Electronic Supplementary Material (ESI) for Lab on a Chip.

Supplementary Information for

‘Deep learning guided image-based droplet sorting for on-demand selection and analysis of single cells and 3D cell cultures’

Vasileios Anagnostidis[1], Benjamin Sherlock[1], Jeremy Metz[1], Philip Mair[2], Florian Hollfelder[2], and Fabrice Gielen[1]

[1] University of Exeter, Living Systems Institute, Exeter, United Kingdom
[2] University of Cambridge, Department of Biochemistry, Cambridge, United Kingdom

Contents

1. Comparison with other label-free droplet sorting studies

We reviewed the most relevant sorting methods published for post-encapsulation droplet sorting, selecting only label-free techniques. CNN based sorting performs in line with previous studies or favourably for throughput and classification accuracy. The main advantage of our technique lies in its adaptability to variation in visual appearance, object sizes and the possibility to perform co-encapsulations with diverse objects.

	Method for object identification	**Efficiency of single cells identification**	**Throughput**	**Resolution**	**Adaptability to variation in visual appearance, object sizes and possibility to perform co-encapsulations**
Image-based methods					
This work	Convolutional neural networks	80-90%	40 Hz	478 x478	Yes
[1]	Image binarization and segmentation	NA	100 Hz	72x72	Difficult
[2]	Template matching algorithm	~ 90%	10 Hz	Not mentioned	Not shown
Non-image-based methods					
[3]	Acoustic waves	97%	40 Hz	NA	No
[4]	Hydrodynamic instabilities	~70-80%	160 Hz	NA	No
[5]	~60-78%	~60-78%	5 kHz	NA	No

Table S1. Comparison of CNN based droplet sorting for micro-encapsulation with other published studies using label-free droplet-based techniques.

2. Detailed microdevice fabrication procedure

The device was fabricated following classical soft-lithography procedures by using a high-resolution acetate mask (Microlithography Services Ltd.) Negative photoresist SU-8 3050 (MicroChem, Newton, MA) was deposited onto clean silicon wafers to a thickness of 80 µm and patterned by exposure to UV light through a transparency

photomask. Prior of immersing the master in propylene glycol monomethyl ether acetate (PGMEA, Sigma-Aldrich) for development, a second layer of SU-8 2100 was applied for coating and UV exposed for the development of a serpentine channel (height, 150 µm; width, 150 um diameter). Uncured polydimethylsiloxane (PDMS) consisting of a 10:1 polymer to cross-linker mixture (Sylgard 184) was poured onto the master, degassed, and baked at 70 ℃ for 4 hours. The PDMS mould was then cut and peeled from the master, punched with a 1 mm biopsy punch (Kai Medical) for inlet ports, and plasma bonded (Diener Zepto) to a microscope slide (76 x 26 mm, 1.00 – 1.2 mm thick, Academy Science). Hydrophobic surface treatment was performed after bonding by flushing with 1% (v/v) Trichloro (1H, 1H, 2H, 2H-perfluorooctyl) silane (Aldrich) in HFE-7500 and, subsequently, placed in a 65 ℃ oven for 30 min.

3. Optical setup

The trinocular port of the inverted microscope was used for triggering, imaging, and recording of high-speed videos. The distances between the optical elements are depicted in Figure S1.

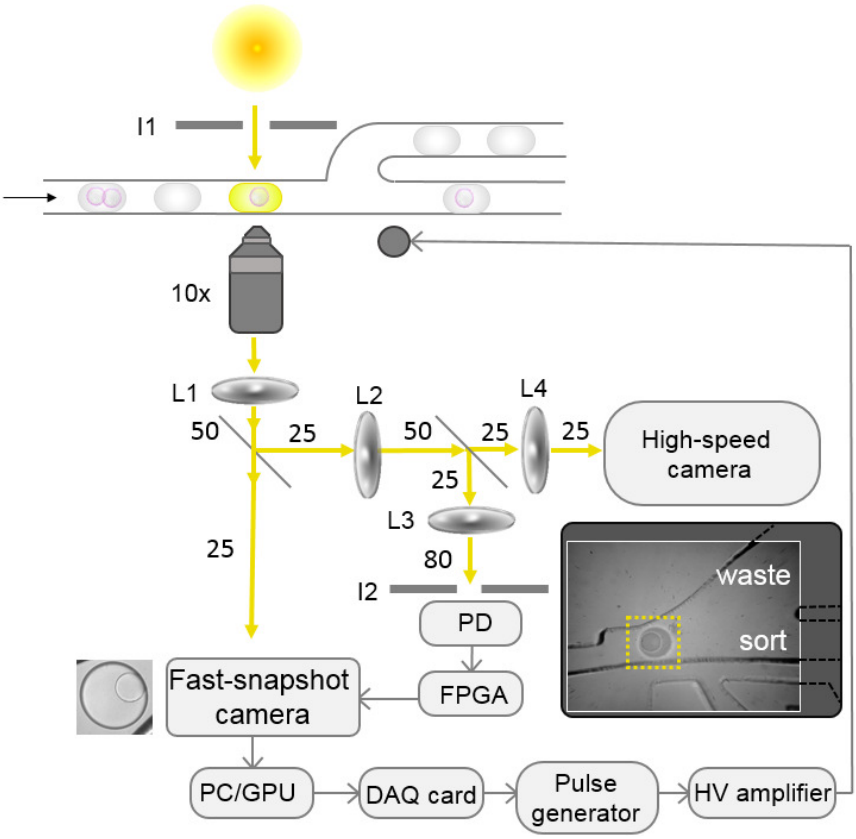

Figure S1. Optical setup with location of the optical elements. The lengths are indicated by numbers placed along the light path and are expressed in millimetres. Plano-convex lenses had focal lenses of 50 mm for L1 and L2, 25.4 mm for L3 and L4.

4 Accuracy and number of training classes

We tested whether CNNs could classify images into two (‘non single object’ and ‘single object’) rather than three classes (‘0’,’1’and ‘>1’ object). Figure S2 shows that for a typical CNN architecture (as shown in Figure 2), accuracy reduces slightly for PA (1% loss) and MCF7 (5% loss), more significantly for PS (~17% loss). The training images consisted of 200 single objects and 2x 200 non single objects. We kept this class imbalance for validating the models.

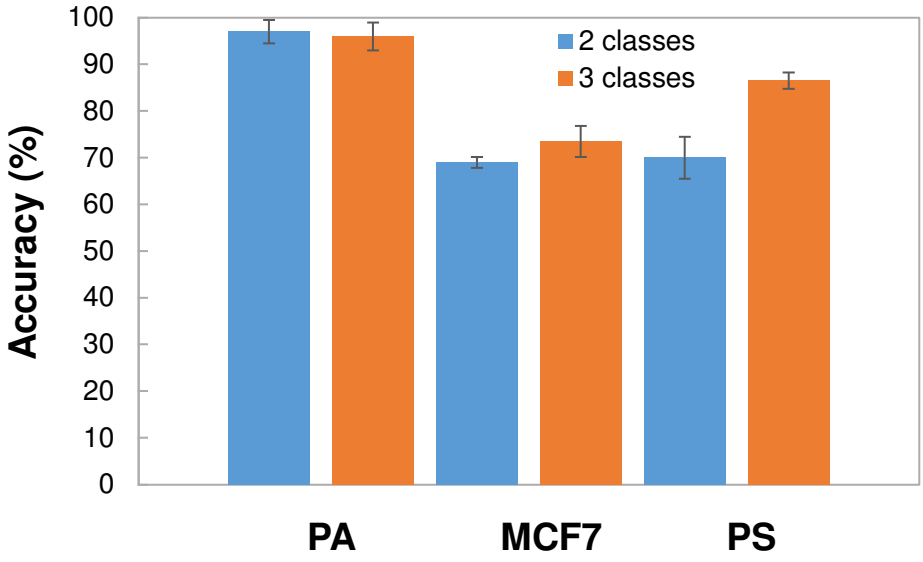

Figure S2. Accuracy versus number of training classes (n= 3).

5. Timing for training and single image evaluation for real-time classification

5.1 Real-time saving of images

Images could be acquired and saved to an SSD drive and a circular mask was applied to exclude irrelevant parts of the images outside the droplets. Frame grabbing time was found to be below 1 ms for all image sizes. The saving time, however, increased exponentially with the pixel number and was typically 22 ms for 480 x 480 pixels (Figure S3A).

5.2 Comparison between single CPU and single GPU

We compared time performance between a CPU and a GPU for both training and testing CNNs. Therefore, after acquisition and labelling of the training data, the offline CNN training was done using either a CPU or a GPU. The time it took for training using

the GPU was typically 7 minutes for 478 x 478 pixels images with 1200 images per class and 10 epochs (each image was passed 10 times through the network). By contrast, the CPU required 200 minutes to perform the same task (Figure S3B). After training, the CNN was applied in real-time on single images being received by the area scan camera. Figure S3C shows the testing time for a single image using the GPU/CPU. With the GPU, it took 5 ms to process an image independently of image size, while the CPU timing increased exponentially from 5 ms at 50 x 50 pixels to 213 ms for 480 x 480 pixels.

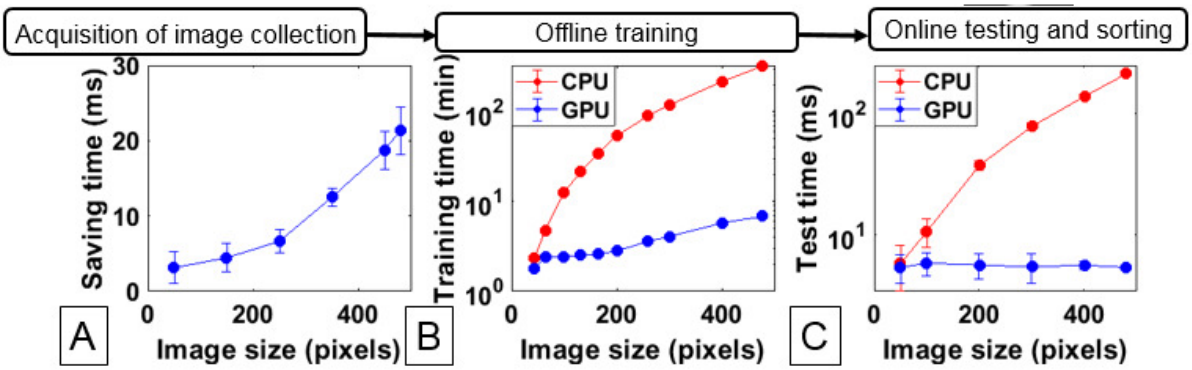

Figure S3. Timings for the real-time saving of images (n= 3) (A), CNN training (B) and single image testing through the network (C). Training time assumes a collection of 3 x 1200 images with 10 epochs.

5.3 Influence of CNN parameters on time performance

The training time using a GPU was assessed with different kernel sizes and number of filters in a 3 convolutional layers CNN with 128 neurons in the first fully connected layer and a 40% dropout rate (Figure S4-A). We also evaluated time performance to classify a single image with a network having 3 convolutional layers with different number of filters and kernel size. The relationship is plotted in Figure S4-B. Significant increases in training and testing time are seen when kernel size exceeds 20 and with higher number of filters.

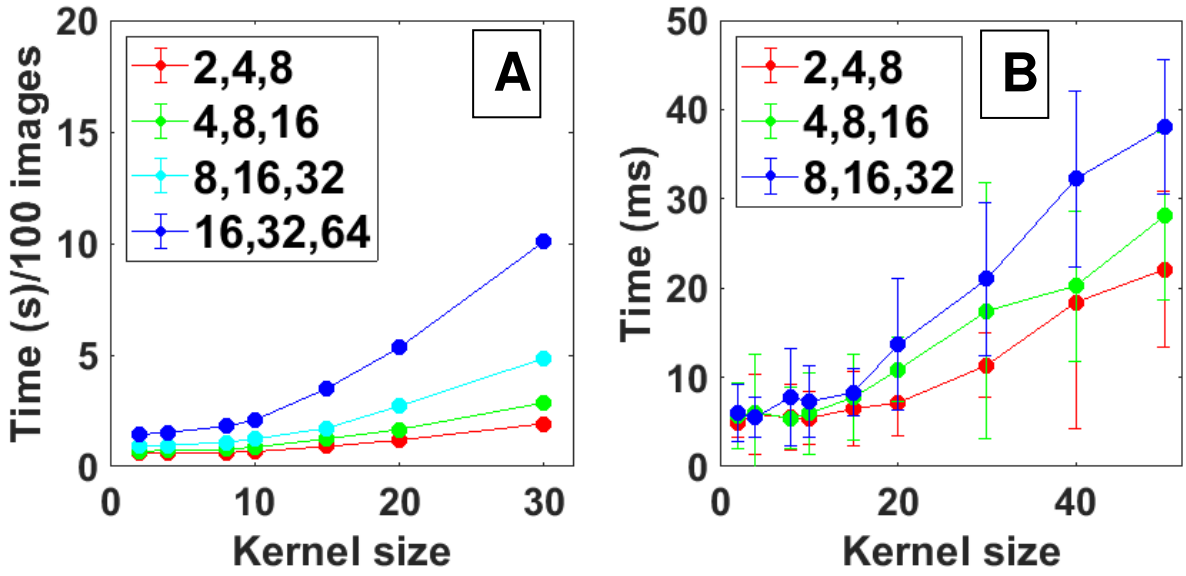

Figure S4. (A) Dependence between training time per 100 images using a GPU and kernel size, number of filters in a 3 convolutional layers CNN (n= 3). (B) Dependence between testing time for a single image and Kernel size and number of filters with a 3 convolutional layers network (n= 3).

6. Optimization of CNN architecture

6.1 Influence of number of filters per convolutional layer

We compared the accuracy on model validations sets for networks with varying depths (number of convolutional layers) from 1 to 4. Small kernel sizes below 5 x 5 pixels did not result in accurate classification, presumably because of the limited type of features that small kernels are able to detect. On the other hand, kernel sizes over 15 x 15 pixels appeared to be too large as they did not improve accuracy any longer or even decreased performance. Single layer architectures could not classify beyond 50-70% accuracy while two classes reached >90% accuracies for PA and PS. Three- and four-layers networks outperformed shallower networks and identified objects with similar accuracies although the loss for PA models was much smaller for four layers models (Figure S5). Accuracies reached close to 100% for PA beads for kernel sizes ranging from 4 to 20 and 3 or 4 layers CNNs, independently of the number of filters. Losses were minimal for kernel size around 15. For MCF7 and PS, the kernel size had to exceed 10 to reach peak accuracy of the models.

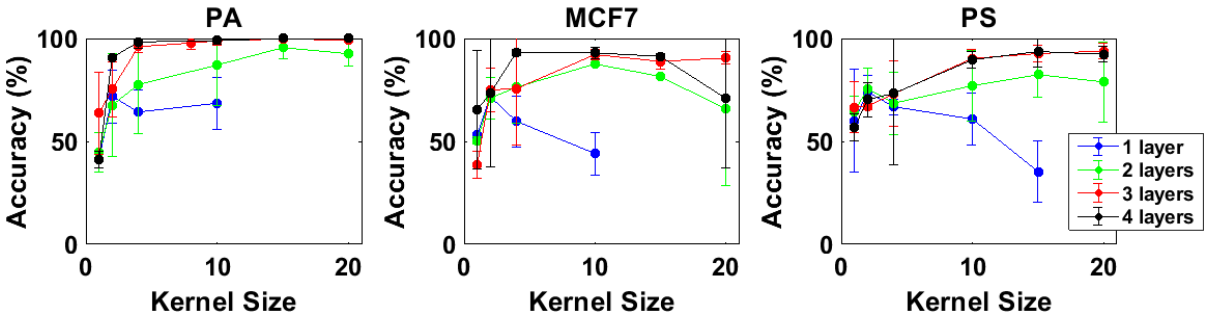

Figure S5. Comparison of model performance depending on the number of convolutional layers and kernel size (n= 3).

Losses associated to training with different network depths for PA, MCF7 and PS are shown in Figure S6. Corresponding accuracies are shown in Figure S5.

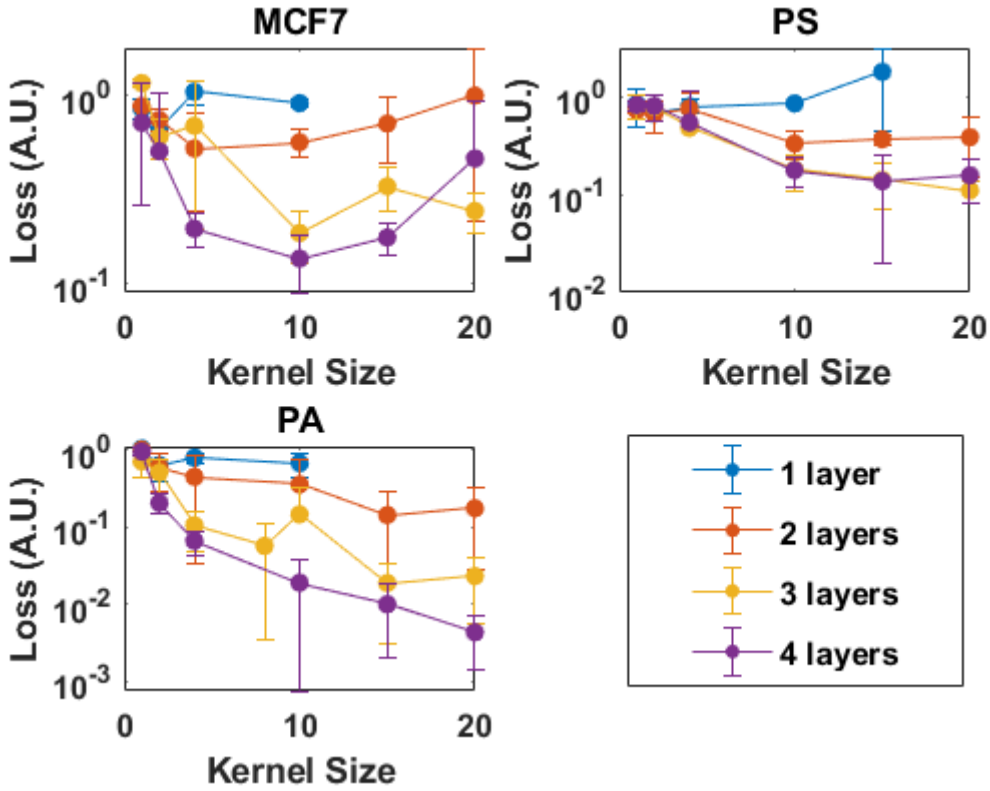

Figure S6. CNN models losses depending on the number of convolutional layers for PA, MCF7 and PS (n= 3).

6.2 Influence of number of filters per convolutional layer

We compared the accuracy on model validation sets for networks with varying number of filters, depths and kernel size. Figure S7 shows that there are no significant differences when changing the number of filters but that the more the depth the better the model.

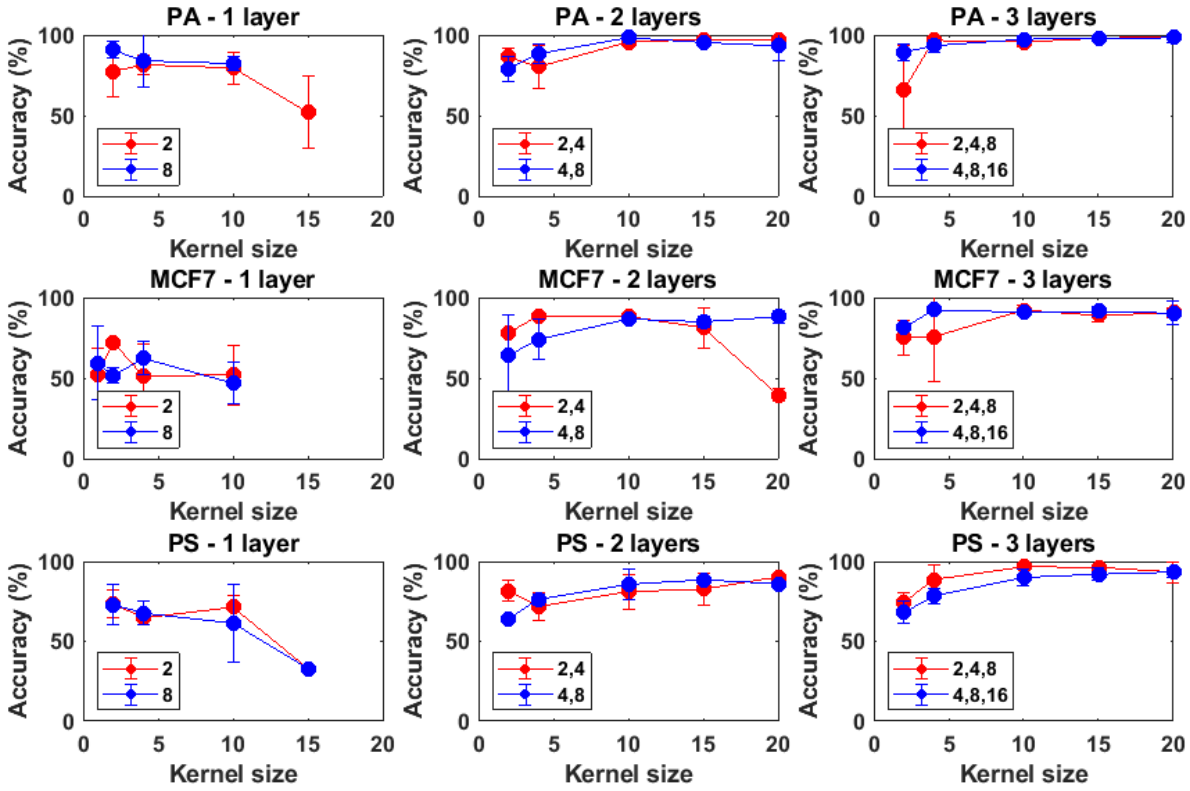

Figure S7. Comparison of model performance depending on the number of filters per convolutional layer for a 1 layer (2 versus 8 filters), 2 layers (2 and 4 versus 4 and 8 filters) and 3 layers (2, 4 and 8 versus 4, 8 and 16 filters) convolutional networks (n= 3).

6.3 Influence of number of neurons in first fully connected layer

The number of neurons in the first fully connected layer was varied from 16 to 256 (Figure S8). The accuracy was found to be weakly dependent on this parameter and we kept 128 neurons for all models.

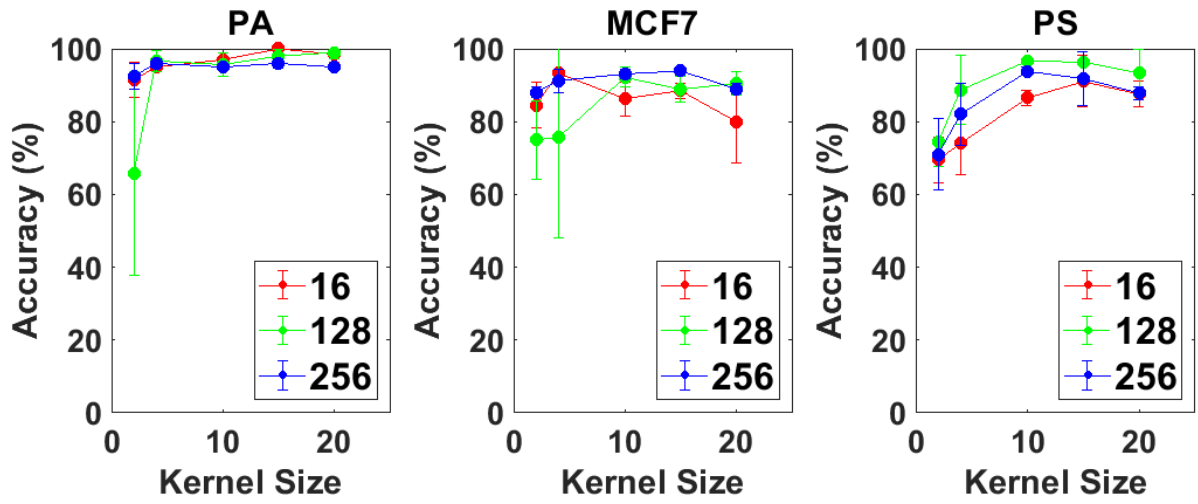

Figure S8. Dependence between number of neurons in the first fully connected layer, kernel size and model accuracy for a 3 convolutional layers network with 2, 4 and 8 filters and a dropout rate of 40% (n= 3).

6.4 Influence of dropout layer rate

The dropout rate was varied from 0 to 80% (Figure S9). The accuracy was found to be weakly dependent on this parameter and we kept 40% for all models.

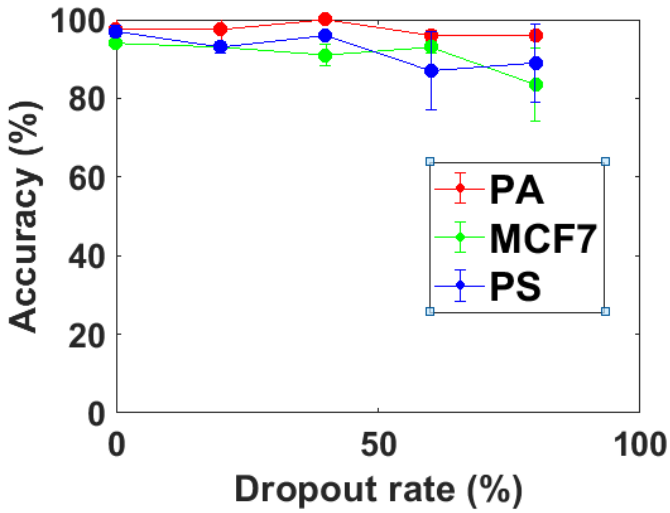

Figure S9. Dependence between dropout rate and model accuracy for a 3 convolutional layers network with 2, 4 and 8 filters and 128 neurons for the first fully connected layer (n= 3).

7. Data augmentation

Data augmentation is a proven technique to provide artificial training data for CNNs. We tested rotations, translations or mirroring and found they could improve model

accuracy. However, while this gain was large when working with few examples (<100), there were diminishing returns with larger training image collections. Interestingly, the loss of the models using translations with PA was an order of magnitude lower compared to other transformations (Figure S10). It is worth pointing out that while accuracies did not reach 100% for both PS and MCF7, images misclassified will often be associated with low class probability and can therefore be excluded in sorting runs. The MCF-7 models performed usually poorly when classifying images of attached cell doublet/dividing cells, presumably because of the lack of training examples.

Losses associated to training with original or augmented training sets for PA, MCF7 and PS are shown in Figure S10. Corresponding accuracies are shown in Figure 3A in the main manuscript.

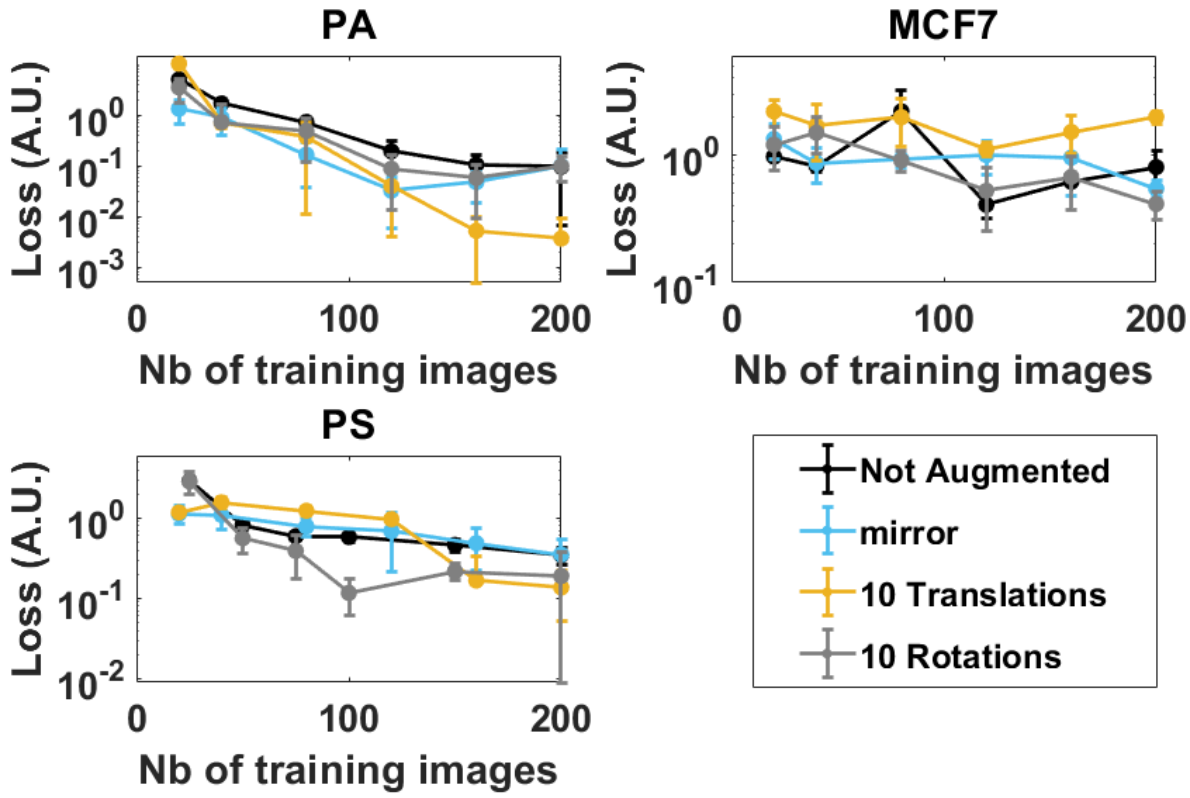

Figure S10. Models losses depending on the type of augmentation (n= 3).

8. Example of activation layers

Figures S11-S13 give examples of the filters at the ReLU activation layers of the networks with networks previously validated on model image sets.

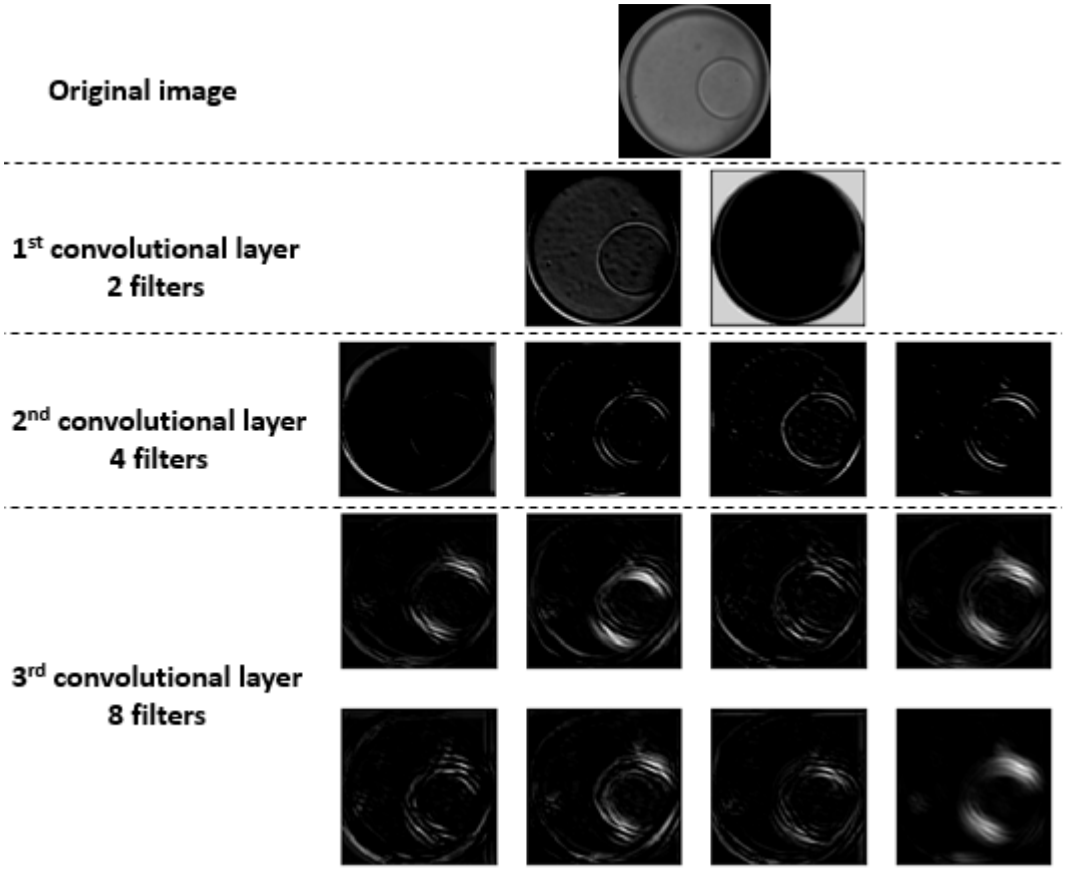

Figure S11. Filters from activation (ReLU) layers for PA beads with a CNN with 3 convolutional layers having 2, 4 and 8 filters respectively.

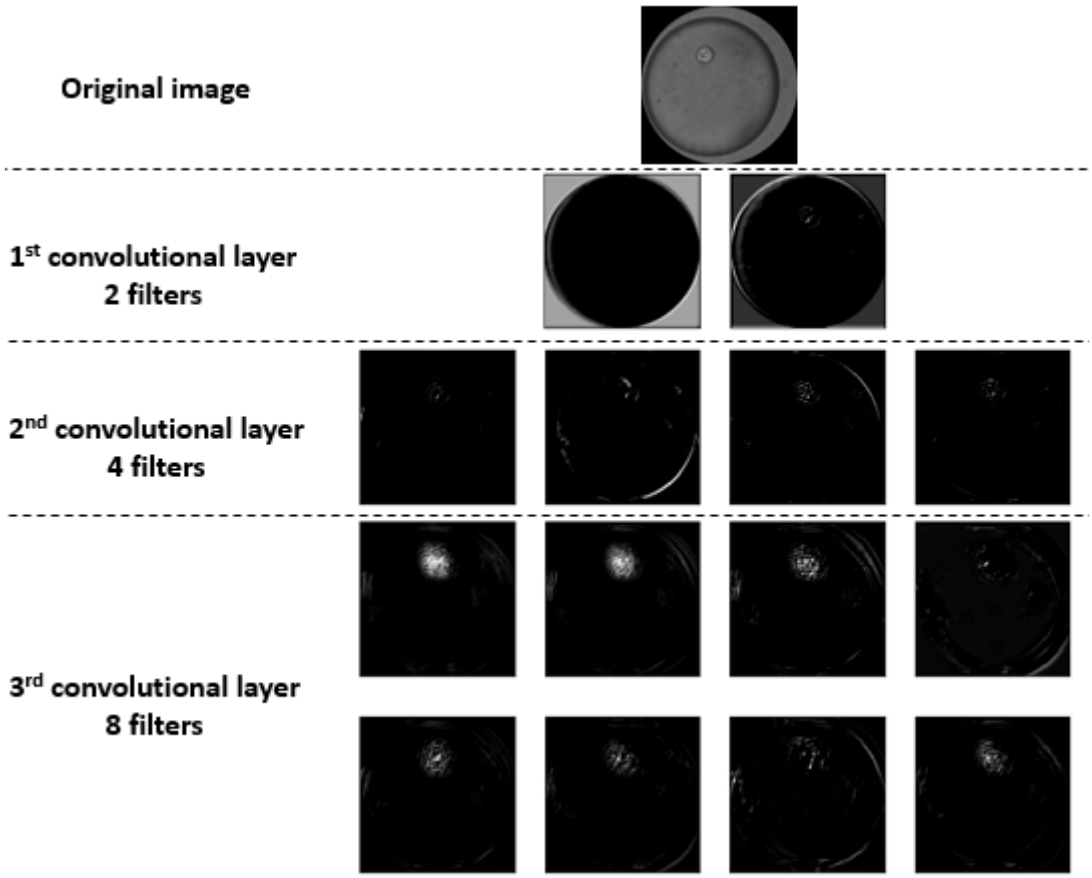

Figure S12. Filters from activation layers for MCF-7 cells with a CNN with 3 convolutional layers having 2, 4 and 8 filters respectively.

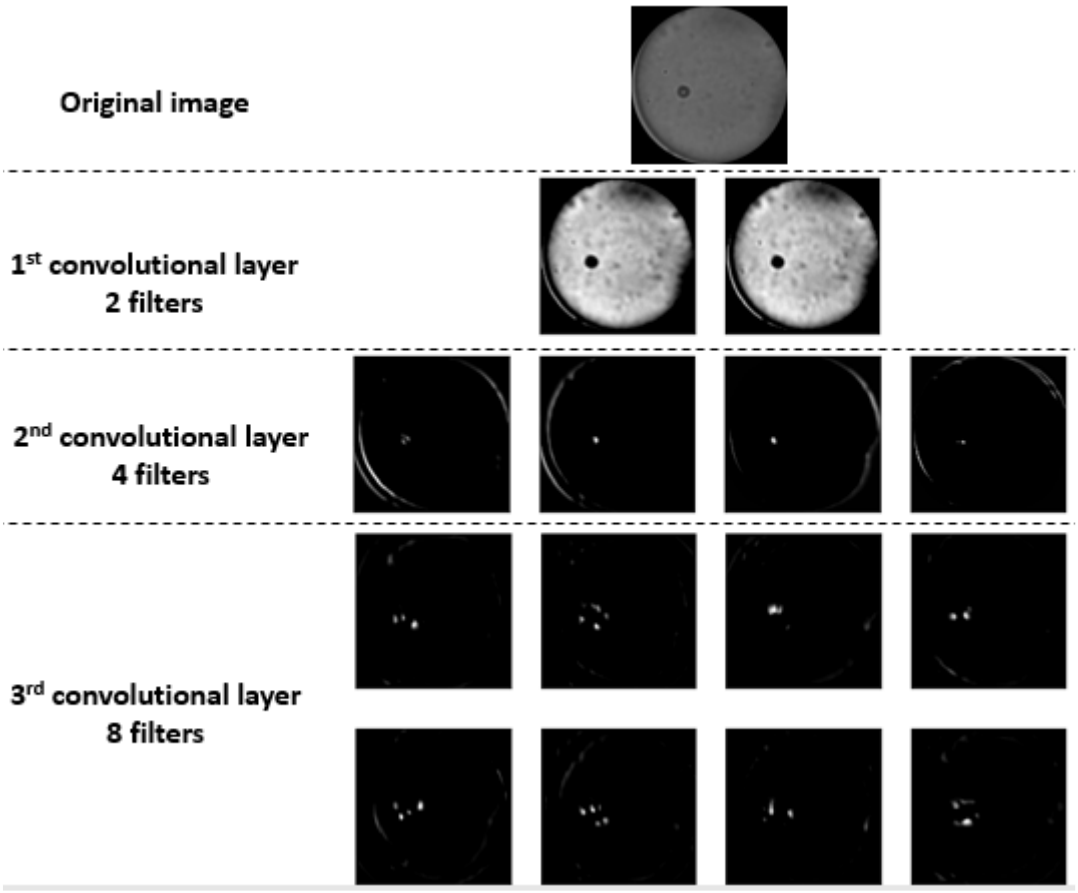

Figure S13. Filters from activation layers for PS spheres with a CNN with 3 convolutional layers having 2, 4 and 8 filters respectively.

9. Robustness of CNN models towards exposure settings

We tested CNN models robustness to exposure settings by assessing the classification of PA beads with 4 different exposure times (30 µs, 50 µs, 100 µs, 300 µs) and 3 different gains (5 dB, 10 dB and 15 dB) with constant light intensity (Figure S14). Models trained with average (100 µs exposure) and low light intensities (30 µs exposure) can perform well on images obtained with other exposure times except 300 µs for which accuracy decreases. Figure S15 summarizes the accuracies obtained. Similarly, models trained with average (100 µs exposure) and low light intensity (30 µs exposure) at fixed high gain (15 dB) can perform well on images obtained with other gains (5 and 10 dB). We hypothesized that the loss in accuracy at high exposure is due to reaching pixel saturation, leading to a disappearance of image features. We ascribe this robustness to the normalization of the images (subtraction of mean followed by division by standard deviation) before processing by the CNN.

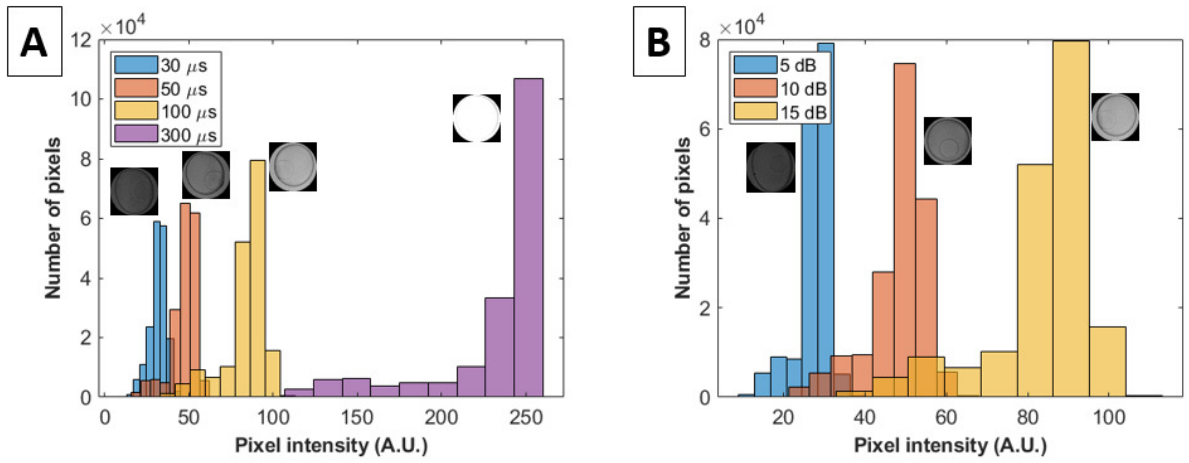

Figure S14. Acquisition of datasets using different exposure settings with representative images. (A) Exposure change from 30 to 300 μs at fixed 15 dB gain. (B) Gain change from 5 to 15 dB at fixed 100 μs exposure.

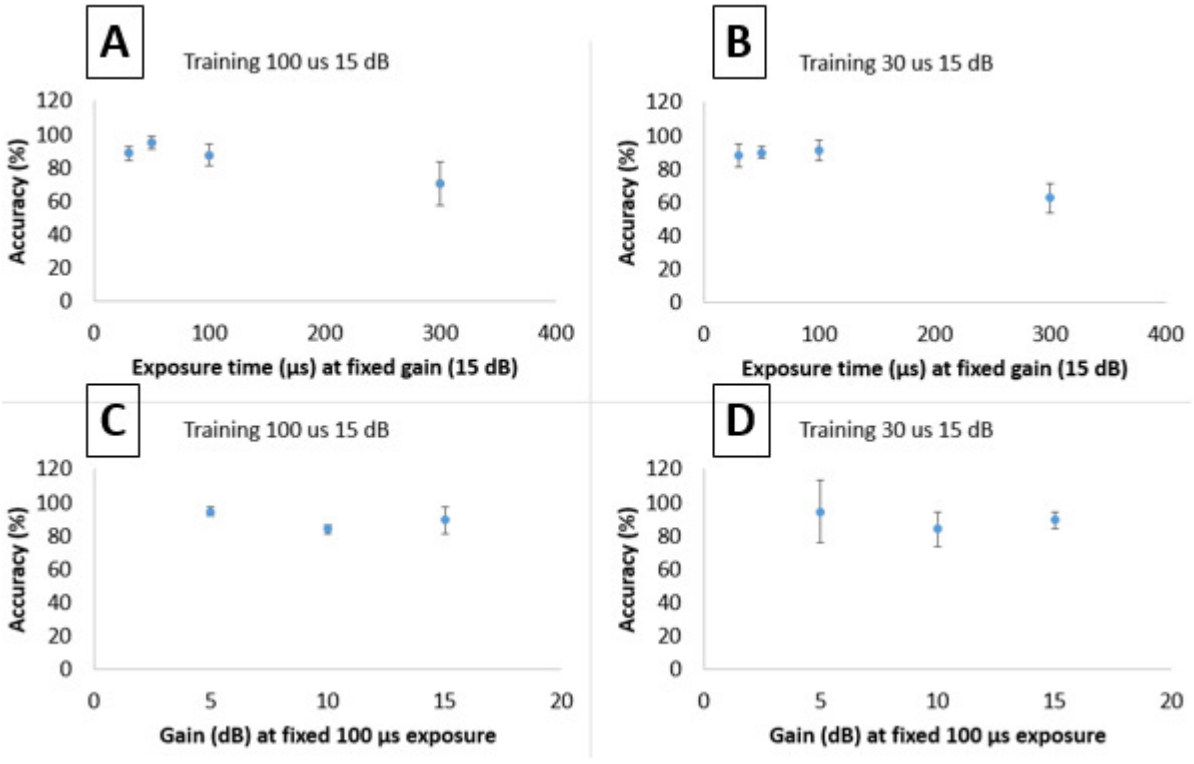

Figure S15. Accuracy of the CNN models when trained with fixed exposure settings. (A) and (B). Training with 100 μs and 30 μs, respectively, (gain fixed at 15 dB) and testing the models on images acquired with a range of exposure times (n= 3). (C) and (D) Training with 100 μs and 30 μs, respectively, (gain fixed at 15 dB) and testing the models on images acquired with a range of camera gains (n= 3).

10. Example of activation layers for spheroid sorting

Figure S16 gives examples of the filters at the ReLU activation layers of the networks validated on model image sets.

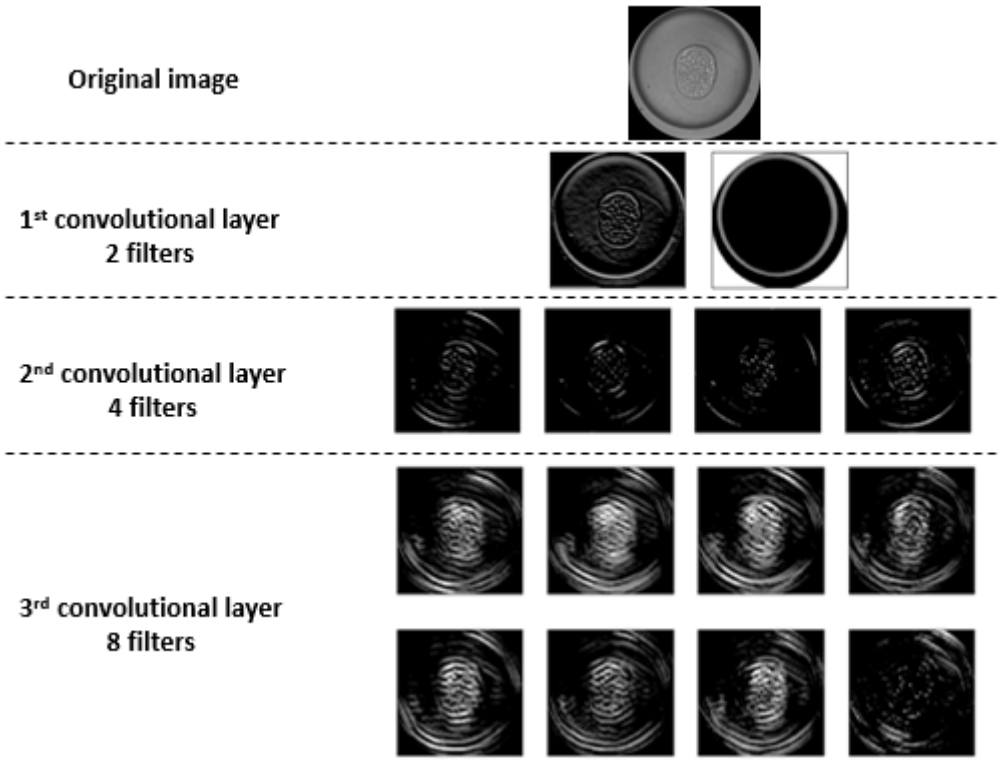

Figure S16. Filters from activation layers for multicellular spheroids with a CNN with 3 convolutional layers having 2, 4 and 8 filters, respectively.

11. Viability test of sorted spheroids

We have stained the sorted spheroids by adding 2 µM of Calcein Green AM to 1 mL of media, incubating 30 minutes in a cell incubator, then adding 20 µg/mL propidium iodide and 20 µM Hoechst 33342 for 10 minutes. The stained spheroids were centrifugated at 300 rpm for 3 minutes, resuspended in 1 mL PBS and imaged using a high-content screener (ImageXpress, Molecular Devices). The images are displayed in Figure S17, confirming viability of the cells forming the 3D cell cultures.

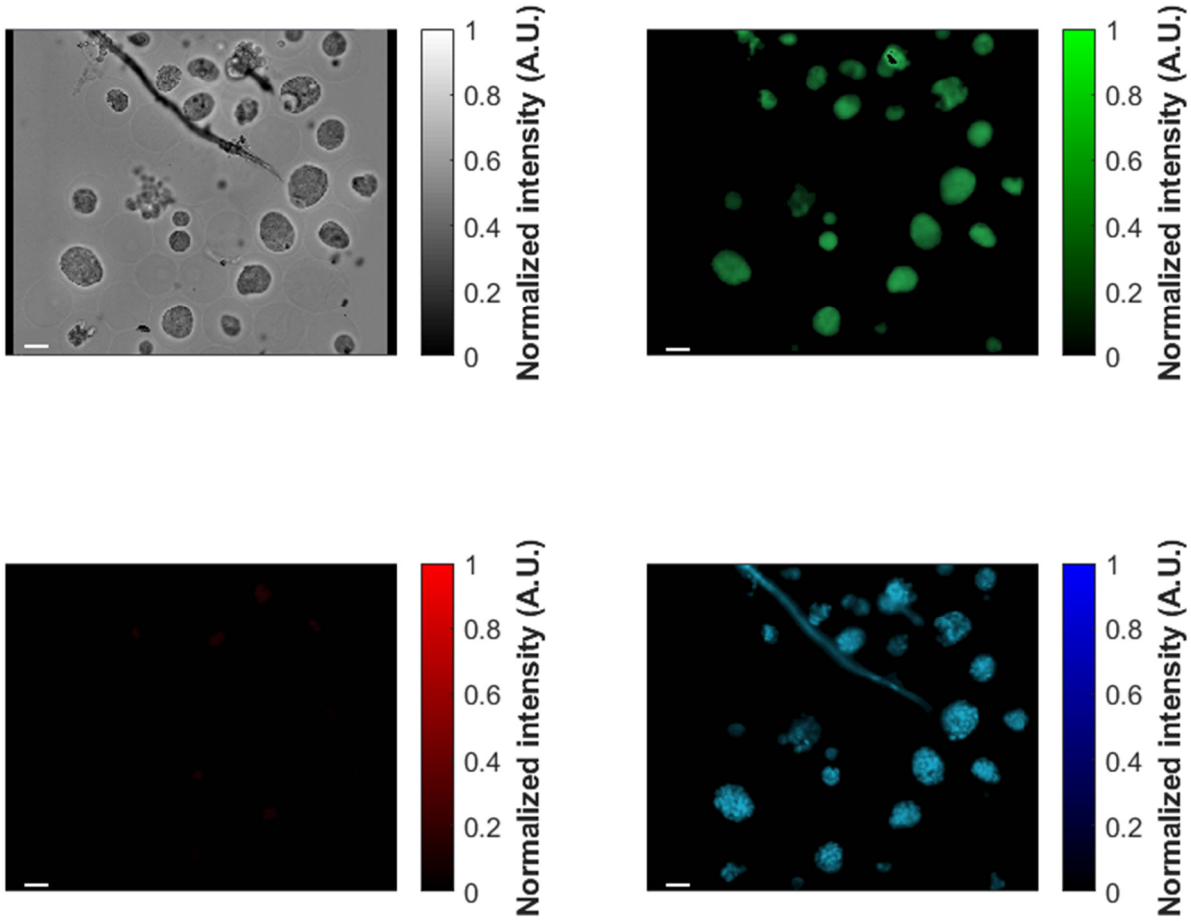

Figure S17. Images of stained spheroids obtained after microfluidic sorting followed by 24h incubation. (A) Bright-field. (B) Calcein green. (C) Propidium iodide. (D) Hoechst 33342. Scalebars = 50 µm.

References

(1) Zang, E., Brandes, S., Tovar, M., Martin, K., Mech, F., Horbert, P., Henkel, T., Figge, M. T., Roth, M., *Lab Chip.* 2013, *13*. 3707-3713.
(2) Girault, M., Kim, H., Arakawa, H., Matsuura, K., Odaka, M., Hattori, A., Terazono, H., Yasuda, K., *Scientific Reports.* 2017, *7*. .
(3) Nam, J., Lim, H., Kim, C., Kang, J. Y., Shin, S., *Biomicrofluidics.* 2012, *6*. .
(4) Chabert, M., Viovy, J. L., *Proceedings of the National Academy of Sciences of the United States of America.* 2008, *105*. 3191-3196.
(5) Joensson, H. N., Uhlen, M., Svahn, H. A., *Lab Chip.* 2011, *11*. 1305-1310.