CVPR 2012

2D/3D Rotation-Invariant Detection using Equivariant Filters and Kernel Weighted Mapping

Kun Liu[1,3], Qing Wang[1], Wolfgang Driever[2,3], and Olaf Ronneberger[1,3]
[1]Department of Computer Science [2]Institute of Biology I
[3]BIOSS Center for Biological Signaling Studies
University of Freiburg, Germany
http://lmb.informatik.uni-freiburg.de

Abstract

In many vision problems, rotation-invariant analysis is necessary or preferred. Popular solutions are mainly based on pose normalization or brute-force learning, neglecting the intrinsic properties of rotations. In this paper, we present a rotation invariant detection approach built on the equivariant filter framework, with a new model for learning the filtering behavior. The special properties of the harmonic basis, which is related to the irreducible representation of the rotation group, directly guarantees rotation invariance of the whole approach. The proposed kernel weighted mapping ensures high learning capability while respecting the invariance constraint. We demonstrate its performance on 2D object detection with in-plane rotations, and a 3D application on rotation-invariant landmark detection in microscopic volumetric data.

1. Introduction

Rotation invariance is useful when objects of the same class can appear in different poses. Common solutions in computer vision are based on either pose normalization (*e.g.* SIFT[11]) or learning (*e.g.* Random Ferns [12], Structured SVM [18]). The reliability of orientation assignment is always a concern for pose normalization [3], and it becomes even more critical in 3D [1]. The learning based methods just absorb the complexity into the classification problem, which works well in case of a restricted set of possible rotations but is inefficient otherwise. Especially when going from 2D to 3D, sampling all possible rotations becomes unattractive. While sampling one object under 2D rotations in 10-degree steps leads to 36 samples, it leads to approximately 15000 samples for full 3D rotations, as three angles are required to determine a 3D pose.

In this paper, we show that a powerful tool can be created by combining filters based on the 2D/3D *harmonic basis* with a kernel weighted model. In the presented approach, the complexity caused by the rotations is absorbed by taking advantage of the property of the harmonic basis, which is related to *the irreducible representation of rotations* in the group representation theory [10]. The proposed kernel weighted model assembles the filters built with the harmonic basis in a flexible way.

Our basic framework follows the equivariant filter [14], which uses two layers of filtering: one layer for description

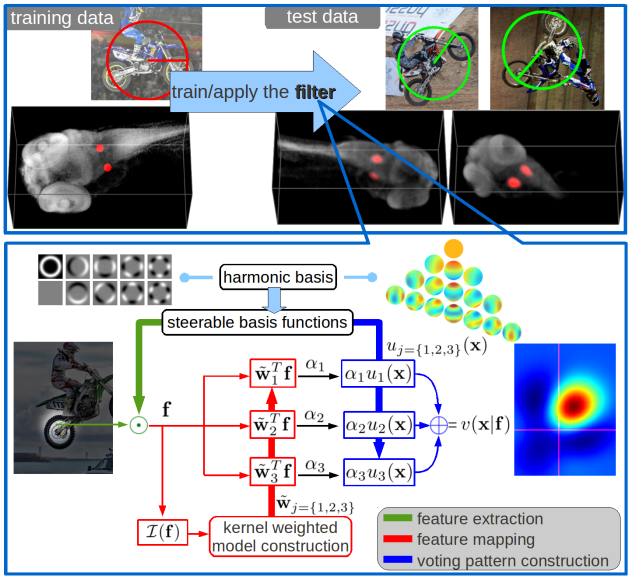

Figure 1. Overview of the presented approach. The bottom graph illustrates the work-flow on a single patch, which finally contributes a steered voting for the object center (assuming $j_{\max} = 3$). $\odot$: projection of the local patch on the basis functions, which creates the local feature vector $\mathbf{f}$ (Sec. 3.2). $\mathcal{I}$ makes rotation-invariant features from $\mathbf{f}$ (Sec. 4.1). $\tilde{\mathbf{w}}$: the local model given by the *kernel weighted mapping* (Sec. 4.2). $\alpha_{j=\{1,2,3\}}$: the voting coefficients which weight and steer the voting basis $u_{j=\{1,2,3\}}(\mathbf{x})$ into the final voting pattern $v(\mathbf{x})$ (Sec. 3.3).

and one layer for voting. A trainable *mapping* can be applied to the output of the description filters to create the coefficients which then drive the voting filters. Rotation invariance is ensured by the construction of the coefficients. Fig. 1 shows an overview of the presented approach. Our work focuses on the mapping part, which essentially decides on the filtering behavior and the performance of the whole approach.

The main contribution of this paper is to introduce a kernel weighted model for the feature mapping in the equivariant filters. By using the raw features and rotation-invariant features together, we find a sound solution for constructing a nonlinear mapping under the rotation-invariant constraint. The new model provides a simple and reliable learning mechanism for the filter framework, thus significantly improves its performance on challenging tasks.

The issue of rotation exists both in 2D and 3D and both settings share many properties. The presented approach can be applied to both of them. Since the 2D case is more intuitive and easier to understand we base our explanation mainly on the 2D case, but the generalization from 2D to 3D is quite straightforward. The 1D angular basis on the unit circle just needs to be replaced by a spherical basis on the unit sphere.

2. Related work

In this paper, the key element to achieve rotation invariance is the 2D/3D harmonic basis [21] with its self-steerability [2, 5]. A 2D equivariant filter, the *holomorphic filter*, was first proposed for low-level vision tasks in [14], using group integration [15] as a constructing tool. Then the harmonic basis in spherical coordinates helps to extend this tool to 3D problems as an efficient feature detector [13]. While the feature design and rotation properties have been analyzed in depth, those equivariant filters base their nonlinear behavior on feature coupling, which in fact is just a linear model on coupled features. More challenging tasks demand a more flexible model for the mapping part, which is the main motivation behind this work.

The detection problem is a topic covered by intensive research in computer vision field. Our basic filter framework can be related to the generalized Hough transform [9]. It can also be considered as a two-layer convolutional network [8], in which the architecture and filters are specially designed for the rotation invariance.

The 2D categorical object detection with in-plane rotations is a possible application of our approach, but we focus more on the landmark detection in *microscopic volumetric data*. Our high-level application is similar to the work presented in [7], but in a more challenging 3D setting, where the recorded objects have undetermined poses and large deformations. The landmark detection hence becomes a key element in the whole pipeline, for providing reliable point correspondences and a high-quality initialization for the elastic registration.

Figure 2. Complex Gaussian derivatives, which has the Fourier basis as the angular part (only showing the real part).

3. Equivariant filter revisited

In this section, we will use the property of Fourier basis and a voting scheme to derive the equivariant filters.

3.1. Invariance and equivariance

In detection tasks, the rotation *invariance* is defined w.r.t. to the object coordinate system. In the image coordinate system, we can abstract the detection process as a transform H on the input image I. When a rotation $\mathfrak{g}$ acts on the image, the detection behavior we actually need is $H(\mathfrak{g}I) = \mathfrak{g}H(I)$[1], *i.e.*, the output of the transform rotates together with the input image. This relative invariance is called *equivariance* [14].

3.2. Fourier basis and self-steerability

To investigate a function under a rotation, without loss of generality, the origin can always be defined at the rotation center. For analyzing 2D functions, the ideal basis should take a separable form $\Psi(r,\varphi) = R(r)\Phi(\varphi)$, where (r,φ) are polar coordinates. In practice, while the radial part $R(r)$ can be defined in many ways, the optimal choice for the angular part is the Fourier basis $\Phi_m(\varphi) = \frac{1}{\sqrt{2\pi}} e^{\mathbf{i}m\varphi}$, where m is an integer [21]. It is optimal because it can be steered by a factor $e^{-\mathbf{i}m\beta_\mathfrak{g}}$, as $\Phi_m(\varphi - \beta_\mathfrak{g}) = e^{-\mathbf{i}m\beta_\mathfrak{g}}\Phi_m(\varphi)$, and the functions like $e^{\mathbf{i}m\beta}$ give the irreducible representations of the 2D rotation group, in the group representation theory [10]. These Fourier basis functions form harmonics on the circle.

With an arbitrary radial profile $R(r)$, a basis function like $u = R(r)e^{\mathbf{i}m\varphi}$ has the property as $u(r,\varphi-\beta) = e^{-\mathbf{i}m\beta}u(r,\varphi)$. This property is called self-steerability [5], as the function itself can be steered to any orientation by a simple multiplication. Considering the filtering (convolution) with such a function, $H(I) = u * I$, on a rotated image, we have

$$H(I(r,\varphi-\beta)) = e^{\mathbf{i}m\beta}[H(I)](r,\varphi-\beta) \quad . \qquad (1)$$

We call the output function $H(I)$ *covariant* w.r.t. the image rotations, and refer to m as the *rotation order* for both

[1]The rotation action $\mathfrak{g}$ means "rotating the field while keeping its physical meaning". This is trivial for scalar fields like images, but not for high-order fields, *e.g.* the gradient field $\mathbf{d}(\mathbf{x})$ transforms as $[\mathfrak{g}\mathbf{d}](\mathbf{x}) = \mathbf{U}_\mathfrak{g}\,\mathbf{d}(\mathbf{U}_\mathfrak{g}^T\mathbf{x})$, where $\mathbf{U}_\mathfrak{g}$ is the rotation matrix in Cartesian coordinates.

$H(I)$ and u. The rotation order of the filter output can be manipulated by either multiplications or convolutions, *e.g.* if H_j and H_k come from the filters in the form of $R(r)e^{\mathbf{i}m\varphi}$, the rotation orders of $H_k(H_j(I))$ and $H_k(I)H_j(I)$ are both $m_j + m_k$. Then it is easy to find out that the condition to fulfill equivariance in such a compound filter is $m_j + m_k = 0$.

As an example, the basis functions used in the holomorphic filter [14] are the complex Gaussian derivatives (shown in Fig.2). They can be efficiently computed with finite differences.

3.3. Dense equivariant voting

For the detection task, we consider a voting process from a dense feature map $\mathbf{F} : \mathbb{R}^2 \to \mathbb{C}^{d_{\max}}; \mathbf{x} \mapsto \mathbf{f}$ (where $d_{\max}$ indicates the feature dimension), as

$$S(\mathbf{x}) = \int_{\mathbb{R}^2} v(\mathbf{x} - \mathbf{y}|\mathbf{F}(\mathbf{y}))d\mathbf{y} \quad , \tag{2}$$

where S is the detection score, and the *voting pattern* $v(\mathbf{x}|\mathbf{f})$ represents the vote to the relative position $\mathbf{x}$ given the local feature $\mathbf{f}$. To make the voting pattern easy to learn, it is parameterized as a linear combination of basis functions, with feature dependency, as $v(\mathbf{x}|\mathbf{f}) = \sum_{j=1}^{j_{\max}} A_j(\mathbf{f})u_j(\mathbf{x})$. $A_j : \mathbb{C}^{d_{\max}} \to \mathbb{C}; \mathbf{f} \mapsto \alpha_j$ is the *feature mapping* to learn. See the illustration in Fig.1. Inserting A_j into Eq.(2), we get

$$S(\mathbf{x}) = \int_{\mathbb{R}^2} \sum_j A_j(\mathbf{F}(\mathbf{y}))u_j(\mathbf{x} - \mathbf{y})d\mathbf{y} = \sum_j \tilde{A}_j * u_j \ , \tag{3}$$

where $\tilde{A}_j : \mathbb{R}^2 \to \mathbb{C}$ is introduced by $\tilde{A}_j(\mathbf{x}) = A_j(\mathbf{F}(\mathbf{x}))$. Thus, after the basis is selected, the voting behavior is completely decided by the mappings $A_{j=\{1,\dots,j_{\max}\}}$.

To achieve the equivariance, we make use of the covariant features from Sec.3.2, which are computed by the self-steerable basis with small support range, and use similar basis functions with larger support as u_j. Then with a proper model for A_j, we can manipulate the rotation order of each term in Eq.(3), by the method explained in Sec.3.2.

When the local features are also created by linear filters, we must create some nonlinearity by the feature mapping, otherwise the whole approach will collapse into a single linear filter. In [14], A_j is a weighted sum of coupled features, namely $A_j(\mathbf{f}) = \sum_{m_j+m_k+m_l=0} \gamma_{jkl}(f_l f_k)$, where $\{m_j; m_k, m_l\}$ indicate the rotation orders for the voting basis function u_j and the features. The coefficients γ are the parameters to learn. The summing-to-zero constraint guarantees the equivariance, but this model has limited capacity to approximate the optimal nonlinear mapping.

3.4. From 2D to 3D

The analogous tools for the 3D analysis in the spherical coordinates (r, θ, φ) are not as well-known as their 2D counterparts. First of all, we need the harmonic basis defined on the spheres, which is called *Spherical Harmonics*. It is intuitive to consider them as vector-valued functions $\mathbf{Y}^\ell : S_2 \to \mathbb{C}^{2\ell+1}$. Accordingly, the steering factors (the counterpart of $e^{\mathbf{i}m\varphi}$) become matrices, called Wigner-D matrices [16], which give the irreducible representation of 3D rotations. They are $(2\ell+1) \times (2\ell+1)$ unitary matrices, for the ℓ^{th} order. For the convenience of analysis under 3D rotations, the spherical tensor algebra was developed [16, 13]. Note this tensor concept is not special for the 3D case, as the Fourier analysis in the polar coordinate is also related to the 2D tensors [17]. To be specific, instead of scalar values, in 3D we have $f_d \in \mathbb{C}^{2\ell_d+1}$, $\alpha_j \in \mathbb{C}^{2\ell_j+1}$ and $u_j : \mathbb{R}^3 \to \mathbb{C}^{2\ell_j+1}$. They are all spherical tensors of certain rotation orders (ℓ_d, ℓ_j). Analogously, the equivariance is achieved by making the filter output to be a zero-order tensor (scalar) field.

Similar to the complex Gaussian derivatives in the 2D case, there exists a convenient basis from the *spherical derivatives* of a 3D Gaussian, with Spherical Harmonics as its angular part. This is the only tool we actually need in the computation. For more details, we refer the readers to [13, 16].

4. Modeling the feature mapping

The variation among objects demands nonlinearity in the detection process. In those popular nonlinear models, like the kernel SVM or the codebook in Hough voting, a similarity measure (kernel) is required in the modeling process. However, in the equivariant filter framework, we need to make sure that the similarity measure respects the equivariance. For example, applying the common Euclidean distance $||\mathbf{f} - \mathbf{f}'||^2$ on two covariant feature vectors will cause trouble, because we have no simple way to describe the change of the distance when one patch rotates with respect to another.

4.1. Rotation-invariant kernel

A simple solution to this problem is to compute rotation-invariant features from the features we have, and then to perform the comparison between them. Consider a kernel function

$$\mathcal{K}_{\mathcal{I}}(\mathbf{f}, \mathbf{f}') = K(\mathcal{I}(\mathbf{f}), \mathcal{I}(\mathbf{f}')) \quad , \tag{4}$$

where $\mathcal{I}$ is an operator to create a rotation-invariant feature vector from given covariant features, K can be any standard kernel (*e.g.* a RBF kernel). As explained in Sec.3.2, from a group of covariant features, we can get rotation invariant features by coupling two features like $\overline{f_i} f_j$, when f_i and f_j have the same rotation orders. Note, coupling a feature with itself is equivalent to taking the magnitude of this feature. Similar techniques exist for 3D [21]. The kernel defined in

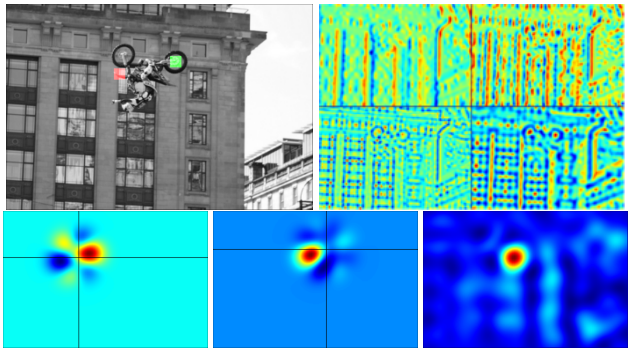

Figure 3. Detecting motorbike. Top row: the raw image and the computed local covariant features (real part of 4 features). Bottom row: The voting pattern contributed from the red/green patch and final detection output (created by Filter I in Sec.5.1).

Eq.(4) is rotation-invariant, so it is totally safe to use it as a similarity measure in the equivariant filters.

4.2. Kernel weighted mapping

We can easily have a nonlinear model for $A_j(\mathbf{f})$ with a suitable $\mathcal{K}_\mathcal{I}$. However, to get the rotation-invariant feature and similarity measure, we have lost a lot of information, including the orientation of the patches. If we model $A_j(\mathbf{f})$ without the orientation information, the voting will not be able to have any orientation selectivity.

An effective model for $A_j(\mathbf{f})$ can be created by using the rotation-invariant feature and covariant feature together. A_j should change smoothly w.r.t. the feature, thus a linear model can approximate the optimal mapping well in a local region in the feature space. Thus, we model $A_j(\mathbf{f})$ as an interpolation among liner models

$$A_j(\mathbf{f}) = [\frac{\sum_k \mathcal{K}_\mathcal{I}(\mathbf{f}_k, \mathbf{f})\mathbf{w}_{jk}}{\sum_k \mathcal{K}_\mathcal{I}(\mathbf{f}_k, \mathbf{f})}]^{\mathrm{T}}\mathbf{f} \quad , \tag{5}$$

where $^{\mathrm{T}}$ indicates a transpose, $\mathbf{f}_{k=\{1,\dots,k_{\max}\}}$ is a set of points distributed in the feature space, $\mathbf{w}_{jk} \in \mathbb{R}^{d_{\max}}$ is the local linear model at $\mathbf{f}_k$, which are the parameters to estimate.[2] The kernel is now defined as $\mathcal{K}_\mathcal{I}(\mathbf{f}_k, \mathbf{f}) = K(\mathcal{I}(\mathbf{f}_k), \mathcal{I}(\mathbf{f}))$, where $K(\mathbf{p}, \mathbf{q}) = e^{-||\mathbf{p}-\mathbf{q}||^2/2h^2}$. Eq.(5) creates a interpolated linear model for each $\mathbf{f}$. By defining $\tilde{\mathcal{K}}_k(\mathbf{x}) = \frac{\mathcal{K}_\mathcal{I}(\mathbf{f}_k, \mathbf{F}(\mathbf{x}))}{\sum_{k'} \mathcal{K}_\mathcal{I}(\mathbf{f}_{k'}, \mathbf{F}(\mathbf{x}))}$, the interpolated model for the feature vector at $\mathbf{x}$ can be written as $\tilde{\mathbf{W}}_j(\mathbf{x}) = \sum_k \tilde{\mathcal{K}}_k(\mathbf{x})\mathbf{w}_{jk}$. Inserting this into Eq.(3), we have the complete model for the proposed approach

$$S(\mathbf{x}) = \sum_j (\tilde{\mathbf{W}}_j{}^{\mathrm{T}}\mathbf{F}) * u_j = \sum_j (\sum_k \tilde{\mathcal{K}}_k \mathbf{w}_{jk}{}^{\mathrm{T}}\mathbf{F}) * u_j \ . \tag{6}$$

[2] We can use $\mathbf{w}_{jk} \in \mathbb{C}^{d_{\max}}$ in 2D problems. This benefits the feature mapping with an extra steering effect, encoded in the complex phase of $\mathbf{w}_{jk}$. However, in 3D we can not get the same benefit in such an easy way.

As shown in Fig.1, for each position, kernel weighted models are constructed based on the generated rotation-invariant features, and then applied on the raw covariant features to get the voting coefficients α_j. While the localization (in the feature space) is actually done by considering the rotation-invariant features, the covariant features still directly drive the voting basis through the constructed models. Thus the local orientation information is used to steer the voting pattern. Note, $\mathbf{w}_{jk}$ is sparse, because we need to force its d^{th} element $w_{jkd} \equiv 0$ when the rotation orders $m_j + m_d \neq 0$, for the equivariance.

Equation (6) can be reformulated as

$$S = \sum_{\substack{j,k,d \\ m_j+m_d=0}} w_{jkd}((\tilde{\mathcal{K}}_k F_d) * u_j) \ . \tag{7}$$

With preselected $\mathbf{f}_k$, the terms $(\tilde{\mathcal{K}}_k F_d) * u_j$ can be computed. So the final optimization problem can be easily solved by

$$\min_{\mathbf{W}} \int L(y(\mathbf{x}), S(\mathbf{x}|\mathbf{W}))d\mathbf{x} \quad , \tag{8}$$

where $\mathbf{W}$ denotes all parameters w_{jkd}, $y(\mathbf{x})$ is the ground-truth output (which is usually a binary image for detection problem), and L is a suitable loss function. The nonlinearity in this model is based on the localization in the feature space, all the other parts are pure linear. Thus the model enjoys the high reliability from the linear optimization. The implementation of the approach is simple. We show the training procedure in Algorithm 1. For detection we just reorder the computations into a much faster way (Fig.1), like reformulating Eq.(7) to Eq.(6) .

The remaining problem is how to select $\mathbf{f}_k$ and the kernel bandwidth. Because we actually apply the localization in the space of $\mathcal{I}(\mathbf{f})$, we only need to select $\hat{\mathbf{f}}_k = \mathcal{I}(\mathbf{f}_k)$. Although it is possible to optimize the position of $\hat{\mathbf{f}}_k$ with adaptive bandwidths, we use a simple k-means clustering on training data to find $k_{\max}$ cluster centers as $\hat{\mathbf{f}}_k$, and empirically set the kernel bandwidth h to be the half of the median value of all the nearest neighbor distances among $\hat{\mathbf{f}}_k$. In the experiments, we set $k_{\max} \leq 50$ for a fast and reliable training.

Although the model is developed to meet some special requirements, it has support from standard learning approaches. The idea using a kernel to localize the model estimation is similar to the *local linear regression* method in statistical learning [4]. Learning a nonlinear mapping by relating features to cluster centers is similar to the codebook methods, especially the *super vector coding* [23].

5. Experiments

We demonstrate our approach in both 2D and 3D, to show that the proposed kernel weighted mapping brings a

Algorithm 1 Training Algorithm

Input: image I, target output y $^{(*)}$
Output: selected $\hat{\mathbf{f}}_k$, parameter w_{jkd}

compute covariant features $\mathbf{F}(\mathbf{x})$ and $\mathcal{I}(\mathbf{F}(\mathbf{x}))$
use clustering to select $\hat{\mathbf{f}}_k$
compute weight $\tilde{\mathcal{K}}_k(\mathbf{x}) = \frac{\mathcal{K}_{\mathcal{I}}(\mathbf{f}_k, \mathbf{F}(\mathbf{x}))}{\sum_{k'} \mathcal{K}_{\mathcal{I}}(\mathbf{f}_{k'}, \mathbf{F}(\mathbf{x}))}$
for each covariant feature f_d **do**
  for each k **do**
    for each voting basis u_j with $m_j = -m_d$ **do**
      voting term $v_{jkd} = (\tilde{\mathcal{K}}_k F_d) * u_j$
    end for
  end for
end for
solve $\min_{w_{jkd}} \int \|w_{jkd} v_{jkd}(\mathbf{x}) - y(\mathbf{x})\|^2 d\mathbf{x}$

$^{(*)}$ Assume only one training image for simplicity.

large improvement over the equivariant filters in the literature, and get competitive or better performance comparing to some other state-of-the-art rotation-invariant methods. The 2D experiment is also designed to show the flexible usage of our approach and the combination with HOG feature. The 3D experiment shows the real application for which our approach is developed.

5.1. 2D rotation-invariant detection

A *Freestyle Motocross* dataset has been collected by Villamizar *et al*. [19]. They use Random Ferns classifiers in a two-step approach, with an estimation stage and a classification stage. There are two image sets, one without rotations (69 images) and one with rotations (100 images).

Implementation Instead of simple steerable filters, we use a HOG based covariant description for the natural images. Its design is also based on the polar Fourier analysis. A histogram of gradient orientations can be considered as a function of the angle, so we can use Fourier series to represent it. From the HOG represented on the Fourier basis, we construct a HOG based descriptor, which compute local covariant features containing the information similar to a 4×4 HOG window. See the supplementary material for the implementation detail. From this descriptor, we get 28-dimensional covariant features (with rotation order lower than 5). By taking the magnitude and simple couplings, we create 56-dimensional rotation-invariant features from the covariant ones. Except for this, we implement the approach as explained above. The complex-valued parameters $\mathbf{w}_{jk}$ are optimized by a simple least-square-error method. Only 7 complex Gaussian derivatives are used as the voting basis (shown in Fig.2). Fig.3 shows an example for the detection process. With this simple basis we can not synthesize a sharp voting pattern, but it already produces satisfying result.

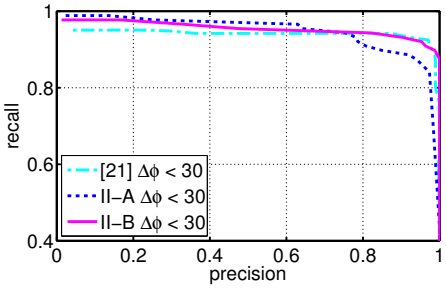

Figure 4. The performance of Filter II trained on Set A/B with 30-degree error margin for pose estimation, compared to the performance (after steered classification) reported in [19]

We create a training set A by taking the first 40 images from the first dataset (without rotations), and a training set B containing the first 40 images from the second dataset (with rotations). The test is done on the remaining 60 images from the second dataset. By using the two training sets separately (and no artificial rotations), we can show that, our approach is totally pose-independent in the training/detection procedure. Two filters are constructed: Filter I is trained by setting the target output $y(\mathbf{x})$ to a binary image with $y(\mathbf{c}) = 1$, where $\mathbf{c}$ is the object center. Filter II is trained to predict object position and pose simultaneously, by setting $y(\mathbf{c}) = e^{\mathbf{i}\phi}$, where $(\mathbf{c}, \phi_i)$ is the center and pose angle of the object (all $\phi_i = 0$ in Set A). Accordingly, we need to make the filter output to rotate like a vector field, *i.e*. $\mathfrak{g}S = e^{\mathbf{i}\beta_{\mathfrak{g}}} S(r, \varphi - \beta_{\mathfrak{g}})$, by using a different constraint for $\mathbf{w}_{jk}$ in Eq.(6): $w_{jkd} \equiv 0$ when $m_j + m_d \neq 1$. For the number of $\hat{\mathbf{f}}_k$, we report all the results with k = 50. Larger numbers do not further increase the performance, perhaps because the bottleneck shifts to other parts of the approach.

Experimental result To separate the position and pose, the objects are considered as circles. The diameter is set to the mean value of the object width and height. We use the *overlap* > 0.5 *union* criteria for position. Filter I produces the response in the real part of its output. It gets 91/90% EER (with no pose estimation) when trained on Set A/B separately. Filter II produces the detection response in the magnitude, while the phase angle indicates the estimated pose of the motorbike. It gets 89/92% EER (with a 30-degree error margin on pose estimation) when trained on Set A/B separately. In our straight implementation in Matlab, it takes about 2 seconds for the multi-scale detection at 10 scales on a desktop computer, while 85% of the time is used for feature computation. The precision-recall curves of Filter II are shown in Fig.4. Comparing to [19], their best result is 91/93% EER with 15/30-degree margin. However, these are the results after evaluating steered classifiers on a large group of candidates. In contrast, we get all the estimation in one step, and the performance is already comparable. To

Figure 5. Detection result of Filter II-B. Detections/ground-truth are drawn in green/red. Orientations are indicated by the bars from circle center.

compare our approach to the holomorphic filter [14], we use 350 dimensional coupled covariant features from the same HOG based description, to train a rotation-invariant filter like Filter I. Its performance is 70% EER, much lower than the presented approach.

Discussion The terms used in the Filter I and II are totally different, and both groups carry sufficient information. This suggest that we should try combining the terms with different orders, but that may require explicit steering. The limited angle resolution of Filter II can be explained by the fact that all the information are collected in one additive voting step, so the pose encoded in the phase angle is hard to be very accurate.

5.2. Landmark detection in 3D microscopic images

A common challenge in the biomedical research is the alignment of a new volumetric image to a standard image (atlas) by an elastic registration [20, 7]. Here the volumetric images are confocal microscopic recordings of *zebrafish embryos*. In the planned fully-automated high-content work-flow, the recorded sample will have a random orientation. Furthermore we have to deal with the morphological variations in the organism. Thus, to get a high-quality initialization and some reliable point correspondences for the elastic registration, we need the presented rotation-invariant detection method to locate a group of landmarks robustly.

In this experiment, 63 volumetric images are obtained by recordings from two sides of zebrafish embryos and an image fusion step. In the embryo, 14 anatomical landmarks are defined based on their unique and repeatable appearance (see Fig. 6). For efficiency, we take a coarse-to-fine strategy, first train and apply the detection filters on downsampled images, then the most probable global constellation of all landmarks are selected based on the individual probabilities and their pairwise distances, solved as a max-sum problem [22]. Finally, we refine the detected landmarks using the filter trained on high-resolution images, to get highly accurate localizations.

Implementation Similar to the 3D harmonic

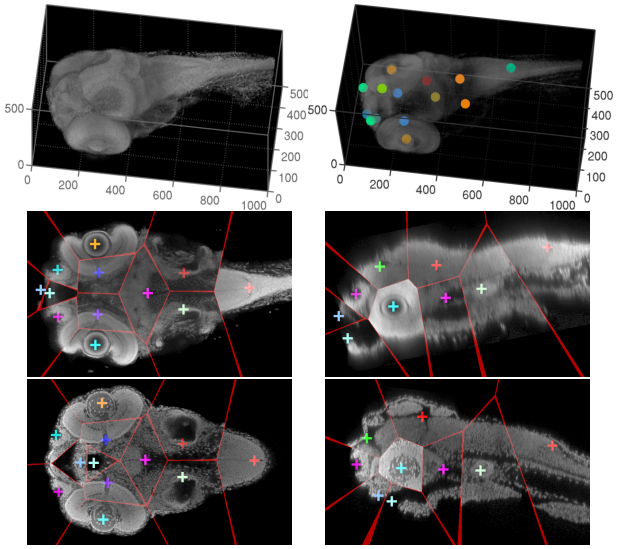

Figure 6. Top: an zebrafish embryo and the landmarks (the voxel-size is $1\mu m^3$). Middle and bottom: two embryos with their final detected landmarks shown in cropped slices, after rigid alignment based on the landmarks. The data from different experiment groups have different imaging qualities. For the large morphological variations, a rigid alignment can not unify the poses of all landmarks.

filter [13], we use the spherical Gaussian derivatives (SGD) as the local descriptor and the voting basis. The SGD is denoted as $\nabla_d^u G_\sigma$, where u and d indicate its derivative order. On a volumetric data V, the features compute as $F_{ud} = \nabla_d^u (G_{\sigma_d} * V)$, where σ_d is the selected scale for local description and the feature F_{ud} is a spherical tensor field of order $(u-d)$. The rotation-invariant features are computed by taking the L2 norm of each derivative feature. Then $\hat{\mathbf{f}}_k$ are found by a k-means clustering with $k_{\max}$=30. From the mapping as Eq.(5), the $A_j(\mathbf{F}(\mathbf{x}))$ will also be spherical tensor fields, so they can convolve with SGDs (with scale σ_v) of the same order to fulfill the equivariant voting. More implementation details are given in the supplementary material. Our implementation is publicly available on our website[3].

For the first step, the symmetric landmarks (*e.g.* left/right eye) are considered as the same class. We work at the downsampled images with a voxel-size of $(6\mu m)^3$. The parameter settings $u+d \leq 6, \sigma_d = 20\mu m, \sigma_v = 40\mu m$ are used for all landmarks. As a result, we obtain 15 covariant features and 15 invariant features. We manually labeled all landmarks in 7 embryos for training, and further manually labeled 6 classes of landmarks in other 56 images, for a quantitative evaluation.

Reference methods Here we compare our approach to the 3D SIFT [1] and the harmonic filter. Our SIFT based ex-

[3] http://lmb.informatik.uni-freiburg.de/people/liu/landmark3d

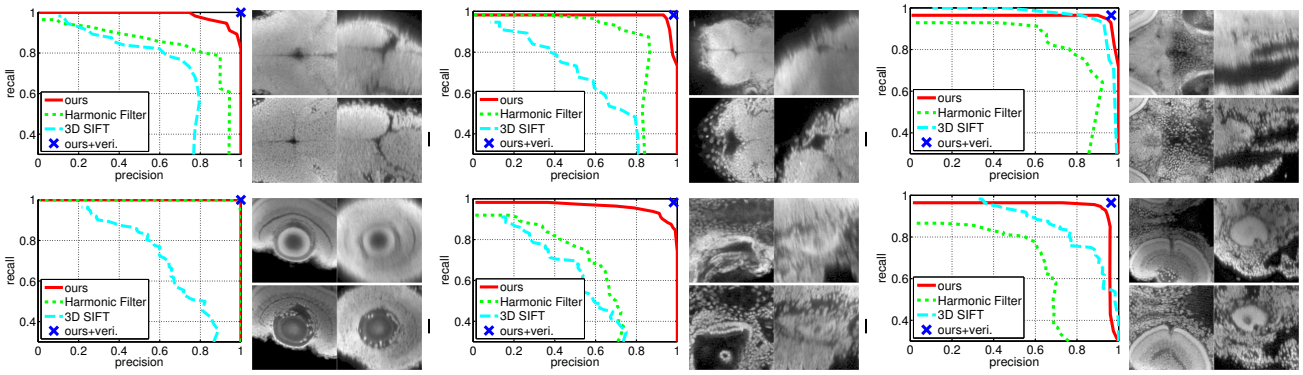

Figure 7. The precision-recall curves for detections of 6 classes of landmarks. All the 6 filters are trained with the same parameters. For each landmark, we show the precision-recall curve, two examples from the X-Y/X-Z plane views in the aligned embryos, and a small black bar indicating the scale of the error margin for recall (top row: 30,30,45, bottom row: 30,30,45(μm)). The landmarks are: mid-hindbrain boundary, 2nd ventricle, notochord tip, eye, ear, and optical nerve exit point. The presented approach always produces the best result for all landmarks.

periment goes as following: the salient points are extracted in a high density based on the determinant of Hessian matrix. It extracts 5000∼8000 points in each image, covering the landmarks well. Then 3D SIFT features are computed following [1][4]. The positive samples are extracted from the 7 training images (on all salient points within $24\mu m$ range to the manually labeled point). With such few samples, the nearest-neighbor classifier outperforms the SVM classifier in a classification test. So for each salient point, we compute the detection probability based on the smallest Chi-Square distance between the feature vector(s) on the point and the feature vectors from the training samples. The harmonic filter is implemented following the reference, in which we get 64 coupled features (in the orders lower than 6) to drive the 3D voting.

Experimental result The evaluations and examples of the detections are shown in Fig.7 and Fig.8. The detection candidates from the filter approaches are the local maximums. As the SIFT is computed with a high density and multiple pose selections, the harmonic Filter is not always performing better than SIFT, but our approach always give the best result on all evaluated landmarks. The running time for the filters (implemented in C) on a 3.2GHz×4 CPU, is: 6s for local feature computation including kernel evaluations, and additionally 4s for each landmark class, while the harmonic filter needs about 25s for each class. The reason is that the coupling of spherical tensors is not cheap, and we avoid this by using kernel based nonlinearity. The 3D SIFT is also expensive, the feature computation alone needs about 20s for every 1000 points. Further qualitative results are: after the fast max-

[4]The Hessian is computed over 5 scales in $6\mu m \sim 24\mu m$, only to capture the landmarks in different scales. The 3D SIFT is computed at a single scale on the downsampled image with bin size = $36\mu m$ and $4\times4\times4$ spatial bins. Each point can have multiple descriptions according to several competing poses.

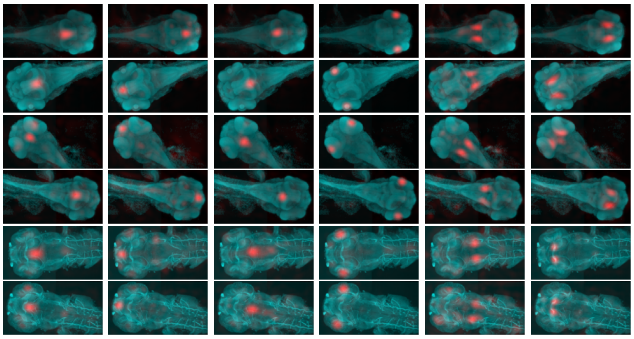

Figure 8. Maximum intensity projection of the filter output (red) overlaid on embryos (cyan), for the 6 evaluated landmarks. The bottom two rows show the results on anti-AcTub (acetylated tubulin) immunostaining data, by using the same parameters in filter training.

sum verification, all landmarks are found in about $50\mu m$ range. By running the high-resolution ($(1.5\mu m)^3$) filters in the neighborhood of each landmark, all the landmarks are refined to a high accuracy (see Fig.9). The final refined landmarks provide a group of reliable point correspondences distributed in the embryo, which are interpolated into a high-quality initialization for the elastic registration, making the registration faster and more accurate. The wide-applicability of our method has also be checked. It has been tested on zebrafish with different staining techniques. Some qualitative result is shown in Fig.8.

Discussion We can see that the harmonic filter is perfect for simple features like the eyes, but is not sufficient for complex structures. For the 3D SIFT, by a pose normalization, its output is not continuous on the underlying data, and hence it might need more training data to well describe a class. Our approach is continuous on the underlying data, making the classification simpler, and has good description

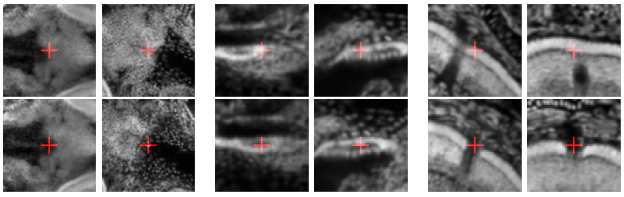

Figure 9. The effect of refinement on landmarks. Top: before refinement. Bottom: after refinement.

ability from the voting mechanism.

In [13] an ISM based 3D voting method has been designed, and showed to be less effective than the harmonic filter. The ISM based method reported in [6] is developed for the 3D shapes represented by surfaces. They use interest point detection and assign an unique orientation to each point. These could be error-prone in volumetric data. The low reliability of the interest point detection is the main reason that we prefer an approach based on dense features. After all, our filter framework is simple in the implementation, which just need two rounds of fast convolutions with a voxel-wise feature mapping between them.

6. Conclusion

Based on the fundamental theory about harmonic basis and rotations, we present a practical way to build a flexible nonlinear model under the equivariance constraint, developing the classical equivariant filters to more powerful tools. The presented approach guarantees rotation invariance, a well performing nonlinear model and a high computational efficiency. It produces competitive rotation-invariant detection performance in 2D images, and works very well on the rotation-invariant landmark detection task in 3D microscopic volumetric images.

Acknowledgement

This study was supported by the Excellence Initiative of the German Federal and State Governments (EXC 294).

References

[1] S. Allaire, J. Kim, S. Breen, D. Jaffray, and V. Pekar. Full orientation invariance and improved feature selectivity of 3D SIFT with application to medical image analysis. In *CVPR Workshop 2008*. 1, 6, 7

[2] W. Freeman and E. Adelson. The design and use of steerable filters. *IEEE Trans. Pattern Anal. Mach. Intell.*, 13(9):891–906, 1991. 2

[3] S. Gauglitz. Improving keypoint orientation assignment. In *BMVC 2011*. 1

[4] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning*. Springer, 2001. 4

[5] G. Jacovitti and A. Neri. Multiresolution circular harmonic decomposition. *IEEE Trans. Signal Process.*, 48(11):3242–3247, 2000. 2

[6] J. Knopp, M. Prasad, G. Willems, R. Timofte, and L. Van Gool. Hough transform and 3D SURF for robust three dimensional classification. In *ECCV 2010*. 8

[7] U. Kurkure, Y. Le, N. Paragios, J. Carson, T. Ju, and I. Kakadiaris. Landmark/image-based deformable registration of gene expression data. In *CVPR 2011*. 2, 6

[8] S. Lawrence, C. Giles, A. Tsoi, and A. Back. Face recognition: A convolutional neural-network approach. *IEEE Trans. Neural Netw.*, 8(1):98–113, 1997. 2

[9] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *Int. J. Comput. Vision*, 77(1):259–289, 2008. 2

[10] R. Lenz. *Group theoretical methods in image processing*. Springer, 1990. 1, 2

[11] D. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004. 1

[12] M. Özuysal, M. Calonder, V. Lepetit, and P. Fua. Fast keypoint recognition using random ferns. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(3):448–461, 2010. 1

[13] M. Reisert and H. Burkhardt. Harmonic filters for generic feature detection in 3D. In *DAGM 2009*. 2, 3, 6, 8

[14] M. Reisert and H. Burkhardt. Equivariant holomorphic filters for contour denoising and rapid object detection. *IEEE Trans. Image Process.*, 17(2):190–203, 2008. 1, 2, 3, 6

[15] O. Ronneberger, H. Burkhardt, and E. Schultz. 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 *ICPR 2002*. 2

[16] M. Rose. *Elementary theory of angular momentum*. Dover Publications, 1995. 3

[17] T. Schultz, J. Weickert, and H. Seidel. A higher-order structure tensor. In *Visualization and Processing of Tensor Fields*, pages 263–279. Springer, 2009. 3

[18] A. Vedaldi, M. Blaschko, and A. Zisserman. Learning equivariant structured output svm regressors. In *ICCV 2011*. 1

[19] M. Villamizar, F. Moreno-Noguer, J. Andrade-Cetto, and A. Sanfeliu. Efficient rotation invariant object detection using boosted random ferns. In *CVPR 2010*. 5

[20] T. Walter, D. Shattuck, R. Baldock, M. Bastin, A. Carpenter, S. Duce, J. Ellenberg, A. Fraser, N. Hamilton, S. Pieper, et al. Visualization of image data from cells to organisms. *Nat. Methods*, 7:S26–S41, 2010. 6

[21] Q. Wang, O. Ronneberger, and H. Burkhardt. Rotational invariance based on fourier analysis in polar and spherical coordinates. *IEEE Trans. Pattern Anal. Mach. Intell.*, 31(9):1715–1722, 2009. 2, 3

[22] T. Werner. A linear programming approach to max-sum problem: A review. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(7):1165–1179, 2007. 6

[23] X. Zhou, K. Yu, T. Zhang, and T. Huang. Image classification using super-vector coding of local image descriptors. In *ECCV 2010*. 4