CONRAD Introduction

Marco Bögel, Room 09.155 marco.boegel@fau.de

Exercise

Read through the following tutorial. Use it to fill in the missing lines in *Intro.java*.

Tutorial

Getting Started

In order to install the CONRAD Framework, please read through the *CONRAD_installation.pdf* and follow the instructions.
Navigate to *src→edu.stanford.rsl→tutorial→dmip* and open *Intro.java*

How can I run my program?

Right-click on the class you want to run (provided a *public static void main(String[] args)* method exists), choose *Run As → Java Application* or *Debug As → Java Application*.

Let's start programming

Vectors

```
//Create a vector
SimpleVector v = new SimpleVector(1.0, 2.0, 3.0);

//Min element of a vector
double min = v.min();

//Euclidean norm of a vector
double normL2 = v.norm(VectorNormType.VEC_NORM_L2);
```

Matrices

```
//Create a 3x4 matrix
SimpleMatrix M = new SimpleMatrix(3,4);
```

```
//Access an element
double val = M.getElement(1,2);

//Access a column vector of M
SimpleVector col = M.getCol(2);

//Multiply M*v
SimpleVector result = SimpleOperators.multiply(M, v);

//Frobeniusnorm of M
double normF = M.norm(MatrixNormType.MAT_NORM_FROBENIUS);

//SVD(M)
DecompositionSVD svd = new DecompositionSVD(M);
```

Signals

```
//Use double arrays to represent a signal
double[] x = new double{...};
double[] y = new double{...};

//Plot the signal
VisualizationUtil.createPlot(x, y,"Title","x","y").show();
```

Images

```
//Create an image of size 128x128
Grid2D image = new Grid2D(128, 128);

//Set a pixel
image.setAtIndex(x, y, value);

//create GUI to inspect pixel values, etc.
ImageJ ij = new ImageJ();

//display the image
image.show("Title");

//Load an Image from disk
String filename = "C:/.../image.png";
Grid2D imageDisk = ImageUtil.wrapImagePlus(IJ.openImage(filename));

//Convert Grid images to ImageJ ImageProcessors
FloatProcessor imgProc = ImageUtil.wrapGrid2D(imageDisk);
```

```
//Perform a convolution
Convolver conv = new Convolver();
float[] kernel = new float[kernelWidth*kernelHeight];
...
conv.convolve(imgProc, kernel, kernelWidth, kernelHeight);
```