

Software tools for automated transmission electron microscopy

Martin Schorb^{1*}, Isabella Haberbosch^{2,3}, Wim J. H. Hagen⁴, Yannick Schwab^{1,3} and David N. Mastronarde^{5*}

The demand for high-throughput data collection in electron microscopy is increasing for applications in structural and cellular biology. Here we present a combination of software tools that enable automated acquisition guided by image analysis for a variety of transmission electron microscopy acquisition schemes. SerialEM controls microscopes and detectors and can trigger automated tasks at multiple positions with high flexibility. Py-EM interfaces with SerialEM to enact specimen-specific image-analysis pipelines that enable feedback microscopy. As example applications, we demonstrate dose reduction in cryo-electron microscopy experiments, fully automated acquisition of every cell in a plastic section and automated targeting on serial sections for 3D volume imaging across multiple grids.

As quantitative observations become standard in biological research, the need for increased data-acquisition throughput grows. Improvements in detectors, microscope hardware and software and increased computing performance have led to a significant speed-up in transmission electron microscopy (TEM) imaging, while at the same time improving data quality^{1–3}. The major bottleneck during the acquisition procedure remains the selection of features of interest, and in particular the identification of their exact coordinates for acquisition, which is done mostly manually. In image analysis, current procedures allow the automatic detection of specific features^{4–6}. In high-throughput fluorescence microscopy, feedback from image-analysis pipelines is applied to define subsequent acquisition targets^{7,8}. For TEM, existing acquisition software solutions integrate automation for specific tasks such as single-particle acquisition and tomography⁹. In cryo-TEM, feedback from image analysis is used to identify and judge the quality of particles for the acquisition of imaging^{10–12} or diffraction data^{13–15}. In materials science TEM, similar approaches exist that enable ultra-stable imaging^{16–18}. So far, however, there is no solution that incorporates advanced image-analysis procedures into the various kinds of TEM data-acquisition workflows in a generic way.

Here we present two software tools that, when combined, allow the user to incorporate established image-analysis tools and thus create specimen-specific automated feedback TEM workflows.

1. SerialEM¹⁹ offers control of the microscope and its imaging detectors in a very flexible manner. In the last years of this software's development, new features for automation were introduced, which we describe in this article. The Navigator functionality is the key element for imaging multiple regions of a specimen with predefined acquisition parameters in an automated manner. By using the built-in scripting functionality, users can create and carry out highly customized acquisition routines.
2. To provide SerialEM with the necessary coordinate and image information for automated acquisition, we have developed py-EM, a Python module that integrates common image-analysis

applications. The automatic identification of features of interest from micrographs via specimen-specific image-analysis pipelines consequently enables feedback microscopy.

Information on how to obtain the software is provided in Box 1.

Software

SerialEM. SerialEM¹⁹ is a versatile software for TEM data collection, originally intended mainly for electron tomography acquisition. While its core has been made available as open-source software, it can communicate with microscope and camera hardware from various manufacturers via a plugin scheme. Over the years, the software has been improved with considerable additional functionality to increase automation. Users can control and modify most of this functionality by using SerialEM's built-in scripting feature and thus highly individualize their acquisition procedures. External software tools such as py-EM can be triggered and run through SerialEM's graphical user interface.

A detailed introduction to the main components of SerialEM required for automated acquisition is provided in Box 2. These functions have been developed since the initial description of the software¹⁹, but they have not been described in a subsequent scientific publication. The following list of terms defines the nomenclature of SerialEM's key functionality for automated acquisition:

- Map: Navigator item that links an image with coordinates, microscope parameters and imaging state
- Montage: image of overlapping single tiles, often used as a map
- Navigator: the main tool for positioning and targeting in SerialEM; information is stored in Navigator files
- Realign to Item: procedure to position the acquisition area precisely in relation to an image feature from a map
- Registration: independent coordinate system of objects
- Virtual map: map item created as a combination of an externally generated image and microscope parameters from a reference map template; used to enable the Realign to Item process

¹Electron Microscopy Core Facility, EMBL, Heidelberg, Germany. ²Department of Hematology, Oncology and Rheumatology, University Hospital Heidelberg, Heidelberg Research Center for Molecular Medicine, EMBL, Heidelberg, Germany. ³Cell Biology and Biophysics Unit, EMBL, Heidelberg, Germany. ⁴Structural and Computational Biology Unit and Cryo-Electron Microscopy Service Platform, EMBL, Heidelberg, Germany. ⁵Department of Molecular, Cellular & Developmental Biology, University of Colorado, Boulder, CO, USA. *e-mail: martin.schorb@embl.de; mast@colorado.edu

The SerialEM software, all documentation and detailed tutorials are available at <http://bio3d.colorado.edu/SerialEM/>.

The py-EM module. Py-EM is a collection of Python functions that interpret and modify SerialEM's Navigator files and the items within, and also perform or trigger image-analysis tasks on maps. It serves as an interface between SerialEM—and hence the microscope—and any available software tool for image analysis and processing enabling feedback pipelines. We have put a strong focus on making this interface as modular as possible to leave maximum freedom in the choice of external tools for a specific experiment.

Py-EM contains functions that make use of stitching routines available in the IMOD²⁰ software package to process maps consisting of multiple tiles (i.e., montages). It then makes the resulting single stitched image available for image processing. At the same time, py-EM preserves the Navigator's coordinate system for that map. As a result of the processing, py-EM generates a Navigator file to be used for further acquisition.

The key concept of our automation approach is to provide SerialEM with a 'virtual map' that is generated for each acquisition item. With the information about the visual features of the target object in the associated image, an accurately positioned acquisition can be guaranteed through the 'Realign to Item' procedure. Once a single map has been generated with the desired acquisition settings from anywhere on the specimen, its Navigator entry will be used as a template containing all important parameters other than position and template image.

Because of the way SerialEM handles the storage locations of map images using relative paths, it is usually possible to process data directly at the microscope for immediate feedback, or transfer the entire data directory including the Navigator file to another computer for processing and then back again for acquisition.

We have used py-EM workflows to automate TEM acquisition by either running them as standalone Python scripts or embedding them into a KNIME workflow. This generic data-analysis software²¹ includes image-processing capabilities²² but also can incorporate modules from other platforms (e.g., ImageJ^{23,24}, CellProfiler^{25,26}, Ilastik⁵, R²⁷, MATLAB) into a single workflow.

The modular approach of the py-EM functions combined with the KNIME framework thus allows one to incorporate any type of existing image-processing routine identifying features of interest (Supplementary Fig. 1). Together with KNIME's graphical representation of the process, this makes the image-analysis procedure very user-friendly.

Py-EM, step-by-step tutorials and installation instructions are available at <https://git.embl.de/schorb/pyem>.

Applications

High-yield automated cryo-EM data acquisition of large particles. *Challenges.* The structural determination of protein complexes at high resolution is made possible through the averaging of several thousand cryo-electron microscopy (cryo-EM) images. To achieve the necessary throughput, systematic acquisition strategies follow a regular array of acquisition positions¹⁰. SerialEM offers tools to help users create these acquisition patterns, and its batch acquisition and scripting routines are optimized for such experiments. Image analysis can be used to target holes on holey carbon grids with good ice conditions⁹. However, neither of these strategies can capture the quality or concentration of the target particles, and thus they produce large amounts of unusable data when the sample is heterogeneous.

If target particles are large enough to be identified in overview images, they can be manually picked. In this case, the user needs to acquire additional maps at higher magnification at each point in order to successfully relocate and acquire targets precisely at the selected positions. This exposes the specimen to a higher

Box 1 | Software downloads, documentation and development

Downloadable versions of SerialEM, all documentation and detailed tutorials are available at <http://bio3d.colorado.edu/SerialEM/>.

The features presented in this paper are available in version 3.7. SerialEM can be installed on any Windows PC and run without connection to a microscope. As it handles the storage location of map images using relative paths, it enables data processing at a separate workstation while the acquisition of maps or data continues at the microscope.

External software such as py-EM scripts can be incorporated into the SerialEM GUI via the Tools menu (http://bio3d.colorado.edu/SerialEM/hlp/html/menu_tools.htm).

A collection of SerialEM scripts for various purposes, including all scripts used in the experiments described in this paper, is hosted at the public Script Repository (<https://serialscripts.nexperion.net/>).

Both SerialEM (MIT license) and the SerialEMCCD plugin to DigitalMicrograph (GPL v.2) are open-source and available at <http://bio3d.colorado.edu/SerialEM/OpenSerialEM> and <http://bio3d.colorado.edu/SerialEM/SerialEMCCD>, respectively.

The py-EM Python module is available open-source (GPL v.3) for download and further collaborative development at <https://git.embl.de/schorb/pyem>. This repository also contains detailed installation instructions and various step-by-step tutorials to introduce the functionality.

Py-EM was tested using Python 3.6 and 2.7. The module depends on the following Python packages: numpy⁴⁰, scipy⁴⁰, scikit-image⁴¹, mrcfile^{42,43}, and pandas⁴⁴ for KNIME integration. Most of these are included in the Anaconda framework (<https://anaconda.org/>). An interactive Jupyter notebook (<https://git.embl.de/schorb/pyem/raw/master/pyEM.ipynb?inline=false>) is available to guide the user through the installation procedure.

The exact versions of SerialEM and py-EM described in this paper are provided as Supplementary Software.

dose before high-resolution imaging and takes a considerable amount of time.

Solutions using SerialEM and py-EM. We have developed a workflow that allows high-magnification acquisition of manually selected target items without the need to acquire additional maps. Figure 1 shows the workflow and resulting virtual maps for a cryo-EM observation of filamentous TORC1²⁸ particles. The procedure illustrated in Fig. 1a uses py-EM functionality to generate the maps needed for realignment from the already acquired medium-magnification overview maps. The py-EM script crops the respective image areas from the grid-square maps (Fig. 1b) for each selected Navigator item and transforms the extracted images, accounting for both the relative scaling and the rotation to match a template map at medium-high magnification (Fig. 1c). Thus the extracted image that is provided as a new virtual map is suitable for direct correlation with an acquired image for realignment. A detailed description of the procedure is provided in Supplementary Protocol 1.

The script maps_acquire.py is provided in the applications directory of the py-EM code repository.

Box 2 | SerialEM key functionality for automated TEM**Navigator**

The Navigator module is the main tool for finding, keeping track of and positioning at targets of interest. The user can record positions of single points either by storing stage coordinates while browsing the specimen or by marking points on an acquired image.

Maps

A map is a specific image for which the Navigator remembers its storage location, the corresponding acquisition parameters for microscope and camera, and how to display the image to the user. Maps are generally used to provide an overview for picking acquisition targets, but they also play an important role in allowing repositioning at a target during batch acquisition.

Polygons

Polygons are connected points drawn on an image to outline an area. Their main use is to define the area to be acquired with a 'montage' at a higher magnification.

Montages

When the area of interest is larger than the field of view of a single image, SerialEM can acquire a montage of overlapping tiles. Montages are always taken at regular spacing in a rectangular pattern, but acquisition at tiles that are not needed to fill a polygon can be skipped (Fig. 3a). Users can define very large areas, such as a ribbon of sections, by moving the stage to each corner of the area and recording a 'corner point' there.

Tile positions are reached with electronic image shift, as during tilt series acquisition, or with stage movement for larger areas. Periodic focusing and other features support the acquisition of high-quality images from arbitrarily large areas⁴⁵. Montages are commonly used as overview images for finding targets for data collection.

Realign to Item

The user marks target positions on a map (either a single image or a montage). Drift will change the exact specimen position over time. Thus, the target feature will not always be centered at the recorded stage position for acquisition. To accurately reposition the acquisition area to the marked position, SerialEM has a procedure, Realign to Item, that takes images under the same conditions as those used to acquire a map and cross-correlates those images with the map. After alignment to an overview map, the realignment routine can also align to a second map acquired at higher magnification, such as one to define the target area for tilt series acquisition.

Anchor map

To acquire a pair of maps, the user indicates the lower-magnification map at the first target as the anchor map template. Thereafter, the user can take images of target areas at the acquisition magnification by pressing the 'Anchor Map' button; the program will save that image as a map, then go to the lower magnification and take and save the anchor map.

Batch acquisition

Batch acquisition by the Navigator is initiated with the 'Acquire at Points' dialog, which can acquire an image, a map or a tilt series or, alternatively, run a script at each item marked for acquisition. The dialog offers a choice of operations to be run before acquisition, such as focusing or centering the beam. The Realign to Item routine will typically be chosen to guarantee accurate repositioning.

Registrations

Navigator items that belong to the same coordinate system share the same registration number. When a grid is rotated, or removed and reinserted, the coordinate system changes, and any new Navigator items need to be assigned different registration numbers. The program then determines the transformation from one registration to another, using a set of points that correspond between the registrations (for example, features that the user marks in both an imported light microscopy image and an electron microscopy map). Alternatively, the program can find the rotation and shift that relate an image before and after rotation of the grid for dual-axis tomography by cross-correlation. This procedure is called 'Align with Rotation'. SerialEM also transforms map images of a registration accordingly before using them with Realign to Item.

Virtual map templates

Each map item contains the necessary operating conditions for the electron microscope that enable SerialEM to align to features that resemble the map image on the current specimen using Realign to Item. Linking a Navigator map item to an image while adding microscope parameters duplicated from a template map will create a virtual map entry. We use this approach to precisely realign to a specific cell in neighboring serial sections duplicating maps of target cells from a previous section to identify and target the same cell on subsequent sections ('Automated serial-section TEM', Fig. 3b,c).

Gain in throughput and dose reduction. Cryo-EM data collection is typically divided into two phases: selection of acquisition areas, and automated data collection. For large particles, the first 24 h of a 72-h acquisition slot are spent loading grids, identifying good grid squares and acquiring maps of those squares. In parallel, the manual picking of particles and running of the py-EM scripts can be done offline. Instead of waiting for the acquisition of additional maps for realignment, which would take around 8 h, the data acquisition can start directly thereafter. This increases the time available for high-resolution imaging by 20%, from 40 to 48 h.

Because the acquisition of additional maps is not required, the specimen gets irradiated with about $0.1\text{--}0.2\text{ e}^-/\text{\AA}^2$ less at each position. If maps of close particles overlapped, this extra dose would apply multiple times, compromising resolution considerably.

The published result reached 27 Å resolution from 50 micrographs of TORC1 filaments acquired entirely manually²⁸. With automation and realignment to virtual maps, the presented experiment

generated 2,660 micrographs that allowed processing to a resolution at which secondary structure elements could be identified.

Systematic acquisition of individual cells on resin sections using feedback TEM. *Challenges.* A few organelles, such as centrosomes, are present only as single copies in a given cell. In a typical TEM analysis, the probability of observing a centriole on a 200 nm section of resin-embedded cells ranges between 3% and 5%. When trying to identify a sufficient number of cells that contain such a structure, it is necessary to screen the entire population of a section at an intermediate magnification. An alternative would be to implement targeting strategies based on tagging of the event of interest, such as in correlative light and electron microscopy approaches^{29,30}. The adaptation of labeling protocols for primary samples often leads to a loss of ultrastructural detail. Systematic screens by conventional TEM are extremely challenging, as it is hard to ensure that each cell is imaged only once. This makes the task very tedious and

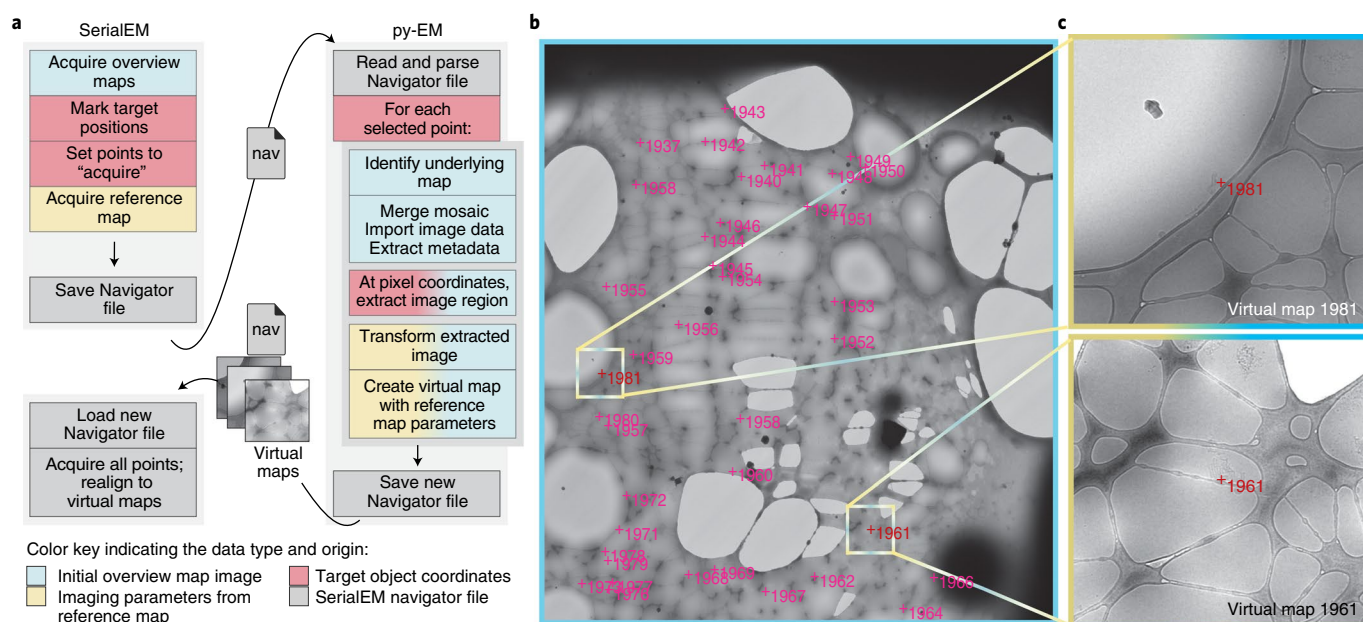


Fig. 1 | Automatic generation of virtual maps used to acquire TORC1 filamentous particles with cryo-EM. The color-coding indicates the data type and origin as defined in the key at the bottom of the figure. **a**, The application workflow depicting the individual steps in the procedure, and the communication and data transfer between SerialEM and py-EM using Navigator files and virtual map images. **b**, Overview map: a 2 × 2 montage covering a single grid square at 2,250× magnification with a pixel size of 6.2 nm without binning. The magenta crosses indicate the target positions that were manually clicked on within this map. Colored boxes and marks indicate the example regions from which the corresponding virtual maps were extracted. Scale: field of view, 47 μm; boxed regions, 3.8 μm. **c**, Two example virtual maps that the script extracted from the map shown in **b**. The image data were extracted from the overview map, while the microscope parameters were copied from the reference map. Scale: field of view, 3.8 μm. The presented experiment comprised 2,500 points from 125 maps on one grid. This procedure has so far been successfully applied in nine experiments on different specimens.

time-consuming. The application presented here is a study of centriole morphology in leukemia cells collected from individuals. To statistically assess ultrastructural diversity, we needed to acquire a large number of centrioles in different samples.

Solutions using SerialEM and py-EM. We have developed a workflow to automatically acquire individual cells on serial plastic sections. With dimensions of about 500 × 200 μm, each section typically displays from several hundred up to 1,500 cell cross-sections.

A schematic of the procedure, the input map, the output from the automated cell detection, the resulting Navigator file in SerialEM and an example virtual map are shown in Fig. 2.

To successfully identify each cell, we used the KNIME platform^{21,22} to link and incorporate various independent image-analysis tools (in our case, Python and ImageJ/Fiji) (Fig. 2a). The user can easily control image-analysis parameters through a graphical dialog without any programming knowledge. Supplementary Fig. 1a shows the layout of the KNIME workflow and the graphical controls where users can modify a few selected parameters. The image-analysis procedure is illustrated in Supplementary Fig. 1b.

In our example, the workflow uses py-EM to merge the low-magnification mosaic map(s) and to load the map image(s) into the KNIME analysis pipeline for the identification of single cells. It then extracts virtual maps at each detected position, generates polygons that match the outline of each cell linked to the corresponding virtual maps, and writes a Navigator file ready for acquisition.

We acquired an image stack containing one image per cell that could then be screened easily to categorize cells according to cell type, phenotype or specific morphological features. In the presented example, the software automatically detected 1,325 cell profiles, out of which we selected 44 cells that displayed parts of a centriole (discussed further in the section 'Automated serial-section TEM').

This manual curation step takes about 1 h. The entire image stack is presented in Supplementary Video 1.

The files needed to run the workflow in KNIME are provided as part of the py-EM code.

Gain in throughput and experimental design possibilities. Previous studies illustrating centriole morphology^{31,32} that used manual screening required several weeks to months at the microscope but resulted in only a few dozen individual observations. A comparative phenotypic analysis of a statistically relevant number of centrioles is therefore not possible with a conventional approach.

In the presented experiment, we were able to identify several hundred centrioles in leukocytes from different subjects for further high-resolution studies, with only a few days of screening required for each. The resulting virtual maps allowed SerialEM to accurately return to any chosen cell for further acquisition.

Acquiring the entire section at sufficient resolution would require around 70 × 30 tiles, resulting in a 2D image with a size that would easily exceed the memory of most current workstations. Therefore, the identification of interesting cells by systematic scanning through such a large montage is challenging both manually and via image analysis.

Using automated targeted acquisition, the user can screen a stack of images centered on every individual cell. This makes it easy to focus on morphological features and avoids redundant acquisitions. Also, because every single detected cell is imaged, statistics and classification based on observed phenotype are readily available.

Automated serial-section TEM. Challenges. In a serial-section TEM experiment, one wants to follow a feature across a series of sections that are adjacent on the TEM grid³³. When full cell volumes are being reconstructed, series of sections can easily span dozens of grids. For the acquisition of high-resolution data, such

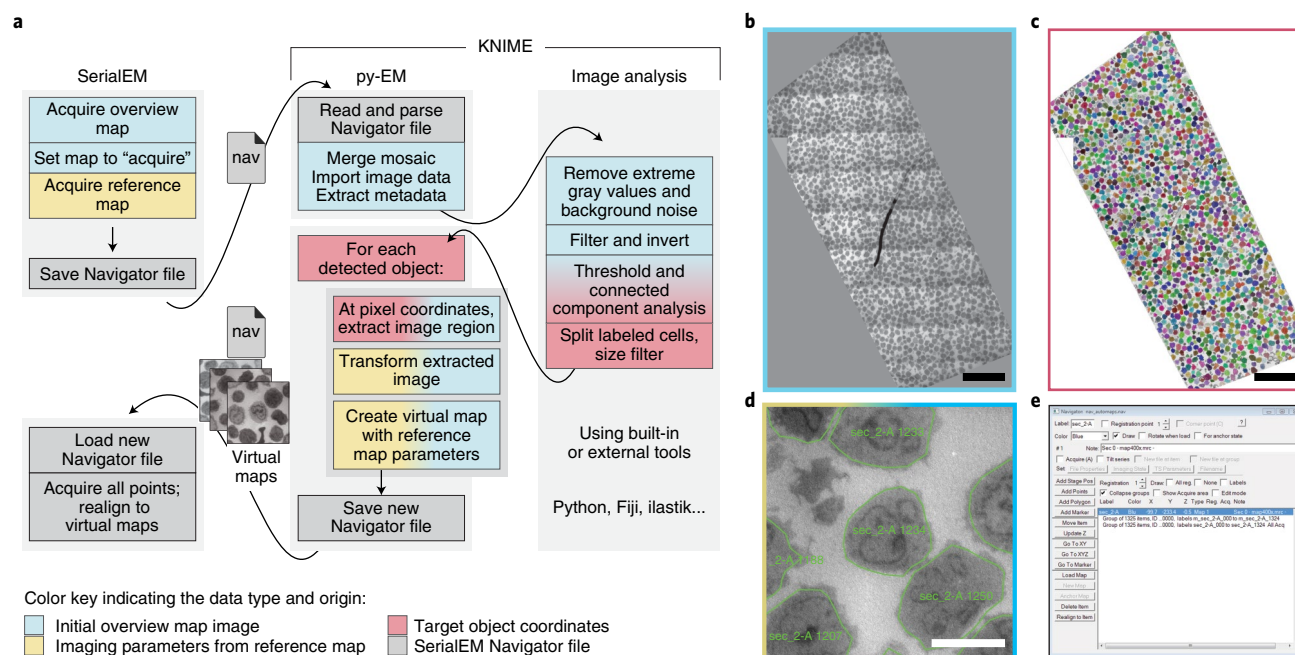


Fig. 2 | Illustration of the workflow used to automatically acquire all cells on a resin section. The color-coding indicates the data type and origin as defined in the key at the bottom of the figure. **a**, The application workflow depicting the individual steps in the procedure, and the communication and data transfer using Navigator files and virtual map images. The KNIME platform includes py-EM functionality and the image-analysis procedure. Interaction with SerialEM happens through an exchange of Navigator files and virtual maps. **b**, Overview map: a montage of the entire section. The image limits are defined manually during the merging procedure. **c**, The result of the automated cell detection. Each cell is represented by a distinct label value, displayed as a different color. **d**, An example virtual map that has been extracted from the overview map for a single cell. The green outlines depict the polygons that determine the acquisition area. **e**, The resulting Navigator file contains a virtual map and the polygon outline for each of the 1,325 detected cells. Scale bars, 50 μm (**b,c**) or 5 μm (**d**). The automated identification of cells presented in this figure was successfully applied to 26 sections with about 1,000 cells each for this experiment, and with modified image-analysis pipelines for two different specimens.

as tomograms, positional information needs to be propagated for each location on each section. While this can be done manually when one is navigating to only a handful of features per section, it quickly becomes a tedious and very time-consuming task when more objects are targeted.

Solutions using SerialEM and py-EM. In our example, we selected 60 human leukocyte cells on one of the serial sections (thickness: 200 nm) as described above. We wanted to acquire the exact same cells on the four other sections of this grid (Fig. 3a), as well as on the consecutive grids, and thus follow the selected cells in 3D. We used py-EM scripts to duplicate the list of target items and then used SerialEM's registration and realignment functionality to successfully locate cells on neighboring sections on the basis of the maps from the preceding section (Fig. 3b). For each cell of interest, these maps at moderate magnification covered a sufficiently large field of view for realignment (Fig. 3c). We then automatically acquired new maps at every target location wherein we defined the exact position for high-magnification acquisition. The SerialEM script that we used to run the acquisition maintains consistent file names and Navigator labels throughout the procedure.

We used this approach in the presented example to successfully follow each of 60 cells through a series of 45 serial sections across nine grids.

Figure 3d shows a cell acquired from 30 sections on six grids. An example of aligned images of a cell across 42 of these sections is shown in Supplementary Video 2.

A detailed description of the procedure is provided in Supplementary Protocol 2. Supplementary Video 3 provides a step-by-step tutorial on the use of SerialEM to register maps from two adjacent sections.

To resolve the structure of the respective centrosomes at higher resolution, we acquired tilt series at positions manually defined in the respective maps on the consecutive sections. We use a custom script for fast tomography acquisition. We automatically processed the large amount of acquired tilt series using IMOD's batch tomogram reconstruction³⁴ on a high-performance compute cluster, which resulted in full volumes of the desired objects. An example of such a reconstructed tomographic volume is shown in Supplementary Video 4.

We have made all SerialEM scripts used in this experiment available at the public Script Repository (<https://serialemscripts.next-perion.net/>).

Gain in throughput and experimental design possibilities. Manual selection of target cells on consecutive grids is feasible for only a small number of targets and strongly depends on the presence of easily recognizable features on the specimen. Without these, it can take an expert a day to locate just a few target positions across serial sections³⁵. Our automated approach enables parallel tracking and acquisition of hundreds of features.

Before the scripts were introduced, it took a full day to transfer maps of 60 cells across five sections by hand. With the scripts, setting up a transition from section to section takes about 10 min, independent of the number of objects. Because of the usually high number of acquisition points, the necessary imaging time for the maps strongly depends on secondary factors such as the speed of the microscope stage and the desired autofocus procedure. For 60 cells, the map acquisition takes between 15 and 30 min per section.

The main determining factor for the success rate in efforts to automatically relocate cells is the quality of serial sectioning. Failure to register maps to adjacent sections can occur if large cracks or folds are present.

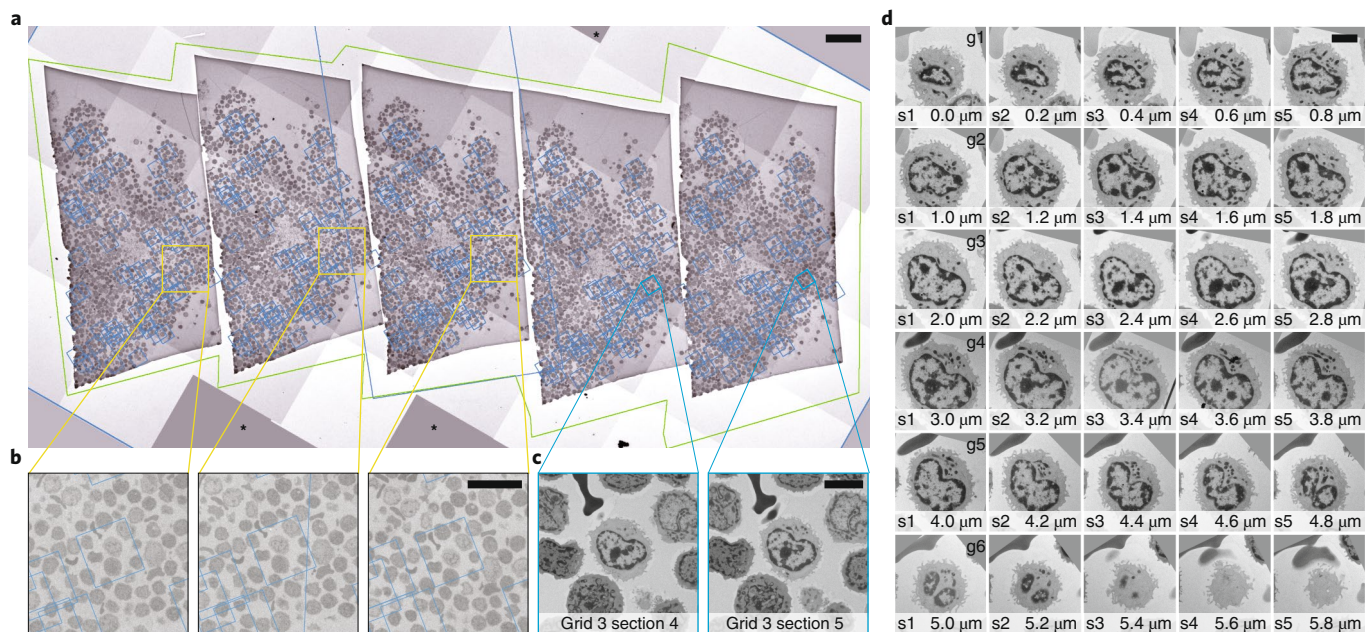


Fig. 3 | Automated acquisition of cells on serial sections. **a**, Overview of a ribbon of sections placed on a slot grid. The green outline marks the polygon used to acquire this montage map. The blue boxes denote the locations of the maps used to realign to each cell. Asterisks indicate tiles that are not acquired. **b**, Magnified regions corresponding to the yellow boxes in **a**, showing the location of one cell of interest. It is not crucial for the blue box that denotes the location of the corresponding maps to be precisely centered on the cell for repositioning. The image information used for realignment is taken from the map itself. **c**, Two maps of a single cell in neighboring sections. These images are used during the Realign to Item task. **d**, Gallery of images of an individual cell across 30 sections (thickness: 200 nm) spread across six grids. The images were automatically aligned using TrakEM2³⁹. Another cell that spans nine sections is shown in Supplementary Video 3. Scale bars, 50 μm (**a**), 20 μm (**b**) or 5 μm (**c,d**). For the presented experiment, we followed a total of 120 cells across 100 sections on 20 grids for 3 different specimens.

In our experiments, the transfer of maps from one section to the next worked almost perfectly for every cell. Problems in identifying the correct feature arose only when the edge of a section, a grid bar, a crack or fold, or large contaminations were blocking the field of view.

When features of cells or tissue do not change dramatically from one section to the next and the quality of the serial sections is good, an alignment can be done for all sections of a grid in one procedure. The reference maps will then be taken from the last section of the previous grid. There is no longer a need for user interaction after each individual section, and the procedure can be fully automated.

The maps cover not only the cell of interest but also the constellation of neighboring cells. Even when a given cell ends in z , the region is usually correctly found.

We then manually screen the maps for ending cells and misalignments when setting up the tomography acquisition.

The throughput in tomography acquisition has also increased dramatically, thanks to the availability of fast sCMOS (scientific complementary metal-oxide semiconductor)-based detectors. When combined with a sufficiently stable microscopy stage, the built-in drift correction and short exposure times allow one to skip otherwise necessary tracking steps. Compared with a traditional tilt series, acquisition is about twice as fast and the necessary time for a single-axis tomogram with 121 tilt images is reduced to around 7 min.

In the presented experiment, we followed a total of 120 cells across 100 sections on 20 grids for three different specimens. In these cells, we acquired a total of 992 tomograms in series on consecutive sections.

Discussion

We have developed both software packages described in this article with a strong focus on flexibility. SerialEM is capable of

controlling transmission electron microscopes, accessories and detectors from a range of different vendors, and covers acquisition strategies such as automated single-particle acquisition, cryo-electron tomography, large 2D mosaic tiling and batch tomography on serial sections. Data presented in this article were acquired with the following microscopes: JEM 2100Plus (JEOL Ltd., Akishima, Japan), Tecnai F30 and Titan Krios G3 (Thermo Fisher Scientific, Waltham, MA, USA).

The py-EM module aims to universally link SerialEM with a variety of software options joining image analysis and TEM acquisitions in a feedback logic. This means that the main task in setting up an automated TEM workflow for a specific application is now reduced to the development of a suitable image-analysis routine. These routines are obviously not limited to the examples described in this article, and could also include advanced feature detection using machine learning^{5,6} including convoluted neuronal networks^{36,37}. Such approaches will reduce the need for manual identification of particles in cryo-EM by automatically picking holes or large particles. In particular, given the inherently low signal-to-noise ratios in cryo-EM and cryo-tomography, machine learning approaches might enable the acquisition of objects that would not be recognizable by eye³⁸.

The possibility of imaging every entity of a specimen in a controlled manner offers entirely new possibilities for TEM. Automated screening and classification of different cell types and phenotypes or the detection of aberrant morphology can be done by computational analysis when one has large amounts of comparable data at hand. The automation of serial-section TEM reduces the workload required for this technique substantially and introduces new possibilities for large-scale 3D electron microscopy while maintaining high lateral resolution. The increased throughput and resulting capability to generate a lot more data that can subsequently be fed into automated analysis pipelines create possibilities for entirely new types of quantitative TEM experiments.

Reporting Summary

Further information on research design is available in the Nature Research Reporting Summary linked to this article.

Data availability

All raw data presented in the ‘Applications’ section of this paper are available from the corresponding authors upon reasonable request.

Received: 10 August 2018; Accepted: 15 March 2019;

Published online: 13 May 2019

References

- Winey, M., Meehl, J. B., O’Toole, E. T. & Giddings, T. H. Jr. Conventional transmission electron microscopy. *Mol. Biol. Cell* **25**, 319–323 (2014).
- McMullan, G., Faruqi, A. R. & Henderson, R. Direct electron detectors. *Methods Enzymol.* **579**, 1–17 (2016).
- Frank, J. Advances in the field of single-particle cryo-electron microscopy over the last decade. *Nat. Protoc.* **12**, 209–212 (2017).
- Meijering, E., Carpenter, A. E., Peng, H., Hamprecht, F. A. & Olivo-Marín, J.-C. Imagining the future of bioimage analysis. *Nat. Biotechnol.* **34**, 1250–1255 (2016).
- Kreshuk, A., Koethe, U., Pax, E., Bock, D. D. & Hamprecht, F. A. Automated detection of synapses in serial section transmission electron microscopy image stacks. *PLoS One* **9**, e87351 (2014).
- Arganda-Carreras, I. et al. Trainable Weka Segmentation: a machine learning tool for microscopy pixel classification. *Bioinformatics* **33**, 2424–2426 (2017).
- Conrad, C. et al. Micropilot: automation of fluorescence microscopy-based imaging for systems biology. *Nat. Methods* **8**, 246–249 (2011).
- Neumann, B. et al. Phenotypic profiling of the human genome by time-lapse microscopy reveals cell division genes. *Nature* **464**, 721–727 (2010).
- Tan, Y. Z., Cheng, A., Potter, C. S. & Carragher, B. Automated data collection in single particle electron microscopy. *Microscopy (Oxf.)* **65**, 43–56 (2016).
- Suloway, C. et al. Automated molecular microscopy: the new Legimon system. *J. Struct. Biol.* **151**, 41–60 (2005).
- Rice, W. J. et al. Routine determination of ice thickness for cryo-EM grids. *J. Struct. Biol.* **204**, 38–44 (2018).
- Nicholson, W. V., White, H. & Trinick, J. An approach to automated acquisition of cryoEM images from lacey carbon grids. *J. Struct. Biol.* **172**, 395–399 (2010).
- Coudray, N. et al. Automated screening of 2D crystallization trials using transmission electron microscopy: a high-throughput tool-chain for sample preparation and microscopic analysis. *J. Struct. Biol.* **173**, 365–374 (2011).
- Hu, M. et al. Automated electron microscopy for evaluating two-dimensional crystallization of membrane proteins. *J. Struct. Biol.* **171**, 102–110 (2010).
- Cheng, A. Automation of data acquisition in electron crystallography. *Methods Mol. Biol.* **955**, 307–312 (2013).
- Gatel, C., Dupuy, J., Houdellier, F. & Hytch, M. J. Unlimited acquisition time in electron holography by automated feedback control of transmission electron microscope. *Appl. Phys. Lett.* **113**, 133102 (2018).
- Tejada, A., den Dekker, A. J. & Van den Broek, W. Introducing measure-by-wire, the systematic use of systems and control theory in transmission electron microscopy. *Ultramicroscopy* **111**, 1581–1591 (2011).
- Liu, J. et al. Fully mechanically controlled automated electron microscopic tomography. *Sci. Rep.* **6**, 29231 (2016).
- Mastronarde, D. N. Automated electron microscope tomography using robust prediction of specimen movements. *J. Struct. Biol.* **152**, 36–51 (2005).
- Kremer, J. R., Mastronarde, D. N. & McIntosh, J. R. Computer visualization of three-dimensional image data using IMOD. *J. Struct. Biol.* **116**, 71–76 (1996).
- Berthold, M. R. et al. KNIME: the Konstanz Information Miner. In *Data Analysis, Machine Learning and Applications* (eds. Preisach, C. et al.) 319–326 (Springer, 2008).
- Dietz, C. & Berthold, M. R. KNIME for open-source bioimage analysis: a tutorial. In *Focus on Bio-Image Informatics* (eds. Vos, W. H. D., Munck, S. & Timmermans, J.-P.) 179–197 (Springer, 2016).
- Schindelin, J. et al. Fiji: an open-source platform for biological-image analysis. *Nat. Methods* **9**, 676–682 (2012).
- Schneider, C. A., Rasband, W. S. & Eliceiri, K. W. NIH Image to ImageJ: 25 years of image analysis. *Nat. Methods* **9**, 671–675 (2012).
- Carpenter, A. E. et al. CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biol.* **7**, R100 (2006).
- McQuinn, C. et al. CellProfiler 3.0: next-generation image processing for biology. *PLoS Biol.* **16**, e2005970 (2018).
- Pau, G., Fuchs, F., Sklyar, O., Boutros, M. & Huber, W. EBIImage—an R package for image processing with applications to cellular phenotypes. *Bioinformatics* **26**, 979–981 (2010).
- Prouteau, M. et al. TORC1 organized in inhibited domains (TOROIDS) regulate TORC1 activity. *Nature* **550**, 265–269 (2017).
- de Boer, P., Hoogenboom, J. P. & Giepmans, B. N. G. Correlated light and electron microscopy: ultrastructure lights up! *Nat. Methods* **12**, 503–513 (2015).
- Kukulski, W. et al. Correlated fluorescence and 3D electron microscopy with high sensitivity and spatial precision. *J. Cell Biol.* **192**, 111–119 (2011).
- Elserafy, M. et al. Molecular mechanisms that restrict yeast centrosome duplication to one event per cell cycle. *Curr. Biol.* **24**, 1456–1466 (2014).
- Marteil, G. et al. Over-elongation of centrioles in cancer promotes centriole amplification and chromosome missegregation. *Nat. Commun.* **9**, 1258 (2018).
- Bron, C. et al. Three-dimensional electron microscopy of entire cells. *J. Microsc.* **157**, 115–126 (1990).
- Mastronarde, D. N. & Held, S. R. Automated tilt series alignment and tomographic reconstruction in IMOD. *J. Struct. Biol.* **197**, 102–113 (2017).
- Quinn, T. A. et al. Electrotonic coupling of excitable and nonexcitable cells in the heart revealed by optogenetics. *Proc. Natl Acad. Sci. USA* **113**, 14852–14857 (2016).
- Cruz-Roa, A. et al. High-throughput adaptive sampling for whole-slide histopathology image analysis (HASHI) via convolutional neural networks: application to invasive breast cancer detection. *PLoS One* **13**, e0196828 (2018).
- Tajbakhsh, N. et al. Convolutional neural networks for medical image analysis: full training or fine tuning? *IEEE Trans. Med. Imaging* **35**, 1299–1312 (2016).
- Buchholz, T.-O., Jordan, M., Pigino, G. & Jug, F. Cryo-CARE: content-aware image restoration for cryo-transmission electron microscopy data. Preprint at <https://arxiv.org/abs/1810.05420> (2018).
- Cardona, A. et al. TrakEM2 software for neural circuit reconstruction. *PLoS One* **7**, e38011 (2012).
- Oliphant, T. E. Python for scientific computing. *Comput. Sci. Eng.* **9**, 10–20 (2007).
- van der Walt, S. et al. scikit-image: image processing in Python. *PeerJ* **2**, e453 (2014).
- Wood, C. et al. Collaborative computational project for electron cryo-microscopy. *Acta Crystallogr. D Biol. Crystallogr.* **71**, 123–126 (2015).
- Burnley, T., Palmer, C. M. & Winn, M. Recent developments in the CCP-EM software suite. *Acta Crystallogr. D Struct. Biol.* **73**, 469–477 (2017).
- McKinney, W. Data structures for statistical computing in Python. In *Proc. 9th Python in Science Conference* (eds. van der Walt, S. & Millman, J.) 51–56 (SciPy, 2010).
- Anderson, J. R. et al. Exploring the retinal connectome. *Mol. Vis.* **17**, 355–379 (2011).

Acknowledgements

We thank A. Krämer for initiating and coordinating the scientific project on leukocytes. We thank A. Desfosses and M. Prouteau for providing specimens for testing and application of the cryo-EM workflow. We acknowledge support from C. Palmer for mrcfile, and C. Dietz and C. von Schwerin for KNIME Image Processing and Python bindings. We thank all staff of the EM Core Facility at EMBL for helpful discussions and ideas. We thank R. Mellwig for critical reading of and comments on the manuscript. We also acknowledge support from the EMBL Center of Bioimage Analysis (CBA). I.H. is the recipient of an HRCMM (Heidelberg Research Center for Molecular Medicine) Career Development Fellowship. Work on SerialEM was supported originally by grants from NIH and more recently by contributions from users and from JEOL USA, Inc., as well as by payments for specific projects by Hitachi High Technologies America, Inc., JEOL USA, Inc., and Direct Electron, LP.

Author contributions

M.S., I.H., W.J.H.H. and Y.S. designed the experimental applications. M.S. and D.N.M. did the software development. I.H., W.J.H.H. and M.S. collected the data. All authors contributed valuable suggestions on the necessary software functionality and edited the article. M.S. and D.N.M. wrote the initial manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information is available for this paper at <https://doi.org/10.1038/s41592-019-0396-9>.

Reprints and permissions information is available at www.nature.com/reprints.

Correspondence should be addressed to M.S. or D.N.M.

Publisher’s note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

© Springer Nature America, Inc. 2019

Reporting Summary

Nature Research wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Research policies, see [Authors & Referees](#) and the [Editorial Policy Checklist](#).

Statistical parameters

When statistical analyses are reported, confirm that the following items are present in the relevant location (e.g. figure legend, table legend, main text, or Methods section).

n/a Confirmed

- ☒ ☐ The exact sample size (n) for each experimental group/condition, given as a discrete number and unit of measurement
- ☒ ☐ An indication of whether measurements were taken from distinct samples or whether the same sample was measured repeatedly
- ☒ ☐ The statistical test(s) used AND whether they are one- or two-sided
Only common tests should be described solely by name; describe more complex techniques in the Methods section.
- ☒ ☐ A description of all covariates tested
- ☒ ☐ A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons
- ☒ ☐ A full description of the statistics including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals)
- ☒ ☐ For null hypothesis testing, the test statistic (e.g. F , t , r) with confidence intervals, effect sizes, degrees of freedom and P value noted
Give P values as exact values whenever suitable.
- ☒ ☐ For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings
- ☒ ☐ For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes
- ☒ ☐ Estimates of effect sizes (e.g. Cohen's d , Pearson's r), indicating how they were calculated
- ☒ ☐ Clearly defined error bars
State explicitly what error bars represent (e.g. SD, SE, CI)

Our web collection on [statistics for biologists](#) may be useful.

Software and code

Policy information about [availability of computer code](#)

Data collection

SerialEM, v.3.7.1 and py-EM v. 20190315. Detailed description, references and download links provided in manuscript. All other software mentioned in the manuscript can be integrated into the presented workflows but was not used in the specific application examples.

Data analysis

py-EM (20190315) for Python 3.7.1, with numpy 1.15.4, scikit-image 0.14.1 and mrcfile1.1.2 packages; KNIME v. 3.7.0 with KNIME Image Processing 1.7.0, KNIP ImageJ integration 0.11.4, KNIP Python Integration 1.0.7; IMOD 4.10.21. Detailed description, references and download links provided in manuscript. All other software mentioned in the manuscript can be integrated into the presented workflows but was not used in the specific application examples.

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors/reviewers upon request. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Research [guidelines for submitting code & software](#) for further information.

Data

Policy information about [availability of data](#)

All manuscripts must include a [data availability statement](#). This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A list of figures that have associated raw data
- A description of any restrictions on data availability

All raw data presented in the applications section of this manuscript is available from the authors upon request.

Field-specific reporting

Please select the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

☒ Life sciences ☐ Behavioural & social sciences ☐ Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see nature.com/authors/policies/ReportingSummary-flat.pdf

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size	Sample sizes for the specific experiments are provided in the respective figure legends.
Data exclusions	Does not apply. The automated acquisition workflows can be defined in a specimen-specific manner to minimize false-positive events. Therefore, all acquired data can be considered.
Replication	Application 1 is independent of specimen-specific parameters. Application 2 has run successfully with minor modification of the specimen-specific image-analysis parameters. Application 3 depends on the quality of serial sections provided. A detailed discussion of the effects is provided in the manuscript.
Randomization	Does not apply. The experiments were not targeted at obtaining statistically representative population data.
Blinding	Does not apply. The experiments were not subject to information bias.

Reporting for specific materials, systems and methods

Materials & experimental systems

n/a	Involved in the study
<input checked="" type="checkbox"/>	<input type="checkbox"/> Unique biological materials
<input checked="" type="checkbox"/>	<input type="checkbox"/> Antibodies
<input checked="" type="checkbox"/>	<input type="checkbox"/> Eukaryotic cell lines
<input checked="" type="checkbox"/>	<input type="checkbox"/> Palaeontology
<input checked="" type="checkbox"/>	<input type="checkbox"/> Animals and other organisms
<input type="checkbox"/>	<input checked="" type="checkbox"/> Human research participants

Methods

n/a	Involved in the study
<input checked="" type="checkbox"/>	<input type="checkbox"/> ChIP-seq
<input checked="" type="checkbox"/>	<input type="checkbox"/> Flow cytometry
<input checked="" type="checkbox"/>	<input type="checkbox"/> MRI-based neuroimaging

Human research participants

Policy information about [studies involving human research participants](#)

Population characteristics	We obtain samples from a small number of individual patients suffering from Acute Myeloid Leukemia (AML). No link to any additional clinical or personal data is provided. Population characteristics do not affect the outcome of the described experiments as the sole purpose of the described methodology is the targeting of morphological features for further analysis. At no point any biological or medical conclusions were drawn from the acquired data.
Recruitment	Recruitment and consent of patients was performed within the NCT and DTK MASTER (Molecularly Aided Stratification for Tumor Eradication) Program (https://www.nct-heidelberg.de/forschung/nct-master.html). The decision on selecting cells from a patient for the described experiments was based on a high blast count that results in a rather homogenous cell population. This

could affect specific parameters of the image analysis. At no point any biological or medical conclusions were drawn from the acquired data.