What is GUVdetector?

GUVdetector is a MATLAB program for the analysis of Giant Unilamellar Vesicles (GUVs). It determines permeabilities, radii and radial intensity profiles of the vesicles by applying image processing techniques. As input, the tool requires images or movies consisting of up to 4 channels. The main feature of this program is the fully automatic GUV detection. Since vesicles are spherical objects, GUVdetector uses *Circular Hough Transformation* for the detection of the GUVs.

Once detected, the mean pixel intensity within the GUVs is compared with the background intensity. The degree of filling is the ratio of these values and is a measure for the permeabilized state of the vesicle.

The radial profile is the mean intensity over the perimeter of a circle as a function of the circle radius. GUVdetector determines the radial profiles for all detected GUVs in the radius range from 0 up to the double vesicle radius. The maximum value corresponds to the averaged membrane intensity. This analysis is convenient to determine how much a protein binds to the membrane. Radial profiles are not visualized in the user interface but saved together with the other output data.

What has to be considered during GUV preparation and imaging?
For GUV preparation have a look on the protocols by *M. I. Angelova et al., Faraday Discussions, 1986*, for instance. Use at least two fluorescent dyes: One binding to the GUV membrane and another one that is free in solution, surrounding the GUVs. Next step is to image the GUVs with a microscope. If a GUV permeabilizes, the dye in the solution partially enters the GUV resulting in a lower contrast between the region within and outside of the GUV. You can clearly see it on the output image and this is exactly what GUVdetector measures quantitatively.

GUVdetector is originally created for *lsm* image/movie format, the standard image format generated by Zeiss microscopes. It is also compatible with other formats.

How to install GUVdetector?

There are two ways to use GUVdetector:

1) When you have MATLAB on your computer it is best to use it directly as a MATLAB script. Open *GUVdetector.m* in the MATLAB editor and run it. The user interface appears and you can use it now for your purposes.
    Note: You need at least MATLAB version R2012b including Image Processing Toolbox and Signal Processing Toolbox.

2) You can use GUVdetector as a standalone program, without having MATLAB installed. If you do so, install first the *MCRInstaller*. Then, double-click *GUVdetector.exe* to start the program.

GUVdetector overview

Load and prepare files
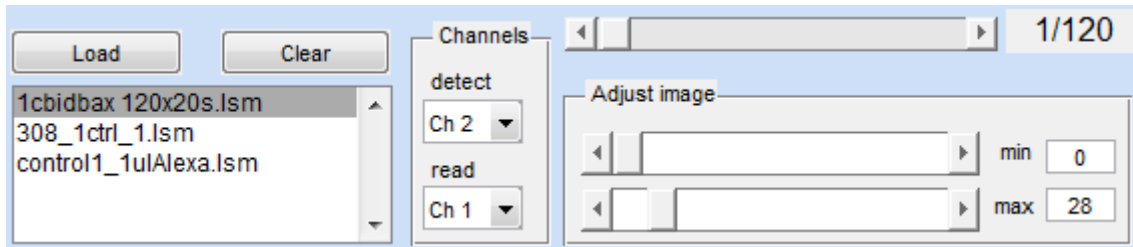
Load

Opens a window containing all files in the current work folder. Change the folder if necessary. Double-click on the file you would like to analyze and the window will close again. It is possible to multiselect files. Wait for background and offset determination; it can take some seconds (images) or minutes (movies). Now you find the new file(s) in the list below. The image and the corresponding parameters are displayed. You can load additional files to the list at any time; display them by clicking on the corresponding file name. For each file, the corresponding parameters are saved automatically.

Clear

Removes the current file from the list below.

Channels --- detect

Based on this channel the program detects GUVs. Therefore, choose the channel where the GUV membrane is emphasized. The detecting channel is always displayed in the user interface.

Channels --- read

Chose a reading channel to show its corresponding background, offset and (if already analyzed) permeability list. This channel has no effect on GUV detection; it is not displayed in the user interface.

Frame slider

Only for movies; change the movie frame to see its corresponding background, offset and (if already analyzed) permeability list. The number of the current frame and the total number of frames are shown in the box to the right.

Adjust image

This is for contrast and brightness enhancement. Adjust the image to get clear and bright GUVs in the detecting channel for best detection performance! The pixel intensities between *min* and *max* are mapped to the maximum pixel range 0….255. In most cases, best performance is achieved when *min* remains 0 while you drastically decrease *max*. In any case, try out several *min* and *max* combinations to achieve best image enhancement. You can also put *min* and *max* by hand in the corresponding boxes next to the sliders. The allowed value range is between 0 and 255.

Hough Transformation

Hough Transformation

10	Radius_min
50	Radius_max
1.24	Sensitivity
0.2	Threshold
0.5	Radius_corr
15	Inside

Reset | For all

The **Circular Hough Transformation (CHT)** is a powerful tool to find circle-shaped objects in digital images. It is a transformation from the image space to an accumulator space which has the same dimension as the image itself but is initially empty. Prior to the CHT, edge detection is carried out on the image. For each edge point a series of virtual circles within a certain radius range are drawn in the accumulator space, being the edge point the center point of the virtual circles. Incrementing all coordinates that the perimeter of the circle passes yields a structure where maxima correspond to circles in the image space. Note that a circle is fully described by its center point and radius.
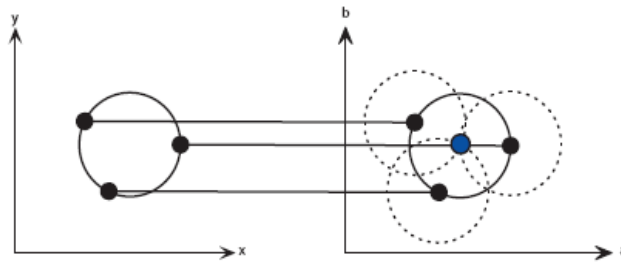
https://course.ku.ac.th/lms/files/resources_files/74825/127268/circle_hough.pdf

Radius Range *default: 10 - 50*
Minimum and maximum radius for Hough Transformation. Virtual circles with radii within this range are drawn in the accumulator space. Both values should approximately correspond to the radii of the smallest and biggest GUVs on your image, measured in pixels. A too wide range may immensely decrease detection speed and efficiency due to highly overloaded accumulator space.

Sensitivity *default: 1.24*
This value determines how sensitively the program searches for maxima in the Hough space, so it can be seen as the sensitivity for recognizing circles. A smaller value leads to higher sensitivity and thus, in most cases to more detected circles. An overestimated can "blind" the program, which means that the number of misdetections increases while real GUVs are overseen. On the other hand, a low sensitivity leads to poor GUV detection efficiency.

Threshold *default: 0.2*
Minimum fraction of in the highest accumulation peak in the Hough space, which is still interpreted as an individual peak. Regions below the threshold are not considered as maxima and thus not as circles/GUVs. The effect of *Threshold* is similar to *Sensitivity*. Smaller values lead to higher sensitivity.

Radius Correction *default: 0.8*
Factor to scale the radii of the masks, which are used to read out the inner vesicle intensity. To ensure that the mean pixel intensity is only read out in the lumen of the GUVs, usually a smaller radius needs to be taken then the real radius given by the CHT. If you change this value after analysis, the program recalculates the permeabilization values and displays the new result immediately.

Inside *default: 50*

Maximum acceptable inner vesicle intensity in the membrane channel. This value is for distinguishing GUVs from multilamellar vesicles. They are characterized by continuously high fluorescence intensity over the whole area (not only the rim). Only circles with inner mean intensity below *Inside* are considered as GUVs. The range for *Inside* is between 0 and 255. Depending on the pixel intensity distribution of your image, you have to set an appropriate value.
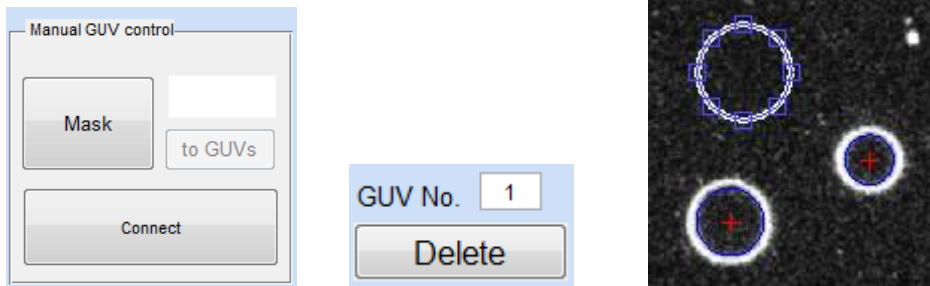
Reset

Resets all parameters for Hough Transformation and post-selection to default values and enables blocked pushbuttons in case of a program error.

For all

Adopts the current parameters and also the contrast settings to all images loaded, except of Radius_corr, which is not adopted for already analyzed images.

Manual GUV control

With this option you can check the mean intensity of any area on your image and also add GUVs which have not been detected automatically.

Mask

Click *Mask* and move the cursor over the image. You see that the cursor now appears as a cross. By left-clicking you can drag an ellipse over any area. Stop holding the left-click and you see the corresponding mean pixel intensity in the white box.
If you want to remove the ellipse you can for instance click on one of the *Adjust image* sliders, even without moving them and the ellipse will disappear.

to GUVs

If you want the area, which was previously selected with *Mask* to be an additional GUV, click on *to GUVs* and the corresponding permeability value is added to the GUV list on the right side of the user interface. The corresponding area is analyzed as the rest of the detected GUVs. The permeability value is displayed in the GUV list.
Attention: If you first add GUVs by hand and then run the automatic detection, the manually added GUVs will be overwritten or removed. In any case you should always run the automatic detection first and then add missing GUVs by hand.

Connect

This button is only enabled when a movie is currently analyzed. It tracks single GUVs along the movie frames and connects detections which belong to the same GUV. Before connecting, display the frame from where you would like to start the tracking. A window will appear upon

clicking, asking you to confirm that the right starting frame is chosen. After the connecting process, only those GUVs will remain in the analysis that could be tracked until the last frame.
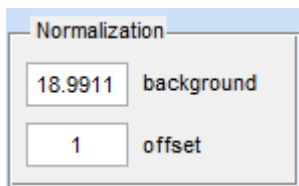
GUV No.
This is the number of the GUV that is highlighted in the permeability list. On the other hand, if you enter the number of a GUV, this GUV gets highlighted. This is convenient when you search for a GUV by its number, as it is often the case when you work with already analyzed data.

Delete
Removes all GUVs marked with a red cross. You can mark a GUV either by clicking on its permeability value or directly on the GUV itself. In this case you can even multiselect them, that means you can mark as many GUVs as you like while a click on the permeability value only marks the corresponding GUV and unmarks all the others. When you click on a GUV, its permeability value is automatically highlighted. Clicking on a GUV that already has a red cross unmarks it again.

Normalization

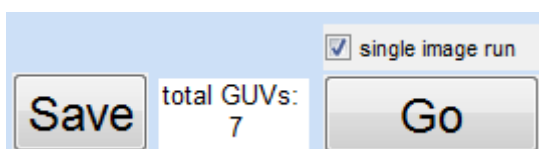$$Perm = 100 * \frac{Intensity - Offset}{Background - Offset}$$

Background
This is the mean pixel intensity of free dye outside the GUVs. The mean intensity within the GUV respect to the background (and considering the offset) is the degree of permeabilization of the GUV. The background is determined automatically for each channel and, in case of movies, for each frame when you load the file. You can also enter a value by hand.

Offset
This is the pixel intensity of the noise in your image. Ideally, it is simply 0 but there is mostly a positive value that has to be subtracted from all pixel intensities to get correctly normalized permeabilization results. This is related to the sample nature and the imaging conditions. The offset is determined automatically for each channel and, in case of movies, for each frame when you load the file. You can also enter a value by hand.

Run analysis and save results

Go

Runs the Hough Transformation on all image files with corresponding settings. Depending on the complexity of your images and your computational power, the analysis may take some minutes. Be sure you have set the detecting channel and the image contrast appropriately to achieve best performance. As a result, you get a permeability list for each channel and each frame. Use *Channel --- read* and the frame slider to switch between the lists.

Single image run

If selected, *Go* only analyzes the current image, otherwise the whole image set.

Total GUVs

Shows the total amount of detected GUVs on the current image.

Save

Saves the permeabilization degrees (in %), the radii (in μm) and the radial membrane intensities (intensity values 0…255) for all GUVs in all files, channels and frames. The data format is ASCII and the location is the same folder from where you selected the image/movie files. The values are saved in matrices having the form:

<u>Permeability:</u>
<u>Rows</u>: GUVs
<u>Columns</u>: Channel_1/Frame_1; Channel_2/Frame_1; …; Channel_1/Frame_2; …

<u>Radial Intensity:</u>
<u>Rows</u>: GUVs
<u>Columns</u>: Channel_1/Frame_1; Channel_2/Frame_1; …; Channel_1/Frame_2; …
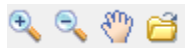
<u>Radii:</u>
<u>Rows</u>: GUVs
<u>Columns</u>: Frames

In addition, an m-file containing the whole analysis result in a Matlab-intern format is created at the same location. You can reload your analysis by loading this file.

Toolbar

Use the zoom-function especially when you add GUVs by hand. It is advisable that the GUV you wish to add fill out the whole image axis before you put the mask over it. The hand is for moving the displayed region of the zoomed image.

With the yellow folder you can reload an analysis you have saved before. Click on the folder to open a window containing all m-files in the current work folder. You can change the folder if you need. Double-click on the file you would like to reload and the window closes again. The file is now added to the file-list. The image and the corresponding parameters are displayed just like you have saved it. From now on, you can handle the file as any other. Saving it will overwrite the corresponding m-file.

How to use GUVdetector? A short example

1)   Click *Load* and select an image or a movie you would like to analyze. You can add as various files.

2)   Go to *Channels --- detect* and look at the channels your image contains. Select the channel where you would like to localize the GUVs, not the one where you would like to check the permeabilization! Hint: Use the channel where the membrane is emphasized.

3)   Move the sliders in *Adjust image* to increase the contrast. The rims in the detecting channel must be as bright as possible! In most cases it is best to decrease *max* and remain *min* at 0.

4)   *Hough Transformation:* Are all parameters appropriately set? If you are not sure about that, let them unchanged and if the performance is not satisfactory vary them in the next run.

5)   Have *Background* and *Offset* plausible values? If not, correct them manually. You can also use the *Mask* function to add a "GUV" that consists of pure background/offset and compute the permeabilities later by hand on base of the raw intensities and using that "GUV" as background/offset.

6)   Click *Go* to start the automatic GUV analysis. It may take some minutes. The files you have selected are now analyzed one by one. At the end you see which GUVs were detected (marked with a blue circle) and their corresponding permeability values. Hint: If you select *single image run* only the current file is analyzed. This may be convenient to find the best parameters for the actual analysis.

7)   Check the permeabilities in different channels by changing *Channel --- read*. If you have analyzed a movie check also the permeabilities in different frames by moving the frame slider.

8)   Remove misdetections: Click directly on the GUVs you would like to remove. By clicking on them, they get a red cross and their permeability is highlighted. You can also multiselect them. *Remove* removes all GUVs with a red cross inside. Hint: By clicking on a permeability value, a red cross marks only the corresponding GUV.

9)   Are there some GUVs which have not been detected automatically? Add them manually by using the *Mask* function. Drag a circle over the missing GUV and click *to GUVs*. Its permeability is now added to the list.

10)   *Save* your results. You find them in the same folder where your files are from. Radii, permeability and absolute intensities for each image/movie are saved in separate ASCII files

11)   You can reload files that have been saved before. Click on the yellow folder on the tool bar above the user interface and select a mat-file that contains all the saved data of your file. The mat-file has the same name as the file itself. By doing so, the image as well as its parameters and analysis data are displayed in the user interface.

Advice & tricks

Installation
If you want to use GUVdetector as a MATLAB script, be sure you have at least MATLAB version R2012b and both Image and Signal Processing Toolbox. Otherwise, use the standalone version.

Files
GUVdetector is originally created for the analysis of *lsm* images and movies. *lsm* is an extension of the TIFF multiple image stack file format (http://en.wikipedia.org/wiki/LSM_%28Zeiss%29). You might get into troubles trying to work with other formats when recording movies. The way to call a certain lsm image channel in MATLAB is:

$$I = imread('myimage.lsm');$$
$$Channel\_N = I(:,:,N);$$

The way to call a certain lsm movie frame is:

$$M = imread('mymovie.lsm', frame);$$

If frames and channels are called in the same way for your format, GUVdetector should be able to analyze them correctly. Anyway, if you call lsm movie frames in this way, only every second frame is a real frame. Therefore, GUVdetector is restricted to frame = 1, 3, 5, 7, 9, … Please contact me if you encounter this situation: eduard.hermann@is.mpg.de.

Parameters
Most parameters in the user interface can be changed by hand so be sure not to put values which don't make sense. For instance, none of the parameters should be negative. Furthermore, it doesn't make any sense to set *Radius_min* higher than *Radius_max* or *offset* higher than *background*. Take your time to play with different parameter combinations to optimize the detection result. The by far most important parameter is the contrast!

Channels
You should really ensure that your analysis runs on the correct detecting channel. This must be the channel where the GUV rims are clearly seen. In the other channels non-permeabilized GUV are detected with higher preference than permeabilized ones leading to biased detection.

Go
If one file cannot be analyzed, GUVdetector automatically jumps to the next file. So if you don't find any results after analysis there is probably a problem with your file or with the settings.

Multiselection
When you load or reload files or remove false positives, use the multiselect option! It safes a lot of time!

Adding GUVs by hand
When you add GUVs by hand, first zoom as far as possible into the region of interest and then put the mask on it. Putting the mask on small regions is exhausting if you want to place it with high precision.

Radius correction
For the radius correction, take a value around 0.5 to ensure, that no background contributes to the GUV intensities. For very small GUVs even smaller values are appropriate. You can change the radius correction whenever you want, even after running the analysis.

Don't separate files!
When you reload old results, make sure that the image file is in the same directory as the corresponding m-file.

About the code

The program itself consists of 5 principal (and several small) m-files:

GUVdetector.m
This is the main file which contains the graphical user interface. All relevant parameters are stored in a main variable called *ship*. It is a cell array and each cell element corresponds to an image/movie.

showimage.m
Updates the image as well as the parameters.

set2zero.m
Removes the image and resets all parameters to default values.

test.m
Executes when you click on *Go*. It does some pre-processing like contrast enhancement and box filtering. It calls the Hough methods and passes the results to *readout.m*.

readout.m
Determines the GUV permeabilities, removes multilamellar vesicles and misdetections (if you use post-selection). The output contains radii, permeabilities and membrane intensities of the GUVs.

CircularHough_Grd.m
Hough routine written by Tao Peng, University of Maryland, USA. It is available on the User File Exchange platform on www.mathworks.com. The output contains radii and center points of all detected circles.

What else?

GUVdetector was developed in 2011/2012 as a diploma project by Eduard Hermann, physics student at the Max-Planck-Institute for Intelligent Systems in Stuttgart, Germany, and his colleagues. Special thank is dedicated to Yamunadevi Subburaj, Stephanie Bleicken and Ana García Sáez for their support, the discussions and many fruitful ideas. We are willing to continuously improve the program and we are happy about your feedback! Please send us all your questions, critics, suggestions, bug reports or whatever you would like to let us know about the program to eduard.hermann@is.mpg.de.

The quality of the automatic detection depends mainly on the quality of the image. Noisy images and/or distorted GUVs affect the detection efficiency. On the other hand, a high amount of multilamellar vesicles will result in many false positives. Multilamellar vesicles have the same circular shape as GUVs but they are not usable for permeabilization analysis and have to be deleted by hand. Up to now, we haven't implemented an efficient selection algorithm to discriminate for multilamellar vesicles. However, it is easy and quick to remove false positives and you should make use of this option.

Remember that the single-vesicle approach, for which this program has been developed, depends on a high amount of data points to be statistically significant. As a rule of thumb and according to the Poisson distribution, the statistical quality of the analysis result can be quantified by the square root of the number of measured GUVs. In general, an amount of 100 GUVs already carries a representable image of what is happening in your experiment. If you double the number of GUVs, the quality of the result only increases by around 40%. In other words, when you already analyzed 100 GUVs, it is not very probable that additional 100 GUVs change your result. We recommend the user to generate many images of the same experiment in order to have a sufficient amount of GUVs for the analysis. Sometimes you might have images containing several hundred of GUVs and you realize that the detection efficiency is not outstanding; for instance, of 500 GUVs only 150 are detected. This is no reason to panic! The 150 GUVs will provide you a very accurate result and the tiny increase in statistical accuracy is not worth the effort of adding manually the remaining 350 GUVs. A convenient way of analyzing GUVs is to take around 5 images, each containing typically 50-200 GUVs, run the program and remove the false positives.