


SOFTWARE

Open Access



PartSeg: a tool for quantitative feature extraction from 3D microscopy images for dummies

Grzegorz Bokota^{1,2}, Jacek Sroka², Subhadip Basu³, Nirmal Das³, Pawel Trzaskoma⁴, Yana Yushkevich⁴, Agnieszka Grabowska⁴, Adriana Magalska^{4*} and Dariusz Plewczynski^{1,5*} 

*Correspondence:

a.magalska@nencki.edu.pl;
d.plewczynski@cent.uw.edu.pl

⁴ Nencki Institute of Experimental Biology, Polish Academy of Sciences, Warsaw, Poland

⁵ Laboratory of Bioinformatics and Computational Genomics, Faculty of Mathematics and Information Science, Warsaw University of Technology, Warsaw, Poland

Full list of author information is available at the end of the article

Abstract

Background: Bioimaging techniques offer a robust tool for studying molecular pathways and morphological phenotypes of cell populations subjected to various conditions. As modern high-resolution 3D microscopy provides access to an ever-increasing amount of high-quality images, there arises a need for their analysis in an automated, unbiased, and simple way. Segmentation of structures within the cell nucleus, which is the focus of this paper, presents a new layer of complexity in the form of dense packing and significant signal overlap. At the same time, the available segmentation tools provide a steep learning curve for new users with a limited technical background. This is especially apparent in the bulk processing of image sets, which requires the use of some form of programming notation.

Results: In this paper, we present PartSeg, a tool for segmentation and reconstruction of 3D microscopy images, optimised for the study of the cell nucleus. PartSeg integrates refined versions of several state-of-the-art algorithms, including a new multi-scale approach for segmentation and quantitative analysis of 3D microscopy images. The features and user-friendly interface of PartSeg were carefully planned with biologists in mind, based on analysis of multiple use cases and difficulties encountered with other tools, to offer an ergonomic interface with a minimal entry barrier. Bulk processing in an ad-hoc manner is possible without the need for programmer support. As the size of datasets of interest grows, such bulk processing solutions become essential for proper statistical analysis of results. Advanced users can use PartSeg components as a library within Python data processing and visualisation pipelines, for example within Jupyter notebooks. The tool is extensible so that new functionality and algorithms can be added by the use of plugins. For biologists, the utility of PartSeg is presented in several scenarios, showing the quantitative analysis of nuclear structures.

Conclusions: In this paper, we have presented PartSeg which is a tool for precise and verifiable segmentation and reconstruction of 3D microscopy images. PartSeg is optimised for cell nucleus analysis and offers multi-scale segmentation algorithms best-suited for this task. PartSeg can also be used for the bulk processing of multiple images and its components can be reused in other systems or computational experiments.



Keywords: Chromatin, Super-resolution microscopy, 3D FISH, Electron microscopy, 3D reconstruction, Bioimaging, Nucleus, Segmentation, Batch processing

Background

For a decade, high-throughput bioimaging techniques offered a robust tool for studying molecular pathways and morphological phenotypes of cell populations subjected to various conditions [1, 2]. Due to recent advances in light and electron microscopy, a large number of input images can be produced in a relatively short time span. Therefore, it becomes critical to extract numerical features from imaging data in an automated, unbiased and simple way.

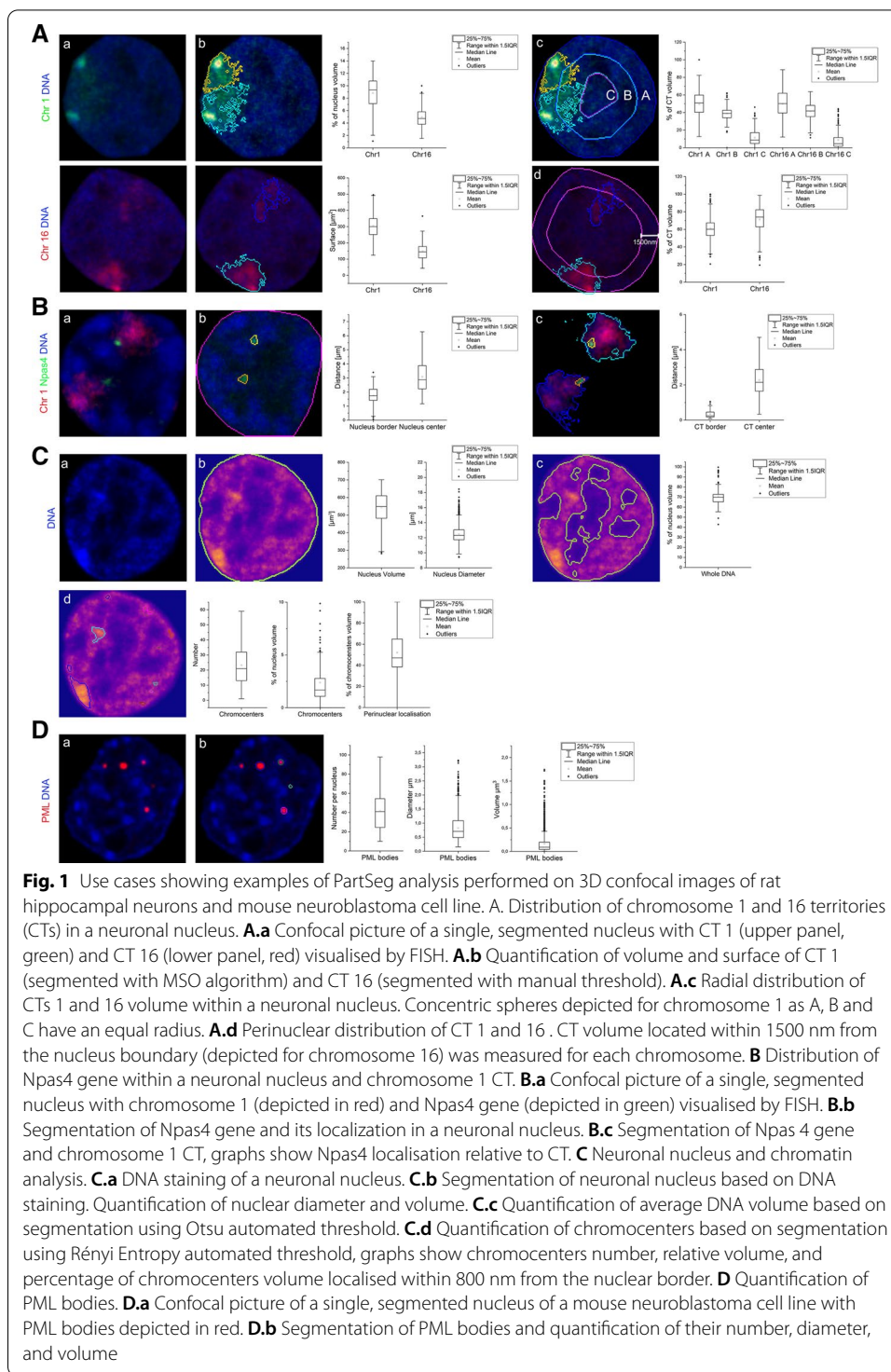
The cell nucleus is a highly organised and crowded organelle, composed of many functional and structural domains (Fig. 1). The past 2 decades of research show that changes in higher-order chromatin structures, that is, spatial and temporal rearrangements of chromatin, are involved in transcriptional control and other cellular functions [3, 4]. The 3-C based methods, which decipher folding of chromatin, quantify the probability of interaction between two genomic fragments, such as promoters and enhancers, that are in close spatial proximity, but may be dispersed far across the genome. To refine this biochemical data, showing the outcome averaged over millions of cells, precise microscopic analysis of nuclear components like genes, chromatin, and nuclear domains is necessary. Therefore, microscopy is often used for verification, at scales of individual cells or even genetic copies.

Automatic segmentation of nuclear structures poses many challenges because of confinement boundaries imposed by the nucleus, dense packing, and significant background noise, resulting in the signal overlap. Optimal segmentation of boundaries via intensity-threshold based methods is difficult, if not impossible, especially for conjoined structures. Yet, segmentation of ROI is essential for expert evaluation and serves as a basis for calculating numerical descriptors of data.

Because of this, there is no possibility of creating a parameterless, universal algorithm for processing data from innovative experiments. Rather a flexible methodology for tuning data processing methods to current datasets is preferable. We take this task one step further by facilitating the process for users without programmatic knowledge.

In this paper, we present PartSeg, which offers a novel segmentation algorithm for high-throughput imaging data and is specially optimised for the analysis of cell nuclei. PartSeg relies on 3D segmentation of regions of interest (ROI) and subsequent analysis over a large number of morphological features of the segmented structures. It is equipped with an easy-to-use graphical user interface (GUI) and can alternatively be utilised as a Python library in composite data processing pipelines. Therefore, it meets the expectations of both biologists and bioinformaticians. Moreover, PartSeg allows for batch processing of datasets coming from different sources e.g. 2D/3D images from light or electron microscopy, which can be managed without the support of a programmer by the use of rapid prototyping on sample data.

The main design goal of PartSeg was to make it simple to learn for new users. To achieve this, we determined typical use cases for users interested in studying cell nuclei (see “[Typical steps of the nucleus analysis workflows](#)” section) and planned the GUI so that swift proficiency in the usage of the toolkit can be quickly achieved. It is consistent,



ergonomic, and does not introduce unnecessary notions or mechanisms. For example, compatibility is maintained by supporting a diverse set of 3D image formats stored in the filesystem enabling interoperability with all popular platforms. This way no importing, or preprocessing is needed and the user can focus on the tasks at hand, rather than

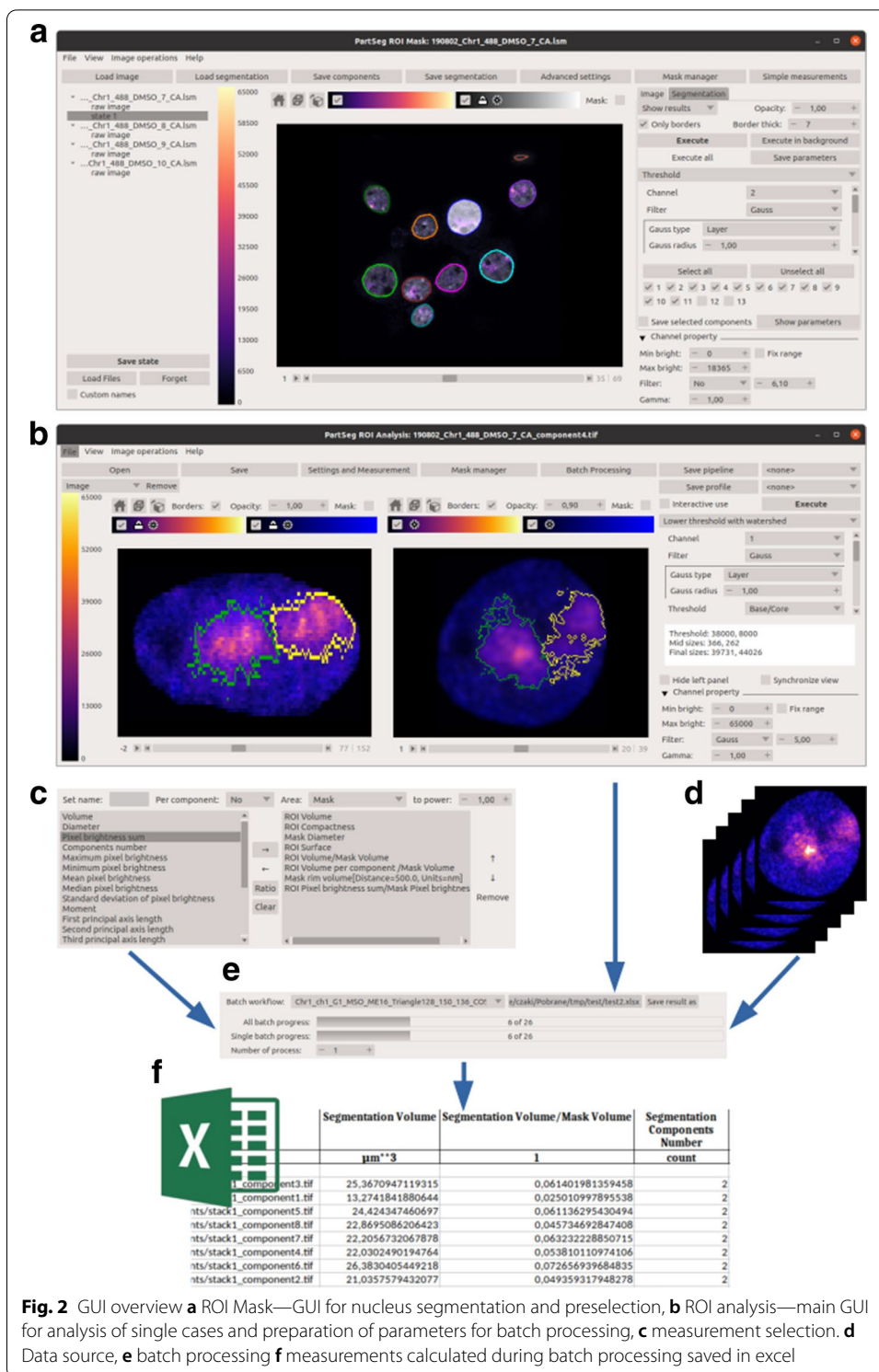


Fig. 2 GUI overview **a** ROI Mask—GUI for nucleus segmentation and preselection, **b** ROI analysis—main GUI for analysis of single cases and preparation of parameters for batch processing, **c** measurement selection. **d** Data source, **e** batch processing **f** measurements calculated during batch processing saved in excel

the software’s inner workings. This follows Human-Computer Interaction studies [5], which show that users expect systems to work and will choose those that are easier to use.

The current revision of the PartSeg interface includes support for 2D and 3D multi-channel data. It consists of two separate tools named *ROI Mask* and *ROI Analysis*. The former is intended for the extraction of objects of interest from the data set and can store the basic information on those objects in a separate mask file. The latter can be used for the measurements of morphological parameters of specific structures within the extracted objects.

Finally, some ergonomics improvements are incorporated following examination of typical usage patterns. For example, segmentation parameters are automatically saved during and in between sessions, and specific segmentation profiles can be saved by users. The segmented area is graphically presented on top of the data which simplifies visual validation of utilised parameters. PartSeg ROI Analysis is also equipped with a synchronised view, where two windows are used simultaneously, permitting the user to compare the current segmentation with multichannel raw data or with other segmentation computed with different parameters. Last but not least, parallel batch processing is available, with the number of processes adjusted by the user depending on available system resources.

Implementation

To optimise PartSeg GUI we have identified typical steps of nucleus analysis workflow by examining several real-life use cases.

Typical steps of the nucleus analysis workflows

Preselection of the nuclei in 3D images In order to quantitatively analyse structures within the cell nucleus, it is necessary to segment individual nuclei from the 3D image. Nuclei can be segmented based on any nuclear staining and thresholding method (see Fig. 2a).

Selection and verification of parameters for extraction of complex ROI inside the nucleus' territory Next, segmentation of ROI within nuclei is necessary in order to obtain numerical values for nuclear assemblies. Extraction of complex structures from 3D images usually requires testing of several algorithms in order to select optimal parameters. Visualisation of performed segmentation on top of raw data allows the user to select methods and settings facilitating the extraction process. As was mentioned before, nuclear space is very crowded. For example pairs of chromosome territories (CTs) are often localised close to each other, which makes segmentation of a single copy very difficult even for an experienced eye. This step is tuned on a sample subset of nuclei (see Fig. 2b).

Measurement set A choice of numerical parameters (measurements) needs to be made (see Fig. 2c). Values for those parameters are calculated for each image as a result of manual or batch segmentation processing and presented to the user in the form of a spreadsheet. The user then performs some simple computations and plots the results. Results should be based on the initial picture resolution and expressed in physical units.

Usually, aforementioned steps are performed manually on a small subset of the data. This confirms the proper selection of the segmentation method and parameters before processing the whole dataset.

Batch processing When single image analysis is considered satisfactory, it can be repeated by executing batch processing on all input images to collect data for statistical analysis (see Fig. 2d, e). Results are shown for each individual nucleus and structure (see Fig. 2f). This allows conclusions to be drawn for the entire cell population based on data acquired from individual nuclei simplifying the subsequent tasks of categorising nuclei or structures and performing statistical analysis. Comparing, aforementioned biochemical techniques show an averaged outcome from large populations of cells, which does not necessarily reflect the biological heterogeneity of individual components.

In the "[Examples of PartSeg application on real data](#)" section, we provide four examples of analysis, which have been done according to such a workflow.

Although several general and mature bioimaging tools like ImageJ [6], Icy [7], CellProfiler [8], and ImagePy [9] exist, none of these provide support for all of the aforementioned steps nor do they possess an interface optimised for the workflow as a whole. A detailed comparison with other systems is provided in the "[Comparison to existing tools](#)" section.

PartSeg modules

Based on a typical workflow identified for analysis of a cell nucleus we have created two separate modules named *ROI Mask* and *ROI Analysis*, which allow for ROI selection and ROI analysis. We have decided to split the tool into two modules to make the user interface more coherent and task-oriented. Modules are described in the following chapter and presented in Fig. 2.

ROI mask: a module for selection of nuclei in 3D images

To facilitate quantitative analysis of structures within the selected ROI, e.g. cell nucleus, we have developed a separate module *ROI Mask*. It allows to create masks for individual ROIs, and based on those masks cut them out from the 3D picture as shown in Fig. 2a. *ROI Mask* is equipped with a basic threshold-based algorithm with a small extension to better extract cell nuclei. Additional algorithms can be added using the plugin system. For further details please see the "[Plugins](#)" section. Usually, in the case of heterogeneous biological samples with diverse staining intensities, one set of parameters is not enough to properly extract all ROI. Therefore we have implemented the possibility of multiple rounds of segmentation performed with different parameters adjusted for individual ROIs. Such an approach allows for the selection of specific ROI from the whole population present in the picture. For example, it provides the opportunity to study cells in a particular cell cycle phase or those of a specific type, like single cancerous cells embedded in the normal tissue. After final ROIs are obtained it is possible to save each ROI as a separated file or save the project with information on how each ROI is created.

ROI analysis: a module for measuring ROI parameters

The second module *ROI Analysis* is a more general tool in which both segmentation and analysis can be performed (see Fig. 2b). This module is equipped with several segmentation algorithms described in the "[Algorithms for ROI extraction and analysis](#)" section. As

an input raw data, as well as collection of ROIs together with their masks created with *ROI Mask*, can be used. The analysis can be done based on the measuring set defined by the user (see Fig. 2c) for a single ROI or in a batch mode (see Fig. 2d, e). As an output, a spreadsheet is generated containing measurements for each ROI as well as a detailed description of parameters used in the segmentation (see Fig. 2f). After the final ROI is obtained it is possible to save the project including the image, ROI, and all steps needed to calculate ROI. It is also possible to save the steps alone as a pipeline for future use in batch processing.

Algorithms for ROI extraction and analysis

In PartSeg there are several provided algorithms for segmentation and measurements of ROI, which we listed below, and more can be added as plugins. Here, we categorise the algorithms into two groups. The first contains algorithms, which are designed for ROI extraction from an initial 3D image, outputting a collection of ROI. The second contains algorithms designed for measuring various ROI features. These algorithms take a collection of ROIs as an input and use it to compute numerical or aggregate features.

ROI extraction

The algorithms for finding collections of ROI rely on threshold segmentation and include:

- *Thresholding* ROI is defined by thresholding followed by identification of connected components and minimum size filtering. The upper or lower threshold level can be set manually by the user or calculated with common methods such as Otsu, Shanbhag, etc.
- *Range thresholding* allows the user to set a specific threshold range marking ROI, which can be used to eliminate background staining.
- *Multiple threshold Otsu* is a generalisation of the histogram-based Otsu method (see [10]) which identifies multiple types of ROIs using a set of thresholds.
- *Thresholding with watershed* ROI is found by two-stage thresholding. First core objects are identified as in *Thresholding*, after which second thresholding marks the

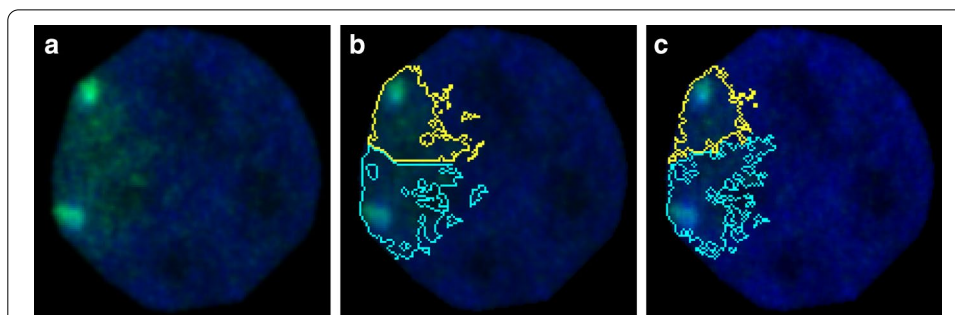


Fig. 3 Watershed and multiscale opening comparison. Difference between object separation with Multiscale Opening and Watershed transform. **a** Confocal picture of a single, segmented nucleus with the FISH signal from a chromosome 1 probe (depicted in green). **b** Segmentation of chromosome 1 signal using the Watershed algorithm. **c** Segmentation of chromosome 1 signal using the Multiscale Opening algorithm

whole area of interest. Voxels from this area are found using selected watershed-like methods initiated from core objects.

- *Multiscale Opening* is a type of watershed-like thresholding that is unique to PartSeg, therefore we provide more details. Often in the analysis of imaging data, the distance between two ROIs is smaller than the angular resolution of the used imaging method. Historically, the first approach to separate such ROIs was the watershed [11] method. However if an object has a diverse morphology i.e. exhibits regions of higher brightness, classical watershed will incorrectly split such an object into multiple ROIs. To overcome this problem, the watershed transform was developed [12]. Unfortunately, watershed transform works best for objects, which are spherical in a chosen metric. Because many nuclear domains are densely packed and non-spherical, it was important to develop methods that are capable of reliable segmentation. Therefore in PartSeg, we have implemented a novel Multiscale Opening [13]. It takes into account both the change in brightness and the physical distance along the joining path. The main difference between other watershed like algorithms and Multiscale Opening is its iterative voxel labeling. Voxels can be labeled only if they are closer to any object than to the background. This approach produces better results for stretched and non-symmetrical objects. A comparison of segmentation of two adjoining chromosomal territories made using watershed and Multiscale Opening is presented in Fig. 3.

ROI extraction can be performed multiple times in an incremental manner. After determining the first collection, the selected ROI can be converted to a mask and used as an input for the next level of extraction. When the final collection is obtained, its numerical description can be determined using algorithms described in the next section.

Measuring of ROI features

Measurements can be performed on an area defined with *ROI*, collection of *ROI*, or a *Mask*. Several common measurement methods like *Volume* or *Diameter* are available which require no explanation.

PartSeg also provides more sophisticated methods of ROI measurements on *Mask*. They can, for example, be used to measure gene positions in the nucleus or to measure the relative difference of concentrations of proteins within regions of the nucleus. These methods include:

- *Border rim* allows measuring the total volume or pixel brightness of the selected ROI, which is located within a given distance from the border of the mask. An example application would be to identify a gene or the portion of chromosome territory positioned in close proximity to the nuclear rim (see Fig. 1A.d. and [14])
- *Mask distance splitting* splits the mask into concentric regions of increasing distance from the mask center, which can be of the equal radius or equal volume. It allows measuring the volume and pixel brightness of ROI found within the designated regions. For example, it shows the radial position of CTs within the nucleus (see Fig. 1A.c. and [15]).

- *Mask-ROI distance* distance from ROI to mask is calculated based on their *mass center* (taking brightness into account), *geometrical center*, or *border distance*.

An example of an application would be to identify gene positioning within the nucleus (see Fig. 1A.d. and [14])

Some other noteworthy measurement types include:

- *Moment* One of possible measurements of mass (pixel brightness) distribution inside ROI. It allows us to determine if the structural mass is concentrated or distributed evenly. The formula is $\sum_{v \in ROI} m_v r_v^2$, where m_v is the brightness of a voxel and r_v is its distance from the ROI's center of mass. The interpretation is similar to the classical moment of inertia. This measure assumes that voxel brightness is one-to-one correlated with object density.
- *First/Second/Third principal axis* Aligns ROI using weighted PCA, then calculates ROI length along the corresponding axis. The weights correspond to voxel brightness and their position vector is in relation to the ROI's center of mass. This measurement can be used to determine basic shape of ROI.

Plugins

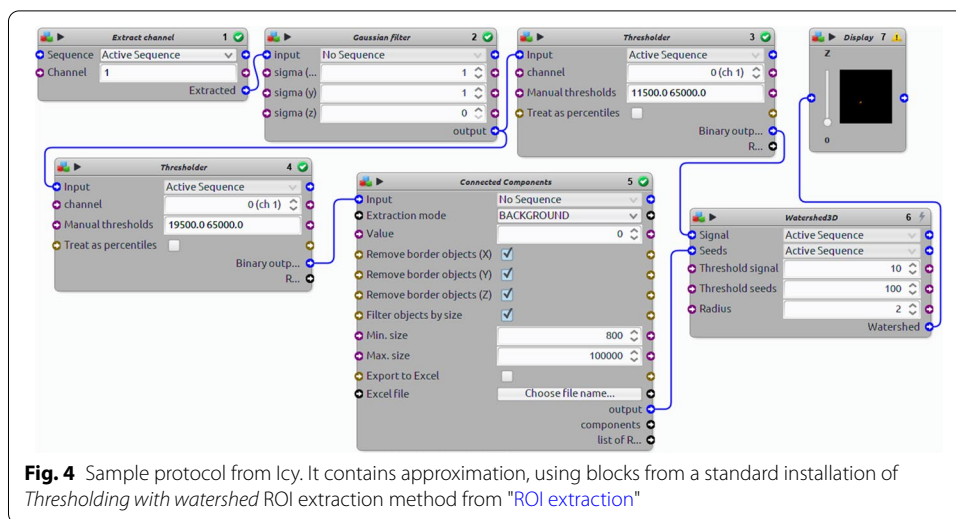
PartSeg has a plugin system which allows to expand it with additional features. This is convenient if one wants to add experiment specific computational methods and include tailored dependencies that need to be selected to match specific computer configuration. As an example, we have provided plugins incorporating deep learning algorithms. It is well recognised that Deep Learning gives very good results when applied to the nucleus segmentation. There are many published models (both networks topologies, and their software implementations), for example, Stardist [16, 17] and Cellpose [18]. Typically such highly-optimised statistical methodologies have specific configurations that depend on processor type, graphic card, and installed drivers, hence we implement them as plugins. Our plugins for these two models are available at: <https://pypi.org/project/PartSeg-stardist/> and <https://pypi.org/project/PartSeg-cellpose/>.

PartSeg with GUI versus PartSeg as a Python library

PartSeg can be used as a standalone program or as a Python library for example, as part of a larger pipeline. We provide two tutorials, which show how to use PartSeg from the biologist's perspective with the Graphical User Interface (GUI) and from the bioinformatician's perspective as a library.

Using PartSeg GUI

To show how our workflow can be applied to some real-life data with GUI we have prepared a step by step tutorial https://github.com/4DNucleome/PartSeg/blob/master/tutorials/tutorial-chromosome-1/tutorial-chromosome1_16.md. It presents how PartSeg can be used to segment nuclei from 3D confocal images and subsequently analyse several parameters of chromosomal territories in chromosomes 1 and 16. This use case can be broken down into three parts.



First, the segmentation of nuclei is performed using the DNA signal. Segmented nuclei are cut from the original pictures and the mask files containing segmentation parameters are created.

Second, in order to quantify features of chromosome 1 territories (CT1), segmentation of its specific signal is carried out. The parameters for segmentation are adjusted to accurately cover the chromosome’s staining. Next, the method for measuring several morphological parameters of CT1 is presented. These parameters are calculated based on a fixed threshold value. Additionally, the volume ratio of CT1 and its nucleus is calculated.

Finally, the previously established settings profile is used in batch mode to measure features of the nuclei and CT1s. After repeating the process for chromosome 16, a comparison of size between chromosome 1 and 16 can be done in a fully automated way (see Fig. 1) A. In Fig. 2 we utilise the PartSeg GUI to perform steps from the typical workflow in the "Typical steps of the nucleus analysis workflows" section.

Using PartSeg as a Python library

https://github.com/4DNucleome/PartSeg/blob/master/tutorials/tutorial_neuron_types/Neuron_types_example.ipynb In this tutorial we present how PartSeg components can be used as a Python library. Images of cells immunostained for markers Prox1 and CamKII and counterstained with DNA dye were acquired with confocal microscopy. Initially, segmentation of hippocampal neuron nuclei is shown. Next, a combination of PartSeg components and custom code is presented in order to assign segmented nuclei to 4 different classes based on the aforementioned markers. Finally, the segmentation of chromatin, based on DNA staining is performed to obtain a set of measurements. In the final step, Matplotlib plots of the obtained data are generated. The tutorial shows how to create the aforementioned pipeline using API method for segmentation, or simply load it from a file exported in PartSeg GUI.

API documentation of PartSeg is available at <https://partseg.readthedocs.io/en/stable/>. PartSeg is build using well known Python libraries like napari [19], numpy [20], and SimpleITK [21].

Discussion

Comparison to existing tools

There exist several general tools for image processing and analysis like ImageJ [6], Icy [7], CellProfiler [8]. These are robust apps that have been developed over years and contain a wealth of options and plugins for many applications. For inexperienced users, such a multiplicity is difficult to overcome and the general nature of these tools does not enforce a workflow adapted to nucleus analysis.

ImageJ's, Icy's and ImagePy's main workflow revolve around atomic operations like thresholding or filtering. In those tools presentation of processing results in the context of input data requires many additional steps. For example, Icy allows for the creation of a pipeline in a graphical setting but does not feature a simplified view or export of key parameters (see Fig. 4). This type of work resembles graphical programming, where user needs to think in the context of atomic operations and not a semantic meaning. For now, such an editor is unavailable for ImageJ and ImagePy, both offer only scripting, which demands basic programming knowledge.

CellProfiler provides a tool for pipeline creation and execution, however, it does have an exploration mode. Although it gives a possibility to preview intermediate results, the implemented viewer was not designed for 3D data, therefore it is restricted to only a few layers automatically selected by the tool. The user is expected to define a multi-step computational pipeline himself, this often proves too difficult a task to overcome without programmatic experience and results in an interface less ergonomic in this respect to one offered by PartSeg.

We have taken an alternative approach, where the GUI is simple, compact, and provides an easy exploration of various algorithms and their wide range of parameters with immediate visualisation of results. The GUI is organised around the workflow, which we have defined with our users (see "Implementation" section). Even though our central focus is on the analysis of cell nuclei, PartSeg is general enough to cover various other use cases. The modular structure of PartSeg allows saving of ROI segmentation parameter settings at many levels of complexity: from simple profiles, through pipelines, to projects containing entire segmentation and imaging data. This enables easy collaboration and facilitates repeatable and verifiable research as the entire segmentation can be published as supplementary material and easily repeated by anyone interested. On the other hand, power users can integrate PartSeg in their analysis pipelines with the use of Python API that we provide.

From the two other tools dedicated specifically to analysis of cell nucleus, NEMO [22] and TANGO [23] only the latter still can be obtained. TANGO, reviewed in [24], requires the installation of external dependencies to work with full efficiency. Such installation might be challenging for a typical user and cannot be done without administrative privileges. Moreover, TANGO's GUI seems to be highly unintuitive. It seems that all aforementioned tools are better suited for bioinformaticians, than wet-lab scientist. Yet, it is a wet lab scientist, who the best understands whole experimental setup

and visual outcome and can assess if ROI is segmented properly. The philosophy behind PartSeg is to provide GUI equipped in high-level operations with semantic descriptions, as well as Python library with API. Implementation details are hidden from the user, but at the same time, they are empowered with a multichannel 3D viewer, capable of showing ROI in the context of input data and recording the analysis parameters for reuse on other datasets. Users with more advanced programming knowledge can use the Python API and easily combine PartSeg with other data analysis libraries in their processing pipelines.

Examples of PartSeg application on real data

This section contains four examples of the workflow aforementioned in the "[Implementation](#)" section applied to 3D confocal images of rat hippocampal neurons cultured in vitro (Fig. 1a–c) and mouse neuroblastoma cell line (Fig. 1d). Graphics and values shown on Fig. 1 were obtained using PartSeg.

First, we analysed the surface, relative volume, and distribution of territories of chromosomes 1 and 16 in nuclei of rat hippocampal neurons (Fig. 1a). The chromosomes were stained using FISH with chromosome 1 and 16 paint probes. The measured volume of both chromosome territories (CT) roughly correlates with their size in Mbp—267.9 Mbp for chromosome 1 and 90.2 Mbp for chromosome 16, which is 10 and 3,5% of the whole genome accordingly. Some discrepancies are expected due to varying distributions of heterochromatin and euchromatin in both chromosomes, as well as conditions of hybridisation, which require DNA heat denaturation. PartSeg allows us to analyse the radial distribution of segmented structures within the nucleus (Fig. 1A, c), together with their proximity to the nuclear border (Fig. 1A, d, and B, b). We checked localisation of both CTs in 3D nuclear space. Both CTs were in close proximity to the nuclear periphery (Fig. 1A, c, and d) and on average most of the CT volume was located within 1500 nm from the nucleus boundary (50% of the volume of CT1 and 70% of CT16). It was shown that chromatin of mouse and human cells show the presence of lamina-associated domains (LADs) distributed along all chromosomes, which cover around one-third of the whole genome, therefore all chromosomes are in contact with nuclear lamina located at the nuclear border [25]. However the spatial organisation of CTs is flexible, permitting many local and long-range contacts of genes and regulatory elements, which influence their function [26].

Data suggest that radial positioning of genes often correlates with a transcriptional state, with actively transcribed genes located in the interior and silent genes at the periphery of the nucleus. Also, the gene position within CT reflects its state of expression, where active genes tend to localise the CT boundary [27]. Therefore, we have analysed the distribution of the *Npas4* gene in rat hippocampal neurons subjected to sequential FISH with chromosome 1 paint probe and *Npas4* probe (Fig. 1b). *Npas4* is a transcription factor involved in structural and functional plasticity, actively transcribed in neurons [28]. In mature rat hippocampal neurons *Npas4* gene was located close to the border of chromosome 1 (Fig. 1B, c) at the same time remaining relatively far from the nuclear periphery (Fig. 1B, b), which is in agreement with the aforementioned observations for active genes.

Next, we looked at nuclei and chromatin of mature rat hippocampal neurons (Fig. 1C). For 750 nuclei we have determined an average volume of $556 \mu\text{m}^3$ and a diameter of $12,5 \mu\text{m}$ (Fig. 1C, b). DNA stained with Hoechst dye occupied on average 70% of the nucleus volume (Fig. 1C, c). We have also calculated the average number and volume of chromocenters, which contain highly condensed and constitutively silenced, pericentromeric chromatin (Fig. 1C, d). We found an average number of 20 chromocenters per nucleus, which occupied around 2% of the nuclear volume, with 50% of them localised within 800 nm of the nuclear border. Quantification of e.g. chromatin volume is interesting as its condensation is a dynamic process dependent on many physiological and environmental factors [29]. Chromatin compaction changes accompany cell differentiation, cell division, cell death, senescence, ischemia and oxidative stress as well as constitutive expression silencing [30].

As the last example, we have calculated the number, diameter, and volume of PML bodies in mouse neuroblastoma cell lines (Fig. 1d). PML (promyelocytic leukemia) bodies are matrix associated, spherical, nuclear bodies, with a diameter of $0.1\text{--}1.0 \mu\text{m}$, which can be found in most cell lines and many tissues. Nuclei of asynchronous neuroblastoma cell lines had on average 40 PML bodies per nucleus, with a diameter of $0,8 \mu\text{m}$ and volume of $0,17 \mu\text{m}^3$. It was shown that the size and number of PML bodies heavily depend on the cell cycle phase, cell type, and stage of differentiation. Changes in the number and volume of these nuclear domains accompany induction of stress, senescence, and tumorigenesis [31].

Figure 1 presents some examples of nuclei and nuclear domains, like chromosome territories, genes, PML bodies, chromatin, and chromocenters, which can be quantitatively analyzed in PartSeg. However, PartSeg can easily be adapted to quantify other cellular structures like mitochondria or Golgi apparatus.

Conclusions

In this paper, we have presented PartSeg which is a tool for precise and verifiable segmentation and reconstruction of 3D microscopy images. It enables easy collaboration and facilitates repeatable and verifiable research in the sense that the entire segmentation with parameters can be published as supplementary material and easily repeated by anyone interested. PartSeg is optimised for cell nucleus analysis and offers multi-scale segmentation algorithms best-suited for this task. It can also be used for bulk processing of multiple images and its components can be reused programmatically in other systems or computational experiments.

Furthermore, simple storing of the whole segmentation process with settings and data empowers cooperation and independent verification of results in the spirit of data provenance [32], and open and verifiable science.

Abbreviations

3-C: Chromosome Conformation Capture-method of chromatin interaction study; API: Application Programming Interface; BSD: Berkeley Software Distribution Licenses; CTs: Chromosome territories; GPL: GNU General Public License; GUI: Graphical User Interface; PML: Promyelocytic leukemia proteins; ROI: Regions of Interest; Mbp: Mega base pairs.

Acknowledgments

This paper is dedicated as a memorial to Professor Grzegorz Wilczyński, who suddenly passed away on 13 July 2020. Grzegorz has contributed to the idea and initial development of PartSeg.

Authors' contributions

GB and DP proposed the project, and the software concept; GB performed the code development; DP, AM, JS provided the guidance, and the project management; GB, AM, JS and DP were involved in software testing; PT, AG, YY, and AM performed the biological experiments; SB and ND developed the multiscale opening (MSO) segmentation algorithm; GB implemented the MSO in PartSeg; JS, AM, PT, DP, GB were responsible for the writing of the manuscript. All authors read and approved the final manuscript.

Funding

The computational part of the manuscript was supported by the Polish National Science Centre (2019/35/O/ST6/02484, 2014/15/B/ST6/05082, 2020/37/B/NZ2/03757), and the Foundation for Polish Science co-financed by the European Union under the European Regional Development Fund (TEAM to DP). The experimental data used in the study were co-financed by Polish National Science Centre (UMO-2014/15/N/NZ2/00379, UMO-2015/18/E/NZ3/00730). DP was co-funded from funded by POB Cybersecurity and data analysis of Warsaw University of Technology within the Excellence Initiative: Research University (IDUB) programme, and by Warsaw University of Technology within the Excellence Initiative: Research University (IDUB) programme. The publication was co-supported by European Commission Horizon 2020 Marie Skłodowska-Curie ITN Enhpathy grant 'Molecular Basis of Human enhanceropathies' and by the National Institute of Health USA 4DNucleome grant 1U54DK107967-01 "Nucleome Positioning System for Spatiotemporal Genome Organization and Regulation". The MSO algorithm development was supported by the Government of India [CSIR-HRDG, BT/PR16356/BID/7/596/2016]. The funding body plays no role in the design of the study and collection, analysis, and interpretation of data and in writing the manuscript.

Availability of data and materials

Project name: PartSeg Project home page: <https://4dnucleome.cent.uw.edu.pl/PartSeg/>, <https://github.com/4DNucleome/PartSeg> Operatseg_imging system(s): Platform independent Programming language: Python, C++ Other requirements: Python 3.6+ License: BSD (with GPL dependencies) Any restrictions to use by non-academics: No. Tutorials and documentation: <http://4dnucleome.cent.uw.edu.pl/PartSeg/>. Fluorescence images of mouse neuroblastoma and rat hippocampal neurons were acquired using Zeiss LSM800 Airyscan or LSM780 Microscope (Zeiss, Germany) equipped with 63x PlanApo oil immersion objective (NA 1.4). Pixel size was set according to the Nyquist criterion, to 70 nm in xy-direction and 210 nm in the z-direction. All images were collected in the Nencki Institute of Experimental Biology to show Partseg utilities and for training purposes of the Partseg tutorial.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

Dariusz Plewczyński is Associate Editor of BMC Bioinformatics Journal. The other authors declare that they have no competing interests.

Author details

¹ Laboratory of Functional and Structural Genomics, Centre of New Technologies, University of Warsaw, Warsaw, Poland. ² Institute of Informatics, University of Warsaw, Warsaw, Poland. ³ Department of Computer Science and Engineering, Jadavpur University, Kolkata, India. ⁴ Nencki Institute of Experimental Biology, Polish Academy of Sciences, Warsaw, Poland. ⁵ Laboratory of Bioinformatics and Computational Genomics, Faculty of Mathematics and Information Science, Warsaw University of Technology, Warsaw, Poland.

Received: 16 July 2020 Accepted: 27 January 2021

Published online: 17 February 2021

References

- Peddie CJ, Collinson LM. Exploring the third dimension: volume electron microscopy comes of age. *Micron*. 2014;61:9–19.
- Pegoraro G, Misteli T. High-throughput imaging for the discovery of cellular mechanisms of disease. *Trends Genet*. 2017;33(9):604–15. <https://doi.org/10.1016/j.tig.2017.06.005>.
- Woodcock CL. Chromatin architecture. *Curr Opin Struct Biol*. 2006;16(2):213–20. <https://doi.org/10.1016/j.sbi.2006.02.005>.
- Fraser P, Bickmore W. Nuclear organization of the genome and the potential for gene regulation. *Nature*. 2007;447(7143):413–7. <https://doi.org/10.1038/nature05916>.
- Dix A, Finlay J, Abowd GD, Beale R. Human computer interaction. 3rd ed. Harlow, England: Pearson Prentice Hall; 2003.
- Rueden CT, Schindelin J, Hiner MC, DeZonia BE, Walter AE, Arena ET, Eliceiri KW. Image J2: ImageJ for the next generation of scientific image data. *BMC Bioinform*. 2017;18(1):529. <https://doi.org/10.1186/s12859-017-1934-z>.
- De Chaumont F, Dallongeville S, Chenouard N, Hervé N, Pop S, Provoost T, Meas-Yedid V, Pankajakshan P, Lecomte T, Le Montagner Y, Lagache T, Dufour A, Olivo-Marin JC. Icy: an open bioimage informatics platform for extended reproducible research. *Nat Methods*. 2012;9(7):690–6. <https://doi.org/10.1038/nmeth.2075>.
- Carpenter AE, Jones TR, Lamprecht MR, Clarke C, Kang IH, Friman O, Guertin DA, Chang JH, Lindquist RA, Moffat J, Golland P, Sabatini DM. Cell profiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biol*. 2006;7(10):100. <https://doi.org/10.1186/gb-2006-7-10-r100>.

9. Wang A, Yan X, Wei Z. ImagePy: an open-source, Python-based and platform-independent software package for bioimage analysis. *Bioinformatics*. 2018;34(18):3238–40. <https://doi.org/10.1093/bioinformatics/bty313>.
10. Liao P-S, Chen T-S, Chung P-C. A fast algorithm for multilevel thresholding. *J Inf Sci Eng*. 2001;17:713–27.
11. Soille P, Vincent LM. Determining watersheds in digital pictures via flooding simulations. In: Kunt M, editor. *Visual communications and image processing '90: fifth in a series*. SPIE. 1990. <https://doi.org/10.1117/12.24211>.
12. Roerdink JBTM, Arnold M. The watershed transform: definitions, algorithms and parallelization strategies. *Fundam Inform*. 2000;41(12):187–228. <https://doi.org/10.3233/FI-2000-411207>.
13. Saha PK, Basu S, Hoffman EA. Multiscale opening of conjoined fuzzy objects: theory and applications. *IEEE Trans Fuzzy Syst*. 2016;24(5):1121–33.
14. Walczak A, Szczepankiewicz AA, Ruszczycki B, Magalska A, Zamlynska K, Dzwonek J, Wilczek E, Zybura-Broda K, Rylski M, Malinowska M, et al. Novel higher-order epigenetic regulation of the *bdnf* gene upon seizures. *J Neurosci*. 2013;33(6):2507–11.
15. Tanabe H, Müller S, Neusser M, von Hase J, Calcagno E, Cremer M, Solovei I, Cremer C, Cremer T. Evolutionary conservation of chromosome territory arrangements in cell nuclei from higher primates. *Proc Natl Acad Sci*. 2002;99(7):4424–9. <https://doi.org/10.1073/pnas.072618599>.
16. Schmidt U, Weigert M, Broaddus C, Myers G. Cell detection with star-convex polygons. In: *Medical image computing and computer assisted intervention—MICCAI 2018—21st international conference, Granada, Spain, September 16–20, 2018, proceedings, part II, 2018*; p. 265–273. https://doi.org/10.1007/978-3-030-00934-2_30.
17. Weigert M, Schmidt U, Haase R, Sugawara K, Myers G. Star-convex polyhedra for 3d object detection and segmentation in microscopy. In: *The IEEE winter conference on applications of computer vision (WACV) 2020*.
18. Stringer C, Michaelos M, Pachitariu M. Cellpose: a generalist algorithm for cellular segmentation. *bioRxiv* 2020. <https://doi.org/10.1101/2020.02.02.931238>. <https://www.biorxiv.org/content/early/2020/02/03/2020.02.02.931238.full.pdf>.
19. Sofroniew N, Lambert T, Evans K, Nunez-Iglesias J, Yamauchi K, Solak AC, Bokota G, ziyangzi Buckley, G, Winston P, Tung T, Pop DD, Hector Freeman J, Bussonnier M, Boone P, Royer L, Har-Gil H, Axelrod S, Rokem A, Bryant Kiggins J, Huang M, Vemuri P, Dunham R, Manz T, jakirkham Wood C, de Siqueira A, Chopra B. Napari/napari: 0.3.8. <https://doi.org/10.5281/zenodo.4048613>. <https://doi.org/10.5281/zenodo.4048613>.
20. Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, et al. Array programming with numpy. *Nature*. 2020;585(7825):357–62. <https://doi.org/10.1038/s41586-020-2649-2>.
21. Yaniv Z, Lowekamp BC, Johnson HJ, Beare R. Simpleitk image-analysis notebooks: a collaborative environment for education and reproducible research. *J Digit Imaging*. 2018;31(3):290–303. <https://doi.org/10.1007/s10278-017-0037-8>.
22. Iannuccelli E, Mompert F, Gellin J, Lahbib-Mansais Y, Yerle M, Boudier T. Nemo: a tool for analyzing gene and chromosome territory distributions from 3d-fish experiments. *Bioinformatics*. 2010;26(5):696–7. <https://doi.org/10.1093/bioinformatics/btq013>.
23. Ollion J, Cochenne C, Loll F, Escudé C, Boudier T. Tango: a generic tool for high-throughput 3d image analysis for studying nuclear organization. *Bioinformatics*. 2013;29(14):1840–1. <https://doi.org/10.1093/bioinformatics/btt276>.
24. Chessel A. An overview of data science uses in bioimage informatics. *Methods*. 2017;115:110–8. <https://doi.org/10.1016/j.jmeth.2016.12.014>.
25. van Steensel B, Belmont AS. Lamina-associated domains: links with chromosome architecture, heterochromatin, and gene repression. *Cell*. 2017;169(5):780–91. <https://doi.org/10.1016/j.cell.2017.04.022>.
26. Bickmore WA. The spatial organization of the human genome. *Annu Rev Genom Hum Genet*. 2013;14(1):67–84. <https://doi.org/10.1146/annurev-genom-091212-153515>.
27. Tang Z, Luo OJ, Li X, Zheng M, Zhu JJ, Szalaj P, Trzaskoma P, Magalska A, Włodarczyk J, Ruszczycki B, et al. CTCF-mediated human 3D genome architecture reveals chromatin topology for transcription. *Cell*. 2015;163(7):1611–27.
28. Sun X, Lin Y. Npas4: linking neuronal activity to memory. *Trends Neurosci*. 2016;39(4):264–75. <https://doi.org/10.1016/j.tins.2016.02.003>.
29. Ravi M, Ramanathan S, Krishna K. Factors, mechanisms and implications of chromatin condensation and chromosomal structural maintenance through the cell cycle. *J Cell Physiol*. 2020;235(2):758–75. <https://doi.org/10.1002/jcp.29038>.
30. Illner D, Zinner R, Handtke V, Rouquette J, Strickfaden H, Lanctôt C, Conrad M, Seiler A, Imhof A, Cremer T, Cremer M. Remodeling of nuclear architecture by the thiodioxopiperazine metabolite chaetocin. *Exp Cell Res*. 2010;316(10):1662–80. <https://doi.org/10.1016/j.yexcr.2010.03.008>.
31. Lallemand-Breitenbach V, de Thé H. Pml nuclear bodies: from architecture to function. *Curr Opin Cell Biol*. 2018;52:154–61. <https://doi.org/10.1016/j.ceb.2018.03.011>.
32. Cuevas V, Dey S, Köhler S, Riddle S, Ludäscher B. Scientific workflows and provenance: introduction and research opportunities. *Datenbank-Spektrum*. 2013. <https://doi.org/10.1007/s13222-012-0100-z>.