Robust Multi-Organ Nucleus Segmentation Using a Locally Rotation Invariant Bispectral U-Net

Valentin Oreiller1,2, Julien Fageot3, Vincent Andrearczyk1, John O. Prior2, Adrien Depeursinge1,2
1 Institute of Information Systems, HES-SO, Sierre, Switzerland
2 Service of Nuclear Medicine and Molecular Imaging, CHUV, Lausanne, Switzerland
3 AudioVisual CommunicationS Laboratory (LCAV), EPFL, Lausanne, Switzerland

Editors: Under Review for MIDL 2022

Abstract

Locally Rotation Invariant (LRI) operators have shown great potential to robustly identify biomedical textures where discriminative patterns appear at random positions and orientations. We build LRI operators through the local projection of the image on circular harmonics followed by the computation of the bispectrum, which is LRI by design. This formulation allows to avoid the discretization of the orientations and does not require any criterion to locally align the descriptors. This operator is used in a convolutional layer resulting in LRI Convolutional Neural Networks (LRI CNN). To evaluate the relevance of this approach, we use it to segment cellular nuclei in histopathological images. We compare the proposed bispectral LRI layer against a standard convolutional layer in a U-Net architecture. While they perform equally in terms of F-score, the LRI CNN provides more robust segmentation with respect to orientation, even when rotational data augmentation is used. This robustness is essential when the relevant pattern may vary in orientation, which is often the case in medical images.

Keywords: Local Rotation Invariance, Convolutional Network, Segmentation, Bispectrum

1. Introduction

Robustness of Convolutional Neural Networks (CNNs) to changes in orientations of the input structures (e.g., nucleus, glands) has been little investigated and may have an important impact on the usability of the methods in practice. Biomedical textures are composed of local patterns that appear at random positions and orientations. Local Rotation Invariant (LRI) operators have been shown to be crucial to characterize such texture (Depeursinge et al., 2018). A common strategy to design LRI operators is to align local descriptors. This includes the Maximum Response 8 (MR8) filterbank (Varma and Zisserman, 2005) and Local Binary Patterns (LBP) (Ojala et al., 2002; Ahonen et al., 2009). Other methods relying on steerability have been proposed to avoid error due to orientation sampling, such as steerable filters (Unser and Chenouard, 2013; Zhao and Blu, 2020; Fageot et al., 2021), Riesz (Dicente Cid et al., 2017), and steerable wavelets (Depeursinge et al., 2017; Puspoki et al., 2019). Another well known method is the scale-invariant feature transform (Lowe, 2004). These methods typically require discretizing orientations and an arbitrary criterion.
to select the dominant local orientation on which the descriptor is aligned. Built-in LRI approaches have been proposed to avoid using such arbitrary criterion. For instance, the power spectrum was used in (Depeursinge et al., 2018; Andrearczyk et al., 2019) which allows obtaining a LRI operator continuously defined on the rotation domain. In this work, we design a LRI operator based on a similar but more evolved descriptor, i.e. the bispectrum, that can be embedded in a CNN.

CNNs have revolutionized the field of computer vision and biomedical image analysis. Rotation invariance in CNNs is still mainly induced via data augmentation either at training- or test-time. However, built-in rotation equivariance was shown to outperform both training- and test-time augmentation on histopathological image analysis (Lafarge et al., 2021). Rotation-equivariant networks also showed improved robustness to geometric adversarial perturbations (Dumont et al., 2018). A large body of research has focused on designing networks with built-in rotation equivariance and are mainly based on discretized rotations (i.e. group equivariant CNN) or steerable filters (Weiler et al., 2018, 2017; Cohen and Welling, 2016; Bekkers, 2019; Kondor and Trivedi, 2018; Cohen et al., 2019).

In this work, we propose a CNN design that is invariant to local rotations rather than rotation equivariant. The motivation is that such a design will provide substantial robustness to changes in pattern orientation when identifying biomedical textures. Furthermore, the proposed LRI operators are also globally rotation equivariant (Andrearczyk et al., 2020). While global rotation invariance may be obtained by data augmentation, invariance to local rotation can not be achieved in this way. In (Andrearczyk et al., 2020), the authors proposed two different designs to implement LRI CNNs, one with steerable filters and another one based on the power spectrum. In (Eickenberg et al., 2017), the power spectrum was used within the scattering transform framework (Ablowitz et al., 1974) to obtain global rotation equivariant feature maps. Those works closely relate to our design. One key difference is that we use the bispectrum rather than the power spectrum. Our design relies on the shift invariance property of the bispectrum which translates into rotational invariance for functions defined on the circle. We choose the bispectrum over the power spectrum for its completeness, i.e. the bispectrum completely characterizes a function up to a shift (Kakarala, 2012). We evaluate this design with a simple U-Net (Ronneberger et al., 2015) on histopathological images. The evaluation is greatly inspired by the work of Lafarge et al. (2021), in order to have an external comparison. However, our results are not directly comparable since we do not use the same training/testing/validation split.

2. Methods

In this section, we develop the theoretical background as well as the implementation details to design a LRI CNN. The main idea is to obtain a LRI convolutional layer that is functionally identical to a standard convolutional layer. Then, we evaluate our layer in a CNN and compare it to a CNN with the same architecture but with standard layers. This work focuses on developing 2D CNNs that are invariant to the orientation at which local patterns appear. The proposed design relies on the rotation invariance property of the bispectrum.
2.1. Notations

We consider 2D images as functions $I \in L_2(\mathbb{R}^2)$, where the value $I(x) \in \mathbb{R}$ corresponds to the pixel intensity at location $x = (x_1, x_2) \in \mathbb{R}^2$. The rotation of an image $I$ is written as $I(R \cdot)$, where $R \in SO(2)$ is the corresponding 2D rotation matrix.

The circle is denoted as $S^1 = \{ x \in \mathbb{R}^2 : ||x||_2 = 1 \}$. Polar coordinates are defined as $(x_1, x_2) = (\rho \cos(\theta), \rho \sin(\theta))$ with $\rho \geq 0$ and $\theta \in [0, 2\pi)$. We use the following notation for the mapping from polar to cartesian: $\rho(x) = ||x||$ and $\theta(x)$ the standard mapping from $x$ to its polar angle. For clarity purposes, we often do not disclose the dependency on $x$ for $\rho$ and $\theta$. We consider square-integrable functions defined on the circle $f \in L_2(S^1)$ and express them as functions of the polar angle $f(\theta)$. The inner product is defined by $(f, g)_{L_2(S^1)} = \int_0^{2\pi} f(\theta) \overline{g(\theta)} d\theta$. The rotation of a function $f$ by an angle $\theta_0$ is simply the function “shifted” by that angle i.e. $f(\cdot - \theta_0)$. The Fourier transform of the function $f$ is defined as $\hat{f}[n] = \int_0^{2\pi} f(\theta) e^{-j n \theta} d\theta$ for any $n \in \mathbb{Z}$. The triangle function is referred to as $x \in \mathbb{R} \mapsto \text{tri}(x)$ and is defined as $\text{tri}(x) = 1 - |x|$ if $|x| < 1$ and $\text{tri}(x) = 0$ otherwise.

2.2. LRI Operators

Our mathematical formalism relies on the concepts of image operators acting over continuous images, as presented in (Depeursinge and Fageot, 2017). This work focuses on image operators $G$ that are LRI as previously introduced in (Andrearczyk et al., 2020). An operator $G$ is LRI if it satisfies the three following properties:

- **Locality**: there exists $\rho_0 > 0$ such that, for every $x \in \mathbb{R}^2$ and every image $I \in L_2(\mathbb{R}^2)$, the quantity $G\{I\}(x)$ only depends on local image values $I(y)$ for $||y - x|| \leq \rho_0$.

- **Global equivariance to translations**: For any $I \in L_2(\mathbb{R}^2)$,
  
  $$G\{I(\cdot - x_0)\} = G\{I\}(\cdot - x_0) \quad \text{for any } x_0 \in \mathbb{R}^2.$$  

- **Global equivariance to rotations**: For any $I \in L_2(\mathbb{R}^2)$,
  
  $$G\{I(R_0 \cdot)\} = G\{I\}(R_0 \cdot) \quad \text{for any } R_0 \in SO(2).$$

The simplest example of an LRI operator is the convolution with filter

$$G\{I\}(x) = (I \ast h)(x),$$

where $h$ is isotropic with finite support, i.e. $h(x_1, x_2) = h(\rho)$ is purely radial and vanishes for $\rho > \rho_0$ for some fixed $\rho > 0$ (Andrearczyk et al., 2020, Prop. 1). However, isotropic filters are limited since they discard local directional information. We will now see how we can extend this notion of equivariance to directional sensitive operators using the bispectrum.

2.3. Bispectral LRI Operators and Layers

We introduce the theoretical background and methodology to implement our bispectral image operators. They have the advantage of being LRI and sensitive to directional information. We also detail how these operators can be embedded into a convolutional layer.
2.3.1. The Bispectrum: A Complete Set of Rotation Invariant Features

We first focus on features $\mathcal{H}: L_2(S^1) \to \mathbb{R}$ of circular functions that are rotation invariant, i.e. such that

$$\mathcal{H}\{f(\theta)\} = \mathcal{H}\{f(\theta - \theta_0)\}$$  \hspace{1cm} (2)

for any function $f$ and any angle $\theta_0 \in [0, 2\pi)$. The typical example is the power spectrum of $f$ defined from its Fourier series coefficients $\hat{f}[n]$. For any fixed discrete frequency $n_0 \in \mathbb{Z}$, the Fourier feature $f \mapsto |\hat{f}[n_0]|^2$ is easily shown to be rotation invariant in the sense of (2). However, the power spectrum discards the phase information of $\hat{f}$ and thus, does not allow for the complete characterization of a polar function.

For this reason, we consider a more elaborated Fourier-based feature, named the bispectrum, that retains the rotation invariance while keeping the phase information (Bartelt et al., 1984; Kakarala, 2012). The bispectrum is defined for circular functions in $L_2$. The bispectrum is complete (Kakarala, 2012) in the sense that it discriminates between two functions up to a rotation. This motivates our choice of using the bispectrum over the power spectrum to design LRI layers.

2.3.2. Bispectral LRI Operators

The bispectrum is defined for circular functions in $L_2(S^1)$ and can be used to define image operators. In this section, we fix a radial function $h(\rho) \in L_2(\mathbb{R}^2)$ and consider the steerable kernel $\kappa_n(\rho, \theta) = h(\rho) e^{in\theta}$ associated to the discrete frequency $n \in \mathbb{Z}$. We moreover introduce the convolution operator $C_n\{I\}(x) = (I * \kappa_n)(x)$. We observe that we can write, for any fixed position $x_0 \in \mathbb{R}^2$,

$$C_n\{I\}(x_0) = \int_0^{2\pi} \left( \int_0^{+\infty} (I(x_0 - \cdot))(\rho, \theta) h(\rho) \rho d\rho \right) e^{-jn\theta} d\theta.$$

We can interpret the circular function $\theta \mapsto I_{x_0}^h(\theta) := \int_0^{+\infty} (I(x_0 - \cdot)(\rho, \theta) h(\rho) \rho d\rho$ as the radial projection of the shifted image $I(x_0 - \cdot)$ against the radial profile $h$. Hence, $C_n\{I\}(x_0)$ performs the $n^{th}$ Fourier coefficient of the periodic function $I_{x_0}^h$.

For any $n, n' \in \mathbb{Z}$, we define the image operator $G_{n,n'}$ as

$$G_{n,n'}\{I\}(x) = C_n\{I\}(x) \overline{C_{n'}^n\{I\}(x)}.$$

Then, we see by comparing (4) with (3) that, for any fixed position $x_0$, $G_{n,n'}\{I\}(x_0)$ is the bispectrum of the projection $I_{x_0}^h \in L_2(S^1)$. We call $G_{n,n'}$ the bispectral operator of frequencies $n, n'$. Its main invariance properties are summarized in Theorem 1, whose proof is provided in Appendix A.

**Theorem 1** Let $n, n' \in \mathbb{Z}$ and $\rho \mapsto h(\rho)$ a radial profile with finite support. Then, the bispectrum operator $G_{n,n'}$ is LRI.
2.3.3. Implementation of the LRI Layer

We fix a maximal order \( N \geq 0 \) and consider the bispectral operators \( \mathcal{G}_{n,n'} \) for any \( n, n' \geq 0 \) such that \( n + n' \leq N \). By doing so, we only consider angular frequencies smaller than \( N \). Moreover, we only compute non-repeating pairs as \( \mathcal{G}_{n,n'} = \mathcal{G}_{n',n} \). We define \( L \) as the number of combinations of \( n, n' \) such that \( n \leq n' \) and \( n + n' \leq N \).

For any \( n, n' \), the application of the bispectral operator \( \mathcal{G}_{n,n'} \) at each layer of the bispectral LRI network is implemented in four steps as depicted in Fig. 1. First, the feature maps are computed as a complex convolution \( \mathcal{C}_n(x_0) = \sum_{i=1}^{C_{in}} y_i(x) * h_{i,o}^n(x)e^{-jn\theta(x)}(x_0) \), with \( y_i \) the \( i \)th channel of the previous feature maps, \( h_{i,o}^n \) the filters that are learned by gradient descent. The parametrization of the filters \( h_{i,o}^n \) is detailed in Section 2.3.4. The indices \( i \) and \( o \) run through \([1, \ldots, C_{in}]\) and \([1, \ldots, C_{out}]\) respectively and represent the input and output channels of the layer.

The second step consists in applying (4) to the feature maps \( \mathcal{C}_n(x) \), yielding the desired operator \( \mathcal{G}_{n,n'} \).

The third step is the concatenation of the real and imaginary part of \( \mathcal{G}_{n,n'} \), which is followed by a point-wise non-linearity of the following form \( \sigma(x) = \text{sign}(x) \log(1 + |x|) \). This choice is made to avoid vanishing and exploding gradients with the cubic nature of the bispectral feature maps (see Eq. (4)). Learned biases are added to the resulting feature maps and Rectified Linear Unit (ReLU) is applied.

In the last step, the number of features maps is reduced by a \( 1 \times 1 \) convolution in order to obtain \( C_{out} \) output channels. This whole process results in a layer that takes as an input \( C_{in} \) feature maps and outputs \( C_{out} \) feature maps like a standard convolutional layer.

2.3.4. Parametrization of the Radial Profiles

The radial profiles \( h_{i,o}^n \) are parametrized as follows:

\[
    h_{i,o}^n(\rho) = \sum_{j=0}^{J} w_{i,j,o}^n \psi_j(\rho),
\]

where the \( w_{i,j,o}^n \) are the learnable parameters of the layer, \( i \in [1, \ldots, C_{in}] \), \( o \in [1, \ldots, C_{out}] \) and \( 0 \leq n \leq \hat{N} \). The radial functions \( \psi_j \) are chosen as \( \psi_j(\rho) = \text{tri}(\rho - j) \).

2.4. Dataset

We test our design on a subset of the dataset proposed in the MoNuSeg 2018 challenge (Kumar et al., 2019) which consists of 24 Hematoxylin and Eosin (H&E) stained im-
ages selected from whole slice images acquired at the commonly used 40× magnification provided by The Cancer Genome Atlas (Koboldt et al., 2012). This subset contains 61000 × 1000 images per tissue type for a total of four different tissue types (breast, liver, kidney and prostate). Nuclei instance segmentation is available for these 24 images. We followed a similar splitting scheme than proposed by (Lafarge et al., 2019), namely 4 × 3 images for training, 4 × 1 for validation and 4 × 2 for testing. We repeat 10 random splits to evaluate the variation of the models. As preprocessing, we use the method described in (Macenko et al., 2009) to normalize the H&E images. We adapted the code from https://github.com/schaugf/HEnorm_python to fit our needs.

2.5. Network Architecture and Training

The networks used in this work are based on the U-Net architecture (Ronneberger et al., 2015). However, we use a lighter version of the U-Net, as proposed in (Lafarge et al., 2021), which contains only two levels. All the convolutional layers have a kernel size of 5 × 5 and are connected to a batch normalization followed by a ReLU. The encoder path includes three convolutions layers with max-pooling to reduce the spatial dimension. The number of feature maps for each layer respectively are 8, 16 and 32. The decoder path contains 2 layers preceded by a bi-linear upsampling. The final prediction is modeled as a three classes probability, i.e. nucleus core, nucleus border, and background. The prediction is computed with a 1 × 1 convolution with a softmax activation. The final output is post-processed to obtain instance segmentation of each nucleus. This post-processing consists in binarizing the prediction with a threshold of 0.5, then the core and border prediction are respectively used as seed and landscape for a watershed algorithm\(^1\) (Falcão et al., 2004).

The networks are trained by minimizing the class-balanced cross-entropy with an Adam optimizer and a learning rate of 10\(^{-3}\). The models are trained on patches of 60 × 60 randomly drawn from the training set with a batch size of 16. We apply multiple of 90° rotation as data augmentation\(^2\) as well as random brightness shift. The training is run for a maximum number of epochs of 200 and we apply an early stop monitoring the F-score. The experiments were performed on a Nvidia Tesla K80. The code for the implementation is available on our GitHub repository\(^3\).

2.6. Metrics and Evaluation

Two types of experiments are conducted. We first evaluate the performance of our bispectral LRI U-Net against a standard U-Net. The metric used for this experiment is the F-score and we consider a match when the predicted nucleus had more than 50% overlap with the ground-truth nucleus. Since the radial profiles of the proposed LRI layer do not cover the entire 5 × 5 kernel, we use masked kernels in the standard U-Net to ensure a fair comparison. The masked kernel consists of removing the four pixels in the corners. All models are trained and tested on the same 10 train/validation/testing splits to assess performance variation.

In the second experiment, we evaluate the robustness of the predictions made by the two designs in terms of 90°-rotation equivariance. We feed the networks with the same image

---

2. We also evaluated the training without this augmentation, but did not observed any significant changes.
rotated at different orientations. Then, we apply the inverse rotation to the output maps and compare the difference in pixel-wise prediction. We use the Root Mean Square Error (RMSE) on the three classes of raw probabilities prediction as well as the Dice Similarity Coefficient (DSC) on the post-processed probability to measure the overlap of predictions for each orientation, indicating the robustness of prediction with respect to input orientation.

3. Results

Fig. 3 relates the F-score for varying maximum degree $N$ of the bispectral U-Net and standard U-Net. The best performing bispectral U-Net had a F-score of $0.7157 \pm 0.0328$ with a maximum degree $N = 7$ and 136,147 parameters (standard U-Net 71,571 parameters). Thus, we trained a standard U-Net with more feature maps to increase the number of parameters (134,709 parameters) which obtained an average F-score of $0.7324 \pm 0.0326$.

To account for the discrepancy in the number of parameters at different maximal degree $N$, we train a bispectral U-Net with $N = 0$ and more feature maps to obtain a network of 45,779 parameters (comparable to the number of parameters of the network with $N=3$). The resulting network achieved an average F-score of $0.6592 \pm 0.0286$.

Table 1 summarizes the average RMSE and DSC between predictions of rotated images. The average is taken on all the images from the testing set of one split and all pairs of 90° rotations. Fig. 2 illustrates the pixel-wise variation between predictions when fed to the networks at different orientations.

Table 1: Quantitative evaluation of segmentation robustness. It is worth noting that rotational data augmentation was used during training for both approaches.

<table>
<thead>
<tr>
<th>Model</th>
<th>RMSE border</th>
<th>RMSE core</th>
<th>RMSE background</th>
<th>DSC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard U-Net</td>
<td>8.49 ± 1.35 %</td>
<td>7.66 ± 1.72 %</td>
<td>9.15 ± 1.43 %</td>
<td>0.9153 ± 0.0205</td>
</tr>
<tr>
<td>Bispectral U-Net ($N=7$)</td>
<td>2.50e-5 ±0.78e-5 %</td>
<td>2.26e-5 ±0.77e-5 %</td>
<td>2.53e-5 ±0.87e-5 %</td>
<td>0.9876 ± 0.0044</td>
</tr>
</tbody>
</table>

4. Discussions and Conclusion

We propose a novel 2D LRI layer based on the bispectrum that can be integrated in any CNN architecture. The aim of this design is to improve the robustness of the predictions...
when inputs do not have a standardized orientation, or when local structures of interest can appear at any orientation. While several approaches were proposed to achieve rotational equivariance, most of them required to coarsely discretize orientations (e.g. right angle rotations). Our proposition obviates this constraint by relying on invariants based on bispectral operators that can characterize patterns with high specificity, up to a rotation of the latter, hence the LRI property.

We first present the proposed bispectral operators and demonstrate their LRI property in Section 2.3.2 and Appendix A. We also detail how to use them in a convolutional layer in Section 2.3.3. Second, we incorporate the LRI layer into a U-Net to allow robust segmentation of multi-organ nucleus in histopathology images. We observe that the segmentation performance of the LRI U-Net is on par with a standard U-Net (see Fig. 3). While the bispectral U-Net was slightly outperformed by the standard U-Net, it is difficult to evaluate to which extent the post-processing had a role in this difference (see Appendix B).

However, an important gain is obtained in terms of robustness with respect to the orientation of the input (see Table 1) thanks to the rotation equivariance property of the used image operators. This robustness is crucial for most medical image analysis tasks where structures of interest often appear at various orientations. We observe that standard methods lack robustness, even when rotational data augmentation is used (see Table 1 and Fig. 2). While most studies focused on classification or segmentation performance alone, robustness to changes in input orientation was little investigated and may have important consequences on the usability of the models.

Our work recognizes several limitations. The segmentation performance presented in Fig. 3 is not at the level of the state of the art on this dataset. Our goal was to compare with standard baseline methods such as the U-Net without using refinements e.g. postprocessing of the segmentation maps or ensembling. Future work includes the extension of the bispectral operator to 3D and extensive comparisons with group-equivariant approaches.

Acknowledgments

This work was partially supported by the SNSF grants 205320_179069 and P400P2_194364, the SPHN IMAGINE project and the Hasler Foundation.
References


Appendix A. Proof of Theorem 1

For any $n \in \mathbb{Z}$, the operator $I \mapsto \mathcal{C}_n \{I\} = I * \kappa_n$ is equivariant to translations (as a convolution) and local (due to the fact that $h$, and therefore $\kappa_n(\rho, \theta) = h(\rho)e^{jn\theta}$, have a finite support). Then, the operator $\mathcal{G}_{n,n'}$ inherits these properties, as is clear from its definition in Eq. (4).

We moreover observe that, for any rotation matrix $R_{\theta_0}$ associated to the angle $\theta_0$, we have that

$$I(R_{\theta_0} \cdot) * \kappa_n = (I * \kappa_n(R_{\theta_0}^{-1} \cdot))(R_{\theta_0} \cdot) = e^{in\theta_0}(I * \kappa_n)(R_{\theta_0} \cdot), \tag{6}$$

where the first inequality comes from the relation $f(R_{\theta_0} \cdot) * g = (f * g(R_{\theta_0}^{-1} \cdot))(R_{\theta_0} \cdot)$ valid for any $f, g \in L_2(\mathbb{R}^2)$ and the second uses that $\kappa_n(R_{\theta_0} \cdot) = e^{in\theta_0}\kappa_n$. This implies that, for any image $I$ and any $x \in \mathbb{R}^2$,

$$\mathcal{C}_n \{I(R_{\theta_0} \cdot)\} = I * (\kappa_n(R_{\theta_0} \cdot))(x) = e^{in\theta_0}\mathcal{C}_n \{I\}(R_{\theta_0} x). \tag{7}$$

We have therefore that

$$\mathcal{G}_{n,n'} \{I(R_{\theta_0} \cdot)\}(x) = e^{in\theta_0}e^{in'\theta_0}e^{in+n'\theta_0}\mathcal{C}_n \{I\}(R_{\theta_0} x)\mathcal{C}_{n'} \{I\}(R_{\theta_0} x)\mathcal{C}_{n+n'} \{I\}(R_{\theta_0} x)$$

and the operator $\mathcal{G}_{n,n'}$ is globally rotation equivariant. Being local and equivariant to shifts and rotations, $\mathcal{G}_{n,n'}$ is LRI.

Appendix B. Additional Results

In this Appendix, we report additional results. First, we computed additional metrics to investigate the performance difference between the bispectral and the standard U-Nets. The precision and recall for the bispectral U-Net ($N=7$) were $0.7004 \pm 0.0617$ and $0.7500 \pm 0.0350$, respectively. For the standard U-Net we obtained $0.7156 \pm 0.0505$ and $0.7686 \pm 0.0400$. We also trained a standard U-Net without masking the kernels, obtaining an F-score of $0.7318 \pm 0.0220$.

Second, we compare the computational time between the two approaches. The average forward time on 80 1000x1000 images for the bispectral U-Net with $N=0, 2, 4$, and 7 were 0.14, 0.42, 1.05, and 2.17 seconds respectively. The average forward time for the standard U-Net was 0.07 seconds.

Finally, we report some examples of predictions made by the bispectral and standard U-Net. Fig. 4 illustrates the predictions on a patch where the bispectral U-Net outperforms the standard U-Net. Fig. 5 shows a patch where the bispectral U-Net over segments and thus performs worse than the spectral U-Net.
Figure 4: Illustration of predictions where the bispectral U-Net outperforms the standard U-Net. The F-score on this patch for the bispectral and standard U-Net are, respectively, 0.8929 and 0.7931. The colors in the top row images are used to highlight the different nucleus instances obtained after the application of the watershed algorithm.
Figure 5: Illustration of predictions where the bispectral U-Net over segments. The F-score on this patch for the bispectral and standard U-Net are, respectively, 0.6667 and 0.7619. The colors in the top row images are used to highlight the different nucleus instances obtained after the application of the watershed algorithm.

The quantitative results seem to indicate that the bispectral U-Net always performed a little worse than the standard U-Net. However, as highlighted in Fig. 4 and 5, it is difficult to assess whether these differences come from the post-processing step.

Appendix C. Comparison with Spectral U-Net

This appendix describes the results with a similar architecture to the bispectral U-Net. However, the invariant used here is the spectrum yielding a spectral U-Net, which is very similar to the design proposed in (Eickenberg et al., 2017) or in (Andrearczyk et al., 2020). The design of the LRI layer is almost the same and the associated LRI operator is defined as:

\[ G_n \{ I \}(x) = C_n \{ I \}(x) C_n^\dagger \{ I \}(x). \]  

(8)
Figure 6: Performance evaluation of the different networks. The average F-scores across 10 repetitions of the proposed bispectral (blue) and spectral (orange) U-Net evaluated at different maximum degrees \( N \) are reported.

This invariant is equivalent to taking the modulus, \( i.e. \) spectrum, of the Fourier coefficients \( C_n \). The results are reported in Fig. 6.

The results suggest slightly better performance for the bispectral U-Net, but the difference remains too marginal to draw conclusions.