed Fourier HOG descriptors on rotated digit images (LeCun et al. 1998). (More details about the feature computation will be given in Sect.8.2.) The output features were compared in each feature dimension.

We found out that the image resolution of the objects and the basis functions, i.e., whether the objects and the basis functions are sufficiently sampled in their discretized representations, is the only factor that influences the invariance level in practice. This effect is illustrated in Fig.9.

Noise performance To test the performance of our descriptor under increasing noise levels (and low sampling rate), we added Gaussian noise to the rotated digit images (32 × 32 pixels). For each of the 10 digit images shown in Fig.10, we generated 18 rotated samples and computed the corresponding descriptors.

The performance was compared with the standard HOG (using the implementation from Felzenszwalb et al. (2010))

[6] http://lmb.informatik.uni-freiburg.de/resources/opensource/FourierHOG/

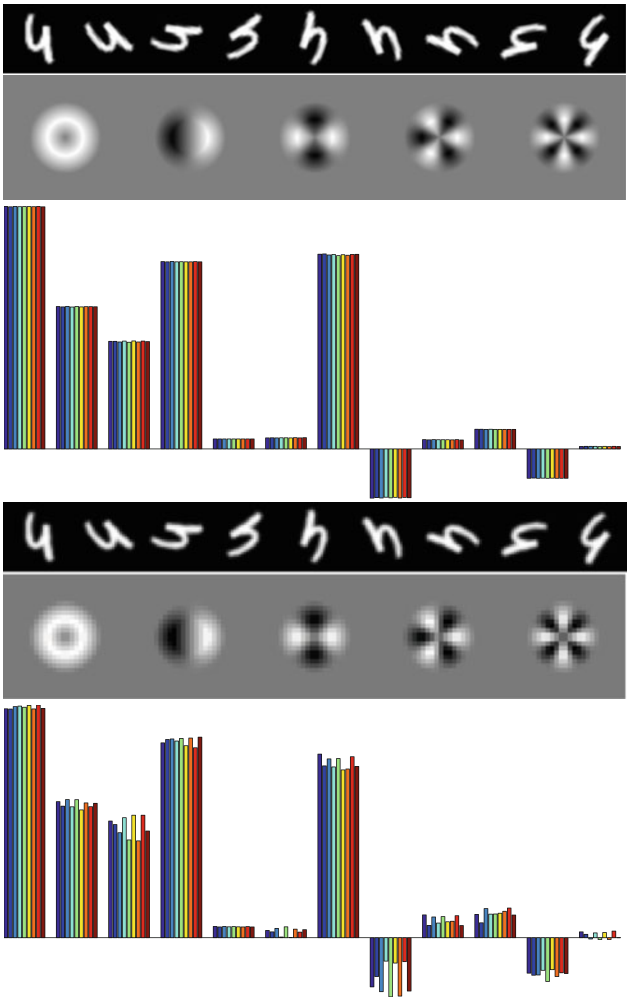


Fig. 9 Test images, basis functions and the output features. *Top* high resolution setting (224 × 224 pixels). *Bottom* low resolution setting (32 × 32 pixels). 12 (5 %) feature dimensions are shown in the *bar plots*. The test images have been rotated into 9 different orientations, so every group in the bar plots consists of 9 feature values which ideally should be the same. This figure demonstrates that the resolution (sufficient sampling) is critical for achieving an ideal invariance

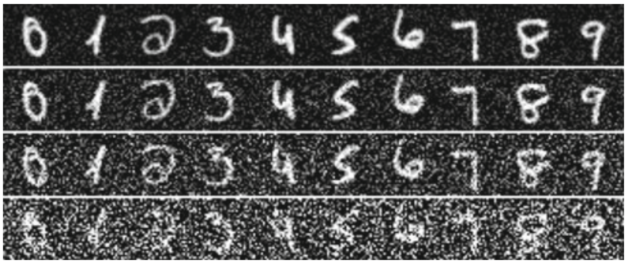


Fig. 10 Digit images with increasing Gaussian noise. From the *top* to the *bottom*, the peak signal-to-noise ratio (PSNR $= 20 \log_{10} \frac{1}{\sigma}$) is 20, 15, 10, 5

computed on pose-normalized images, based on the ground-truth orientations. That is, the samples all had the same orientation for evaluating standard HOG, while we included rotated samples for evaluating Fourier HOG.

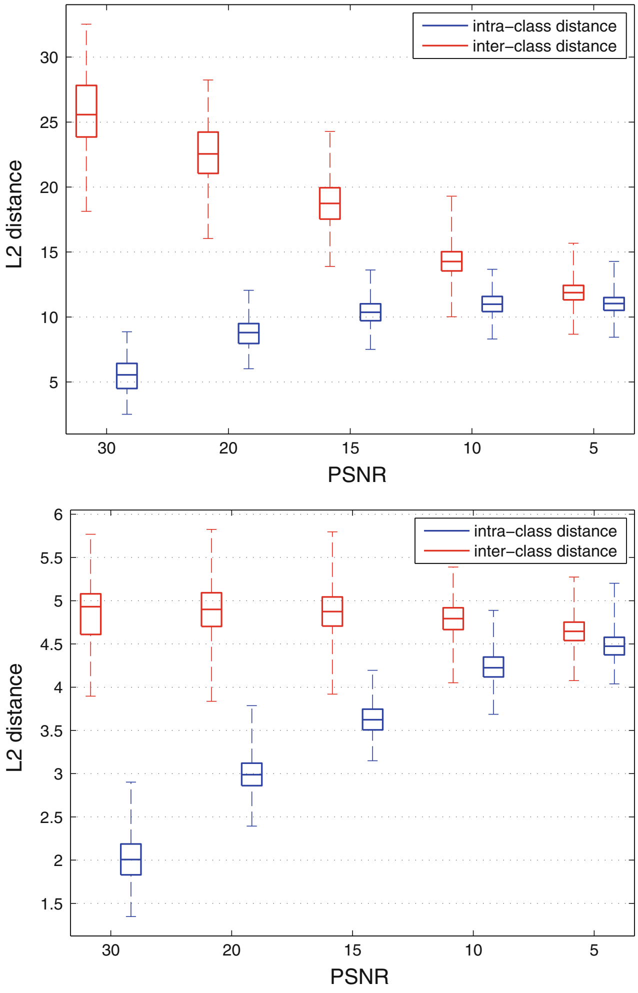


Fig. 11 Distances between different digits (inter-class) and among the same digits (intra-class). *Top* Fourier HOG on differently rotated samples. *Bottom* standard HOG on equally oriented samples. The boxplot depicts data groups through their five-number summaries: the minimum, lower quartile (25 %), median, upper quartile (75 %), and maximum. This comparison demonstrates that the Fourier HOG is as robust to noise as the standard HOG although it is rotation invariant

The noise performance is hard to analyze if solely based on the numerical change of the descriptor. We choose to use the separation of the intra/inter-class distances among the samples to evaluate the noise sensitivity. The boxplots in Fig.11 show the distributions of the L2 distances between feature vectors under different noise levels. The intra-class distance is the distance between two noisy samples of the same digit, while the inter-class distance is the one between two different digits.

In Fig.11, we can observe that the performance of the two descriptors are very close in this simple test, although the standard HOG needs the ground-truth orientations.

Based on these simple experiments, we can conclude that, the theoretically deduced rotation-invariance can be implemented in the discretized setting, given that the images and basis functions are sufficiently sampled. Even under the conditions of unideal sampling and noisy imaging, which allows for only approximate rotation-invariance, the descriptor performs reasonably well. Our rotation invariant descriptor is as robust under noise as the conventional HOG descriptor.

8.2 Dense Description for Sliding Window Detection

Dataset In this experiment we show a 2D task on a publicly available dataset, which has been used in Heitz and Koller (2008); Vedaldi et al. (2011); Schmidt and Roth (2012). This dataset consists of 30 aerial images with a total of 1319 cars annotated (Fig.12). The cars are rotated arbitrarily in the images. The task is challenging due to the low resolution and the varying illumination conditions caused by the shadows of buildings. We detect the cars by running a sliding window classifier on densely computed rotation-invariant HOG descriptors on a single scale. The detection performance is measured with the code provided in the dataset package from Heitz and Koller (2008). Like the compared work, we perform 5-fold cross-validation.

Implementation In the images, the maximal length of normal cars is about 40 pixels. We set our scale related parameters accordingly.

Step 1: The gradient magnitude is locally normalized w.r.t. the local gradient energy as $\mathbf{D}' = \frac{\mathbf{D}}{\|\mathbf{D}\| * K}$, where we use an isotropic triangle kernel K of a half-width equal to 12 pixels. The image gradients are projected into Fourier space according to Eq.(17). We empirically choose to use the first 5 degrees, which leads to $\hat{F}_m : m \in \{0, 1, 2, 3, 4\}$.
Step 2: We apply the filters $U_{j,k} = \Lambda(r - r_j, \sigma)e^{ik\theta}$ (Eq.(22)) to each $\hat{F}_m$ to get the regional descriptor. We set the radial profiles to be the triangles with half-width $\sigma = 6$ pixels and sampled at 4 radii, which are $r_j \in \{0, 6, 12, 18\}$ pixels. Only the lower degrees $-4 \leq k \leq 4$ are considered. The constructed basis functions are visualized in Fig.6 for radii larger than 0 and non-negative k. For $r_0 = 0$, we only need one isotropic triangle kernel. As a rule of thumb, we concentrate on the filtering results with lower rotation orders by only considering the terms with rotation order in the range of $-4 \leq k - m \leq 4$. This leads to 31 combinations of k and m. Together with the center triangle kernel, we have $31 \times 3 + 5 = 98$ complex features $f_{j,k,m}$.
Step 3: In practice, the coupling in Eq.(21) can lead to many possible combinations which may result in a very long feature vector. We test two settings. One is to only use the magnitude features $|f_{j,k,m}|$, which would lead to a 98-dimensional real-valued, rotation invariant descriptor. However, because some complex valued features are already rotation-invariant ($k - m = 0$), we can simply put their real and imaginary part into the descriptor to obtain a 110-dimensional real-valued feature vector. In a second setting, we use additional coupling between the features on different radii. To couple the two features $f_{j,k,m}$ and $f_{j',k,m}$, we compute $(\overline{f_{j,k,m}} f_{j',k,m})/\sqrt{|\overline{f_{j,k,m}} f_{j',k,m}|}$. The denominator serves for the purpose to make the coupled features compatible with the magnitude features while keeping the phase. Using the coupling between r_1, r_2 and r_2, r_3, we obtain a total of 232 real-valued features (a complex value is represented by two real values) in the final rotation-invariant descriptor. [7]

Classification and detection: We tested a linear Support-Vector-Machine (SVM) (Cortes and Vapnik 1995) using "LIBLINEAR" (Fan et al. 2008), and a Random Forest (RF) classifier (Breiman 2001). From the densely computed descriptors, the positive samples are taken from the center of the bounding boxes, while the negative samples are randomly sampled from areas outside the bounding boxes. To work with the linear SVM, we normalize each feature dimension into the range $[-1, 1]$. For detection we apply the trained classifier at every position. A non-maximum-suppression is used as in the normal sliding-window approaches Felzenszwalb et al. (2010).
With our desktop computer (4 × 3.2GHz CPU) and a pure Matlab implementation, feature computation on a 792 × 636 image takes 18 seconds (features are densely computed for each pixel). Most time is spent on the convolutions in step 2.
Evaluation result Figure13 shows the precision-recall curves of all detection results from the 5 tests in the cross-validation. We compare our results with other methods in Table1. Our method outperforms the two very recent publications which address the rotation problem in different ways. Vedaldi et al. (2011) uses structured SVM and the standard HOG feature. Schmidt and Roth (2012) focuses on features and descriptors. They propose rotation-aware feature learning with Restricted Boltzmann Machine and a rotation-invariant descriptor based on a polar grid, which shares some common concept with our work.
In contrast to standard HOG and discrete grid based methods, our descriptors can avoid most discretization artifacts and keep the substantial information about the gradient patterns. The mapping from the raw image to the final descriptors is continuous and smooth, besides being rotation-invariant. Thus it does not require nonlinearity

[7] The coupling used here is only a portion of all possible combinations. We prefer these simple choices since we only want to demonstrate the description power of the proposed method. We believe that the optimal feature selection is application-dependent. Using a classifier like linear SVM or Random Forest, which have built-in feature selection ability, allows to increase the dimensionality of the feature vector by adding more coupled features.

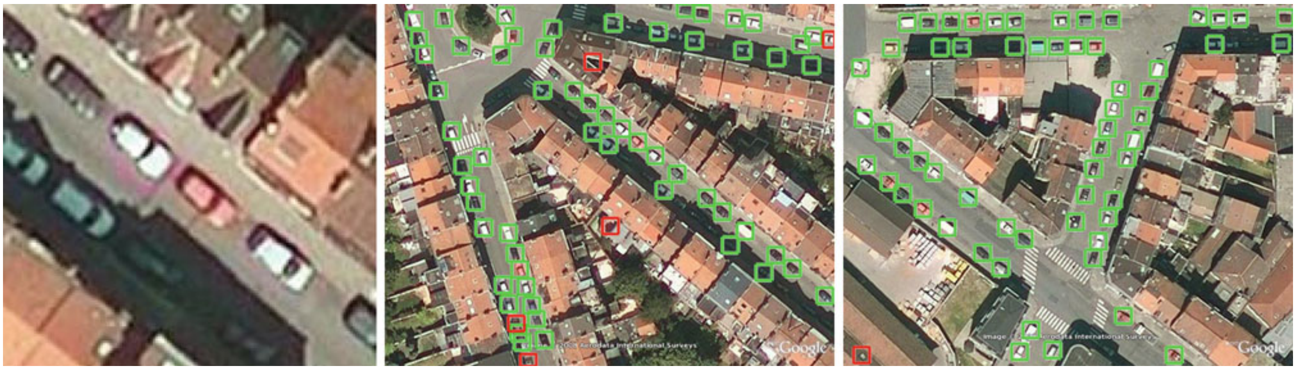


Fig. 12 An example of the aerial images and the qualitative results (true positive in *green*, false positive in *red*) from our descriptor in conjunction with a linear SVM (Color figure online)

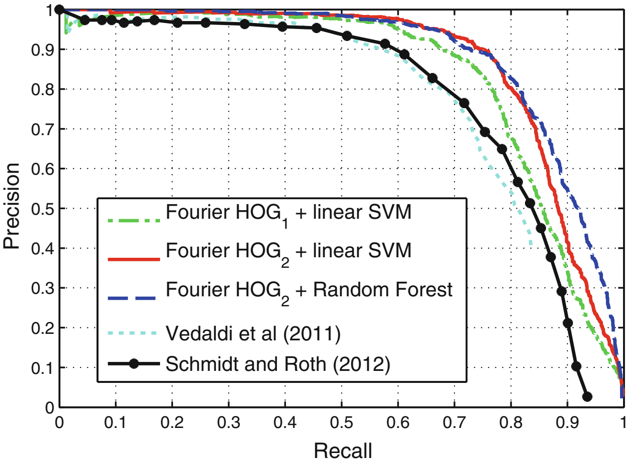


Fig. 13 Precision-Recall curve for the aerial car detection using our rotation-invariant HOG descriptor. The two settings of our Fourier HOG descriptors are discussed in the *implementation* step 3. The PR curves for the reference methods are depicted according to the original papers

Table 1 The average precision comparison for different methods on the aerial car detection

Feature	Classifier	AP(%)
Fourier HOG$_1$	linear SVM	79.9
Fourier HOG$_2$	linear SVM	82.6
Fourier HOG$_2$	Random Forest	84.2
Learned feature+grad.	linear SVM	77.6 [2]
standard HOG	nonlinear structured SVM	75.7 [1]
Learned feature	linear SVM	72.7 [2]
standard HOG	linear SVM	54.5 [2]

For our method (Fourier HOG) the subscripts indicate the two settings discussed in the *implementation* step 3. [1]: result cited from Vedaldi et al. (2011), [2]: result cited from Schmidt and Roth (2012)

to be built in the classifier and works well with the linear SVM. As a nonlinear classier, Random Forest further improves the classification. With the RF we can skip the feature normalization step. While the focus of this work is on the feature and descriptor level, our results may be incorporated into more involved machine learning methods. As we establish rotation invariance already analytically, the learning machinery can focus on other intrinsic variations of the samples.

Other related work on detection Besides the sliding window type detection, there is a fast rotation-invariant detection framework, the *equivariant filters* proposed by Reisert and Burkhardt (2008) based on Group Integration and Fourier analysis. This detection framework is related to Hough voting methods and is much faster at the cost of the discrimination ability. Equivariant filters using the HOG represented in Fourier/SH space have been demonstrated with biological applications in Skibbe and Reisert (2012); Skibbe et al. (2011). An improved equivariant filter using the Fourier HOG based regional descriptors has been shown to perform comparably with the state-of-the-art for motorbike detection in images from freestyle motocross (Liu et al. 2012).

8.3 Rotation-Invariant Descriptor for 3D Shapes

Dataset To evaluate our descriptor in 3D space, we use two 3D shape retrieval benchmarks. The 3D shapes are represented by triangle meshes and presented in varying poses. The employed benchmarks are the Princeton Shape Benchmark (PSB) (Shilane et al. 2004), in which the test partition has 907 objects in 92 classes (examples shown in Fig. 14), and SHREC 2009 Generic Shape Benchmark (Akgül et al. 2009) which runs 80 queries on 720 objects in 40 classes. They both provide evaluation tools for easy performance comparison, which basically check whether the computed pair-wise distances among shape models are compatible with the ground-truth similarities (shape categories). Our descriptor is designed for volumetric data, so the shape models are

Fig. 14 Examples of the 3D shapes in the Princeton Shape Benchmark

voxelized into a volume of 150^3 voxels (using "binvox"[8]), after the normalization for translation and scale (based on the mass center of all surface points, and their mean square distance to the mass center).

Implementation As we can assume a good alignment based on the center of mass, we only need to compute one centered description for the whole volume. This is a typical scenario where shell-wise expansion is the suitable strategy. Following the "SHD" method in Shilane et al. (2004), we first apply a distance transform on the binary volume data, then compute features on the transformed data.

Step 1: Voxel-wise SH HOG representations are only computed up to the first 5 degrees (Eq.(37)), leading to 5 spherical tensor fields. For this specific application, we do not apply any local normalization. Each SH HOG component $\hat{F}^{\ell}_{m}$ is smoothed by Gaussian convolutions.
Step 2: The tensorial harmonic expansion is computed on a number of sampled spherical shells, using Eqs.(34, 42, 43). We cut off the expansion at $j_{max} = 7$, leading to 160 tensorial expansion coefficients $\mathbf{a}^{j,k}_{\ell,n}$. Several parameter settings for the radial sampling step and the scale of Gaussian (in **Step 1**) were tried on the training partition of PSB, where we get the best result when using a large smoothing with $\sigma = 10$ pixels and sampling 12 spherical shells. These parameters are used for the PSB test partition and SHREC 2009.
Step 3: Based on Eq.(40), we compute the rotation-invariant magnitudes for each tensorial expansion coefficients by $\sqrt{\mathbf{a}^{j,k\dagger}_{\ell,n}\mathbf{a}^{j,k}_{\ell,n}}$. We further add coupled terms like $\left(\mathbf{a}^{j,k\dagger}_{\ell,n}\mathbf{a}^{j',k'}_{\ell',n}\right)/\sqrt{\left|\mathbf{a}^{j,k\dagger}_{\ell,n}\mathbf{a}^{j',k'}_{\ell',n}\right|}$ with $j + k = j' + k'$. Again the denominator is used to make the coupled terms compatible with the magnitudes. Since there are a large number of possible couplings, we focus on the low-frequency terms by restricting the combinations with an empirically set condition $j+k+\ell+j'+k'+\ell' \leq 6$, which gives 56 coupling combinations. We finally obtain a 216-dimensional rotation-invariant feature vector on each sampled spherical shell, thus we have 216×12 features for the whole volume, which is still a compact descriptor for a 3D volume especially when using HOG features. L1-norm distances are used to compute the pair-wise distance matrix.

With an unoptimized implementation, the running time for computing the descriptor is about 10 seconds for one 3D model (150^3 voxels) on a 3.2GHz CPU. The spatial smoothing alone costs about 5 seconds, where we apply Gaussian convolutions to each component of the SH HOG field. Besides the proposed "SH HOG + spherical tensor" approach, we also implemented and tested the standard SH expansion and the spherical tensorial expansion of the structure tensor field (Skibbe et al. 2009), for which we compute the expansion to $\ell_{max} = 16$ and $j_{max} = 16$.

Evaluation result Using the tool provided in the benchmarks, five measures are evaluated based on the ranked retrieval results for each query shape. *Nearest Neighbor* (NN) measures the classification accuracy of the nearest neighbor shape. *First Tier* (FT) and *Second Tier* (ST) measure the ratio of models in the same class that appear in the first K results. If C is the total number of models in the class of the query model, $K = C - 1$ for the first tier and $K = 2(C - 1)$ for the second tier. *E-measure* (E) is a combined measure of the precision and recall for a fixed number of results. *Discounted Cumulative Gain* (DCG) is a statistic measure of the entire ranked list that weights correct results near the front of the list more than correct results later. This is under the assumption that a user is less likely to consider elements near the end of the list. Results on PSB are listed in Table 2. We refer to the benchmark paper (Shilane et al. 2004) for more baseline methods evaluated on PSB, and some more detailed explanations on the evaluation measures.

Our descriptor consistently outperforms all other compared methods, independent of the evaluation measure. The proposed method is different to the classical SH based volumetric descriptors ("StrT-ST" and "SH"). This is not only due to the rank of tensors, but also by the fact that the local gradient patterns can be better preserved under smoothing in our method, because they are encoded into HOG-like features. Being a frequency-domain approach, SH based descriptors tend to be sensitive to small disturbance. That might explain why using the HOG feature together with a large smoothing can bring a significant improvement. In contrast to other methods, our method also benefits from the rotation-invariance which does not depend on any pose estimation or orientation discretization.

The results on SHREC 2009 are listed in Table3. We only list the other two closely related methods (which use a simple distance measure on a single type of descriptors). Several more sophisticated approaches are reported in Akgül et al. (2009). Most of them are specially designed for 3D surfaces and go beyond the scope of rotation-invariant descriptors. Nonetheless we have a good ranking position among them (rank 3 out of 19 for the Nearest Neighbor measure, rank 8 out of 19 for the DCG measure).

[8] Patrick Min, https://www.google.com/search?q=binvox

Table 2 The retrieval statistics for compared methods on the Princeton Shape Benchmark

Method	V/S	NN(%)	FT(%)	ST(%)	E(%)	DCG(%)
HOG_{SH}-ST	V	**69.1**	**41.5**	**53.2**	**30.8**	**67.4**
$\text{SH}_{\text{silhouette}}$	S	67.3	41.2	50.2	29.6	65.9
LFD	S	65.7	38.0	48.7	28.0	64.3
REXT	S	60.2	32.7	43.2	25.4	60.1
StrT-ST	V	61.7	30.7	39.6	23.2	58.2
SH	V	56.0	28.4	37.6	22.3	56.0
HOG_{norm}	V	58	27	35	21	55

The methods are sorted according to the Discounted Cumulative Gain, which is a measure of the similarity ranking. The "V/S" symbol indicates whether the method is suitable for general volumetric description or is specifically designed for surfaces. Three of the methods are implemented by ourselves, including HOG_{SH}-*ST* our proposed approach, *StrT-ST* spherical tensorial expansion on structure tensor field (Skibbe et al. 2009), *SH* the standard SH expansion. Other results are taken from literature. $\text{SH}_{\text{silhouette}}$: a SH based rotation-invariant descriptor built on uniformly sampled silhouette images on the view sphere (Makadia and Daniilidis 2010), *LFD* Light Field Descriptor, a collection of images rendered from uniformly sampled positions on the view sphere, which is the best method reported in the benchmark paper (Shilane et al. 2004), *REXT* Radialized Spherical Extent Function, a collection of spherical functions giving the maximal distance from center of mass, which is the second best method reported in the benchmark paper, $\textit{HOG}_{\text{norm}}$ HOG feature computed on pose-normalized 3D objects (Scherer et al. 2010)

Table 3 The retrieval statistics for closely related methods on the SHREC 2009 Generic Shape Benchmark

Method	NN(%)	FT(%)	ST(%)	E(%)	DCG(%)
HOG-ST	**92.5**	**55.0**	**68.7**	**47.8**	**82.4**
StrT-ST	81.2	39.0	49.3	34.1	71.2
HOG_{norm} (Scherer et al. 2010)	75	41	52	35	71

Throughout the experiments in this section, we have demonstrated the good description ability of our descriptor, although it was not designed for surface description. The descriptor becomes even more interesting when it is applied to volumetric data.

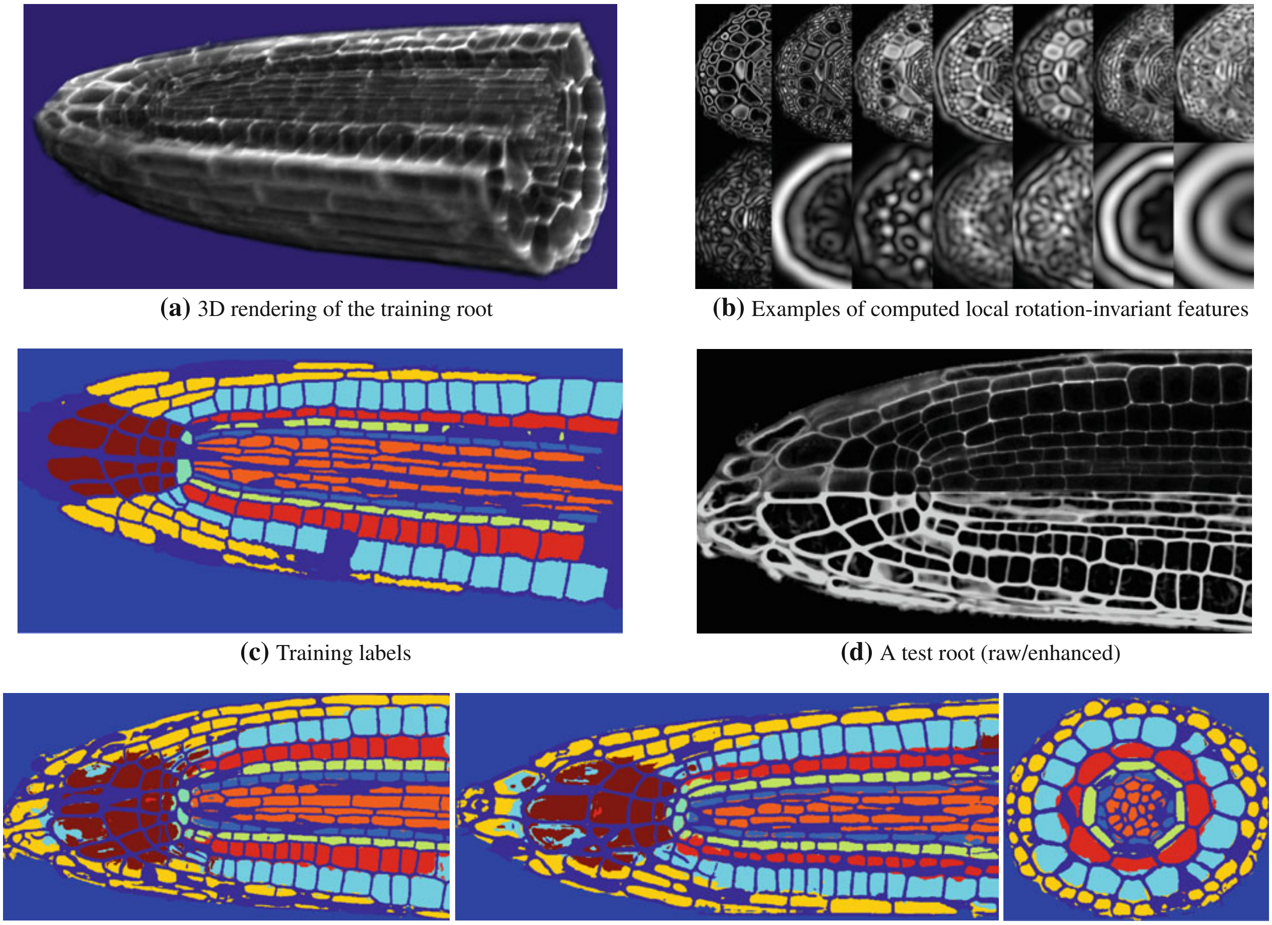
8.4 Dense Features for Labeling in Volumetric Data

Application We show a real application of our invariant descriptor on a labeling (semantic segmentation) problem. The data comes from confocal microscopic imaging of Arabidopsis roots with stained cell walls (Fig. 15a). The cell wall gives a representation of the cell's outer shape, which could be used for analyzing cell development or serve as a reference structure for sub-cellular event description. Here we aim for a preliminary cell segmentation and an additional structural segmentation which assigns cells to different layers to support model fitting and further analysis. It is a difficult problem due to the uneven imaging quality in the data and the large variance of the cell shapes even in the same layer. We choose to do the segmentation by voxel-wise classification, which means embedding all neighboring information into voxel-wise features and labeling each voxel by a trained classifier. Our rotation-invariant descriptor is necessary for this task as the cells in the same layer are oriented in different directions. The classification task are split into two subproblems: a two-class problem (cell-wall/none-cell-wall) and a multi-class problem (different root layers in none-cell-wall regions).

Implementation In this experiment, the pre-processing includes nonlinear diffusion for denoising and Hessian-matrix based edge enhancement. The proposed approach starts from representing the local gradient with SH coefficients using the first 5 degrees (Eq.(37)). To get dense descriptions, we choose to use the SGD basis functions. An implementation of SGD filters is available from Skibbe (2012). The filtering (Eqs. (41,48) is applied on the SH HOG field at 7 scales from $0.25\mu m$ to $16\mu m$ (the diameter of a root is around $100\mu m$), limiting $j_u + j_d \leq 5$ (we reduce the order of derivatives at small scales to respect the sampling theorem). The invariant features are derived from just the magnitude ($L2$-norm) of the tensor-valued filter output, without any complex couplings. Finally, together with the local intensity value, we get 240-dimensional voxel-wise features. Some examples are shown in Fig. 15b. The features at different scales encode information from the local edge strength to the global geometrical feature. On these features, SVMs with an RBF kernel are trained from one root with manual labels (Fig. 15c)[9], using "LIBSVM" (Chang and Lin 2011).

Results As the manual labeling is very expensive, we only created ground-truth for one root image. We run a simple

[9] We created the ground-truth by editing a watershed segmentation result manually. Some very badly segmented regions were discarded and were not used for training.



(a) 3D rendering of the training root

(b) Examples of computed local rotation-invariant features

(c) Training labels

(d) A test root (raw/enhanced)

(e) Qualitative result on a test root

(f) Qualitative result on another test root

(g) Qualitative result (cross section)

Fig. 15 Segmentation on Arabidopsis root. **(c)** training labels include 10 classes: background / cell wall / 6 layers and 2 regions

Table 4 Classification accuracy on randomly sampled voxels using different numbers of training samples

	Cell-wall		Root-layers	
$N_{training}$	SGD_{HOG}	$SGD_{intensity}$	SGD_{HOG}	$SGD_{intensity}$
400	95.1 %	94.6 %	92.7 %	77.1 %
800	96.1 %	95.5 %	95.6 %	82.9 %
1600	96.5 %	96.0 %	96.4 %	87.2 %
4000	97.8 %	96.5 %	96.5 %	90.6 %

quantitative test: we sample 17,000 points in the labeled root, and separate them into two parts by a mid-plane in the root, then a cross-validation is carried out by using different numbers of training samples from one part and testing on another part. The performance is summarized in Table 4, with the comparison with a SGD filtering on the intensity value (Skibbe 2012). The SH HOG based features clearly perform better, especially for the root layer labeling with a small number of training samples, demonstrating its more powerful description ability. Because of the high symmetry in one root, the ascending accuracy of SGD filtering on intensity possibly comes from over-fitting. When testing on other roots, the HOG-based features give satisfying results (raw classification results, without any post-processing, are shown in Fig.15(d∼f)) while the direct SGD filtering fails.

By fitting an adaptive cylindrical coordinate to this rough layer segmentation result (Schmidt et al. 2012), one can obtain a coordinate system, which is useful in many quantitative cell analysis studies in plant roots.

9 Summary

In this paper, we have presented a new approach to create rotation-invariant HOG descriptors for 2D and 3D images, which connects the HOG idea with the analytical rotation-invariance from Fourier analysis. The key idea is to consider a gradient histogram (a HOG cell) as a continuous signal in polar or spherical coordinates, and to represent it with a harmonic basis. We further compute Fourier based features on the Fourier HOG field, from which we can easily generate rotation-invariant descriptors thanks to the simple rotation behavior of the Fourier basis. This method can be generalized by replacing the gradient with any other image features that have directional information, *e.g.*, steerable filters.

We have demonstrated the generality and the effectiveness of our descriptor in three experiments: the car detection task in aerial images, the 3D shape retrieval benchmarks and a biological application – semantic segmentation in Arabidopsis roots. In all cases we obtained state-of-the-art performance.

By construction, the descriptor stays fixed when the object rotates, and changes continuously w.r.t. the change of the object appearance. This is the main advantage of the proposed descriptor, especially for recognition tasks where the rotation is the major intra-class variation.

There are also disadvantages compared with standard HOG. The simplicity in spatial selectivity and an intuitive visualization are lost. Also computation is more expensive for the use of convolutions.

The underlying concept in this paper is the analysis and manipulation of the rotation behavior in the polar/spherical Fourier analysis framework, which is applicable to every analysis step: raw images, gradients, HOG features, expansion coefficients, and filtering results. We believe many other problems regarding rotations can benefit from the analysis method summarized and demonstrated in this paper.

Acknowledgments This study was supported by the Excellence Initiative of the German Federal and State Governments: BIOSS Centre for Biological Signalling Studies (EXC 294) and the Bundesministerium für Bildung und Forschung (German Federal Ministry of Education and Research) Project: New Methods in Systems Biology (SYSTEC, 0101-31P5914) – Quantitative 3D and 4D cell analysis in living organisms. Henrik Skibbe is indebted to the Baden-Württemberg Stiftung for the financial support by the Elite Program for Post-docs. Dr. Thomas Blein was supported by a long-term post-doctoral fellowship from European Molecular Biology Organization (EMBO, ALTF250-2009). Dr. Thomas Blein and Prof. Klaus Palme are also supported by Deutsches Zentrum für Luft und Raumfahrt (DLR 50WB1022) and the European Union Framework 6 Program (AUTOSCREEN, LSHG-CT-2007-037897).

Appendix

Computation of the Tensorial Harmonic Expansion

Given a spherical tensor field $\mathbf{F} \in \mathcal{T}^{\ell}$, we have a way to compute the tensorial harmonic expansion in Eq.(34), which is more efficient than the direct projections.

First we compute the scalar (SH) expansion on each individual tensor component $F_m : \mathbb{R}^3 \to \mathbb{C}$ as

$$F_m(r,\theta,\varphi) = \sum_{j=0}^{\infty} \sum_{n=-j}^{j} \overline{\hat{b}^j_{m,n}}(r) Y^j_n(\theta,\varphi), \tag{42}$$

then the tensorial expansion coefficients $\mathbf{a}^{j,k}(r)$ can be computed from the above component-wise expansions by a derived relation as

$$a^{j,k}_{m'}(r) = \frac{2(j+k)+1}{2\ell+1} \sum_{m,n} \hat{b}^j_{m,n}(r) C(\ell, m | j+k, m', j, n), \tag{43}$$

where $-(j+k) \leq m' \leq j+k$. See Reisert and Burkhardt (2009) for proofs. We need to compute the ClebschGordan coefficients C in this circumstance. An easy way is to use their relation to the Wigner 3-j symbols $\begin{pmatrix} j_1 & j_2 & j_3 \\ m_1 & m_2 & m_3 \end{pmatrix}$ (Brink and Satchler 1968), which is written as

$$C(j_3, m_3 | j_1, m_1, j_2, m_2) = (-1)^{j_1 - j_2 + m_3} \sqrt{2j_3+1} \times \begin{pmatrix} j_1 & j_2 & j_3 \\ m_1 & m_2 & m_3 \end{pmatrix}. \tag{44}$$

One can use the function "gsl_sf_coupling_3j" in the GNU Scientific Library to compute the Wigner 3-j symbol.

Spherical Gaussian Derivatives

Let $\mathbf{F} \in \mathcal{T}_{\ell}$, the spherical up-derivative $\boldsymbol{\nabla}^1 : \mathcal{T}_{\ell} \to \mathcal{T}_{\ell+1}$ and the down-derivative $\boldsymbol{\nabla}_1 : \mathcal{T}_{\ell} \to \mathcal{T}_{\ell-1}$ (Reisert and Burkhardt 2009) are defined as

$$\boldsymbol{\nabla}^1 \mathbf{F} := \nabla \bullet_{(\ell+1|1,\ell)} \mathbf{F}, \tag{45}$$

$$\boldsymbol{\nabla}_1 \mathbf{F} := \nabla \bullet_{(\ell-1|1,\ell)} \mathbf{F}, \tag{46}$$

where $\nabla = (\frac{1}{\sqrt{2}}(\partial_x - \mathrm{i}\partial_y), \partial_z, -\frac{1}{\sqrt{2}}(\partial_x + \mathrm{i}\partial_y))$ is the spherical gradient operator with $\partial_x, \partial_y, \partial_z$ being the standard partial derivatives. It is further defined that $\boldsymbol{\nabla}^{j_u}_{j_d} \mathbf{V} = \underbrace{\boldsymbol{\nabla}_1 \ldots \boldsymbol{\nabla}_1}_{j_d \text{ times}} \underbrace{\boldsymbol{\nabla}^1 \ldots \boldsymbol{\nabla}^1}_{j_u \text{times}} \mathbf{V}$. One important property of this operation is that it maps a spherical tensor field to a higher or lower rank spherical tensor field. This is analogous to the fact that computing derivatives on a scalar field produces a gradient field, which is a rank-1 tensor, and a subsequent derivative can either produce the Hessian (rank-2 tensor) or the divergence (rank-0 tensor).

For $\mathbf{V} = \boldsymbol{\nabla}^1 \mathbf{V}'$, where $\mathbf{V}' : \mathbb{R}^3 \to \mathbb{C}^{2(\ell-1)+1}, \mathbf{V} : \mathbb{R}^3 \to \mathbb{C}^{2\ell+1}$, by indexing the elements of $\mathbf{V}$ and $\mathbf{V}'$ as $\{V_{-\ell}, \ldots, V_{\ell}\}$ and $\{V'_{-\ell+1}, \ldots, V'_{\ell-1}\}$, the computation rule of $\boldsymbol{\nabla}^1$ is:

$$\begin{aligned} V_m = {} & w(\ell, m, -1) \, \frac{1}{\sqrt{2}} (\partial_x - \mathrm{i}\partial_y) V'_{m+1} \\ & + w(\ell, m, 0) \, \partial_z V'_m \\ & - w(\ell, m, 1) \, \frac{1}{\sqrt{2}} (\partial_x + \mathrm{i}\partial_y) V'_{m-1} \quad , \end{aligned} \tag{47}$$

where w is the weighting coefficients which can be precomputed from two *Clebsch-Gordan* coefficients as

$w(\ell, m, a) = \frac{C(\ell,m|\ell-1,m-a,1,a)}{C(\ell,0|\ell-1,0,1,0)}$. Thus the computation of the spherical tensor derivatives is just a group of weighted combinations of normal Cartesian derivatives.

Equation (47) also fits the spherical down-derivative $\mathbf{V} = \boldsymbol{\nabla}_1 \mathbf{V}'$, where $\mathbf{V} : \mathbb{R}^3 \rightarrow \mathbb{C}^{2\ell+1}$ and $\mathbf{V}' : \mathbb{R}^3 \rightarrow \mathbb{C}^{2(\ell+1)+1}$. The only difference are the coefficients: $w(\ell, m, a) = \frac{C(\ell,m|\ell+1,m-a,1,a)}{C(\ell,0|\ell+1,0,1,0)}$.

A fast filtering tool is derived by computing the derivatives on an isotropic Gaussian function, which creates a series of basis function of different tensor ranks, as $\boldsymbol{\nabla}^{j_u}_{j_d} G \in \mathcal{T}_{j_u - j_d}$ (where $j_u \geq j_d$, G is a Gaussian function). The convolution with the spherical Gaussian derivatives can be computed efficiently like the standard Gaussian derivatives based on the commutativity of the convolution and differentiation. As an example, let $\mathbf{F} \in \mathcal{T}_\ell$ be a spherical tensor field, we have

$$\boldsymbol{\nabla}^{j_u}_{j_d} G \,\widetilde{\bullet}_{(\ell+j_u-j_d|j_u-j_d,\ell)}\, \mathbf{F} = \boldsymbol{\nabla}^{j_u}_{j_d} (G \,\widetilde{\bullet}_{(\ell|0,\ell)} \mathbf{F}). \tag{48}$$

We can therefore compute multiple filtering outputs (for different $\{j_u, j_d\}$) by a single tensorial convolution plus differentiations. Note, the convolution like $G \,\widetilde{\bullet}_{(\ell|0,\ell)}\, \mathbf{F}$ is equivalent to normal Gaussian convolutions as $[G \,\widetilde{\bullet}_{(\ell|0,\ell)} \mathbf{F}]_m = G * F_m$ (because $C(\ell, m|\ell, m, 0, 0) = 1$). The output is a tensor field of rank $\ell + j_u - j_d$. In the context of this paper, we can take the SGD as derivatives after a scale-space selection by Gaussian convolution. The only important property for the rotation-invariance is that the introduced basis functions are spherical tensor fields.

References

Ahonen, T., Matas, J., He, C., Pietikäinen, M. (2009). *Rotation invariant image description with local binary pattern histogram Fourier features*. In Scandinavian Conference on Image, Analysis, pp. 61–70.

Akgül, C., Axenopoulos, A., Bustos, B., Chaouch, M., Daras, P., Dutagaci, H., Furuya, T., Godil, A., Kreft, S., Lian, Z., et al. (2009). *SHREC 2009-Generic Shape Retrieval contest*. In Eurographics workshop on 3D object retrieval.

Allaire, S., Kim, J., Breen, S., Jaffray, D., & Pekar, V. (2008). *Full orientation invariance and improved feature selectivity of 3D SIFT with application to medical image analysis*. In CVPR Workshops.

Arsenault, H., & Sheng, Y. (1986). Properties of the circular harmonic expansion for rotation-invariant pattern recognition. *Applied Optics*, *25*(18), 3225–3229.

Bendale, P., Triggs, B., & Kingsbury, N. (2010). *Multiscale keypoint analysis based on complex wavelets*. In British Machine Vision Conference, pp. *49*(1–49), 10.

Bourdev, L., Malik, J. (2009). *Poselets: Body part detectors trained using 3D human pose annotations*. In International Conference on Computer Vision, pp. 1365–1372.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Brink, D., & Satchler, G. (1968). *Angular momentum*. Oxford: Clarendon Press.

Bülow, T. (2004). Spherical diffusion for 3D surface smoothing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*(12), 1650–1654.

Burkhardt, H., & Siggelkow, S. (2001). Invariant features in pattern recognition—fundamentals and applications. In C. Kotropoulos & I. Pitas (Eds.), *Nonlinear model-based image/video processing and analysis* (pp. 269–307). New York: Wiley.

Chang, C.-C., Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology, 2*,27:1–27:27. Software available at http://www.csie.ntu.edu.tw/cjlin/libsvm

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297.

Dalal, N., Triggs, B. (2005). *Histograms of oriented gradients for human detection*. In IEEE Conference on Computer Vision and, Pattern Recognition, pp. 886–893.

Driscoll, J., & Healy, D. (1994). Computing Fourier transforms and convolutions on the 2-sphere. *Advances in Applied Mathematics*, *15*(2), 202–250.

Fan, R., Chang, K., Hsieh, C., Wang, X., & Lin, C. (2008). LIBLINEAR: A library for large linear classification. *The Journal of Machine Learning Research*, *9*, 1871–1874.

Fehr, J. (2010). *Local rotation invariant patch descriptors for 3D vector fields*. In International Conference on, Pattern Recognition, pp. 1381–1384.

Felzenszwalb, P., Girshick, R., McAllester, D., & Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(9), 1627–1645.

Flitton, G., Breckon, T., & Megherbi, N. (2010). *Object recognition using 3D SIFT in complex CT volumes*. In British Machine Vision Conference, pp. 11(1–11), 12.

Fornasier, M., & Toniolo, D. (2005). Fast, robust and efficient 2D pattern recognition for re-assembling fragmented images. *Pattern Recognition*, *38*(11), 2074–2087.

Förstner, W., Gülch, E. (1987). *A fast operator for detection and precise location of distinct points, corners and centres of circular features*. In ISPRS intercommission conference on fast processing of photogrammetric data, pp. 281–305.

Freeman, W., & Adelson, E. (1991). The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *13*(9), 891–906.

Gauglitz, S. (2011). *Improving keypoint orientation assignment*. In British Machine Vision Conference, pp. *93*(1–93), 11.

Giannakis, G. (1989). Signal reconstruction from multiple correlations: frequency- and time-domain approaches. *Journal of Optical Society of America A*, *6*(5), 682–697.

Golub, G., & Van Loan, C. (1996). *Matrix computations*. Baltimore: Johns Hopkins Univ Press.

Green, R. (2003). Spherical harmonic lighting: The gritty details. *In Game Developers Conference*, *2*, 2–3.

Haasdonk, B., & Burkhardt, H. (2007). Invariant kernel functions for pattern analysis and machine learning. *Machine Learning*, *68*(1), 35–61.

Heitz, G., Koller, D. (2008). *Learning spatial context: Using stuff to find things*. In European Conference on Computer Vision, pp. 30–43.

Jacovitti, G., & Neri, A. (2000). Multiresolution circular harmonic decomposition. *IEEE Transaction on Signal Processing*, *48*(11), 3242–3247.

Kavukcuoglu, K., Ranzato, M., Fergus, R., Le-Cun, Y. (2009). *Learning invariant features through topographic filter maps*. In IEEE Conference on Computer Vision and, Pattern Recognition, pp. 1605–1612.

Kazhdan, M., Funkhouser, T., Rusinkiewicz, S. (2003). *Rotation invariant spherical harmonic representation of 3D shape descriptors*. In Eurographics/ACM SIGGRAPH symposium on Geometry processing, pp. 156–164.

Kläser, A., Marszałek, M., Schmid, C. (2008). *A spatio-temporal descriptor based on 3D-gradients*. In British Machine Vision Conference, pp. 995–1004.

Knopp, J., Prasad, M., Van Gool, L. (2010a). *Orientation invariant 3D object classification using Hough transform based methods*. In ACM Multimedia, Workshop, pp. 15–20.

Knopp, J., Prasad, M., Willems, G., Timofte, R., Van Gool, L. (2010b). *Hough transform and 3D SURF for robust three dimensional classification*. In European Conference on Computer Vision, pp. 589–602.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278–2324.

Lenz, R. (1990). *Group theoretical methods in image processing*. Berlin: Springer.

Lin, W., Liu, L., Matsushita, Y., Low, K., Liu, S. (2012). *Aligning images in the wild*. In IEEE Conference on Computer Vision and, Pattern Recognition, pp. 1–8.

Liu, K., Skibbe, H., Schmidt, T., Blein, T., Palme, K., & Ronneberger, O. (2011). *3D rotation-invariant description from tensor operation on spherical HOG field*. In British Machine Vision Conference, pp. *33*(1-33), 12.

Liu, K., Wang, Q., Driever, W., Ronneberger, O. (2012). *2D/3D Rotation-invariant Detection using Equivariant Filters and Kernel Weighted Mapping*. In IEEE Conference on Computer Vision and, Pattern Recognition, pp. 917–924.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, *60*(2), 91–110.

Makadia, A., & Daniilidis, K. (2010). Spherical correlation of visual representations for 3D model retrieval. *International Journal of Computer Vision*, *89*(2), 193–210.

Memisevic, R., & Hinton, G. (2010). Learning to represent spatial transformations with factored higher-order boltzmann machines. *Neural Computation*, *22*(6), 1473–1492.

Özuysal, M., Calonder, M., Lepetit, V., & Fua, P. (2010). Fast keypoint recognition using random ferns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(3), 448–461.

Ponce, C., & Singer, A. (2011). Computing steerable principal components of a large set of images and their rotations. *IEEE Transactions on Image Processing*, *20*(11), 3051–3062.

Reisert, M., & Burkhardt, H. (2008a). *Efficient tensor voting with 3D tensorial harmonics*. In CVPR Workshops.

Reisert, M., & Burkhardt, H. (2008b). Equivariant holomorphic filters for contour denoising and rapid object detection. *IEEE Transactions on Image Processing*, *17*(2), 190–203.

Reisert M., Burkhardt H. (2009) Spherical Tensor Calculus for Local Adaptive Filtering. In: Aja-Fernández S., de Luis García R., Tao D., Li X. (eds) Tensors in Image Processing and Computer Vision Advances in Pattern Recognition. Springer, USA, pp. 153–178.

Ronneberger, O., Burkhardt, H., & Schultz, E. (2002). *General-purpose Object Recognition in 3D Volume Data Sets using Gray-Scale Invariants—Classification of Airborne Pollen-Grains Recorded with a Confocal Laser Scanning Microscope.* In International Conference on Pattern Recognition, *2*, 290–295.

Ronneberger, O., Liu, K., Rath, M., Ruess, D., Mueller, T., Skibbe, H., et al. (2012). ViBE-Z: a framework for 3D virtual colocalization analysis in zebrafish larval brains. *Nature Methods*, *9*(7), 735–742.

Ronneberger, O., Wang, Q., & Burkhardt, H. (2007). *3D Invariants with High Robustness to Local Deformations for Automated Pollen Recognition* (pp. 455–435). Pattern recognition: In DAGM conference on.

Rose, M. (1957). *Elementary theory of angular momentum*. New York: Wiley.

Scherer, M., Walter, M., & Schreck, T. (2010). *Histograms of Oriented Gradients for 3D Model Retrieval* (pp. 41–48). Visualization and Computer Vision: In International Conference in Central Europe on Computer Graphics.

Schmidt, T., Keuper, M., Pasternak, T., Palme, K., & Ronneberger, O. (2012). *Modeling of Sparsely Sampled Tubular Surfaces Using Coupled Curves* (pp. 83–92). Pattern recognition: In DAGM conference on.

Schmidt, U., Roth, S. (2012). Learning rotation-aware features: From invariant priors to equivariant descriptors. In IEEE Conference on Computer Vision and, Pattern Recognition, pp. 2050–2057.

Schultz, T., Weickert, J., & Seidel, H. (2009). A higher-order structure tensor. In D. Laidlaw & J. Weickert (Eds.), *Visualization and processing of tensor fields* (pp. 263–279). Berlin: Springer.

Sheng, Y., & Arsenault, H. (1986). Experiments on pattern recognition using invariant Fourier-Mellin descriptors. *Journal of Optical Society of America A*, *3*(6), 771–776.

Shilane, P., Min, P., Kazhdan, M., Funkhouser, T. (2004). The Princeton Shape Benchmark. In International Conference on Shape Modeling and Applications, pp. 167–178.

Skibbe, H., & Reisert, M. (2012). *Circular Fourier-HOG features for rotation invariant object detection in biomedical images*. In IEEE International Symposium on Biomedical Imaging, pp. 450–453.

Skibbe, H., Reisert, M., & Burkhardt, H. (2011). *SHOG-spherical HOG descriptors for rotation invariant 3D object detection*. In DAGM conference on Pattern recognition, pp. 142–151.

Skibbe, H., Reisert, M., Ronneberger, O., & Burkhardt, H. (2009). *Increasing the dimension of creativity in rotation invariant feature design using 3D tensorial harmonics*. In DAGM conference on Pattern recognition, pp. 141–150.

Skibbe, H., Reisert, M., Schmidt, T., Brox, T., Ronneberger, O., Burkhardt, H. (2012). Fast rotation invariant 3D feature computation utilizing efficient local neighborhood operators. IEEE Transactions on Pattern Analysis and Machine Intelligence, 34(8):1563–1575. Software available at https://bitbucket.org/skibbe/sta-imagetoolbox

Takacs, G., Chandrasekhar, V., Tsai, S., Chen, D., Grzeszczuk, R., Girod, B. (2010). *Unified real-time tracking and recognition with rotation-invariant fast features*. In IEEE Conference on Computer Vision and, Pattern Recognition, pp. 934–941.

Vedaldi, A., Blaschko, M., Zisserman, A. (2011). *Learning equivariant structured output SVM regressors*. In International Conference on Computer Vision, pp. 959–966.

Villamizar, M., Moreno-Noguer, F., Andrade-Cetto, J., Sanfeliu, A. (2010). *Efficient rotation invariant object detection using boosted random ferns*. In IEEE Conference on Computer Vision and, Pattern Recognition, pp. 1038–1045.

Wang, Q., Ronneberger, O., & Burkhardt, H. (2009). Rotational invariance based on fourier analysis in polar and spherical coordinates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *31*, 1715–1722.

Wolberg, G., Zokai, S. (2000). *Robust image registration using log-polar transform.* In IEEE International Conference on Image Processing, pp. 493–496.