



## Blatt 4

**Aufgabe 4.1** (4 Punkte) Implementieren Sie einen Hit-And-Missoperator in JAI. Sie können z.B. zunächst den Abschnitt 14.3 des JAI Users guide lesen (siehe Material zur Vorlesung). Sie benötigen allerdings nur Teile, da der Hit-And-Missoperator aus vorhandenen Operatoren zusammengesetzt werden kann. Ein Beispiel hierfür ist auf der Web-Page zur Vorlesung für die Opening-Operation zu finden. Sie müssen zwei Klassen implementieren, die eine leitet `OperationDescriptorImpl` ab, die andere implementiert das Interface `RenderedImageFactory`. Um einen Operator per `JAI.create(...)` benutzen zu können, muß er registriert sein. Dies kann wie in `testMyopeningOp.java` explizit per Hand gemacht werden, oder aber über eine Datei `registry.jai`, die in dem Directory `META-INF` im `CASSPATH` liegen muß.

**Aufgabe 4.2** (2 Punkte) Beweisen Sie folgende Eigenschaften:

- (i)  $A \subset A \bullet B$ .
- (ii)  $C \subset D, \Rightarrow C \bullet B \subset D \bullet B$ .
- (iii)  $(A \bullet B) \bullet B = A \bullet B$ . (Idempotenz)

**Aufgabe 4.3** (4 Punkte) Entwickeln Sie (mit den bisher zur Verfügung stehenden Mittel) ein Verfahren, um die gelben Signale aus den Radarbildern zu extrahieren. Neben den originalen Farbbildern stehen auch die *hue-Bilder* zur Verfügung. Der hue-Wert repräsentiert den Farbton, also z.B. rot oder grün oder gelb, eines Bildpunktes. Wie können Sie “Rauschen” entfernen oder die “richtigen” Vogelsignale extrahieren?

**Hinweis:** Leider ist der JAI-Operator `Multiply` – zumindest unter Linux – fehlerhaft. EJAI stellt deswegen einen korrekten `Emultiply`-Operator zur Verfügung.