A Comparison of Lossless Compression Methods in Microscopy Data Storage Applications

Logan A. Walker
loganaw@umich.edu
University of Michigan
Ann Arbor, Michigan, USA

Ye Li
cnliye@umich.edu
University of Michigan
Ann Arbor, Michigan, USA

Maggie McGlothlin
mamcglot@umich.edu
University of Michigan
Ann Arbor, Michigan, USA

Dawen Cai
dwcai@umich.edu
University of Michigan
Ann Arbor, Michigan, USA

ABSTRACT

Modern high-throughput microscopy methods such as light-sheet imaging and electron microscopy are capable of producing petabytes of data inside of a single experiment. Storage of these large images, however, is challenging because of the difficulty of moving, storing, and analyzing such vast amounts of data, which is often collected at very high data rates (>1GBps). In this report, we provide a comparison of the performance of several compression algorithms using a collection of published and unpublished datasets including confocal, fMOST, and pathology images. We also use simulated data to demonstrate the efficiency of each algorithm as image content or entropy increases. As a result of this work, we recommend the use of the `BLOSC` algorithm combined with `ZSTD` for various microscopy applications, as it produces the best compression ratio over a collection of conditions.

CCS CONCEPTS

• **Applied computing** → **Bioinformatics**; **Imaging**.

KEYWORDS

microscopy, data management, compression

1 INTRODUCTION

The biomedical sciences have been fundamentally altered by the introduction of ever-improving imaging techniques, such as confocal [26, 27], light-sheet [3, 10, 21, 24], super-resolution microscopy [11, 18, 23], and expansion microscopy [20, 22]. As microscopes become faster and higher resolution, their utility have increased. Some of the applications of these methods in neuroscience include multi-spectral imaging of Brainbow samples to identify neuronal connections [5, 20], fMOST imaging for whole-brain neuron reconstruction [4, 17], and single molecule imaging techniques for spatial transcriptomics, which allow scientists to identify spatial profiles of RNA expression [14]. As experiments become more content-rich and data sizes increase, it is increasingly difficult to handle the data, which may can contain many single images reaching >10TB [9, 26].

To deal with data of this scale, most microscopy images are stored in some form of compression, however, there is little consensus on the best way to perform compression to optimize both the compression ratio and the data read/write speed. For archival and distribution, it is not uncommon to see data distributed as compressed TIFF files which have been processed using `GZIP` or `ZIP`, because these tools are commonly-available on most desktop computer and integrated with various image visualization tools. In other cases, matrix storage libraries like Zarr [15] or HDF5 [8, 12] are used to store images in a chunked format which can then be compressed using a variety of methods. Despite this variability, there have been few efforts to benchmark the best compression practices in microscopy data storage. In one recent example, Datta and colleagues showed that different compression algorithms for storage of reduced-representation electron microscopy can have significantly different performance [7], showing the importance of this type of comparison for designing experimental protocols.

Here, we will perform benchmarking of many of the most common lossless image compression algorithms, including `GZIP`, `LZ4`, `BLOSC`, and `ZSTD` using a variety of real-world and simulated biomedical images. From these results we provide a recommendation for compression of biomedical images.

2 DATASETS

2.1 Imaging data

We collected 9 different datasets from a variety of sources and imaging modalities for benchmarking, including published and unpublished data (Table 1). As specified in the table, each dataset were assigned a label to make their referencing in the text simpler and the collection of datasets spans many orders of magnitude in size. The published datasets include traditional confocal imaging of Brainbow-labeled neurological tissue (`brainbow`) [5, 17], fMOST

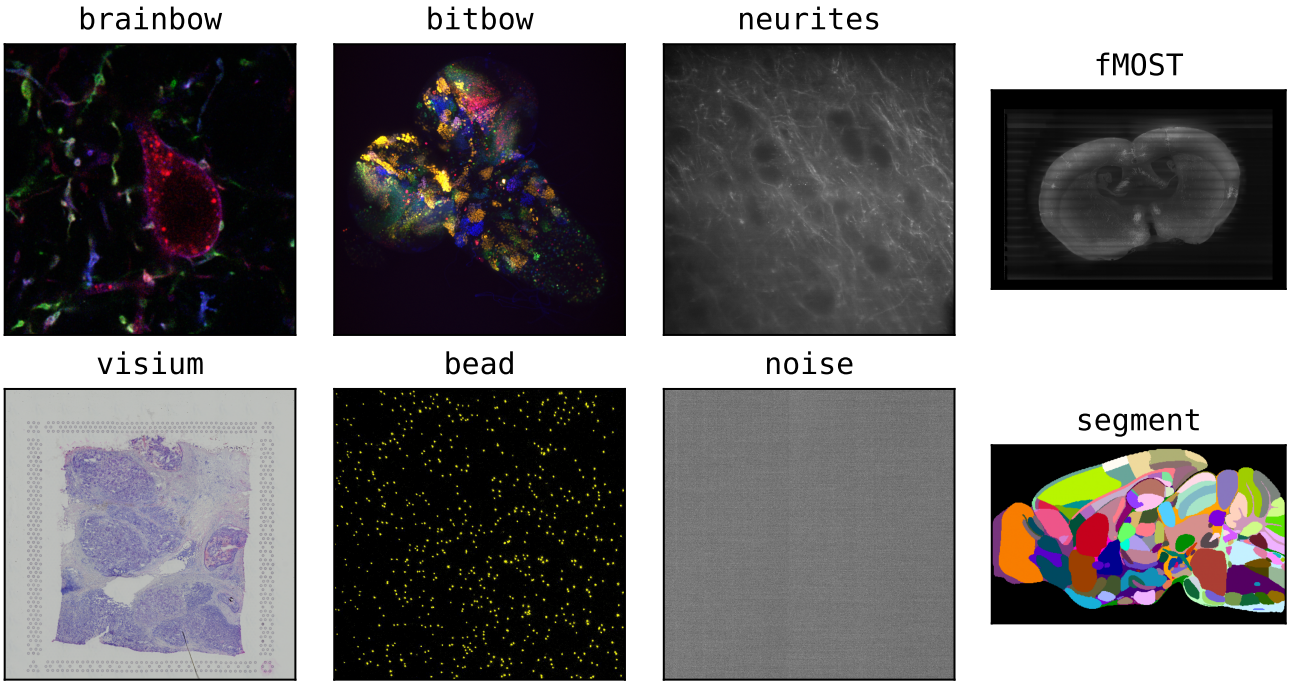

Figure 1: Example z-axis slices from each dataset presented in Table 1. In some images, maximum projections have been performed to increase the visibility of the image in print.

imaging of an entire mouse brain (`fMOST`) [26], and breast cancer tissue pathology images collected as part of a 10× visium experiment (`visium`). In addition to this, we collected confocal imaging of DiD-dyed neurites from mouse cortical tissue (`neurites`), 5-channel *Drosophila melanogaster* (fruit fly) Bitbow images as previously described in [13] (`bitbow`), and fluorescent latex beads (`bead`). We also include an example segmentations from the mouse common coordinate framework (CCF; `segment`) [25], which is a categorical image representing the locations of different brain regions; this is a meaningful comparison because categorical segmentation images are commonly used to identify structures in different micrographs (e.g., [6]). Negative control images of background noise collected from an ORCA-Fusion sCMOS camera (Hammatsu Photonics K.K.) are included at both 8 and 16 bit (`noise8`,16), because they represent the lowest-content image that can be produced by a typical microscopy system. Figure 1 displays example frames from each image.

2.2 Simulated Data

In addition to the real datasets presented above, we used custom Python code to generate *in silico* confocal images of simulated point sources (beads), randomly distributed throughout the frame (see **Methods**). In total, we created 27 images of $(256\ \text{px})^3$ at 9 logarithmically-spaced bead densities which were used for titration experiments at a signal-to-noise-ratio (SNR) of 5.

3 METHODS

3.1 Compute Environment

Dataset compression tests (**Section 4.1**) were performed on a custom-built server with two EPYC 7351 16 core processors and 512GB of DDR4 memory running at 2133 MT/s. This system was chosen because it is representative of real-world situations, where high performance computing is needed to handle data of this scale. Simulation results (**Section 4.2**) were performed on a MacBook Pro with a M1 Pro processor and 16GB of LPDDR5 memory at 6400 MT/s, because of it's higher speed than the sever described above. We forced all benchmarks to run on a single core, because some compression methods are unable to be multithreaded, but we note that,

Label	Size (px)	Type	Source
`brainbow`	$4 \times 136 \times 512 \times 512$	`uint16`	[17]
`neurites`	$1 \times 1000 \times 2304 \times 2304$	`uint16`	(this work)
`bitbow`	$5 \times 97 \times 3000 \times 3000$	`uint16`	(this work)
`fMOST`	$2 \times 11464 \times 20821 \times 30801$	`uint16`	[16]
`visium`	$3 \times 1 \times 24240 \times 24240$	`uint8`	10xgenomics.com
`bead`	$1 \times 1000 \times 2304 \times 2304$	`uint16`	(this work)
`noise16`	$1 \times 100 \times 1000 \times 1000$	`uint16`	(this work)
`noise8`	$1 \times 100 \times 1000 \times 1000$	`uint8`	(this work)
`segment`	$1 \times 456 \times 528 \times 320$	`uint32`	[25]

Table 1: A list of the datasets collected for this study, with associated metadata. Image size is given in $c \times z \times x \times y$.

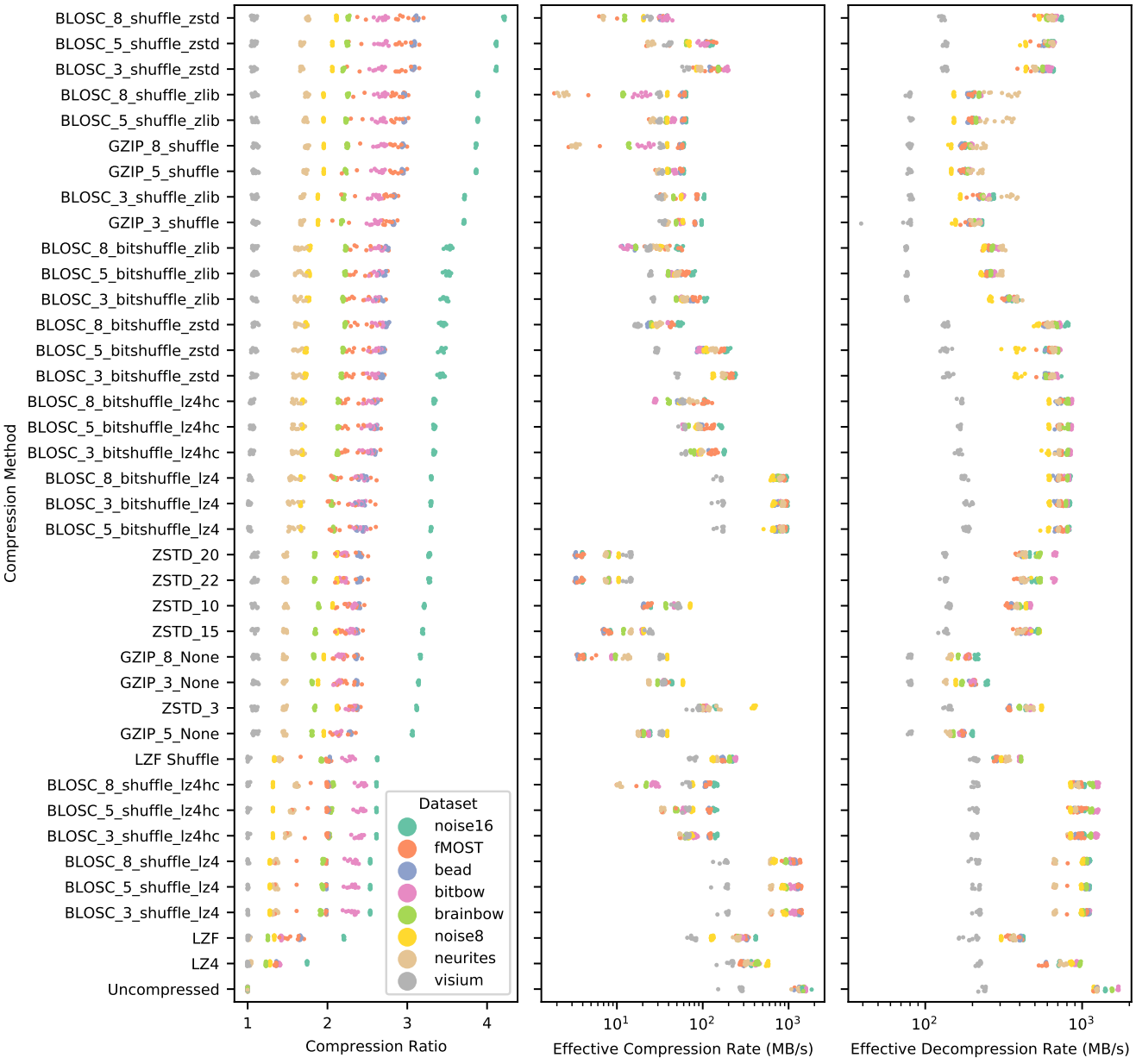

Figure 2: A strip plot of the compression ratio achieved by different compression algorithms when applied to our benchmark datasets. Methods are sorted by the maximum compression rate achieved by any dataset for visualization.

in practice, many datasets can be compressed in multiple streams to achieve higher throughput than we present here. All files were saved into a Python memory buffers rather than a hard drive in order to ensure that disk latency or bandwidth did not impact benchmark times. As such, we use a parameter set labeled "Uncompressed" as a control for the throughput of the testing scripts themselves. We used Anaconda Python (v3.10) to implement our testing scripts, as it provides a high-level programming interface, while having a minimal impact on performance because the libraries used in this study are implemented as wrappers around the low-level compression libraries.

Compression methods tested included `LZ4`, `LZF`, `GZIP`, `ZSTD`, and `BLOSC` [1]. For methods which accept compression level parameters, we tested a range of parameters between the minimum and maximum. BLOSC uses a "meta-compression" and blocking strategy which allows the use of multiple compressors as well as filters, such as `bitshuffle` and `shuffle`. These images transpose the data at the bit or byte level in order to align similarly-value bits in

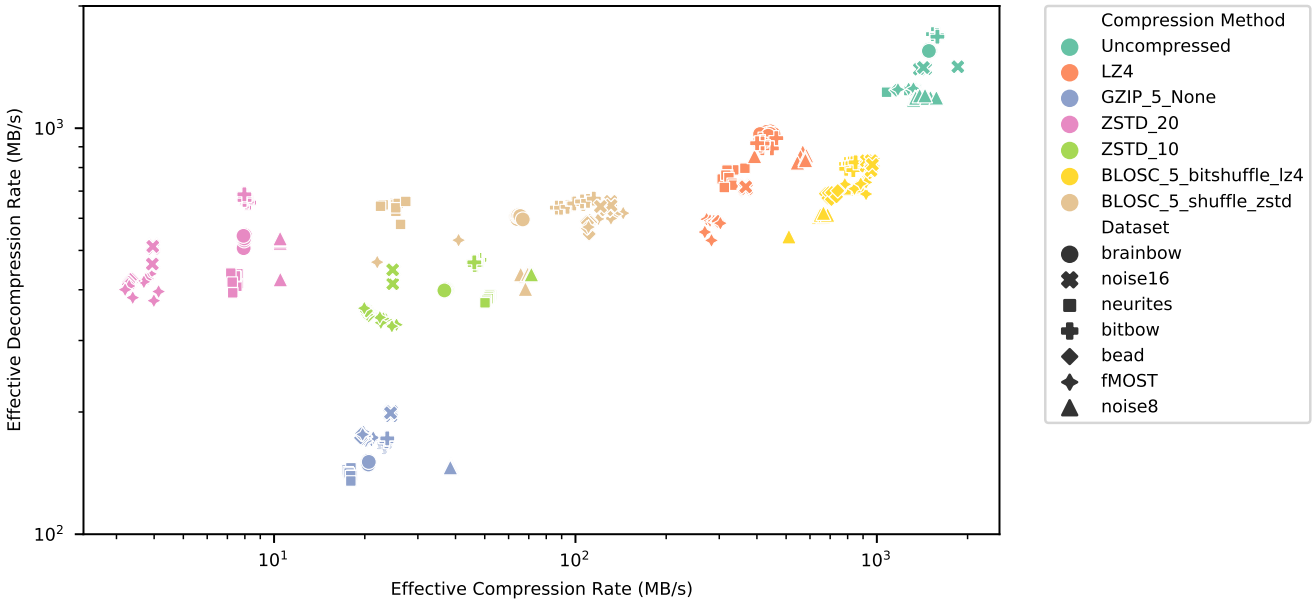

Figure 3: A scatter plot of the compression compression and decompression rates for the different compression algorithms tested.

integer data, and require a negligible amount of computation to perform. We excluded comparisons without BLOSC shuffling filters, because we found that they consistently performed worse than the shuffled equivalent (data not shown). In the rest of this report, we will describe each compression method using the combination of parameters used for each comparison.

3.2 Simulated data generation

We used custom Python scripts to generate simulated confocal images of varying complexity using the `python-sdt` simulation package [19]. Specifically, random 3D bead coordinates were generated for each specified density, which were then modeled as point sources with an intensity of 500 au and standard deviations of 2 px. Each 2D frame's coordinates were then calculated by slicing the 3D bead coordinates into individual planes assuming a Gaussian intensity distribution for each point. The image was then offset by 100 au and noise was added by sampling with the Poisson distribution with λ equal to the simulated image intensity:

$$P(v)|_\lambda = \frac{\lambda^v e^{-\lambda}}{v!} \tag{1}$$

Each 2D frame is then stacked to form composite 3D images which were exported for benchmarking.

3.3 Compression comparison

Each dataset was sampled into 10 independent $256 \times 256 \times 256$ px chunks across all channels in $x \times y \times z$. Each sampled image was compressed into HDF5 files with a chunk size of $32 \times 32 \times 32$ px using default parameters, with the exception of the compression settings which were changed for each experiment. We quantify compression and decompression rates as an "Effective" throughput, which is the size of uncompressed data divided by the total time to open, parse, and decompress an image into RAM. This allows us to quantify the speed that a given application will be able to actually access data, rather than just the algorithm speed. The chunk size was reduced for the `visium` dataset, because the image is a single z plane. We used compression filters for HDF5 which were included in the base Python HDF5 (`h5py`[1]) library or the `hdf5plugin`[2] `pip` package, which extends the number of available compression plugins available.

4 RESULTS AND DISCUSSION

4.1 Compression Benchmark

The compression ratios (c.r.), decompression, and compression speed of each compression experiment are shown in Figure 2. For visualization, we removed the `segment` layer because it had very high compression rates which were not representative of the other datasets (c.r. range 11.3-96.6) with ZSTD-based approaches performing the best. First, we observe `noise16` consistently returns the largest compression ratio, while `noise8` performs poorly due to it's much higher relative entropy. `visium` also performs poorly, likely due to the 2D nature of the dataset causing smaller chunking sizes. In this sorted figure, we see that the shuffle and bitshuffle filters are essential for achieving the highest compression ratios in integer data. `BLOSC-ZSTD-SHUFFLE-8` is found to be the highest compression ratio by a very small margin, but we recommend the use of `BLOSC-ZSTD-SHUFFLE-5` when high compression ratios are needed because the higher compression speed found is much higher.

[1] https://www.h5py.org/
[2] https://pypi.org/project/hdf5plugin/

Next, we compared the compression and decompression speed of a subset of compression methods which were selected to represent the full dynamic range of speed and compression ratios (Figure 3). Here, we can see that the algorithms using LZ4 are universally the fastest methods, with LZ4 and BLOSC-LZ4-BITSHUFFLE-5 performing only slightly slower than a the uncompressed control. We highlight these methods as a good candidates for real-time data processing, such as intermediate compression of data being produced by a microscope due to these results, with the BLOSC variant being prefered due to the higher compression ratio (Figure 2). The BLOSC-ZSTD-SHUFFLE-5 method identified above is able to be decompressed quickly, however it's compression rates are slow enough to be a limiting for real-time applications. GZIP is too slow to be usable by in any real-time microscopy application.

4.2 Simulated Confocal Imaging Benchmark

We created simulated micrographs with varying densities of fluorescent beads, similar to the beads dataset and with simulated image noise (Figure 4, left). We compared the compression ratios of BLOSC-ZSTD-BITSHUFFLE-5 with a commonly-used standard compression scheme represented by GZIP-5. Here, we find that BLOSC performs significantly better, however there is a reduction in the compression ratio as the bead count increases. This matches our expectation, as increased bead density is directly correlated with an increase in image entropy, making the theoretical compression limit lower.

Next, to demonstrate the effectiveness of simple data manipulation for improving we performed background subtraction above the approximate noise floor of the images described above (120au; Figure 4, right). We repeated the compression testing, finding that there is an order-of-magnitude increase in the achieved compression ratio, with GZIP performing better than BLOSC in this case. While this transform is lossy, it highlights the potential value of preprocessing microscopy data before storage, as well as the potential for other lossy compression methods.

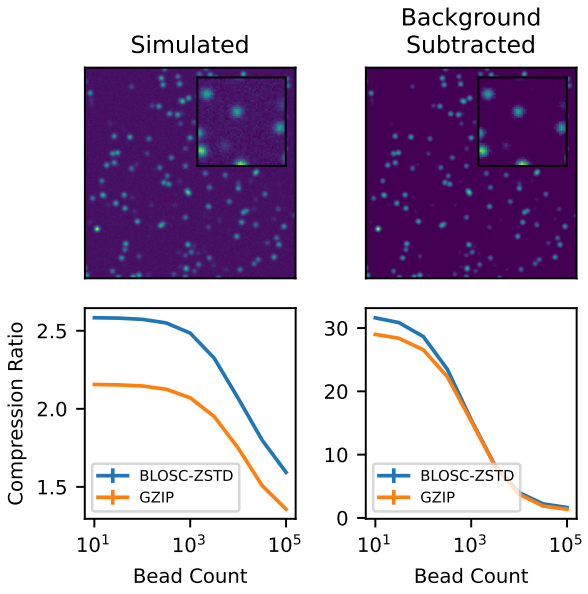

Figure 4: Performance of BLOSC-ZSTD and GZIP compression at compressing simulated confocal data.

5 CONCLUSION AND FUTURE DIRECTION

Compression is incredibly important for making biomedical imaging both economically and, in many cases, technically feasible by reducing the hard disk space needed for a given experiment. We have shown benchmarks of 39 different compression parameter combinations applied to 9 different biomedical image datasets derived from different neurological tissues and imaging methods, as well as simulated data. Through this quantification, excluding segment and visium due to their outlier characteristics, we find that BLOSC-ZSTD-SHUFFLE-5 performs the best of all methods compared (c.r. 2.7 ± 0.8), however, at the expense of slow compression (86 ± 36 MB/s). In cases where fast or real-time computation is needed, BLOSC-LZ4-BITSHUFFLE-8 provides a good compromise (c.r. 2.3 ± 0.5) while being fast to compress (790 ± 92 MB/s). These results provide an important starting point for choosing a compression method for any specific experiment, and we provide our code under an GPL license to allow testing on different imaging types or CPU architectures, as we expect results may vary.

While we presented a comprehensive view of the applications of lossless compression in microscopy, there are several factors which should be considered as future directions or factors when deciding which compression method to use for a specific microscopy application. First, NVIDIA has recently developed toolkits for lossless compression which take advantage of GPU compression, potentially making much higher throughputs possible by application of these tools[3]. Next, while we used a chunk size of $(32\text{px})^3$ in this study because it allows quick access in all dimensions, it is possible that different datasets may have different spatial correlations which make increased chunk sizes more efficient. Finally, as shown in Section 4.2, image filtering and preprocessing has a strong potential for improving the efficiency of data compression, assuming you can make guarantees about the scientific validity of the decompressed data. In the future, various lossy compression methods may be able to fill this gap, similar to the previously-reported B^3D method [2].

DATA AND CODE AVAILABILITY

The code used in this article is available from the Cai Lab GitHub page under a GPL license[4]. The novel images presented in this comparison have been uploaded to the University of Michigan Deep Blue Data repository (DOI pending) and are freely available under a Creative Commons Attribution-ShareAlike 4.0 International (CC BY-SA 4.0) license[5].

AUTHOR CONTRIBUTIONS

LAW and DC conceived this project. LAW wrote associated analysis code and first draft of the manuscript. LAW, YL, and MM performed biological sample preparation and microscopy for novel datasets

[3] https://github.com/NVIDIA/nvcomp
[4] https://github.com/Cai-Lab-at-University-of-Michigan
[5] https://creativecommons.org/licenses/by-sa/4.0/

used in analysis. All authors edited and approved the final manuscript.

ACKNOWLEDGMENTS

LAW was supported by a University of Michigan Rackham Predoctoral Fellowship. This work was supported by NIH-RF1MH123402, and NIH-RF1MH124611 to DC.

REFERENCES

[1] Francesc Alted. 2010. Why Modern CPUs Are Starving and What Can Be Done about It. *Comput. Sci. Eng.* 12, 2 (March 2010), 68–71. https://doi.org/10.1109/MCSE.2010.51

[2] Bálint Balázs, Joran Deschamps, Marvin Albert, Jonas Ries, and Lars Hufnagel. 2017. A real-time compression library for microscopy images. (July 2017), 164624 pages. https://doi.org/10.1101/164624

[3] Matthew B Bouchard, Venkatakaushik Voleti, César S Mendes, Clay Lacefield, Wesley B Grueber, Richard S Mann, Randy M Bruno, and Elizabeth M C Hillman. 2015. Swept confocally-aligned planar excitation (SCAPE) microscopy for high speed volumetric imaging of behaving organisms. *Nat. Photonics* 9, 2 (Feb. 2015), 113–119. https://doi.org/10.1038/nphoton.2014.323

[4] BRAIN Initiative Cell Census Network (BICCN). 2021. A multimodal cell census and atlas of the mammalian primary motor cortex. *Nature* 598, 7879 (Oct. 2021), 86–102. https://doi.org/10.1038/s41586-021-03950-0

[5] Dawen Cai, Kimberly B Cohen, Tuanlian Luo, Jeff W Lichtman, and Joshua R Sanes. 2013. Improved tools for the Brainbow toolbox. *Nat. Methods* 10, 6 (May 2013), 540–547. https://doi.org/10.1038/nmeth.2450

[6] Anne E Carpenter, Thouis R Jones, Michael R Lamprecht, Colin Clarke, In Han Kang, Ola Friman, David A Guertin, Joo Han Chang, Robert A Lindquist, Jason Moffat, Polina Golland, and David M Sabatini. 2006. CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biol.* 7, 10 (Oct. 2006), R100. https://doi.org/10.1186/gb-2006-7-10-r100

[7] Abhik Datta, Kian Fong Ng, Deepan Balakrishnan, Melissa Ding, See Wee Chee, Yvonne Ban, Jian Shi, and N Duane Loh. 2021. A data reduction and compression description for high throughput time-resolved electron microscopy. *Nat. Commun.* 12, 1 (Jan. 2021), 664. https://doi.org/10.1038/s41467-020-20694-z

[8] Matthew T Dougherty, Michael J Folk, Erez Zadok, Herbert J Bernstein, Frances C Bernstein, Kevin W Eliceiri, Werner Benger, and Christoph Best. 2009. Unifying Biological Image Formats with HDF5. *Commun. ACM* 52, 10 (Oct. 2009), 42–47. https://doi.org/10.1145/1562764.1562781

[9] Holly C Gibbs, Sakina M Mota, Nathan A Hart, Sun Won Min, Alex O Vernino, Anna L Pritchard, Anindito Sen, Stan Vitha, Sreeja Sarasamma, Avery L McIntosh, Alvin T Yeh, Arne C Lekven, Dylan A McCreedy, Kristen C Maitland, and Lisa M Perez. 2021. Navigating the Light-Sheet Image Analysis Software Landscape: Concepts for Driving Cohesion From Data Acquisition to Analysis. *Front Cell Dev Biol* 9 (Nov. 2021), 739079. https://doi.org/10.3389/fcell.2021.739079

[10] Adam K Glaser, Kevin W Bishop, Lindsey A Barner, Etsuo A Susaki, Shimpei I Kubota, Gan Gao, Robert B Serafin, Pooja Balaram, Emily Turschak, Philip R Nicovich, Hoyin Lai, Luciano A G Lucas, Yating Yi, Eva K Nichols, Hongyi Huang, Nicholas P Reder, Jasmine J Wilson, Ramya Sivakumar, Elya Shamskhou, Caleb R Stoltzfus, Xing Wei, Andrew K Hempton, Marko Pende, Prayag Murawala, Hans-Ulrich Dodt, Takato Imaizumi, Jay Shendure, Brian J Beliveau, Michael Y Gerner, Li Xin, Hu Zhao, Lawrence D True, R Clay Reid, Jayaram Chandrashekar, Hiroki R Ueda, Karel Svoboda, and Jonathan T C Liu. 2022. A hybrid open-top light-sheet microscope for versatile multi-scale imaging of cleared tissues. *Nat. Methods* 19, 5 (May 2022), 613–619. https://doi.org/10.1038/s41592-022-01468-5

[11] S W Hell and J Wichmann. 1994. Breaking the diffraction resolution limit by stimulated emission: stimulated-emission-depletion fluorescence microscopy. *Opt. Lett.* 19, 11 (June 1994), 780–782. https://doi.org/10.1364/ol.19.000780

[12] Quincey Koziol, Elena Pourmal, and Neil Fortner. 2014. RFC: HDF5 virtual dataset. *Documentation, The HDF5 Group* (2014).

[13] Ye Li, Logan A Walker, Yimeng Zhao, Erica M Edwards, Nigel S Michki, Hon Pong Jimmy Cheng, Marya Ghazzi, Tiffany Y Chen, Maggie Chen, Douglas H Roossien, and Dawen Cai. 2021. Bitbow Enables Highly Efficient Neuronal Lineage Tracing and Morphology Reconstruction in Single Drosophila Brains. *Front. Neural Circuits* 15 (Oct. 2021), 732183. https://doi.org/10.3389/fncir.2021.732183

[14] Jeffrey R Moffitt, Emma Lundberg, and Holger Heyn. 2022. The emerging landscape of spatial profiling technologies. *Nat. Rev. Genet.* (July 2022). https://doi.org/10.1038/s41576-022-00515-3

[15] Josh Moore, Chris Allan, Sébastien Besson, Jean-Marie Burel, Erin Diel, David Gault, Kevin Kozlowski, Dominik Lindner, Melissa Linkert, Trevor Manz, Will Moore, Constantin Pape, Christian Tischer, and Jason R Swedlow. 2021. OME-NGFF: a next-generation file format for expanding bioimaging data-access strategies. *Nat. Methods* 18, 12 (Dec. 2021), 1496–1498. https://doi.org/10.1038/s41592-021-01326-w

[16] Hanchuan Peng, Peng Xie, Lijuan Liu, Xiuli Kuang, Yimin Wang, Lei Qu, Hui Gong, Shengdian Jiang, Anan Li, Zongcai Ruan, Liya Ding, Zizhen Yao, Chao Chen, Mengya Chen, Tanya L Daigle, Rachel Dalley, Zhangcan Ding, Yanjun Duan, Aaron Feiner, Ping He, Chris Hill, Karla E Hirokawa, Guodong Hong, Lei Huang, Sara Kebede, Hsien-Chi Kuo, Rachael Larsen, Phil Lesnar, Longfei Li, Qi Li, Xiangning Li, Yaoyao Li, Yuanyuan Li, An Liu, Donghuan Lu, Stephanie Mok, Lydia Ng, Thuc Nghi Nguyen, Qiang Ouyang, Jintao Pan, Elise Shen, Yuanyuan Song, Susan M Sunkin, Bosiljka Tasic, Matthew B Veldman, Wayne Wakeman, Wan Wan, Peng Wang, Quanxin Wang, Tao Wang, Yaping Wang, Feng Xiong, Wei Xiong, Wenjie Xu, Min Ye, Lulu Yin, Yang Yu, Jia Yuan, Jing Yuan, Zhixi Yun, Shaoqun Zeng, Shichen Zhang, Sujun Zhao, Zijun Zhao, Zhi Zhou, Z Josh Huang, Luke Esposito, Michael J Hawrylycz, Staci A Sorensen, X William Yang, Yefeng Zheng, Zhongze Gu, Wei Xie, Christof Koch, Qingming Luo, Julie A Harris, Yun Wang, and Hongkui Zeng. 2021. Morphological diversity of single neurons in molecularly defined cell types. *Nature* 598, 7879 (Oct. 2021), 174–181. https://doi.org/10.1038/s41586-021-03941-1

[17] Douglas H Roossien, Benjamin V Sadis, Yan Yan, John M Webb, Lia Y Min, Aslan S Dizaji, Luke J Bogart, Cristina Mazuski, Robert S Huth, Johanna S Stecher, Sriakhila Akula, Fred Shen, Ye Li, Tingxin Xiao, Madeleine Vandenbrink, Jeff W Lichtman, Takao K Hensch, Erik D Herzog, and Dawen Cai. 2019. Multispectral tracing in densely labeled mouse brain with nTracer. *Bioinformatics* 35, 18 (Sept. 2019), 3544–3546. https://doi.org/10.1093/bioinformatics/btz084

[18] Michael J Rust, Mark Bates, and Xiaowei Zhuang. 2006. Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM). *Nat. Methods* 3, 10 (Oct. 2006), 793–795. https://doi.org/10.1038/nmeth929

[19] Lukas Schrangl. 2022. sdt-python: Python library for fluorescence microscopy data analysis (v17.2). https://doi.org/10.5281/zenodo.6366740

[20] Fred Y Shen, Margaret M Harrington, Logan A Walker, Hon Pong Jimmy Cheng, Edward S Boyden, and Dawen Cai. 2020. Light microscopy based approach for mapping connectivity with molecular specificity. *Nat. Commun.* 11, 1 (Sept. 2020), 4632. https://doi.org/10.1038/s41467-020-18422-8

[21] Ernst H K Stelzer, Frederic Strobl, Bo-Jui Chang, Friedrich Preusser, Stephan Preibisch, Katie McDole, and Reto Fiolka. 2021. Light sheet fluorescence microscopy. *Nature Reviews Methods Primers* 1, 1 (Nov. 2021), 1–25. https://doi.org/10.1038/s43586-021-00069-4

[22] Paul W Tillberg, Fei Chen, Kiryl D Piatkevich, Yongxin Zhao, Chih-Chieh Jay Yu, Brian P English, Linyi Gao, Anthony Martorell, Ho-Jun Suk, Fumiaki Yoshida, Ellen M DeGennaro, Douglas H Roossien, Guanyu Gong, Uthpala Seneviratne, Steven R Tannenbaum, Robert Desimone, Dawen Cai, and Edward S Boyden. 2016. Protein-retention expansion microscopy of cells and tissues labeled using standard fluorescent proteins and antibodies. *Nat. Biotechnol.* 34, 9 (Sept. 2016), 987–992. https://doi.org/10.1038/nbt.3625

[23] Giuseppe Vicidomini, Paolo Bianchini, and Alberto Diaspro. 2018. STED super-resolved microscopy. *Nat. Methods* 15, 3 (March 2018), 173–182. https://doi.org/10.1038/nmeth.4593

[24] Venkatakaushik Voleti, Kripa B Patel, Wenze Li, Citlali Perez Campos, Srinidhi Bharadwaj, Hang Yu, Caitlin Ford, Malte J Casper, Richard Wenwei Yan, Wenxuan Liang, Chentao Wen, Koutarou D Kimura, Kimara L Targoff, and Elizabeth M C Hillman. 2019. Real-time volumetric microscopy of in vivo dynamics and large-scale samples with SCAPE 2.0. *Nat. Methods* 16, 10 (Oct. 2019), 1054–1062. https://doi.org/10.1038/s41592-019-0579-4

[25] Quanxin Wang, Song-Lin Ding, Yang Li, Josh Royall, David Feng, Phil Lesnar, Nile Graddis, Maitham Naeemi, Benjamin Facer, Anh Ho, Tim Dolbeare, Brandon Blanchard, Nick Dee, Wayne Wakeman, Karla E Hirokawa, Aaron Szafer, Susan M Sunkin, Seung Wook Oh, Amy Bernard, John W Phillips, Michael Hawrylycz, Christof Koch, Hongkui Zeng, Julie A Harris, and Lydia Ng. 2020. The Allen Mouse Brain Common Coordinate Framework: A 3D Reference Atlas. *Cell* 181, 4 (May 2020), 936–953.e20. https://doi.org/10.1016/j.cell.2020.04.007

[26] Xiaojun Wang, Hanqing Xiong, Yurong Liu, Tao Yang, Anan Li, Fei Huang, Fangfang Yin, Lei Su, Ling Liu, Ning Li, Longhui Li, Shenghua Cheng, Xiaoxiang Liu, Xiaohua Lv, Xiuli Liu, Jun Chu, Tonghui Xu, Fuqiang Xu, Hui Gong, Qingming Luo, Jing Yuan, and Shaoqun Zeng. 2021. Chemical sectioning fluorescence tomography: high-throughput, high-contrast, multicolor, whole-brain imaging at subcellular resolution. *Cell Rep.* 34, 5 (Feb. 2021), 108709. https://doi.org/10.1016/j.celrep.2021.108709

[27] Robert H Webb. 1999. Confocal optical microscopy. *Rep. Prog. Phys.* 59, 3 (Jan. 1999), 427. https://doi.org/10.1088/0034-4885/59/3/003