

ASAP guide: methodology and workflow

Version 1.0

Last updated: 21.03.2019

Introduction

ASAP combines existing automation, computer vision, machine learning and data visualization tools to push the frontiers of super resolution microscopy data analysis. ASAP is capable of simulating, identifying, analyzing and classifying complex nanoscopic structures with the aim of turning super resolution microscopy into a tool for structural biology. In this document, the modular workflow of ASAP is elaborated to allow users understand operations performed at the backend and developers identify opportunities for improvement.

Description of parameters

The operation of ASAP depends on parameters to be set by the user. Complete description of all the ASAP parameters can be found in the **ASAP guide: description of parameters and selection of values**.

Simulation (Simulate) module

The simulator receives an image as input draws a convex hull around the putative border of the sample within, generates pointillistic positions across the field of view and accumulates 2D Gaussian profiles to mimic actual diffraction-limited fluorescent bursts in single molecule-based localization microscopy.

Generation of convex hull

For the generation of the convex hull (i.e. smallest polygon encompassing all super-resolved structures), the simulator finds the edge of all bright pixels in an image, dilates the edges with a sphere-shaped structured element, fills the dilated spaces, selects the region of maximum area and assigns the convex hull to the selected region. The `segmentationLevel` parameter (P4) is provided as input. If the `segment` parameter (P3) is assigned `false`, the convex hull is chosen to be the entire image.

Generation of positions

For the generation of positions, the simulator receives as an input the `simulationMode` parameter (P1) which takes one value `Random` or `Structured`.

When assigned `Random`, the simulator generates random positions across the convex hull. The number of positions generated is given by the `numFluorophore` parameter (P9) input to the simulator function.

When assigned `Structured`, the simulator generates positions for n-epitope rings which are scattered randomly across the convex hull. The number and size of the n-epitope rings is given by the `numStructure` and `structureSize` parameters (P11 and P12) as well the `numEpitope` parameter (P10).

Image generation

If the `simulationMode` parameter is assigned `Structured`, each structure is assigned a labeling efficiency which conforms with the average value assigned to the `labelingEfficiency` parameter (P13). For `Structured` and `Random` assignments, ASAP assigns a labeling angle for each fluorophore located at each of the generated positions. The generated positions are then adjusted by the `labelLength` parameter (P7) multiplied by the labeling angle for each emission cycle. Gaussian profiles are then accumulated at the generated positions using the `photonCount` parameter (P6) as the amplitude and the `lateralPrecision` parameter (P8) as the width of the Gaussian profile

Please refer to the **supplementary text** and the **ASAP guide: description of parameters and selection of values** for sample images exported by this module.

Identification (**Identify**) module

The identifier receives as input a super-resolved image, generates a convex hull, segments bright-from-dark pixels, clears the image of stray localizations located close to its border, isolates structures from background and excludes structures containing a below-minimum and above-maximum number of pixels.

Generation of convex hull

For the generation of the convex hull (i.e. smallest polygon encompassing all super-resolved structures), the identifier finds the edge of all bright pixels in an image, dilates the edges with a sphere-shaped structured element, fills the dilated spaces, selects the region of maximum area and assigns the convex hull to the selected region. The `segmentationLevel` parameter (P17) is provided as input. If the `segment` parameter (P16) is assigned `false`, the convex hull is chosen to be the entire image

Intensity segmentation

To segment bright-from-dark pixels, ASAP offers the user two threshold modes which are accessible through the `thresholdMode` parameter (P18), `Fixed` and `Relative`. Automatically-generated single- and multi- value threshold levels can be further refined by the user through the `thresholdMultiplier` parameter (P19).

Image clean-up

The intensity-segmented image is further cleaned from lone localizations using the `maxClearParticleSize` and `maxClearSearchRadius` parameters (P21 and P22). ASAP deploys a running window to search for lone localizations. If a bright localization is not surrounded by bright pixels more than the value assigned to the `maxClearParticleSize` within a radius equal to the value assigned to `maxClearSearchRadius`, that pixel is removed. Additionally, ASAP will remove structures located close to the image border if the `clearBorder` parameter is assigned `true`.

Structure identification and mark-up

ASAP identifies structures using 2 distinct modes which are accessible through the `identificationMode` parameter (P15). If `Connectivity` is assigned, ASAP assigns bright pixels connected to each other unique cluster IDs. If `Size` is assigned, ASAP searches for structures smaller than the value assigned to the `maxStructureSize` parameter (P25) using a novel size-based clustering algorithm. Following identification, structures are marked with white-boxes for visual inspection.

Structure exclusion

To ensure that exceedingly small structures and / or large structures are not being processed further, the user can exclude such structures using the `minSize` and `maxSize` parameters (P26 and P27).

Please refer to the **supplementary text** and the **ASAP guide: description of parameters and selection of values** for sample segmented images exported by this module.

Analysis (**Analyze**) module

Morphology operations

Morphological operations are performed on the structures extracted from the output file previously produced in the `Identify` module. Morphological operations may enhance or downgrade the fidelity of the analysis and classification process depending on the initial morphology of the underlying structures. The user is advised to consult [here](#) for further information.

Descriptor extraction

Following assignment of the `pixelSize`, `analysisPlatform`, `analysisMethod` and `maxRingSize` parameters (P28 – P31), ASAP extracts a list of shape descriptors from each structure. The extracted descriptors depend on the value assigned to `analysisMethod`. If `pixelCounting` is assigned, the following descriptors are calculated by counting the bright pixels alone:

Descriptor	Mathematical definition	Description
Area	# bright pixels	Area of structure
Filled area	# bright pixels + # dark pixels surrounded by bright pixels	Area of structure assuming empty core is bright. As an example, a disc will have the same area and filled area. However, a ring or doughnut will have different area and filled area.
Convex area	# pixels inside the smallest polygon with least number of sides (also known as convex hull) that encloses the structure	Area of structure assuming empty grooves are bright. As an example, a disk will all have the same area and convex area. However, an arc, ring or doughnut will have a different area and convex area.
Perimeter	# pixels on the periphery of a structure	Perimeter of structure
Euler number	# structures - # holes in a structure	Number of holes in a structure. As an example, a disc will have an Euler number of 1. A ring will have an Euler number of 0. Note: for unconnected structures the # structures will be the # connected structures.
Eccentricity	Ratio of the distance between the foci of an enclosing ellipse and its major axis length	A disc, circle or doughnut will have an eccentricity of 0, while a line will have an eccentricity of 1.
Solidity	Area / convex area	Proportion of pixels in region that are also in convex hull. As an example, a disk will have a solidity of 1. However, an arc, ring or doughnut will have a solidity less than 1.
Orientation	Angle between major axis length of an enclosing ellipse and the x-axis	Can be used to differentiate between connected structures with different orientations.
Extent	Area / Bounding box area	Proportion of pixels in region that are also in bounding box. As an example, a line or rectangle will have an extent of 1, while, a disk, arc, ring or doughnut will have an extent less than 1.
Major axis length	Longest segment of an enclosing ellipse passing through its center	Major axis length of an enclosing ellipse
Minor axis length	Shortest segment of an enclosing ellipse passing through its center	Minor axis length of an enclosing ellipse
Form factor	$4 * \text{Area} / \text{Perimeter}^2$	Measure of the ruggedness of a structure's perimeter. As an example, can be used to differentiate between perfect and rugged ring.
Roundness	$4 * \text{Area} / \text{Major axis length}^2$	Measure of curvedness of a structure. As an example, can be used to differentiate between ellipse, disc and rectangle.
Elongation	Major axis length / minor axis length	Measure of elongation, can be used to differentiate between round and elongated structures.
Fill ratio	Filled area / ($\pi * \text{major axis length} * \text{minor axis length} / 4$)	Proportion of pixels in filled region to that in an enclosing ellipse. As an example, a ring has a higher fill ratio than an arc.
Mean intensity	Sum of bright pixels in a structure	Measure of labeling density. Can be used to differentiate between two equally-sized structures with different labeling densities.
Minima number	# local minima in a structure	Can be used to differentiate between a structure with single hole and another with multiple holes.

Minima intensity	Mean intensity of minima	Can be used to differentiate between a structure with a shallow hole and another with a deeper hole.
Minima eccentricity	Mean eccentricity of minima	Can be used to differentiate between a structure with a circular hole and another with an elliptical hole.
Minima area	Sum of minima area in a structure	Can be used to differentiate between a structure with a large hole and another with a smaller one.
Minima convex area	Sum of minima convex area in a structure	Together with Minima area can be used to differentiate between a structure with a circular hole and another with an arc shaped hole.
Maxima number	# local maxima in a structure	Can be used to differentiate between a structure with single peak and another with multiple peaks.
Maxima intensity	Mean intensity of maxima	Can be used to differentiate between a structure with a shallow peak and another with a higher peak.
Maxima eccentricity	Mean eccentricity of maxima	Can be used to differentiate between a structure with a circular peak and another with an elliptical peak.
Maxima area	Sum of maxima area in a structure	Can be used to differentiate between a structure with a large peak and another with a smaller peak.
Maxima convex area	Sum of maxima convex area in a structure	Together with Maxima area can be used to differentiate between a structure with a circular peak and another with an arc shaped peak.
Segment total length	Total length of the skeleton of a structure	Can be used to differentiate between structures with differing skeleton length.
Segment number intersections	# intersections in a structure	Can be used to differentiate between structures with different number of intersections.
Objects	# connected objects	Can be used to differentiate between a structure that is entirely connected and another that is composed of sub structures.

If `ellipseFitting` is assigned, the following descriptors are calculated by fitting the bright pixels with an ellipse using the least squares criterion:

Descriptor	Mathematical definition	Description
Major axis length fit	Longest segment of the fitting ellipse passing through its center	Can be used to measure the length of an unconnected ring or ellipse and to differentiate between both if having the same area.
Minor axis length fit	Shortest segment of the fitting ellipse passing through its center	Can be used to measure the width of an unconnected ring or ellipse and to differentiate between both if having the same area.
Elongation fit	Major axis length fit / minor axis length fit	Measure of elongation, can be used to differentiate between round and elongated structures.
Relative centroid	Ellipse centroid - centroid of bounding box	Can be used to differentiate between closed and non-closed structures.
Orientation fit	Angle between major axis length of an enclosing ellipse and the x-axis	Can be used to differentiate between non-connected structures with different orientations.

If `radialProfiling` is assigned, the following descriptors are calculated by counting the number of pixels between one-pixel-spaced centering circles reaching a maximum size given by `maxRingSize`. The resulting radial profile can be later fit to a mathematical form that describes the shape geometry in the `plot` module.

Descriptor	Mathematical definition
Raw radial profile	Accumulation of the intensity of the bright pixels located between 1 pixel spaced centering

	circles as function of distance from the centroid of the structures.
Intensity normalized radial profile	Accumulation of the intensity of the bright pixels located between 1 pixel spaced centering circles as function of distance from the centroid of the structures divided by maximum intensity.
Area normalized radial profile	Accumulation of the intensity of the bright pixels located between 1 pixel spaced centering circles as function of distance from the centroid of the structures divided by the cumulative intensity.
Raw radial density profile	Accumulation of the intensity of the bright pixels located between 1 pixel spaced centering circles divided by the number of bright pixels as function of distance from the centroid of the structures.
Intensity normalized radial density profile	Accumulation of the intensity of the bright pixels located between 1 pixel spaced centering circles divided by the number of bright pixels as function of distance from the centroid of the structures divided by maximum intensity.
Area normalized radial density profile	Accumulation of the intensity of the bright pixels located between 1 pixel spaced centering circles divided by the number of bright pixels as function of distance from the centroid of the structures divided by the cumulative intensity.

Training (Train) module

Structure annotation

The user categorizes (annotates) the identified and analyzed structures into different shapes (classes) dictated prior to the annotation process.

Classifier training

Following annotation of a subset of structures, an assigned classifier can be trained to recognize the annotated structures using one or more of the previously selected shape descriptors. Selection of the classifier and descriptor(s) can introduce significant biases to the classification process or accelerate the convergence of the classifier.

For higher classification accuracy, the user should reject structures with low mean signal-to-noise ratio (or insufficient localizations in case of single molecule localization microscopy), experiment with the different classifier types available and refine the list of shape descriptors used for training.

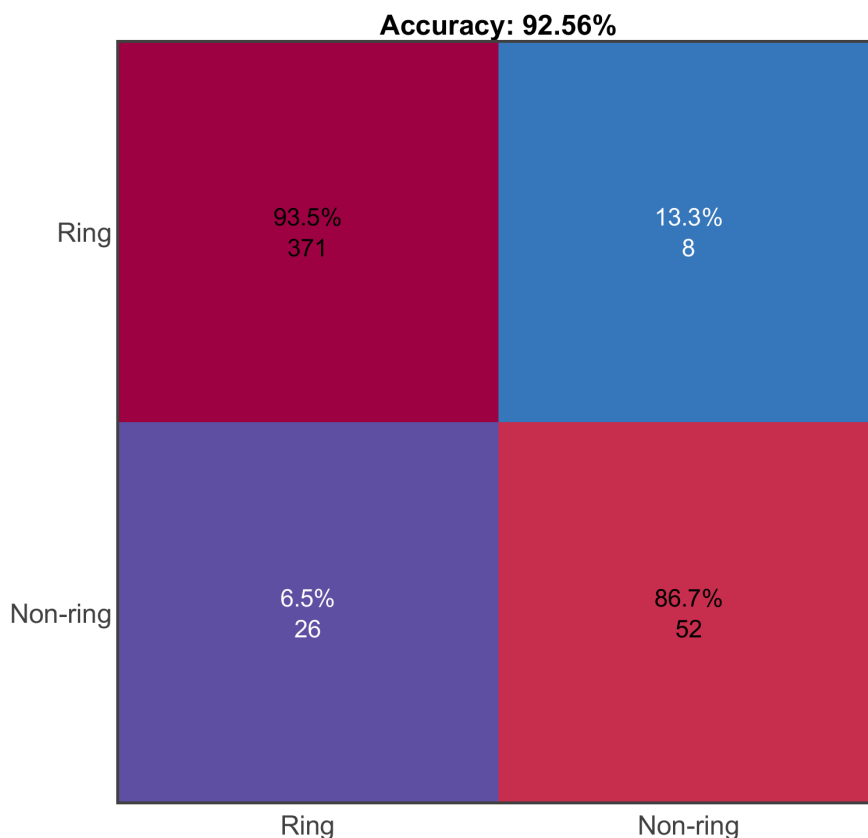


Figure 1 an example confusion matrix produced after training the classifier to recognize ring and no-ring structures. In this particular example, the classifier achieved a 93.5% accuracy in recognizing rings and 86.7% accuracy in recognizing non-rings structures. The numbers beneath the classification accuracy of each shape is the number of structures the classifier was trained on.

Classification (**Classify**) module

Following the training process on a subset of the identified structures, the user proceeds to classifying all the identified structures. This is a simple process merely comprised of selecting all the ASAP files containing data of the identified structures. The software automatically searches for the classification model file produced during the training process and saved in the project folder. Using the classification model, ASAP automatically classifies the rest of structures.

Prior to classification, the user is given the option to perform shape re-assignment. This is particularly useful when one or more of the underlying shapes is not abundant enough. For shape re-assignment, the user can choose one or more shapes to be re-assigned to one shape based on bounds imposed on a shape descriptor. Multiple re-assignments could be performed to fully assign the minority of inappropriately classified structures to their correct shape categories. Following shape re-assignment and classification, the user can review the annotation of each structure and correct the assignment of each, some all or none as deemed appropriate.

Clustering (**Cluster**) module

Following classification, the user can further cluster structures belonging to one or more shape class(es) into groups based on the values of their descriptors. Clustering is a powerful tool to detect morphological differences in similarly-classified structures from one protein and correlate this with the spatiotemporal organization of the cellular milieu and / or groups of other structures originating from another protein.

The number of clusters can be assigned to the `clusterNum` parameter (P46) if known *a priori* and as assigned to the `clusterNumKnown` parameter (P45). If the number of clusters is unknown, the software

can automatically cluster structures to maximize the silhouette coefficient. The silhouette coefficient measures the similarity between one structure and others in the same group to those in other groups.

To improve the accuracy of clustering when the number of clusters is not known *a priori*, the user may choose to cluster data from a large number of similar data sets through assignment of the `clusterAll` parameter (P44). The user should warrant the similarity between the different data sets to produce trustable data clusters.

Following clustering, and if more than one data set is being clustered at a time, ASAP proceeds with calculating the Jaccard Indices to measure the similarity between the groups in one image and another.

Please refer to the **supplementary text** and the **ASAP guide: description of parameters and selection of values** for sample data exported by this module.

Plotting (Plot) module

All previously-generated data (except for clustering data) can be flexibly plotted and mathematically-fitted using the plot module. The module is based on the GRAMM (Grammar of graphics plotting in MATLAB) [1] data visualization tool box. The user can choose to vary a large number of parameters including the files to include, shape class, plot affiliation, descriptors, fit type & equation and appearance. Plotted data is exported as image as well raw data files which can be disseminated or used for further analysis.

Please refer to the **main** and **supplementary texts** for sample plots exported by this module.

Montaging (Montage) module

The identified and classified structures can be montaged into a gallery using this module. The user can vary a wide array of parameters to modify the gallery including the shape class of each structure, scale bar, label, file and position in gallery.

Please refer to the **main** and **supplementary texts** for sample galleries exported by this module.

Data export

Data is exported by all the modules of ASAP. ASAP exports raw data in the tab-delimited `.txt` file format which can be opened in many text editors and spreadsheet software. Additionally, ASAP exports unique files in each module which can be openable using ASAP and which are only useable by certain modules. The following is a list of the different files exported by ASAP:

Extension	Exporting module	Importing module
asid	Identify	Analyze
asan	Analyze	Train, Classify
astr	Train	Classify
ascl	Classify	Cluster, Plot, Montage
ascu	Cluster	

Automation

Full automation of the ASAP workflow can be achieved using the automation script; a single text file containing all parameters required to run ASAP.

Please refer to the **ASAP guide: automation script syntax** for further information on the syntax and execution of the automation script.

References

[1] P. Morel (2018). Gramm: grammar of graphics plotting in Matlab. Journal of Open Source Software, 3(23), 588.