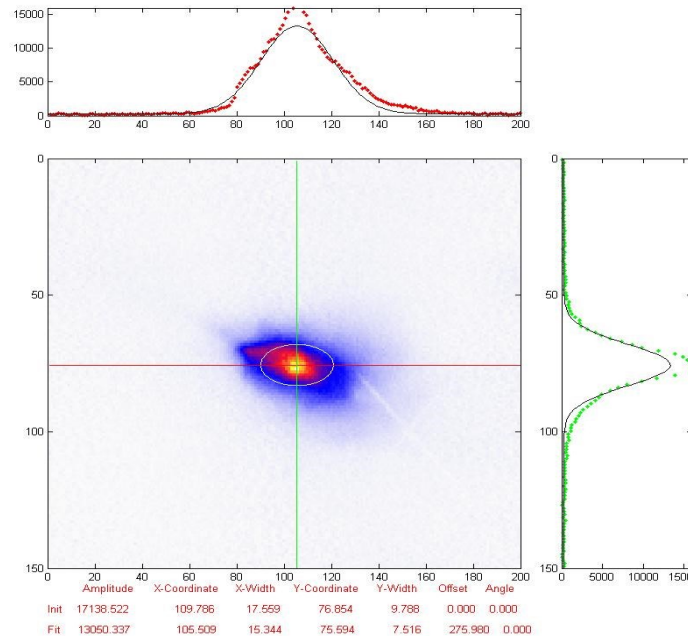


Spotsize

A coherent definition for the width and position of 2 dimensional objects and how to calculate them
introducing threshold masked centroids

- > Koschitzki, Christian
- > Berlin Desy-Zeuthen, 29.8.2017



On a good day the world is gaussian

- and the signal is bright
- in the center of the image
- about half as big as the image
- which is less than 300 x 300 pixel
- and the signal noise ratio is high

One may use a 2D gaussian fit routine like: `ck_fit2Dgaussian` for Matlab and the gaussian fit parameter sigma defines the width

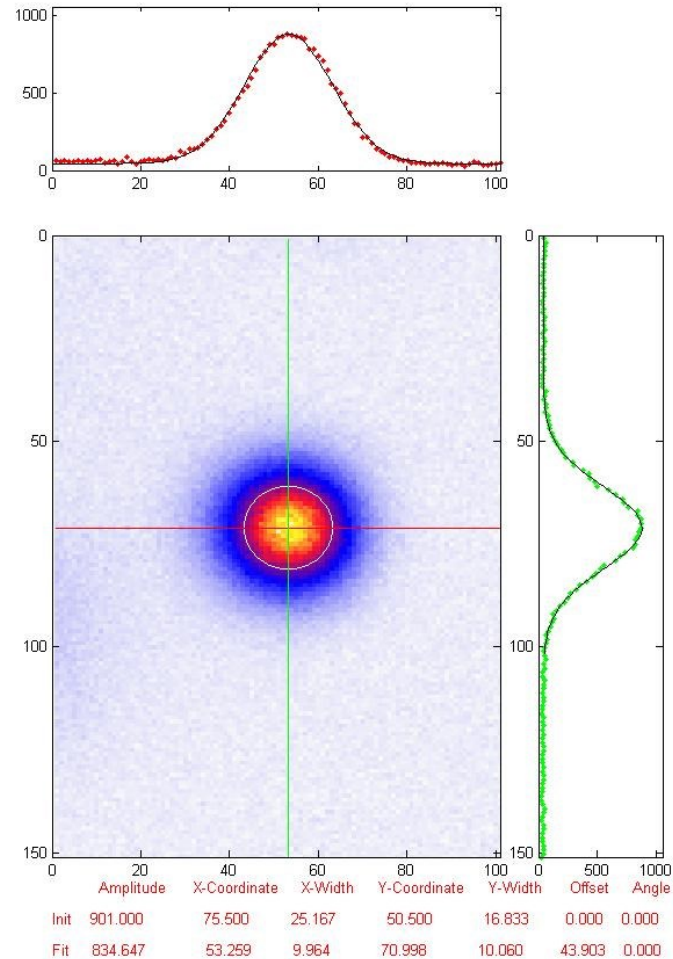
$$f(x, y) = BG + Ae^{-\left(\frac{(x-x_0)^2}{2\sigma_x^2} + \frac{(y-y_0)^2}{2\sigma_y^2}\right)}$$

$$C_{x,y} = x_0, y_0$$

Centroid for BG=0:

$$C_{RMSx,y} = \sigma_{x,y}$$

$$FWHM_{Gauss} = 2\sqrt{2\ln 2}\sigma$$



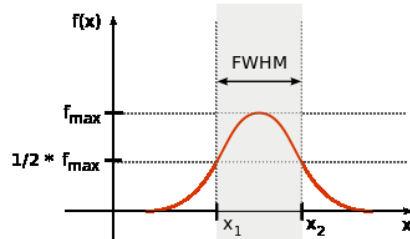
On a bad day it's just a blob

For practical reasons Gauss fit's have been used to determine spot width and position for non gaussian distribution. This is bad practice.

Fitters are for fitting

If no analytical function describes the Distribution, use width definitions that are defined for arbitrary distributions:

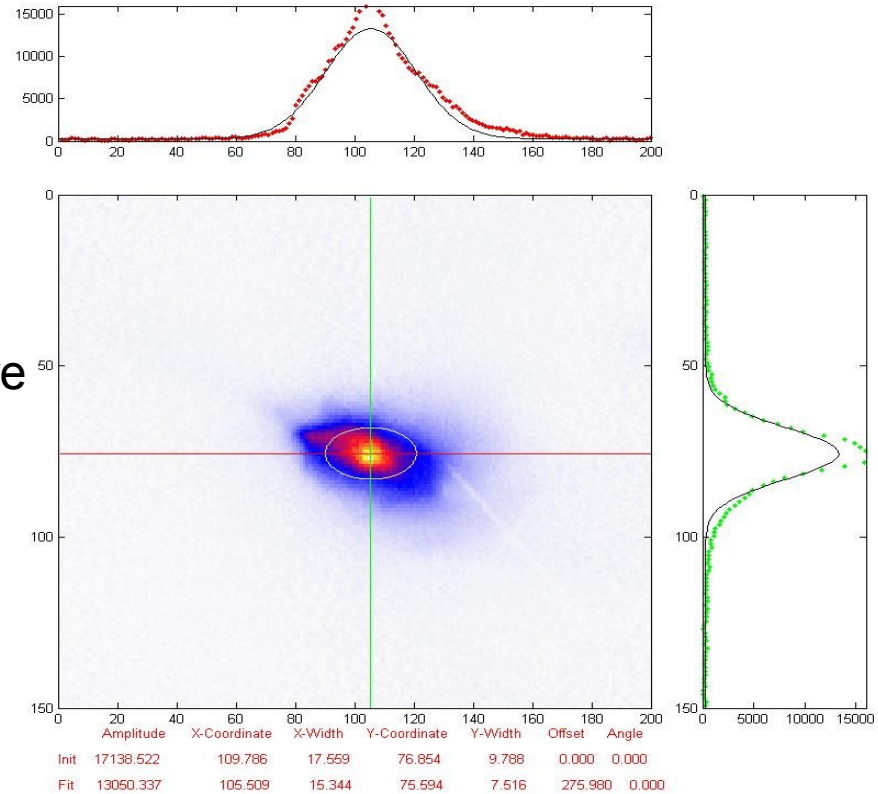
FWHM:



Centroid:

$$C_x = \frac{\sum C_{ix} A_i}{\sum A_i}$$

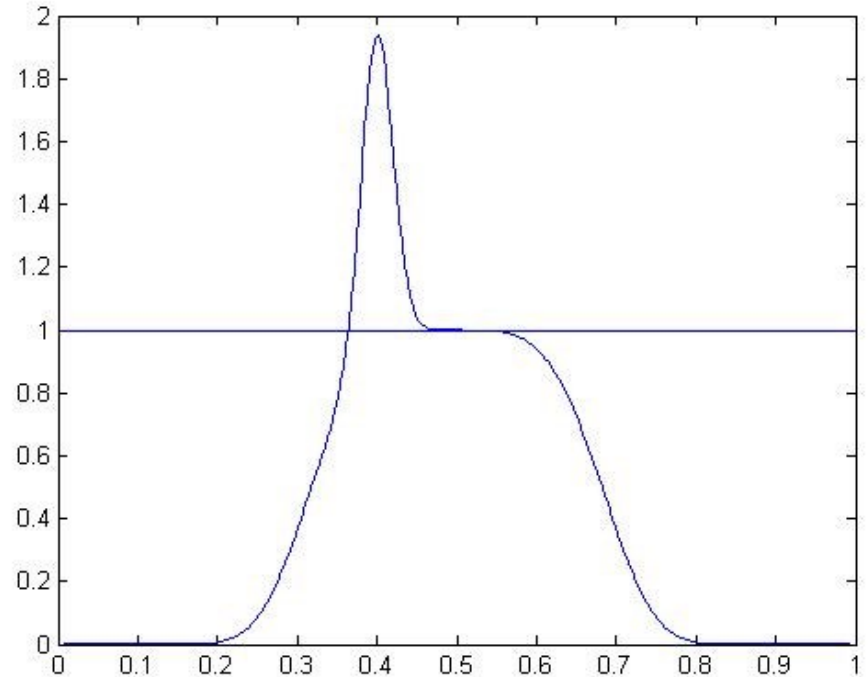
$$C_{RMSx} = \sqrt{\frac{\sum (C_x - C_{ix})^2 A_i}{\sum A_i}}$$



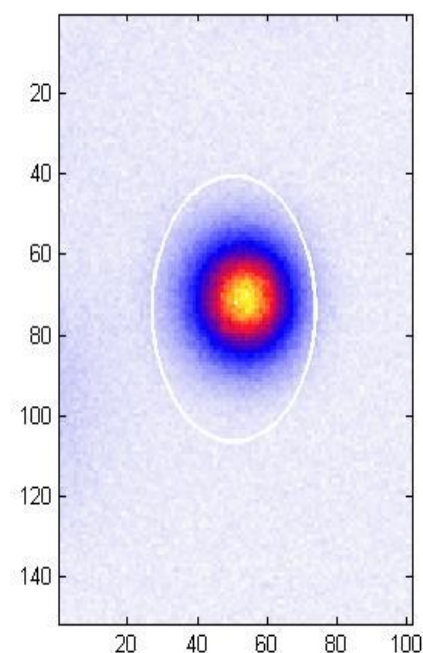
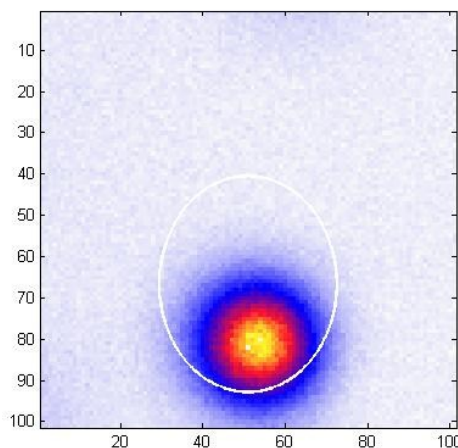
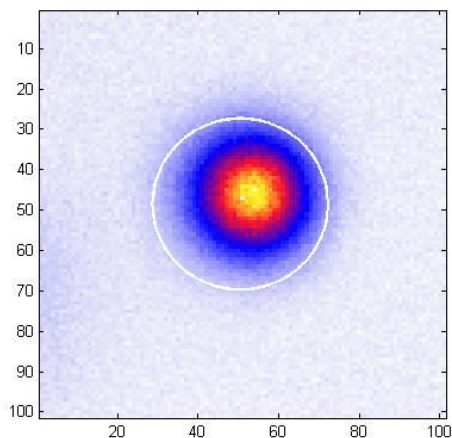
The problem with FWHMs

FWHM depends strongly on the maximum. Singular high but narrow peaks (broken pixels, hot spots) can fool this method. Particularly if the derivation around HM gets small.

FWHM: Unnecessarily Sensitive to signal properties.



Centroids with noise



Centroid is very sensitive to background and ROI (Region of Interest).

Center gets pulled toward image center.

RMS gets pulled toward edges.

Only works properly with zero background.

FWHM has problems with signal.

Centroid with background. Rather deal with background. Here is how ...

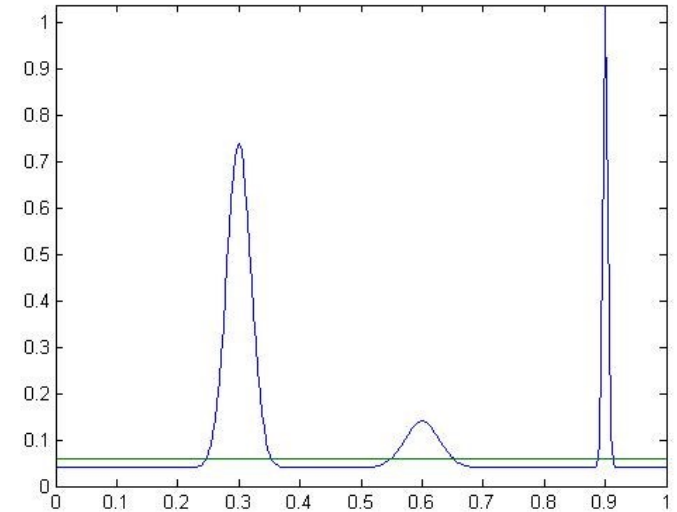
$$C_x = \frac{\sum C_{ix} A_i}{\sum A_i}$$

$$C_{RMSx} = \sqrt{\frac{\sum (C_x - C_{ix})^2 A_i}{\sum A_i}}$$

Solution with region thresholds

```
centstats = ck_centroid2d(image,threshold)
```

1. Find maximum of image
2. Find the region of connected pixels around maximum that are above threshold. Crop.
- 3.
4. Calculate Centroid



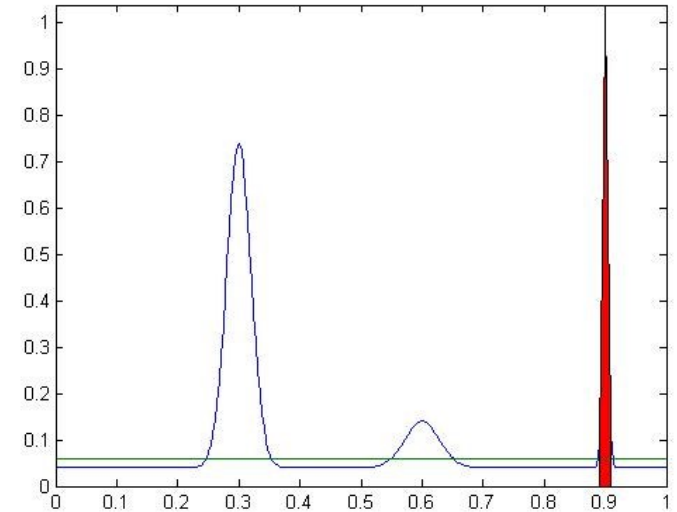
Solution with region thresholds

```
centstats = ck_centroid2d(image,threshold)
```

1. Find maximum of image
2. Find the region of connected pixels around maximum that are above threshold. Crop.
3. (optional) Check if region has enough pixels in them.
If not: set region to zero and start over
4. Calculate Centroid

Step 3 avoids broken pixel and gamma traces.

```
ck_centroid2d(image,50,{'minsize',100});
```



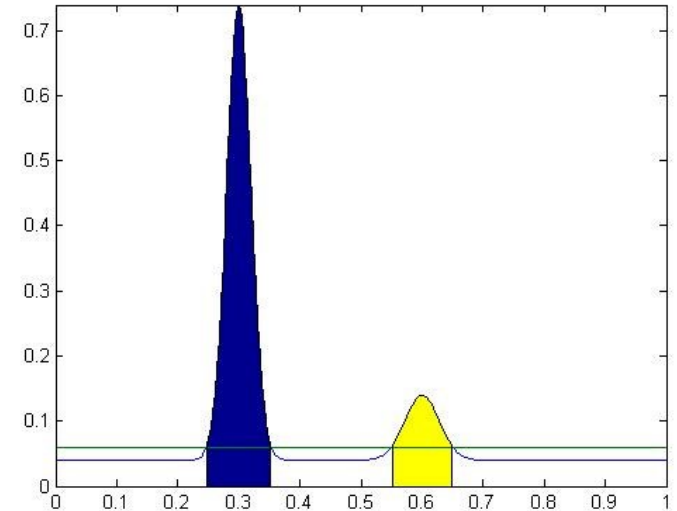
Solution with region thresholds

```
centstats = ck_centroid2d(image,threshold)
```

1. Find maximum of image
2. Find the region of connected pixels around maximum that are above threshold. Crop.
3. (optional) Check if region has enough pixels in them.
If not: set region to zero and start over
4. Calculate Centroid

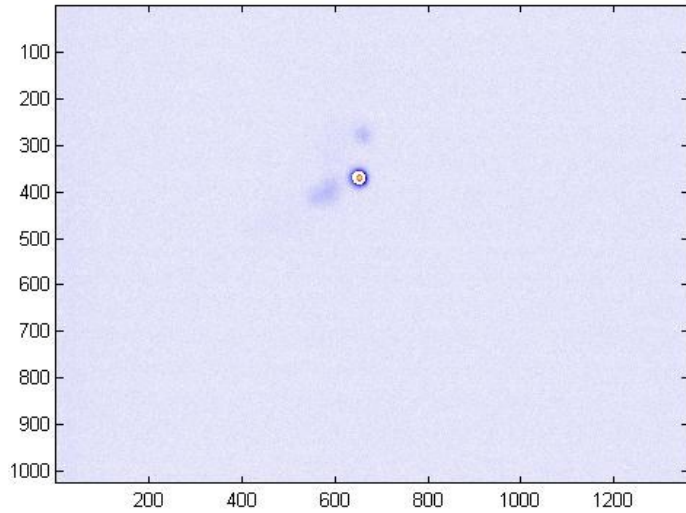
Step 3 avoids broken pixel and gamma traces.

```
ck_centroid2d(image,50,{'minsize',100});
```

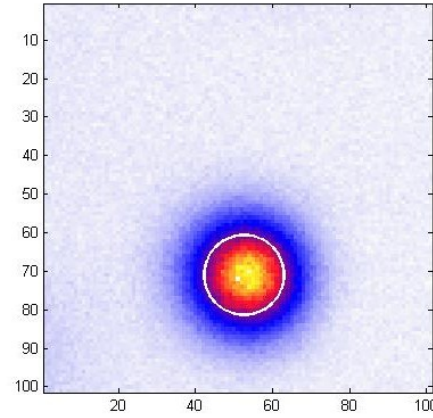


Threshold masked centroid

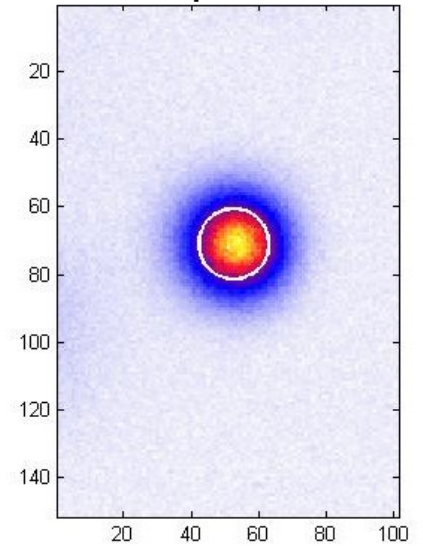
Full Pic



Offset

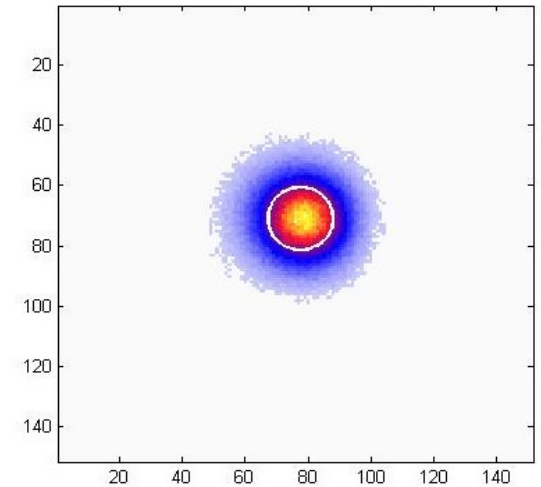


Aspect



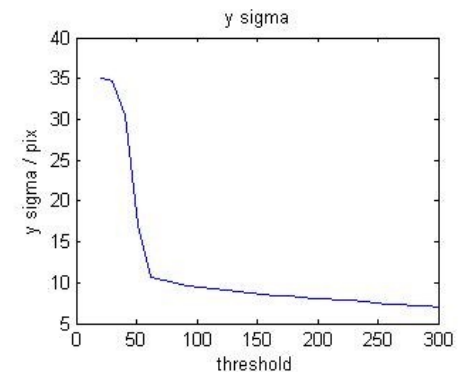
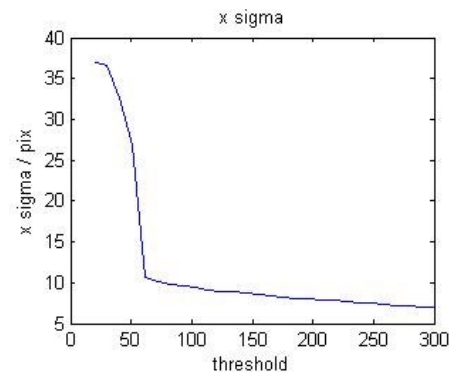
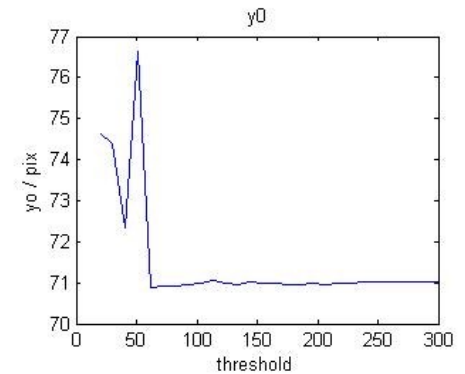
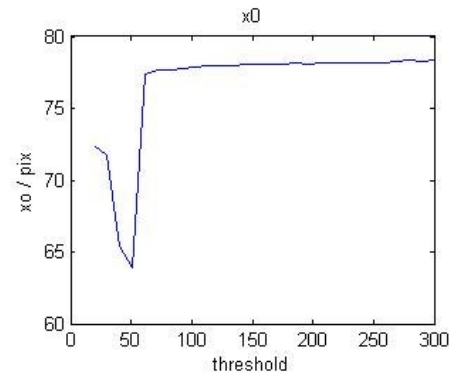
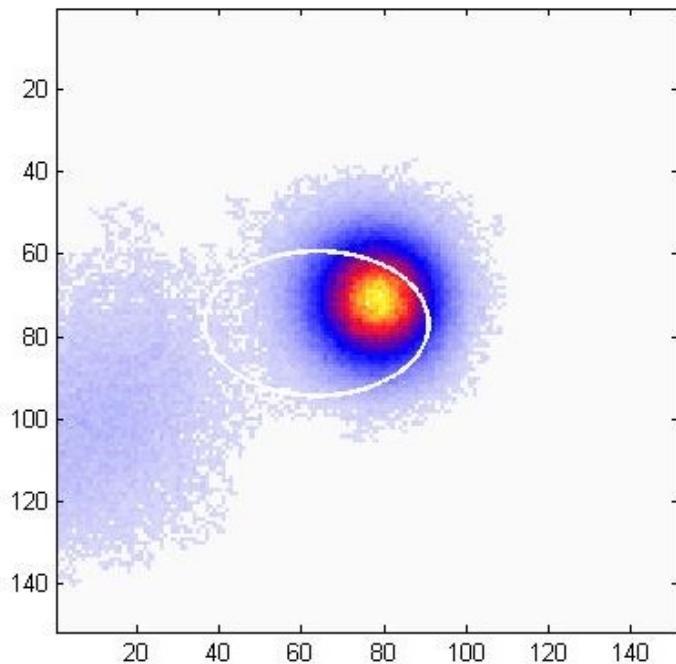
Results using threshold region (threshold = 70)

	max	x0	xsigma	y0	ysigma
Full Pic	901.0000	651.6243	10.2776	369.9308	10.2842
Offset	901.0000	52.6243	10.2776	70.9308	10.2842
Aspect	901.0000	52.6243	10.2776	70.9308	10.2842
Gauss2D	834.6468	53.2590	9.9644	70.9979	10.0598



Threshold dependent performance

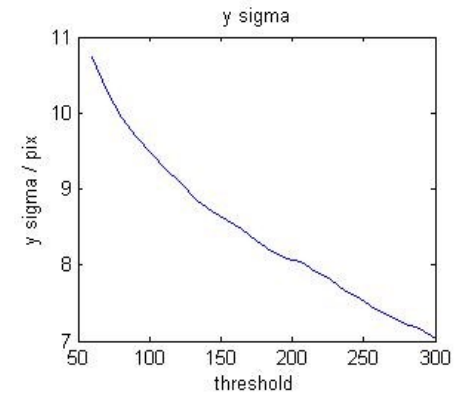
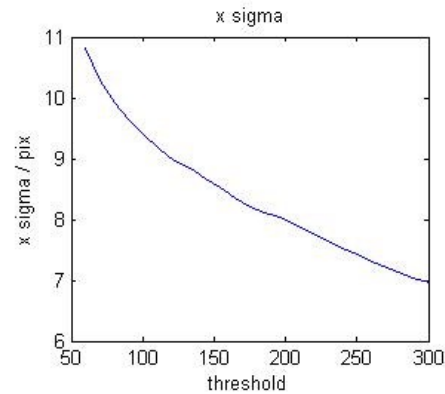
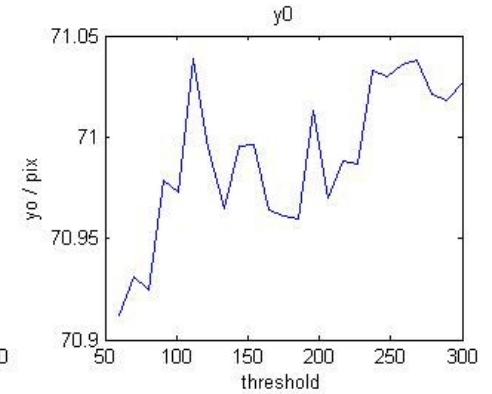
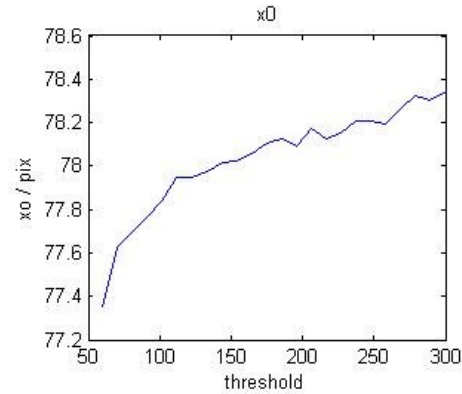
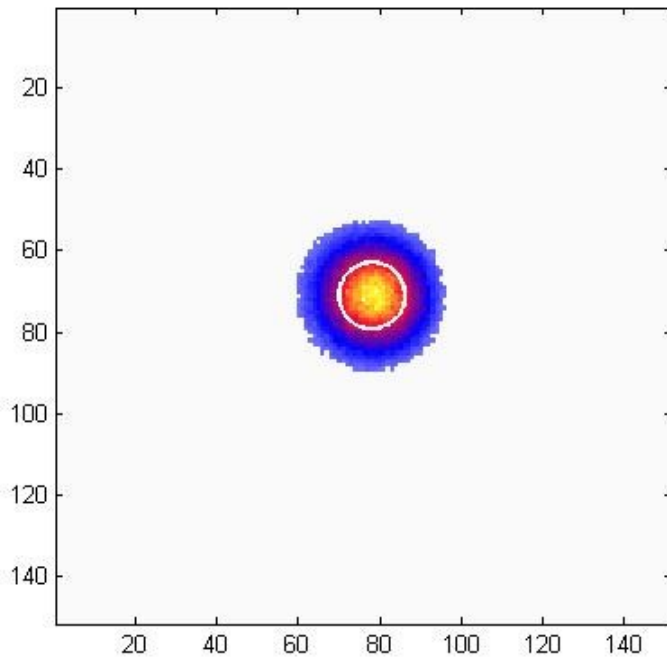
Thresh 50



Performance depends slightly on threshold. It can however be considered consistent for a set of data analyzed with the same threshold.

Threshold dependent performance

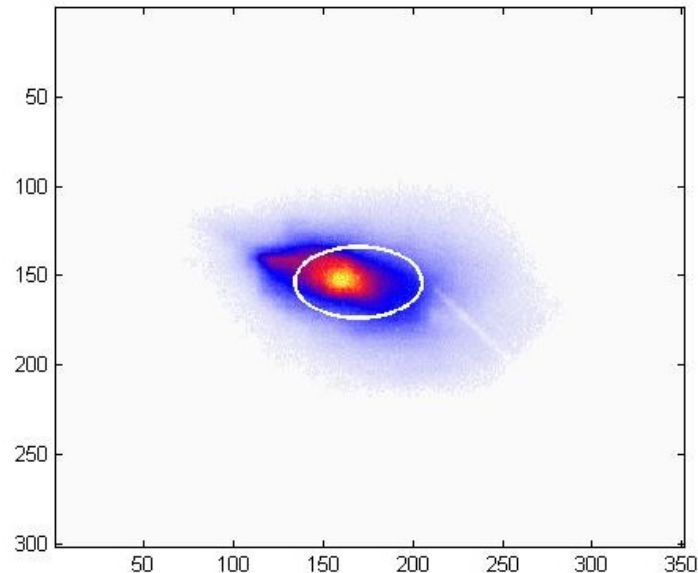
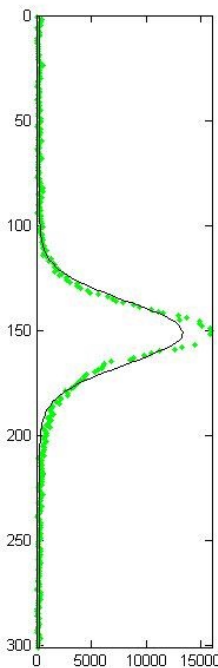
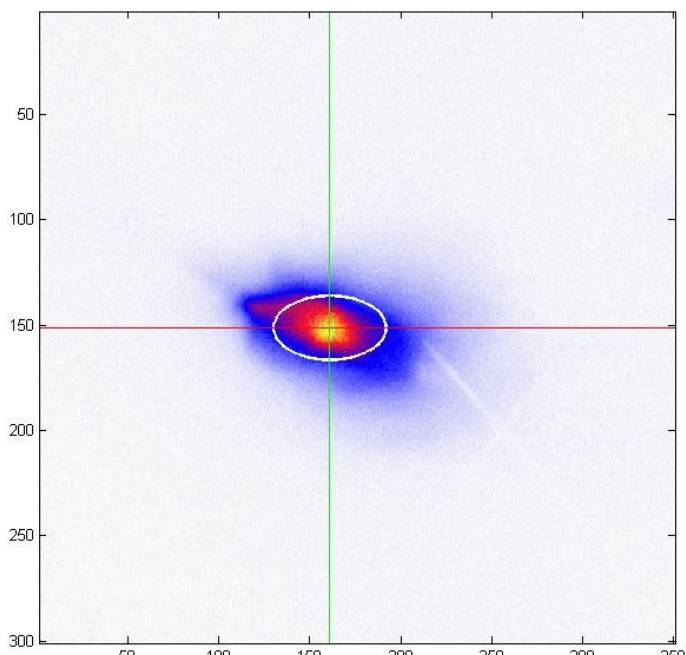
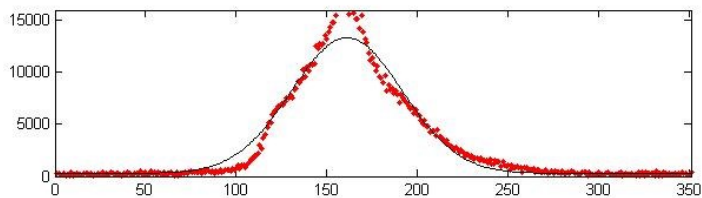
Thresh 200



Performance depends slightly on threshold. It can however be considered consistent for a set of data analyzed with the same threshold.



Comparison to Gaussfit



	xsig	ysig
Gauss :	30.7567	15.0926
Centroid:	35.3461	19.6700

Use `ck_centroid2d` as initial estimate

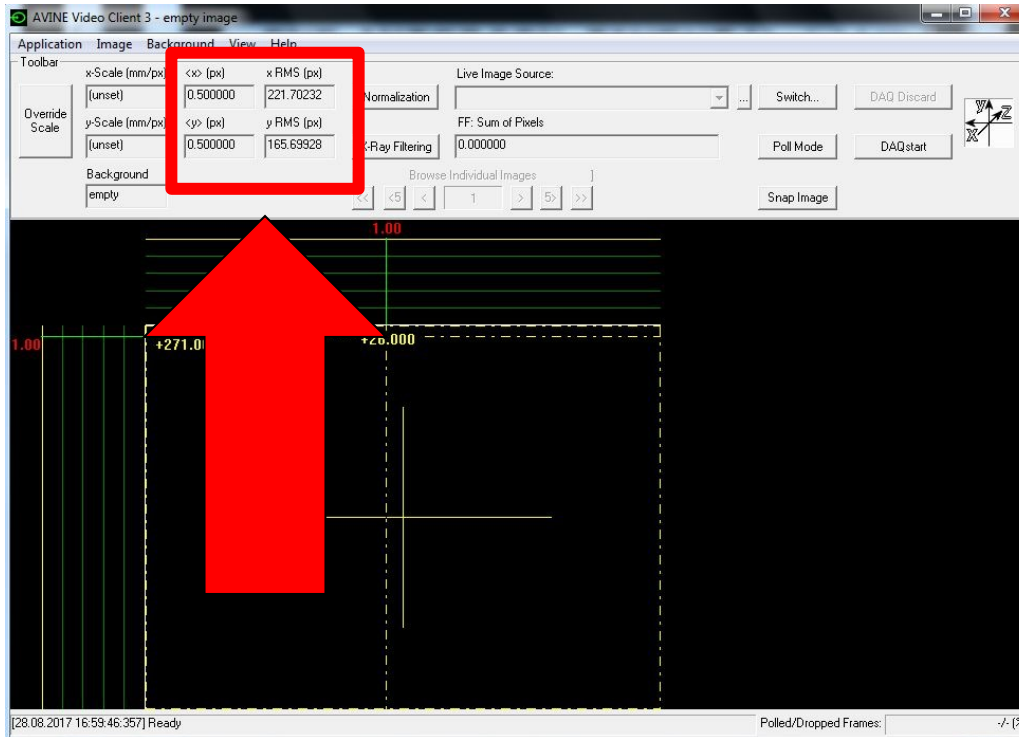


```
fitvals=ck_fit2dgaussian(image,{'showplot',1,'centest',threshold});
```

	Amplitude	X-Coordinate	X-Width	Y-Coordinate	Y-Width	Offset	Angle
Init	17268.000	169.405	34.421	153.655	18.959	0.000	0.000
Fit	13059.292	161.076	30.757	151.184	15.093	290.163	0.000



How to use it



More functions:
`ck_set_cmap();`
`ck_keyword_set();`
`ck_where();`
...
More coming

Find my functions here:

`addpath(' \\afs\ifh.de\group\pitz\doocs\measure\scripts\SVN\MatlabFunctions\CK_Tools ');`



Summary

- > Don't fit unless you know the distribution function
- > Centroid is defined for any distribution
- > Using regional mask, centroid is a robust and fast method

Find my functions here:

`addpath(' \afs\ifh.de\group\pitz\doocs\measure\scripts\SVN\MatlabFunctions\CK_Tools ');`

