3D single-cell shape analysis using geometric deep learning

Matt De Vries
matt.devries@icr.ac.uk

Lucas Dent
lucas.dent@icr.ac.uk

Nathan Curry
nathan.curry08@imperial.ac.uk

Leo Rowe-Brown
l.rowe-brown19@imperial.ac.uk

Adam Tyson
adam@adamltyson.com

Christopher Dunsby
christopher.dunsby@imperial.ac.uk

Chris Bakal
chris.bakal@icr.ac.uk

Abstract

3D cell shape is linked to disease. Despite this, there is a paucity of methods to quantify 3D cell shapes. Current methods utilise pre-defined measures of cell geometry as opposed to data-driven approaches. Furthermore, we do not fully understand the cell shape landscape of cancer cells in 3D and the different shape classes that exist in a dataset or how these are correlated with drug treatments. To address this, we have developed a geometric deep learning method to learn 3D cell shape features and classes simultaneously. This technique combines an existing dynamic graph convolutional neural network encoder with a foldingnet decoder and improved deep embedded clustering. We use this model to learn meaningful shape representations of over 70 000 drug-treated melanoma cells imaged by 3D light-sheet microscopy. We propose describing cells and treatments by a 3D quantitative morphological signature, representing a cell's similarity to the learned shape classes in the dataset. This led to the insight that drugs treated with similar inhibitors share morphological signatures.

1 Introduction

Aberrations in cell geometry are linked to cell signalling and disease [1]. Traditionally, we understand shape using predetermined or "classical" features. These are pre-selected, handcrafted measures of cell geometry that often require a preconceived idea of which features are important. These descriptors include region-based shape descriptors such as area, perimeter, and eccentricity [2, 3, 4, 5, 6]. In the context of studying living cells in 2D images, these geometric approaches have been successful at identifying major categories of cell shape, and linking cell shapes to the molecular signalling pathways that control geometry [7, 8], or respond to changes in geometry [9, 10, 11]. 2D shape understanding has been illuminating, however, cells exist in 3D environments, and their 3D geometry is a rich source of information about the inner workings of cells and tissues. For example, recent pioneering work has demonstrated that shape motifs extracted from 3D images can be linked to lipid membrane signalling and cancer signalling pathways [12], and changes in 3D nuclear morphometry are linked to chromatin organisation [13]. At the scale of developing tissues, the discovery of new 3D geometries has explained how cells pack stably into curved organs [14]. Now, the development of high throughput light-sheet imaging [15, 16] has made it possible to image large numbers of cells in 3D with sub-cellular resolution, leading to a growing need for accurate 3D shape representations. We have addressed this need using unsupervised geometric deep learning (GDL) to learn shape representations of cells from 3D microscopy images of metastatic melanoma cells embedded in collagen tissue-like matrices. We have also used improved deep embedded clustering to learn the shape features and classes simultaneously. Our methods are implemented as a Python package for the ease of use by the biological research community [1]

[1] https://github.com/DeVriesMatt/cellshape

Learning Meaningful Representation of Life Workshop (NeurIPS 2022).

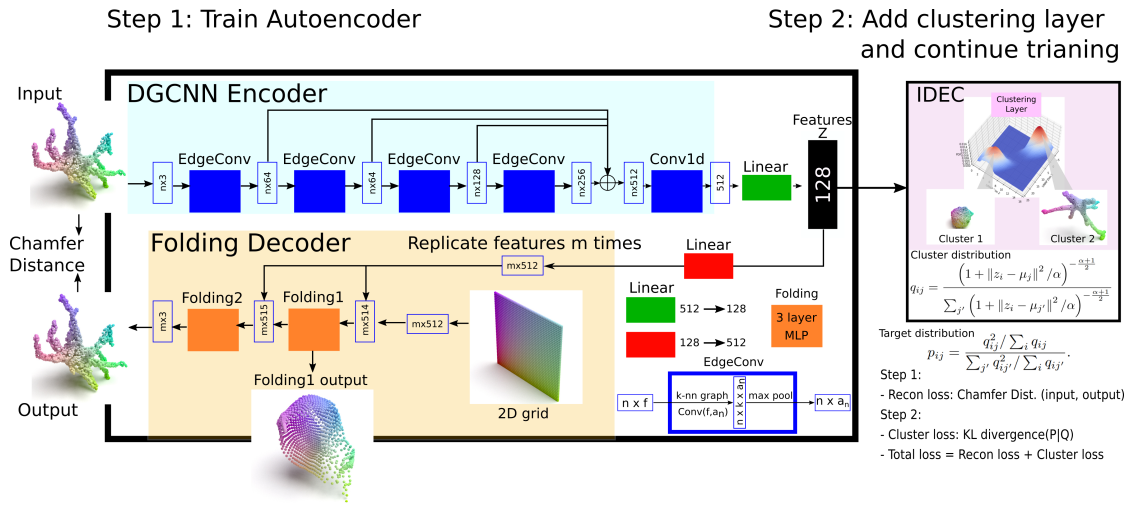

Figure 1: Our method consists of two parts, a DFN autoencoder and a clustering layer, and two stages of training. The DFN autoencoder comprises a DGCNN encoder (light blue) and a FoldingNet decoder (light orange). The DGCNN encoder is made up of a series of edge convolutional (EdgeConv) operations (blue square). EdgeConv constructs a local neighbourhood graph on the input point cloud and performs convolution on the adjacency matrix of that graph to output a lower-dimensional embedding. After the forth EdgeConv, we concatenate the output of each EdgeConv and pass this through a convolutional layer and then an MLP to obtain an embedding size of 128. The FoldingNet decoder takes the embedded feature vector as input, passes this through a linear layer to obtain a vector size of 512. This is then replicated m times ($m \times 512$) and is concatenated with an $m \times 2$ matrix of points sampled from a grid. This $m \times 514$ tensor is then fed through two folding operations which are 3 layered MLP's. The autoencoder, made up of the light blue and light orange parts only, is trained in the first stage by minimising the Chamfer Distance between the input and output point cloud (reconstruction loss). The clustering layer (light purple) is added after the autoencoder has been sufficiently trained. This clustering layer takes the embedded feature vector as input and maps this to the cluster centres of the feature space through a Student's t-distribution (pink square). We define a target distribution, $\mathbf{P}$, as a function of the output of the clustering layer, $\mathbf{Q}$, and minimise the KL divergence between $\mathbf{P}$ and $\mathbf{Q}$ as well as the reconstruction loss.

2 Methods

2.1 Dataset

WM266.4 melanoma cells were suspended in a collagen matrix and treated with several different drugs. We imaged over 70 000 cells in 3D using oblique-plane microscopy [16]. We then segmented these 3D single cells, using active contours and Ostu's thresholding, and used marching cubes to create mesh objects of vertices faces and normals. We then sampled 2048 points from the surfaces of the meshes to obtain point cloud representation of each cell and nuclei.

2.2 Dynamic graph convolutional foldingnet

The dynamic graph convolutional foldingnet (DFN) follows the design of the FoldingNet [17] with a dynamic graph convolutional neural network (DGCNN) [18] as the encoder (Step 1 in Figure 1). Our reason for the amendment of the encoder was from results on ModelNet40 [19], where we achieved higher transfer learning accuracies when using a DGCNN encoder as opposed to original FoldingNet encoder. Step 1 of our model training is done by minimising the Chamfer Distance used in [17]. We have implemented several encoder-decoder combinations in our Python package for further experimentation.

2.3 Improved deep embedded clustering

Deep embedded clustering (DEC) [20] is a specialised clustering technique that learns feature representations and cluster assignments using autoencoders, simultaneously. This involves an addition to an autoencoder in the form of a clustering layer. DEC [20] proposed pre-training an autoencoder and then disregarding the decoder part and replacing it with the clustering layer, but [21, 22] showed this weakens the representitativeness of the features learned. Improved deep embedded clustering (IDEC) [23] proposed keeping the decoder both and the clustering layer in the second stage of training and jointly optimising the

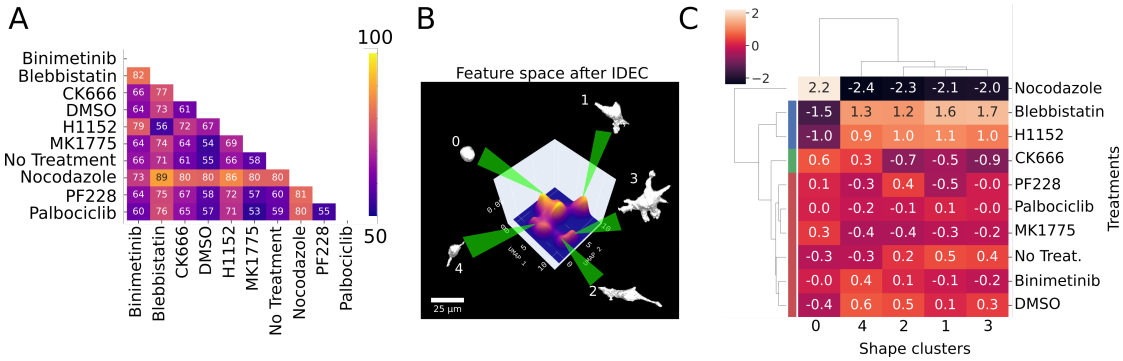

Figure 2: (A) shows the classification one-vs-one classification accuracies for different treatments in our dataset using 128 features extracted from the cell shape and 128 from nuclei shape. (B) shows the cell shape landscape after IDEC estimated using a kernel density approximation of the UMAP of the embedded feature space. We also show the five cluster centre shapes. (C) shows the hierarchical clustering of the normalised 3DQMS extracted for each treatment class.

reconstruction loss and the clustering loss simultaneously (bottom right corner of Figure 1). We followed IDEC [23]. After an autoencoder, such as the DFN, is sufficiently trained, the clustering layer takes, as input, the lower-dimensional feature vector output from the encoder and maps this to the cluster centres of the feature space. The cluster centres of this feature space are initialised through k-means clustering and then are trainable weights of the resulting neural network. The clustering layer assigns soft labels (q_{ij}) to each input by mapping features to clusters based on the Student's t-distribution [24] as a kernel that represents the similarity between a feature vector ($\mathbf{z_i}$) and a cluster centre ($\mu_\mathbf{j}$) (Step 2 in Figure 1). This can be interpreted as the probability of assigning input cell shape i to cluster j, hence why this is a soft assignment. Soft assignments with high probabilities are considered trustworthy, and thus DEC designs a target distribution, p_{ij}, which raises q_{ij} to the second power to place more emphasis on these confident assignments (See Figure 1 target distribution). We then minimise a clustering loss defined as the Kullback–Leibler divergence between the cluster and target distribution. This target distribution is updated every T steps and training is stopped when a divergence tolerance is reached. This divergence tolerance is the proportion of samples that change cluster assignments during the next update. We set the divergence tolerance to 0.001. The features at cluster centres become representations of 'example' shapes of each shape class. If there are k clusters in the dataset, each cell is assigned k soft labels (one for each cluster), and each soft label contains scores from zero to one, reflecting how similar the cell is to each of the different cluster centres (with a value of one being most similar). Since the 3D shape of cells in a fixed assay is a continuous distribution with an indefinite number of distinct classes (cells may be very similar in shape, yet almost never the exact same), it is more natural to describe a shape by a signature of how similar it is to a number of different exemplar classes. For example, we may wish to describe the 3D shape of a cell to be 30% similar to shape cluster 0, 10% similar to shape cluster 1, 5% similar to shape cluster 2 for shape classes $j = 0, 1, ...k$, such that the sum of similarity percentages add to 100. This can be defined as a quantitative morphological signature for the 3D shape of a cell (3DQMS). This is a direct output of the clustering layer. Step 2 in our model training is defined by minimising the clustering loss and the reconstruction loss simultaneously.

3 Results

3.1 Learned shape features predict drug treatment with greater accuracy than previous approaches

Initially we trained our DFN autoencoder without the addition of the clustering layer to extract 128 shape features from point cloud representations of cells and nuclei. Many studies in computer vision and 3D shape representation learning test the effectiveness of their representations with downstream classification tasks. This involves using a trained model to extract features from a dataset and then using these features to train a linear support vector machine (SVM) classifier [25, 26, 27, 28]. These tasks generally involve classification between objects from distinct semantic classes [19]. Our dataset is comprised of a single cell type and did not have such distinct semantic classes, so to test our shape representations, we predicted our drug treatment labels. Following similar procedures to previous researchers [25, 26, 27, 28], we used the trained DFN without the addition of the clustering layer to extract features from each cell. These features were then used to train a linear SVM. For all experiments, we use one-versus-one SVM classifiers

with an L2 penalty (C = 1), balanced class weights, and intercept scaling during training. We performed 10-fold cross-validation for each experiment and report mean accuracies. We found the greatest prediction accuracy resulted from combining both cytoplasmic and nuclear shape features, with predictions up to 89% for cells treated with myosin inhibitors and microtubule inhibitors (Figure 2 A). This is opposed to an accuracy of 86% when using the original FoldingNet encoder and 82% when using classical features. Furthermore, we tested whether prediction accuracies were affected by a cell's environment (closer to the coverslip - in a rigid environment - or further away from the coverslip - in a more tissue-like environment). Higher accuracies were seen in cells that were closer to the coverslip.

3.2 Improved deep embedded clustering learns a 3D quantitative morphological signature that generates new biological insights

After training the DFN autoencoder to learn 3D cell shape representation, we performed improved deep embedded clustering (IDEC) [23] to learn the 3DQMS of every cell. We chose 5 clusters as a hyperparameter. However, this could also be estimated by prior knowledge, or through more formal methods such as the elbow, average silhouette or gap statistic methods when undertaking the initial k-means clustering. Deep clustering found 5 shape clusters (Figure 2 B). We extracted the 3DQMS for every cell and grouped this by treatment, to show the average 3DQMS for cells treated with different inhibitors. Blebbistatin treated cells had high scores for protrusive shape classes, with Nocodazole treated cells having high scores for the round shape class (Figure 2 C). Furthermore, we noted that two distinct inhibitors of myosin activation (Blebbistatin and H1152) grouped together, indicating that the 3DQMS may be a powerful tool for using cell shape to identify drugs that act on similar pathways. Experiments on large drug screens is needed to determine if this is the case in general. The other classes of treatment effects on shape included Nocodazole, which inhibits the polymerisation of microtubules, CK666, which prevents actin branching and then control cells grouped with those treatments that do not have significant effects on cell shape.

4 Conclusion

Here, we have utilised a combination of state-of-the-art computer vision techniques and applied them to cancer biology. We have learned the shape representations of 3D cancer cells using GDL and have shown that these outperform traditional measures of cell shape when predicting drug treatment. We propose defining cell shapes by their similarity to shape modes in the dataset - known as their 3DQMS - and show that this is useful in linking treatments which affect the same pathway. Further experimentation on where GDL techniques will have a significant edge over other methods such as spherical harmonics is needed. We suggest that GDL methods might prevail on complex datasets. Traditional, hand-crafted features are inherently biased and may miss important, undefined phenotypes Finally, we implemented our methods as a Python package for ease of use by the research community.

References

[1] Pei-Hsun Wu, Daniele M. Gilkes, Jude M. Phillip, Akshay Narkar, Thomas Wen-Tao Cheng, Jorge Marchand, Meng-Horng Lee, Rong Li, and Denis Wirtz. Single-cell morphology encodes metastatic potential. *Science Advances*, 6(4):eaaw6938, January 2020. Publisher: American Association for the Advancement of Science.

[2] Veronika Graml, Xenia Studera, Jonathan L.D. Lawson, Anatole Chessel, Marco Geymonat, Miriam Bortfeld-Miller, Thomas Walter, Laura Wagstaff, Eugenia Piddini, and Rafael E. Carazo-Salas. A genomic multiprocess survey of machineries that control and link cell shape, microtubule organization, and cell-cycle progression. *Developmental Cell*, 31(2):227–239, 2014.

[3] Saurav Basu, Soheil Kolouri, and Gustavo K. Rohde. Detecting and visualizing cell phenotype differences from microscopy images using transport-based morphometry. *Proceedings of the National Academy of Sciences*, 111(9):3448–3453, 2014.

[4] Thouis R. Jones, Anne E. Carpenter, Michael R. Lamprecht, Jason Moffat, Serena J. Silver, Jennifer K. Grenier, Adam B. Castoreno, Ulrike S. Eggert, David E. Root, Polina Golland, and David M. Sabatini. Scoring diverse cellular morphologies in image-based screens with iterative feedback and machine learning. *Proceedings of the National Academy of Sciences*, 106(6):1826–1831, 2009.

[5] Chris Bakal, John Aach, George Church, and Norbert Perrimon. Quantitative morphological signatures define local signaling networks regulating cell morphology. *Science (New York, N.Y.)*, 316(5832):1753–1756, June 2007. Number: 5832.

[6] Zheng Yin, Amine Sadok, Heba Sailem, Afshan McCarthy, Xiaofeng Xia, Fuhai Li, Mar Arias Garcia, Louise Evans, Alexis R. Barr, Norbert Perrimon, Christopher J. Marshall, Stephen T. C. Wong, and Chris Bakal. A screen for morphological complexity identifies regulators of switch-like transitions between discrete cell shapes. *Nature Cell Biology*, 15(7):860–871, July 2013.

[7] Sam Cooper, Amine Sadok, Vicky Bousgouni, and Chris Bakal. Apolar and polar transitions drive the conversion between amoeboid and mesenchymal shapes in melanoma cells. *Molecular biology of the cell*, 26(22):4163–4170, 11 2015.

[8] Erik Sahai and Christopher J. Marshall. Differing modes of tumour cell invasion have distinct requirements for Rho/ROCK signalling and extracellular proteolysis. *Nature Cell Biology*, 5(8):711–719, August 2003.

[9] Julia E. Sero, Heba Zuhair Sailem, Rico Chandra Ardy, Hannah Almuttaqi, Tongli Zhang, and Chris Bakal. Cell shape and the microenvironment regulate nuclear translocation of NF-B in breast epithelial and tumor cells. *Molecular Systems Biology*, 11(3):790, March 2015.

[10] Julia E. Sero and Chris Bakal. Multiparametric Analysis of Cell Shape Demonstrates that -PIX Directly Couples YAP Activation to Extracellular Matrix Adhesion. *Cell Systems*, 4(1):84–96.e6, January 2017. Number: 1.

[11] Patricia Pascual-Vargas, Samuel Cooper, Julia Sero, Vicky Bousgouni, Mar Arias-Garcia, and Chris Bakal. RNAi screens for Rho GTPase regulators of cell shape and YAP/TAZ localisation in triple negative breast cancer. *Scientific Data*, 4:170018, March 2017.

[12] Meghan K. Driscoll, Erik S. Welf, Andrew R. Jamieson, Kevin M. Dean, Tadamoto Isogai, Reto Fiolka, and Gaudenz Danuser. Robust and automated detection of subcellular morphological motifs in 3D microscopy images. *Nature Methods*, 16(10):1037–1044, October 2019.

[13] Alexandr A. Kalinin, Xinhai Hou, Alex S. Ade, Gordon-Victor Fon, Walter Meixner, Gerald A. Higgins, Jonathan Z. Sexton, Xiang Wan, Ivo D. Dinov, Matthew J. O'Meara, and Brian D. Athey. Valproic acid-induced changes of 4D nuclear morphology in astrocyte cells. *Molecular Biology of the Cell*, 32(18):1624–1633, August 2021. Publisher: American Society for Cell Biology (mboc).

[14] Pedro Gómez-Gálvez, Pablo Vicente-Munuera, Antonio Tagua, Cristina Forja, Ana M. Castro, Marta Letrán, Andrea Valencia-Expósito, Clara Grima, Marina Bermúdez-Gallardo, Óscar Serrano-Pérez-Higueras, Florencia Cavodeassi, Sol Sotillos, María D. Martín-Bermudo, Alberto Márquez, Javier Buceta, and Luis M. Escudero. Scutoids are a geometrical solution to three-dimensional packing of epithelia. *Nature Communications*, 9(1):1–14, July 2018. Number: 1 Publisher: Nature Publishing Group.

[15] C. Dunsby. Optically sectioned imaging by oblique plane microscopy. *Optics Express*, 16(25):20306–20316, December 2008.

[16] Vincent Maioli, George Chennell, Hugh Sparks, Tobia Lana, Sunil Kumar, David Carling, Alessandro Sardini, and Chris Dunsby. Time-lapse 3-d measurements of a glucose biosensor in multicellular spheroids by light sheet fluorescence microscopy in commercial 96-well plates. *Scientific Reports*, 6(1):37777, Nov 2016.

[17] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation, 2018.

[18] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (TOG)*, 2019.

[19] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1912–1920, 2015.

[20] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 478–487, New York, New York, USA, 20–22 Jun 2016. PMLR.

[21] Xifeng Guo, Xinwang Liu, En Zhu, and Jianping Yin. Deep clustering with convolutional autoencoders. In Derong Liu, Shengli Xie, Yuanqing Li, Dongbin Zhao, and El-Sayed M. El-Alfy, editors, *Neural Information Processing*, pages 373–382, Cham, 2017. Springer International Publishing.

[22] Xifeng Guo, En Zhu, Xinwang Liu, and Jianping Yin. Deep embedded clustering with data augmentation. In Jun Zhu and Ichiro Takeuchi, editors, *Proceedings of The 10th Asian Conference on Machine Learning*, volume 95 of *Proceedings of Machine Learning Research*, pages 550–565. PMLR, 14–16 Nov 2018.

[23] Xifeng Guo, Long Gao, Xinwang Liu, and Jianping Yin. Improved deep embedded clustering with local structure preservation. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 1753–1759, 2017.

[24] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.

[25] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 206–215, 2018.

[26] Ling Zhang and Zhigang Zhu. Unsupervised feature learning for point cloud understanding by contrasting and clustering using graph convolutional neural networks. In *2019 International Conference on 3D Vision (3DV)*, pages 395–404, 2019.

[27] Xiang Gao, Wei Hu, and Guo-Jun Qi. GraphTER: Unsupervised learning of graph transformation equivariant representations via auto-encoding node-wise transformations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[28] Juyoung Yang, Pyunghwan Ahn, Doyeon Kim, Haeil Lee, and Junmo Kim. Progressive seed generation auto-encoder for unsupervised point cloud learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6413–6422, October 2021.