Using convolutional neural networks for supervised classification of ImageStream images in R

Hefin Rhys[1], Asheigh Howes[2], Paul Rees[3], Andreas Wack[2], Andy Riddell[1]

1. Flow Cytometry STP, Francis Crick Institute, London; 2. Immunoregulation group, Francis Crick Institute, London; 3. Enginering, Swansea University

1. Introduction

The ImageStream imaging cytometry platform allows us to capture brightfield, scatter, and fluorescence images of thousands of cells per minute. The spatial data generated facilitates the discrimination of cell types based on their morphological, as well as biochemical, differences.

Sometimes, morphologically-distinct cell types are present in a sample, that are difficult to discriminate using manual gating. If humans can identify patterns in images, it's likely that computer vision algorithms, such as convolutional neural networks (convnets) can do the same.

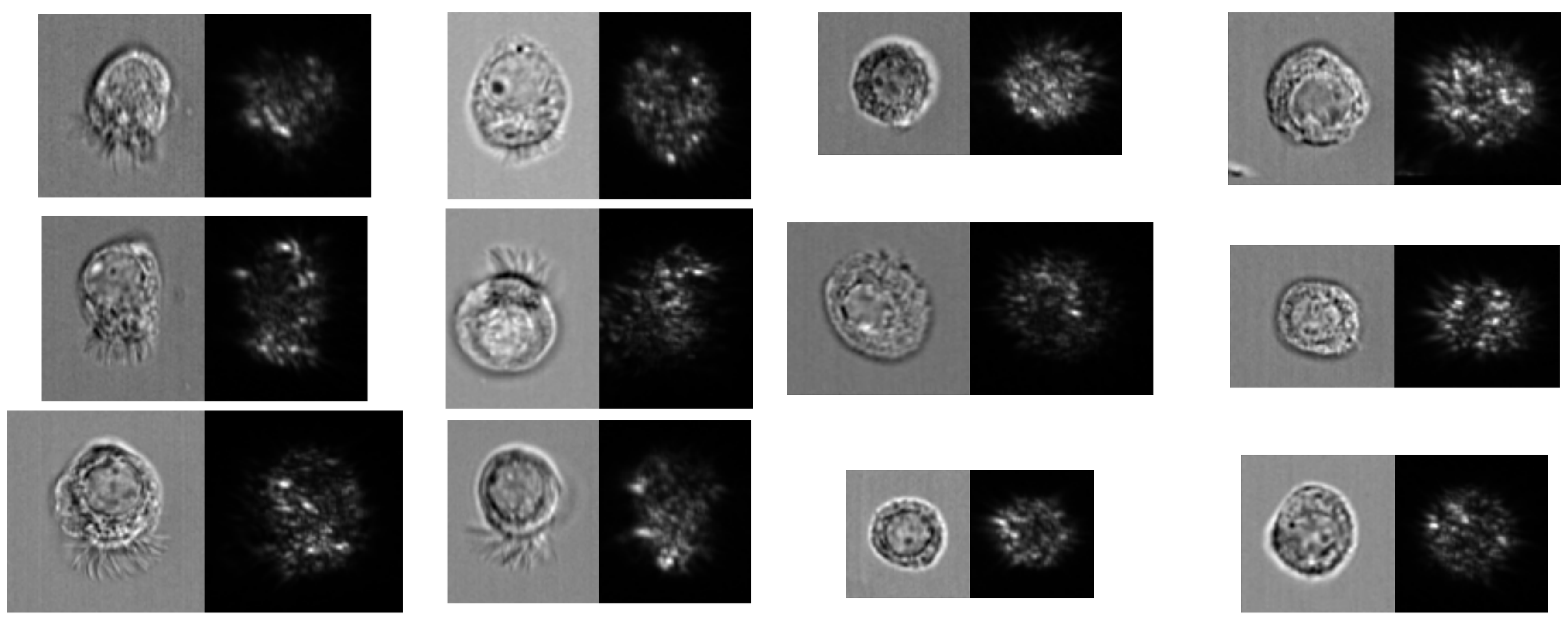
The goal of this project was to set up a pipeline to facilitate the training of convnet machine learning models, to perform supervised classification of ImageStream images, using an example problem.

2. The problem: ciliated or unciliated?

The example problem used to set up and test the pipeline was a simple, binary classification problem: in a mixed sample of ciliated and unciliated epithelial cells, can a convnet be trained to distinguish between them, using only their brightfield (BF) and side scatter (SSC) images?

Ciliated | Unciliated

BF SSC BF SSC | BF SSC BF SSC



3. The approach

Manually tag and export images from IDEAS
(unpadded, 8-bit, channels 1, 6, and 9)

↓

Merge channels of each cell into single .tif
(using batch Fiji script)

↓

Split images into train, validate, and test sets
(50:50 split between classes in each)

88 | 36 | 30

↓

Reshape images into arrays of 150 x 150 x 3

↓

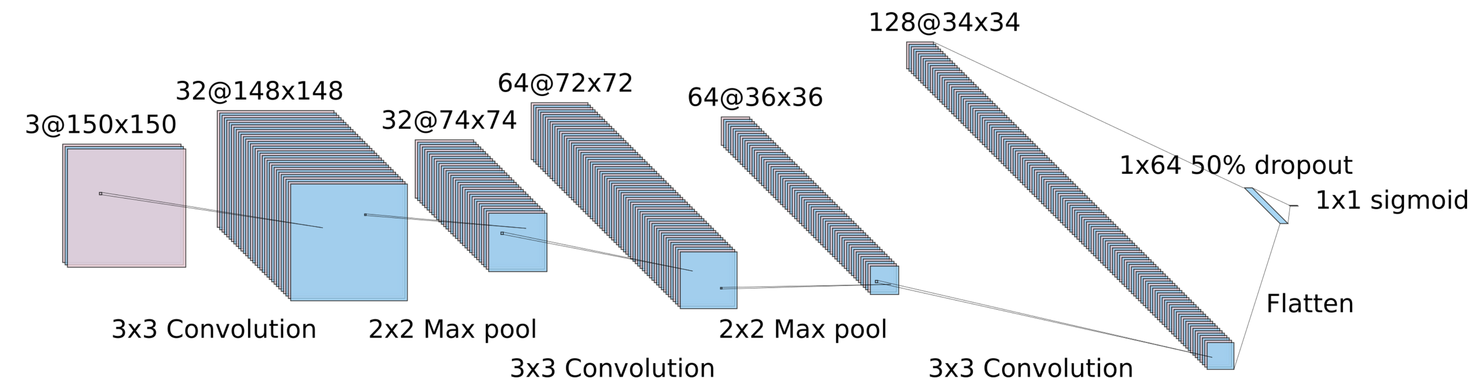
Train convnet from scratch → without data augmentation; with data augmentation

Use pre-trained network to extract image features → without fine tuning; with fine tuning

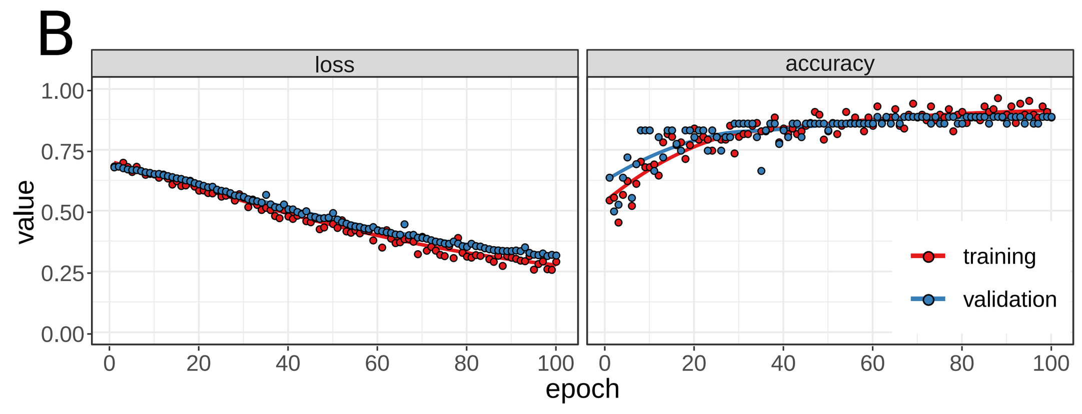
Compare performance on test set

4. Results: training a convnet from scratch

A



B



C

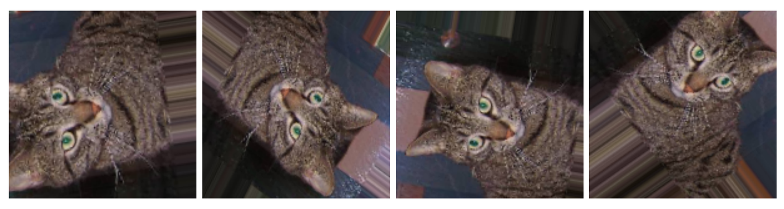
Truth \ Predicted	Ciliated	Unciliated
Ciliated	11	4
Unciliated	2	13

Accuracy = 0.800
Precision = 0.733
Recall = 0.846
F_1 = 0.786

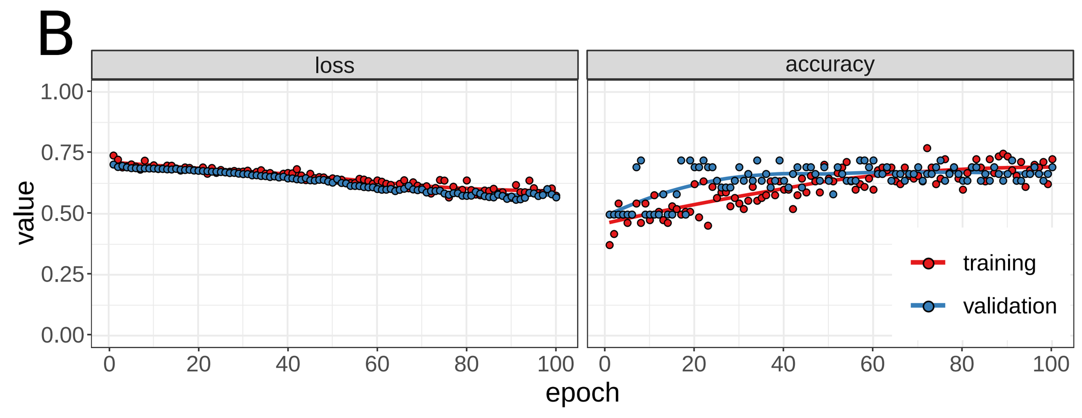
A Graphical representation of the manually-defined convnet with three convolutional and two max-pooling layers.
B Loss (binary cross entropy) and accuracy for training and validation sets for each training epoch.
C Confusion matrix and performance metrics when model is trained for 100 epochs and evaluated on the test set.

5. Results: including data augmentation

A Data augmentation is the process of applying transformations to image data (such as rotation, flipping, and rescaling) to help the network learn from more varied images, without collecting more data.



B



C

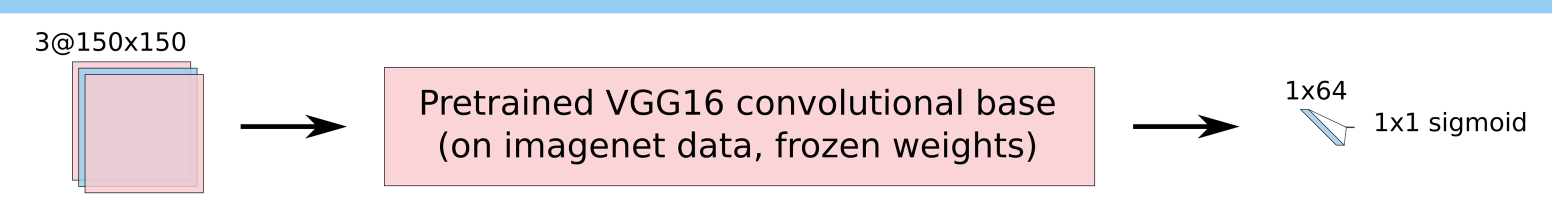
Truth \ Predicted	Ciliated	Unciliated
Ciliated	8	7
Unciliated	4	11

Accuracy = 0.633
Precision = 0.533
Recall = 0.667
F_1 = 0.593

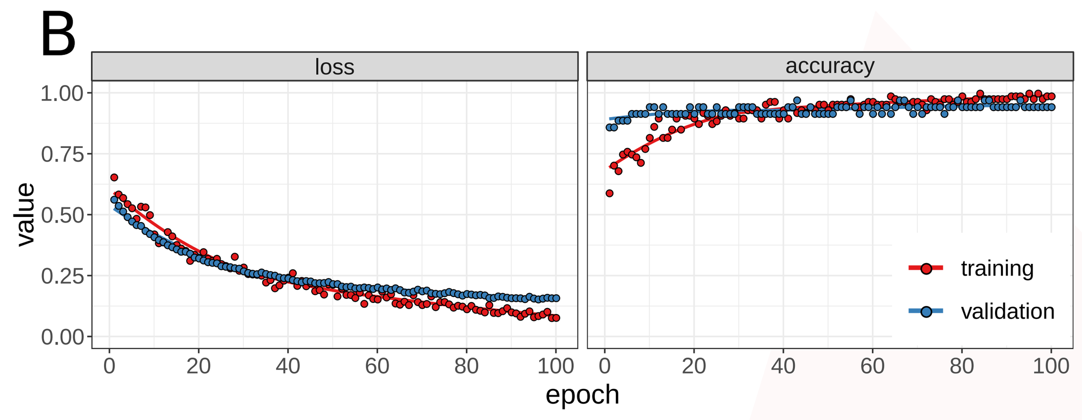
A Explanation and example of data augmentation.
B Loss and accuracy for training and validation sets for each training epoch.
C Confusion matrix and performance metrics when model is trained for 100 epochs and evaluated on the test set.

6. Results: using a pre-trained network

A



B



C

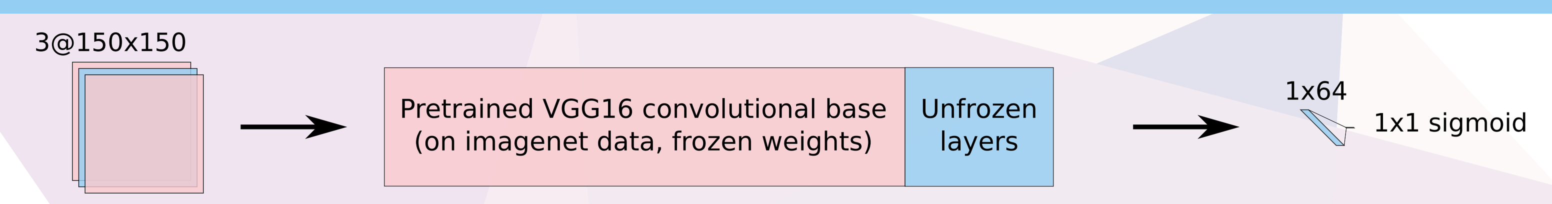
Truth \ Predicted	Ciliated	Unciliated
Ciliated	13	2
Unciliated	0	15

Accuracy = 0.933
Precision = 0.867
Recall = 1.000
F_1 = 0.929

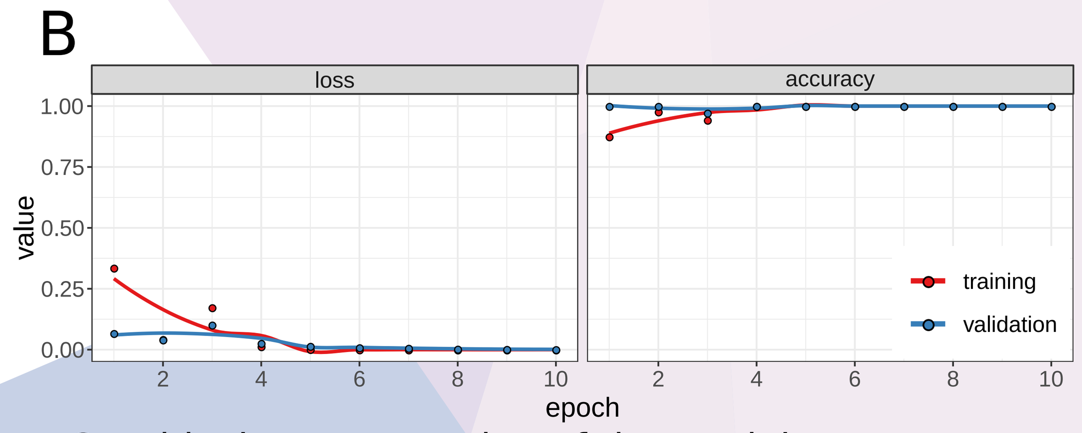
A Graphical representation of the model.
B Loss and accuracy for training and validation sets for each training epoch.
C Confusion matrix and performance metrics when model is trained for 40 epochs and evaluated on the test set.

7. Results: fine tuning a pre-trained network

A



B



C

Truth \ Predicted	Ciliated	Unciliated
Ciliated	15	0
Unciliated	0	15

Accuracy = 1.000
Precision = 1.000
Recall = 1.000
F_1 = 1.000

A Graphical representation of the model.
B Loss and accuracy for training and validation sets for each training epoch.
C Confusion matrix and performance metrics when model is trained for 10 epochs and evaluated on the test set.

8. Conclusions

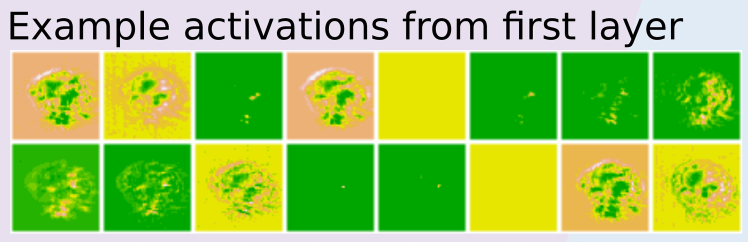
1. Convnets can learn effective models for supervised classification of ImageStream images

2. Data augmentation deteriorated model fit by introducing bias in the absence of overfitting

3. Using a model pre-trained on a large image dataset is more effective for small datasets

4. Fine tuning a pretrained model gave excellent performance, by tailoring the model to the problem

Example activations from first layer



Example activations from final layer