Complex Image Analysis Tasks: What, How, and Who

Beth A Cimini[1]*

[1]=Broad Institute of MIT and Harvard, Cambridge, MA, USA
* = To whom correspondence should be addressed; Contact details:

Dr. Beth Cimini,
Imaging Platform,
Broad Institute,
415 Main St,
Cambridge, MA 02142
Email: bcimini@broadinstitute.org
Phone: 617-714-7000

Abstract

As microscopy diversifies and becomes more complex, the problem of quantification of microscopy images has emerged as a major roadblock for many researchers. There are challenges that all researchers face, independent of their scientific question and the images they've generated. Complexities may arise at many stages throughout the analysis process, including handling of the source data, image pre-processing, object finding, or measurement, and statistical analysis. While the exact solution required for each complexity will be problem-specific, by understanding tools and tradeoffs, optimizing data quality, breaking workflows and data sets into chunks, talking to experts, and thoroughly documenting what has been done, analysts at any experience level can learn to overcome these challenges and create better and easier image analyses.

Introduction

There are few constants across microscopy's long and varied history, except perhaps for the original goal: to make sense of the world that is smaller than what our eyes can perceive. Early microscopy began as descriptions of the natural world, followed by hypothesis generation; it is therefore no surprise that even after the invention of the digital camera, microscopy's history has often rested on "representative image shown". As we now enter a more quantitative scientific era, microscopy must face the challenge of *image analysis*: turning the astonishing variety of things people do with microscopes into numerical data. Since most things researchers are studying are complex, most image analysis workflows are complex. Here, we review several sources of complexity, common themes in approaching difficult tasks, and discuss community efforts available to help image analysis learners; while many examples here are pulled from the

field of light microscopy of biological samples, general principles apply across disciplines and microscope types.

Common sources of image analysis complexity

We review here several common sources of image analysis complexity. Not all experiments will have all sources, but many experiments will have many of these sources, each of which must be addressed for analyses to be properly interpreted. These are summarized graphically in Figure 1. We do not deeply discuss here complexities in sample preparation and/or image acquisition, but complexities exist in those steps as well; consideration of those steps is critical in a fully quantitative imaging experiment[1], and the best solution for some of the complexities discussed below may be to return to those early stages and create different images.

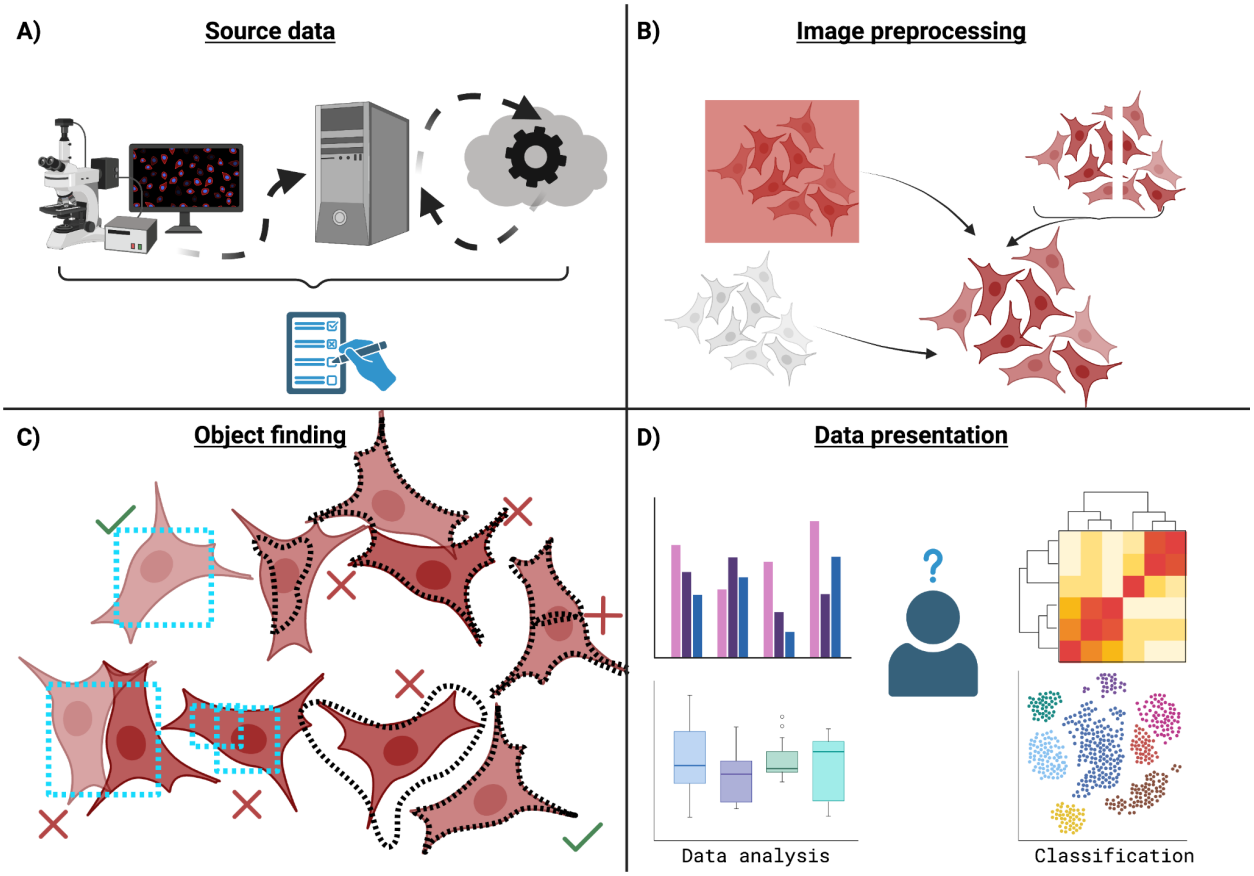

Figure 1: Common sources of image analysis complexity. A: to analyze the source data, analysts may need to navigate proprietary file format export, issues with data size, location where the data must be processed, as well as associating and tracking metadata for future reporting. B: images may need to be pre-processed before analysis, such as by stitching, denoising/background removal, or pixel classification or "virtual staining". C: the analyst must determine if they wish to perform object detection (left side, blue boxes) or segmentation (right side, black outlines). In either case, objects may be erroneously merged or split, and segmented

objects may not be identified with accurate boundaries. D: the analyst must determine the appropriate ways to normalize, group, and present the data (left). If performing classification, the analyst must decide between supervised and discrete classifications (top right) versus more continuous and/or unsupervised groupings (bottom right).

Source data

The first major source of complexity a user must usually work around are the source data files themselves. Many microscopes *export* data in proprietary formats that make it difficult to open the files outside of specialist programs; care must also be taken that the export settings do not alter or destroy image data (such as by clipping 16 bit data to 8 bit, or exporting four fluorescent channels as a single RGB image). Once exported, one must deal with the size and structure of the data - how many files are present, and in what configuration? Some microscopes create large single files; others (such as some formats associated with slide-scanning microscopes) create many files per scan, which must be kept in a certain relationship to one another or file reading will break. The user must determine how much data has been generated, where it will be immediately stored, if sufficient computational power exists on the immediate storage machine to perform required post-processing and analysis, and what (if any) plans exist for long-term data storage.

A key and underappreciated aspect of source data handling is the handling of associated *metadata* - how was the sample generated, how was it imaged, and with what experimental question in mind. The answers must be carefully tracked, and ideally permanently associated as closely as possible to the image data so that in the future, it is easy to determine what any image depicts and how it was made. This facilitates not only maximally correct analysis at the time, but eventual data reuse [2].

Image preprocessing

Once the user has their data, it will often (though not always) require some amount of preprocessing before eventual analysis. If the microscopy method used starts with multiple individual images (such as in single-molecule localization microscopy (SMLM), slide-scanning microscopy, multi-objective image capture methods, or highly-multiplexed imaging methods), customized integration may need to be done to create one "logical image" per sample. Since fluorescence microscopists typically try to minimize the light on their sample to reduce photodamage and bleaching, *deconvolution* and/or denoising may be needed to enhance features of interest, especially in thick samples. The user may also lack a stain for particular regions of interest in the image, or decide the existing stain is insufficiently specific; in these cases, *semantic segmentation* (often called "pixel classification") tools may be called on to create machine learning algorithms that will allow the user to "virtually stain" regions of interest that they wish to measure later.

Object finding

Many (though not all) microscopy analyses rely on identifying objects within the image. Computer scientists differentiate between *object detection* (finding how many objects are present, typically with a centroid and perhaps a bounding box of where the object can be found) and *instance segmentation* (often simply referred to as "segmentation") where it is important to find the exact boundary of the object. Users first must decide which task to perform: tasks which rely primarily on counting and classification ("how many cells here are infected, and how many uninfected?") are suited to object detection; cases where the user wants to know properties of the specific objects ("how big are infected cells and uninfected cells?") require segmentation. Segmentation is typically considered to be a much more difficult problem, since it requires much more precision, but it is far more commonly used since it can ultimately provide more information.

Segmentation can typically be performed either using standard computer vision techniques or deep learning. Classical methods typically require that the objects of interest are bright and everything else in the image is dark, which if not already the case requires image pre-processing steps (see above). If this pre-processing is onerous, the user can consider using deep learning techniques for either object detection or segmentation, but in the absence of an existing pre-trained model for the user's task, training such a network can require substantial data and computational expertise.

Measurement, classification, and interpretation

Once all image transformations are complete and any objects of interest have been found, the user is finally ready to use their images to answer questions. Major sources of complexity at this step begin with simply deciding on the exact metrics to use - does one want to know the total amount of stain in an image or object? The mean amount? How the distribution of the stain has changed? Determining what each metric precisely means, and which is the best match to the scientific question of interest, can take significant knowledge and/or expertise. Statistical treatment of the data also requires a careful approach - is the appropriate unit of comparison an object, an image, or a sample? Does the data need to be normalized for cross-batch comparison, and if so, what constitutes a batch and how will the normalization be performed?

If performing classification analyses, the user must consider if their phenomenon is reasonably discrete (each class is distinct, with few examples of intermediate phenotypes) or continuous (smooth progression between states). Different classification methods or techniques may be more appropriate to each class of problem.

Common themes for approaching complex tasks

While deep-dive examinations of how to approach all of the possible sources of complexity described above are beyond the scope of this work, several general principles about *how to solve for any given complexity* can be derived.

Understand the tools and their limitations

Ultimately, all image analyses (and indeed, all experiments) one performs in the lab are models - after performing some set of experiments on some finite number of samples, we are attempting to create a quantitative picture of what the larger world looks like. Ultimately, as in all models, there will be limitations and inaccuracies - as George Box said, "All models are wrong, but some models are useful". Performing an image analysis is therefore not a matter of performing some perfect series of steps, but rather in creating a model that is most correct/least wrong. Error tolerance is a practical part of any image analysis, with the level of tolerance linked to the expected size of the quantitative change: for example, a 10% error tolerance is perfectly acceptable when looking at a 10-fold change but not when quantifying a 20% change.

As in sample preparation and imaging, image analysis ultimately involves balancing a series of tradeoffs, including tradeoffs between aspects of sample preparation and analysis. In sample preparation and microscopy, it is common to balance the cost, work, and fidelity of individual steps of sample preparation and imaging, or even between kinds of imaging to use. When selecting an analysis tool or approach, a similar set of tradeoffs must be considered: does this approach do what I need? How easy is each tool in this approach to install? How easy is it to use? How easily can the analysis scale from an initial prototype to many images? How easy is it to inspect each step to make sure it is done correctly? How easily can I document what was done for the future so it can be repeated? Depending on the user's needs, comfort level with individual tools and with scripting/coding in general, there are nearly always many correct approaches to any problem.

It is also important to understand how the tools and the settings within them may affect the final data. As an example, if the researcher's object of interest is usually around 20 pixels in diameter and thus the analyst sets a hard cutoff during segmentation that only objects between 10 and 30 pixels in diameter are "real" and should be accepted, perturbations or conditions that cause very large (>30 pixel diameter) or very small (<10 pixel diameter) objects may not be detected, because these objects will be thrown out due to the cutoff. Such a cutoff may still be good and least-wrong if it throws out relatively few real objects relative to many pieces of debris that would otherwise have thrown off the quantification (see Figure 2), but it means the results should not be interpreted to mean that Perturbation X does not create 35-pixel-diameter objects.

When adopting any new tool, but most especially tools using deep learning, it is important to understand the data types a tool expects (such as fluorescence vs brightfield vs EM, individual object crops vs individual slices vs whole volumes, etc.) as well as conditions under which it does and does not work well[3]. This can be more challenging in deep learning because the user is not typically manually setting cutoffs as they might be in a more conventional analysis tool, and networks may be overfit to certain data types in ways that produce unexpected results. While deep learning undoubtedly solves many problems in microscopy that conventional approaches have not [4], it must be used with especial caution. Whether one should try a deep-learning-based tool vs a conventional tool is not a simple answer for most tasks, and will be based on ease of use, performance metrics, and how many conventional-tool-steps a deep learning tool might replace.

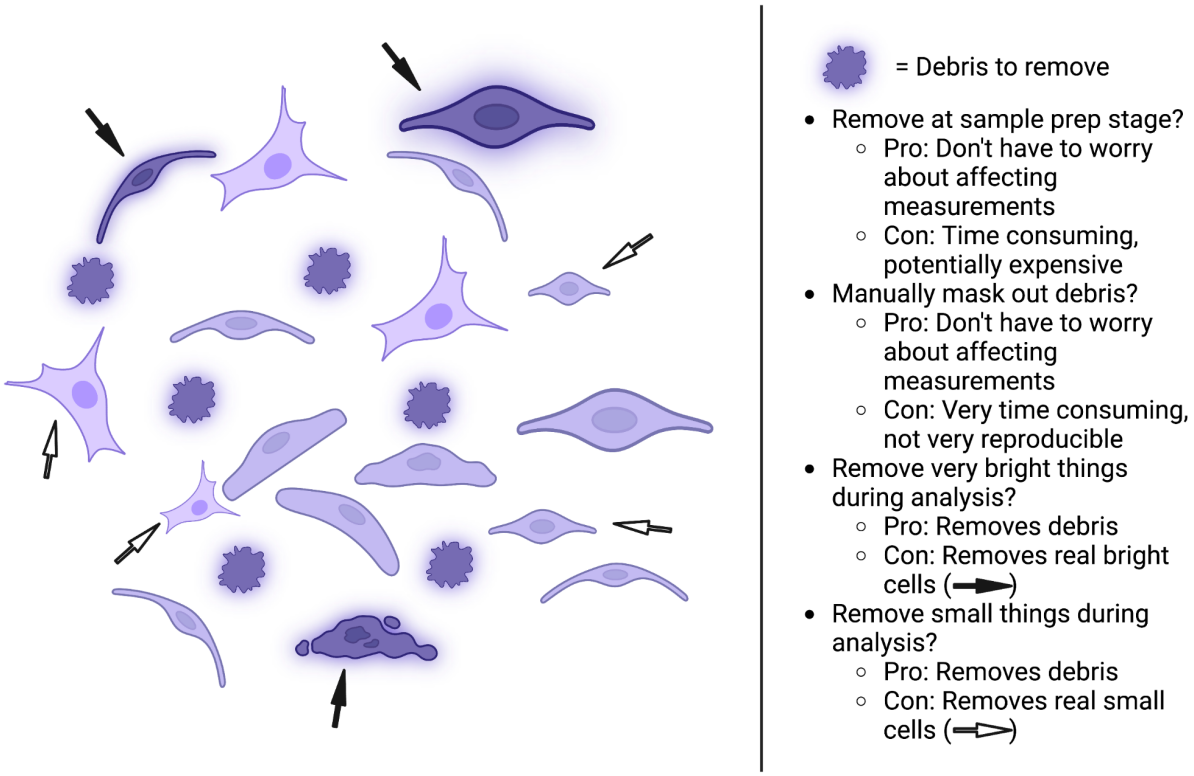

Figure 2: Example of real-world considerations when removing small, bright debris particles from image analysis. A few of many possible options are presented; each has both advantages and disadvantages. Color intensity represents brightness of the debris/cell object. Arrows mark cells that would be inadvertently removed from analysis by a debris-removal method due to brightness (filled arrows) or size (unfilled arrows).

Optimize data quality

It may sound obvious, but a general critical factor in image analysis and computing in general is "garbage-in, garbage-out": to generate high-quality analyses, one needs to have sufficient-quality images. Often, visual assessment is sufficient for this - is the image in focus, not saturated, and reasonably clear of debris? (Figure 2) Are any objects the researcher wishes to computationally identify visible, and are their boundaries defined enough that the researcher can assess if segmentation is proceeding accurately? As discussed above, perfect images are typically impossible to generate (and ultimately not required), but the images should be of good enough quality that it does not push the analysis workflow outside the error tolerances of the problem. The eventual problem and its tolerance will guide what good-enough means: in a hypothetical experiment with a nuclear marker, cell boundary marker, and marker for some other biology (Marker X) where the goal is to assess how much of Marker X is in the nucleus, a dim/blurry cell boundary marker is likely tolerable, but if the goal is to measure the amount of Marker X present at the cell membrane, it likely is not.

Work in "right-size", representative chunks

Essentially all complex image analysis problems are multi-stage workflows: many steps will ultimately take place between the microscope hard drive and a final answer [5]. Especially as data sets get larger and/or the user accumulates more of them, knowing how best to handle this complexity becomes more and more essential to a high-quality final product and the least-painful experience for the analyst.

First, the workflow should be optimized in pieces, with data quality (see also above) assessed at every stage: in an example workflow that involves alignment, followed by denoising, followed by segmentation, followed by measurement, the aligned images, the denoised images, and the segmentations should all be checked for errors introduced at each step along the way. This optimization needs to happen sequentially in the order that the data will travel through the workflow, as changes in later steps will need to be re-assessed anytime an earlier step changes. While it may be tempting to "roll the dice" and only examine the final product to bypass the time spent in quality checks, adding quality control steps along the workflow is ultimately far faster in the long run for the vast majority of cases since it becomes easier to trace the sources of the errors and solve them piecewise.

Unless one's data is extremely small, this prototyping process is made much easier by only working with a subset of the total data set during the prototyping phase. For large images, this may mean a few crops; for sets that consist of many smaller images, a few well-chosen images. It is extremely important that such subset not just consist of the first few images or the prettiest areas, but the full range of phenotypes present in the whole data set, or else the analysis workflow will be *overfit* to the kind of data in the subset; this is an especially critical factor with deep learning workflows [3]. As an example, when working in multiwell plates where each well is differently perturbed, pulling one image from each well is often sufficient to ensure one's workflow is robust to the whole experiment before running the final optimized workflow on the full data set.

Bring together the right experts

It is entirely understandable if a new image analyst has become overwhelmed by this point in the process: it seems like there is far too much for anyone to learn alongside all of the domain-specific and technique-specific knowledge they need to keep up with. This is ultimately true, as image analysis becomes accepted as a discipline in its own right. Understanding these complexities, the open-source image analysis community has created a number of resources in order to help users get started and/or improve their analyses, including tool lists [6] and best practices guides [1,5,7,8] . In 2018, the Scientific Community Image Forum (forum.image.sc) was launched to create a single central place for users to ask questions related to image analysis [9], and as of mid-2023, serves as a central help forum for 56 individual open-source image analysis tools and contains tens of thousands of posts that are free for users to search for answers. Image analysis also has become an increasingly common option within imaging facilities, and a few stand-alone image analysis facilities now exist [10].

While image analysis experts are critical, image analysis expertise is not the only knowledge needed: expertise in what the samples are, how they were created, and how they were imaged

is critical to determining what can and what cannot be learned from any given image. This information may be provided by the researcher, the image analyst, and/or by a microscopy specialist involved in creation of the image. Finally, the most indispensable expert is the researcher, as they are the expert in their scientific question, and therefore which metrics are and are not important to gather and which compromises can be accepted without derailing their analysis. Many local and global organizations now exist to help users figure out how to improve their understanding of imaging and image analysis; a non-exhaustive living list with a focus on bioimaging experiments is available at bioimagingguide.org [1].

Document everything

Even experts may disagree on the right approach to solving a particular complex problem; image analysis is a constantly-evolving discipline, where new tools and approaches emerge seemingly daily. Ultimately, there is rarely a single "right" answer to any complexity, and certainly there is no single correct workflow for any given class of problem. Ultimately, the correctness of one's analysis rests on the ability of the reader to understand what was done, how, and why. Thorough documentation of every step taken is critical for scientific validation, including metadata of the images, steps taken, programs used, tool versions, order of operations, and beyond. Checklists have been proposed to guide users through the necessary documentation for analyses [7] as well as for the kinds of metadata that are critical to capture in a bioimaging experiment [2], but in general, one will rarely regret the time taken to document an analysis, if only for one's own future understanding when publication time rolls around.

Conclusion

It takes many hours of study and work to become an expert in all aspects of image analysis, and in such an ever-changing discipline, by the time one has become an expert many aspects of one's knowledge have become out-of-date. In such a dynamic field, knowing *how to solve* problems is therefore far more valuable than *knowing all possible solutions*. While it is tempting to think professional image analysts represent a total solution to the problem of the complexity of image analysis, the tradeoffs and considerations required in designing any image analysis workflow are such that the researcher's scientific input on which tradeoffs are and are not acceptable is critical, meaning that while image analysts are critical in modern science [10], input and understanding from the researchers creating the images is indispensable.

By understanding the potential pitfalls, working in stages, ensuring high-quality data at every step, understanding how individual steps are shaping the analysis, working with experts, and documenting what was done, even a novice analyst can create high-quality analyses that fairly model the scientific question that their images sought to probe. While exact tools to perform these steps will no doubt change year by year, approaches to solving problems are likely to serve users in good stead for many years to come, and help is available in local and online global resources for those who wish to improve their skills.

Keywords

image analysis, metadata, deep learning, image processing, object detection, segmentation, workflows, best practices

Practitioner points

- Image analysis involves a number of possible complexities that may arise between the acquired image and the final answer, including dealing with source data, preprocessing images, finding objects, and measuring and/or classifying the images.
- Several common principles apply to overcoming most kinds of complexities, including to work in stages or "chunks", understanding the tools being used, assessing data quality at every stage of the process, talking to the right experts, and documenting all steps taken.

Acknowledgements

The author gratefully acknowledges members of the Cimini lab for feedback on this review. Figures were created with BioRender.com.

Funding

The work was supported by National Institute of General Medical Sciences P41 GM135019 and grant number 2020-225720 from the Chan Zuckerberg Initiative DAF, an advised fund of Silicon Valley Community Foundation. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interest

The author declares that there are no competing interests associated with the manuscript.

References

1. Senft, R. A. *et al.* A biologist's guide to planning and performing quantitative bioimaging experiments. *PLoS Biol.* **21**, e3002167 (2023).
2. Sarkans, U. *et al.* REMBI: Recommended Metadata for Biological Images-enabling reuse of

microscopy data in biology. *Nat. Methods* **18**, 1418–1422 (2021).

3. Lee, B. D. *et al.* Ten quick tips for deep learning in biology. *PLoS Comput. Biol.* **18**, e1009803 (2022).
4. Volpe, G. *et al.* Roadmap on Deep Learning for Microscopy. *arXiv [physics.optics]* (2023).
5. Miura, K. & Nørrelykke, S. F. Reproducible image handling and analysis. *EMBO J.* **40**, e105889 (2021).
6. Haase, R. *et al.* A Hitchhiker's guide through the bio-image analysis software universe. *FEBS Lett.* **596**, 2472–2485 (2022).
7. Schmied, C. *et al.* Community-developed checklists for publishing images and image analysis. *arXiv [q-bio.OT]* (2023) doi:10.48550/arXiv.2302.07005.
8. Maier-Hein, L. *et al.* Metrics reloaded: Pitfalls and recommendations for image analysis validation. *arXiv [cs.CV]* (2022).
9. Rueden, C. T. *et al.* Scientific Community Image Forum: A discussion forum for scientific image software. *PLoS Biol.* **17**, e3000340 (2019).
10. Soltwedel, J. R. & Haase, R. Challenges and opportunities for bioimage analysis core-facilities. *J. Microsc.* (2023) doi:10.1111/jmi.13192.