

Prompt Engineering: How to Generate Good and Functional Code

With the collaboration of the  Biolmage.IO Chatbot

Caterina Fuster-Barceló

Caterina Fuster-Barceló, PhD



 Universidad Carlos III de Madrid, Spain



 cafuster@pa.uc3m.es

 cfusterbarcelo.github.io



 [@cfusterbarcelo](https://twitter.com/cfusterbarcelo)

[Find me in person in Milano in I2K!](#)



Segment Anything beyond two classes?

Image Analysis segmentation, python



3

44

4m

Preserve micromanager metadata after adjustment

Image Analysis imagej, python, micro-manager



0

2

41m

Skimage RichardsonLucy Deconvolution

Image Analysis python



22

631

14h

QuPath GeoJson question

Image Analysis qupath, python, openslide



5

27

3d

How many calculation methods are used between adjacent pixels on skeleton image analysis?

Image Analysis imagej, python



1

24

5d

Recognize and remove light grey text

Image Analysis python



4

22

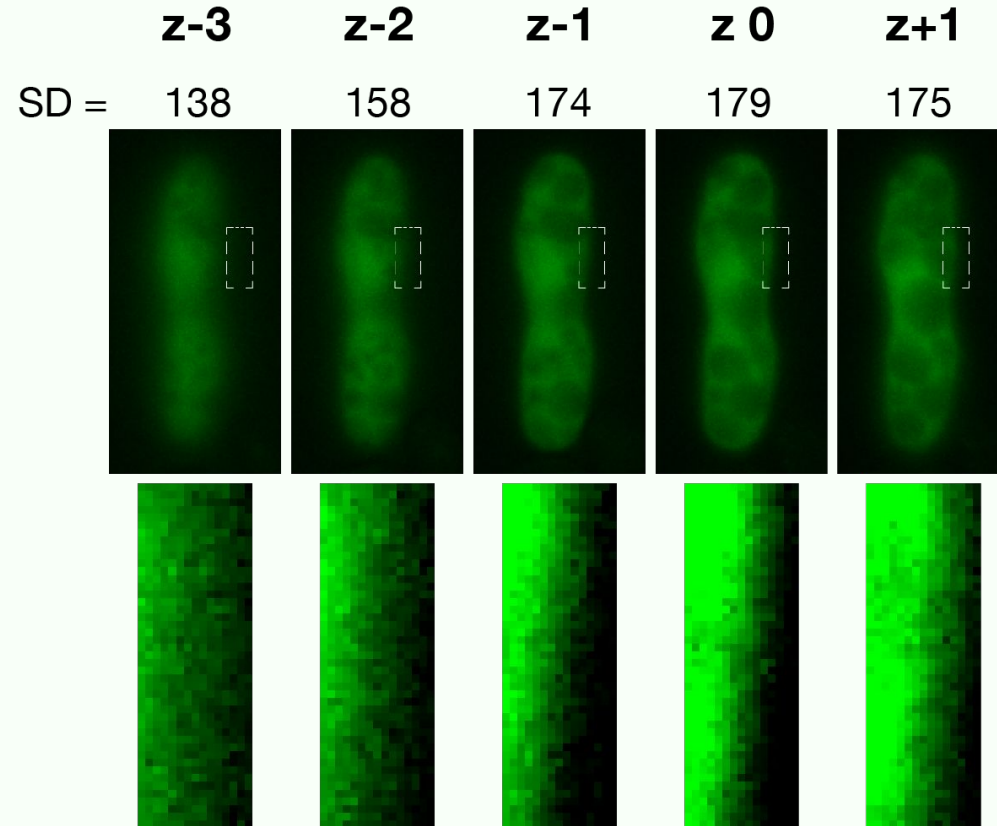
5d

Real-life situation: Find the best focused slice in a Z-stack with Python



Clàudia S. C.

Claudiasc89



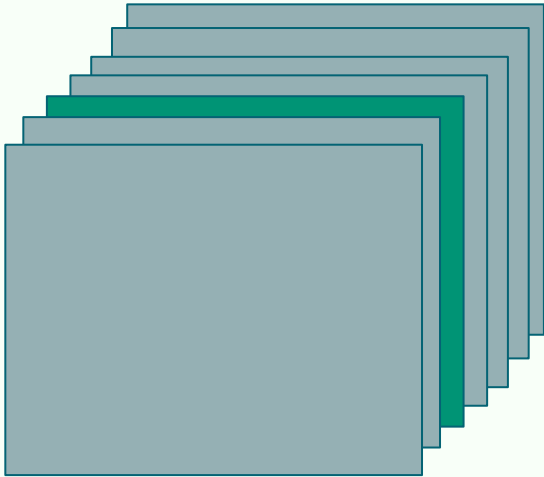
GFP Fluorescent Cells

forum.image.sc: Finding the best focused slice in Z-stack

Thanks Claudiasc89!

Chatbots and prompt engineering

The problem: Find the best focused slice in a Z-stack with Python



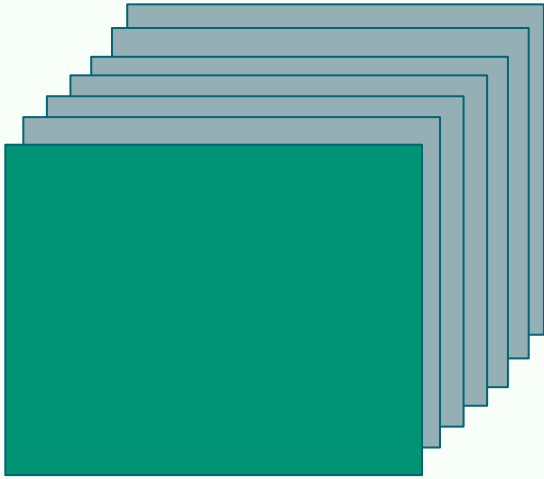
Solution 1: Calculate standard deviation (SD) of all pixels in an image. The highest the SD, the more focused.

Result: Accuracy of 60%

```
def focused_z(image, dimensions_indices):  
    axis=(dimensions_indices['x'],dimensions_indices['y'])  
    sd=image.std(axis=axis)  
    best_focused=sd.argmax()  
    return sd, best_focused
```

Chatbots and prompt engineering

The problem: Find the best focused slice in a Z-stack with Python



Solution 2: Variance of the Laplacian.

Result: always identifying first slice as the most focused.

```
def laplacian_var(image, dimensions_indices):
    axis = dimensions_indices['z']
    z_slices = dimensions_map['z']
    var_array = np.zeros(z_slices, dtype=np.float64)
    for stack in range(z_slices):
        ...
    best_focused = var_array.argmax()
    return var_array, best_focused
```

forum.image.sc: Finding the best focused slice in Z-stack

Thanks Claudiasc89!

Chatbots and prompt engineering

The problem: Find the best focused slice in a Z-stack with Python



extract best focused slice in z-stack Python

 ImageJ Wiki
<https://imagej.net> › imaging › z-... › Traducir esta página

Z-functions

Sobel Filter Based Focusing The Stack Focuser plugin uses a Sobel edge focus". Try "3" for the kernel value in the first instance. ...

 Google Sites
<https://sites.google.com> › find-focus › Traducir esta página

ImageJ plugins by Qingzong TSENG - Find Focus

This ImageJ plugin will pick up **focused slices** from a **Z stack**. Focus q evaluated by autofocus algorithm "Normalized variance".

Falta: Python | Buscar con: Python

Background & analysis goals

Hello,

I am comparing different methods to find the best focused slice among Z-stack series. My input images are fluorescent microscope images (not brightfield). So far, a simple rule of thumb has worked best for me: the simpler the method, the better the results...

> the code <

Challenges

1. Is the code okay?
2. Which method suits you best?

 I used the following code some time ago!

> the code <

I am a post-doc in image analysis with some python knowledge and i am willing to find the best focused slice in a Z-stack.

Background & analysis goals

I am comparing different methods to find the best focused slice among Z-stack series. My input...
[...]

Challenges

1. Is the code okay?
2. Which are other methods I could try?

1. Check your code:
 - a. Normalization of image values
 - b. Border effects in the Laplacian
2. Other methods:
 - a. Tenengrad Gradient Method...

forum.image.sc: Finding the best focused slice in Z-stack


Thanks Claudiasc89!

Chatbots and prompt engineering

The problem: Find the best focused slice in a Z-stack with Python



extract best focused slice in z-stack Python

 **Background & analysis goals**
Hello,
I am comparing different methods to find the best focused slice among Z stack series. My input images

I am a post-doc in image analysis with some python knowledge and i am willing to find the best focused slice in a Z-stack.

Which one is the best?



Z-fur

Sobel
focus".




ImageJ plugins by Qingzong TSENG - Find Focus

This ImageJ plugin will pick up **focused slices** from a **Z stack**. Focus q
evaluated by autofocus algorithm "Normalized variance".

Falta: Python | Buscar con: Python

2. which method suits you best?

 I used the following code some time ago!

> the code <

1. Check your code:
 - a. Normalization of image values
 - b. Border effects in the Laplacian
2. Other methods:
 - a. Tenengrad Gradient Method...

forum.image.sc: Finding the best focused slice in Z-stack

Thanks Claudiasc89!

First steps: from text to prompt



Background & analysis goals

Hello,

I am comparing different methods to find the best focused slice among Z-stack series. My input images are fluorescent microscope images (not brightfield). So far, a simple rule of thumb has worked best for me: the simpler the method, the better the results...

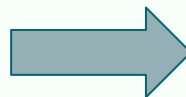
> the code <

Challenges

1. Is the code okay?
2. Which method suits you best?

I used the following code some time ago!

> the code <



I am a post-doc in image analysis with some python knowledge and i am willing to find the best focused slice in a Z-stack.

Background & analysis goals

I am comparing different methods to find the best focused slice among Z-stack series. My input... [...]

Challenges

1. Is the code okay?
2. Which are other methods I could try?

1. Check your code:
 - a. Normalization of image values
 - b. Border effects in the Laplacian
2. Other methods:
 - a. Tenengrad Gradient Method...

First steps: from text to prompt



Hello,



I am **[your position]**
specialized in **[your
background and knowledge]**
and I am currently working on
[the topic].

First steps: from text to prompt



```
> the code <
```



```
```python  
> the code <
```
```

First steps: from text to prompt



Challenges

1. Can you spot any mistake in the code of the Laplacian variance?
2. Which is the method that works best for you to detect the best focused slice?



Challenges

1. Can you spot any mistake in the code of the Laplacian variance?
2. **Could you list** me other methods to detect the best focused slice?

First steps: from text to prompt



Challenges

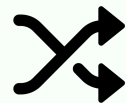
1. Can you spot any mistake in the code of the Laplacian variance?
2. Which is the method that works best for you to detect the best focused slice?



Challenges

1. Can you spot any mistake in the code of the Laplacian variance?
2. **Could you list** me other methods to detect the best focused slice?

Challenges in Using LLMs



Limited reproducibility.
Inconsistent answers to the same questions.



Knowledge cut-off.
Pending on updates.



Bias.
How was it trained?



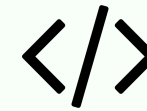
Hallucinations.
Lack of awareness.

Challenges in Using LLMs for Python Coding



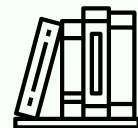
Ambiguity in prompts.

Prompts with no context or enough explanation.



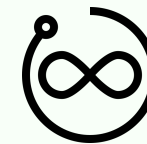
Incomplete or inaccurate code.

Generated code might not work correctly or as expected.



Difficulty handling complex libraries.

Especially for new or less-documented libraries.



Handling Edge Cases.

Might not account for null values, large datasets, unsupported data types, etc

Biolmage.IO Chatbot: A Community-Driven AI Assistant for Integrative Computational Bioimaging

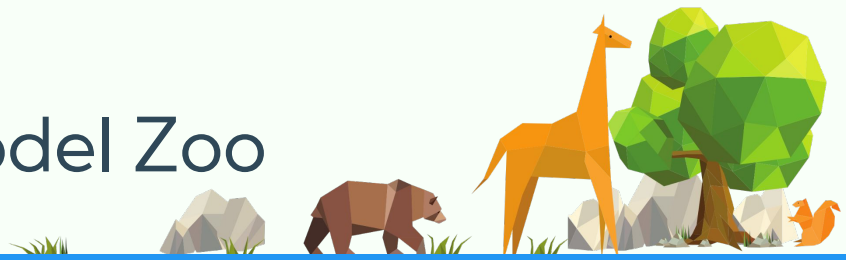
Wanlu Lei, **Caterina Fuster-Barceló**, Gabriel Reder, Arrate Muñoz-Barrutia
and Wei Ouyang



<https://doi.org/10.1038/s41592-024-02370-y>

Read it in Nat. Methods!

Biolmage.io Model Zoo



 Biolmage.IO

[+ Upload](#) [≡ Documentation](#) [ⓘ About](#) 

Biolmage Model Zoo

Advanced AI models in one-click

- Integrated with Fiji, ilastik, ImJoy and more
- Try model instantly with BioEngine
- Contribute your models via Github
- Link models to datasets and applications

 [Explore the Zoo](#)



Community Partners



[All](#) [models](#) [applications](#) [datasets](#)

Tags & Filters  

BioImage Model Zoo

- **Community-driven**
- **Open source**
- Community-based process for **contributing models**
 - Cross-compatible models among the consumer software, should always run on at least one
 - Model should be well documented
 - Model should be public and reusable under the chosen licensing conditions
- Users choose the consumer software & download
- Make **sharing** and **application** of **pre-trained neural networks** available and easy!

The screenshot shows the BioImage Model Zoo website. At the top, there's a navigation bar with 'BioImage.IO', '+ Upload', 'Documentation', and 'About'. Below this is a large blue header with the title 'BioImage Model Zoo' and the tagline 'Advanced AI models in one-click'. A sub-header lists integration options: 'Integrate with Fiji, Ilastik, ImJoy and more', 'Try model instantly with BioEngine', 'Contribute your models via Github', and 'Link models to datasets and applications'. A 'Explore the Zoo' button is present. Below the header is a 'Community Partners' section with logos for various organizations. The main content area features a search bar with the placeholder 'Type a keyword and press enter' and a 'Tags & Filters' dropdown. Below the search bar is a grid of model cards. Each card displays a thumbnail image, a title, a brief description, a list of tags (e.g., '3d', 'electron-microscopy', 'mitochondria'), and a license indicator (e.g., 'CC-BY-4.0'). The models shown include 'Mitochondria Segmentation', 'Neuron Segmentation in EM', 'Nuclei Segmentation Boundaries', 'LiveCell Segmentation Boundaries', 'Platynereis EM cells Segmentation', 'Platynereis EM nuclei Segmentation', 'HPA Cell Segmentation (DP)', and 'Cell Segmentation from Me...'. The interface is clean and modern, with a focus on user-friendly navigation and clear presentation of model information.

Community Partners



Icy

icy.bioimageanalysis.org

QuPath

qupath.github.io

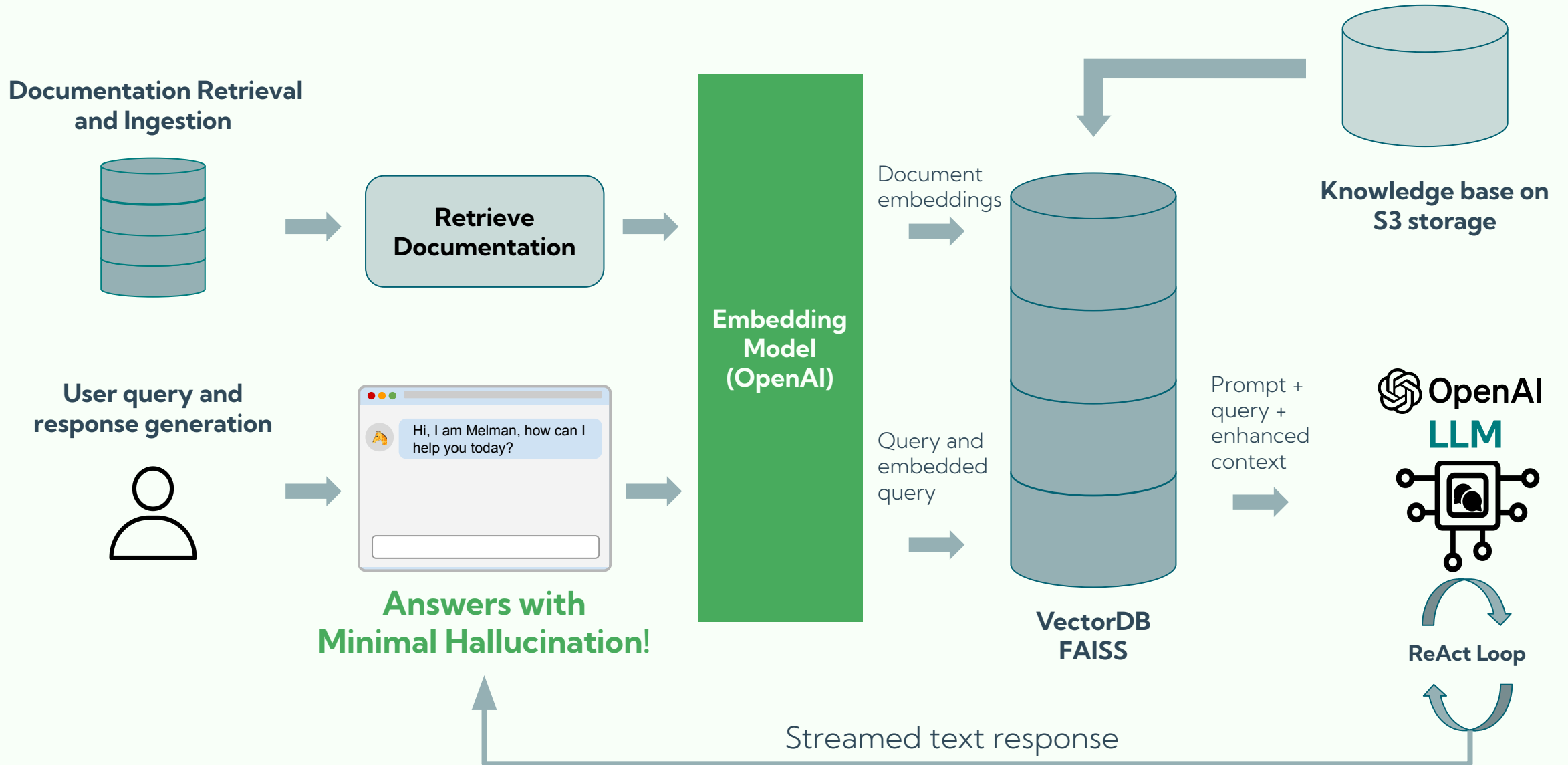
StarDist

stardist.github.io

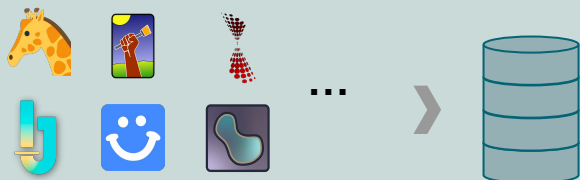


Key Feature: Retrieved Augmented Generation (RAG)

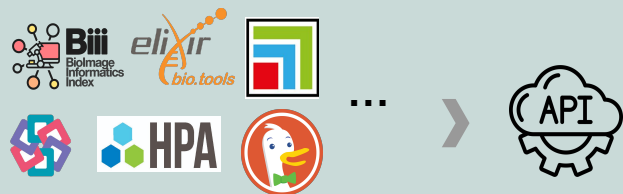
Using RAG to augment LLMs with a Community Knowledge Base



Bioimage Analysis Documentation, Databases & Tools



Documentation & Books



Online Databases & Services



AI models & Analysis tools

LLM-powered Chat Assistant

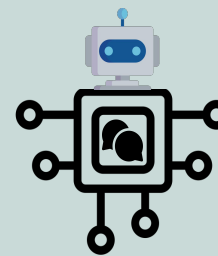
BioImage Seeker
(Melman)



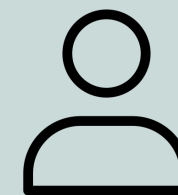
BioImage Tutor
(Bridget)



BioImage Analyst
(Nina)



ReAct Loop

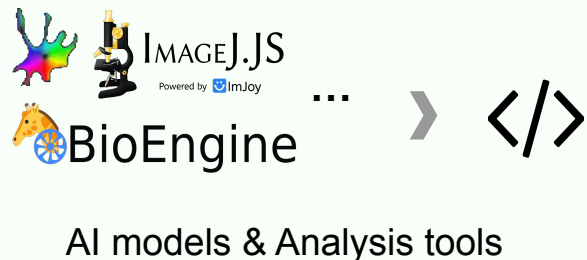
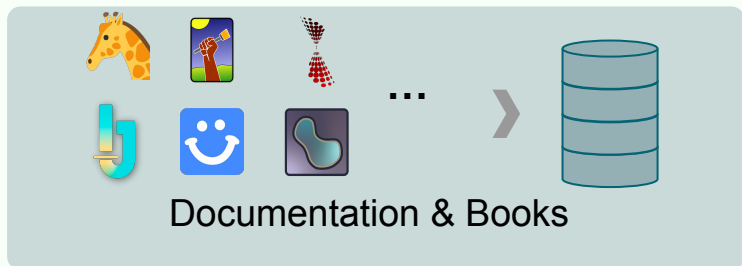


User's profile

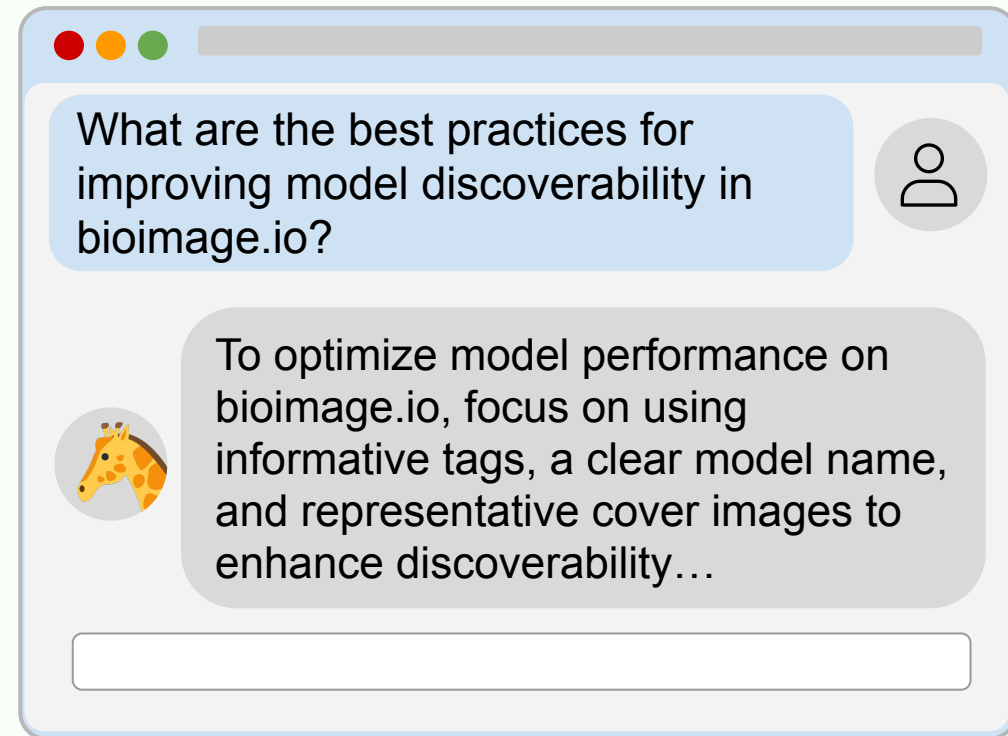
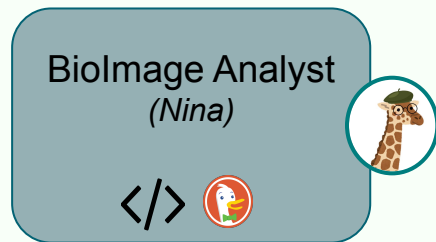
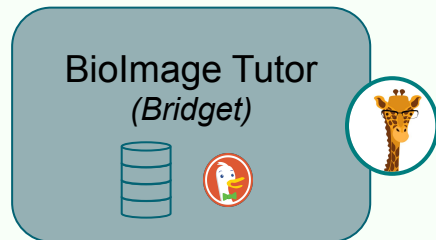
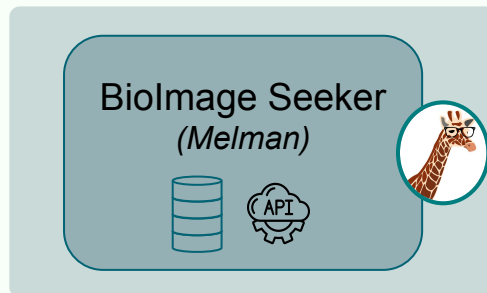
Conversation context

User's query

Bioimage Analysis Documentation, Databases & Tools

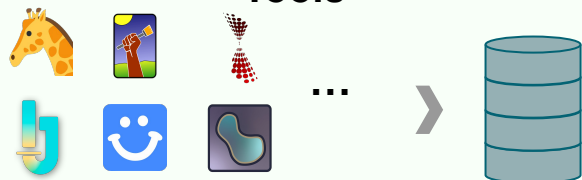


LLM-powered Chat Assistant

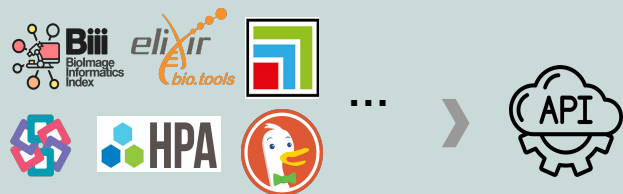


Searching in Documentation

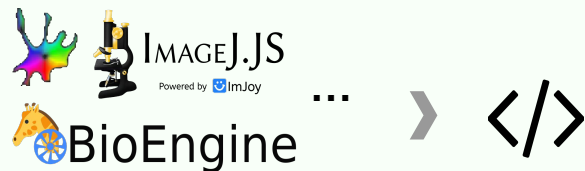
Bioimage Analysis Documentation, Databases & Tools



Documentation & Books



Online Databases & Services



AI models & Analysis tools

LLM-powered Chat Assistant

BioImage Seeker
(Melman)



BioImage Tutor
(Bridget)



BioImage Analyst
(Nina)



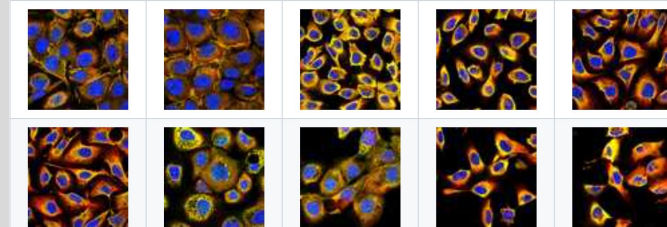
Tell me about PML protein and show me some cell images, please.



PML Protein Information:

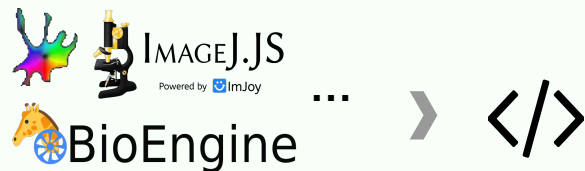
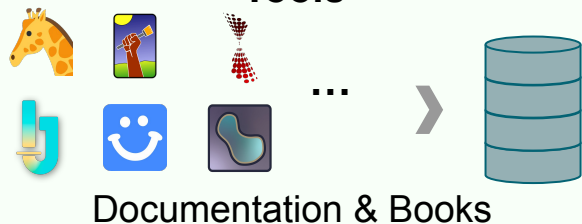
- **Gene Name:** PML
- **Gene Synonyms:** MYL, RNF71, TRIM19...

Cell Images with PML Protein Tagged in Green Channel

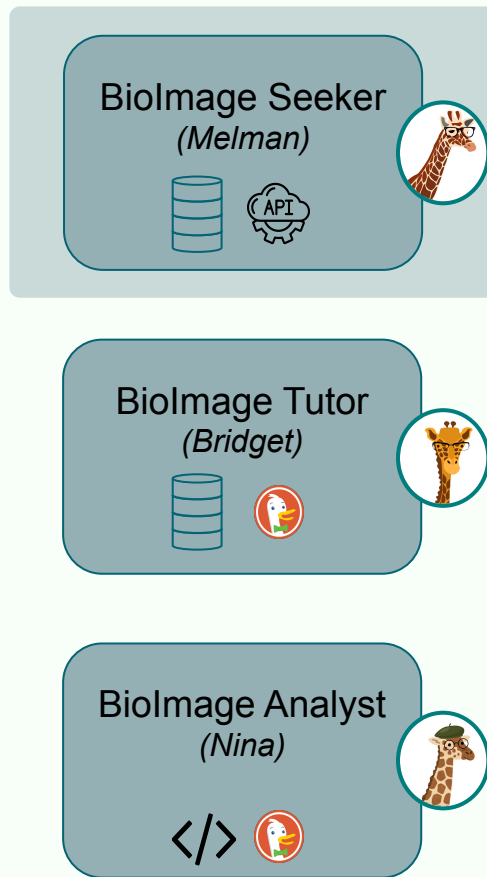


Exploring the Human Protein Atlas

Bioimage Analysis Documentation, Databases & Tools



LLM-powered Chat Assistant



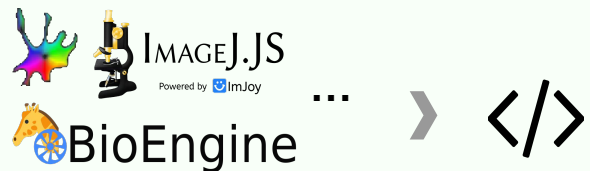
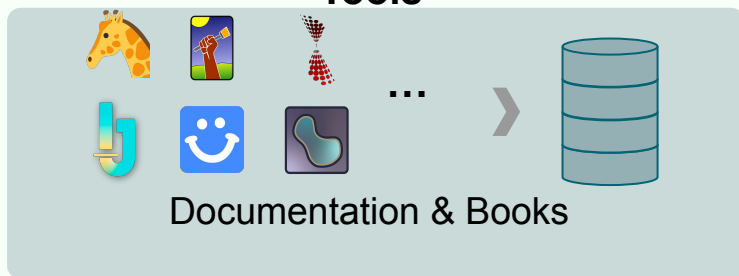
Can you find a dataset with cell images at G1 phase?

Dataset with Cell Images at G1 Phase from BiImage Archive

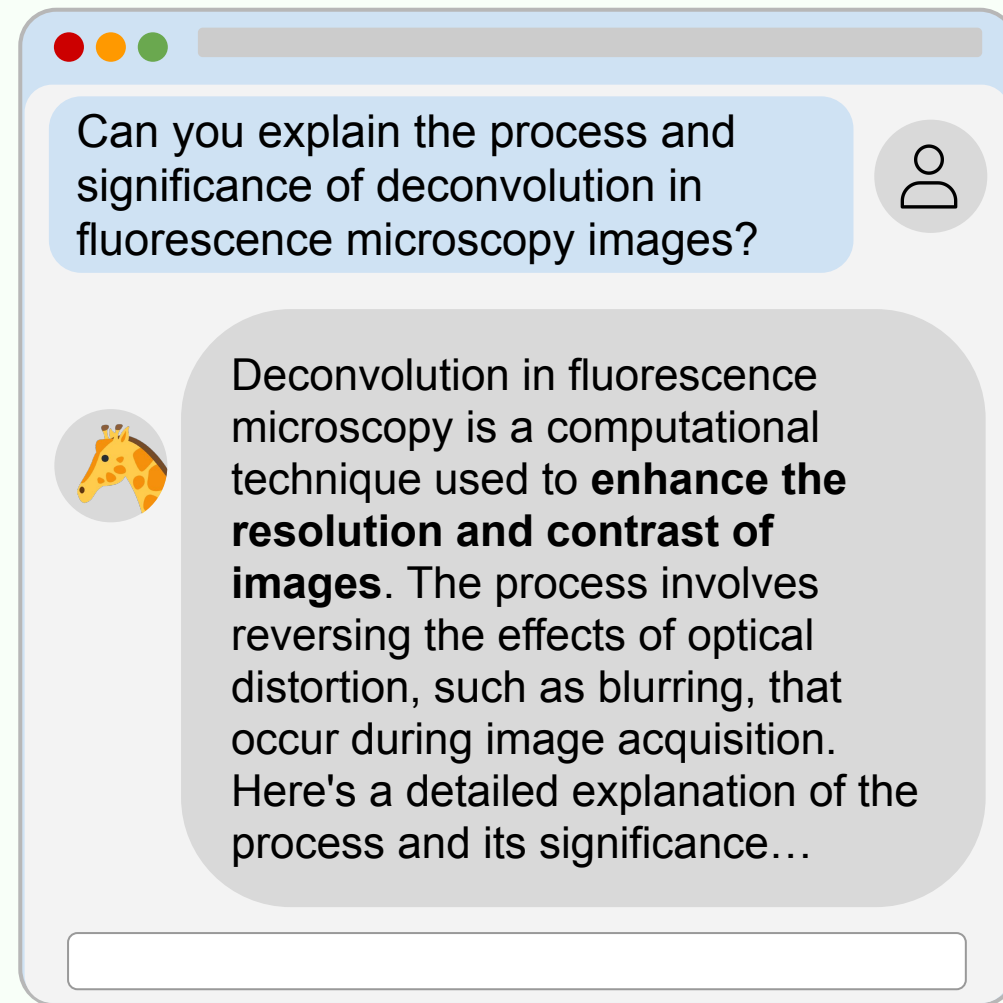
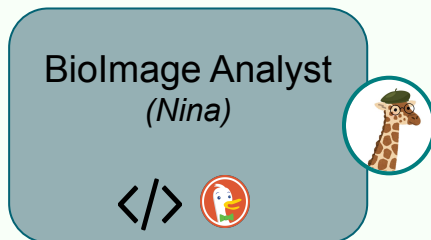
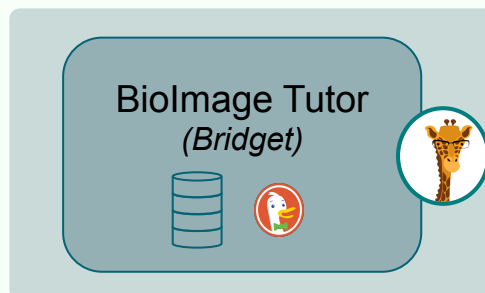
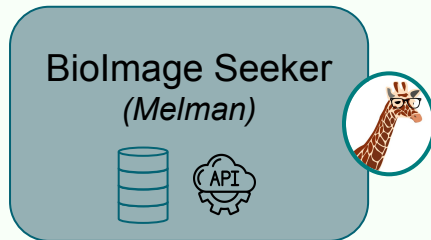
- **Title:** mRNA of the G1/S repressor WHI5 is found at similar levels throughout the cell cycle
- **Accession:** [S-BIAD491](#)
- **Description:** This dataset includes images related to the expression of WHI5 mRNA in single live cells during G1 phase, ...

Exploring the BiImage Archive

Bioimage Analysis Documentation, Databases & Tools

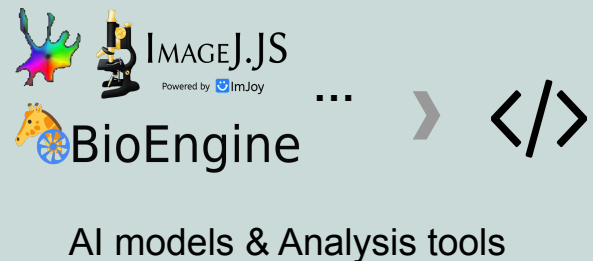
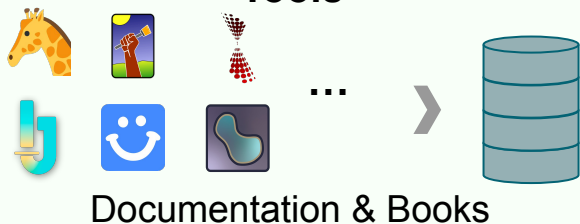


LLM-powered Chat Assistant

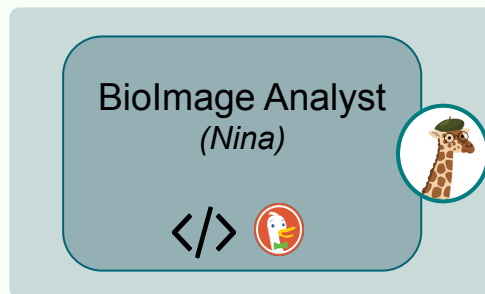
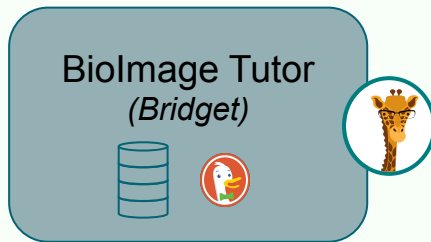
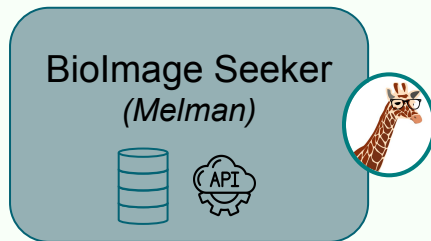


**Asking technical questions
related to BIA and AI**

Bioimage Analysis Documentation, Databases & Tools



LLM-powered Chat Assistant

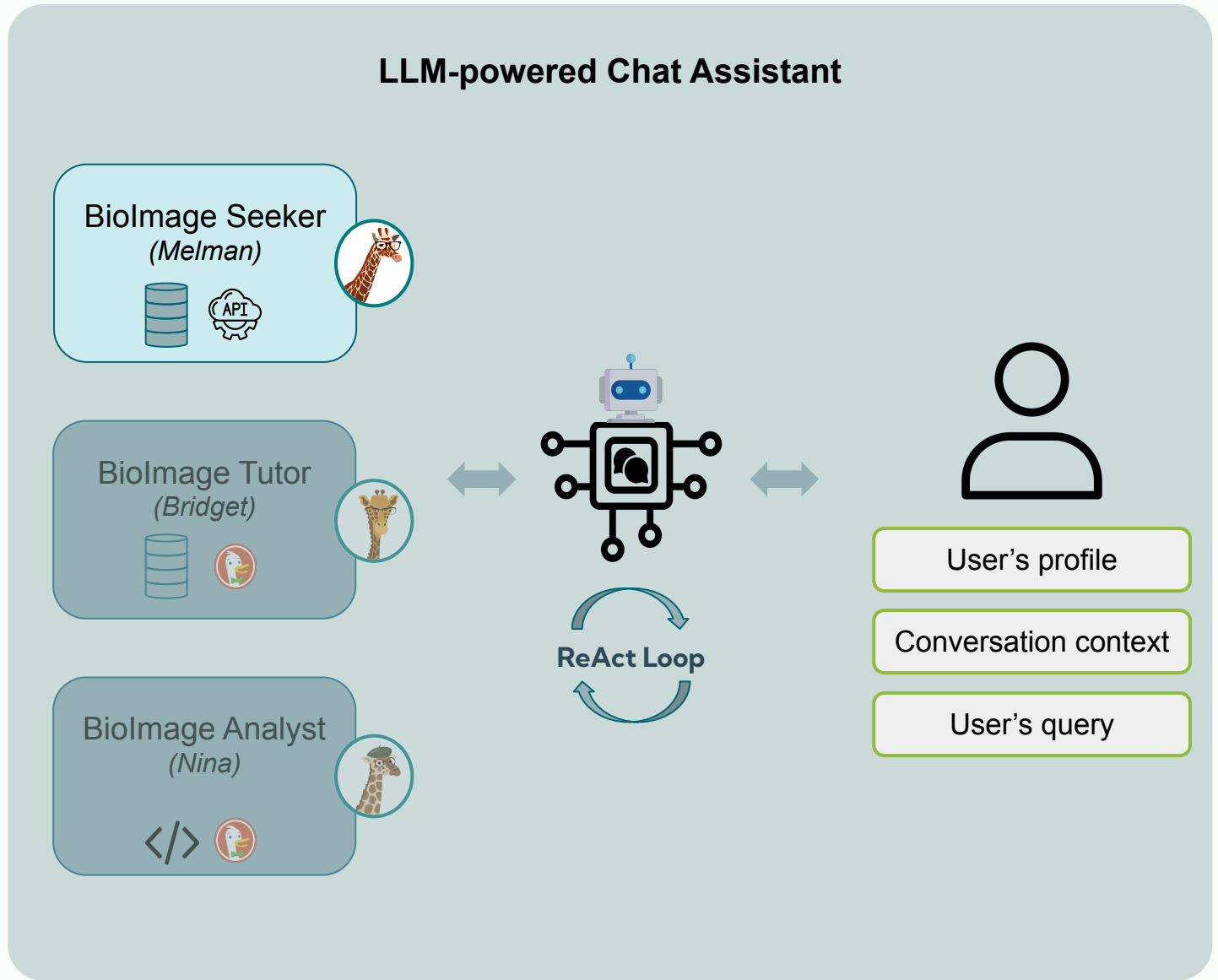


Segment the image in
`/mnt/nuclei.tif` using CellPose

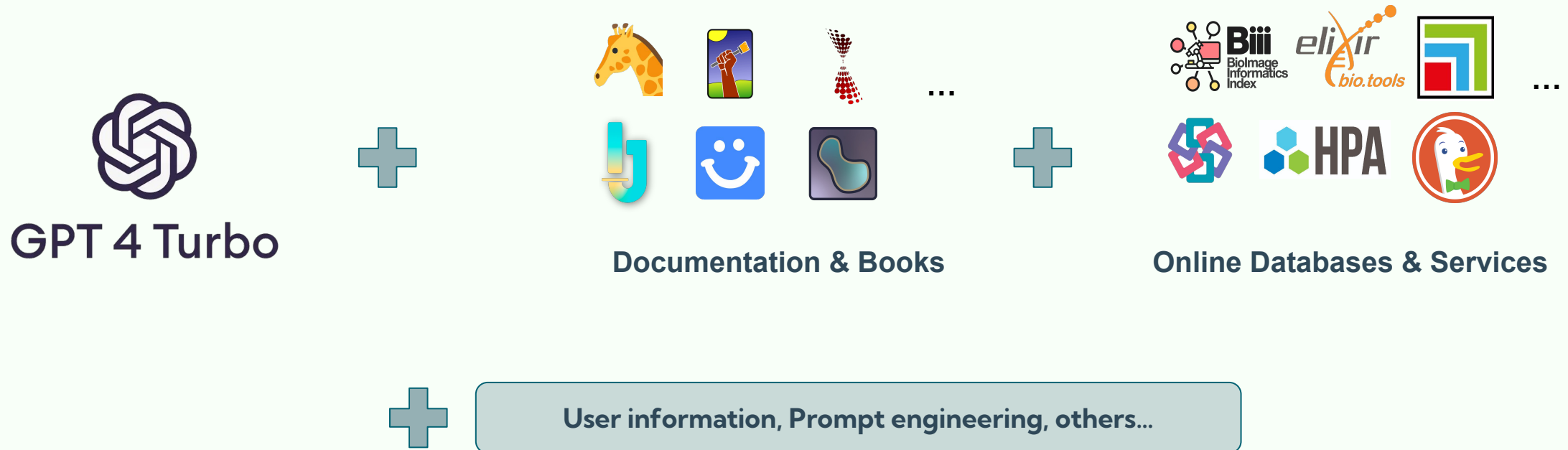
I've successfully run the CellPose model on your image using the BioEngine Service. Here's the result:

Original Image Segmented Mask

Running AI models



How does the BioImage.IO Chatbot generates code?



Chatbots and prompt engineering

The problem: Find the best focused slice in a Z-stack with Python



extract best focused slice in z-stack Python

 ImageJ Wiki
<https://imagej.net> › imaging › z-... › Traducir esta página

Z-functions

Sobel Filter Based Focusing The Stack Focuser plugin uses a Sobel edge focus". Try "3" for the kernel value in the first instance. ...

 Google Sites
<https://sites.google.com> › find-focus › Traducir esta página

ImageJ plugins by Qingzong TSENG - Find Focus

This ImageJ plugin will pick up **focused slices** from a **Z stack**. Focus q evaluated by autofocus algorithm "Normalized variance".

Falta: Python | Buscar con: Python

Background & analysis goals


Hello,

I am comparing different methods to find the best focused slice among Z-stack series. My input images are fluorescent microscope images (not brightfield). So far, a simple rule of thumb has worked best for me: the simpler the method, the better the results...

> the code <

Challenges

1. Is the code okay?
2. Which method suits you best?

 I used the following code some time ago!

> the code <

I am a post-doc in image analysis with some python knowledge and i am willing to find the best focused slice in a Z-stack.

Background & analysis goals

I am comparing different methods to find the best focused slice among Z-stack series. My input...
[...]

Challenges

1. Is the code okay?
2. Which are other methods I could try?

1. Check your code:
 - a. Normalization of image values
 - b. Border effects in the Laplacian
2. Other methods:
 - a. Tenengrad Gradient Method...

forum.image.sc: Finding the best focused slice in Z-stack

Thanks Claudiasc89!



Hi there! I'm Melman. I am help you navigate the bioimaging tools and provide information about bioimaging. How can I help you today?



I am a post-doc in image analysis with some python knowledge and i am willing to find the best focused slice in a Z-stack Background & analysis goals I am comparing different methods to find the best focused slice among Z-stack series. My input images are fluorescent microscope images (not brightfield). So far, a simple rule of thumb has worked best for me: the simpler the method, the better the results. The method that has given me the best outcomes is calculating the standard deviation (SD) of all the pixels in each image and selecting the image with the highest SD as the most focused one. Here's the code for that approach: `python def focused_z(image, dimensions_indices): axis=(dimensions_indices['x'],dimensions_indices['y']) sd=image.std(axis=axis) #by axis we are telling it to flatten the first and second dimensions, so Y and X and get the sd of all the values best_focused=sd.argmax() return sd, best_focused`

However, the accuracy of this method is still realtively low (below 60%) so I am looking for a more suitable approach. I've been experimenting with the variance of the Laplacian, but it consistently identifies the first slice as the best-focused one. Below is the code I'm using for this method: `python def laplacian_var (image, dimensions_indices): # Get the axis that contains the z parameter axis = dimensions_indices['z'] # Get the number of z-stacks z_slices = dimensions_map['z'] # Initialize a NumPy array to store variance results var_array = np.zeros(z_slices, dtype=np.float64) for stack in range(z_slices): # Take a 2D image img_z = image.take(stack, axis= axis) # Apply Laplacian operator in the required depth laplacian = cv2.Laplacian(img_z, cv2.CV_64F) # Calculate the variance of the Laplacian variance = laplacian.var() # Store the result in the NumPy array var_array[stack] = variance # The bet focused slice will be the one with the maximum variance value best_focused = var_array.argmax() return var_array, best_focused`

So, my questions here are the follwoing: 1. is there any mistake in the code of the laplacian variance? 2. which are other methods i could try to detect the best focused slice?



Analysis of the Laplacian Variance Code

1. **Code Review:** Your code for calculating the variance of the Laplacian seems logically correct. However, ensure that the `dimensions_map` variable is correctly defined and matches the structure of your image data. Also, verify that the `cv2.Laplacian` function is applied correctly to the 2D slices.
2. **Common Issues:** If the method consistently identifies the first slice as the best-focused one, consider normalizing the images or adjusting the Laplacian kernel size. This might help in better distinguishing the focused slice.

Alternative Methods for Focus Detection

1. **Normalized Variance:** A method shared on the Image.sc Forum involves calculating the normalized variance of each slice. This method is based on the algorithm described by Sun et al., 2004, and might provide better results for your fluorescent images.
2. **Mean of Absolute Laplacian:** Another approach is to use a combination of variance and the mean of the absolute Laplacian, as suggested by a forum user. This method might offer a more balanced focus metric.
3. **ImageJ Plugins:** Consider using ImageJ plugins like "Find Focused Slices" by Qingzong Tseng, which are specifically designed for fluorescent images. These plugins might offer a more robust solution for your needs.
4. **EDF Algorithm:** The Extended Depth of Field (EDF) algorithm from EPFL is another option. It has been used successfully in various applications and might be worth exploring.
5. **Laplacian of Gaussian (LoG):** According to the scikit-image documentation, the Laplacian of Gaussian method is accurate for detecting focused areas, though it might be slower. This could be a potential method to try if speed is not a primary concern.

For further details and code examples, you can refer to the [Image.sc Forum discussion](#) and the [scikit-image documentation](#). These resources provide additional insights and community-shared solutions that might be beneficial for your analysis.

▶  More Details

Finding the best focused slice in a Z-stack

■ Image Analysis ● python



Claudiasc89

1  3d

Background & analysis goals

Hello,

I am comparing different methods to find the best focused slice among Z-stack series. My input images are fluorescent microscope images (not brightfield).

So far, a simple rule of thumb has worked best for me: **the simpler the method, the better the results**. The method that has given me the best outcomes is calculating the standard deviation (SD) of all the pixels in each image and selecting the image with the highest SD as the most focused one.

Here's the code for that approach:

```
def focused_z(image, dimensions_indices):  
    axis=(dimensions_indices['x'],dimensions_indices['y'])  
    sd=image.std(axis=axis) #by axis we are telling it to flatten the first ar  
    best_focused=sd.argmax()  
    return sd, best_focused
```

Bonus track: Macro Programming

Certainly! Here's a simple **macro script for Fiji (ImageJ)** that calculates the standard deviation of each slice in a Z-stack to find the most focused slice:

```
open("path/to/your/z-stack.tif");
nSlices = nSlices();
maxSD = 0;
bestSlice = 1;
for (i = 1; i <= nSlices; i++) {
    // Set the current slice
    setSlice(i);
    // Get the standard deviation of the current slice
    run("Measure");
    sd = getResult("StdDev");
    // Check if this slice has the highest standard deviation
    if (sd > maxSD) {
        maxSD = sd;
        bestSlice = i;
    }
}
// Set the stack to the best-focused slice
setSlice(bestSlice);
// Optionally, save the best-focused slice
saveAs("Tiff", "path/to/save/best_focused_slice.tif");
```

Bonus track: Finding more

Here are some plugins and methods you can use in Fiji (ImageJ) to find the most focused slice in a Z-stack:

1. **Autofocus Plugin:** This plugin uses the "Normalized Variance" algorithm to select the in-focus frame from each slice of a hyperstack. It creates a new stack of just the in-focus frames. You can find more about it on the [BioImage Informatics Index](#).
2. **Find Focused Slices Plugin:** An ImageJ plugin specifically designed for selecting the most focused plane among multiple slices in an image stack. It uses a specific algorithm to determine focus. More details can be found [here](#).
3. **Extended Depth of Focus Plugin:** This plugin creates an entirely focused composite image from a Z-stack. It might be useful if you want to combine the best-focused parts of each slice. More information is available on the [BioImage Informatics Index](#).

Additionally, discussions on the [Image.sc Forum](#) suggest using the TFM package for automated best-focused slice detection, which has been successfully used in various applications.

These plugins and methods should help you in finding the most focused slice in your Z-stack images without needing to write a macro from scratch.

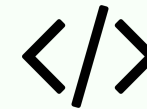
 [More Details](#)

Good Practices in Using LLMs for Python Coding



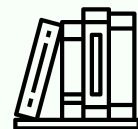
Be precise in prompts.

Give as much context and explanations as possible.



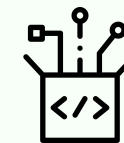
Test and debug.

Use it to modify your code or as a starting point, not base your own code on it.



If you know, you know.

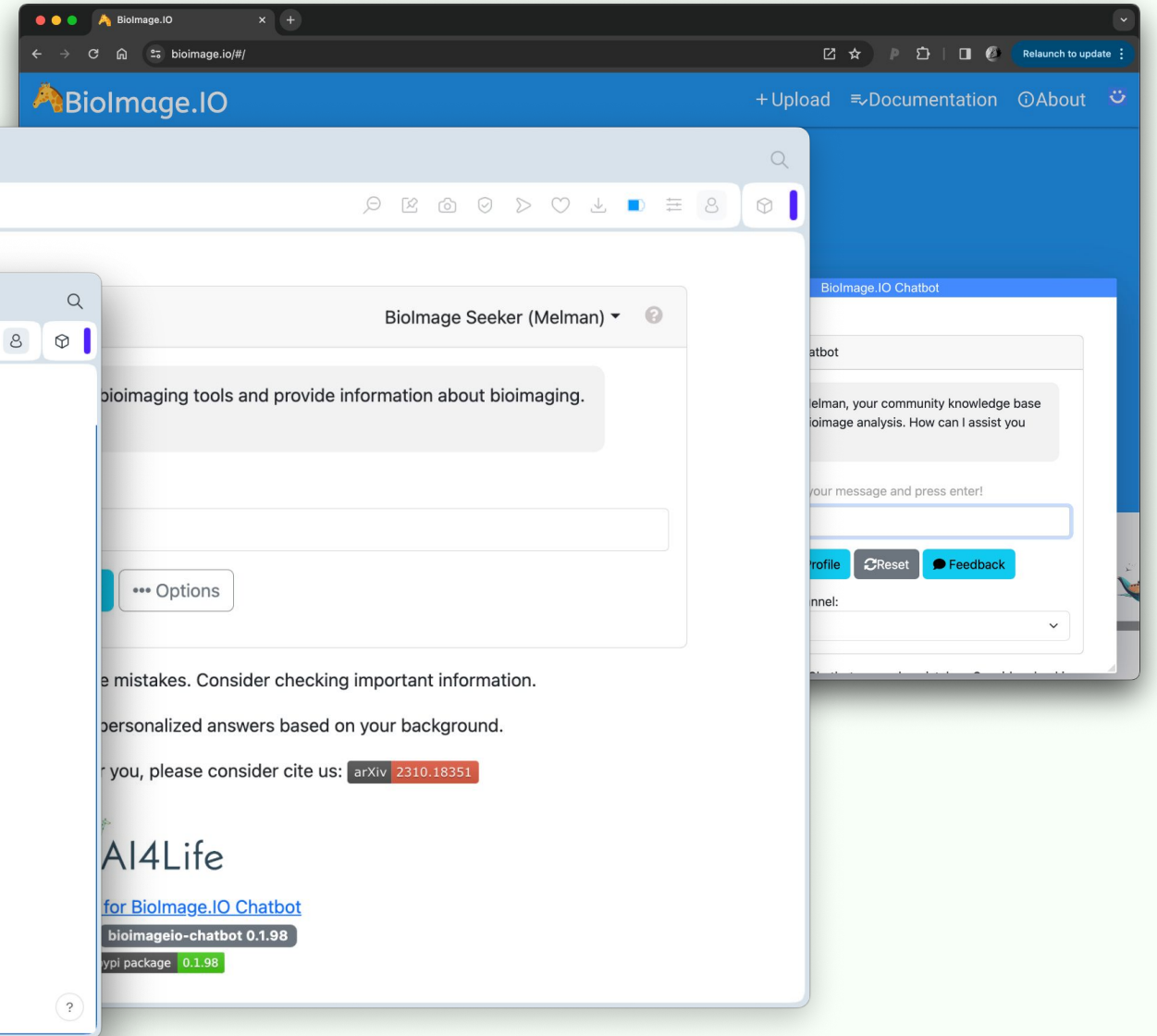
If you know what libraries you want to use or which pipeline to follow, explain it.



Use it for what it's worth.

Efficiency, commenting style, finding different solutions, etc

How can you access it?



How can you contribute?



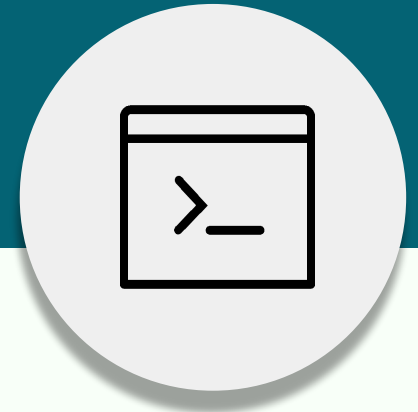
Developing Extensions



**Adding Tool
Documentation**



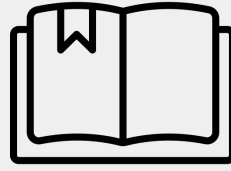
**Feedback and User
Experience**



**Sending PR to our GitHub
project**

[GitHub Repo](#)

What have **we** learned from the Biolmage.IO Chatbot?



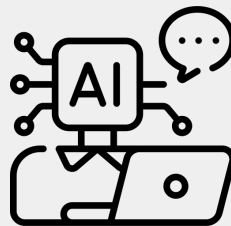
Improve your documentation!

Your documentation should be not only human-readable but **AI-processable**! The more explained and **self-contained** the better!



Work on prompt engineering...

It is game-changing! If you are more **precise** with your prompts and **context**, responses will be much better!



Do we need documentation for the Chatbot itself?

There is some documentation we hope you don't need to read. Just ask the chatbot!

Try it yourself!
<https://bioimage.io/chat/>

Acknowledgments

Wanlu Lei

Gabriel Reder

Arrate Muñoz-Barrutia

Wei Ouyang

AI4Life





Caterina Fuster-Barceló

cafuster@pa.uc3m.es