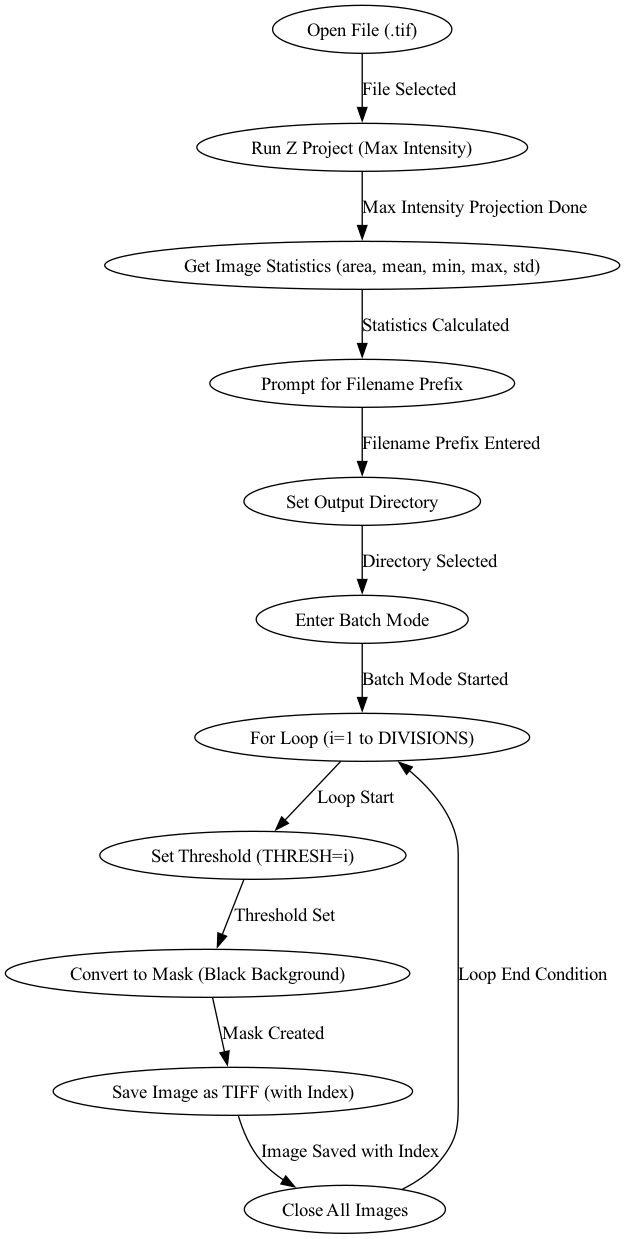
**Marco1**

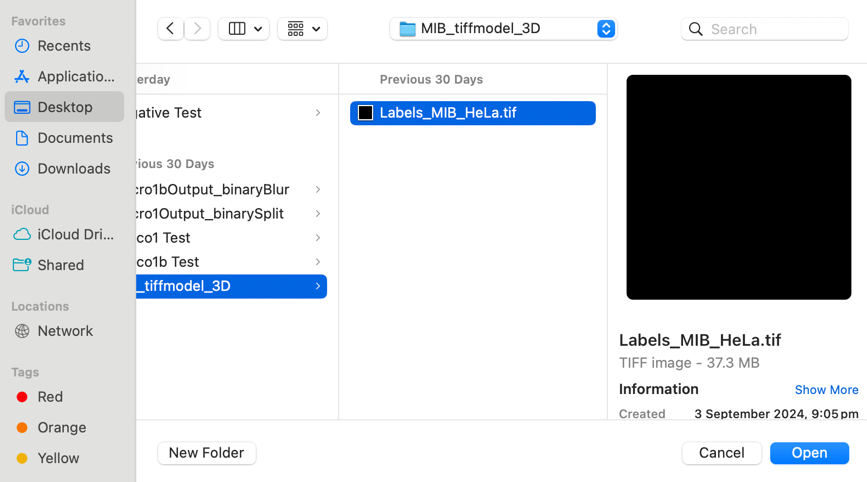
1. What’s Marco1 use?

Marco 1 is designed to split binary masks created after 3D image stack segmentation. This process is essential for separating segmented structures into distinct parts for detailed analysis. It is particularly useful for complex biological structures like mitochondria, where precise division is needed for accurate study. By splitting masks along the xy, xz, and yz planes, Marco1 enhances the refinement and clarity of segmented regions, ensuring the data is ready for further investigation.

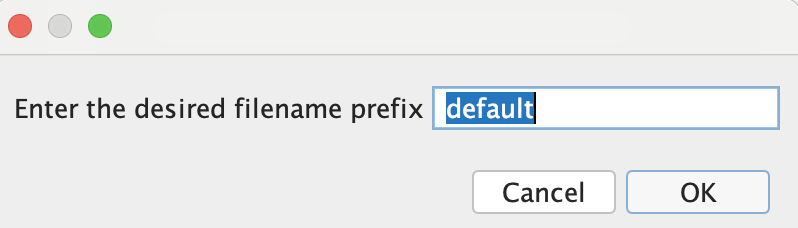
1. Process diagram



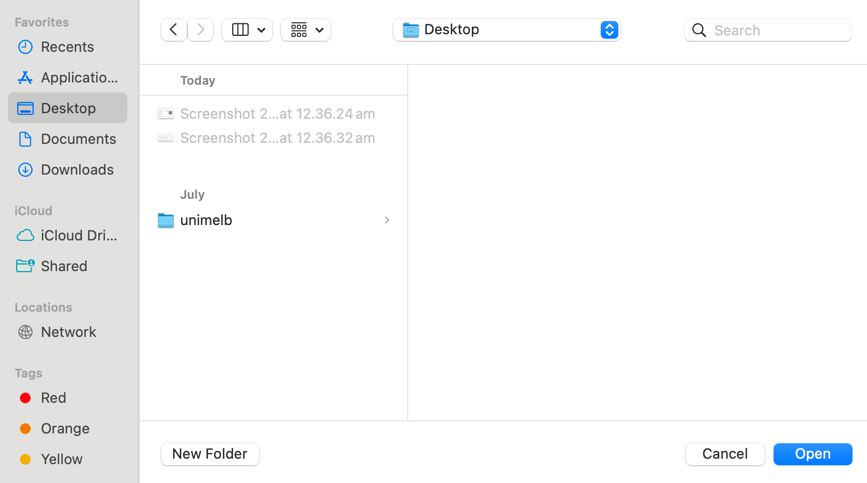
1. **Open File (.tif):** Select and open a .tif file (an image stack) for processing.
2. **Run Z Project (Max Intensity):** Create a single 2D image from a 3D stack by taking the brightest pixel at each point.
3. **Get Image Statistics:** Calculate key statistics like area and average brightness to understand the image better.
4. **Prompt for Filename Prefix:** Enter a name that will start all the output files, helping organize the results.
5. **Set Output Directory:** Choose where the processed images will be saved.
6. **Enter Batch Mode:** Automate the processing of images without stopping between steps.
7. **Loop through Divisions:** For each predefined setting (division), adjust the image to highlight different parts.
8. **Set Threshold:** Define a brightness cutoff to focus on important features in the image.
9. **Convert to Mask:** Change the image to show only the features above the threshold against a black background.
10. **Save Image as TIFF:** Save the processed images with a name that includes the division number.
11. **Close All Images:** Finish processing by closing all open images.
12. How to use it in Fiji and Python
    1. Fiji (Based on JAVA language)
       1. **Select input file:** Look the .tif file



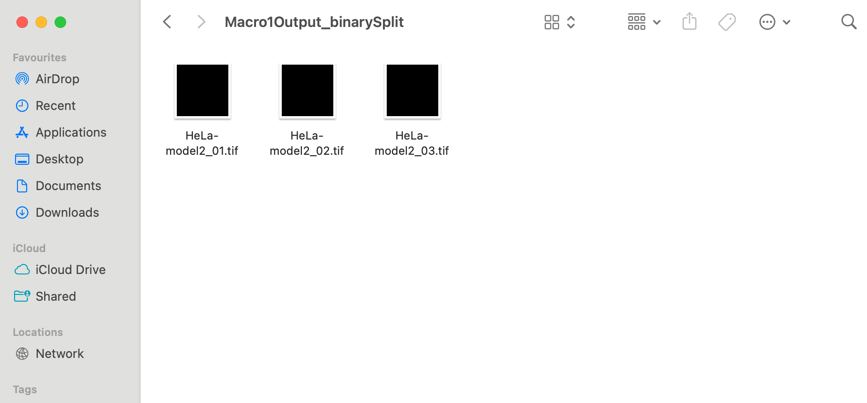
* + 1. **Enter the filename prefix:** default or anything you want



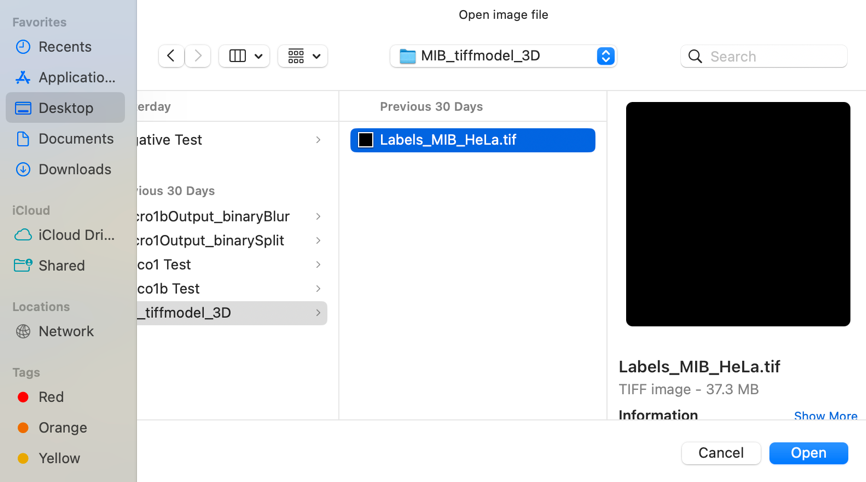
* + 1. **Select the output directory:** Define the directory for saving the processed output



* + 1. **Save the result:** The final blurred images are saved



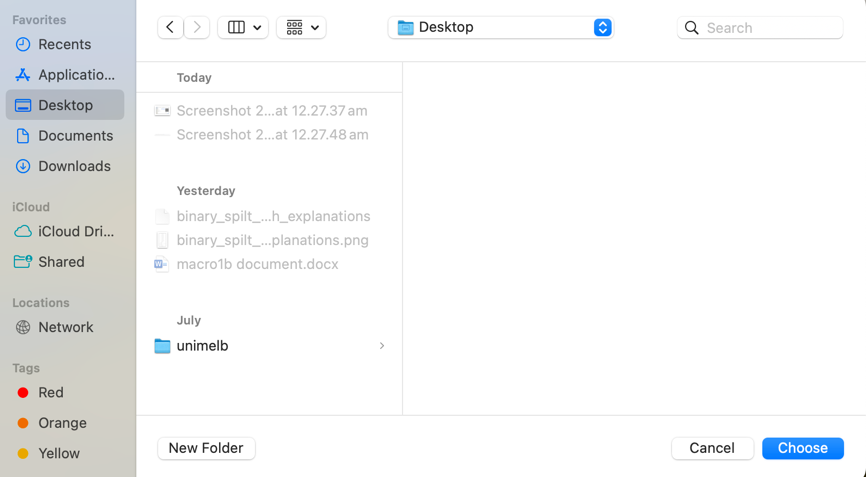
* 1. Python
     1. **Select input file:** Look the .tif file



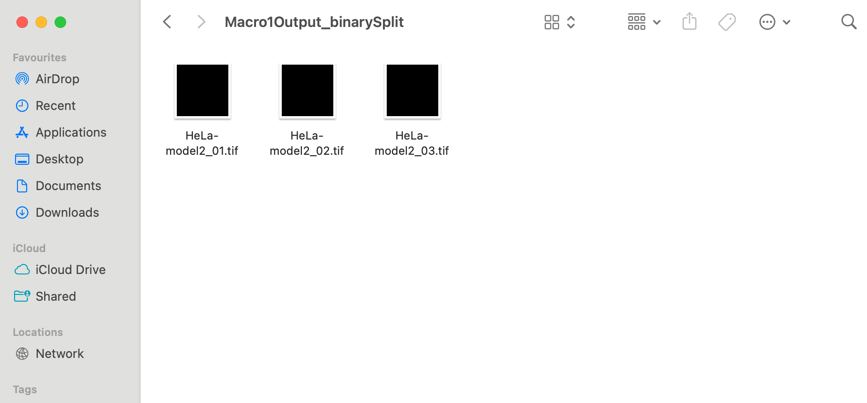
* + 1. **Enter the filename prefix:** default or anything you want



* + 1. **Select the output directory:** Define the directory for saving the processed output



* + 1. **Save the result:** The final blurred images are saved



1. Limitation
   1. **Computational Complexity**: Marco1 may struggle with more complex or highly detailed structures like neurons.
   2. **Code Optimization**: The lack of early proper structure or thorough documentation, leading to challenging in its implementation and use.
   3. **Lack of Parameter Control**: The marco1 currently lacks code for controlling parameters, making it difficult to fine-tune the process for different datasets.
   4. **No Error Messaging**: There is no implementation of error messages, which limits the ability to identify problems during execution.
2. How to improve?
   1. **Parallel Processing**: Implementing parallel processing could significantly reduce the time required to split large datasets, improving the overall efficiency of Marco1.
   2. **Parameter Auto-Tuning**: Adding smart defaults or an auto-tuning feature could make the macro more user-friendly by reducing the need for manual adjustments. Currently, we don't have any parameter tuning.
   3. **Enhanced Error Handling:** Improving error detection and feedback during the processing could help users troubleshoot issues, especially with different input file types or poorly segmented data.