TABLE 2 Open-source tools for bioimage analysis.

Tool	Website	Type of tool	Purpose	Input Images	Computational knowledge required	User interface	Compatibility and dependencies
General Purpose Image Analysis Platforms							
ImageJ/Fiji[72–74]	https://imagej.net/	Image Analysis Software	Modular platform for bioimage analysis Plugins available for segmentation, tracking, and so on	Any	None, optional for macro editing CUDA installation required if using DL models	GUI, scripting	Windows, Mac, or Linux TensorFlow/PyTorch and CUDA for running DL (see TensorFlow-GPU plugin)
CellProfiler[75]	https://cellprofiler.org/	Workflow management software	Segmentation, feature extraction DL plugins available	BF, DIC, FL,TEM	None, optional for command line	GUI, command line	Windows, Mac, or Linux
Icy[80]	https://icy.bioimageanalysis.org/	Workflow management software	Workflow organization, ImageJ integration	Any	None, but useful	GUI, visual code, scripting	Windows, Mac, or Linux
KNIME[79]	https://www.knime.com/knime-analytics-platform	Workflow management software	Code-free pipelines, bridging ML and non-ML tools	Any	None	GUI, visual code	Windows, Mac, or Linux
QuPath[78]	https://qupath.github.io/	Image analysis software	Image analysis of whole-slide images	BF[a], FL, TEM	None to start, optional for scripting	GUI, scripting	Windows, Mac, or Linux
OMERO[82]	https://www.openmicroscopy.org/omero/	Image repository platform	Cloud storage and collaboration for microscopy image analysis	Any	None, but optional for Python or Java scripting	GUI, scripting, ImageJ interface	Any
BIAFLOWS[84]	https://biaflows-sandbox.neubias.org/#/	Image repository platform	Cloud-based platform for collaborative workflows, batch processing, and remote data visualization	Any	None, but optional for Python extensions	GUI, scripting	Any
napari[81]	https://napari.org/stable/	Python-based image viewer	Image viewer integrated with Python packages	Any	None for basic functions, some programming knowledge to integrate Python code	GUI	Windows, Mac, or Linux

(Continues)

TABLE 2 (Continued)

Tool	Website	Type of tool	Purpose	Input Images	Computational knowledge required	User interface	Compatibility and dependencies
Jupyter Notebook[91]	https://jupyter.org/	Interactive code documents with integrated explanations	User-friendly interface for running Python code, including DL models	n/a	None to run code, scripting knowledge needed to edit and make new notebooks	GUI, code	Windows, Mac, or Linux
Ilastik[85]	https://www.ilastik.org/index.html	ML-based image analysis software	ML Training, segmentation, counting, classification, analysis, and tracking	BF, DIC, FL, PC, SEM, TEM	None	GUI	Windows, Mac, or Linux
ZeroCostDL4Mic[92]	https://github.com/HenriquesLab/ZeroCostDL4Mic	Free cloud-based workspace to train and run DL models	Small-scale testing/training of other DL models Segmentation, denosing, super-resolution, object detection, image translation	Any	None	Jupyter notebooks	Google Colab
DeepImageJ[93]	https://deepimagej.github.io	ImageJ plugin	Deployment of pre-trained DL models	Any	None to run pretrained models, Python experience needed to install GPU connection	GUI	ImageJ, optional CUDA and TensorFlow
ImJoy[95]	https://imjoy.io/#/	Browser-based tool	Image analysis pipelines using DL tools	Any	None for web app, command line to use Conda functionalities	Browser GUI, command line	Web browser
DeepCellKiosk[96,129]	https://github.com/vanvalenlab/kiosk-imageJ-plugin	Cloud-based DL ecosystem	Segmentation, tracking	FL	None for small scale analysis, Google Cloud experience for large scale deployment	Browser GUI, ImageJ plugin, Kubernetes server	Mac, Linux, or Windows with a VM, Web browser

(Continues)

TABLE 2 (Continued)

Tool	Website	Type of tool	Purpose	Input Images	Computational knowledge required	User interface	Compatibility and dependencies
Image Preprocessing ML/DL Tools							
Noise2Void[99]	https://csbdeep.bioimagecomputing.com/tools/n2v/	Untrained DL algorithm in Python ImageJ plugin or Python	Unsupervised denoising	Any	None if used in ZeroCostDL4Mic or DeepImageJ Good ML/DL and Python knowledge for CUDA and TensorFlow installation	GUI or scripting	Requires TensorFlow and CUDA
CARE[40]	https://csbdeep.bioimagecomputing.com/tools/care/	Untrained DL algorithm in Python ImageJ or KNIME plugin	Digital image restoration, denoising	Any	None if used in ZeroCostDL4Mic or DeepImageJ Good Python and ML/DL knowledge for CUDA and Tensorflow installation	GUI, command line in TensorFlow	Requires TensorFlow and CUDA
Microscope Image Focus Quality Classifier[101]	https://github.com/google/microscopeimagequality	Pre-trained DL algorithm in Python	Classification of out-of-focus images	FL	Some Python experience for installation	GUI, Python command line	ImageJ and CellProfiler plugin
Object Segmentation Tools							
Trainable WEKA segmentation[104]	https://imagej.net/plugins/tws/	ImageJ plugin integrated with WEKA software	Image segmentation using trainable ML	BF, FL, TEM	Basic knowledge of WEKA ML models	GUI	ImageJ and WEKA
Labkit[49]	https://imagej.net/plugins/labkit/	ImageJ plugin	Manual and automated pixel classification for segmentation	BF, FL, LS, TEM	None, optional for using macro scripts	GUI	ImageJ
Microscopy Image Browser[106]	http://mib.helsinki.fi/index.html	Software package for multi-dimensional image analysis	Segmentation using traditional methods or DL with the DeepMIB plugin	FL PC, SE,TEM	None	GUI	Standalone Installation or MATLAB package
StarDist[108]	https://github.com/stardist/stardist#plugins-for-other-software	DL segmentation network in Python Plugin for ImageJ, KNIME, Icy, QuPath, ZeroCostDL4Mic, and CellProfiler	Pretrained for nuclei segmentation, can be trained for whole cell	BF, FL	None if running pre-trained models Some Python and ML/DL knowledge for training	GUI or Python scripting	Compatible software for plugins Requires TensorFlow and CUDA to train locally

(Continues)

TABLE 2 (Continued)

Tool	Website	Type of tool	Purpose	Input Images	Computational knowledge required	User interface	Compatibility and dependencies
SplineDist[109]	https://github.com/uhlmanngroup/splinedist	DL segmentation network in Python Accessible through ZeroCostDL4Mic	Whole-cell segmentation; accommodates protruding cells	BF, FL	None if running pre-trained models Some Python and ML/DL knowledge for training	Jupyter Notebook or Python scripting	Web browser Requires TensorFlow and CUDA to train locally
Cellpose[111]	https://www.cellpose.org/	DL algorithm in python Browser tool and plugin for CellProfiler, TrackMate, and ZeroCostDL4Mic	Generalist whole-cell/nuclei segmentation Annotation tool for training DL models	BF, DIC, FL, PC	None for running in browser or in DeepImageJ Basic Python and ML/DL knowledge for installation and training	Web GUI, desktop GUI, Python scripting	Web browser for testing Requires PyTorch and CUDA to train
NucleAIzer[110]	http://www.nucleaizer.org/	DL algorithm in Python Browser tool and CellPro-filer/ZeroCostDL4Mic plugin	Nuclei segmentation	BF[a], FL	None for testing-only browser tool Good Python and MATLAB and ML/DL knowledge for CUDA and TensorFlow for trainable native installation	Web GUI or desktop GUI	Runnable from browser Requires MATLAB, TensorFlow and CUDA to run natively
Mesmer[117]	https://deepcell.org/predict	DL algorithm in Python; part of DeepCell ecosystem	Whole cell segmentation using pre-trained model Re-trainable with GPU	FL	None for browser tool or ImageJ plugin Google Cloud experience for large-scale deployments	Web GUI, client application, or Jupyter notebook	Pre-trained models from browser or DeepCellKiosk DeepCell-tf requires TensorFlow and CUDA
CDeep3M[97]	https://cdeep3m.crbs.ucsd.edu/cdeep3m	Low-setup DL algorithm in Python built into browser	Pre-trained models for segmentation in 2D/3D Trainable	ET, FL, SEM	None for pre-trained web application Cloud computing or ML/DL experience for training	Web GUI, command line	Web browser Cloud computing accounts or CUDA and TensorFlow installation

(Continues)

TABLE 2 (Continued)

Tool	Website	Type of tool	Purpose	Input Images	Computational knowledge required	User interface	Compatibility and dependencies
YeastSpotter[116]	http://yeastspotter.csb.utoronto.ca/	Browser-based DL segmentation tool	Pre-trained whole cell segmentation for budding yeast images	BF, FL, PC	None for web demo Basic Python skills to run code	Web GUI, Python code	Web browser, Python 3 installation
LACSS[115]	https://github.com/jiyuuchc/lacss/	CNN library using weakly-supervised learning	Whole-cell segmentation of crowded images using approximate locations of objects	BF, FL	Basic scripting knowledge	Jupyter notebooks GUI	Google Colab
Object Classification & Post-processing Tools							
Orange[121]	https://orangedatamining.com	Data mining and workflow management software	Visual programming to build workflows with integrated ML/DL models	Any	Basic knowledge of data mining and statistics	GUI	Windows, Mac, or Linux
WEKA[105]	https://www.cs.waikato.ac.nz/ml/weka/	Java-based software distribution	Data mining using ML models	Any	Basic knowledge of data mining and Java	GUI, Java code	Windows, Mac, or Linux
CellProfiler Analyst[133]	https://cellprofileranalyst.org	Data analysis and visualization software	Classification of cellular phenotypes using extracted CellProfiler features	Any	CellProfiler knowledge	GUI	Windows or Mac
Advanced Cell Classifier[134]	https://www.microlist.org/listing/advanced-cell-classifier/	ML object classification software	Classification of cell phenotypes with manual annotation Integration with CellProfiler features	FL	Optional CellProfiler knowledge	GUI	Windows, Mac, or Linux
CellCognition Explorer[7,138]	https://software.cellcognition-project.org/	Standalone ML software Integration with CellProfiler and ImageJ	Phenotypic detection and classification Unsupervised DL algorithms for novelty detection	FL	None for standalone GUI Python and DL experience to install DL features	GUI	Windows or Mac, Python 3+ and Theano
Svetlana[135]	https://www.napari-hub.org/plugins/napari-svetlana	Napari plugin	Training of DL classifiers for 2D/3D cell phenotype classification	Any	Must use command line to install	GUI within napari	Any with Python 3.9+

(Continues)

TABLE 2 (Continued)

Tool	Website	Type of tool	Purpose	Input Images	Computational knowledge required	User interface	Compatibility and dependencies
YAPiC[139]	https://yapic.github.io/yapic/	DL algorithm for pixel classification in Python Integration with DeepImageJ and Ilastik	Generation of maps for classifying or discovering cell phenotypes	Any	Requires Ilastik Some Python and DL experience to install and train	Command line, Python scripting, Ilastik viewer	Linux only, requires TensorFlow and CUDA
Object Tracking Tools							
TrackMate[125]	https://imagej.net/plugins/trackmate/	ImageJ plugin	Single object tracking, that is, nuclei, whole cells, intracellular components Integrated DL models, that is, Cellpose	Any	None to run without DL Good Python and ML/DL knowledge to install for ML/DL model integration	GUI	ImageJ Requires CUDA and other dependencies for DL models
DeepTrack 2[128]	https://github.com/softmatterlab/DeepTrack2	General DL framework for microscopy, mainly for tracking	Single or multiple object tracking, particle tracing, cell counting, and so on.	FL, BF	Basic Python command line experience for the GUI app Python and Jupyter notebook for specific workflows	GUI, Jupyter notebooks, or Python scripting	Windows or Mac, Python 3.6+
MaMuT[130]	https://imagej.net/plugins/mamut/	ImageJ plugin	Combination of TrackMate and BigDataViewer for large-scale tracking and annotation	FL, LS	Experience using TrackMate	GUI	ImageJ and dependencies for TrackMate
ELEPHANT[131]	https://elephant-track.github.io/#/	ImageJ plugin with cloud functionality	Extension of Mastodon, for large-scale object tracking-by-detection Annotation tool for DL training with sparse annotation	FL, LS	Experience using TrackMate and cloud GPU setup	ImageJ GUI, cloud GUI	ImageJ and dependencies for TrackMate Cloud server requires CUDA

Note: The following table summarizes the reviewed open-source software and tools for bioimage analysis. Information is provided on the usage, input data type, user interface, installation, and the computational knowledge requirements. Links to the developer's websites contain installation instructions and documentation.

Abbreviations: BF, bright-field; DIC, differential interference contrast; ET, electron tomography; FL, fluorescence; LS, light sheet; PC, phase contrast; SEM, scanning electron microscopy; TEM, transmission electron microscopy.

[a]Tools specialized for whole-slide bright-field images such as H&E or immunohistochemical stains.