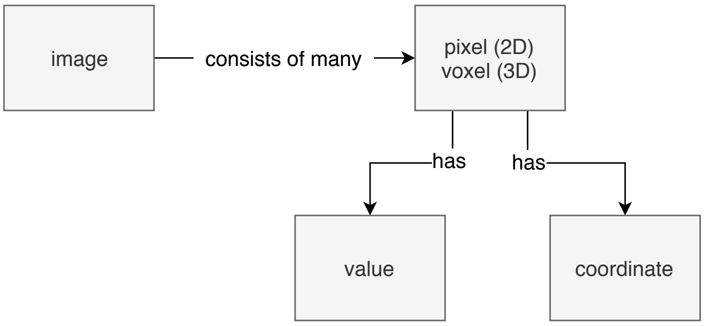
Image Analysis Fundamentals

Basic image properties

- Scientific images are measurements. Thus, it is very important to know how to access the actual values of pixels.

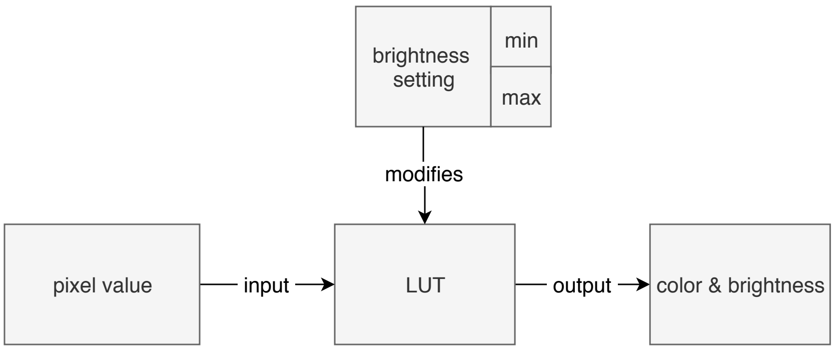


Activities

- Explore different ways of accessing pixel values and coordinates in an image

Lookup tables (LUTs)

- Scientific images are just a collection of many many numbers. However, human brains are typically not very good at taking the information in like this. Thus, lookup tables are used to convert the numbers into colored images that can be much better perceived by our brains.
- Changing the LUT severely changes the perception of an image! Thus, scientist have a great responsibility in carefully choosing the appropriate LUTs (=> "image ethics").

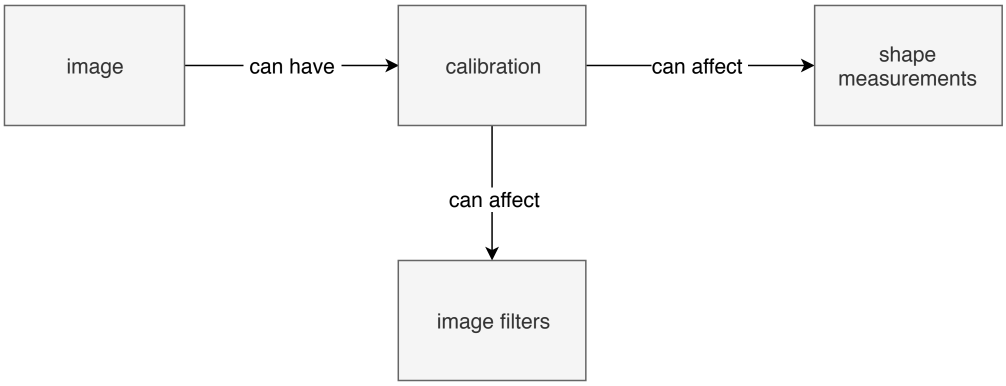


Activities

- Understand how a single color LUT works mathematically: brightness [0,1] = (value - min) / (max - min)
- Explore how "contrast" and "brightness" relate to "min" and "max" settings
- Explore different LUT and brightness settings and discuss their pros and cons
- Discuss how brightness settings can be (incorrectly) used to hide objects in a image
- Visualize current LUT setting by adding a calibration bar
- Learn the importance of equal LUT settings for quantitative comparative image display

Calibration of pixel sizes

- The pixels in an image can be calibrated, that means they can have physical dimensions associated with them.
- In fluorescence microscopy, reflecting the diffration limit, pixel sizes are typically around 100 nm.
- If an image has a calibration the output of shape measurements can be in these units.
- If an image has a calibration the input of image filter parameters can be in these units.
- It often is confusing whether inputs and outputs to and from functions are calibrated or not, thus one needs to be very careful. One option also is to actively remove the calibration...

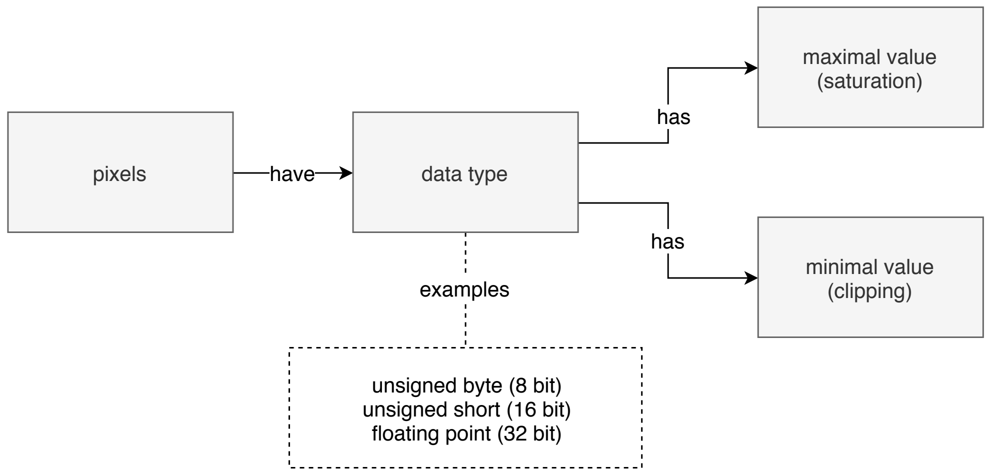


Activities

- Measure length of a line or area of a region while changing the calibration of an image

Image (pixel) data types

- Pixels in images are of a certain data type, which limits the values a pixel can take.
- There are integer data types, where numbers cannot have decimal points, and there are floating point data types with decimal points.

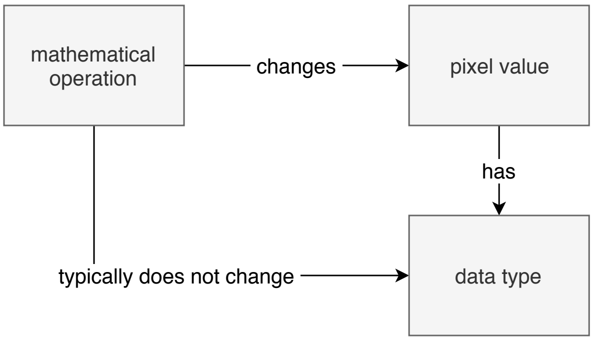


Activities

- Discuss the value ranges of different data types (looking at example images).

Image math

- Many image processing operations change the values of pixels in an image. However, due to the limitations of the pixel data types the results are not always correct. For proper scientific image analysis one needs to understand in which circumstances such incorrect results can occur and how to circumvent this.

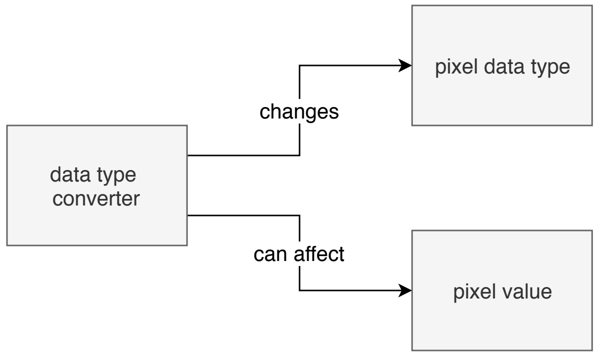


Activities

- Apply mathematical operations on images and observe how the pixel values change (the results can be unexpected, due to limitations of the respective data type).

Data type conversions

- Data type conversions are sometimes necessary, but one has to be very careful, because they may change the pixel values (usually without explicit warning).
- For example, saving images in different file formats can change the pixel values.
- Data type conversions are a common source of errors in image analysis.

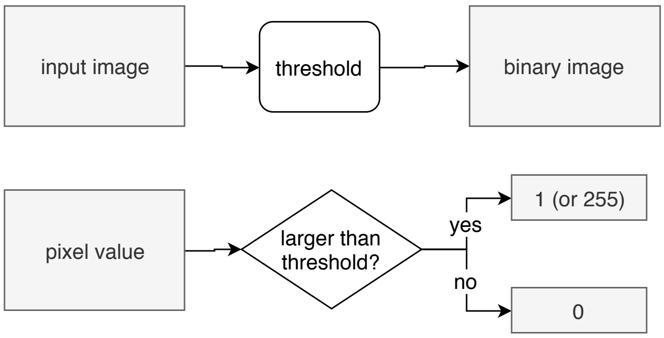


Activities

- Explore whether and how data type conversions change the pixel values.
- Explore how saving images in different file formats changes the pixel values.

Threshold

- In order to find objects in a image, the first step is to determine whether a pixel is part of an object or of the image background. In the vast majority of cases this is done by thresholding.

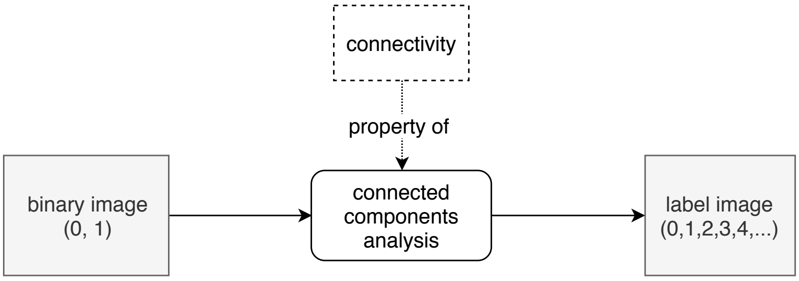


Activities

- Practice thresholding of an image (discuss the difference to gating).

Connected component analysis

- In order to introduce the concept of objects in an image connected component analysis is of utmost importance and is used in all image analysis packages.
- Label images are a great way of representing objects, because (i) it is a well accepted standard, enabling interoperability between different software packages, and (ii) it works in 2D and 3D, (iii) they can be compressed very efficiently.

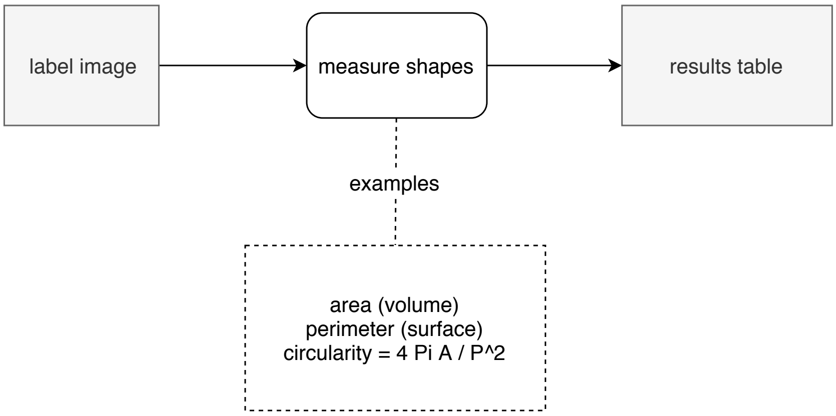


Activities

- Perform connected component analysis on a binary image
- Discuss different choices for connectivity
- Explore multi-color LUTs for visualization of the label images
- Explore removing and joining labels by changing label indices

Shape measurements

- Once objects have been identified, very often one measures their shapes, either to directly address a scientific question or in order to filter out objects that do not obey certain criteria.
- Shape measurements can depend on the pixel size (image calibration).
- Shape measurements are inaccurate for objects close to the resolution limit of the imaging device.
- Especially surface and perimeter measurements are affected by sampling and resolution (see for example: https://en.wikipedia.org/wiki/Coastline_paradox).

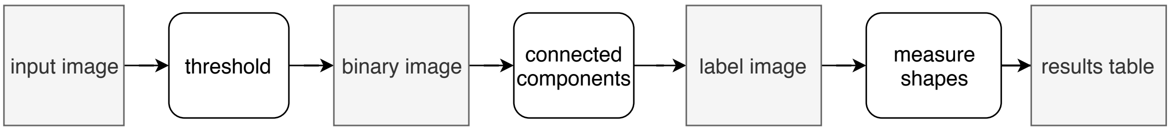


Activities

- Perform shape measurements discuss their meanings.
- Test whether your shape measurements take the image calibration into account at all.
- Discuss which shape measurements depend on image calibration conceptually.
- Draw an image with test shapes to see how the measurements behave.

Object shape measurement workflow

- Measuring the shapes of objects is a very typical image analysis workflow.
- In the simplest case it consist of the components: threshold, connected components analysis, shape measurement.

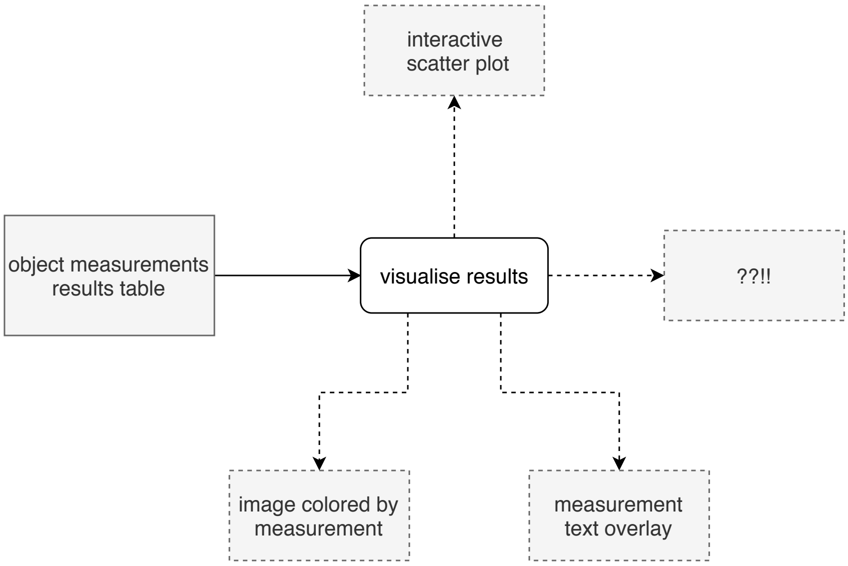


Activities

- Peform above workflow and devise code to automate it.

Results visualisations

- Analysing a object measurements results table is often not very satisfactory, because, for a given measurement, one naturally would like to know where this object was in the image, and how it looked like.
- Thus, knowing good ways of linking the numerical results back to the underlying object (image data) is very very important.

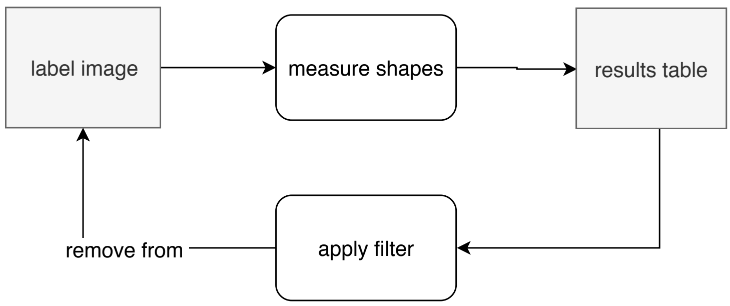


Activities

- Create an image where pixels values whitin each object correspond to a measurement result
- Overlay the measurement results as text on the original image

Filter objects

- Once object shapes have been measured, one often filters the objects based on a measurement results. For example, very small objects may be noise rather than real objects.

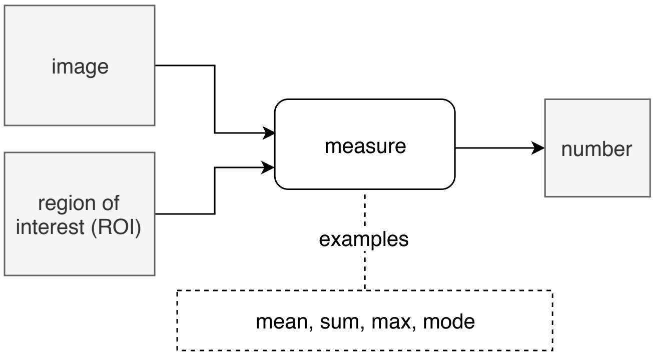


Activities

- Manually remove objects from a label image, settings the label to zero.
- Devise code to remove objects below a certain size.

Intensity measurements

- Intensity measurements in an image are achieved by mathematical operations on a collection of selected pixel values (region of interest, ROI).
- Intensity measurements are very important in biology, e.g. to measure the concentration (or expression level) of proteins in certain locations.

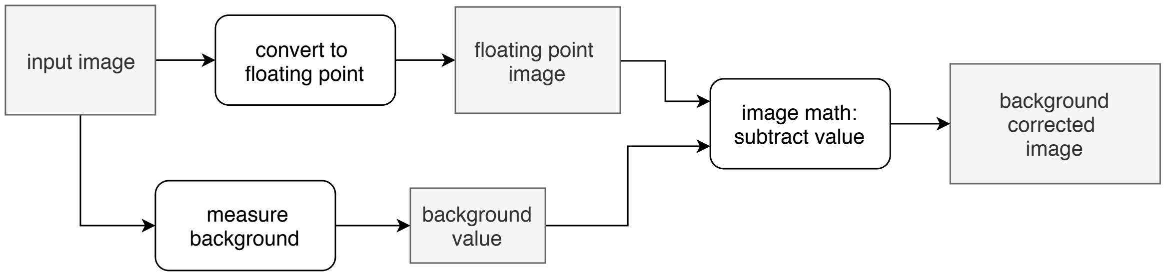


Activities

- Practice intensity measurements in an image using manually selected regions of interest
- Discuss the biological meaning of different measurements
- Discuss the different names for the same measurement
- Discuss how some of the measurements relate to each other
- Observe how the sum intensity (in a background subtracted image) hardly changes with ROI size

Global background subtraction workflow

- Most biological images have non-zero intensity values in regions outside of the objects of interest.
- In order to properly quantify the intensities of objects such background must be subtracted.
- For example, most cameras on microscopes have a read noise with can be many hundred gray values (for 12bit or 16bit detection). As such read noise is typically constant across the whole image, subtracting a constant background value from each pixel is possible.
- Before background subtraction, images should be converted to floating point to accomodate negative values and floating point values.
- Automated measurements of the background intensities often is a challenging aspect of biological image analysis.

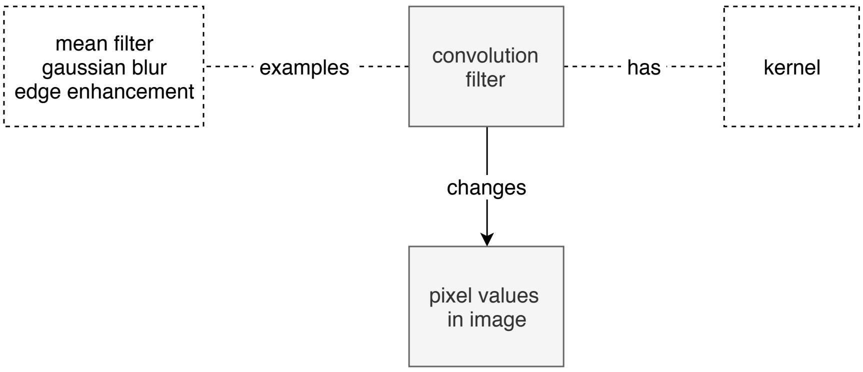


Activities

- Perform background subtraction on an image.
- Observe that one obtains negative values.
- Discuss automated background estimation methods (e.g., mode, out-side objects)

Convolution filters

- Image filtering is very useful, e.g. to reduce noise and thereby make segmentation easier.
- Convolutional filters are a very important class of image filters.
- Convolutional filters are local, i.e. the result of the filter operation depends only a neighborhood of pixels around a central pixel

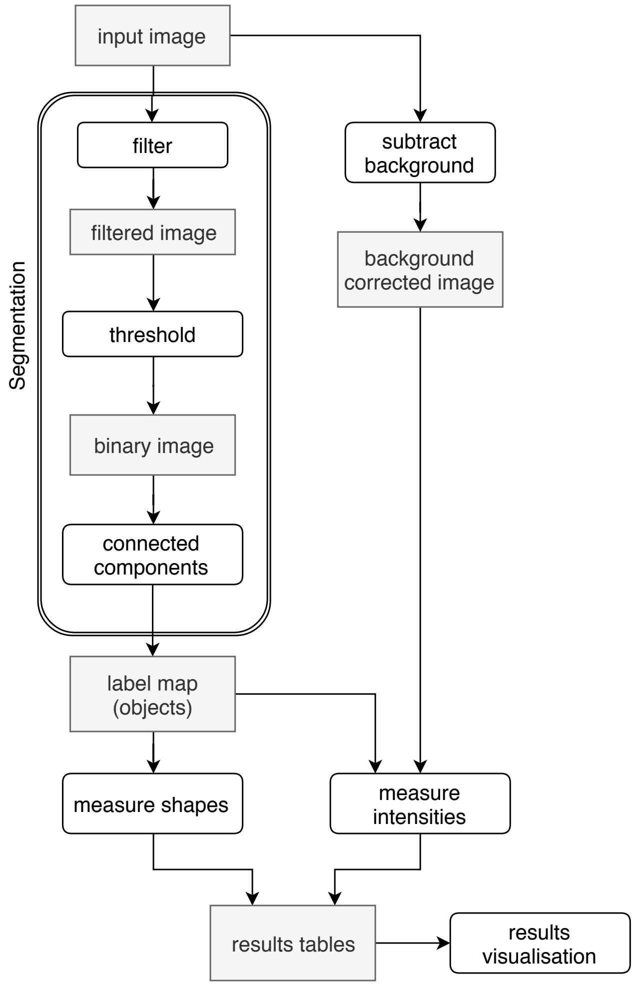


Activities

- Discuss the kernel of a mean filter and practice applying it.
- Discuss the kernel of a gaussian filter (look it up in the internet).
- Discuss that one can sometimes normalise filters such that they do not change the sum intensity of the whole image (but this is not always the case and one should not rely on it).
- Explore the effects of different kernels.
- Application: Apply a smoothing (e.g., mean or gaussian) filter to reduce noise and thereby make thresholding easier

Typical image analysis workflow

- Here, one of the most typical image analysis workflow is depicted. Each of the steps can be of varying complexity, depending on the input data.
- Note that intensity quantification follows a different branch than segmentation. This is important, because image filtering during segmentation might corrupt intensity values.



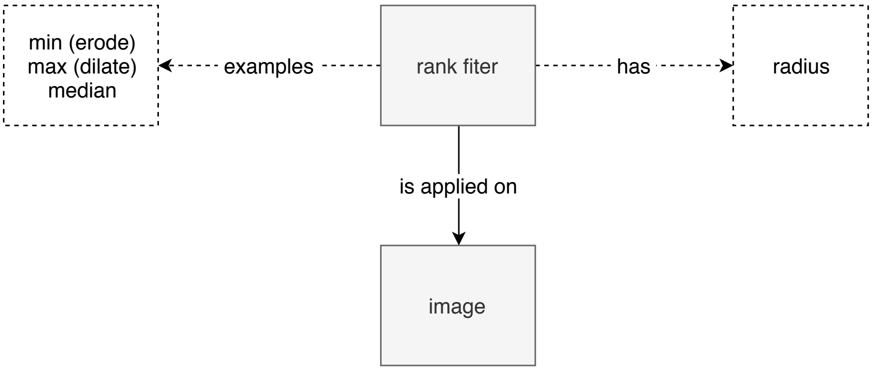
Activities

- Implement such a workflow.
- Discuss each step in the pipeline and discuss alternative implementations.
- Discuss where object filtering could happen (pros and cons)

Rank Filters

Rank filters

- Rank filters are local filters based on sorting (ranking) pixel values.
- Rank filters work like this: (i) sort the pixel values in a certain neighborhood around a central pixel, (ii) pick one value (x) based on its position (rank) in this sorted list of values, (iii) replace the central pixel by x.
- Rank filters are very useful both for gray scale and binary images.

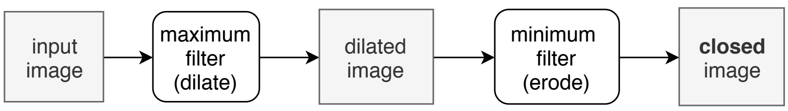


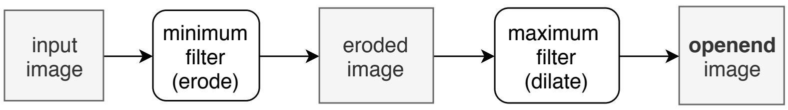
Activities

- Understand how min, max and median rank filters work conceptually (=> whiteboard session)
- Explore applying rank filters to binary images
- Explore applying rank filters to grayscale images
- Discuss the different names of rank filters (minimum, erosion, ...)

Rank filter sequences (morphological filtering)

- Applying rank filters in all kind of sequences is very useful in image analysis.
- This important image analysis field is also called morphological filtering.
- Morphological filtering can be applied both to binary and grayscale images.



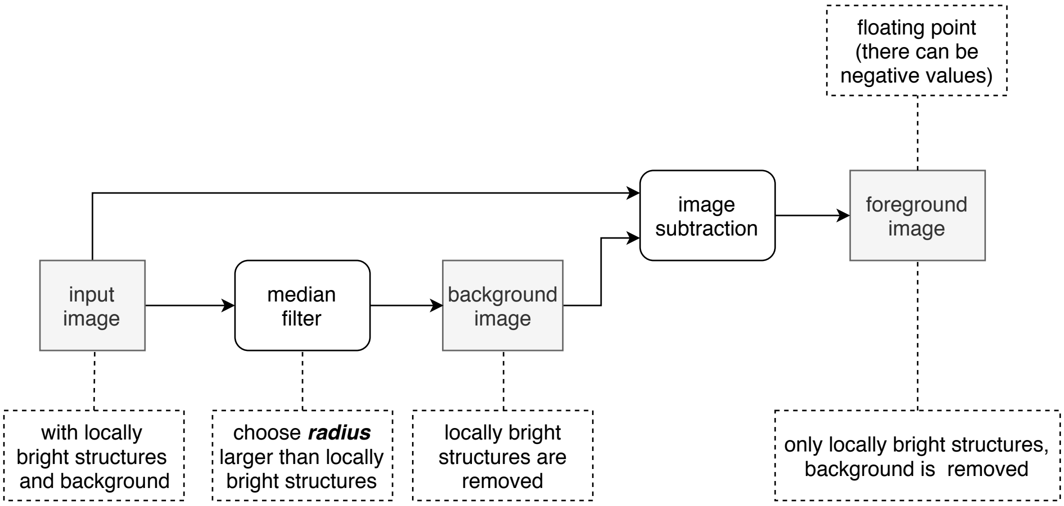


Activities

- Explore the effect of above rank filter sequences on some images

Local background subtraction using a median filter

- Biological images very often have "background" that is not constant across the image.
- For example, it often is the case that one wants to quantify protein localised to small dot-like (locally bright) structures in the prescence of diffuse protein.
- Subtracting a median filtered image with a radius larger than the radii of the objects of interest is a very good and popular method for removing uneven background.
- This works well also in the presence of noise.
- It is not perfect at corners of background structures.

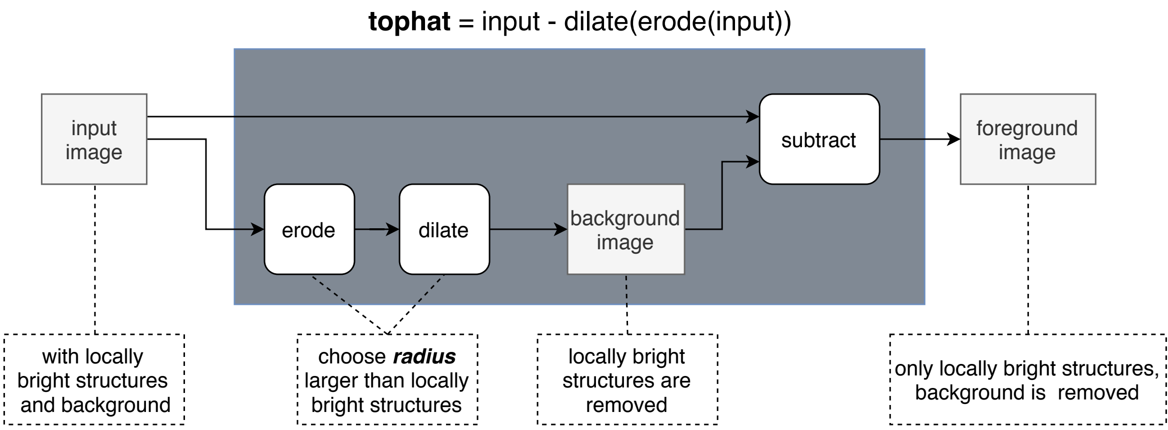


Activities

- Peform above workflow on a example, exploring different radii for the median filter.
- Observe that this approach does not work perfectly at the corners of background structures.

Local background subtraction using a tophat filter

- Biological images very often have "background" that is not constant across the image.
- For example, it often is the case that one wants to quantify protein localised to small dot-like (locally bright) structures in the prescence of diffuse protein.
- Subtracting a morphological opening (i.e. performing a tophat filter) with a radius larger than the radii of the objects of interest is a very good and popular method for removing uneven background.
- In the prescence of noise it leads to non-zero background (=> incorrect intensity quantification).
- It typically works rather well also at corners in the background.



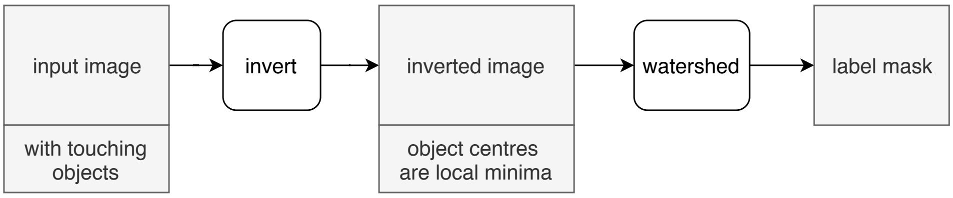
Activities

- Peform above workflow on a example, exploring different radii for the median filter.
- Observe that this approach works rather well at the corners of background structures.

Object splitting & Distance maps

Object splitting by intensity based watershed

- Objects are often very close such that they might end up having the same label during the connected components analysis.
- There are several ways how to split such "touching objects"; one important method is a so-called "intensity-based" watershed filter.

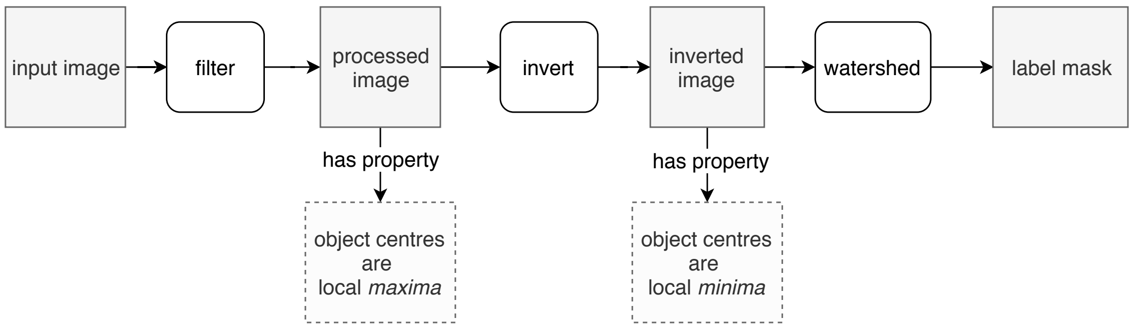


Activities

- Perform above workflow on an example image and understand the concept.

Object splitting by intensity based watershed
Workflow

- Usually some preprocessing (filtering) is required before the watershed filter can be applied.
- The purpose of the filtering is to ensure that the only local maxima in the image are corresponding to object centres (often there are other maxima that one is not interested in at this point).

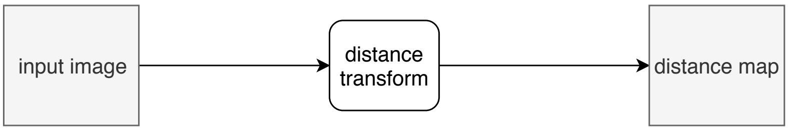


Activities

- Perform above workflow on some examples and create code to automate it

Distance map

- The distance transform changes pixel values from binary pixels to distances.
- It is defined such that the distance is measured to the nearest background pixel (value zero).
- It is very useful, both for several applications: (i) actual distance measurements, (ii) quantify the shape of objects, (iii) shape-based object splitting.

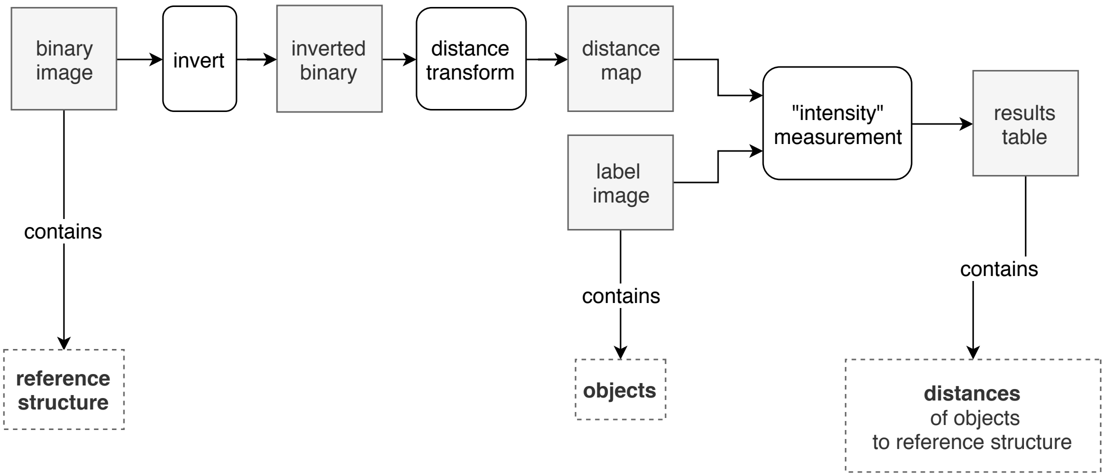


Activities

- Try it out on a binary example image.
- Observe limitations of 8bit output format for distance maps
- Observe limitations of different implementations (e.g., City-Block)

Distance measurement workflow

- Below workflow demonstrates how to measure distances of objects (label image) to a reference structure (binary image).
- Note that distance measurements are always to the closests background pixel (i.e. to the surface of a reference object).
- If you want distances to the center of objects, a thinning step is required.

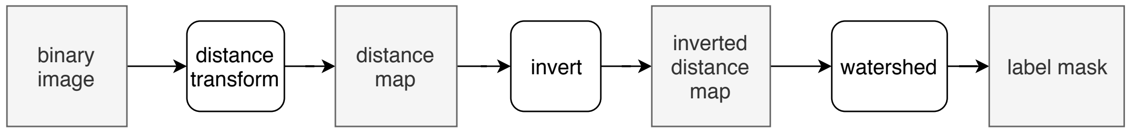


Activities

- Try above workflow on an example.

Object splitting by shape based watershed

- Objects are often very close such that they might end up having the same label during the connected components analysis.
- There are several ways how to split such "touching objects"; one important method is a so-called "shape-based" watershed filter.



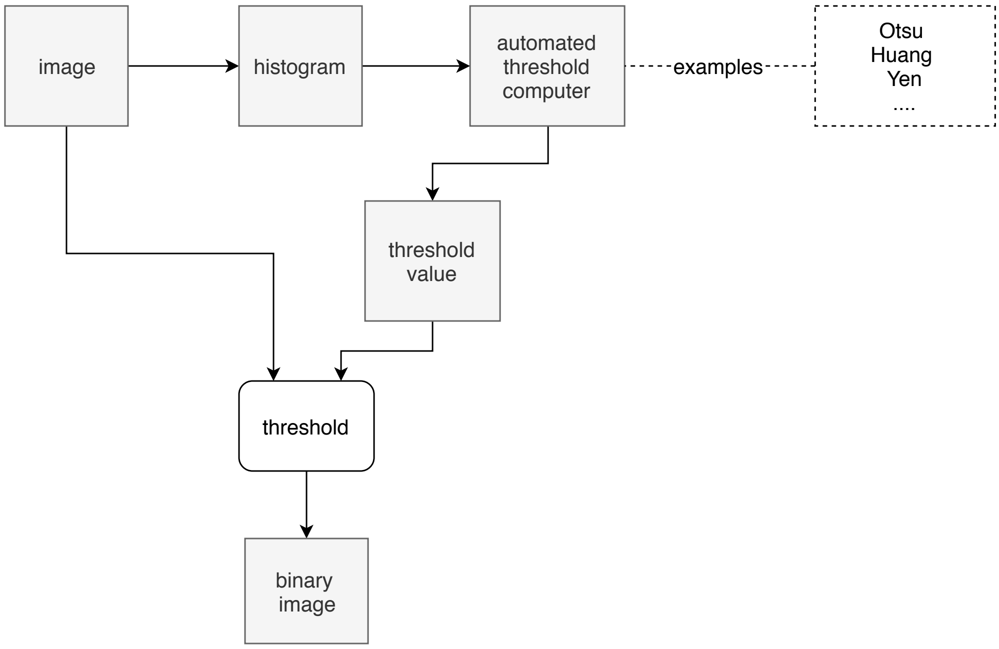
Activities

- Perform above workflow on an example image and understand the concept.

Automated Thresholding

Automated global thresholding

- When batch processing many images one fixed threshold may not always suffice to segment objects. In such cases automated thresholding may be necessary.
- Most automated thresholding methods work on the image histogram, i.e. they do not take spatial context into account.
- Some advanced knowledge is required when choosing the automated thresholding method. For example, some methods work very well when only few percent of the pixels are foreground pixels, while methods work better when half or more of the pixels are foreground.
- The following website contains links to many relevant methods: **https://imagej.net/Auto_Threshold**

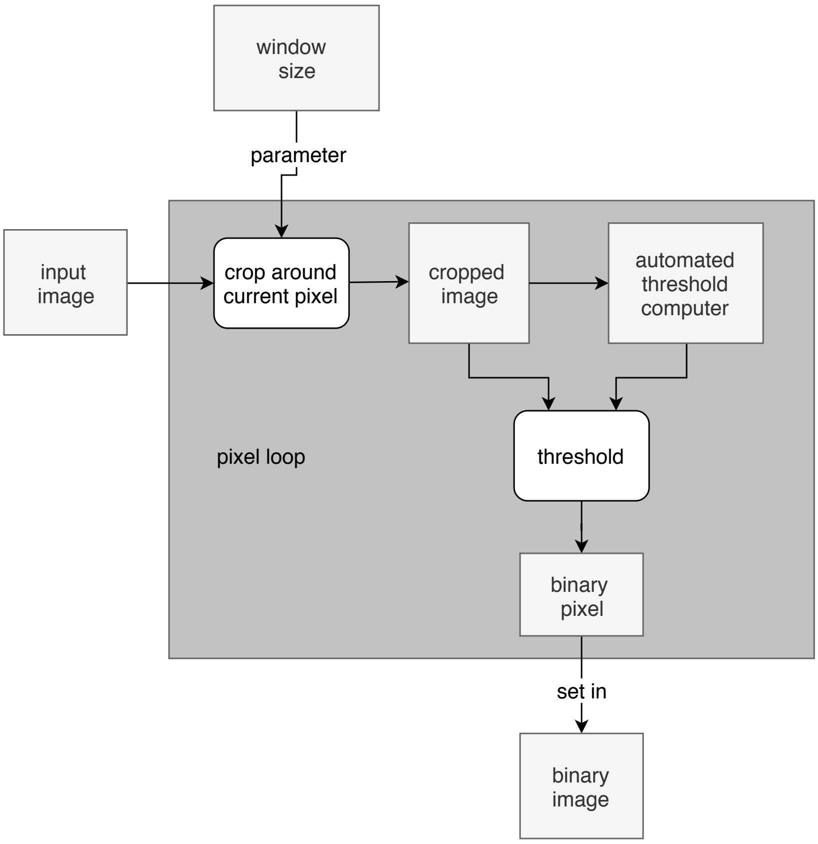


Activities

- Try different automated thresholding methods on different images (make sure to also try with an image containing only "noise").

Automated local thresholding

- Conceputally, automated local threshold means running an automated (global) threshold in small windows across the image and only changing the central pixel of the current window.
- Automated local thresholding can be useful when (i) there is an uneven image background or (ii) when then the objects do not have the same intensities.
- In practice however, the implemented local threshold methods are sometimes entertaining different ideas for finding the threshold that the global implementations.
- A summary of local threshold algorithms with references can be found here: https://imagej.net/Auto_Local_Threshold



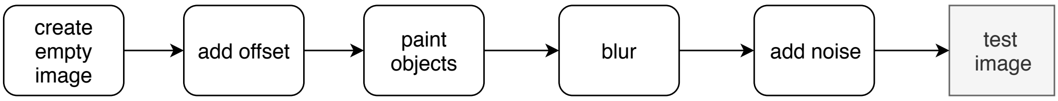
Activities

- Explore different auto-local-threshold methods on different input data.

Test Image
Generation

Test image generation

- To explore how different image analysis algorithms work it is very useful (alomst necessary) to be able to generate test images with well defined properties. For example one would like to control exactly the number of objects, the objects shapes and intensities, the level of blurring, the noise level and the background level.
- Without such test images it can be almost impossible to really check what an image analysis algorithm is doing in detail.



Activities

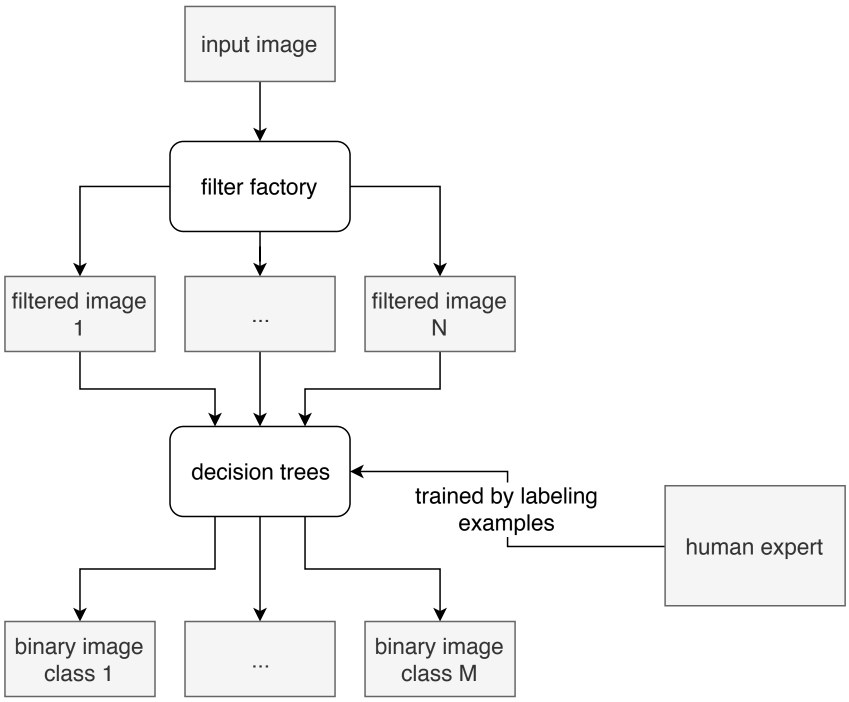
- Generate a test image.

Machine Learning:

Supervised Pixel Classification

Machine learning based pixel classification

- Finding the right combination of filters that lead to an image which can be easily thresholded can be tedious.
- It is possible to let a computer find the right combinations of filters to highlight certain structures in images.



Activities

- Discuss how thresholding can be viewed as a very simple form of "pixel classification"
- Try it out on an image