

# Regions of Interest in the Context of ImgLib2

Tobias Pietzsch      Stephan Saalfeld

April 29, 2012

## Abstract

This document collects some naive thoughts about what we consider a region of interest (ROI). Our ideas are biased towards what we can imagine to use it for and express it in the context of ImgLib2.

## 1 Regions of Interest

A *region of interest* (ROI) is a function  $f : \mathbb{R}^n \rightarrow [0, 1]$  that assigns a real precision interest value ranging from 0 to 1 to each coordinate in an  $n$ -dimensional Euclidean space. Such a function can be interpreted as a probability map. The ‘negation’ of a ROI  $f(x)$  is defined as  $\neg f(x) = 1 - f(x)$ . In natural terms, a coordinate is ‘inside’ of the ROI by an amount of  $f(x)$  and ‘outside’ of the ROI by an amount of  $\neg f(x)$ . The intersection of two ROIs  $f(x)$  and  $g(x)$  can be computed using pairwise multiplication of interest values. Using these basic operations

$$\neg : \quad (\neg f)(x) = 1 - f(x) \quad (1)$$

$$\cap : \quad (f \cap g)(x) = f(x) \cdot g(x) \quad (2)$$

other set operations on ROIs can be defined.

$$\cup : \quad f \cup g = \neg(\neg f \cap \neg g) \quad (3)$$

$$\setminus : \quad f \setminus g = f \cap \neg g \quad (4)$$

Additional operations can be defined that have no straightforward interpretation as probabilities

$$\max : \quad \max(f, g)(x) = f(x) \text{ if } f(x) > g(x), g(x) \text{ otherwise} \quad (5)$$

$$\min : \quad \min(f, g)(x) = f(x) \text{ if } f(x) < g(x), g(x) \text{ otherwise} \quad (6)$$

$$\oplus : \quad (f \oplus g)(x) = \min(1, f(x) + g(x)) \quad (7)$$

$$\ominus : \quad (f \ominus g)(x) = \max(0, f(x) - g(x)) \quad (8)$$

In `ImgLib2`, a generic ROI would implement the `RealRandomAccessible` interface with a specialized `RealType` that implements `add` and `sub` as described above preserving the  $[0, 1]$  range and an in-place `invert` method as described above.

Various forms of ROIs exist. Specializations constrain either the domain or the co-domain of the general ROI. Following a few examples:

Discrete ROI	$f : \mathbb{Z}^n \rightarrow [0, 1]$	(9)
Mask	$f : \mathbb{R}^n \rightarrow \{0, 1\}$	(10)
Discrete Mask	$f : \mathbb{Z}^n \rightarrow \{0, 1\}$	(11)
Polygon Mask	$f : \mathbb{R}^2 \rightarrow \{0, 1\}$	(12)

In `ImgLib2`, a generic discrete ROI would implement the `RandomAccessible` interface. If the domain is limited (as it is in a digital image) it would implement the `RandomAccessibleInterval` and optionally the `IterableInterval` interfaces. Iteration can be achieved through `Views.iterable`.

In a mask, a coordinate is either ‘inside’ or ‘outside’ of the ROI. Two masks can be combined using boolean operations

$$\neg : \quad (\neg f)(x) = \overline{f(x)} \quad (13)$$

$$\cap : \quad (f \cap g)(x) = f(x) \wedge g(x) \quad (14)$$

$$\cup : \quad (f \cup g)(x) = f(x) \vee g(x) \quad (15)$$

$$\setminus : \quad (f \setminus g)(x) = f(x) \wedge \overline{g(x)} \quad (16)$$

which, using  $\perp = 0$  and  $\top = 1$ , is consistent with the above definitions.

A ROI may only specify its non-zero coordinate value pairs explicitly. All other coordinates are then implicitly considered zero that is ‘outside’ of the ROI. Per definition, a discrete ROI may be transferred to real space by assigning all real coordinates in the rounded environment of a discrete coordinate to the value of the discrete coordinate. This is equivalent to nearest neighbor interpolation. Accordingly, this transfer can be generalized to defining an  $n$ -dimensional region of interest in real space as the set of the Voronoi regions of explicitly specified coordinate value pairs.

## 2 The current status in `ImgLib2`

Lee Kamentsky has created the package `net.imglib2.roi` containing several concrete classes that all implement a generic `RegionOfInterest` interface that implements a `RealRandomAccessibleInterval` with a `BitType`. In terms of the above definitions that corresponds to a bounded mask.