Manual

Poisson Noise - An ImageJ Plugin

Janick Cardinale, janick@inf.ethz.ch
MOSAIC Group, www.mosaic.inf.ethz.ch

January 2008 (rev. November 2010)

1 Introduction

This small ImageJ plugin inserts Poisson distributed noise to each pixel of an image or an imagestack like it appear on images of a confocal microscopy due to photon counting. This might be useful to test algorithms on generated (artificial) data.

2 Installation

Copy the `mosaic_plugins.jar` file to the ImageJs `plugins` directory. Restart ImageJ. The plugin can be launched from the `Mosaic` submenu in ImageJs plugin menu.

The plugin handles 8-bit, 16-bit and 32-bit grayscale images or stacks.

Prerequisits: At least Java 5.0 and ImageJ 1.36.

3 How to use the plugin

Open an image or an image stack in ImageJ. In the `Plugin` menu, selecting the `Poisson Noise` entry in the `Mosaic` submenue starts the plugin. You might process only one slice or even the whole stack.

Please note that there is no undo for this operation.

4 Description

At each pixel, a Poisson distributed random variable is sampled and replaced with the original intensity value. The distribution parameter λ is set to the intensity of the original

pixel value (before noise was inserted). If $\lambda > 30$, the distribution is approximated with a Gaussian distribution $\mathcal{N}(\lambda + 0.5, \lambda)$.

The sampling algorithm of the random variable is implemented as followed [1]:

```
double p = 1;
int k = 0;
double vL = Math.exp(-aLambda);
do{
   k++;
   p *= mRandomGenerator.nextDouble();
}while(p >= vL);
return k - 1;
```

5 Disclaimer

IN NO EVENT SHALL THE ETH BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE ETH HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. THE ETH SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE ETH HAS NO OBLIGATIONS TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

References

[1] D. E. Knuth. *Art of Computer Programming, Volume 2: Seminumerical Algorithms (3rd Edition)*. Addison-Wesley Professional, November 1997.