Classification of cellular states for high-speed image-based microfluidic cell sorting

Jarod Rutledge Tanish Jain Tejaswini Ganapathi

1. Abstract

The Steinmetz lab at Stanford has recently developed Image Cell Sorting (ICS) which combines FACS with high-speed laser microscopy and image analysis, enabling high-speed sorting of cells based on microscopy images that are constructed from laser pulses in real time (sorting up to 15,000 images and cells/sec). Coupling this technology with deep learning has the potential to unlock more complex cellular states for image-based sorting. Here, we establish the feasibility of deep learning on ICS cellular imaging by training cell type classifiers on a dataset of ~ 2 million images from published and unpublished ICS experiments. We process this first-of-its-kind dataset from raw 7-channel laser pulse images, and train multiple ResNet and MobileNetV2 models to classify 36 cell classes. Models were chosen to test a range of model sizes and explore tradeoffs between prediction accuracy and inference time at testing, which is an important variable for fast image-based sorting. We also tested whether pre-training models on natural images would improve performance as seen in many other domains, since ICS cell images differ substantially from natural images. Next, we tested the capacity of different architectures to classify cell types based on fewer laser channels, because 4 of the 7 available lasers require special sample preprocessing to add molecular fluorophores that are not compatible with many applications. Lastly, we applied saliency mapping on test set images to explore the feasibility of biological interpretation of deep learning models in this context.

2. Introduction

Here we describe the biological context of the problem and related machine learning work on cellular imaging data. For explicit details of inputs and outputs used in this work, please refer to the **Datasets** section.

2.1. Biological Context

Fifty years ago, fluorescence-activated cell sorting (FACS) technology revolutionized biology by enabling scientists to characterize and isolate cells based on the expression of cell surface protein markers for detailed biological study [6]. Nobel prizes, cancer cures, and surprising discoveries have come from this technology. However, most of cell biology is inaccessible to FACS, which cannot readily distinguish many differences in sub-cellular physiology, protein localization, or dynamic states including the mitotic cell cycle and responses to many important stimuli which cause internal biochemical and structural shifts. Many of these changes can, however, be visualized with fluorescence microscopy imaging. Schraivogel et al. recently described high-speed image-enabled cell sorting (ICS) [17] to combine the power of microscopy and FACS. ICS uses a new laser optics technique called fluorescence radiofrequency-tagged emission (FIRE) to generate diffraction-limited fluorescence microscopy images at a frame rate of 4.4 kHz[9] (Fig 1). This enables imaging of single cells as they travel through a microfluidic channel at high flow-rates in a traditional FACS machine. The ICS machine can image and sort upwards of 10 million cells in a single 8 hour experiment, unlocking a new scale of cellular imaging data. To put this in context, the two largest publicly accessible repositories of cell image data have a combined 3,633,364 images[2, 7], collected by Recursion Pharmaceuticals and the Broad Institute over the course of many years (and millions of dollars).

2.2. Related Work

ICS has many potential applications, all of which would benefit tremendously from more sophisticated cell sorting algorithms and the ability to learn rich representations of cellular health and physiology. To date, deep learning on this imaging data type has never been attempted. Schraivogel et al. previously developed a random forest classifier to sort cells in different stages of cell division [17], and before ICS imaging it was not possible to sort cell cycle stages without treating the cells with harsh drugs that disturb their biology[5].

While there is very little work in this field so far, there has been recent pioneering work in the related field of traditional cell microscopy images. Anne Carpenter and others have pioneered deep learning on images of cells in culture[3] to understand complex phenotypes and perform phenotypic screens. They have generated large tra-

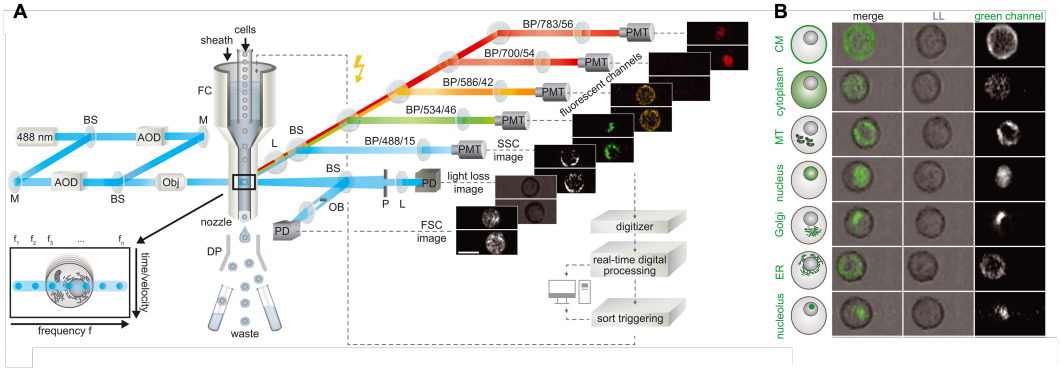


Figure 1. **(A)** Schematic representation of the ICS optical and flow hardware components from Schraivogel et al[17]. See reference for more detail on laser beamlet setup. Example 7-channel output images: HeLa cells expressing the Golgi marker GalNAcT2-green fluorescent protein (GFP) (green) were stained with cell surface marker CD147 PE-CF594 (orange) and DRAQ5 nuclear dye (red). FSC, SSC, and light loss images are shown in grayscale. Scale bar, 20 um. **(B)** ICS-based imaging of HeLa cells expressing GFP- or mNG-tagged fluorescent proteins or stained with organelle-specific green fluorescent dyes. One representative image is shown per organelle; light loss and green fluorescence channel shown.

ditional fluorescence imaging microscopy datasets [2] and applied CNNs[12], GANS[10], and VAEs/B-VAEs/MMD-VAEs[4], among other methods, to classify and segment [1] cells. Recursion Pharmaceutical has also used the imaging modalities pioneered by these academic groups to build one of the fastest growing drug discovery platforms in medicine[7]. Each of these groups have trained classifiers that can accurately classify upwards of 30,000 different cell classes, representing every single gene knockout in the genome and many chemical perturbations[2, 7] (inference on the latent space from this type of general cell state classifier powers Recursion's extremely successful drug discovery engine). It remains to be seen if ICS imaging can reach the performance these two decade-long efforts, particularly because ICS images are at the moment considerably lower resolution (100x100) than traditional microscopy (Fig 1).

Recently, others have attempted to integrate FACS and imaging to perform deep-learning image-based sorting, namely Hiroyuki Noji's [15] and Keisuke Goda's [13] groups in Japan. Notably, although these groups were able to achieve sorting based on additional optical signals from FACS, it has been challenging to reconstruct and sort on images and even more challenging to do so at a high speed[13, 15]. Keisuke's group is the only other to demonstrated image-based classification with FACS. However, their system performs two orders of magnitude slower and they only demonstrate a binary classification task separating 3-um bead particles from 6-um bead particles using a simple custom CNN architecture composed of four convolution layers and two fully connected layers accompanied by two max-pooling and three dropout connections [13]. This is much less challenging than separating different cell types and cell states within the same cell type.

3. Dataset

In this study, we move towards the powerful classifiers and latent-space representations that have been demonstrated for high-resolution cell microscopy by training a cell type classifier on ICS image data using 36 classes from multiple experiments performed on the ICS. The input to our classifier is a custom pre-processed image generated by the ICS machine during cell sorting (Fig 1A) and the output is a predicted class label (Figure 1B shows some classes in the model). As shown in Figure 1A, each image has a light intensity channel (light loss), a forward light scatter channel (FSC), a side light scatter channel (SSC) and 4 laser channels which detect the excitation and emission of fluorescent molecules at the labeled wavelengths.

3.1. Pre-processing

We worked from raw data provided by a collaborator Daniel Schraivogel in the Steinmetz group. Our dataset consisted of 2,060,000 images of mouse bone marrow cells and 430,000 images of human cell lines across 43 cell types and states. Initially, we stored these images on Stanford Drive because of the unlimited storage and ability to test things with google colab. However, we learned that the Drive filesystem was not stable with that large number of images we had uploaded, and we were forced to migrate to a Stanford compute cluster late in the project.

Each input image had shape (7,104,H), where H varied between 3 and 202 pixels. We filtered through all images and identified 21 as a best pixel height based on abundance and visual inspection. We removed images with $H < 21$, and combined center cropping with striding over pixel rows with stride 2 to reduce the height of images with $H > 21$. After filtering we were left with some classes which con-

tained very few or no images, so we removed classes with less than 100 images a leave 309,254 across 36 classes which were split 80% training/validation and 20% testing. There was some class imbalance (Appendix), so we then implemented custom pytorch transforms to per-channel normalize the training images, and to randomly horizontally or vertically flip images, which had to be done from scratch to accommodate the 7-channel image inputs. The team biologist (Jarod) felt other transforms could alter information in the images, so no other transforms were done.

4. Methods

In this section, we will start with describing the overall structure of our implementations and rationale behind our choices. We will then go into the intuition of the different architectures we chose to implement, and explain the reasons for our choice.

Figure 1A shows a sample of one image in our dataset. Each image has 7 channels, representing different device acquisition methods of the cell(s) imaged. We extract a single channel ("light loss"), 3 channel, and also use the full 7 channel image in experiments. We split our dataset 80 percent training, and 10 percent each for validation and testing. After we processed the data for training, we trained for 50 epochs. In every batch (of size 32) we calculated the cross entropy loss function, training and validation accuracy (for model monitoring). Loss is optimized using ADAM which estimates first and second order moments dynamically.

For our application, we are interested in an architecture that have extremely fast inference time (for potentially being as System on Chip) which also does not compromise on accuracy. A popular modern architecture fitting these constraints is a ResNet[11]. This architectures has models of different depths - ResNet18 (with 18 layers), ResNet50, ResNet101 (with 50 and 101 layers, respectively). Here, we considered the ResNet50, which has demonstrated SOTA performance with ImageNet[8], two models with far fewer trainable parameters and less layers, ResNet18 and MobileNetV2[16], to investigate tradeoffs between model complexity and inference time. In the following paragraphs, we will explain the concepts and intuition behind the ResNets and the MobileNetV2. ResNet18 has 11.26M parameters and the MobileNetV2 has 3.9M parameters, compared with 211M trainable parameters on ResNet50.

The premise and motivation for development of the ResNet architecture was the observation that as the depth of neural networks increased, the model was not able to leverage the increased number of parameters to learn more complex patterns, and performance abruptly reduced. One of the reasons this for this was vanishing gradients over larger number of layers, leading to no learning, as most gradients are set close to zero. ResNets have a mechanism to retain the learning capabilities of deep networks with large number of layers and in the best case improve them significantly. The basic building block for a ResNet is a residual block; this concept is explained in the equations below:

$$x_{l+k} = x_l + F(x_l, W_l, W_{l+1}, W_{l+k})$$
$$x_{l+k} = ReLu(x_{l+k})$$

In the above set of equations $F(x_l, W_l, W_{l+1}, W_{l+k})$ is the output of a k layer convolutional sub-network. For this architecture we add the initial input to the convolutional block consisting of k layers to get the output at the $k+1$th layer. With this formulation, during training, the optimization works such that $F(x_l, W_l, W_{l+1}, W_{l+k})$ (the convolutional sub network function) is minimized or close to zero. The additional advantage is during backpropagation, as the gradients don't zero out due to vanishing gradients with this new formulation. ResNet18 and ResNet50 are both architectures which evolve out of this concept.

MobileNets are aimed at building models that train on device, and hence are very lean in the number of parameters. There are two models with this architecture - MobileNetv1 and MobileNetv2 that builds on the former. They are based on the concept of depth-wise separable convolutions. In depth wise convolutions a different filter is learned separately on each channel of the image, and is stacked going into the next layer. A 1x1 convolution completes the block. With depth wise convolutions combined with a 1x1 convolution, the dimensionality is the same as a regular convolution, however the number of parameters is lower by a factor of the filter dimension. Over many layers, this optimization leads to significantly lower memory requirements. MobileNet v2 builds on this concept by adding residual connections between the 1x1 convolution layers, thus integrating the advantages of residual connections (motivated by ResNets) and the low memory footprint of MobileNetV1. It is also important to note that both ResNets and MobileNets use Batch Normalization which also helps in their training.

5. Experiments and Results

Here, we train cell type classifiers and evaluate their performance and properties.

5.1. Cell Classes

The 36 cell classes in this project include 11 different human cell lines, 10 blood cell types isolated from mouse bone marrow, 7 different sub-cellularly localized versions of the same fluorescent label in a single cell type (HeLA cells, Fig 1B), 5 unique combinations of fluorescent stains in HeLa cells, and one case:control drug treatment response test in HeLa cells (Appendix). We expect some of these should be easily separable and some should be difficult to separate. Cells with different fluorescent label combinations should

be separated by intensity on different fluorescent channels, for example, while the mouse bone marrow cells are more challenging to separate because they are classified to the level of cell subtypes, such as CD4+ T Helper Memory cells and CD4+ T Helper Effector Cells. The drug perturbation should also be quite challenging, as it isn't known to have a visible signature on cell.

5.2. Classification with 7-Channel Images

We first tested how well the two model architectures, ResNet and MobileNetV2, performed on the classification task using the 7-channel images. After 50 epochs of training, the MobileNetV2 achieved 78% balanced accuracy (Appendix) and the ResNet50 model achieved the best performance, an 85% balanced accuracy on the test set images (Fig 2) (see Appendix for detailed metrics on other models). This was better than team biologist (Jarod) expected. The model classifies the majority of cell types nearly perfectly, but has trouble with a few cell types in ways that are biologically expected and even interesting. The class the model has the most trouble with is CD4+ Th cells/ill defined, which is as the name suggests, a population of CD4+ T cells that could not correctly be identified into T cell sub-types based on traditional FACS. Interestingly the model resolves this class of reject cells largely into the 2 known CD4+ T cell types Memory, and Naive cells, and a a CD8 T cell lineage which is thought to be closely related, CD8 Naive T cells. It also keeps some cells as undefined. Could this suggest that there is an undetected cell state within this more heterogenous population? An image-based sort could discover new cell type biology by sorting this group for experimental follow-up.

The other major mis-classification, we discovered, is actually a cell class error, as the Fig_2B_S4D_HeLa and Fig_s3_HelA cells are stained the same way, so they should be the same population. Interestingly, the Fig_2D_S4F_and Fig_2D_S4F_Golgiuntr, which are the drug treated and untreated HeLA condition, have near perfect classification accuracy. Overall, we believe these results are very promising and suggest that with additional data and training this modeling approach can be powerful for cell type classification.

5.3. Pre-Training Models

We also asked if pre-training our models on image-net and then fine-tuning on the ICS data would improve classification performance, as has been in many other modalities. Natural images are quite different from ICS cellular images, so it may not be the case here. Indeed, we found no performance boost by fine-tuning pre-trained models across all ResNet and MobileNetV2 architectures we tested. The best performing pretrained ResNet50 had a balanced accuracy of 84% compared to the 85% for the randomly initialized model (Fig 3)

5.4. Reducing Image Channels

Using a "brightfield image" alone (the light loss in our data), most scientists would have difficulty classifying any of these cell types because most cells look alike to the human eye in suspension. Some cell classes here have additional fluorescent labels added, which should give them a more unique profile across the fluorescent channels. However, using fluorescent channels requires an additional cell-labeling step that is not feasible for many applications. Previous literature [15] suggests that machine learning could improve the enable label-free classification of cells. If this method could achieve high performance it could be revolutionary for many applications of FACS. To test this, we built models using only the fluorescence-label free channels: light loss, forward light scatter, and side light scatter, and models only using the light loss ("brightfield") image, and evaluated the performance. Naively, we could expect the models to perform poorly without the signals that guide traditional FACS.

Excitingly, the 3 channel ResNet50 model achieved a balanced accuracy of 73% across the 36 classes, many of which the only known distinguishing between them was actually a fluorescent label (Fig 4). This is particularly true for all of the Fig_1C classes, which all were the same cell type but targeted with organelle-specific green fluorescent dyes. This result suggests that ICS light imageing can pick up subtle light or structure/morphology changes between these very minor perturbations. The Fig_2D_S4F_and Fig_2D_S4F_Golgiuntr classes also do better than would be expected, with 80% correct classification.

The 1 channel ResNet50 model performed worse, with a test accuracy of 35% (Appendix). Of course, this is top 1 accuracy, after only 50 epochs of training, and in an very difficult task, so even this is quite encouraging given the amount of information we are discarding with this approach.

5.5. Inference Time

Since this technology is eventually applied to cell sorting on device, fast inference time is of high importance. We therefore wanted to test the prediction performance of multiple different architectures to understand speed/accuracy trade-offs. We tested 2 sizes of ResNets, as well as a MobileNetV2 (Fig 5), which has fewer parameters and is typically deployed in computation-constrained settings for good performance and speed.

We measured the inference time for one datapoint by running inference on 10,000 randomly-generated tensors of the appropriate shape and calculating the mean inference time. The inference time does not include the GPU warm-up time. The mean inference times and respective standard deviations are shown in Figure 5.

We observed that there was no significant difference in inference time when using a different number of channels

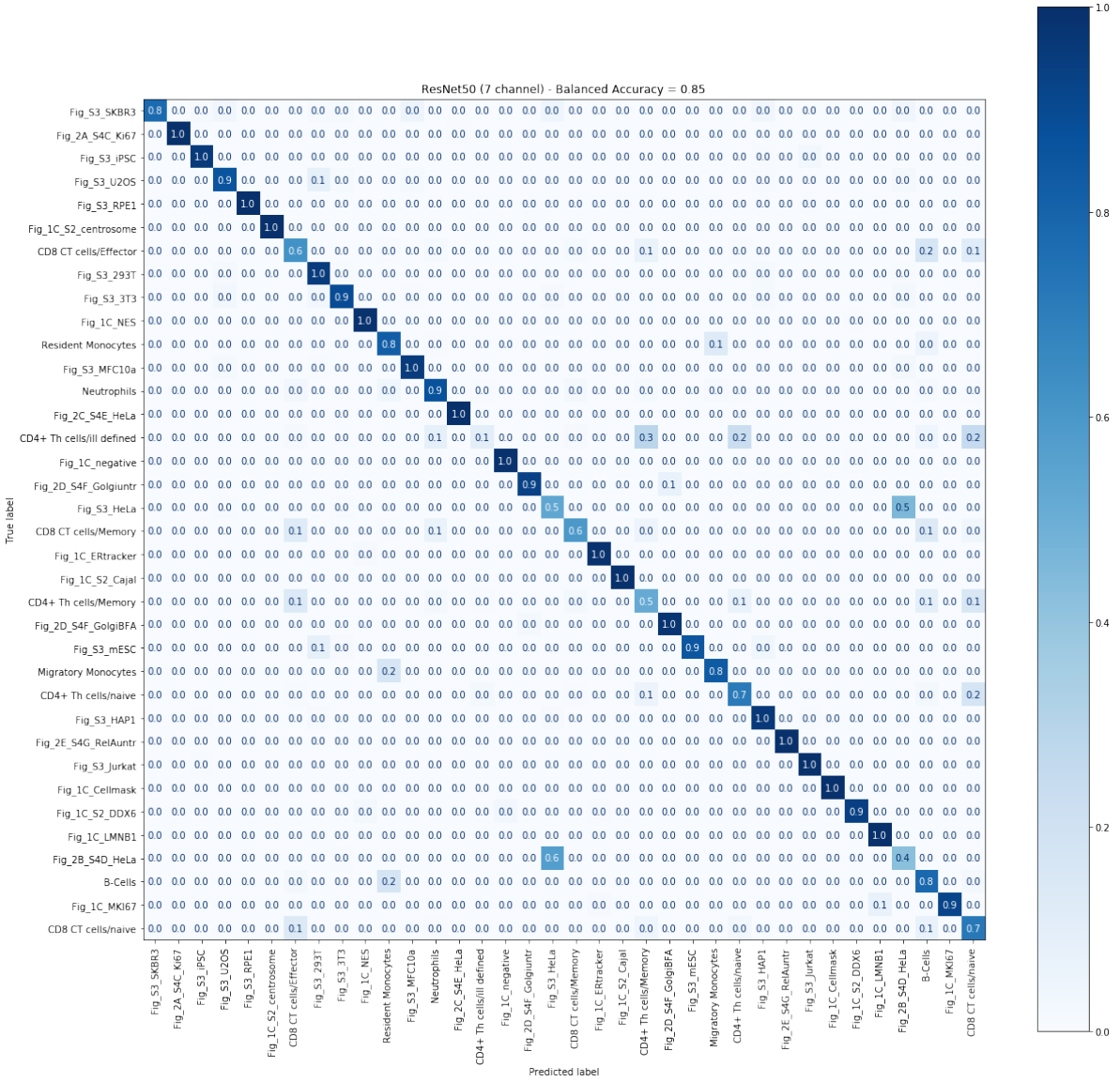


Figure 2. Confusion matrix for 7-channel ResNet50 classification balanced accuracy across 36 cell classes

(i.e., differently-shaped tensors) with a given model architecture. Additionally, unsurprisingly, there was no significant difference between pretrained models and models that were trained solely on our data: after all, they had an equal number of parameters and computational complexity.

Interestingly, the ResNet18 architecture had the lowest inference time at 3.23 ms, averaged across the 6 configurations, to make a prediction on a single image. MobileNetV2 had an average inference time of 5.47 ms, despite having fewer parameters than ResNet18. This may be attributed to the fact that the MobileNetV2 architecture "uses depthwise separable convolutions which are not directly supported in GPU firmware", resulting in larger inference time than an architecture like ResNet18 [14]. However, the MobileNetV2-based models outperform ResNet18-based models and have about a third of the parameters, making them an attractive choice for memory-constrained settings.

Finally, ResNet50 had the highest average inference time of 7.74 ms, which was unsurprising given the relatively large number of parameters it uses. However, ResNet50-based models also had the best performance, achieving a balanced accuracy of 0.85 with the randomly-initialized 7-channel configuration.

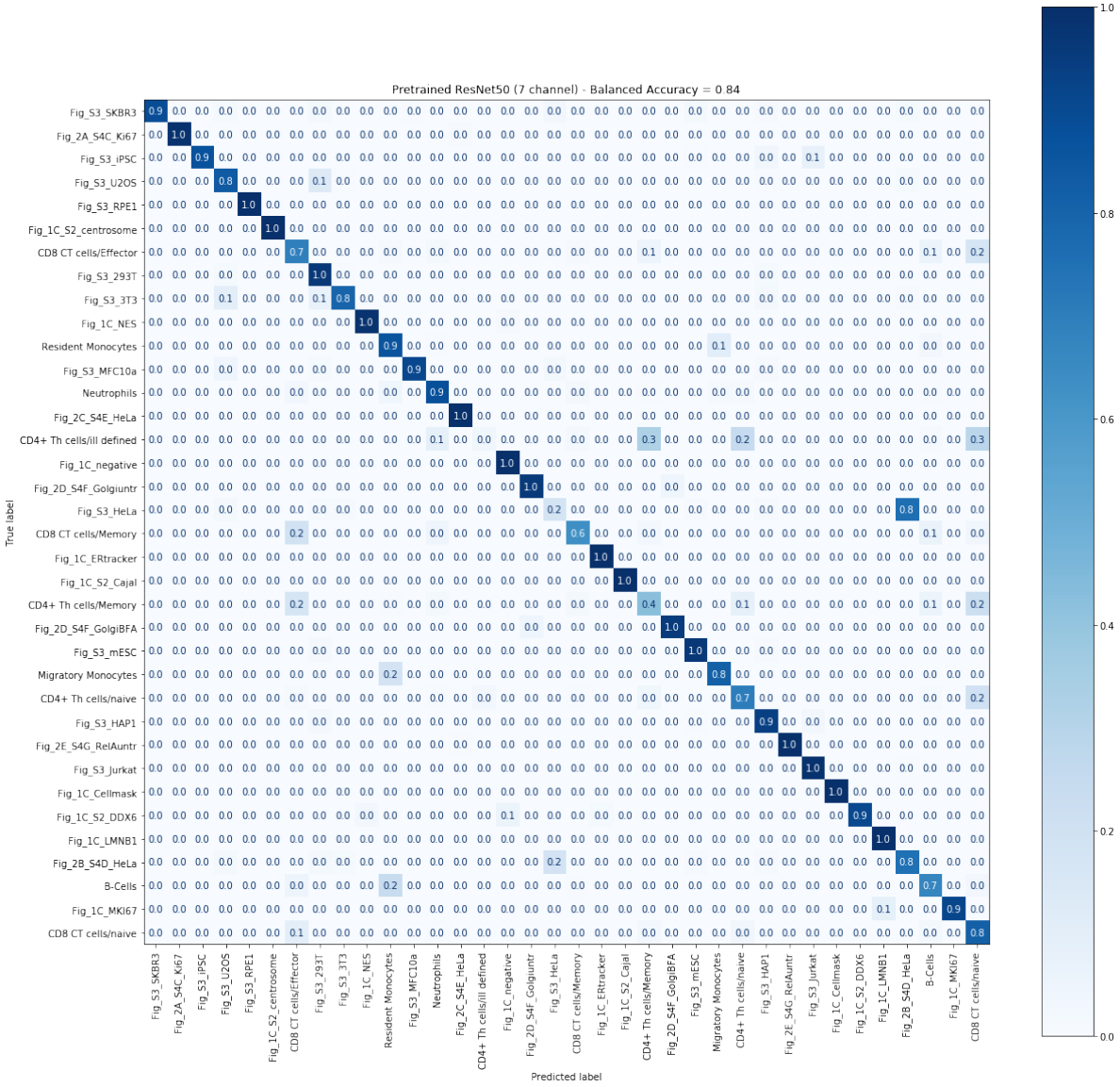


Figure 3. Confusion Matrix for 7-Channel ResNet50 Pre-Trained on ImageNet: Classification balanced accuracy across 36 cell classes

5.6. Saliency Mapping

Finally, we also tested if saliency mapping on particular cell classes and interesting failure cases could increase the interpretability of these models with the goal of learning new cell biology. We performed saliency mapping for the 7-channel ResNet50 model and examined the CD4+ Th Cell/ill defined class, which is not resolvable by traditional FACS. Here we compared the saliency map for an ill-defined cell that has been misclassified as CD4+ Th Cell/Naive (Figure 6) and the saliency map for CD4+ Th Cell/Naive that has been correctly classified as such (Figure 7). We have visualized the max gradient over all 7 channels, and the 3 light loss channels to see the cell. In both cases the saliency map is very difficult to interpret, and seems to depend on features outside of the cell. More experimentation is needed to determine if this is an artefact from the training, or a true informative feature of these images.

6. Conclusions and Future Directions

In this research project, we explored the feasibility of using deep convolutional models to classify multiple cell types and cell states from a new high-throughput cellular imaging modality called Image Cell Sorting. We tested ResNet18, ResNet50, and MobileNetV2 models, compared untrained models with those trained on natural images, built models with reduced light-only image information, mea-

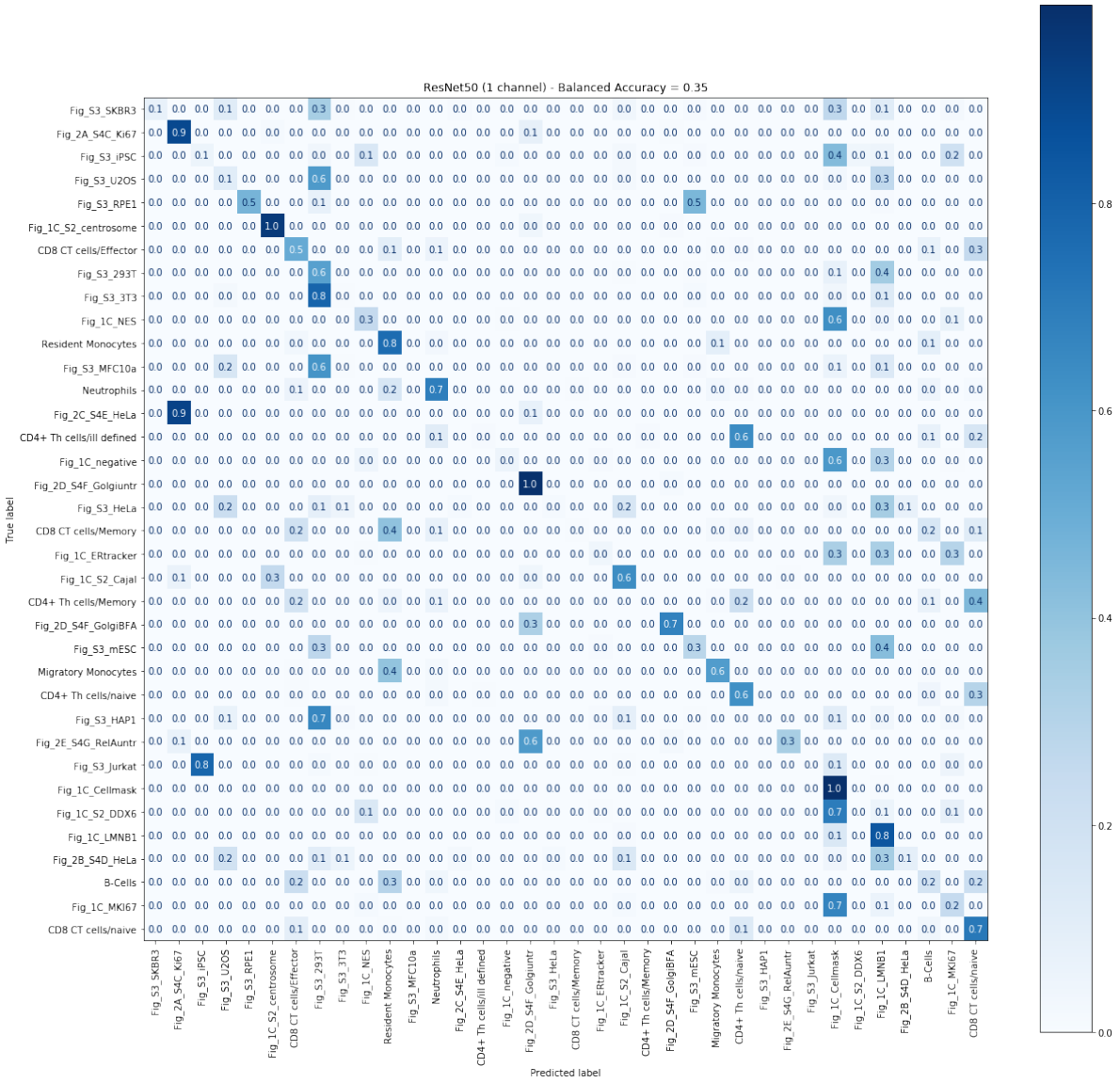


Figure 4. Confusion Matrix for 3-Channel ResNet50 classification balanced accuracy across 36 cell classes

sured the inference time of different architectures on a GPU, and explored saliency mapping as a method for model interpretation.

Overall, we are excited by the results. Our best performing model, a ResNet50, demonstrated impressive classification performance in a difficult classification task, and even did so without fluorescence channel information. This highlights the power of image data as a high-content data type for biology, and should justify further research into this topic. Our MobileNetV2 model also performed quite well, with slightly better performance than the ResNet18 despite having 7,384,064 less parameters. For scientific instruments where memory space is a concern, this could be a significant advantage. However, the MobileNetV2 did not have faster inference time, which was our number one criteria of interest beyond model performance.

In the future, there are a number of exciting directions to go. Given that we had difficulty interpreting saliency maps on these images, exploring other interpretation methods like integrated gradients, DeepLift, and ablation experiments could be more promising alternatives to understand these images. There are also other architectures and preprocessing steps to try which could be helpful. For example, doing an image segmentation step before our classifier could be useful for getting rid of the large amount of blank space in each image, which we don't expect to contribute

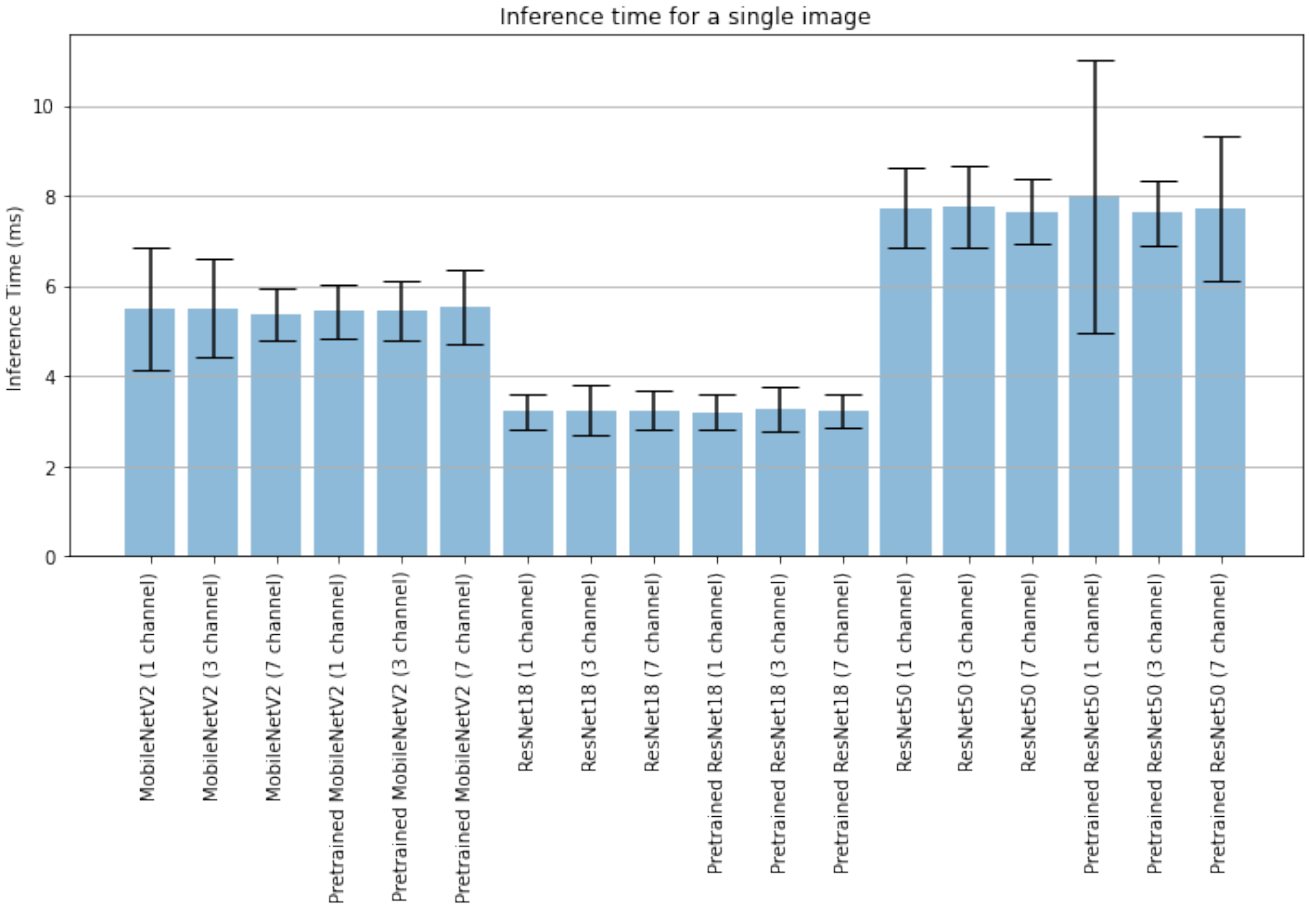


Figure 5. Average inference time for various model configurations.

meaningfully to cell type or cell state differences. Implementing alternative efficient architectures, such as EfficientNet_B0, would also be interesting to see if we could push down the inference time to sub-millisecond without sacrificing performance. Efficient models of this sort combined with additional acceleration using TPUs would be capable of sorting 10000+ images/sec in real time, taking full advantage of deep learning on the ICS system.

7. Contributions Acknowledgements

J.R conceived of the study and experiments. J.R., T.G. and T.J. wrote all code and wrote/edited the manuscript. J.R. provided biological interpretation of findings. Special thanks to Daniel Schraivogel and Lars Steinmetz, who provided unpublished data and guidance on understanding the image type. Special thanks to the Montgomery Laboratory for providing data storage and compute resources.

References

[1] J. C. Caicedo, J. Roth, A. Goodman, T. Becker, K. W. Karhohs, M. Broisin, C. Molnar, C. McQuin, S. Singh, F. J. Theis, and A. E. Carpenter. Evaluation of Deep Learning Strategies for Nucleus Segmentation in Fluorescence Images. *Cytometry Part A*, 95(9):952–965, 2019. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cyto.a.23863.

[2] S. N. Chandrasekaran, B. A. Cimini, A. Goodale, L. Miller, M. Kost-Alimova, N. Jamali, J. Doench, B. Fritchman, A. Skepner, M. Melanson, J. Arevalo, J. C. Caicedo, D. Kuhn, D. Hernandez, J. Berstler, H. Shafqat-Abbasi, D. Root, S. Swalley, S. Singh, and A. E. Carpenter. Three million images and morphological profiles of cells treated with matched chemical and genetic perturbations. Technical report, bioRxiv, Jan. 2022. Section: New Results Type: article.

[3] T. Ching, D. S. Himmelstein, B. K. Beaulieu-Jones, A. A. Kalinin, B. T. Do, G. P. Way, E. Ferrero, P.-M. Agapow, M. Zietz, M. M. Hoffman, W. Xie, G. L. Rosen, B. J. Lengerich, J. Israeli, J. Lanchantin, S. Woloszynek, A. E. Carpenter, A. Shrikumar, J. Xu, E. M. Cofer, C. A. Lavender, S. C. Turaga, A. M. Alexandari, Z. Lu, D. J. Harris, D. DeCaprio, Y. Qi, A. Kundaje, Y. Peng, L. K. Wiley, M. H. S. Segler, S. M. Boca, S. J. Swamidass, A. Huang, A. Gitter, and C. S. Greene. Opportunities and obstacles for deep learning in biology and medicine. *Journal of The Royal Society*

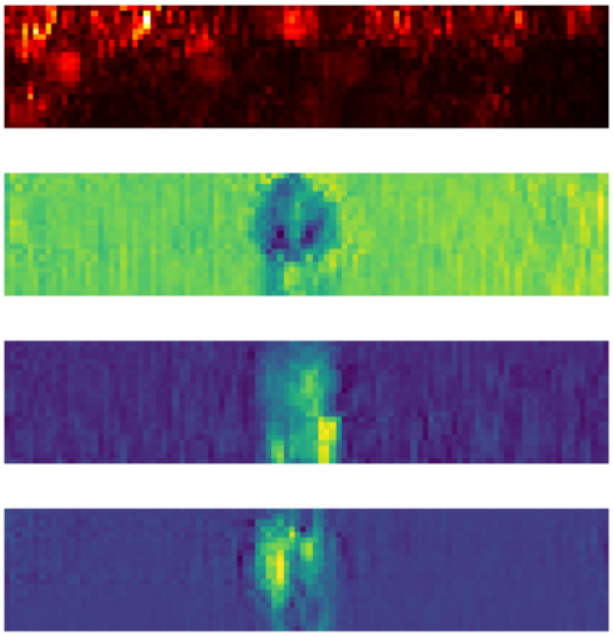


Figure 6. Saliency map for 'CD4+ Th cells/ill defined' sample predicted as 'CD4+ Th cells/naive'

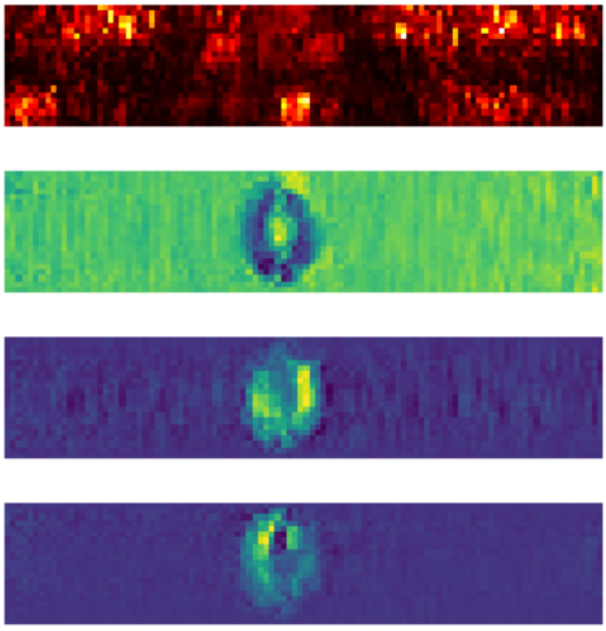


Figure 7. Saliency map for 'CD4+ Th cells/naive sample' predicted as 'CD4+ Th cells/naive'

Interface, 15(141):20170387, Apr. 2018. Publisher: Royal Society.

[4] Y. L. Chow, S. Singh, A. E. Carpenter, and G. P. Way. Predicting drug polypharmacology from cell morphology readouts using variational autoencoder latent space arithmetic. *PLOS Computational Biology*, 18(2):e1009888, Feb. 2022. Publisher: Public Library of Science.

[5] P. Clute and J. Pines. Temporal and spatial control of cyclin B1 destruction in metaphase. *Nature Cell Biology*, 1(2):82–87, June 1999. Number: 2 Publisher: Nature Publishing Group.

[6] A. e. a. Cossarizza. Guidelines for the use of flow cytometry and cell sorting in immunological studies. *Eur. J. Immunol.*, 49:1457–1973, 2019.

[7] M. F. Cuccarese, B. A. Earnshaw, K. Heiser, B. Fogelson, C. T. Davis, P. F. McLean, H. B. Gordon, K.-R. Skelly, F. L. Weathersby, V. Rodic, et al. Functional immune mapping with deep-learning enabled phenomics applied to immunomodulatory and covid-19 drug discovery. *bioRxiv*, 2020.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[9] E. D. Diebold, B. W. Buckley, D. R. Gossett, and B. Jalali. Digitally synthesized beat frequency multiplexing for sub-millisecond fluorescence microscopy. *Nature Photonics*, 7(10):806–810, Oct. 2013. Number: 10 Publisher: Nature Publishing Group.

[10] P. Goldsborough, N. Pawlowski, J. C. Caicedo, S. Singh, and A. E. Carpenter. CytoGAN: Generative Modeling of Cell Images. Technical report, bioRxiv, Dec. 2017. Section: New Results Type: article.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] J. Hung, A. Goodman, D. Ravel, S. C. P. Lopes, G. W. Rangel, O. A. Nery, B. Malleret, F. Nosten, M. V. G. Lacerda, M. U. Ferreira, L. Rénia, M. T. Duraisingh, F. T. M. Costa, M. Marti, and A. E. Carpenter. Keras R-CNN: library for cell detection in biological images using deep neural networks. *BMC Bioinformatics*, 21(1):300, July 2020.

[13] N. Nitta, T. Sugimura, A. Isozaki, H. Mikami, K. Hiraki, S. Sakuma, T. Iino, F. Arai, T. Endo, Y. Fujiwaki, H. Fukuzawa, M. Hase, T. Hayakawa, K. Hiramatsu, Y. Hoshino, M. Inaba, T. Ito, H. Karakawa, Y. Kasai, K. Koizumi, S. Lee, C. Lei, M. Li, T. Maeno, S. Matsusaka, D. Murakami, A. Nakagawa, Y. Oguchi, M. Oikawa, T. Ota, K. Shiba, H. Shintaku, Y. Shirasaki, K. Suga, Y. Suzuki, N. Suzuki, Y. Tanaka, H. Tezuka, C. Toyokawa, Y. Yalikun, M. Yamada, M. Yamagishi, T. Yamano, A. Yasumoto, Y. Yatomi, M. Yazawa, D. Di Carlo, Y. Hosokawa, S. Uemura, Y. Ozeki, and K. Goda. Intelligent Image-Activated Cell Sorting. *Cell*, 175(1):266–276.e13, Sept. 2018.

[14] M. Orsic, I. Kreso, P. Bevandic, and S. Segvic. In defense of pre-trained imagenet architectures for real-time semantic segmentation of road-driving images. *CoRR*, abs/1903.08469, 2019.

[15] S. Ota, R. Horisaki, Y. Kawamura, M. Ugawa, I. Sato, K. Hashimoto, R. Kamesawa, K. Setoyama, S. Yamaguchi, K. Fujiu, K. Waki, and H. Noji. Ghost cytometry. *Science*,

360(6394):1246–1251, June 2018. Publisher: American Association for the Advancement of Science.

[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. 2018.

[17] D. e. a. Schraivogel. High-speed fluorescence image–enabled cell sorting. *Science*, 375:315–320, 2022.

Appendix A: Class Distribution

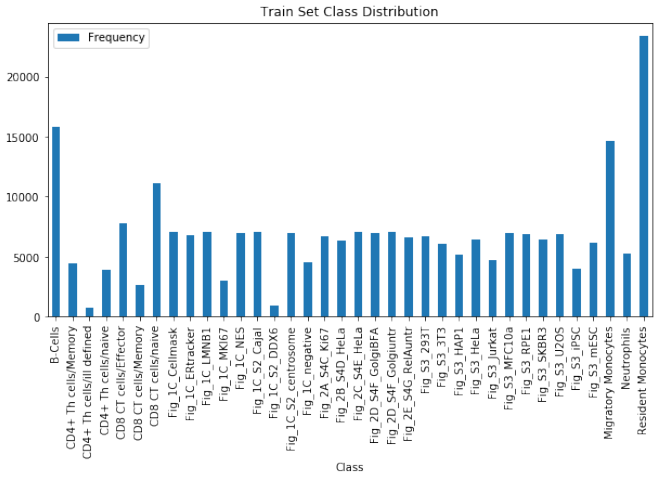


Figure 8. Class distribution in the train set

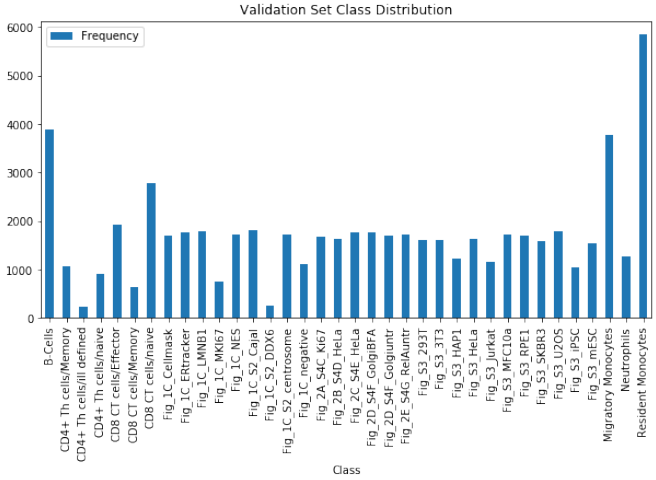


Figure 9. Class distribution in the validation set

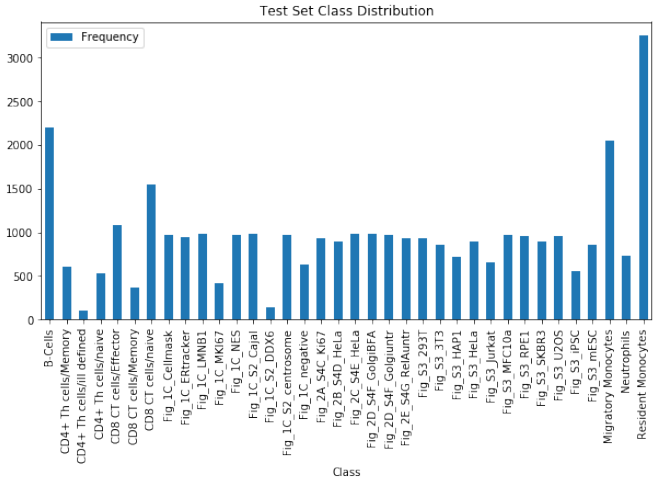
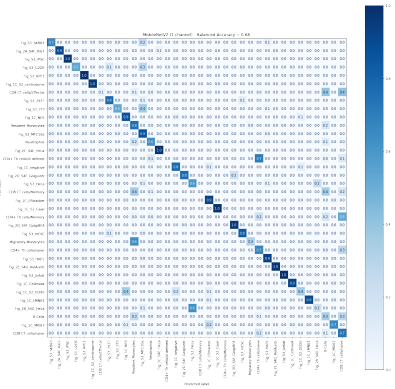
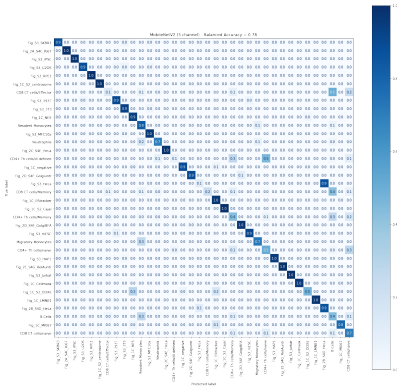


Figure 10. Class distribution in the test set

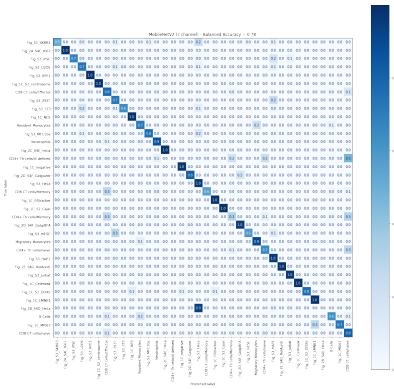
Appendix B: Confusion Matrices



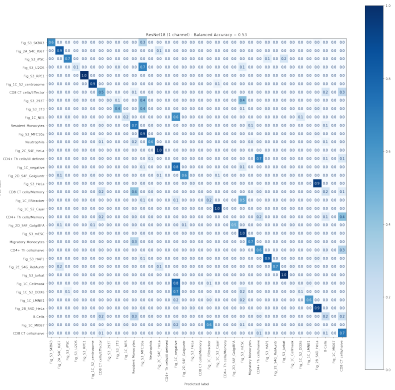
MobileNetV2 (1 channel)



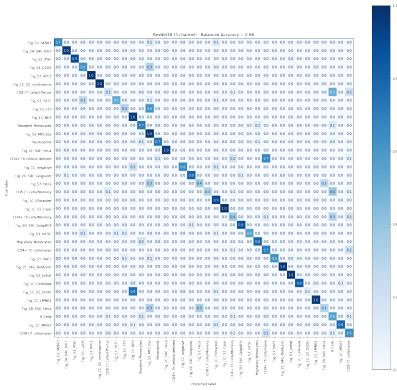
MobileNetV2 (3 channel)



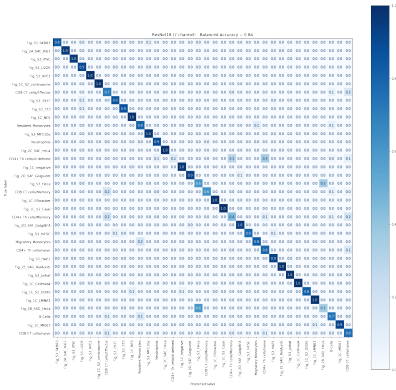
MobileNetV2 (7 channel)



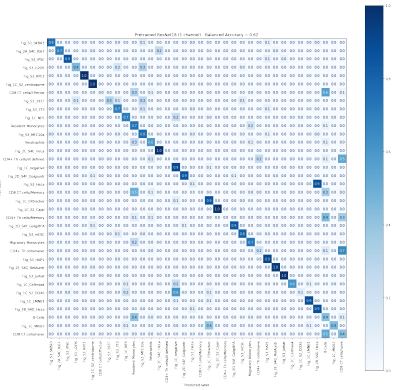
ResNet18 (1 channel)



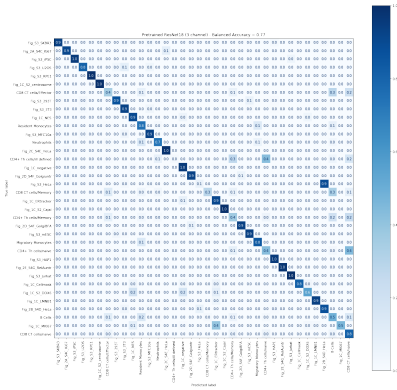
ResNet18 (3 channel)



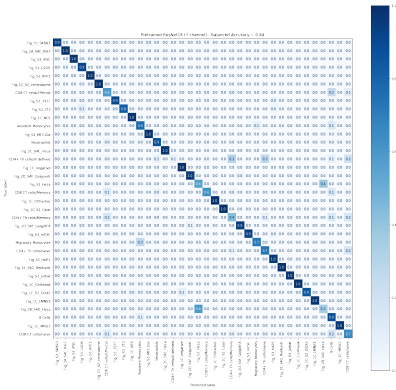
ResNet18 (7 channel)



Pretrained ResNet18 (1 channel)

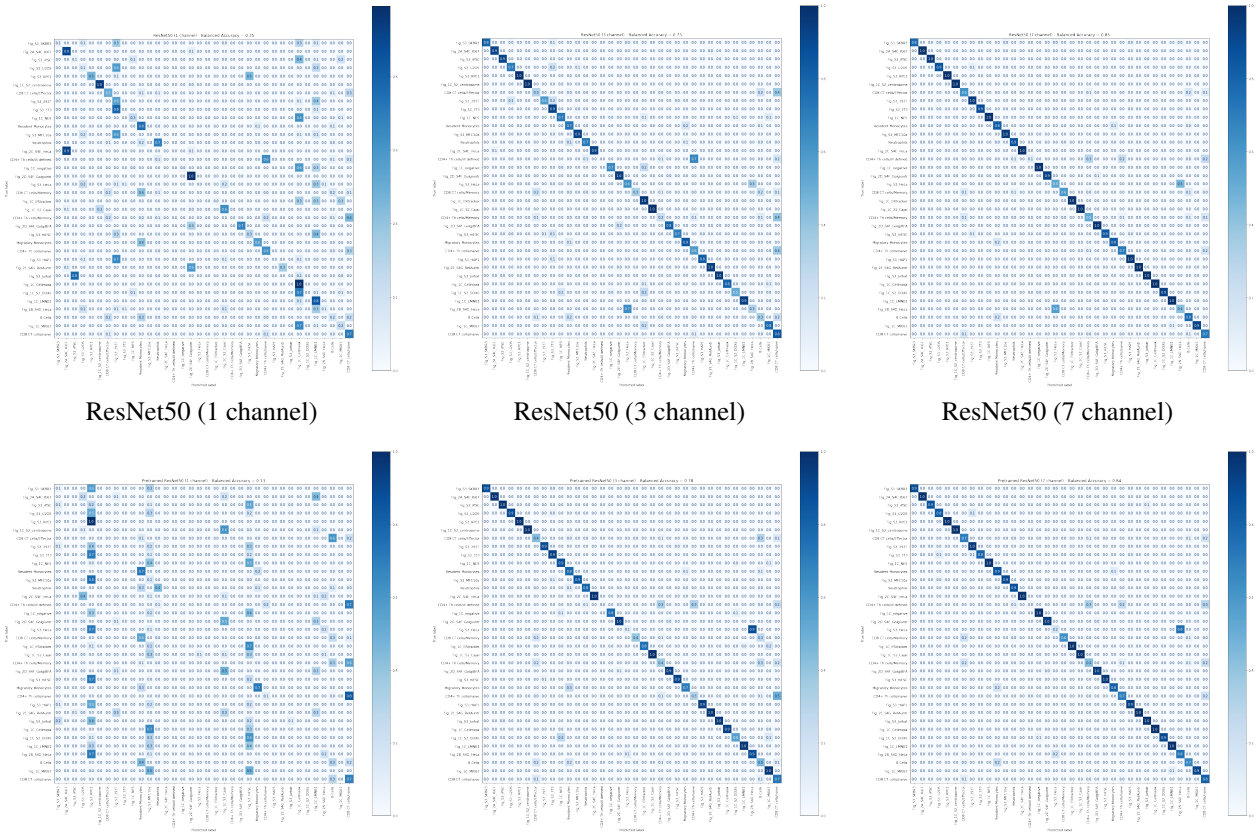


Pretrained ResNet18 (3 channel)



Pretrained ResNet18 (7 channel)

Figure 11. Confusion matrices for different classifiers.



ResNet50 (1 channel) ResNet50 (3 channel) ResNet50 (7 channel)

Pretrained ResNet50 (1 channel) Pretrained ResNet50 (3 channel) Pretrained ResNet50 (7 channel)

Figure 12. Confusion matrices for different classifiers.