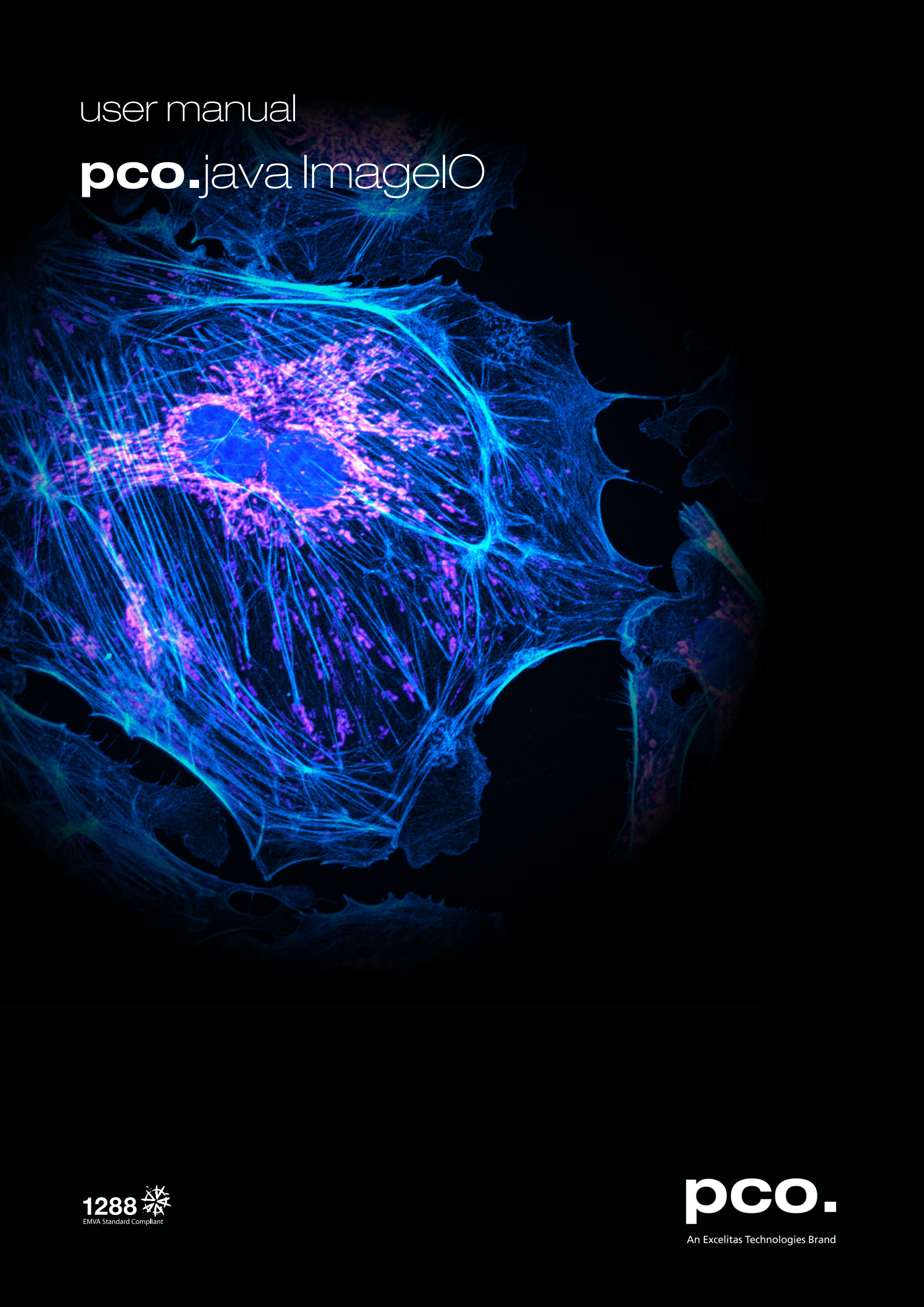


user manual

pco.java ImageIO



1288 
EMVA Standard Compliant

pco.

An Excelitas Technologies Brand

**PCO asks you to carefully read and follow the instructions in this document.
For any questions or comments, please feel free to contact us at any time.**



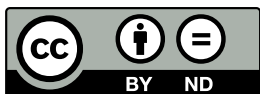
An Excelitas Technologies Brand

telephone:	+49 (0) 9441 2005 50
fax:	+49 (0) 9441 2005 20
postal address:	Excelitas PCO GmbH Donaupark 11 93309 Kelheim, Germany
email:	info@pco.de
web:	www.pco.de

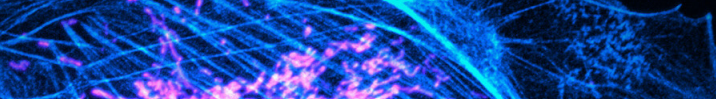
pco.java ImageIO user manual 2.0.1

Released December 2021

©Copyright Excelitas PCO GmbH



This work is licensed under the Creative Commons Attribution-NoDerivatives 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nd/4.0/> or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.



Contents

1 General	4
1.1 Installation	4
1.2 Basic Usage	4
1.3 Advanced Usage	5
2 Example	6
2.1 User manual	6

1 General

The **pco.java** ImageIO package provides a Java ImageIO API reader to display the raw images recorded by **PCO** cameras and the images loaded from the proprietary B16 file format.

Provides the ability to retrieve **PCO**-specific metadata also from the standard TIFF files. Depends on the TIFF package of [TwelveMonkeys ImageIO](#).

1.1 Installation

The project is built using **Apache Maven**.

Maven artifacts on the Maven Central Repository: <https://repo1.maven.org/maven2/de/pco/>
Parent pom.xml: <https://search.maven.org/artifact/de.pco/pco/2.0.0/pom>

Group-ID: **de.pco**

Artifact-ID (Maven modules):

- **pco** – Parent pom.xml
- **pco-common** – Common sources for **pco-camera** and **pco-imageio**
- **pco-camera** – Java interface to control the **PCO** cameras
- **pco-imageio** – Java ImageIO plugin for the **PCO** cameras and B16 files
- **pco-example** – Example application

All jars are compiled and tested for at least Java 8.

If only ImageIO plugin is necessary, add to your pom.xml

```
<dependency>
  <groupId>de.pco</groupId>
  <artifactId>pco-imageio</artifactId>
  <version>2.0.0</version>
</dependency>
```

Binaries and sources are also available directly from www.pco.de.

1.2 Basic Usage

pco-imageio artifact offers a method to get the `BufferedImage` from the data, recorded using **pco-camera** module.

```
ImageData imageData = ... // see pco-camera manual
RawImageReader reader = new RawImageReader();
RawImageInputStream riis = new RawImageInputStream(imageData);
reader.setInput(riis);
BufferedImage image = reader.read(0);
```

pco-imageio artifact contains also the ImageIO plugin for B16 files. After including `pco-common-2.0.0.jar` and `pco-imageio-2.0.0.jar` on the classpath, the standard method of loading image files will also be available for B16.

```
File file = new File("image.b16");
BufferedImage image = ImageIO.read(file);
```

1.3 Advanced Usage

To retrieve the **PCO** metadata from the B16 files

```
B16ImageReader reader = new B16ImageReader();
ImageInputStream iis = ImageIO.createImageInputStream(file);
reader.setInput(iis);
BufferedImage image = reader.read(0);
PcoIIOMetadata metadata = (PcoIIOMetadata) reader.getImageMetadata(0);
```

To retrieve the **PCO** metadata from the TIFF files

```
TIFFImageReader reader = new TIFFImageReader();
...
TIFFImageMetadata tim =
    (TIFFImageMetadata) reader.getImageMetadata(0);
B16ImageWriter writer = new B16ImageWriter();
ImageTypeSpecifier imageType = null;
PcoIIOMetadata metadata = null;
imageType = reader.getImageTypes(0).next();
metadata = (PcoIIOMetadata) writer.convertImageMetadata(tim,
    imageType, null);
```

2 Example

pco-example artifact contains an example GUI application. Its purpose is to get the images from the camera, to display them (including the additional metadata from the camera) and to save a particular image into B16 file. It also enables the user to load and display B16 and TIFF files, edit the metadata from and save the file again.

Run the example application (by installed Java) with the mere double-click on `pco-example/pco-example-2.0.0-jar-with-dependencies.jar` or from the console using

```
java -jar pco-example-2.0.0-jar-with-dependencies.jar
```

Alternatively, get the maven **pco-example** artifact by adding to your pom.xml

```
<dependency>
  <groupId>de.pco</groupId>
  <artifactId>pco-example</artifactId>
  <version>2.0.0</version>
</dependency>
```

The application depends both on **pco-camera** and on **pco-imageio** artifacts. Source codes of the application are in the package `de.pco.example`, the main class is `GuiExample`.

Then you can start the example application from your own main method by calling

```
GuiExample.main(null);
```

2.1 User manual

To open the camera connection click on the **CS** (Camera scanner) button. Select the number of images to be recorded and click on the **Record** button. Then you will be able to switch between the recorded images by the left and right arrow buttons.

On the right-hand side you see a column with the metadata obtained from the camera in addition with the image. You can change the metadata accordingly, e. g. put a commentary in the `TEXT` field.

Save the image and the corresponding metadata into B16 file by the menu option **File**→**Save**. You can load B16 files and also the 8-bit and 16-bit TIFF files by **File**→**Open**. If these files were created using PCO SW, they also contain the camera metadata and the current example application will display it also.

contact

pco europe

+49 9441 2005 50
info@pco.de
pco.de

pco america

+1 866 678 4566
info@pco-tech.com
pco-tech.com

pco asia

+65 6549 7054
info@pco-imaging.com
pco-imaging.com

pco china

+86 512 67634643
info@pco.cn
pco.cn



for application stories
please visit our website

pco.

An Excelitas Technologies Brand