IPSDK 1.5
What's new…

Reactiv'Ip

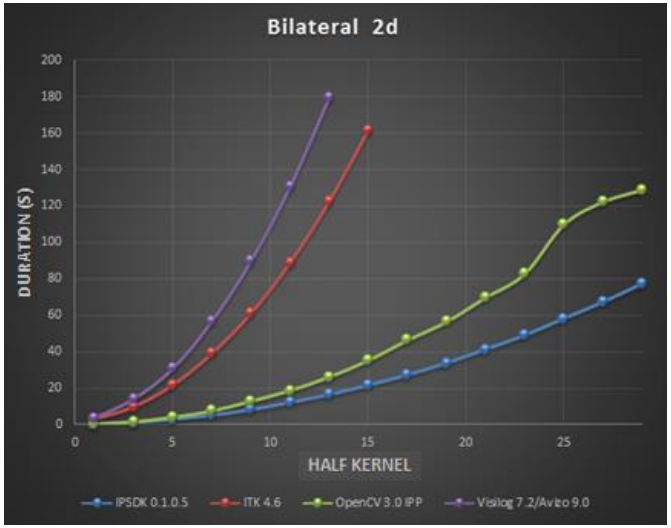
Reactiv'IP
1 place Firmin Gautier
38000 Grenoble - France
Web: www.reactivip.com

Contact:
M. Laurent Bernard
Laurent.bernard@reactivip.com
Tél.: +33 (0)4 76 70 97 29

The product
IPSDK

IPSDK – Full set of features

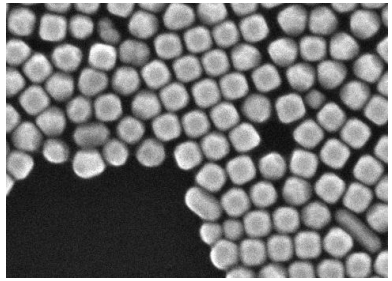
- Full set of image processing features (**2d/3d/sequence**)
  - Morphology (propagation, erosion, dilation, closing, …)
  - Filtering (median, gaussian, bilateral, anisotropic diffusion, NLM, …)
  - Thresholding (Otsu, tophat, …)
  - Statistics, Global measures, Binary operators, color transformations, histograms, textural measures,…
  - Connected components, true distance map, correlation, Hough transform, Watershed,…
  - Blob analysis: many measures (form, intensity, orientation, …)

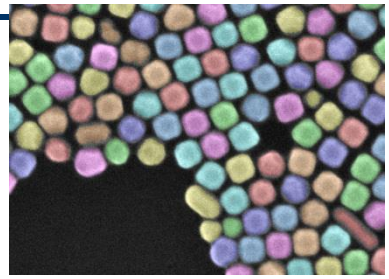


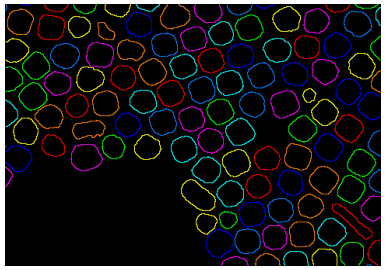
IPSDK
Smart Image Processing

IPSDK Library - Scripting

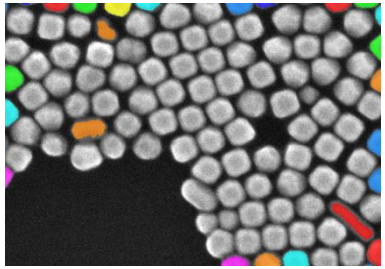
```
# import of IPSDK library
import PyIPSDK
import PyIPSDK.IPSDKIPLAdvancedMorphology as advmorpho
import PyIPSDK.IPSDKIPLBinarization as bin
import PyIPSDK.IPSDKIPLShapeAnalysis as shapeanalysis

# opening of input images
inGreyImg = PyIPSDK.loadTiffImageFile("G:/Sample/blobs_483x348_UInt8.tif")

# automatic binarization
inBinImg,  thrValue = bin.otsuThresholdImg(inGreyImg)

# connected component analysis
inLabelImg2d = advmorpho.connectedComponent2dImg(inBinImg)

# extraction of associated shapes
inShape2dColl = shapeanalysis.labelContourExtraction2d(inLabelImg2d)

# definition of proceeded measure
inMsrInfoSet = PyIPSDK.createMeasureInfoSet2d()
PyIPSDK.createMeasureInfo(inMsrInfoSet, "Circularity2dMsr")

# shape analysis
outMeasureSet = shapeanalysis.shapeAnalysis2d(inGreyImg,  inShape2dColl,  inMsrInfoSet)

# shape filtering computation
outImg = shapeanalysis.shapeFiltering2dImg(inLabelImg2d,  inGreyImg,  "Circularity2dMsr < 0.9")
```







Label Index	Circularity2d	Area2d	Perimeter2d
1	0,73	19,5	21,3
2	0,65	20,0	24,5
3	0,77	260,0	74,6
4	0,82	300,5	74,8
5	0,85	447,0	88,6
6	0,74	220,0	70,8
...	...	...	...
111	0,86	435,0	85,7
112	0,78	339,5	83,2
113	0,59	6,5	15,4



Evolutions

- New supported OS,
- Better integration in Python IDE, new tools,
- New image processing features,
- Perspectives…

New compatible OS

IPSDK 1.5 is now usable on:

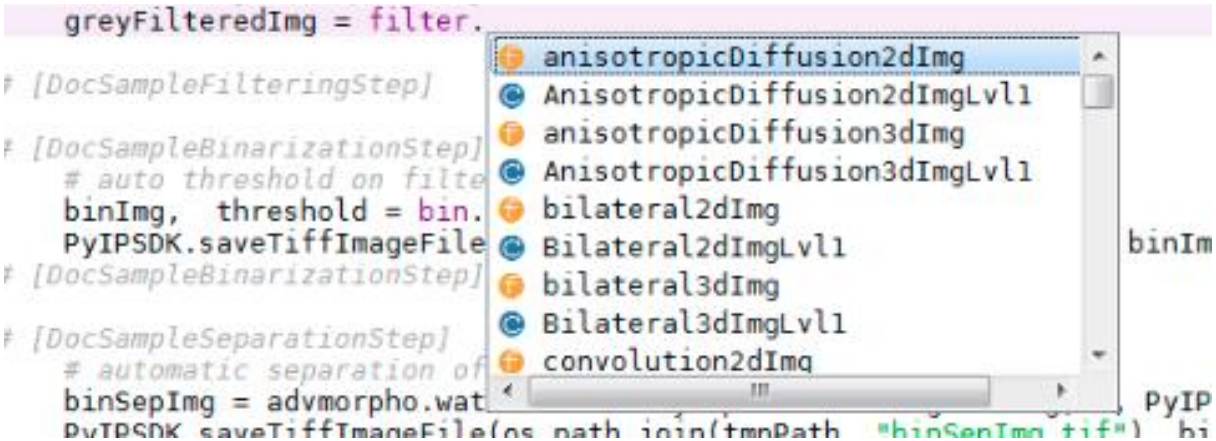
- Windows 7, 8, 10
- Linux Ubuntu 14.04 and more recent
- Linux Suse
- Linux Debian 8

Compilers: gcc and Visual Studio 2015 express

Python 3.5

Simplifying use with Python

- Autocompletion



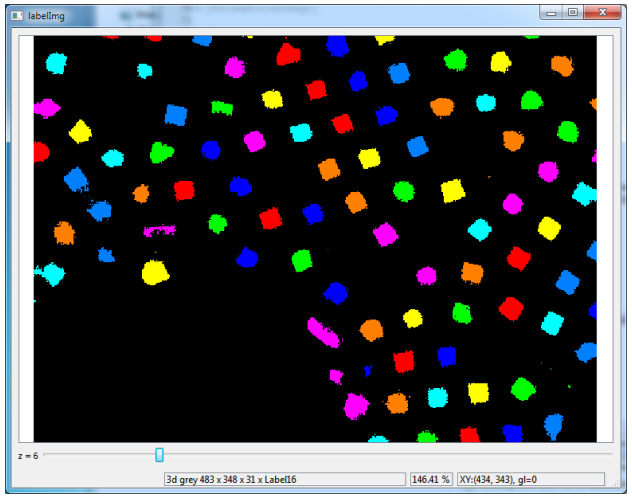
- Simplified access to structure fields

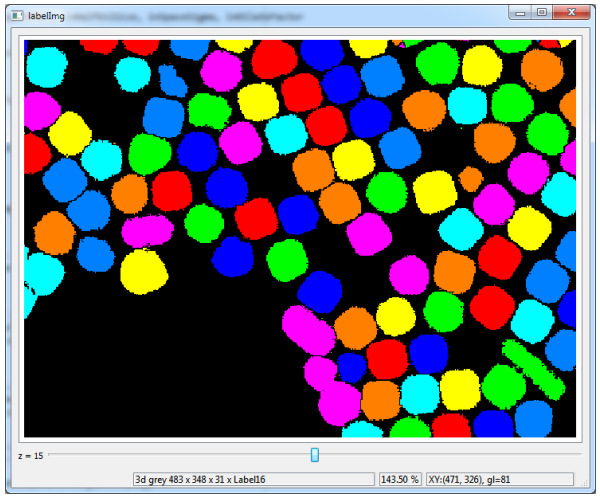
```
# statistics measurement
statsMsrResult = gblmsr.statsMsr3d(inImg)

# retrieve measurement results
mean = statsMsrResult.mean
```

Simplifying use with Python

Ability to **display** images in scripts to help implementation

```
import PyIPSDK.IPSDKUI as gui


labelImg = advmorpho.connectedComponent3dImg(binSepImg)
labelImgView = gui.displayImg(labelImg, "labelImg")
```



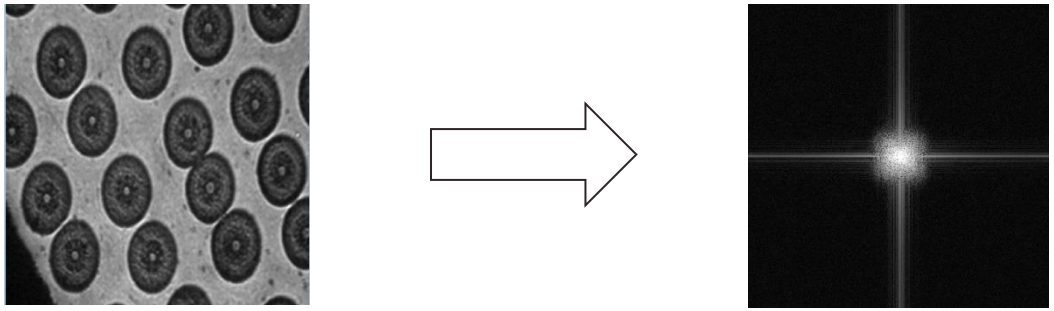


2D display, but possibility to navigate inside 3D or sequence image with the slider

New Image Processing Features

2D and 3D FFT:

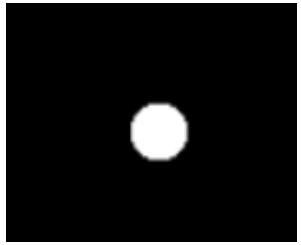
- Forward FFT,
- Backward FFT,
- Polar to cartesian coordinates transformation,
- Cartesian to polar coordinates transformation



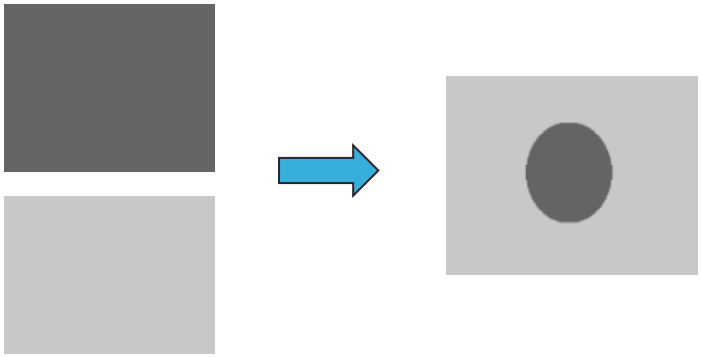
New Image Processing Features

Generation of an image using a formula string (formula 2d/3d Img)

```
outBinImg = arithm.formula2dImg("if(sqrt((x-cx)*(x-cx)+(y-cy)*(y-cy))<=10, True, False)",
                                InOptImg1=inImg)
```



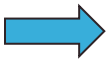
```
outImg = arithm.formula2dImg("if(sqrt((x-cx)*(x-cx)+(y-cy)*(y-cy))<=50, I1, I2)",
                             InOptImg1=inImg1, InOptImg2=inImg2)
```



New Image Processing Features

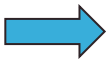
Generation of an image using a formula string (formula 2d/3d Img)

formula2dImg("mean(rect(I1, 3, 2))", InOptImg1=inImg)



Mean filter with rectangular neighborhood

formula2dImg("mean(circle(I1, 4))", InOptImg1=inImg)



Mean filter with circular neighborhood

New Image Processing Features

Resampling along Z axis

- **Resampling from regular slice spaces**
  - ➡ **Useful if a distance map must be calculated**

- **Resampling from non regular slice spaces**
  - ➡ **Useful in FIB or confocal microscopy**

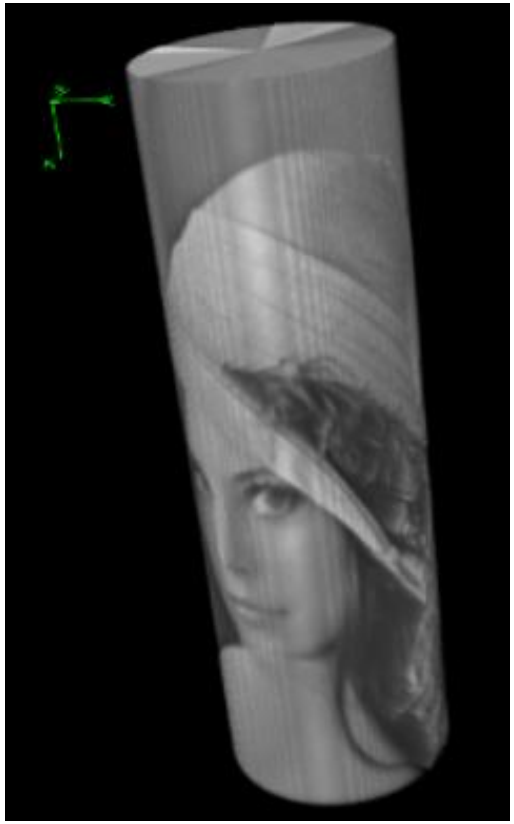
```
// specify distances between z-plans in input 3d image
boost::shared_ptr<attr::ZSteps> pZSteps = attr::ZSteps::createNode();
pZSteps->push_back<attr::ZSteps::StepsColl>(0.990f); // distance between image #0 a
pZSteps->push_back<attr::ZSteps::StepsColl>(1.015f); // distance between image #1 a
pZSteps->push_back<attr::ZSteps::StepsColl>(0.988f); // distance between image #2 a
pZSteps->push_back<attr::ZSteps::StepsColl>(0.994f); // distance between image #3 a
pZSteps->push_back<attr::ZSteps::StepsColl>(1.007f); // distance between image #4 a
pZSteps->push_back<attr::ZSteps::StepsColl>(0.995f); // distance between image #5 a
pZSteps->push_back<attr::ZSteps::StepsColl>(0.997f); // distance between image #6 a
```

```
resampleCustomZStack3dImg(pInImg, pZSteps, 8, attr::eInterpolationPolicy::eIP_Linear);
```

New Image Processing Features

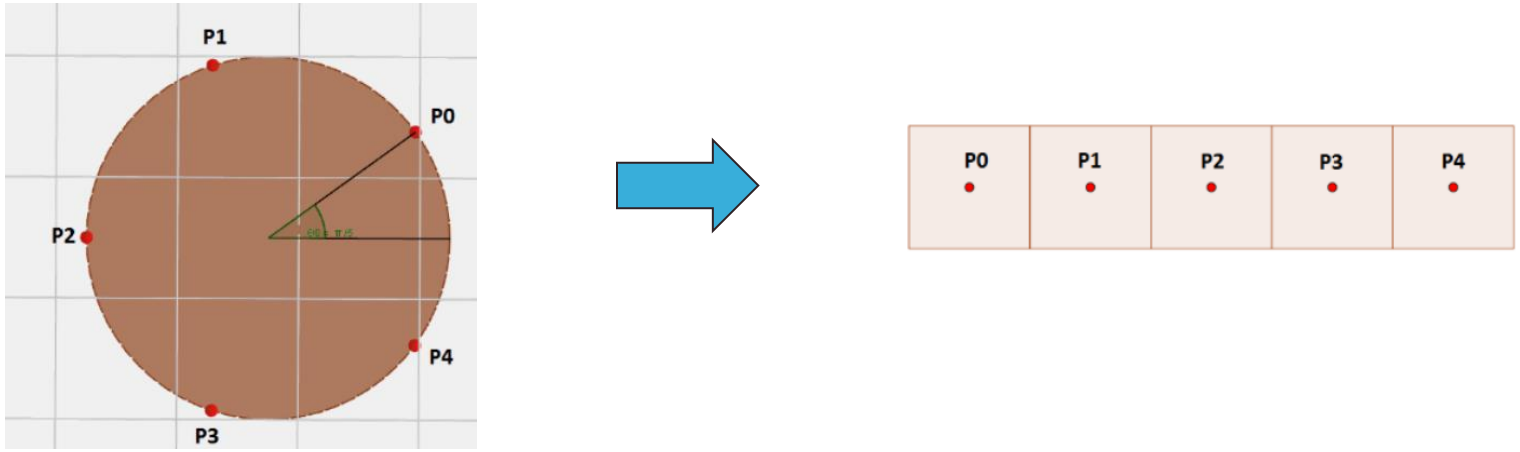
Image cylinder unrolling:

Algorithm unfolding a cylinder contained in a 3D input image into a 2D output image



New Image Processing Features

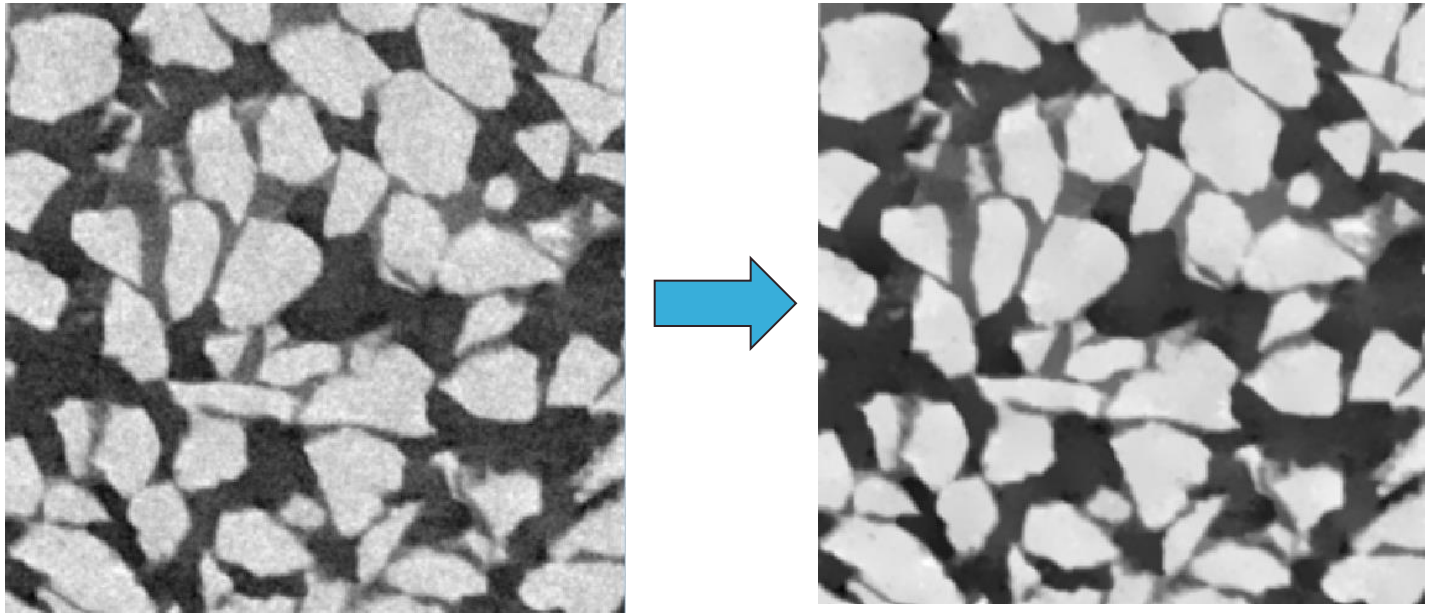
Image cylinder unrolling:



- Several resampling options (linear, cubic, nearest),
- Thickness size editable,
- Several options for the thickness integration (min, max, mean)

New Image Processing Features

- **Despeckle 2D and 3D**
  - Remplace aberrant pixels by the median value of their neighborhoods
- **Sobel gradient 2D and 3D**
- **Patch-based bilateral smoothing 2D and 3D**
  - Optimized version of the Non Local Mean algorithm



New statistical measures

- Local entropy
- Local energy
- Local histogram module
- Law’s texture energy measures
- Masked histogram

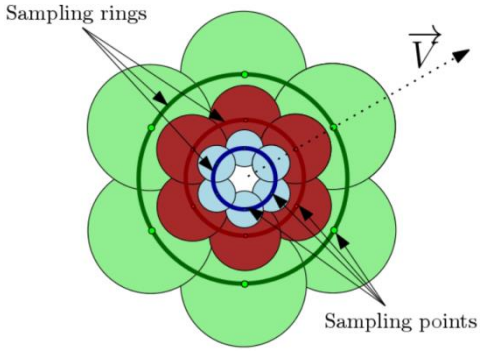
  Histogram of the portion of an image (portion is defined by a mask image)

Detection and registration

Detection:

- **Canny edges 2D detector** — Edge extraction (image 2D)
- **Canny edges 3D delector** — Surface extraction (image 3D)
- **Harris corners** — Corner extraction (2D et 3D)

- **Extract grey signed features** — Extraction of grey signed features from image



Signature independent of rotation and zoom factor

Detection and registration

Registration:

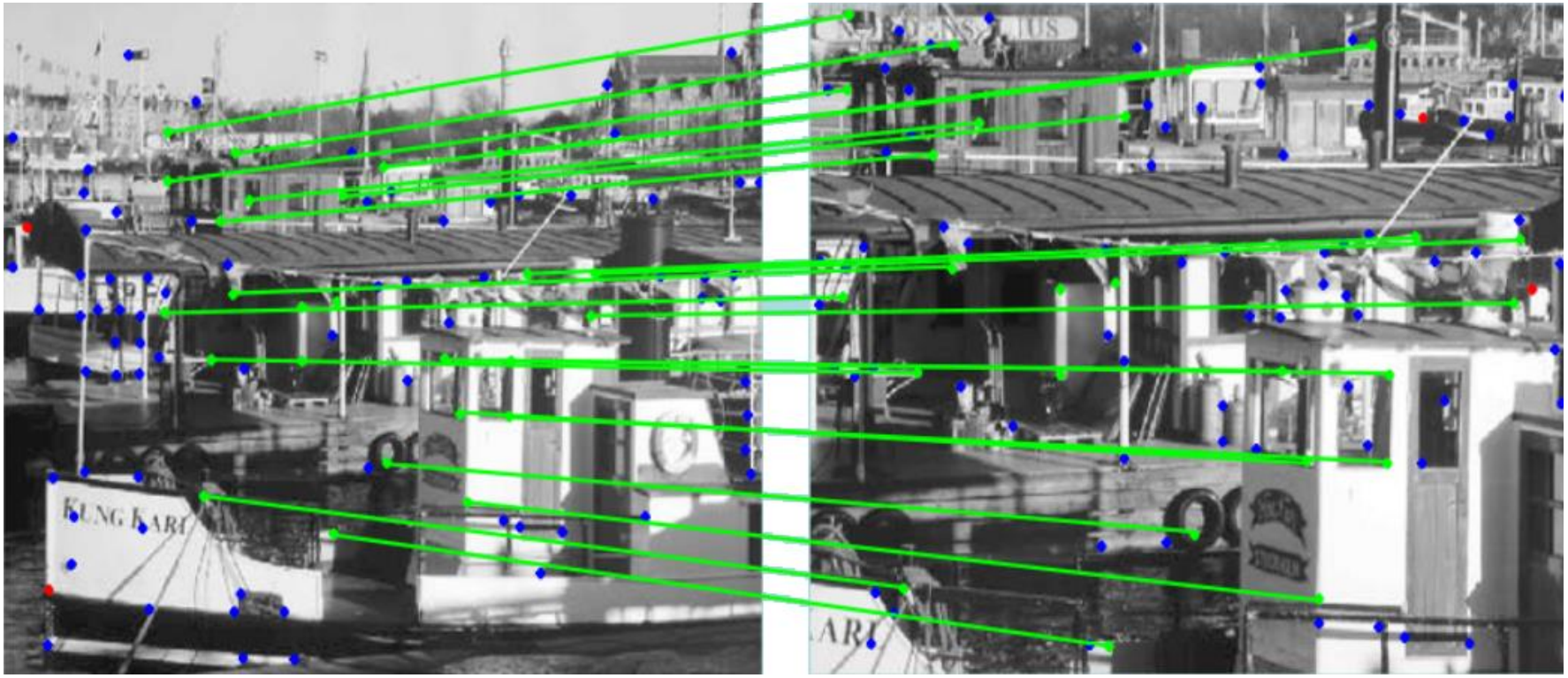
- **greySignedFeaturesImg** Computation of motion transform linking two images based on a grey signed features algorithm

→ Translation, rotation, Homothety

Detection and registration

In blue: All corner points detected

In Green: All corner points linked



Classification by pixel

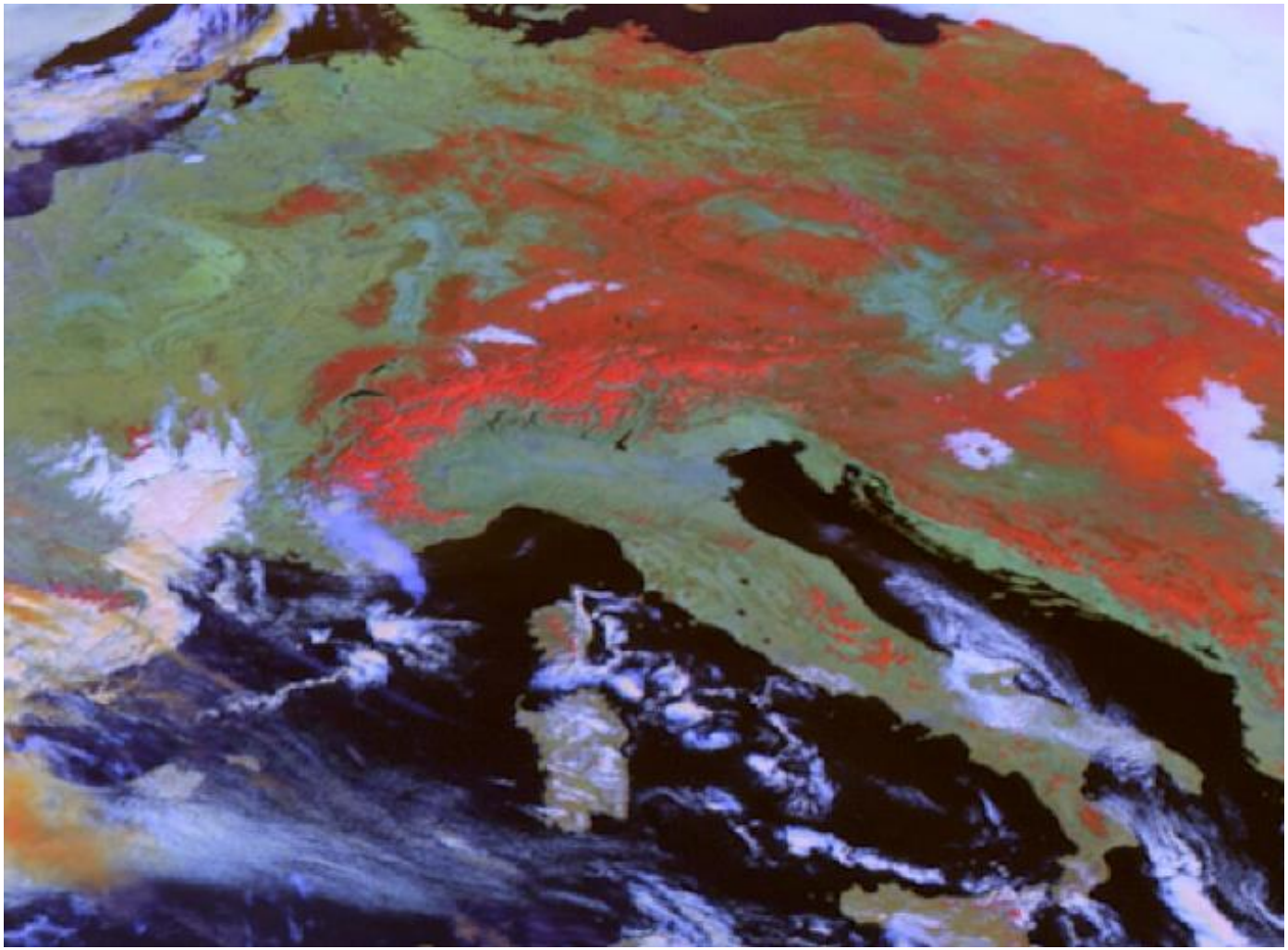
- **K-Means** 2D and 3D pixel K-mean classification

- Classification by imposing the number of classes
- Classification by training (center of classes description)
  - ➡ **kMeansAssignImg, kMeansComputeCenters, kMeansImg,…**

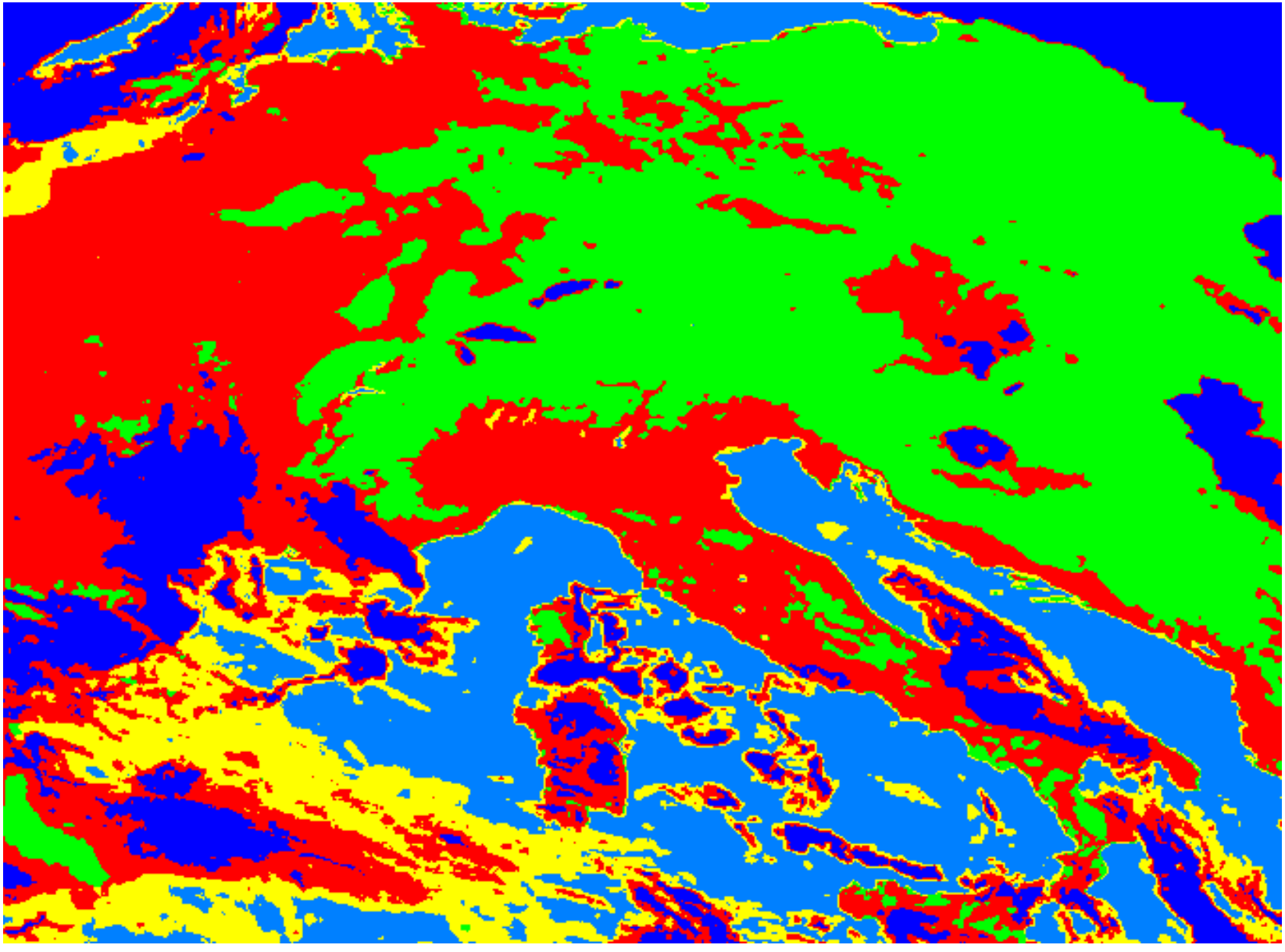
- Use:
  - With color images (or multi-spectral)
  - With vector of texture (one image by attribute)

Classification by pixel

Sample: Automatic classification using K-Mean with 5 classes



Classification by pixel



Perspectives

- GPU computation,
- Increase of scalability,

- And many other innovative image processing features!

Conclusion

IPSDK evaluation version is available

Feel free to ask it !

Reactiv'IP
1 place Firmin Gautier
38000 Grenoble - France
Web: www.reactivip.com

Contact:
M. Laurent Bernard
Laurent.bernard@reactivip.com
Tél.: +33 (0)4 76 70 97 29