Generative Spatiotemporal Modeling Of Neutrophil Behavior

Narita Pandhe[⋆] *Balazs Rada*[†] *Shannon Quinn*[⋆]

[⋆] Department of Computer Science
[†]Department of Infectious Diseases
University of Georgia
naritapandhe@uga.edu, radab@uga.edu, squinn@cs.uga.edu

ABSTRACT

Cell motion and appearance have a strong correlation with cell cycle and disease progression. Many contemporary efforts in machine learning utilize spatio-temporal models to predict a cell's physical state and, consequently, the advancement of disease. Alternatively, generative models learn the underlying distribution of the data, creating holistic representations that can be used in learning. In this work, we propose an aggregate model that combine Generative Adversarial Networks (GANs) and Autoregressive (AR) models to predict cell motion and appearance in human neutrophils imaged by differential interference contrast (DIC) microscopy. We bifurcate the task of learning cell statistics by leveraging GANs for the spatial component and AR models for the temporal component. The aggregate model learned results offer a promising computational environment for studying changes in organellar shape, quantity, and spatial distribution over large sequences.

Index Terms— Generative Adversarial Networks, Autoregressive Process, Biological Images

1. INTRODUCTION

Polymorphonuclear neutrophil granulocytes (neutrophils) are the most abundant white blood cells in most mammals. They are highly motile phagocytic cells that constitute the first line of defense of the innate immune system [1]. Study of neutrophils and their underlying motion patterns provide insights into a host's response and behavior as a function of specific stimulus. Our understanding of cell behavior and the sources of cellular variation can be significantly aided and tested using cell modeling and simulations [2].

Recently, generative models have been extensively utilized for natural images. Examples include Variational Autoencoders [3] and Generative Adversarial Networks (GANs) [4]. Generative models have the ability to learn the underlying statistical distributions over data and, thus, can generate exemplars of the true data set. It can learn sophisticated conditional relationships as well. In 2014, [4] proposed Generative Adversarial Networks (GANs), a framework for learning generative models. GANs do not rely on training objectives related to log-likelihood. Instead, GAN training can be seen as a competitive game between two models: the generator (G) and the discriminator (D). Deep Convolutional GANs (DCGANs) [5] train convolutional networks in adversarial settings in order to generate natural images from CelebA [6], LSUN [7], and Imagenet datasets [8]. [9] applied GANs to biological images to study the coexistence of proteins. Initial GAN models suffered from issues including training instabilities and mode collapse, making them harder to use. Active areas of research include novel applications, optimizing the network architecture, developing best training practices, and improving the cost function.

For computer vision systems, motion synthesis is still a challenging task and is drawing more contemporary research attention. Synthesis can be defined as generating new versions of a dataset which follow the distribution of original and is closely related to modeling. [10] presents an algorithm that synthesizes motions based on annotations that describe it. Motion is constructed by splitting segments of movement from a corpus of motion data and assembling them. Each segment is modeled using an autoregressive process. This helps in modeling complicated non-stationary sequences which a single autoregressive process cannot handle. In [11], frames of original videos are projected into low-dimensional space and then learned as an AR model. [12] extends this approach by overcoming the problems of non-linearities in the data either using a spline-fitting approach or a combined appearance model. Many approaches have employed GANs for video generation and frame prediction. [13] utilizes two convolutional networks, separating foreground and background imagery, to learn directly from a massive dataset of real-world videos.

In this work, we simulate the behavior of human neutrophils. Considering the limited dataset available, we propose an aggregate application of GANs and AR models. We bifurcate our approach as two tasks: generating neutrophil's appearance and its motion, capturing the statistics independently. The GAN learns the appearance and spatial statistics, while the AR model captures the temporal aspect. Simulation is then achieved by sampling a point from the appearance space given the temporal dynamics up to the last observation.

2. RELATED WORKS

Several computational methods have been proposed for constructing from image data, statistical models of cellular and subcellular structures. General shape models such as Active Shape models [14] and cell shape model conditional on the

nucleus shape [15] have been used. To our knowledge, the closest related literature is comprised of [9] for biological image synthesis and [13] for motion synthesis. Differences to [9] include the following. (1) Our GAN architecture is based on DCGANs, while theirs is a modified DCGAN for channel separation. (2) They apply GANs to samples from fluorescent microscopic images consisting of two channels, red and green. We use GANs for DIC microscopy images consisting of a single channel. They tackle a more difficult problem: using the information contained in the red channel learn how to generate a cell with several green-labeled proteins together. We are modeling single channel cell images.

Like this work, [13] uses a two stream model, but differs as follows. (1) They use Long Short Term Memory Networks (LSTMs), while we use DCGANs for content and derive AR processes from motion. (2) Their network learns the temporal dynamics directly from raw pixels, using identified features combined with spatial features to make pixel-level predictions. We assume the background is stationary and only the foreground cells move. All the pixels of the foreground cells move similarly, so we pool them together into an AR model.

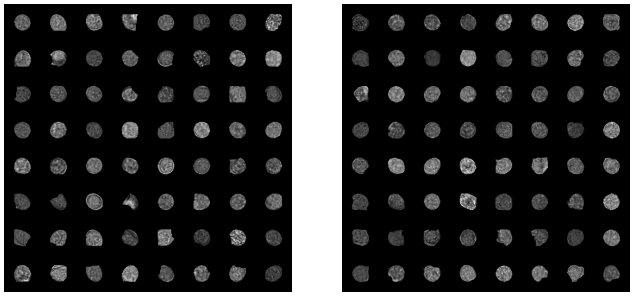

Fig. 1: Real (left) and synthesized (right) images of neutrophil. The synthetic images were created using DCGAN combined with Improved WGAN loss function.

3. DATA

Videos imaging the two dimensional motion of human neutrophilic granulocytes are provided by Balazs Rada (Department of Infectious Diseases, University of Georgia). The videos are recorded using DIC microscopy which is used to enhance the contrast in unstained, transparent samples. The dataset consists of 11 videos, including 3 videos of normal neutrophils and 8 videos of neutrophils treated with an inhibitor, MRS2578, targeting a purinergic receptor. Duration of most of the videos is 3.0secs. We extracted frames of 1024x1024 resolution at 20fps. Individual cells were segmented using fully convolutional DenseNets [16], centered and resized to 64x64 resolution, resulting in 17280 total grayscale images.

4. METHODS

4.1. GANs for Cell Image Synthesis

$$\min_G \max_D V(D,G) = \mathop{\mathbb{E}}_{x\sim p_{data(x)}} \big[\log D(x)\big] + \mathop{\mathbb{E}}_{z\sim p_{z(z)}} \big[\log(1-D(G(z)))\big] \quad \text{(Eq. 1)}$$

$$L = \underbrace{\mathop{\mathbb{E}}_{\widetilde{x}\sim \mathbb{P}_g}\big[D(\widetilde{x})\big] - \mathop{\mathbb{E}}_{x\sim \mathbb{P}_r}\big[D(x)\big]}_{\text{Original Critic Loss}} + \underbrace{\lambda \mathop{\mathbb{E}}_{\hat{x}\sim \mathbb{P}_{\hat{x}}}\big[(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2\big]}_{\text{Gradient Penalty}} \quad \text{(Eq. 2)}$$

GANs consists of 2 neural networks competing against each other: Generator (G) and Discriminator (D). G generates images from random noise. While doing so, it tries to get as close as it can, to the distribution of real images. D classifies between the real images and fake images generated by G. Both are trying to perform best at their respective tasks and maximise their gains. D is characterized as adversarial loss for training G.

Formally, consider a set of training images, $x \in X_{real}$ coming from a real distribution P_d. The generator is a neural network $G(z, \theta G)$ parametrized by θG and discriminator is a neural network $D(x, \theta D)$ parametrized by θD. $G(z, \theta G)$ takes in random noise z from the distribution P_z and generates images $x \in X_{fake}$. $D(x, \theta D)$ takes images from $x \in X_{real}$ and $x \in X_{fake}$, both, and outputs a scalar between [0, 1]. The output is higher if the sample belongs to X_{real} else X_{fake}. Both G and D are trained simultaneously. The goal of D is to maximize the probability of assigning correct labels to an input while G minimizes $\log(1-D(G(z)))$. As a result D and G can be seen as playing a minimax game, as formulated in Eq. 1. Historical attempts to scale up GANs using CNNs to model images have been generally unsuccessful [4]. DCGANs [5] identified a family of architectures that resulted in stable training and can generate higher resolution images. We have adopted the architecture of DCGAN for both generator and discriminator.

Eq. 1 can be reformulated via minimization of the Jensen Shannon (JS) divergence between the data-generating distribution P_d and the distribution P_g induced by P_z and G. [17] theoretically justified that JS minimized by GANs behaves badly and is potentially not continuous w.r.t to the generators parameters. They propose using an alternative distance - Earth Mover's distance (EM) also known as Wasserstein Distance, W(q,p). Since, computing Wasserstein distance is intractable, [17] shows an approximate solution to the same using Kantorovich-Rubinstein duality, wherein D is the set of 1-Lipschitz functions. To enforce the Lipschitz constraint authors propose to clip the weights of the critic (D referred as critic because it's not trained to classify) within a compact space $[-c, c]$. Recently, [18] proposed an alternative way to enforce the Lipschitz constraint. Instead of weight clipping, they penalize the norm of the critic's gradient with respect to its input, for random samples $\hat{x} \sim \mathbb{P}_{\hat{x}}$. The objective function Eq. 2 leads stable training of a wide variety of GAN architectures with almost no hyperparameter tuning.

4.1.1. Experiments

We evaluated performances of models based on DCGAN architecture trained with GAN, Wasserstein GAN (WGAN),

and Improved WGAN loss functions [4, 17, 18]. To evaluate the performance of GANs we utilize the optimization-based approach discussed by [9] to check if the test samples can be reconstructed well. To test for mode collapse, a common failure in GANs, for a fixed trained generator G we examine how well it can reconstruct images from a held out test set. For each image in the test set, we minimize the L2-distance between the generated and test images w.r.t. the noise vector z. We use 50 iterations of L-BFGS and select the best reconstruction out of 3 runs. We also report the negative log likelihood (NLL) w.r.t. the prior P_z of the noise vectors z.

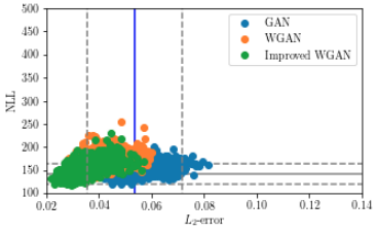

Fig. 2: Reconstruction errors against negative log likelihood (NLL) of the latent vectors found by reconstruction are displayed. The vertical blue line shows the mean L2-error. Horizontal gray line show mean NLL ($\pm$3std) of the noise sampled from the Gaussian prior. Lower values for both are better.

4.1.2. Latent Space Walk

We can interpolate between points in the latent space and understand the landscape. Walking the manifold can identify if there are any sharp transitions and whether the network has memorized. If walking the latent space results in smooth semantic changes to the image generations we can reason that the model has learned relevant, interesting representations [5].

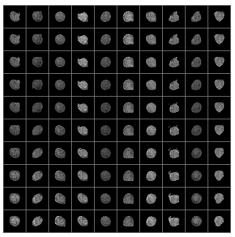

Fig. 3: Interpolation between a series of 10 random points in the latent space depicts that the space learned has smooth transitions. Top row depicts the starting location for each of the 10 points. Last row depicts the respective ending locations.

4.2. AR for Cell Motion Synthesis

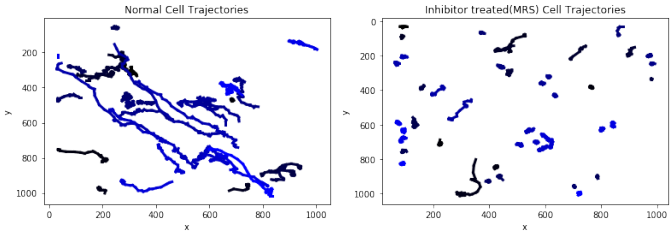

Fig. 4: 2D trajectory plots of normal neutrophil and inhibitor-treated (MRS) neutrophil. The inhibitor-treated (MRS) neutrophil tend to exhibit less movements in comparison to the normal ones.

$$\overrightarrow{y_t} = C\overrightarrow{x_t} + \overrightarrow{u_t} \quad \text{(Eq. 3)}$$

$$\overrightarrow{x_t} = B_1\overrightarrow{x}_{t-1} + B_2\overrightarrow{x}_{t-2} + .. + B_d\overrightarrow{x}_{t-d} + \overrightarrow{v_t} \quad \text{(Eq. 4)}$$

Different motion patterns are observed based on the cell conditions. We build a global motion pattern for normal and inhibited cells respectively, because we assume all the pixels(under the same conditions) move similarly. Based on the existing motion characteristics, new sequences can be synthesized for the corresponding cells. AR models are linear dynamical systems and are able to model a pattern of points in a particular space having a temporal component.

An AR process for a series of points in a d-dimensional space can be modelled as Eq. 3, Eq. 4. Eq. 3 decomposes each video frame $\overrightarrow{y_t}$ into a low-dimensional state vector $\overrightarrow{x_t}$ and a white noise term $\overrightarrow{u_t}$. Eq. 4 denotes new state $\overrightarrow{x_t}$ is a function of the sum of d of its previous states $\overrightarrow{x}_{t-1}, \overrightarrow{x}_{t-2}, .., \overrightarrow{x}_{t-d}$, each multiplied by corresponding coefficients $B = B_1, B_2, ..., B_d$ [19]. The noise terms u and v represent the residual difference between the observed data and the solutions to the linear equations, assumed to be Gaussian White noise.

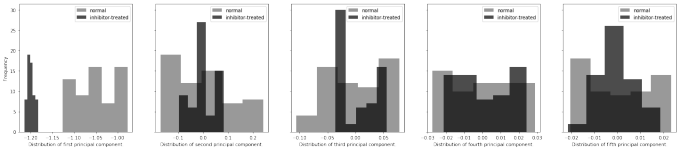

Fig. 5: Histograms show the distributions of values taken by normal (gray) and inhibitor-treated (black) neutrophil for top 5 principal components.

4.2.1. Experiments

Neutrophil motion is represented as trajectories of individual cells consisting of its center Cartesian coordinates across all the frames. Trajectories belonging to normal and inhibited cells are pooled separately and then projected into an eigenspace using SVD, yielding the principal components C. Subsequently, AR coefficients are determined. Parameter q determines the dimensionality of the subspace C; parameter d determines the order of AR coefficients $B = B_1, B_2, ..., B_d$. We performed grid search over $q \in [2, 10]$ and $d \in [1, 10]$.

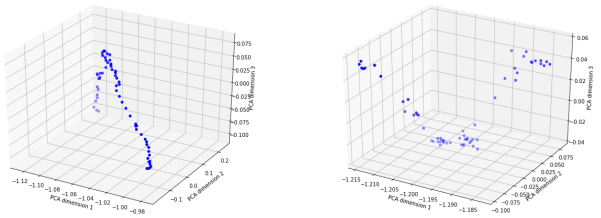

Fig. 6: Neutrophil motion using the first three dimensions of the subspace of the AR model for normal (left) and inhibitor-treated (right). This motion is governed by the AR coefficients.

4.3. Synthesis

Synthesized neutrophil behavior consists of two parts: content and appearance is sampled from our trained generator G and motion is sampled from a point in subspace C. Using Eq. 4, we iteratively generate different sequences. These new sequences are then projected back into the original space,

leading to a new motion pattern synthesized entirely from the eigenvector information. The separation of motion and appearance in two streams enable GANs and AR process to identify the respective key features. This results in movement of only the foreground cells and leaves the rest untouched. It also gives an advantage of synthesizing video clips of different cells following different trajectories but nonetheless looks similar to the existing motion patterns.

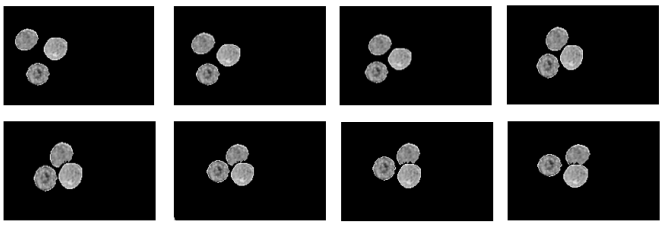

Fig. 7: Sample results of appearance and motion synthesis.

5. CONCLUSION

In this paper we presented a two stream approach to simulate human neutrophil behavior. Owing to the very limited data at our disposal, we utilized GANs to learn the spatial statistics and AR models to learn the temporal statistics. Bifurcation of appearance and motion allows a controlled video generation process. This work can enable us to quantify changes in organellar appearance, spatial distribution and help in understanding how subsets of the organellar ensembles evolve, improving our understanding of cellular mechanisms as they respond to their environments.

6. ACKNOWLEDGMENTS

We thank R. Ceren for constructive criticism of this manuscr-ipt. This work was supported in part by AWS in Education Grant Award. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research.

7. REFERENCES

[1] AW. Sehgal, "How Neutrophils Kill Microbes," *Annual Review of Immunology*, vol. 23, 2005.

[2] GR. Johnson, J. Li, A. Shariff, Rohde GK., and Murphy RF(2015), "Automated Learning of Subcellular Variation among Punctate Protein Patterns and a Generative Model of Their Relation to Microtubules," *"PLoS Comput Biol"*, vol. 11(12): e1004614, 2015.

[3] DP. Kingma and M. Welling, "Auto-encoding variational bayes.," *CoRR*, vol. abs/1312.6114, 2013.

[4] Ian I. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., pp. 2672–2680. Curran Associates, Inc., 2014.

[5] Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *CoRR*, vol. abs/1511.06434, 2015.

[6] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang, "Deep learning face attributes in the wild," in *Proceedings of International Conference on Computer Vision (ICCV)*.

[7] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao, "LSUN: construction of a large-scale image dataset using deep learning with humans in the loop," *CoRR*, vol. abs/1506.03365, 2015.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[9] RE. Carazo Salas A. Osokin, A. Chessel and Federico Vaggi, "GANs for biological image synthesis," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2017.

[10] O. Arikan, D. Forsyth, and JF. O'Brien, "Motion Synthesis from Annotations," *"ACM Transactions on Graphics (TOG)"*, vol. 22, pp. 402–408, 2003.

[11] D. Oziem, N. Campbell, C. Dalton, Gibson D., and Thomas B, "Combining Sampling and Autoregression for Motion Synthesis," *"Proceedings of the Computer Graphics International"*, pp. 510–513, 2004.

[12] NW Campbell, CJ Dalton, DP Gibson, DJ Oziem, and BT Thomas, "Practical generation of video textures using the auto-regressive process," *Image Vision Computing*, vol. 22 (10), pp. 819 – 827, 2004, Publisher: Elsevier.

[13] Ruben Villegas, Jimei Yang, Seunghoon Hong, Xunyu Lin, and Honglak Lee, "Decomposing motion and content for natural video sequence prediction," *CoRR*, vol. abs/1706.08033, 2017.

[14] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models-their training and application," *Comput. Vis. Image Underst.*, vol. 61, no. 1.

[15] Ting Zhao and Robert F Murphy, "Automated learning of generative models for subcellular location: building blocks for systems biology," *Cytometry Part A*, vol. 71, no. 12, pp. 978–990, 2007.

[16] Simon Jégou, Michal Drozdzal, David Vázquez, Adriana Romero, and Yoshua Bengio, "The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation," *CoRR*, vol. abs/1611.09326, 2016.

[17] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," *arXiv preprint arXiv:1701.07875*, 2017.

[18] Ishaan Gulrajani, Faruk Ahmed, Martín Arjovsky, Vincent Dumoulin, and Aaron C. Courville, "Improved training of wasserstein gans," *CoRR*, vol. abs/1704.00028, 2017.

[19] SP. Quinn, MJ. Zahid, Durkin JR, Francis RJ., Lo CW, and Chennubhotla SC, "Automated identification of abnormal respiratory ciliary motion in nasal biopsies," *"Science Translational Medicine"*, vol. 7, pp. 299ra124, 2015.