The Journal of Open Source Software

Rainbow: Automated Air-Liquid Interface Cell Culture Analysis Using Deep Optical Flow

Alphons Gwatimba[1,2¶], **Joseph Ho**[1,2], **Thomas Iosifidis**[*1,3,4], **and Yuliya V. Karpievitch**[†1,5]

1 Wal-yan Respiratory Research Centre, Telethon Kids Institute, University of Western Australia, Nedlands, 6009, Western Australia, Australia **2** School of Computer Science and Software Engineering, University of Western Australia, Nedlands, 6009, Western Australia, Australia **3** Centre for Cell Therapy and Regenerative Medicine, School of Medicine, University of Western Australia, Nedlands, 6009, Western Australia, Australia **4** School of Population Health, Curtin University, Bentley, 6102, Western Australia, Australia **5** School of Molecular Sciences, University of Western Australia, Nedlands, 6009, Western Australia, Australia ¶ Corresponding author

DOI: 10.21105/joss.04080

Software

- Review
- Repository
- Archive

Editor: Jacob Schreiber

Reviewers:

- @tpeulen
- @assistedevolution

Submitted: 22 November 2021
Published: 22 March 2022

Summary

Rainbow is a free, open-source and cross-platform Python-based tool for Air-Liquid Interface (ALI) cell culture time-lapse image analysis. Rainbow accepts input time-lapse images in standard image formats, such as TIFF, or microscopy file formats, such as ND2, and then computes the optical flow; that is, the apparent motion of individual pixels in an image (Beauchemin & Barron, 1995; Turaga et al., 2010; Zhai et al., 2021), across multiple frames using a deep learning based pretrained optical flow model (Jiang et al., 2021). Rainbow then applies circular data analysis to the pixel-level optical flow information to calculate the average magnitude [0, ∞] (μm) and direction [0, 360) (°) of motion between adjacent frames to quantify cell motility. Additionally, the variance of the magnitude [0, ∞] (μm) and direction [0, ∞] (°) of motion between adjacent frames is calculated to quantitatively capture the degree of heterogeneity in cell motility.

For each experiment, a CSV file with the minimum, maximum, mean, standard deviation, and variance of the magnitude and direction of cell movement between adjacent frames in an image sequence is produced. Multiple CSV files from different experiments can be combined into one file that can be analyzed for differences in cell motility across multiple experiments. Rainbow also includes a high-resolution and easily readable unified hue/saturation-based visualization scheme for the instantaneous vector field of motion between adjacent frames of an image sequence to qualitatively show cell motion. Rainbow can be used through a graphical user interface or command line interface and can generate a HTML report containing output images, videos, publication ready figures, and CSV files detailing cell dynamics (refer to Examples folder on GitHub). Importantly, our software is not limited to ALI culture image analysis, and developers can extend the software's existing pipeline to other use cases. For example, the optical flow model can be readily substituted with different models, as we utilized the Factory Method creational software design pattern. The data analysis and report generated can be adjusted through interactive Jupyter Notebooks, allowing for a flexible and versatile system. Some of Rainbow's visualizations are shown in Figure 1.

*co-senior author
†co-senior author

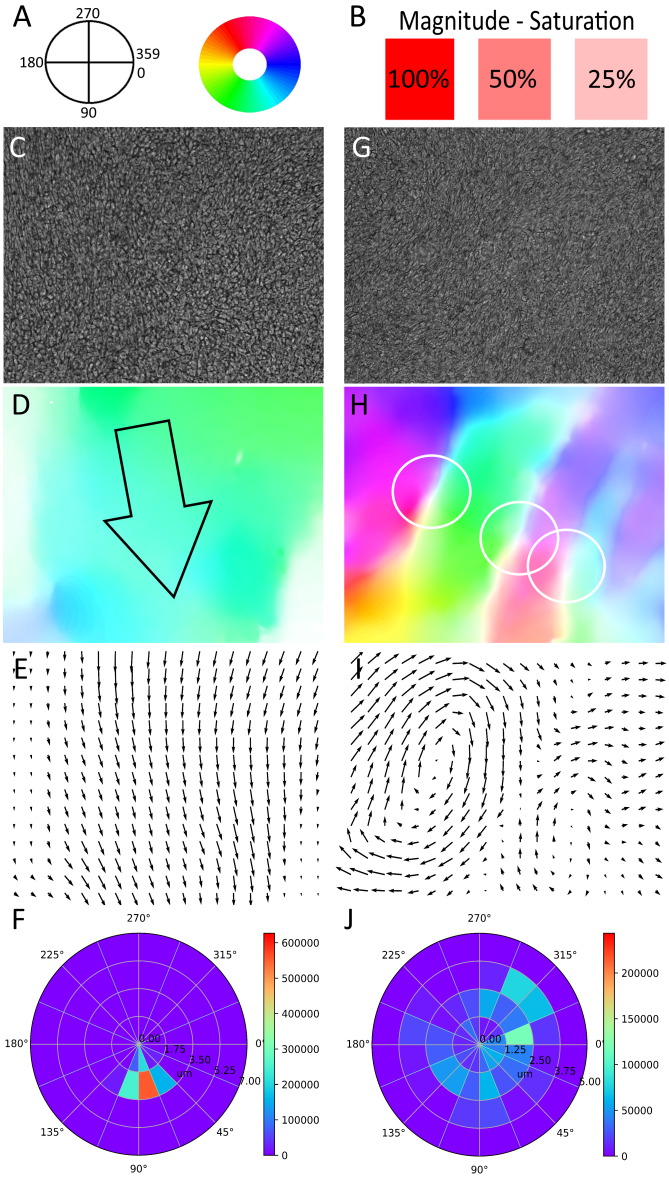

Figure 1: `Rainbow` optical flow visualization. **A:** The direction of motion at any position within `Rainbow`-generated optical flow images is measured clockwise from the initial horizontal position of a unit circle (left) and is shown using hue values (right). **B:** The magnitude of motion at any position within optical flow images is shown using saturation values. High saturation (100%) corresponds to high magnitude of motion and low saturation (25%) corresponds to low magnitude of motion. **C, G:** Still frames taken from two separate ALI culture image sequences. **D, H:** Unified visualization of optical flow magnitude and direction between adjacent frames of two ALI culture image sequences using `Rainbow`. The arrow indicates the average direction of motion across the image sequence. The circles indicate three localized vortexes that the cells move around in a swirl-like motion as they change direction. **E, I:** Traditional visualization of optical flow between adjacent frames of two ALI culture image sequences using quiver plots containing vector arrows at every 70 px. **F, J:** Polar plots visualizing motion magnitude (concentric circles; µm) and direction (azimuthal angle; degrees) in the same frame of two ALI culture image sequences. Colour scale indicates the number of points migrating towards given direction. All left positioned subfigures from row 2 onwards (C-F) correspond to the same ALI culture image sequence while right positioned subfigures correspond to a different image sequence (G-J). For complete insight, refer to Examples folder on GitHub containing videos.

Statement of need

Differentiated primary airway epithelial cells cultured at air-liquid interface (ALI) are commonly used to assess airway epithelial function in vitro in health and disease, such as asthma (Chen & Schoen, 2019; Looi et al., 2018; Martinovich et al., 2017). The integration of image analysis and ALI cell cultures has provided novel insights into cell dynamics, such as the recently identified unjammed-to-jammed transition of AEC characterized by changes in cell motility (Mitchel et al., 2020; Park et al., 2015). In chronic respiratory diseases like asthma, increased cell motility and AEC unjamming have been linked to airway remodeling and disease development (Mitchel et al., 2020; Park et al., 2015). However, the image analyses performed in these studies are limited. For example, handcrafted methods from the MATLAB Computer Vision Toolbox that compute optical flow have been used to extract cell motion information from ALI culture images (Mitchel et al., 2020). This approach requires licenced software, and handcrafted optical flow estimation methods have been outperformed in terms of accuracy by deep learning methods (Savian et al., 2020). Furthermore, commonly used cell motion metrics, such as average cell speed, do not capture all unique aspects of cell motion, such as the heterogeneity of cell migration patterns across time. Cell motion is commonly visualized using vector fields, which are useful but bound by an inverse relationship between resolution and readability (Henkes et al., 2020; Nnetu et al., 2012; O'Sullivan et al., 2020).

To increase understanding of lung disease mechanisms and development of new treatment options for patients, there is a need for open-source solutions for ALI culture image analyses that can be broadly implemented across cell biology laboratories. To the best of our knowledge, `Rainbow` is the first easy-to-use software tool that performs all the above analyses automatically for efficient utilization by non-programmers. `Rainbow` produces automatic cell motion quantifications, figures, and reports that are all easily transferrable into publications. We anticipate that `Rainbow` will provide cell motion characterization for each experiment and allow for easy comparisons among multiple experiments to uncover cell migration mechanisms previously undetermined in health and disease.

Acknowledgements

We would like to thank the contribution and assistance of all the respiratory fellows, anaesthetists, nurses, hospital staff at St John of God Hospital, Subiaco (Human Research Ethics Committee study approval #901), and Western Australian Epithelial Research Program (WAERP) members. We would also like to thank the families and children participating in this project. This work was supported by the Wal-Yan Respiratory Research Centre Inspiration Award, Cystic Fibrosis Charitable Endowment Charles Bateman Charitable Trust, Western Australian Department of Health Merit Award, and BHP-Telethon Kids Blue Sky Award. Furthermore, this project relies on high quality open source Python packages: NumPy (Harris et al., 2020), matplotlib (Hunter, 2007), pandas (Wes McKinney, 2010), PyYaml, ND2Reader, Gooey, Physt, imutils, MoviePy, natsort, PIMS, tqdm (Costa-Luis et al., 2021), pytest (Krekel et al., 2004), FFmpeg (Tomar, 2006), OpenCV (Bradski, 2000), Jupyter Notebook (Kluyver et al., 2016), Astropy (Astropy Collaboration et al., 2018, 2013), PyTorch (Paszke et al., 2019), and scikit-image (Walt et al., 2014).

References

Astropy Collaboration, Price-Whelan, A. M., Sipőcz, B. M., Günther, H. M., Lim, P. L., Crawford, S. M., Conseil, S., Shupe, D. L., Craig, M. W., Dencheva, N., Ginsburg, A., Vand erPlas, J. T., Bradley, L. D., Pérez-Suárez, D., Val-Borro, M. de, Aldcroft, T. L., Cruz, K. L., Robitaille, T. P., Tollerud, E. J., ... Astropy Contributors. (2018). *The astropy project: Building an open-science project and status of the v2.0 core package*. *156*(3), 123. https://doi.org/10.3847/1538-3881/aabc4f

Astropy Collaboration, Robitaille, T. P., Tollerud, E. J., Greenfield, P., Droettboom, M., Bray, E., Aldcroft, T., Davis, M., Ginsburg, A., Price-Whelan, A. M., Kerzendorf, W. E., Conley, A., Crighton, N., Barbary, K., Muna, D., Ferguson, H., Grollier, F., Parikh, M. M., Nair, P. H., … Streicher, O. (2013). *Astropy: A community Python package for astronomy*. *558*, A33. https://doi.org/10.1051/0004-6361/201322068

Beauchemin, S. S., & Barron, J. L. (1995). The computation of optical flow. *ACM Computing Surveys (CSUR)*, *27*(3), 433–466.

Bradski, G. (2000). The OpenCV library. *Dr. Dobb's Journal of Software Tools*.

Chen, S., & Schoen, J. (2019). Air-liquid interface cell culture: From airway epithelium to the female reproductive tract. *Reproduction in Domestic Animals*, *54*(S3), 38–45. https://doi.org/10.1111/rda.13481

Costa-Luis, C. da, Larroque, S. K., Altendorf, K., Mary, H., richardsheridan, Korobov, M., Yorav-Raphael, N., Ivanov, I., Bargull, M., Rodrigues, N., CHEN, G., Lee, A., Newey, C., James, Coales, J., Zugnoni, M., Pagel, M. D., mjstevens777, Dektyarev, M., … Nordlund, M. (2021). *Tqdm: A fast, extensible progress bar for python and CLI*. Zenodo. https://doi.org/10.5281/zenodo.5517697

Harris, C. R., Millman, K. J., Walt, S. J. van der, Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., Kerkwijk, M. H. van, Brett, M., Haldane, A., Río, J. F. del, Wiebe, M., Peterson, P., … Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Henkes, S., Kostanjevec, K., Collinson, J. M., Sknepnek, R., & Bertin, E. (2020). Dense active matter model of motion patterns in confluent cell monolayers. *Nature Communications*, *11*(1), 1405. https://doi.org/10.1038/s41467-020-15164-5

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. https://doi.org/10.1109/MCSE.2007.55

Jiang, S., Campbell, D., Lu, Y., Li, H., & Hartley, R. (2021). *Learning to estimate hidden motions with global motion aggregation*. 9752–9761. https://doi.org/10.1109/ICCV48922.2021.00963

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., Willing, C., & team, J. development. (2016). Jupyter Notebooks - a publishing format for reproducible computational workflows. In F. Loizides & B. Scmidt (Eds.), *Positioning and power in academic publishing: Players, agents and agendas* (pp. 87–90). IOS Press. https://eprints.soton.ac.uk/403913/

Krekel, H., Oliveira, B., Pfannschmidt, R., Bruynooghe, F., Laugher, B., & Bruhin, F. (2004). *Pytest 6.2.5*. https://github.com/pytest-dev/pytest

Looi, K., Buckley, A. G., Rigby, P. J., Garratt, L. W., Iosifidis, T., Zosky, G. R., Larcombe, A. N., Lannigan, F. J., Ling, K.-M., Martinovich, K. M., Kicic-Starcevich, E., Shaw, N. C., Sutanto, E. N., Knight, D. A., Kicic, A., & Stick, S. M. (2018). Effects of human rhinovirus on epithelial barrier integrity and function in children with asthma. *Clinical & Experimental Allergy*, *48*(5), 513–524. https://doi.org/10.1111/cea.13097

Martinovich, K. M., Iosifidis, T., Buckley, A. G., Looi, K., Ling, K.-M., Sutanto, E. N., Kicic-Starcevich, E., Garratt, L. W., Shaw, N. C., Montgomery, S., Lannigan, F. J., Knight, D. A., Kicic, A., & Stick, S. M. (2017). Conditionally reprogrammed primary airway epithelial cells maintain morphology, lineage and disease specific functional characteristics. *Scientific Reports*, *7*(1), 17971. https://doi.org/10.1038/s41598-017-17952-4

Mitchel, J. A., Das, A., O'Sullivan, M. J., Stancil, I. T., DeCamp, S. J., Koehler, S., Ocaña, O. H., Butler, J. P., Fredberg, J. J., Nieto, M. A., Bi, D., & Park, J.-A. (2020). In primary airway epithelial cells, the unjamming transition is distinct from the epithelial-to-mesenchymal transition. *Nature Communications*, *11*. https://doi.org/10.1038/s41467-020-18841-7

Nnetu, K. D., Knorr, M., Käs, J., & Zink, M. (2012). The impact of jamming on boundaries of collectively moving weak-interacting cells. *New Journal of Physics*, *14*(11), 115012. https://doi.org/10.1088/1367-2630/14/11/115012

O'Sullivan, M. J., Mitchel, J. A., Das, A., Koehler, S., Levine, H., Bi, D., Nagel, Z. D., & Park, J.-A. (2020). Irradiation Induces Epithelial Cell Unjamming. *Frontiers in Cell and Developmental Biology*, *8*. https://doi.org/10.3389/fcell.2020.00021

Park, J.-A., Kim, J. H., Bi, D., Mitchel, J. A., Qazvini, N. T., Tantisira, K., Park, C. Y., McGill, M., Kim, S.-H., Gweon, B., Notbohm, J., Steward, R., Burger, S., Randell, S. H., Kho, A. T., Tambe, D. T., Hardin, C., Shore, S. A., Israel, E., … Fredberg, J. J. (2015). Unjamming and cell shape in the asthmatic airway epithelium. *Nature Materials*, *14*(10), 1040–1048. https://doi.org/10.1038/nmat4357

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., … Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems 32* (pp. 8024–8035). Curran Associates, Inc. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

Savian, S., Elahi, M., & Tillo, T. (2020). Optical Flow Estimation with Deep Learning, a Survey on Recent Advances. In R. Jiang, C.-T. Li, D. Crookes, W. Meng, & C. Rosenberger (Eds.), *Deep Biometrics* (pp. 257–287). Springer International Publishing. https://doi.org/10.1007/978-3-030-32583-1_12

Tomar, S. (2006). Converting video formats with FFmpeg. *Linux Journal*, *2006*(146), 10.

Turaga, P., Chellappa, R., & Veeraraghavan, A. (2010). Advances in Video-Based Human Activity Analysis: Challenges and Approaches. In M. V. Zelkowitz (Ed.), *Advances in Computers* (Vol. 80, pp. 237–290). Elsevier. https://doi.org/10.1016/S0065-2458(10)80007-5

Walt, S. van der, Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E., Yu, T., & contributors, the scikit. (2014). Scikit-image: Image processing in Python. *PeerJ*, *2*, e453. https://doi.org/10.7717/peerj.453

Wes McKinney. (2010). Data Structures for Statistical Computing in Python. In S. van der Walt & Jarrod Millman (Eds.), *Proceedings of the 9th Python in Science Conference* (pp. 56–61). https://doi.org/10.25080/Majora-92bf1922-00a

Zhai, M., Xiang, X., Lv, N., & Kong, X. (2021). Optical flow and scene flow estimation: A survey. *Pattern Recognition*, *114*, 107861. https://doi.org/10.1016/j.patcog.2021.107861