Learning Rotation-Aware Features:
From Invariant Priors to Equivariant Descriptors

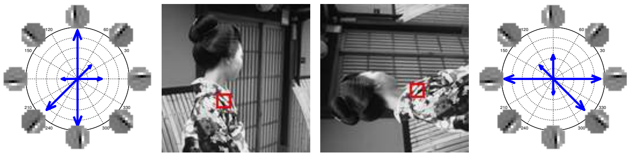
Uwe Schmidt   Stefan Roth
Department of Computer Science, TU Darmstadt

Abstract

Identifying suitable image features is a central challenge in computer vision, ranging from representations for low-level to high-level vision. Due to the difficulty of this task, techniques for learning features directly from example data have recently gained attention. Despite significant benefits, these learned features often have many fewer of the desired invariances or equivariances than their hand-crafted counterparts. While translation in-/equivariance has been addressed, the issue of learning rotation-invariant or equivariant representations is hardly explored. In this paper we describe a general framework for incorporating invariance to linear image transformations into product models for feature learning. A particular benefit is that our approach induces transformation-aware feature learning, i.e. *it yields features that have a notion with which specific image transformation they are used. We focus our study on rotation in-/equivariance and show the advantages of our approach in learning rotation-invariant image priors and in building rotation-equivariant and invariant descriptors of learned features, which result in state-of-the-art performance for rotation-invariant object detection.*

1. Introduction

Despite having been extensively studied, the problem of identifying suitable feature representations for images remains a key challenge in computer vision today. This is true in a diverse set of areas ranging from high-level tasks, such as object classification and detection [2, 6, 17, 20] all the way down to problems as low-level as image restoration [22, 26, 27]. Due to the diversity of areas in which feature representations are crucial, the characteristics of what makes a good feature representation also differ quite widely. One common thread in the recent literature is the increase in methods that learn suitable feature representations for specific tasks from example data [*e.g.*, 12, 21]. One motivation for this is that devising well-performing feature representations manually is a complex process, since it may not be very intuitive which aspects of a feature representation make it perform well in practice [6, 20]. Another is that customizing the feature representation to the task at hand may have significant benefits in practice.



(a)   (b) $E = 490654$ (c) $E = 488655$ (d)

Figure 1. **Rotation invariance and equivariance.** *(b,c)* Current learned image priors (here [23]) are not rotation invariant and assign different energies E depending on the image orientation. We address this issue by learning image models with built-in invariance to certain linear transformations, such as rotations. Furthermore, our approach induces transformation-aware features that allow to derive equivariant feature representations *(a,d)*, *i.e.* it is possible to predict how a transformation of the input transforms the feature activations: The feature response *(a)* for 8 orientations of a learned feature for the image patch marked in red already tells us the transformed feature response *(d)* when the input is rotated *(c)*.

An important shortcoming of many feature learning approaches is that they do not have the same desirable invariances or equivariances with respect to transformations of the input as do traditional hand-crafted representations. In various use cases of object detection it is, for example, reasonable to expect that an object can be detected no matter its orientation in the image. Hand-crafted feature representations [*e.g.*, 24] facilitate this by using a rotation-equivariant[1] feature representation (see Fig. 1(a,d) for an illustration). Feature learning techniques for recognition, on the other hand, have mainly focused on addressing translation in-/equivariance by using convolutional learning architectures [18, 21], or on local rotation invariance [12].

Similarly, it is desirable that an image restoration algorithm is equivariant to certain input transformations: If the input image was shifted or rotated, one would expect that the restored image is shifted or rotated the same way, but otherwise unchanged. Yet while traditional regularizers, such as total variation, are rotation invariant leading to equi-

[1] Formally, a function f is equivariant to a class of transformations $\mathcal{T}$, if for all transformations $\mathbf{T} \in \mathcal{T}$ of the input $\mathbf{x}$, we can predict a corresponding transformation $\mathbf{T}'$ of its output, *i.e.* $f(\mathbf{T}\mathbf{x}) = \mathbf{T}' f(\mathbf{x})$. Moreover, f is invariant to transformations $\mathcal{T}$ if $f(\mathbf{T}\mathbf{x}) = f(\mathbf{x})$ for all $\mathbf{T} \in \mathcal{T}$.

In Proc. of the *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, Providence, RI, June 2012.   doi:10.1109/CVPR.2012.6247909

variant denoising, image models based on learned features are typically not (see Fig. 1(b,c)).

Here we aim to address invariance and equivariance to *linear image transformations beyond translation*. Although not limited to this setting, we particularly focus on *rotations*, since for many applications this is the most important transformation in-/equivariance beyond translation. We first propose a general framework for incorporating transformation invariances into product models for feature learning. We then demonstrate its application by extending Fields of Experts (FoE) image priors [22, 23] to *R-FoEs*, which are invariant to 90° rotations (or multiples thereof) in addition to being translation invariant. Moreover, we show how the methodology can be used to extend convolutional Restricted Boltzmann Machines (C-RBMs) [18, 21] to *RC-RBMs*, which are translation and rotation invariant.

While invariances can be learned directly from training data, this may require inordinate amounts of data. But even if the training data was sufficient to learn invariances without any model provisions, then some of the learned features would be transformed versions of others to account for this invariance [26]. One important shortcoming of this approach is that it is unclear how the different features are related in terms of the image transformations between them. This makes it difficult to build in-/equivariant feature descriptors for invariant object recognition or detection from them. A key property of our approach is that it allows to induce *transformation-aware* features, *i.e.* we can predict how the feature activations change as the input image is being transformed, which we further exploit to define a *rotation-equivariant feature descriptor*, called *EHOF*, based on features learned with an RC-RBM. We also extend EHOF to a fully *rotation-invariant descriptor*, *IHOF*.

We demonstrate the benefits of our approach in two applications. First, we show how learning a rotation-invariant image prior benefits equivariant image restoration. Second, we apply the learned features as well as the proposed rotation in-/equivariant descriptors in the context of object recognition and detection. We test our approach on two challenging data sets for rotation-invariant classification and detection, and in each case outperform state-of-the-art methods from the recent literature.

2. Product Models & Linear Transformations

Many probabilistic models of images and other dense scene representations, such as depth and motion, can be seen as product models in which each factor models a specific property of the data that is extracted using a linear feature transform. If we denote the vectorized image as $\mathbf{x} \in \mathbb{R}^n$ and $\mathcal{F} = \{\mathbf{F}_{(i)} \in \mathbb{R}^{m_i \times n} | i = 1, \ldots\}$ a set of linear feature transformations, we can write an abstract product model as

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{i=1}^{|\mathcal{F}|} \phi_i\big(\mathbf{F}_{(i)}\mathbf{x}; \theta_i\big). \tag{1}$$

Here, the ϕ_i are the individual factors (potentials) that model the result of each linear feature transform $\mathbf{F}_{(i)}$ based on parameters θ_i, and Z is a normalization constant making $p(\mathbf{x})$ a proper density (from now on omitted for brevity).

Markov random fields (MRFs) [19] can be interpreted as one instance of such a product model by defining "cropping" matrices: $\mathbf{C}_{(i)}$ crops out a single pixel i from $\mathbf{x}$ such that $\mathbf{C}_{(i)}\mathbf{x} = x_i$, and $\mathbf{C}_{(k,l)}$ crops out two neighboring pixels k and l such that $\mathbf{C}_{(k,l)}\mathbf{x} = (x_k, x_l)^{\mathrm{T}}$. Then

$$p_{\text{MRF}}(\mathbf{x}) \propto \prod_{i=1}^{n} \phi_i\big(\mathbf{C}_{(i)}\mathbf{x}; \theta_i\big) \prod_{(k,l)\in E} \phi_{kl}\big(\mathbf{C}_{(k,l)}\mathbf{x}; \theta_{kl}\big) \tag{2}$$

denotes a standard pairwise MRF, where ϕ_i are the unaries, and ϕ_{kl} the pairwise terms for each edge $(k,l) \in E$.

If the feature transformation matrices $\mathbf{F}_{(i)}$ are filters (row vectors), *i.e.* $\mathbf{F}_{(i)} = \mathbf{J}_i^{\mathrm{T}} \in \mathbb{R}^{1\times n}$, that project into a 1D subspace, then we also notice that Eq. (1) is a Product of Experts (PoE) with linear feature transforms [10]:

$$p_{\text{PoE}}(\mathbf{x}) \propto \prod_{i=1}^{|\mathcal{F}|} \phi_i\big(\mathbf{J}_i^{\mathrm{T}}\mathbf{x}; \theta_i\big). \tag{3}$$

We note that such PoEs with linear experts directly generalize PCA, ICA, as well as Restricted Boltzmann Machines (RBMs) [10] (see also Sec. 4).

Despite the notational similarity, there are two key differences between the pairwise MRF in Eq. (2) and linear PoEs or RBMs as in Eq. (3). Pairwise MRFs have fixed feature transformations, whereas they are learned from data in case of linear PoEs and RBMs. Moreover, the primary goal of MRFs is usually modeling the prior distribution $p(\mathbf{x})$ itself, *e.g.*, for regularization, but linear PoE models and RBMs often use the probabilistic model only as a tool for learning the features $\mathbf{F}_{(i)}$ for use in other tasks such as recognition.

2.1. Integrating transformation invariance

To see how product models can be made transformation invariant, it is useful to study the MRF model from Eq. (2) in more detail. MRFs in vision are typically made translation invariant by ensuring that the unary terms and the pairwise terms are the same everywhere in the image (*i.e.* ϕ_i and θ_i do not depend on i, and ϕ_{kl} and θ_{kl} only depend on the relative position of pixels k and l). In other words, translation invariance is achieved by taking a product of the same unary and pairwise terms over all possible pixel locations. High-order MRFs [22, 27] and convolutional RBMs [18, 21] do so analogously (*cf*. Secs. 3 & 4).

We here generalize this concept to *arbitrary linear image transformations*. Given a finite set of linear image transformations $\mathcal{T} = \{\mathbf{T}_{(j)} | j = 1, \ldots\}$ of one or more types, we define a transformation-invariant product model w.r.t. $\mathcal{T}$ as

$$p_{\mathcal{T}}(\mathbf{x}) \propto \prod_{j=1}^{|\mathcal{T}|} \prod_{i=1}^{|\mathcal{F}|} \phi_i\big(\mathbf{F}_{(i)} \mathbf{T}_{(j)} \mathbf{x}; \theta_i\big). \qquad (4)$$

To achieve invariance, it is important that both the factor ϕ_i and its parameters θ_i do not depend on $\mathbf{T}_{(j)}$. However, due to the necessarily finite representation of images and the finite transformation class $\mathcal{T}$, such invariances in most cases only hold approximately.

While Eq. (4) may seem like an innocuous change over Eq. (1), it has several important properties: *(1)* the framework generalizes a known mechanism for translation invariance [21, 27] to arbitrary finite sets of linear transformations $\mathcal{T}$, including rotations; *(2)* unlike other attempts to achieve simultaneous invariance to several transformations, *e.g.*, translation and rotation [13], we treat all transformations equally, and do not introduce additional latent variables [8]; *(3)* the formulation is a special case of the generic product model in Eq. (1), in which the factors model the responses to the compound linear transformation $\mathbf{F}_{(i)}\mathbf{T}_{(j)}$, and the type and parameters of the factors are shared between all possible transformations in $\mathcal{T}$; *(4)* transformation invariance can be added to a wide range of product models without substantial modifications to their algorithmic backbone for learning and inference; *(5)* since the factors and their parameters are shared between all transformations, this leads to parsimonious representations with comparatively few parameters that may also be easier to interpret; and finally, *(6)* this will later allow us to construct equivariant descriptors with learned features, which in turn facilitate rotation-invariant object detection.

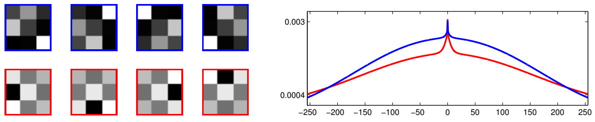
3. Learning Rotation-Invariant Image Priors

Many problems in low-level vision require prior knowledge. In image restoration tasks, such as denoising, deblurring, or inpainting, image priors are crucial for recovering a plausible image from noisy, blurred, or incomplete inputs. While traditionally pairwise MRFs (Eq. (2)) have been the prevalent probabilistic prior model of images [19], recent years have seen an increased adoption of learned high-order priors [22, 27]. They not only benefit from modeling complex image structure in large patches (cliques), but also from learning the model parameters from training data.

It is important to note that several popular image priors can be seen as special cases of our transformation-invariant learning framework. To that end we define a set of "convolutional" transformations as

$$\mathcal{T}_{\text{C}} = \big\{\mathbf{C} \cdot \mathbf{S}_{(k,l)} \big| k = 1, \ldots, r,\ l = 1, \ldots, c\big\}, \qquad (5)$$

where the linear transformation $\mathbf{S}_{(k,l)}$ translates the image such that pixel (k, l) is at the origin, and $\mathbf{C}$ crops a fixed size image patch (*e.g.*, 3×3 pixels) around the origin. Here, $\mathbf{S}_{(k,l)}$ achieves translation invariance, while $\mathbf{C}$ ensures that the model complexity is independent of the image size.



(a) 2 features $\mathbf{J}_i \times 4$ rotations

(b) 2 factors (experts) ϕ_i

Figure 2. **Learned R-FoE model with** 2 **experts and** 4 **rotations.** The features and corresponding expert shapes are color-matched.

It is now quite straightforward to see that the FRAME model [27] and the Field of Experts (FoE) [22] are special cases of Eq. (4) with $\mathcal{T} = \mathcal{T}_{\text{C}}$. In FRAME, the feature transformations $\mathbf{F}_{(i)}$ are hand-chosen filters and the factors ϕ_i are learned from data. The FoE additionally learns the linear features $\mathbf{F}_{(i)} = \mathbf{J}_i^{\text{T}}$ from data.

However, the FoE is not explicitly designed to incorporate any invariances beyond image translations. Since the features are unconstrained during learning, it is for example not guaranteed that horizontal and vertical image structure is modeled equally, which can be argued is a desirable property of an image prior: The quality of a restored image should be the same, regardless of whether the image was restored in portrait or landscape orientation. As Fig. 1 shows, rotating an image by 90° may already substantially change the energy of the image under the non-invariant prior.

We propose to additionally impose the desired invariance to rotations into the model, and define the transformation set as

$$\mathcal{T}_{\text{RC}} = \Big\{\mathbf{R}_{(\omega)} \cdot \mathbf{C} \cdot \mathbf{S}_{(k,l)} \Big|_{k=1,\ldots,r,\ l=1,\ldots,c}^{\omega \in \Omega,}\Big\}. \qquad (6)$$

Here $\mathbf{R}_{(\omega)}$ performs an image rotation of the cropped patch by angle ω, and $\mathbf{S}_{(k,l)}$ and $\mathbf{C}$ are defined as before. Using the transformation set $\mathcal{T}_{\text{RC}}$ – here with 90° rotation increments, *i.e.* $\Omega = \{0°, 90°, 180°, 270°\}$ – we train a rotation-invariant FoE image prior (*R-FoE*)

$$p_{\text{R-FoE}}(\mathbf{x}) \propto \prod_{\omega \in \Omega} \prod_{(k,l)} \prod_{i=1}^{|\mathcal{F}|} \phi_i\big(\mathbf{J}_i^{\text{T}} \cdot \mathbf{R}_{(\omega)} \cdot \mathbf{C} \cdot \mathbf{S}_{(k,l)} \mathbf{x}; \theta_i\big) \quad (7)$$

with $|\mathcal{F}| = 2$ features (filters) $\mathbf{J}_i$ defined on 3×3 patches. The factors (experts) ϕ_i are modeled as Gaussian scale mixtures, and learning is done using contrastive divergence [10] and Gibbs sampling. Fig. 2 shows the 2 learned features with their 4 implicitly induced rotations (as an effect of the $\mathbf{R}_{(\omega)}$), and the corresponding experts. Note that the 4 different rotations share the same expert (and parameters), which ensures that the learned model is fully invariant to image rotations in 90° increments. While finer-grained invariance with smaller angular increments is in principle possible, this necessitates larger filters, which remains challenging due to filters and experts being learned simultaneously.

4. Learning Rotation-Aware Image Features

Besides transformation-invariant image models, our second main goal is to learn transformation-aware image features that will later allow us to derive transformation in-/equivariant feature descriptors for object detection. A widely used model for feature learning is the Restricted Boltzmann Machine (RBM) [10]. For a binary image $\mathbf{x}$ and a set of binary hidden variables $\mathbf{h} \in \{0,1\}^K$ it is defined as

$$p_{\text{RBM}}(\mathbf{x},\mathbf{h}) \propto \exp\left(\mathbf{c}^{\text{T}}\mathbf{x}\right) \prod_{i=1}^{K} \exp\Big(h_i\big(\mathbf{w}_i^{\text{T}}\mathbf{x} + b_i\big)\Big). \tag{8}$$

If the image $\mathbf{x}$ is real-valued, a Gaussian RBM is used instead and defined as

$$p_{\text{GRBM}}(\mathbf{x},\mathbf{h}) \propto \exp(-\|\mathbf{x}\|^2/2) \cdot p_{\text{RBM}}(\mathbf{x},\mathbf{h}). \tag{9}$$

By marginalizing out the hidden variables $\mathbf{h}$ it is possible to rewrite this as a generic product model as in Eq. (1):

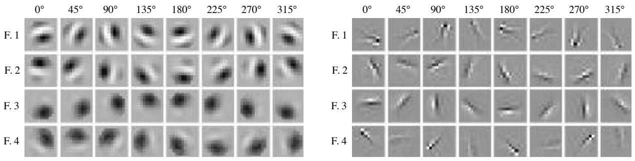
$$p_{\text{RBM}}(\mathbf{x}) \propto \exp\left(\mathbf{c}^{\text{T}}\mathbf{x}\right) \prod_{i=1}^{|\mathcal{F}|} \phi_i\big(\mathbf{F}_{(i)}\mathbf{x}; b_i\big), \tag{10}$$

where the feature transformations $\mathbf{F}_{(i)} = \mathbf{w}_i^{\text{T}} \in \mathbb{R}^{1\times n}$ are single image features (filters) written as a row vector, and $\phi_i(y; b_i) = 1 + \exp(y + b_i)$ with biases b_i of the hidden variables. We keep the biases $\mathbf{c}$ of the visible variables separate and do not make them part of the feature transform.

Standard RBMs are not transformation invariant, but aim to learn pertinent invariances out of the training data, which requires large amounts of data. Moreover, the learned features are not transformation-aware, *i.e.* it is unclear if and how different features relate in terms of image transformations, which makes it difficult to build in-/equivariant feature descriptors from them. Our goal here is to learn transformation-aware features. The most straightforward invariance/awareness to integrate is w.r.t. image translations. For this we apply our framework from Sec. 2.1 with $\mathcal{T} = \mathcal{T}_{\text{C}}$ (see Eq. (5)) to the RBM as given in Eq. (10) and obtain the known convolutional RBM (C-RBM), which has recently been introduced by several authors [18, 21]. C-RBMs naturally extend RBMs to arbitrarily-sized images.

Our contribution is now to generalize C-RBMs to be also invariant to image rotations, which in turn allows to learn features that are both translation- and rotation-aware. To that end we apply our framework to the basic RBM, and use the transformation set $\mathcal{T} = \mathcal{T}_{\text{RC}}$ from Eq. (6):

$$p_{\text{RC-RBM}}(\mathbf{x}) \propto \exp\left(\mathbf{c}^{\text{T}}\mathbf{x}\right) \cdot$$
$$\prod_{\omega\in\Omega} \prod_{(k,l)} \prod_{i=1}^{|\mathcal{F}|} \phi_i\big(\mathbf{w}_i^{\text{T}} \cdot \mathbf{R}_{(\omega)} \cdot \mathbf{C} \cdot \mathbf{S}_{(k,l)}\mathbf{x}; b_i\big). \tag{11}$$



(a) MNIST handwritten digits [1] (b) Natural images (whitened)

Figure 3. **Translation- and rotation-aware** 11×11 **image features.** Each row shows one of 4 features, each column one of 8 implicitly induced feature rotations.

This RC-RBM can also be generalized to continuous-valued images following Eq. (9). Note that the bias terms b_i are shared across all image locations and orientations. If the biases $\mathbf{c}$ of the visible variables differ across the image, Eq. (11) will not be invariant to global image rotations. This is not an issue if the goal is to extract locally equivariant features. If global invariance is desired, we can define $\mathbf{c} = c\cdot\mathbf{1}$.

To train the RC-RBM model, we adapt the contrastive divergence-based learning algorithm for C-RBMs of [21]. No tiling is used; each visible unit corresponds to one pixel. In the examples shown in Fig. 3, we learn 4 features of 11×11 pixels on MNIST binary handwritten digit images (a) and on whitened natural images (b). We use 8 equidistant rotation angles $\Omega = \{0°, 45°, 90°, \ldots, 315°\}$ for a more fine-grained rotation invariance. The matrices $\mathbf{R}_{(\omega)}$ rotate each image patch using bilinear interpolation. To avoid interpolation artifacts in the corners, we only define the feature inside a circular area (visible in Fig. 3(a)).

The RC-RBM has several advantages: It yields transformation-aware features, which allow to predict how the feature activations change when the input is shifted or rotated. It also encourages separate features not to be translations and rotations of one another, since these are already implicitly induced. In this way it leads to a parsimonious and statistically efficient representation [*cf*. 3]. Note that feature extraction with RC-RBMs also does not lead to a higher computational cost than with C-RBMs, since a comparable number of effective features are used in practice.

Other related work. Kivinen and Williams [13] generalize C-RBMs toward rotation-equivariant feature learning, but treat translations and rotations differently – translations in a product framework and rotations using a mixture model. In contrast, our approach is generic and treats all transformations consistently, which for example allows us to rely on existing learning and inference algorithms. Moreover, we apply our method to rotation-equivariant image restoration and object detection. Welling *et al.* [26] and Kavukcuoglu *et al.* [12] learn topographic representations, which allow to assess when two features correspond to similar transformations (*e.g.*, similar rotation angles). By combining feature learning with pooling functions [12], one can obtain locally invariant features. It is not straightforward to extend this to global transformation-equivariance, as is achieved here.

5. Rotation In-/Equivariant Image Descriptor

A simple approach for rotation-invariant object recognition or detection is to model the object class at a canonical orientation and then search over all possible orientations of/in the given image. In practice this is generally not feasible, since at least a traditional feature descriptor would have to be computed at every rotation that is being searched over. At the other end of the spectrum are rotation-invariant image features, which avoid costly computation at many orientations. Unfortunately, these features are usually less powerful at describing the image content, since the class of features that can be considered is restricted. A trivial example is simply using the image intensities or color values. Another approach are annular histogram bins defined by the area between two concentric circles, which allow for rotation-invariant spatial pooling of image features, a strategy for example used by RIFT [17], but known to limit expressiveness [24].

Equivariant features. A tradeoff between the two extremes is offered by rotation-equivariant image features, where a rotation of the input image results in a predictable transformation of the feature activation, which can usually be carried out with little computational effort (*e.g.*, circular shift operations, see Fig. 1). Hence, a rotation-invariant comparison between two image descriptors can be performed quite efficiently (*e.g.*, used by RIFF-Polar [24]).

Standard oriented gradient features, as used by many image descriptors [*e.g.*, 6, 20], have this desirable rotation-equivariance property, which is often exploited. One can, for example, describe the orientations of gradients relative to the dominant orientation at the center of the image patch, thus making the descriptor rotation invariant (*e.g.*, used by RIFT [17]). However, this relies on the assumption that there is a dominant gradient orientation at the patch center, which is true for interest points, but not necessarily for dense feature computation, which is common in sliding-window object detection.

Conventional learned features are difficult to use in this way, since it is not known if and which learned features are rotations of each other, and thus difficult to predict the feature activations given a particular rotation of the image. We now describe a powerful rotation-equivariant descriptor that leverages our rotation-aware RC-RBM features. Note that additional details beyond what can be covered here are available in a separate *supplemental material*.

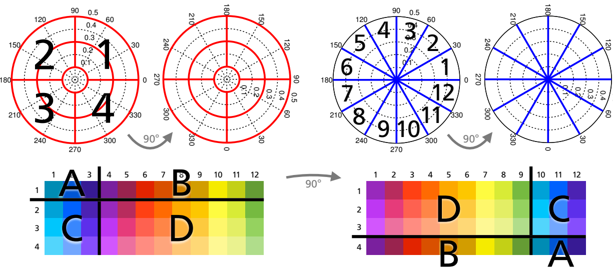


Figure 4. **Simplified descriptor example.** The spatial polar grid (red, left) is divided into $R = 2$ rings with $C = 4$ cells each, besides the central cell, which is treated differently (see text); local images features are computed at $O = 12$ orientations (blue, right). After feature extraction and spatial pooling, the histogram values from all rings can be arranged in a single table (bottom, only one ring shown). The rotation of the image and thus the polar grid (here 90°) results in a cyclical 2D translation of the values in the table, as indicated by the colors and regions denoted A–D.

Equivariant descriptor (EHOF). After extracting features using the RC-RBM from Sec. 4 densely at all locations and orientations (45° increments), we perform non-maximum suppression (NMS) over all orientations for each feature and location. This is akin to standard oriented gradient computation (*e.g.*, in HOG [6]) and significantly increases performance. We then spatially pool (histogram) the NMS results on a polar grid covering the whole image or bounding box, with the intention of converting image rotations to spatial translations of the descriptor. Similar to Takacs *et al*. [24], we use equidistant cell centers (in angle and radius) in polar coordinates (Fig. 4, top left); please note that we allow for an arbitrary number of rings R, cells C, and feature orientations O. The orientation histogram bins in each cell correspond to the rotation angles of the image features; it is important to arrange the histogram bins in order and with equidistant rotation angles apart (Fig. 4, top right).

We then unroll the 3-dimensional histogram $\mathbf{H}_3 \in \mathbb{R}^{R \times C \times O}$ (2 spatial and 1 feature orientation dimension) into the 2-dimensional histogram $\mathbf{H}_2 \in \mathbb{R}^{R \cdot C \times O}$: All spatial cells are assigned a unique ordering by arranging cells from different rings but with neighboring radii together in the rows of the feature matrix $\mathbf{H}_2$ (*i.e.*, first cell 1 from all rings, then all 2nd cells, *etc*.). The columns of $\mathbf{H}_2$ correspond to the histogram orientation bins.

This descriptor layout now has the desired property that a rotation of the image corresponds to a 2-dimensional cyclical translation of the matrix contents (Fig. 4, bottom). If the image is rotated by a multiple of the angular distance between neighboring cell centers in the polar grid, this property holds exactly, and approximately in case of all other rotations. To reduce aliasing artifacts in case of rotations that are not aligned with the polar grid, we use bilinear interpolation in polar coordinates for the spatial pooling. Also, the number of cells per ring should be a multiple of the number of histogram orientation bins (or the other way around), otherwise the translations of rows and columns do not match. An important property of this construction is that a rotation of the input image – and thus translation of the matrix $\mathbf{H}_2$ – does not destroy the relative distribution of spatial locations *and* different orientations. Note that the central cell is a special case, since it does not change its spatial location

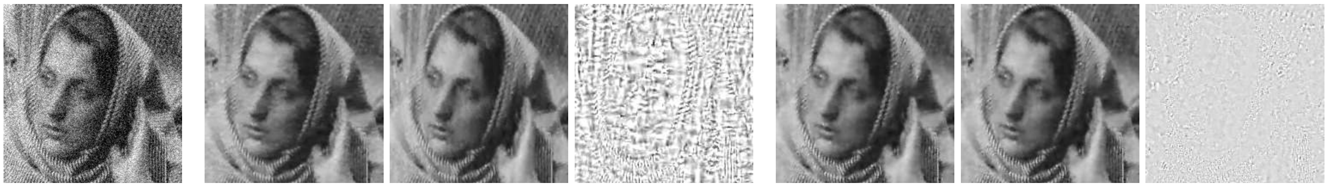

(a) Noisy, 18.67dB (σ=30) (b) FoE, 25.81dB (c) FoE, 26.18dB (90° rot.) (d) $\langle\Delta(\text{b,c})^2\rangle$=11.63 (e) R-FoE, 26.19dB (f) R-FoE, 26.19dB (90° rot.) (g) $\langle\Delta(\text{e,f})^2\rangle$=1.24

Figure 5. **Denoising example (cropped).** *(b,e)* show the results of denoising *(a)*. The results in *(c,f)* are obtained by rotating *(a)* by 90°, denoising the rotated version, and rotating the result back. The results of the non-invariant FoE [23] in *(b,c)* are sensitive to orientation, both visibly and quantitatively (PSNR difference 0.37dB). The difference between the orientations is shown in *(d)*. The proposed rotation-invariant R-FoE *(e,f)* does not suffer from these problems; any difference in *(g)* is due to sampling-based inference. *Best viewed on screen.*

when the input is rotated; only its histogram orientation bins undergo a 1-dimensional cyclical translation.

We term this descriptor an *equivariant histogram of oriented features* (*EHOF*) to emphasize that it can be built from any locally rotation-equivariant feature, including image gradients and steerable filters [7], to yield a globally rotation-equivariant representation.[2]

Invariant descriptor (IHOF). To perform rotation-invariant recognition or detection with this rotation-equivariant descriptor, we could compare two descriptors by defining a custom distance metric as the minimum over all cyclical, 2-dimensional translations between two descriptors (where one of the two is held fixed) that are consistent with an image rotation. A similar strategy is pursued by [24], but since rotation-invariant features are used there, the search reduces to 1-dimensional cyclical shifts of their descriptor vector. An obvious disadvantage is the computational cost for this search (for EHOF over several cyclical, 2-dimensional shifts of the feature matrix). Another issue of embedding rotation invariance in the distance computation is that classification algorithms need to be adapted to this case. A preferable solution is thus to make the descriptor itself invariant. To that end, we compute the 2-dimensional discrete Fourier transform (DFT) of the descriptor matrix and only retain its magnitude, which is well-known to be invariant to cyclical shifts; the same can be done in 1D for the central cell. We term the resulting descriptor an *invariant histogram of oriented features* (*IHOF*). Exploiting the translation invariance of the DFT magnitude has the desired advantage of reducing the computational effort, since it only has to be computed once. Moreover, the IHOF descriptor can be directly used in existing classification frameworks.[2]

While the IHOF descriptor is invariant to rotated inputs, we note that it also remains unchanged for other input transformations, which are presumably unlikely for real images as our experimental findings indicate (Sec. 6).

Other related work. Using the magnitude of the 1-dimensional DFT to build rotation-invariant descriptors is used by Ahonen *et al.* [2] for local binary pattern histograms with applications to classification and recognition. Employing a log-polar transform to convert rotation and scale changes of an image patch to 2D descriptor translations is commonplace in image registration [*cf*. 28]; this includes using the 2D-DFT to retain invariance to rotation and scale variations. Kokkinos and Yuille [14] use the 2D-DFT of the log-polar transform to obtain rotation and scale-invariant image descriptors. One difference of such previous approaches to ours is that they work around sparse (interest) points in the image, where the log-polar region only describes the local structure. In contrast, we obtain a globally rotation-invariant image descriptor with fine-grained spatial binning. We use the 2D-DFT to achieve simultaneous invariance to changes of the spatial and feature dimensions, caused by an in-plane rotation of the whole image.

6. Experiments

We show the benefits of our framework for *(1)* learning rotation-invariant image priors, and *(2)* for learning equivariant features for recognition and detection, both with and without explicit rotation invariance. Please see the *supplemental material* for additional experimental details.

Invariant image denoising. In order to demonstrate the advantage of building explicit invariance to (multiples of) 90° image rotations into learned image priors, we denoise 10 images (from [23]) both in their original orientation, as well as after rotating them by 90°. We compare the FoE implementation of [23] (8 unconstrained features with 3×3 pixels), which does not explicitly enforce rotation invariance, to the R-FoE model proposed in Sec. 3 (8 effective features obtained from 2 learned filters with 3×3 pixels in 4 rotations); denoising is performed using sampling-based MMSE estimation in both cases.

We find that the average performance (PSNR) of an FoE without built-in rotation invariance deteriorates on the rotated images from 32.88dB to 32.77dB (σ=10) and from 28.91dB to 28.75dB (σ=20). In contrast, our rotation-invariant R-FoE achieves exactly the same denoising results of 32.80dB (σ=10) and 28.89dB (σ=20) on original and rotated images, as expected. Both models achieve comparable

[2] MATLAB code is available on the authors' webpages.

results, despite the R-FoE having only $\frac{2}{8}$ as many parameters. Fig. 5 visualizes the difference between both models.

Handwritten digit recognition. To establish a performance baseline for the rotation-aware features learned using the proposed RC-RBM, as well as for the rotation in-/equivariant descriptors, we compare against other feature learning approaches, and also use oriented image derivatives ("gradients") with our descriptors. We always use our descriptor with 1 ring and 8 cells (plus central cell) and extract features at 8 orientation angles. The corresponding EHOF descriptors for each of the 4 learned features (Fig. 3(a)) have 72 dimensions, which we concatenate to represent each digit. We train the RC-RBM on the MNIST handwritten digit dataset [1], which contains 60000 binary training and 10000 test images, and use an rbf-SVM for classification. Tab. 1 gives the recognition results for our method and various competing approaches from the literature. Despite having a parsimonious representation and only using a single model "layer", our approach (EHOF) is competitive with multilayer feature learning approaches, including deep belief networks; somewhat surprisingly, this even holds for simple image derivatives as the sole image feature (akin to HOG [6]). Combining learned features with gradients results in an additional improvement, showing that different properties of the data are captured by each of them. For reference, we also report results with IHOF for MNIST and observe reduced performance, as expected, since MNIST digits do not appear at arbitrary orientations. Otherwise, we see similar behavior, although the RC-RBM features give much better results in this scenario as compared to gradients.

In order to show the benefits of making the rotation-equivariant EHOF descriptor rotation-invariant by using its DFT magnitude, we evaluate the performance on the MNIST-rot dataset [15], containing 12000 images for training and validation, and 50000 test images, in which digits appear at all orientations. Tab. 1 gives the results (following the protocol of [15]) and compares to state-of-the-art techniques from the literature. Even with the EHOF descriptor, we achieve superior results than competing approaches since the rbf-SVM is able to learn necessary invariances from the data. Gradients yield better results than RC-RBM features with EHOF, although the situation is reversed when comparing IHOF performance. Either way, in both cases we gain a substantial improvement when combining image gradients with our learned features. It is important to note that the learned features (alone and combined with gradients) always yield superior results with IHOF. Combining the IHOF descriptors computed from RC-RBM features and image derivatives results in a competitive test error of 3.98%, which is about 50% lower than the previous best result that we are aware of.

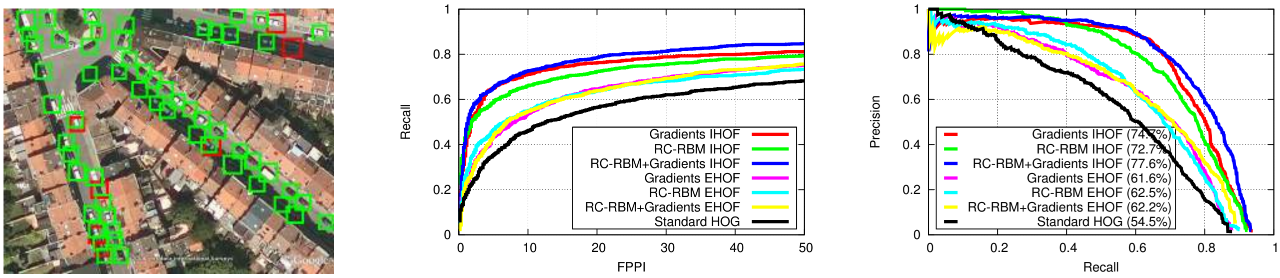
Model / Features	MNIST	MNIST-rot
Multilayer C-RBM, SVM [18]	0.82%	–
Multilayer C-RBM, rbf-SVM [21]	0.67%	–
Deep belief network [11]	1.20%	–
Deep belief network (best from [15])	–	10.30%
SDAIC [16]	–	8.07%
Gradients EHOF	0.97%	5.20%
RC-RBM EHOF	0.85%	6.36%
RC-RBM+Gradients EHOF	0.62%	4.75%
Gradients IHOF	5.82%	8.13%
RC-RBM IHOF	2.66%	5.47%
RC-RBM+Gradients IHOF	2.26%	3.98%

Table 1. **Test error** on MNIST [1] and MNIST-rot [15].

Aerial car detection. Most feature learning approaches from the literature, [21] being a notable exception, only report results for object classification. In contrast, we demonstrate the use of our RC-RBM features and the IHOF image descriptor for rotation-invariant object detection, specifically for finding cars in satellite imagery. We use the dataset introduced by [9], which consists of 30 images, containing a total of 1319 cars that occur at arbitrary orientations and are only annotated with axis-aligned bounding boxes. We perform 5-fold cross validation and report average results across all folds.

Based on a simple and efficient linear SVM classifier, we train a sliding-window detector [6] with fixed window size of 40 × 40 pixels. We use an RC-RBM trained on natural images to extract 4 translation- and rotation-aware features (Fig. 3(b)), each pooled in the EHOF descriptor on a polar grid with 3 rings and 16 cells per ring (plus central cell), and a histogram over 8 feature orientations for each cell. The combined EHOF descriptors have 1568 dimensions in this case. The rotation-invariant IHOF descriptor is obtained using the 2D-DFT magnitude.

As Fig. 6 shows, our IHOF descriptor substantially increases the detection performance over a standard HOG descriptor (also with a linear SVM) from 54.5% average precision (AP) to 72.7%. For reference, we also report the results of using the EHOF descriptor, which underline the benefits of using the rotation-invariant IHOF descriptor for this task. Since the learned RC-RBM features are not as localized as the gradient features used in the successful HOG descriptor, we also evaluated the use of simple gradient features in the rotation-invariant IHOF descriptor. This leads to an improved performance of 74.7% AP, which is close to the recent approach of Vedaldi *et al.* [25]. Their approach is much more complex and uses structured output SVM regressors and non-linear kernels to achieve 75.7% AP. Note that we also clearly outperform the context-based approach of Heitz and Koller [9]. More importantly, the RC-RBM features again contain information that is complementary to gradient features. Combining both boosts the performance to 77.6%, which is a clear improvement over the best performance reported in the literature (75.7% AP [25]).



(a) Detection examples (b) False positives per image (FPPI) vs. Recall (c) Recall vs. Precision

Figure 6. **Aerial car detection.** *(a)* Example of detections with the *IHOF* descriptor encoding *RC-RBM+Gradients* features, where green boxes indicate correct detections and red boxes incorrect ones. *(b,c)* Common performance measures.

Still, we expect to obtain even better results with more advanced variants of RBMs [*e.g.*, 5], or through stacking to obtain deep models. Furthermore, adapting descriptors to features plays an important role for recognition/detection performance, which so far has mostly been explored for gradient features. Hence, an interesting avenue for further research is descriptor learning [*e.g.*, 4].

7. Summary

We proposed a framework for transformation-invariant feature learning using product models, demonstrated how popular translation-invariant models are special cases, and studied its application to inducing rotation invariance. We extended a learned image prior to be (90°) rotation-invariant, and showed its advantages over a conventional prior. We also applied our framework to make convolutional RBMs rotation invariant, and used this RC-RBM for translation- and rotation-aware feature learning. Finally, we employed the learned features, or other oriented features, to build a globally rotation-equivariant image descriptor (EHOF), which can be made rotation-invariant (IHOF) using the 2D-DFT magnitude. We demonstrated state-of-the-art results on two challenging datasets for rotation-invariant recognition and detection.

Acknowledgements. We thank Thorsten Franzel for providing object detection code, and Carsten Rother and Stefan Walk for helpful discussions and suggestions. This work is funded in part by the German Federal Ministry of Education and Research project SICURA (BMBF-13N11124) and by Microsoft Research through its PhD Scholarship Programme.

References

[1] http://yann.lecun.com/exdb/mnist/.
[2] T. Ahonen, J. Matas, C. He, and M. Pietikäinen. Rotation invariant image description with local binary pattern histogram Fourier features. *SCIA 2009*.
[3] J. Bergstra, A. Courville, and Y. Bengio. The statistical inefficiency of sparse coding for images (or, one Gabor to rule them all). Technical Report 1109.6638v2, arXiv, 2011.
[4] M. Brown, G. Hua, and S. Winder. Discriminative learning of local image descriptors. *PAMI*, 33(1), 2011.
[5] A. Courville, J. Bergstra, and Y. Bengio. A spike and slab restricted Boltzmann machine. *AISTATS 2011*.
[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. *CVPR 2005*.
[7] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *PAMI*, 13(9), 1991.
[8] B. J. Frey and N. Jojic. Transformed component analysis: Joint estimation of spatial transformations and image components. *CVPR 1999*.
[9] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. *ECCV 2008*.
[10] G. E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Comput.*, 14(8), 2002.
[11] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786), 2006.
[12] K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. Le-Cun. Learning invariant features through topographic filter maps. *CVPR 2009*.
[13] J. J. Kivinen and C. K. I. Williams. Transformation equivariant Boltzmann machines. *ICANN 2011*.
[14] I. Kokkinos and A. Yuille. Scale invariance without scale selection. *CVPR 2008*.
[15] H. Larochelle, D. Erhan, A. Courville, J. Bergstra, and Y. Bengio. An empirical evaluation of deep architectures on problems with many factors of variation. *ICML 2007*.
[16] H. Larochelle, D. Erhan, and P. Vincent. Deep learning using robust interdependent codes. *AISTATS 2009*.
[17] S. Lazebnik, C. Schmid, and J. Ponce. Semi-local affine parts for object recognition. *BMVC 2004*.
[18] H. Lee, R. Grosse, R. Ranganath, and A. Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. *ICML 2009*.
[19] S. Z. Li. *Markov Random Field Modeling in Image Analysis*. Springer, 2nd edition, 2001.
[20] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2), 2004.
[21] M. Norouzi, M. Ranjbar, and G. Mori. Stacks of convolutional restricted Boltzmann machines for shift-invariant feature learning. *CVPR 2009*.
[22] S. Roth and M. J. Black. Fields of experts. *IJCV*, 82(2), 2009.
[23] U. Schmidt, Q. Gao, and S. Roth. A generative perspective on MRFs in low-level vision. *CVPR 2010*.
[24] G. Takacs, V. Chandrasekhar, H. Chen, D. Chen, S. Tsai, R. Grzeszczuk, and B. Girod. Permutable descriptors for orientation-invariant image matching. *SPIE ADIP 2010*.
[25] A. Vedaldi, M. Blaschko, and A. Zisserman. Learning equivariant structured output SVM regressors. *ICCV 2011*.
[26] M. Welling, G. E. Hinton, and S. Osindero. Learning sparse topographic representations with products of Student-t distributions. *NIPS*2002*.
[27] S. C. Zhu and D. Mumford. Prior learning and Gibbs reaction-diffusion. *PAMI*, 19(11), 1997.
[28] S. Zokai and G. Wolberg. Image registration using log-polar mappings for recovery of large-scale similarity and projective transformations. *IEEE TIP*, 14(10), 2005.