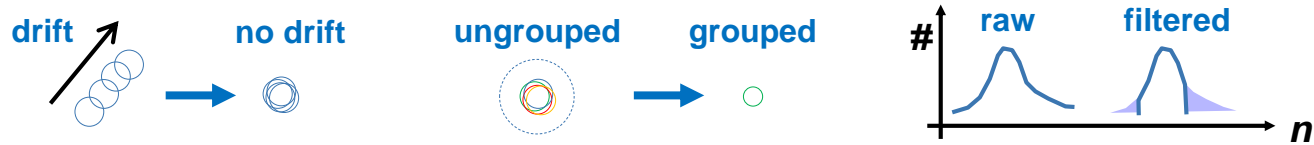# Nutshell – PALM Rendering

PALM Rendering
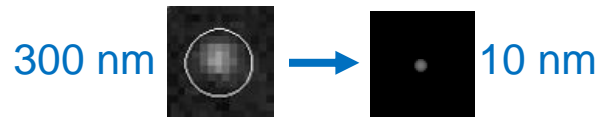*Scope and recommended previous knowledge.*

Here we cover **rendering** as one of the tools in PALM / dSTORM process.
Previous knowledge on PALM / dSTORM and ELYRA is recommended. In particular:
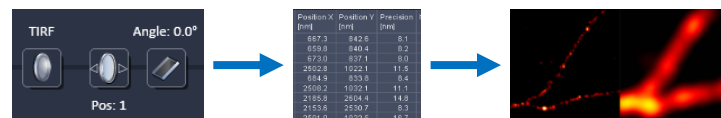
- Drift correction / Grouping / Filtering

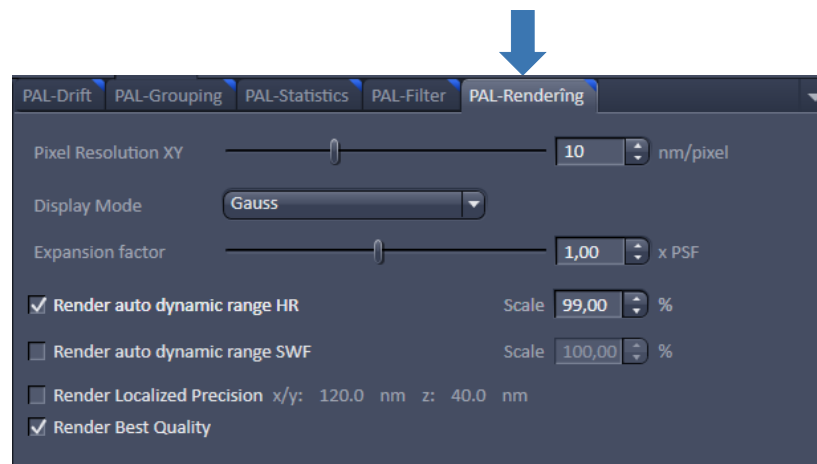- positioning accuracy and the involved scales of length

300 nm 10 nm

- basic knowledge of the PALM / dSTORM workflow in ELYRA
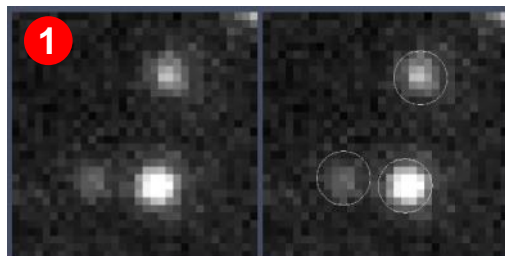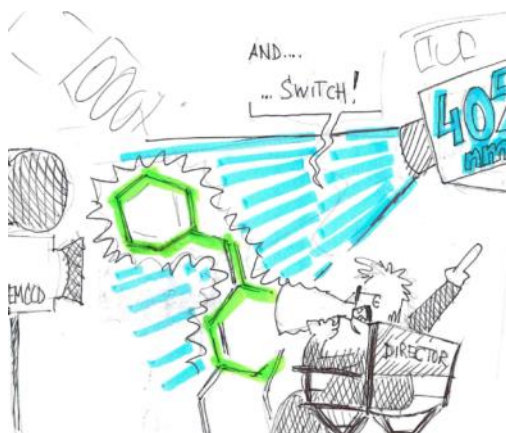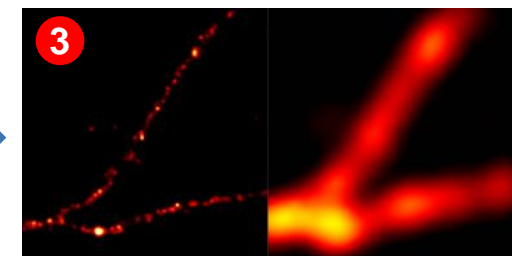
Overview

PALM Rendering - the last step in the process
Simply put, rendering is for viewing only...

Position X [nm]	Position Y [nm]	Precision [nm]	Number Photons	Ba
667.3	842.6	8.1	1966	
659.8	840.4	8.2	1968	
673.0	837.1	8.0	1741	
2502.8	1022.1	11.5	1066	
684.9	833.8	8.4	2411	
2508.2	1032.1	11.1	1806	
2185.8	2604.4	14.8	1534	
2153.6	2530.7	8.3	3396	
2501.0	1022.5	16.7	841	

Recording Data: Acquisition
Laser Intensities / Gain
Peak finding

Editing Data: Filtering
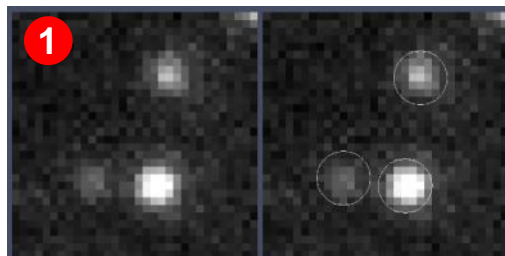Statistics
Drift / Group / Filtering

Viewing Data: Rendering
Display
Brightness / Contrast

PALM Rendering - the last step in the process
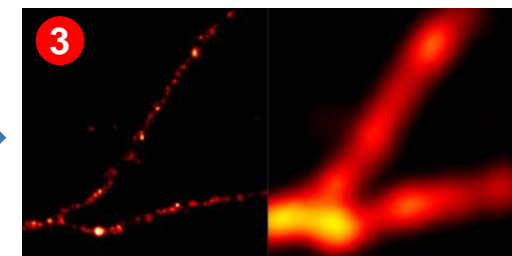*Rendering generates an image from the table.*
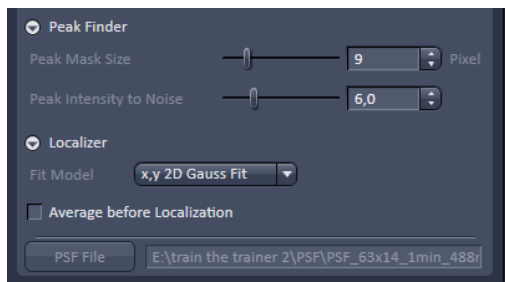*Rendering does <u>not</u> affect the table.*

Single emitter patterns
(original data)

Position and parameter table
(raw data)

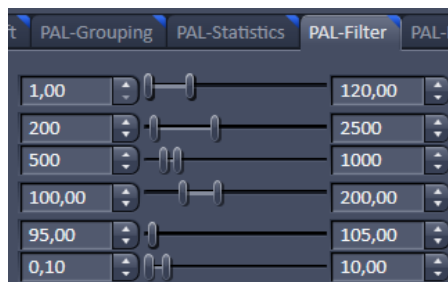Rendered image
(visualization)

Acquisition settings
Peak finding
Accounting for overlap
Has influence on the data
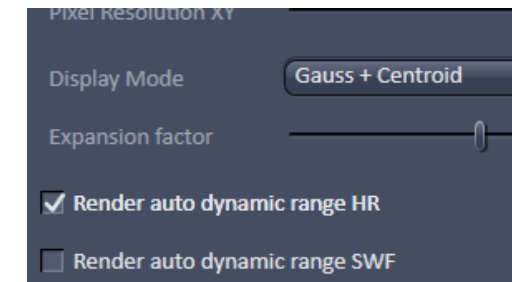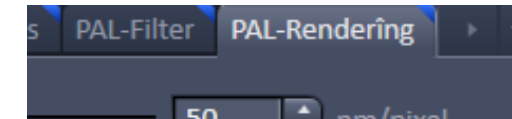Not covered in this presentation

Data analysis
Statistics
Drift / Group / Filtering
Has influence on the data
Not covered in this presentation

Rendering
Display
Brightness / Contrast
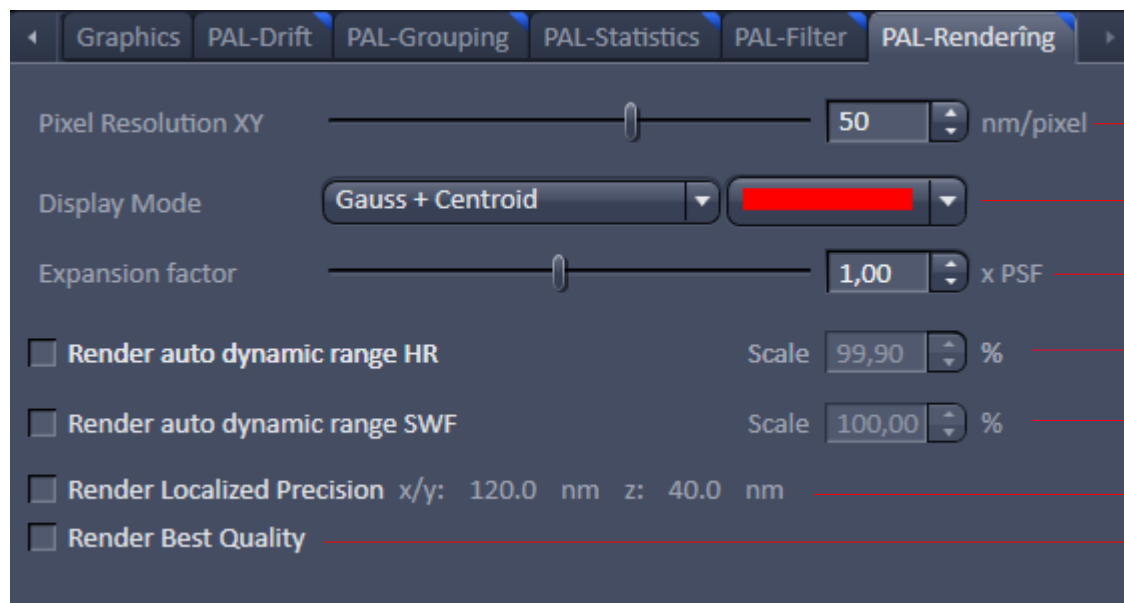**NO INFLUENCE on the data**
Mainstay of this presentation

Overview

Tip of the day: Read also the notes in the presentation for (even) more information

PALM Rendering – Parameters
Main parameters: Pixel size, Gaussian, etc...

In ZEN the **rendering options** are defined in the PAL-Rendering tab:

Tab	Controls

1 Sets the pixel size

2 Selects the display mode

3 Multiplies precision

4 Dynamic Range in HR

5 Dynamic Range in SWF

6 Renders all precision values to this number

7 Pixellation independent rendering

Tip of the day:	Try 'em all out. Don't panic! Raw data will not be affected

100 nm / px

20 nm / px

5 nm / px

Rendering Images from Table
Pixel size: Follow Nyquist reasoning
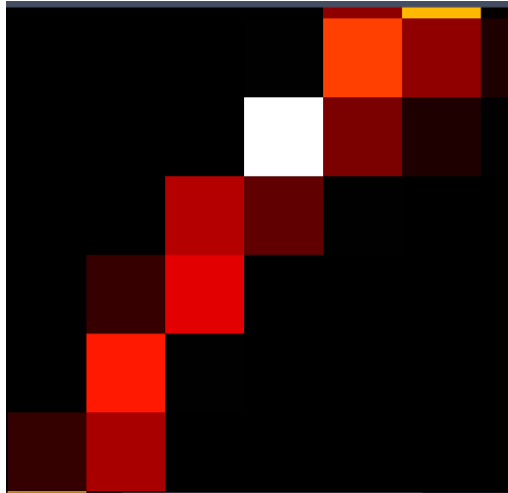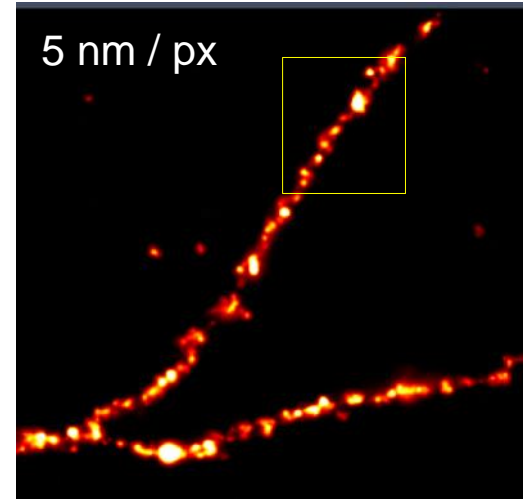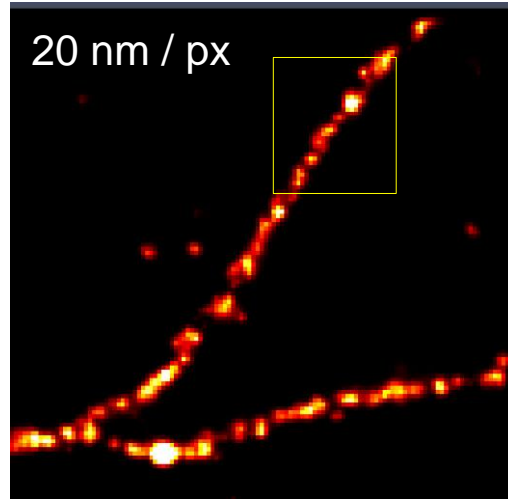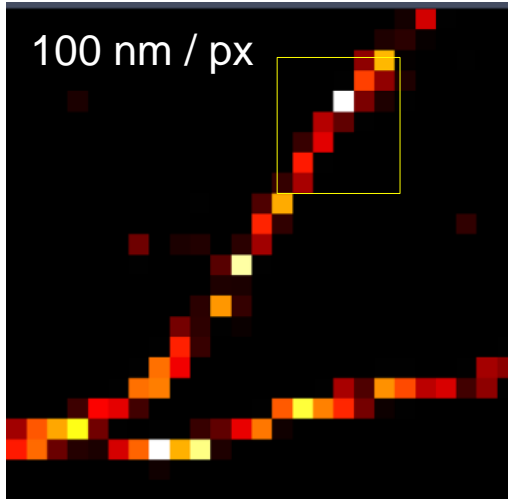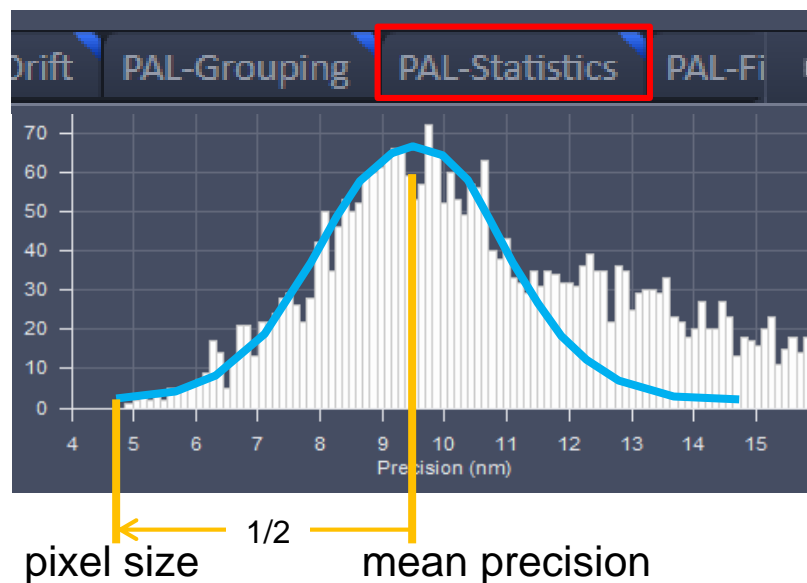
- **FAQ:** Why am I allowed to set the resolution* (pixel size) how I want?
  Answer: The image is rendered from a table - not related to the camera pixel size

- When you halve the pixel size, the files get 4 times bigger.

- If you don't know what to do: Look up precision** statistics, use half of the peak value.

1

pixel size — 1/2 — mean precision

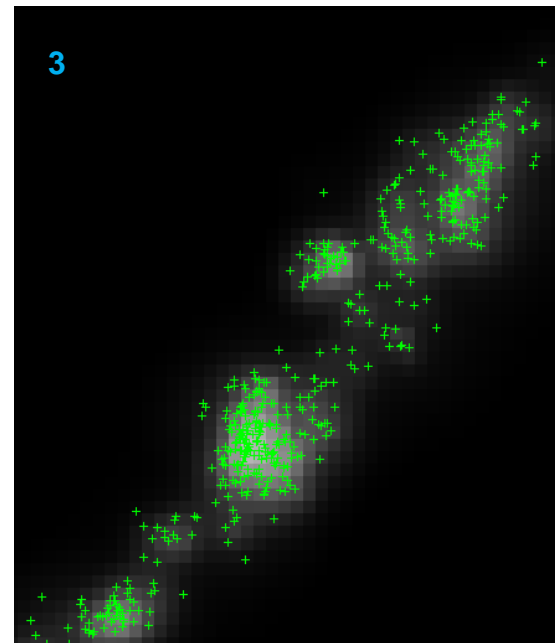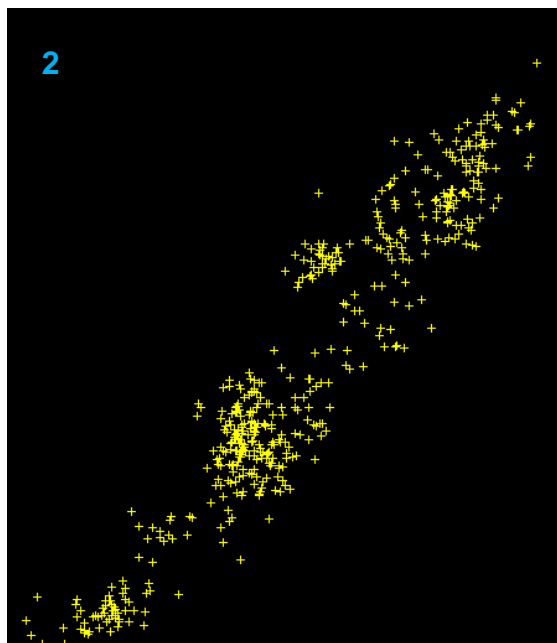* Attention: Some people use pixel size (or quantity) interchangeably with resolution.

** Attention: Precision is not equal to resolution. It's the best case, if the labelling density sufficiently high (see PAL-Statistics)

Rendering Images from Table
Display mode: Depends on the application

Centroid = dot

Special case: see next slides

10 nm

5 Patterns in the 10 x 10 pixel bin

Pixel Resolution XY		1 ⬍ nm/pixel
Display Mode	Molecule Density ▼	
Binning [Pixel]		10 ⬍
☑ Render auto dynamic range HR		Scale 99.90 ⬍ %
☐ Render auto dynamic range SWF		Scale 100.00 ⬍ %
☐ Render Best Quality		

Display mode: Special case – Molecule Density

Gauss — bright / px = 10 nm

mol. density — crowded / px = 10 x 1 nm

overlay

Brightness is <u>not necessarily</u> equal to quantity:

Few bright molecules <u>can</u> "outshine" many dim ones.

Many dim molecules <u>can</u> be more crowded than few bright ones.

=> The brightest spot is <u>not necessarily</u> the most crowded!

vs

60 W total
1 x 60 W
bright
isolated

10 W total
10 x 1 W
dim
crowded

Rendering Images from Table
Expansion Factor: A matter of confidence

The user can choose to expand the Gaussians by any factor. For example to display results with decreased resolution but with increased confidence level. (In the example above the confidence levels are increased from 68.2% using σ to 95.4% using 2σ.) **In case of doubt leave factor as 1x.**

Rendering Images from Table
Dynamic Range: HR and SWF

The dynamic range is (per default) **scaled** from the highest intensity to the lowest.
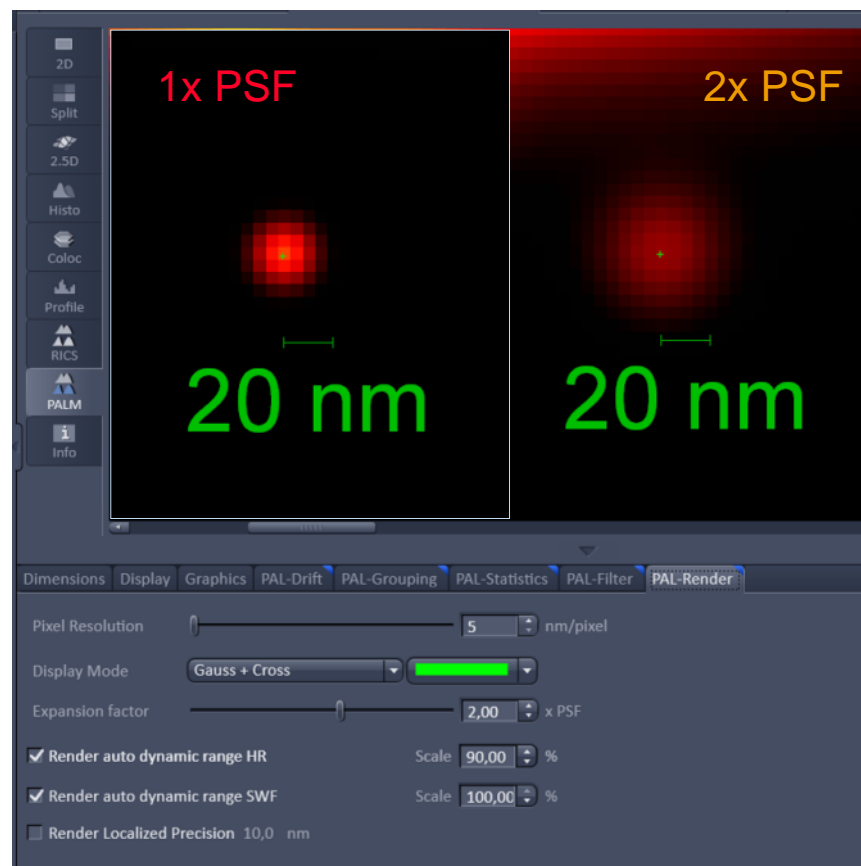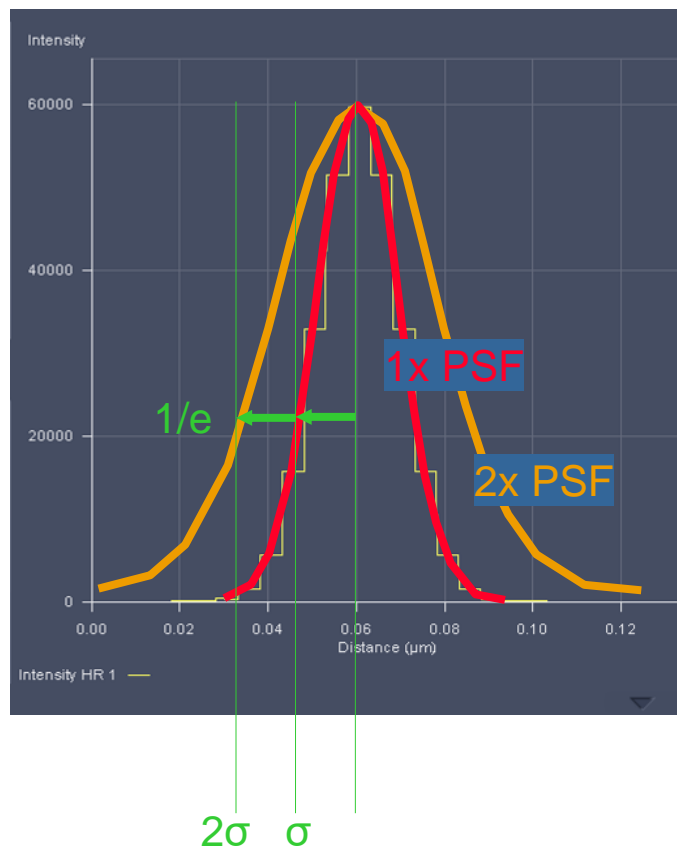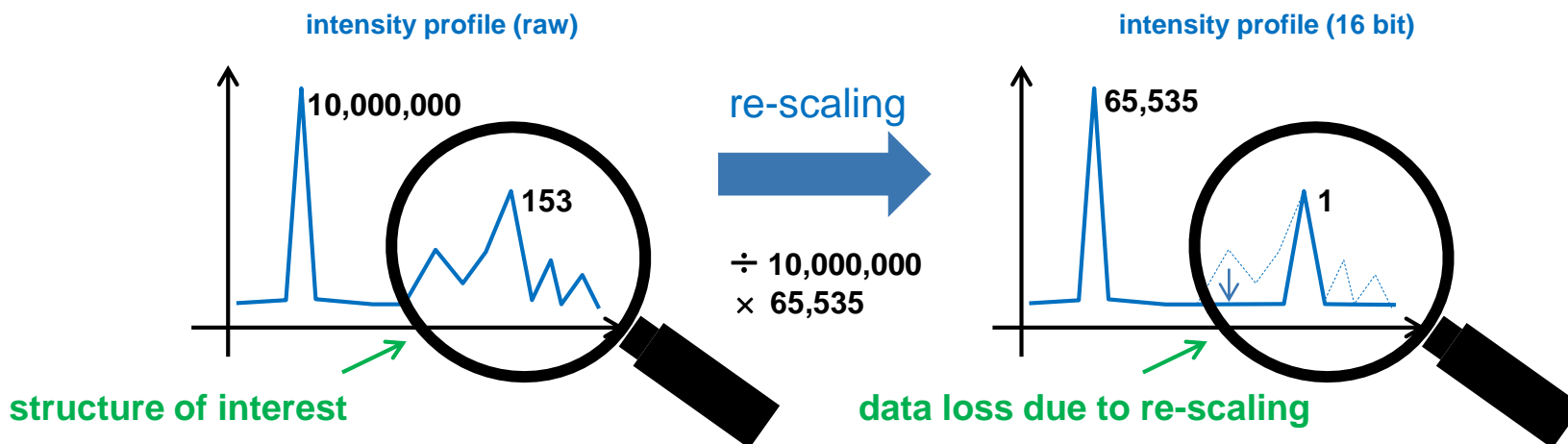
In 16-bit the **highest** greyscale value is 65535 (white) and the **lowest** is 0 (black)

This range is insufficient when one object (like a fiducial marker) – is more than 65,325 times brighter than the structures of interest.

intensity profile (raw)

10,000,000

153

re-scaling

$\div$ **10,000,000**
$\times$ **65,535**

intensity profile (16 bit)

65,535

1

structure of interest

data loss due to re-scaling

Remedy: Apply the scaling to the n-th percentile brightest objects.

| ☑ Render auto dynamic range HR | Scale | 97,00 | % |
| ☑ Render auto dynamic range SWF | Scale | 100,00 | % |

Note: Similar result can be obtained by adjusting brightness and contrast values. However this only works up to a certain limit (dictated by the 16-bit depth). If the brightness discrepancy is too large, the better option is to use the dynamic range option in the PAL-Render tab.
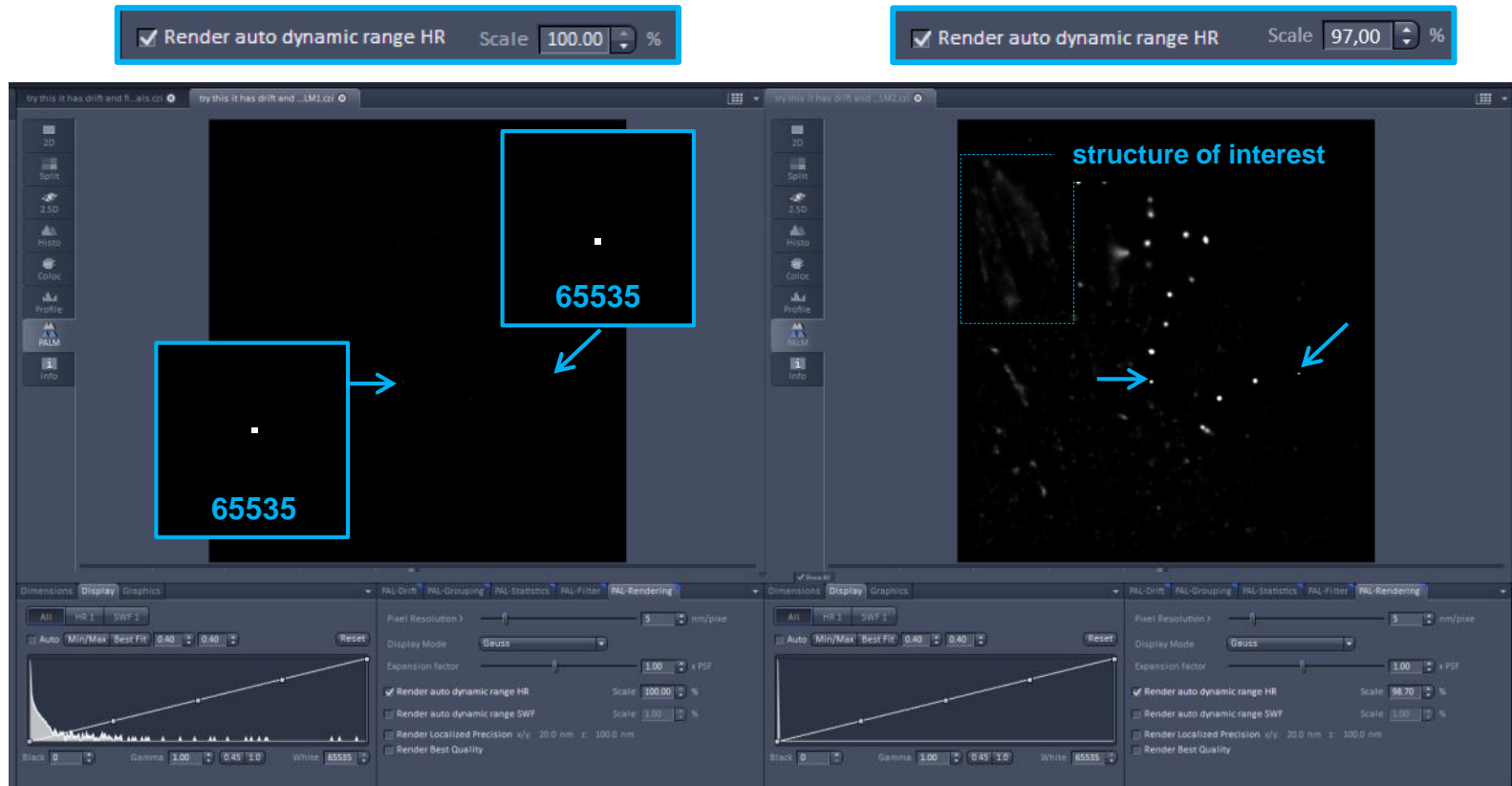
Rendering Images from Table
Dynamic Range: HR and SWF

Note: Similar result can be obtained by adjusting brightness and contrast values. However this only works up to a certain limit (dictated by the 16-bit depth). If the brightness discrepancy is too large, the better option is to use the dynamic range option in the PAL-Render tab.
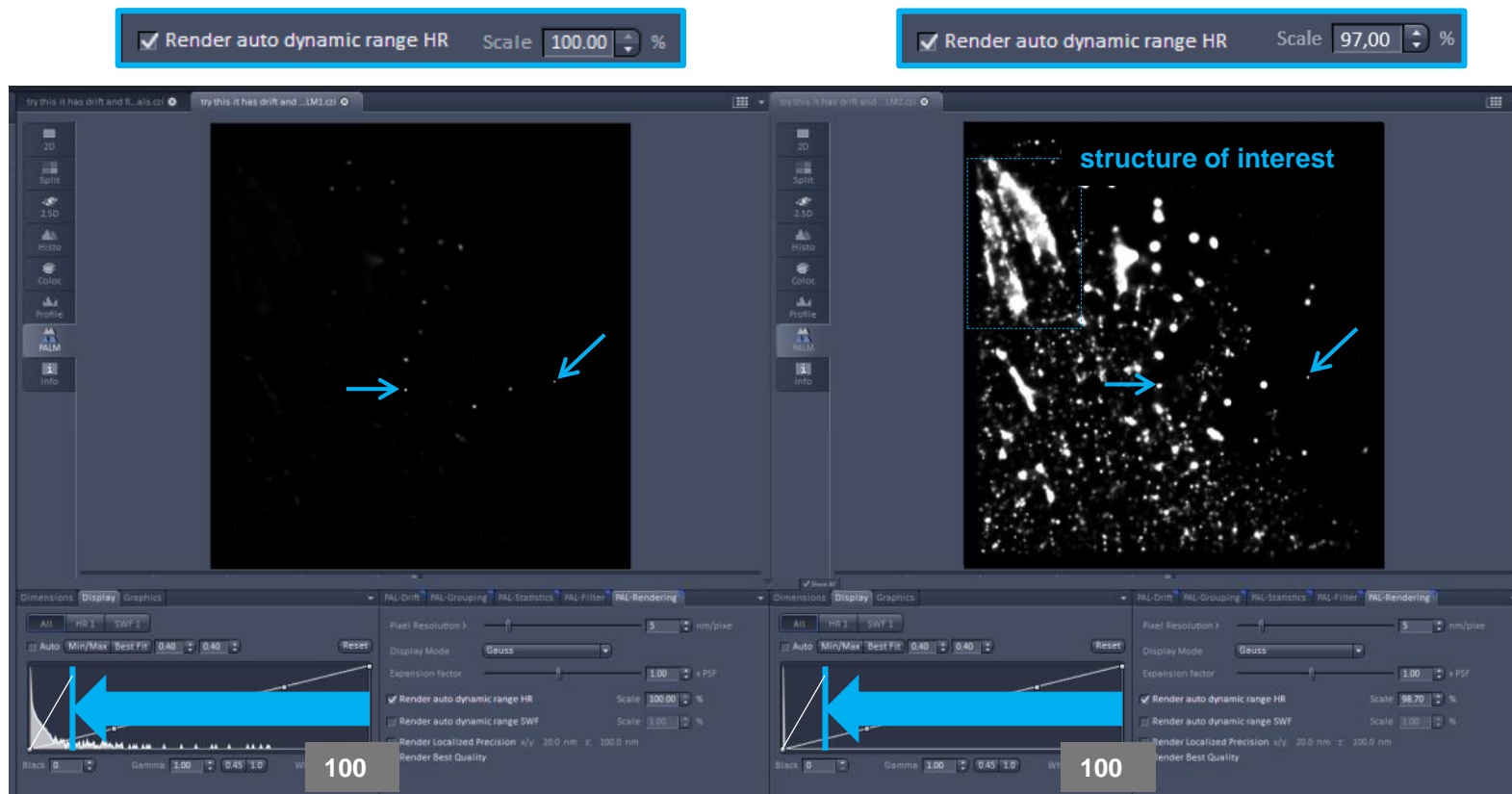
Rendering Images from Table
Dynamic Range: HR and SWF

Note: Similar result can be obtained by adjusting brightness and contrast values. However this only works up to a certain limit (dictated by the 16-bit depth). If the brightness discrepancy is too large, the better option is to use the dynamic range option in the PAL-Render tab.
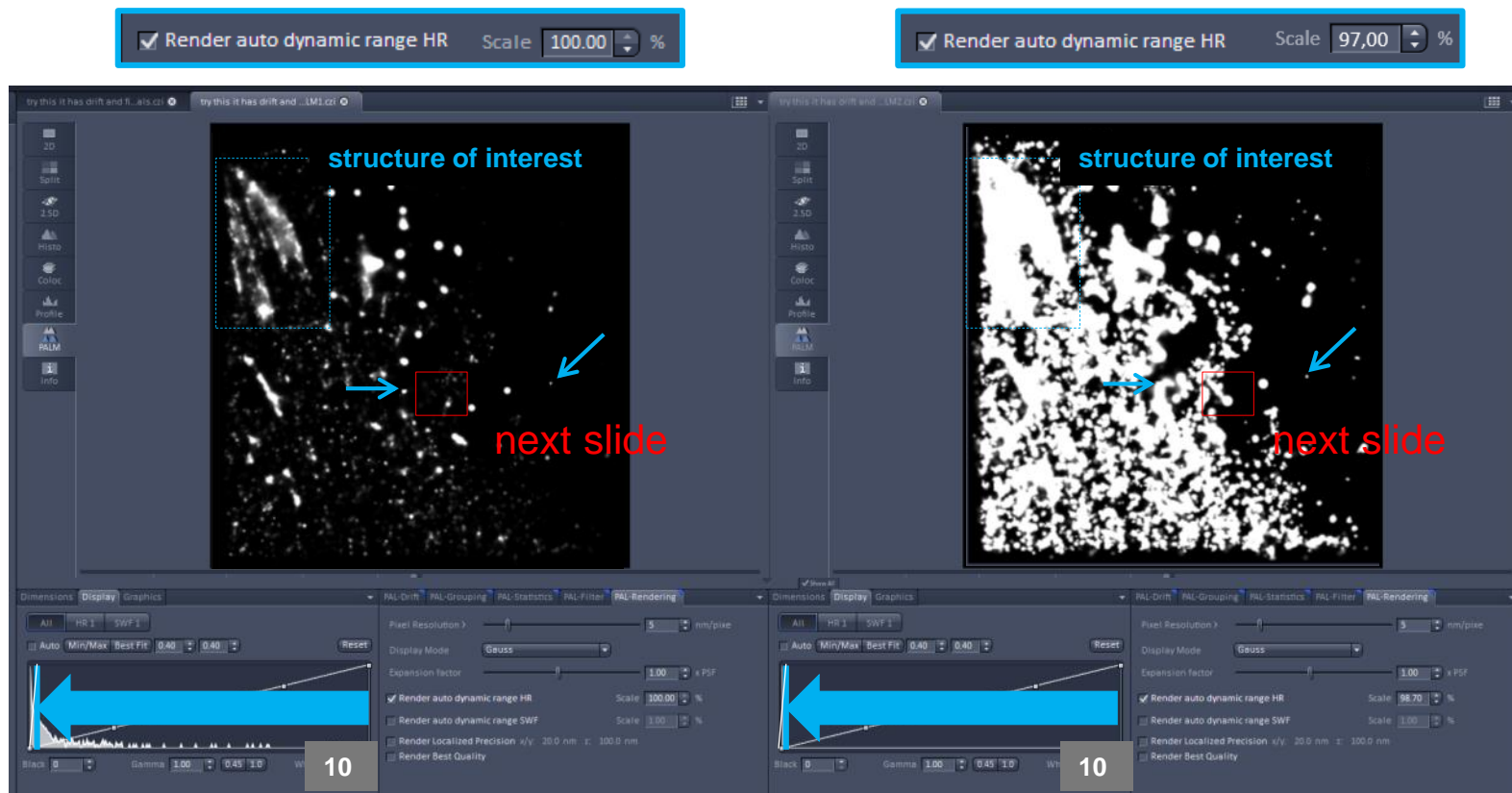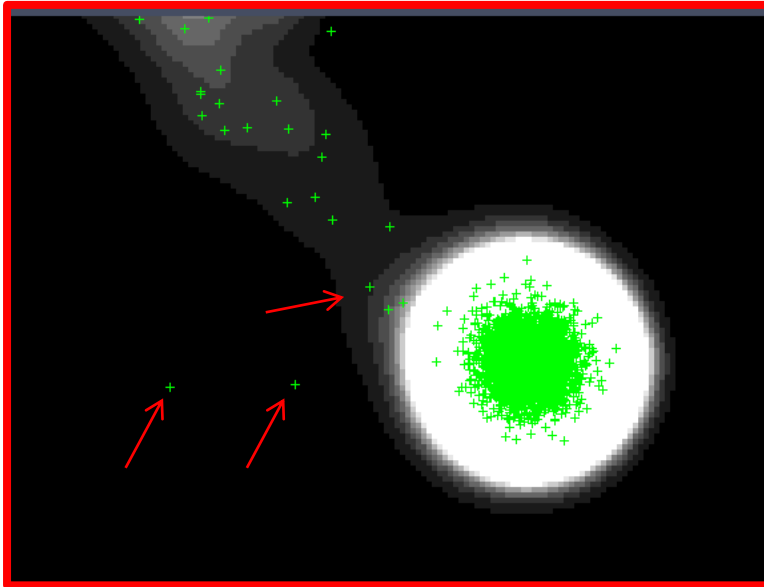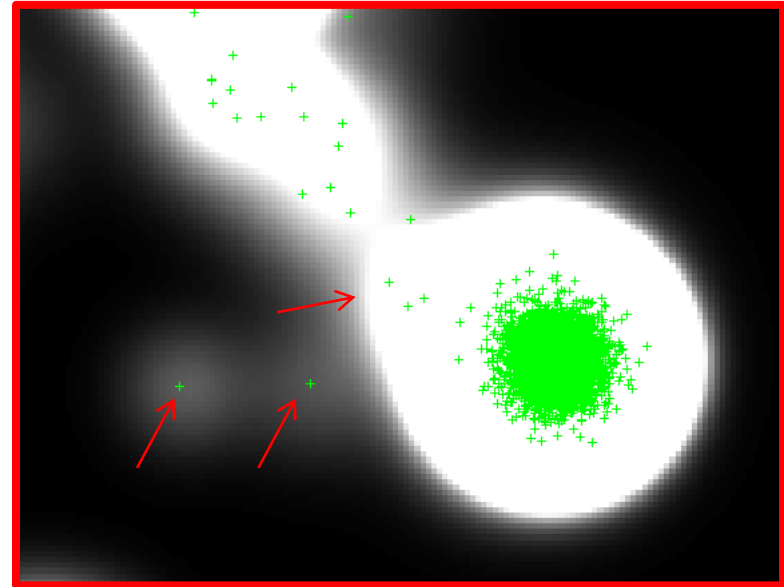
Rendering Images from Table
Dynamic Range: HR and SWF

Note: The left image has 4-bit depth (16 greyscale values), and the arrows point to dim objects that vanish due to the rescaling of the brightest object (their peak values after rescaling are below 1) the crosses for the center of mass are still there. In the right image with 8-bit depth (256 greyscale values) but scaled to the 97-th brightest percentile, the patterns remain clearly visible.

Render Localized Precision

Localized Precision renders all patterns with the largest localization precision value set in PAL-Filter. This value can be adjusted in the PAL-Filter tab and will be used even if left unticked in PAL-Filter). The same applies to precision in z.
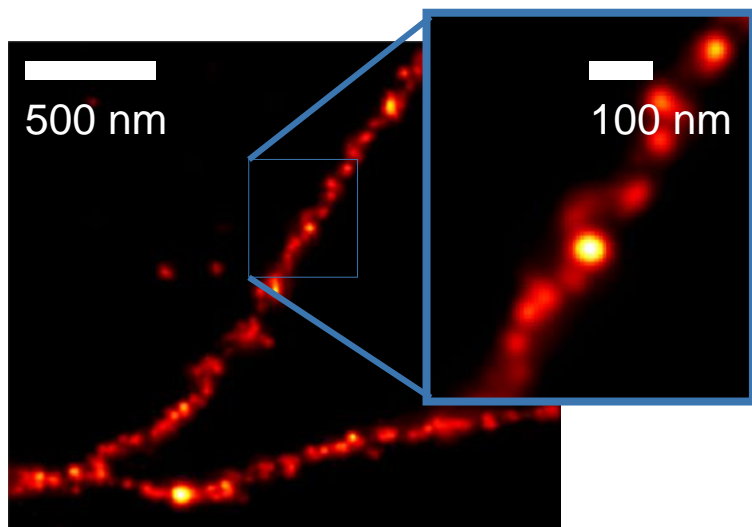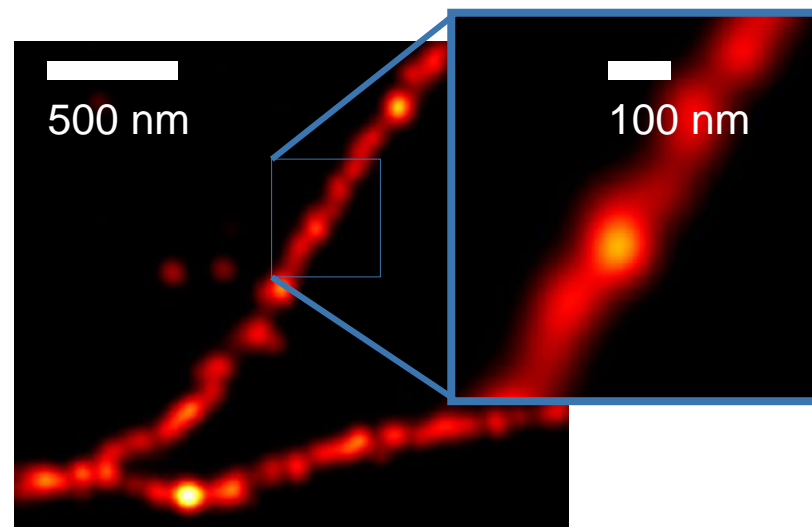
Render Localized Precision

Localized Precision uses the largest localization precision value set in PAL-Filter for all patterns. **The reason for this:** Despite the high precision (e.g. 5 nm) of each individual pattern, this is of little use if the density of patterns is low (i.e. if the object is sparsely labelled or if there are no other patterns nearby to fill the gaps). The resolution therefore must be decreased to appear less spotty.

500 nm   100 nm

☑ Render Localized Precision 10,0 nm

Less „blur" but more „spotty"
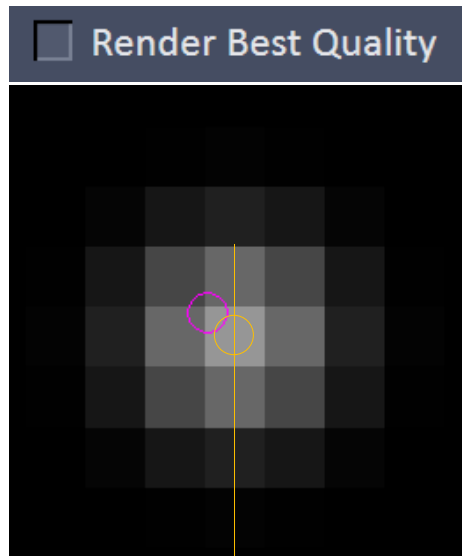
500 nm   100 nm

☑ Render Localized Precision 30,0 nm
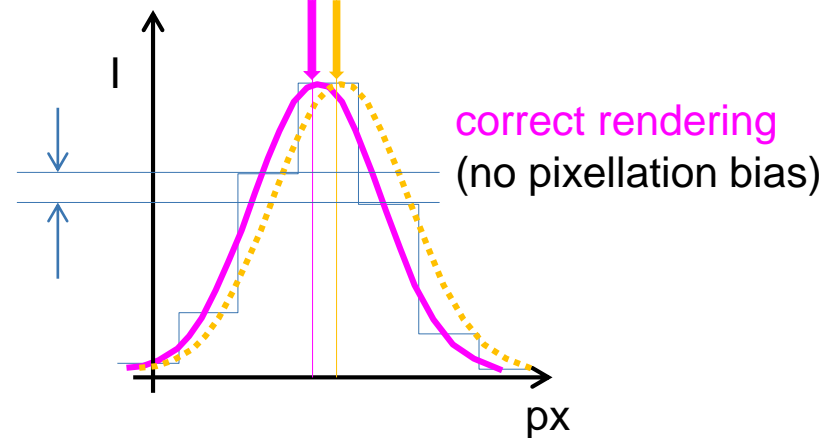
More „blur" but less „spotty"

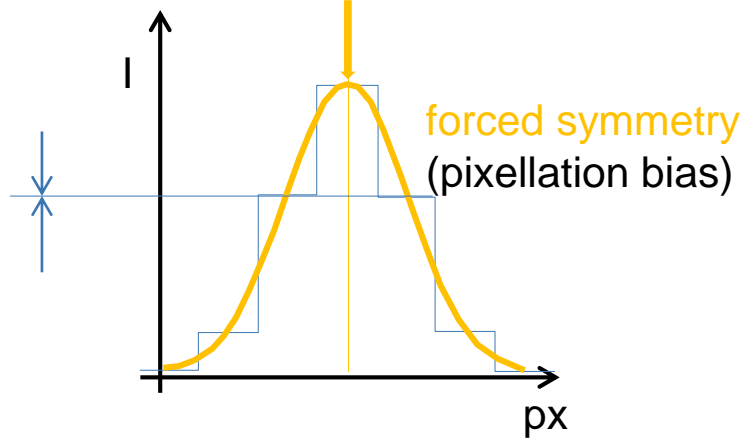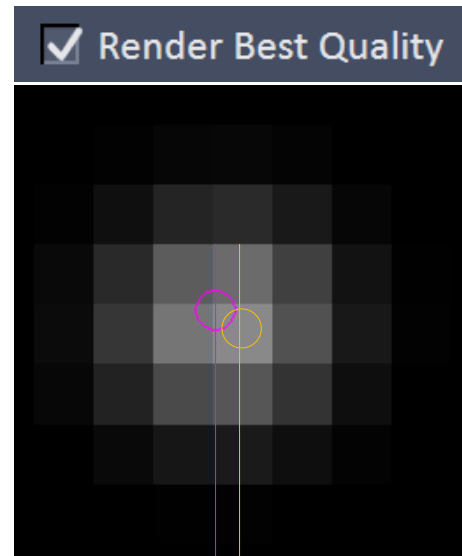# Render Best Quality
Only for pixel sizes above 10 nm

7

☐ Render Best Quality

☑ Render Best Quality

pixelsize < 10 nm
always symmetric

forced symmetry
(pixellation bias)

correct rendering
(no pixellation bias)

I

I

px

px

Overview

Remark 1: Molecule Density
Warning: Readout may be confusing

Note 1:

Density is given in „molecules" per square micron (µm²)

(1) „molecules" is the count of fitted events, under the assumption that an event corresponds to a molecule.
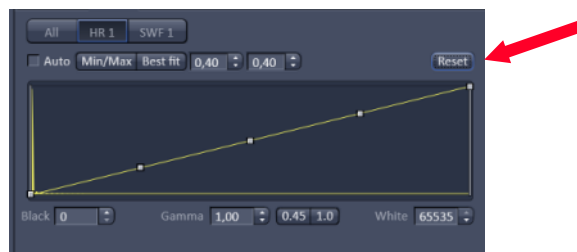
(2) To obtain the number of events per pixel bin, the LUT value has to be multiplied by the square of the pixel size times the binning.

For a white pixel in the depicted case:
212500 multiplied by (0.005µm * 4)² - or 0.0004, gives:
85 events per pixel bin

Note 2:

The LUT is affected by brightness and contrast as well as by rendering of the dynamic range. For best results, keep dynamic range at 100% (see red arrows)
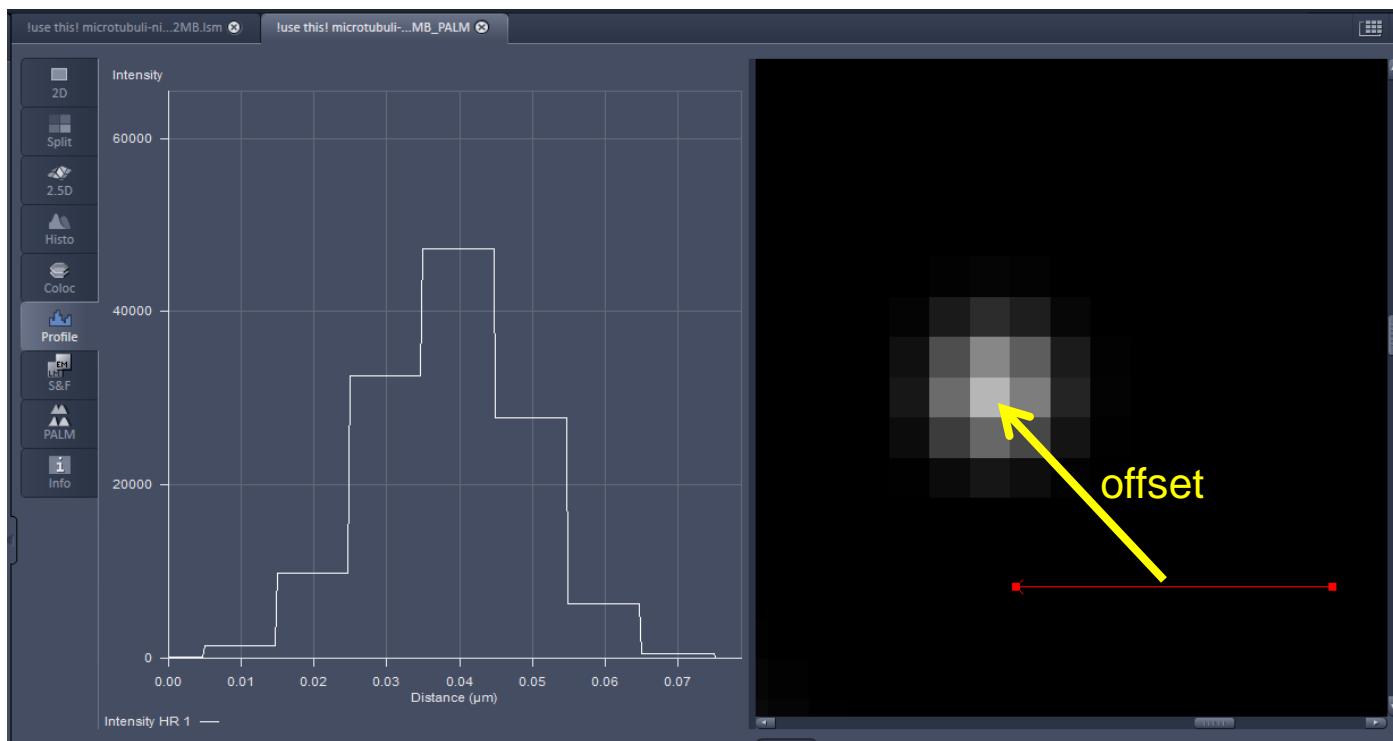
Remark 2: Profiles
Overlay and images are offset (known issue as of 9 / 2015).

Profile overlay and image are offset (typically the image is more to the top left)

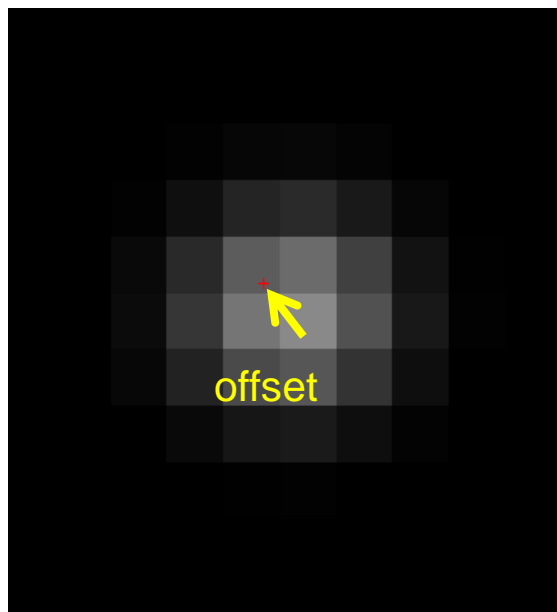The red profile line is displayed at the "wrong position"

Remark 3: Gauss + Cross / Gauss + Centroid
Overlay is mismatched (known issue as of 9 / 2015).

Gauss + cross (or Gauss + Centroids) are offset
(typically the dots and crosses are more to the top left)

The cross is displayed at the "wrong position"

We make it visible.