

Learning to Count Cells: Applications to Lens-free Imaging of Large Fields

G. Flaccavento^{1,2}, V. Lempitsky³, I. Pope^{1,4}, P.R. Barber¹, A. Zisserman³, J.A. Noble², B. Vojnovic¹

¹Gray Institute for Radiation Oncology and Biology University of Oxford, Old Road Campus Research Building, Churchill Hospital, Oxford, OX3 7DQ, UK

²Biomedical Image Analysis (BioMedIA) Laboratory, University of Oxford, Old Road Campus Research Building, Churchill Hospital, Oxford, OX3 7DQ, UK

³Visual Geometry Group, University of Oxford, Parks Road, Oxford, OX1 3PJ, UK

⁴Cardiff University, School of Biosciences, Biomedical Sciences Building, Museum Ave., Cardiff, CF10 3AX, UK

Abstract—We have developed a learning algorithm that counts the number of cells in a large field of view image automatically, and can be used to investigate colony growth in time lapse sequences. The images are acquired using a novel, small, and cost effective diffraction device that can be placed in an incubator during acquisition. This device, termed a CyMap, contains a resonant cavity LED and CMOS camera with no additional optical components or lenses.

The counting method is based on structured output learning, and involves segmentation and computation using a random forest. We show that the algorithm can accurately count thousands of cells in a time suitable for immediate analysis of time lapse sequences. Performance is measured using ground truth annotation from registered images acquired under a different modality.

Index Terms—Cell counting, machine learning, optical diffraction, time lapse imaging

I. INTRODUCTION

The CyMap is a novel, cost effective imaging modality suitable for observing a large area of live cells in time lapse imaging [1]. CyMap imaging can be used in studies that analyse the growth of colonies after cellular insults (e.g. following ionizing radiation or cytotoxic agents) and their consequences on colony growth. Our goal therefore is to build a software tool that can count the number of cells in each of several colonies within a large area imaged by the CyMap device. As CyMap imaging is non-invasive, the cells remain viable and in optimal conditions within the incubator. Analysis may be then carried out for the entire time lapse acquisition, allowing the growth or otherwise of a given colony to be followed in time.

Advantages of imaging with a CyMap device include a field of view of 35 mm^2 as well as a low cost and simplicity that allow many such devices to be used in parallel. However,

Manuscript received June 27, 2011. This work was supported in part by the Engineering and Physical Sciences Research Council (Optical Biochips grant GR/S23483), Cancer Research UK (Programme grant C133/A/1812), and EU ERC grant VisRec no. 228180. V. Lempitsky is also supported by Microsoft Research projects in Russia. Corresponding authors: G. Flaccavento, email: giselle.flaccavento@rob.ox.ac.uk, V. Lempitsky, email: vlem@robots.ox.ac.uk, P. R. Barber, email: paul.barber@rob.ox.ac.uk, I. Pope, email: PopeI@cardiff.ac.uk, A. Zisserman, email: az@robots.ox.ac.uk, A. Noble, email: alison.noble@eng.ox.ac.uk, B. Vojnovic, email: boris.vojnovic@rob.ox.ac.uk

since the data collected are in the form of diffraction patterns, these need to be 'decoded' to derive information about individual cells. Such 'decoding' represents a very challenging, yet interesting image processing task, which can be attacked using supervised machine-learning techniques. Details of this process are presented in this paper. We discuss the task of training our algorithm through the collection of ground truth cell colony data. We show results of the trained algorithm, applied to new data. While the results are still preliminary, the obtained accuracies already show promise for using the CyMap as an alternative to conventional cell counting.

We propose our method as an alternative to two commonly used methods to analyse cell survival in colonies. The first method uses an automated scanning microscope, stain-free phase contrast imaging and an incubator housing to keep the live cells in ideal conditions. That method acquires large high resolution mosaic images showing intra-cellular changes. These systems are inevitably large and costly and rely on the accuracy of auto-focusing algorithms and the precision of motorized stages to maintain image quality [2], [3]. The second method, using clonogenic assays is much simpler: cells are grown in dishes and after an empirically determined time, the cell colonies are fixed and stained with a high contrast chromophore rendering the colonies visible to the eye. The resulting 'spots' are counted using a colony counter pen or microscope [4]. This second method does not allow time lapse data to be collected and the data are usually restricted to large cell colony numbers with no accurate information about the number of cells available.

II. MATERIALS AND METHODS

A. CyMap imaging

The CyMap acquires images of the cell colonies over time, as they divide and grow. The imager area is large enough to study 100 or more colonies of 50 or more cells per experiment. Unlike automated microscopes that make use of phase contrast microscopy with multiple field of views stitched together, the CyMap makes use of diffracted light to create an image. The images acquired by the CyMap are those of diffraction patterns of the cells rather than conventional trans-illuminated microscopy images. Each acquired image of 1280×1024 pixels

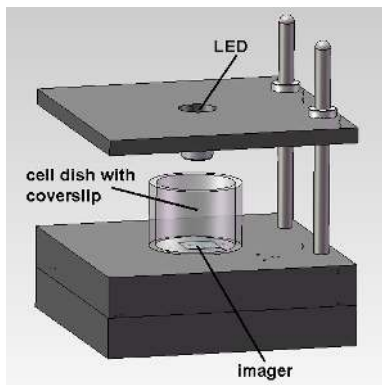


Fig. 1. Components of the CyMap. The imager is placed in contact with the bottom of the coverslip and illuminated from a point source LED. The LED-imager distance is typically 54 mm.

covers an area on the cell dish of 6.66×5.32 mm. When cells are sparsely seeded, their diffraction patterns occupy $\sim 10 \times 10$ pixels and individual cells are easy to identify. As the colonies grow and the diffraction patterns of neighbouring cells overlap, it becomes increasingly hard to see the boundaries between cells and to count the number of cells per colony.

The CyMap uses a resonant cavity LED, operating at ~ 2 mA and emitting at 650 nm and a CMOS imager. As there are no image forming optical components used, the system is afocal. An aperture close to the LED restricts the illumination cone so that it just covers the imager. The lower platform holds a cell dish, with the imager faceplate in contact with the bottom of the dish. As seen in Fig. 1, the geometry is such that a diffraction pattern due to the light passing through and around the cells is captured by the imager.

For the purpose of experiments, HeLa cells were grown in 30 mm diameter glass cell dishes with a coverslip ($170 \mu\text{m}$ thick) bottom, at a density of 2000 cells per dish. This cell density equates to approximately 100 cells per imager field. The HeLa cells were grown in Dulbecco's Modified Eagle's Medium media, supplemented with 100 units/mL of penicillin streptomycin and 10% fetal bovine serum (FBS). The cells were seeded and incubated at 37°C with 5% CO_2 to allow cell attachment and growth.

B. CyMap image analysis

Our goal is to extract useful statistical information about cell colonies from a CyMap image. In particular, in this paper, we focus on the extraction of cell counts from the raw CyMap images. This is achieved in several steps (Fig. 2). We first identify which pixels of a CyMap image correspond to live cells (*segmentation stage*). Based on the output of the segmentation stage, we then infer the density of the cells in different regions of the image (*counting stage*). Finally, we add simple morphological post-processing that separates individual colonies. The number of cells in each individual colony can then be estimated based on the inferred density map.

The segmentation and the density estimation (counting) stages are based on supervised machine-learning techniques, namely supervised classification and a supervised learning-to-count approach [5]. An accurate annotation is required for both

supervised training for these machine learning algorithms as well as quantitative evaluation of the resulting system. We used a microscope to image the cell colonies at a higher magnification, annotated the cell locations and then registered the image sets with the CyMap images. This additional microscopy allowed us to provide 'ground truth' data for the number of cells within each colony on the CyMap images. This process is detailed in the next subsection.

C. Image Acquisition for Training Data

To obtain the annotated data, cell dishes were prepared, and cells allowed to grow in the incubator for up to 6 days. At various time-points during those 6 days, the dishes were imaged with the CyMap. These images were acquired at 8-bit intensity resolution with an exposure time of 5 ms. A total of 9 CyMap images were acquired.

After each CyMap image was acquired, Hoechst 33342 (a fluorescent DNA intercalating dye) at 0.2 mmol/L was added to the cells and the cells incubated for an additional 15 minutes. Image sets were acquired using a 10×0.3 NA objective ($790 \times 602 \mu\text{m}$ fields, 1344×1024 pixels, 345 ± 25 nm excitation and >425 nm emission filters, with a 400 nm dichromatic reflector). To equal the field of view of each CyMap image, 90 epi-fluorescence and 90 phase contrast images were acquired.

The approximate centers of cell nuclei were then manually annotated with reference to the epi-fluorescence images, although the phase contrast images were also used to distinguish between binucleated and overlying cells. The annotations were then registered with the corresponding CyMap images on a per colony basis. The transform for each colony was determined using up to 10 corresponding points between the CyMap and the epi-fluorescence images. A least-squares solution solved the transform for translation, rotation, and scaling. An example of such annotations is shown in Fig. 3. Approximately 900 colonies were annotated from the set of 9 CyMap images. These annotations were used as training and testing data for the learning algorithm.

D. Live cell segmentation

We apply two standard texture banks (the Leung-Malik [6] and Schmid [7]) for the purpose of segmentation. The Leung-Malik filter bank contains rotated sub-sets of its own filters. We replace the filter responses of these groups with a pixelwise maximum and a minimum over the responses to the entire group (a very similar approach to [8]). As a result, the initial filtration stage becomes rotationally independent.

Given the filtered input, we treat the response at each pixel as a high-dimensional vector. During training, we use a random forest classifier [9] based on those features (we used 20 shallow trees having about 256 nodes each). Training is performed in two stages: the data are first subsampled uniformly and then sampled from the hard negatives resulting from the second stage. Each tree was trained on 50,000 positive and 100,000 negative examples.

The training labels are determined by the annotations (example shown in Fig. 3(d)). To obtain, the training data for

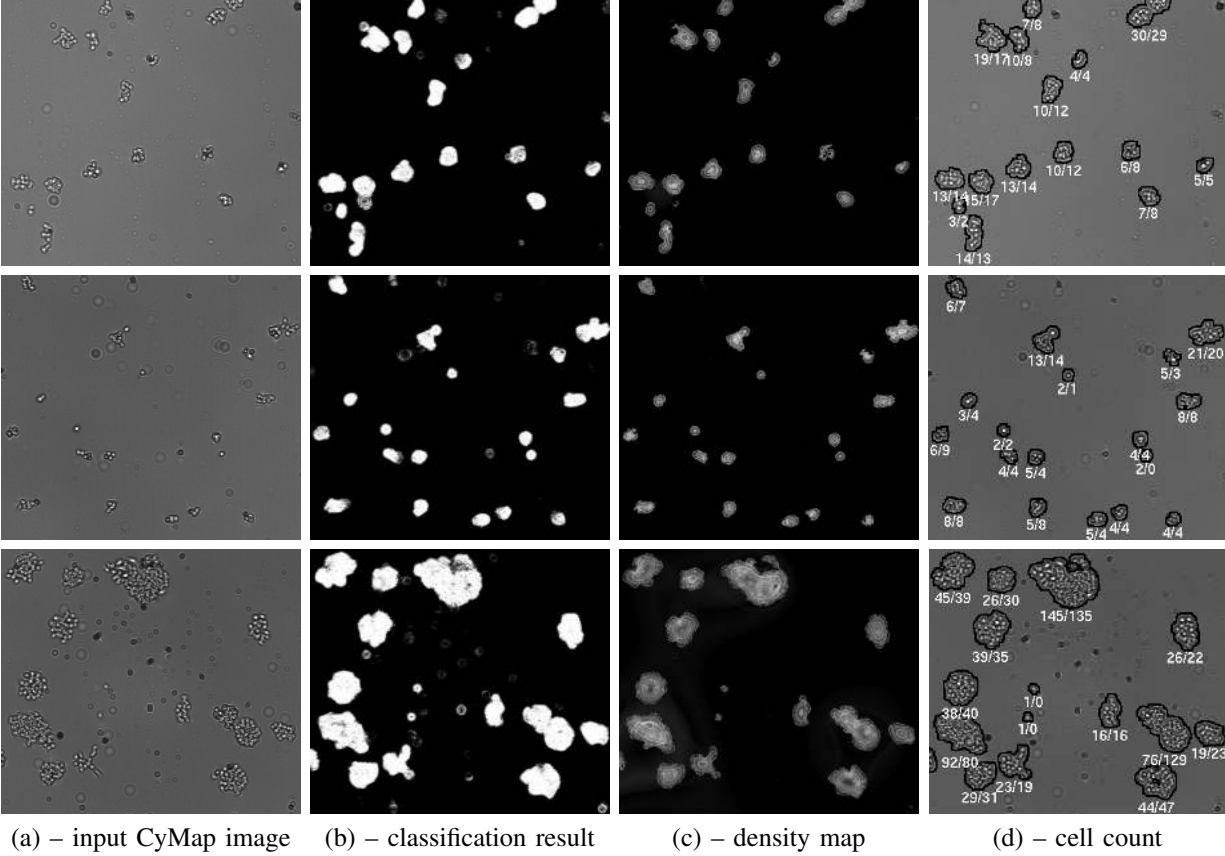


Fig. 2. The counting pipeline for fragments of different CyMap images: given the input CyMap (a), we first classify each pixel as occupied by live cells or not (b), and then infer the density map (c). Finally, we perform a morphological analysis and infer the colony boundaries ((d) black), for which the estimated counts (rounded to integers) are shown below the respective colonies (algorithm estimated / ground truth values).

segmentation, we applied a Gaussian filter ($\sigma = 4$ pixels) to the annotated data and threshold the result with respect to two thresholds (0.005 and 0.001). The result is used to determine pixels from (1) the positive class (highest value), (2) the unknown class (excluded from the training set) and (3) the negative class (lowest values).

The qualitative results for image segmentation resulting from applying the trained classifier to the unseen data are given in Fig. 2. Using the same definition of positive and negative classes as above, we found that the error of the classifier on the held-out data is 7.7%, while the recall and precision are 96% and 90% respectively. One method to improve the segmentation accuracy is the use of random field models, in particular based on graph cuts [10].

E. Cell counting

While segmentation of live cells already provides significant information about the cell population statistics, we aim to provide more detailed information about the imaged cells. Ideally, one would want to detect and segment individual cells. This may, however, be beyond the capability of the machine learning system or require excessive processing time. Instead, we settled on the slightly less ambitious goal of estimating the local density of cells throughout the images. This density map allows us to identify numbers of cells on a per colony basis. We base our density estimation on the learning-to-count framework recently proposed in [5].

The learning-to-count framework [5] is based on supervised learning and assumes that each pixel p in each training image i is encoded with a certain real-valued feature vector x_p^i from a high-dimensional feature space R^N . The framework then learns a linear mapping from the feature space R^N to the density values, so that the density of cells at each pixel in the image is determined by a dot product of the feature vector x_p^i with a learnt vector w (which defines such linear mapping):

$$F_i(p|w) = w^T x_p^i, \quad (1)$$

where $F_i(p|w)$ is the density value estimated for the pixel p in image i .

Given annotated training data, the method [5] defines a 'pseudo-ground truth' density image F_0^i as a simple sum of user-placed dots with a Gaussian kernel. It then estimates the w in a regularized empirical risk minimization framework:

$$w = \arg \min w \left(w^T w + \lambda \sum_{i=1}^N \mathcal{D}(F_0^i(\cdot), F_i(\cdot|w)) \right), \quad (2)$$

where the first term is the regularization term on w , while \mathcal{D} is a distance between the ground truth and the estimated density. The key contribution of [5] is the introduction of a specific distance function between the densities (*MESA-distance*) that is particularly suited for counting tasks. Such distance considers the set \mathbf{B} of all rectangular subwindows B . For each subwindow B , the absolute difference between the

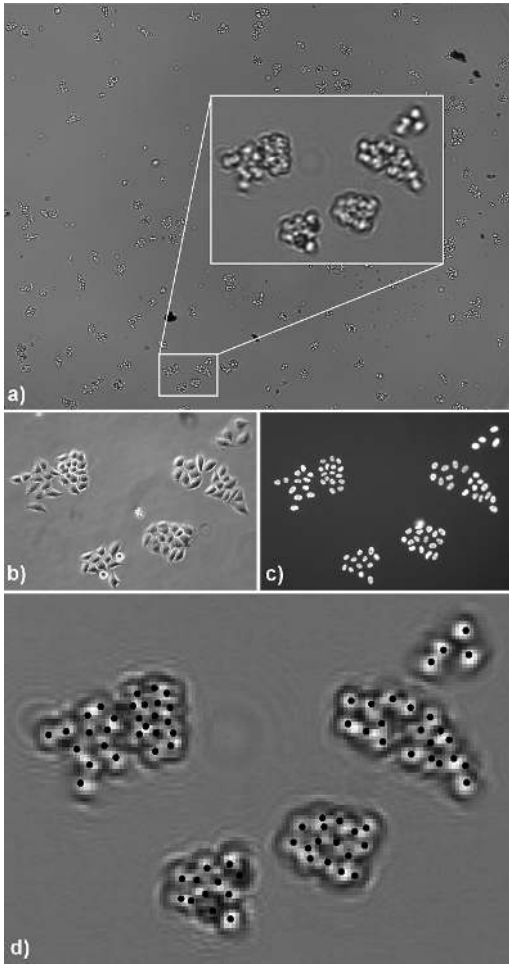


Fig. 3. The original CyMap image of the entire field and multiple colonies shown in (a), with an expanded portion of the image in the smaller box. Phase Contrast (b) and epi-fluorescence (c) images are used to create the annotated CyMap image shown in (d).

counts that the two densities predict for this subwindow B is considered. The distance is then defined as the maximum mismatch between the predicted counts over all subwindows:

$$\mathcal{D}_{MESA}(F_1, F_2) = \max_{B \in \mathcal{B}} \left| \sum_{p \in B} F_1(p) - \sum_{p \in B} F_2(p) \right|. \quad (3)$$

At train time, the approach in [5] finds the optimal w via convex minimization of the objective (2). At test time, it is sufficient to compute the features for each pixel in the processed image and then to map them to local densities via a dot-product with the learned w .

In our current implementation, the features are different from those used in [5]. In particular, let $A(p)$ be the output map of a random forest classifier for a CyMap image. We then take a sequence of blur kernels K_1, K_2, \dots, K_M and define $A_j(p)$ as a convolution of $A(p)$ with the blur kernel (we currently use the 8 Gaussian blur kernels with $\sigma = 0, 1, 2, 4, 6, 8, 32, 64$).

We then encode a pixel p with a sparse vector x_p of length $32 \times M$ that is a concatenation of M 32-dimensional chunks y_p^j encoding the value of p in each of the $A_j(p)$ maps. In particular, we quantize the range between 0 and 1 into 32

levels (bins), and make y_p^j zero everywhere except the two values corresponding to the boundaries of the bin that pixel p falls into. The values for the two non-zero entries sum to 1 and are set according to the distance from $A_j(p)$ to the bin boundaries, e.g. 0.5 and 0.5 if $A_j(p)$ is exactly in the middle of the bin, etc. In other words, y_p^j defines a sparse linear encoding for the value of $A_j(p)$.

The idea of this feature mapping is to allow the linear mapping to distinguish between pixels corresponding to live cells and those in the rest of the image. The live cell pixels can be distinguished based on the non-zero entries within sub-vector chunks corresponding to small blur kernels. A distinction between the pixels in the middle of very large colonies and those on the edge of the large colonies or within smaller colonies is also made. The latter distinction can be made based on the non-zero entries within sub-vector chunks corresponding to large blur kernels.

In general, we experimented with the framework of [5] and modified it slightly. In particular, we replaced the MESA-distance in (2) with its square (this modification has a relatively small impact on the accuracy) as the squared loss function is more suitable for the case when the ‘ground truth’ does not contain major errors. We also imposed the non-negativity constraint $w > 0$, which greatly decreased the convergence time of the cutting-plane based minimization algorithm, while also increasing the counting accuracy.

F. Colony separation and count estimation

Given the density estimate produced by the previous stage, we use simple morphological processing to identify individual colonies. In particular, we threshold the densities and then dilate the resulting mask with a 7×7 square kernel. The connected components of the resulting mask are then treated as individual colonies by summing the estimated density over connected regions. During the evaluation of the system, the predicted cell counts per colony can be compared with ground truth counts (defined as the number of annotation dots inside each connected region).

III. RESULTS AND DISCUSSION

We evaluated the counting accuracy of the proposed system. Our dataset contained 9 annotated images corresponding to different stages of progress of cell growth and containing between 674 and 3385 cells. The experiments were performed in a leave-one-out fashion, i.e. all training (both at the segmentation and at the counting steps) was done on 8 images and tested on the remaining one and this was repeated 9 times. The learning meta-parameter λ was set to $1e-3$, with the results being insensitive to its value. When training the counting stage, a small benefit was achieved by sampling a large number of the overlapping windows (e.g. of size 192×192) from each training image and using them for training, rather than to train the approach on the 8 full images. We report the results for this learning scenario.

Fig. 4 shows the counting accuracy evaluated on rectangular subwindows (of size 256×256 , sampled uniformly and with overlap), thus assessing the accuracy of the learning-to-count

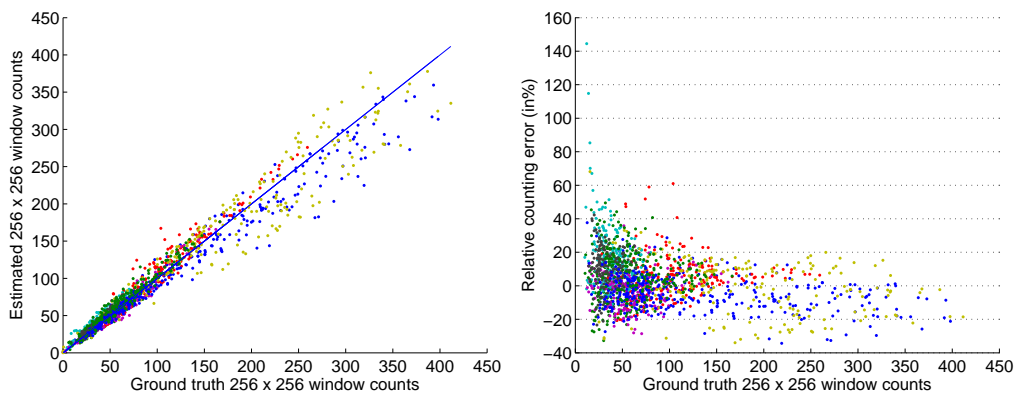


Fig. 4. Counting accuracy for rectangular subwindows. Left – predictions of our system plotted against the ‘ground truth’ counts (the line $x=y$ is superimposed). Right – the relative absolute error plotted against the ‘ground truth’ counts. The windows having less than 10 cells are excluded from the plot. Each color represents a test CyMap image (1 to 9).

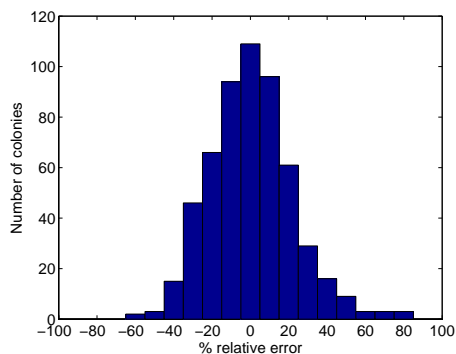


Fig. 5. The distribution of errors in the cell number per colony within the tested data.

framework on our system. It can be observed (Fig. 4–left) that a reasonably accurate match between the ‘ground truth’ counts and the estimated counts was obtained. The average absolute counting error over such subwindows is 9 cells (whereas the average number of cells in the subwindow is 87). As can be observed in (Fig. 4–right), in the majority of cases the relative error is within 30%.

Fig. 6 further assesses the accuracy of counting for individual colonies detected using simple morphological post-processing. Here, we also assessed how much the learning-to-count framework improved the accuracy on top of a simple baseline, predicting the count accuracy based on the colony area. There is an obvious monotonic dependency between the colony area and the number of cells per colony (Fig. 6a). The counts predicted by our system (Fig. 6b) are calibrated and the predicted counts for the colonies with a similar same true count have a smaller relative spread than their area. This is particularly noticeable for colonies containing less than 40 cells.

The improvement brought by the learning-to-count framework is further highlighted by Fig. 6d, when we consider the task of identifying large colonies. We consider three cases, where colonies of interest are defined as having more than 8, 16, or 32 cells respectively. As shown in Fig. 6d, the identification of these colonies is more precise using our

system than when using an area-based approach. Fig. 5 shows the distribution of errors in the cell number per colony within all the tested data. The error of the number of cells per colony for 75% of the colonies is between -30% and $+30\%$.

Conventional clonogenic assays determine the number of colonies, each containing cells above a threshold value, typically 50. The actual cell number is usually not known. With our method, we seek to provide additional data on much smaller colonies and thus gain additional insight into the early phase of colony growth. Fig. 7 shows a breakdown of colony number when these contain different cell numbers. The most relevant comparison data are shown in the right panel of Fig. 7. Although we are dealing with small colony numbers, it is probable that assay errors will be lower than conventional assays which involve counting >1000 colonies, though this remains to be validated and systematic errors (if any) removed. The colony number discrepancy is due to inaccuracies in algorithm-counted cell numbers. A complete validation requires additional testing with a known cell insult and of course larger numbers of images, best obtained by using larger numbers of CyMaps or larger area CyMaps.

IV. OUTLOOK

Other microscopy-based assays image cells directly using phase contrast microscopy: feature complexity is much higher and cell segmentation is much more complex. We expect errors with our method to decrease with larger colony numbers, as conventionally used, and by more appropriate choice of features. Although the method discussed here has concentrated on the use of static images for algorithm training, future work will extend the method by incorporating temporal information. We also intend to extend this work for use with other cell lines and experimental conditions.

ACKNOWLEDGEMENT

The authors acknowledge technical support from G. Pierce, R.G. Newman and J. Prentice, cell culture support from G. Bowey and J. Thompson at the Gray Institute for Radiation Oncology and Biology.

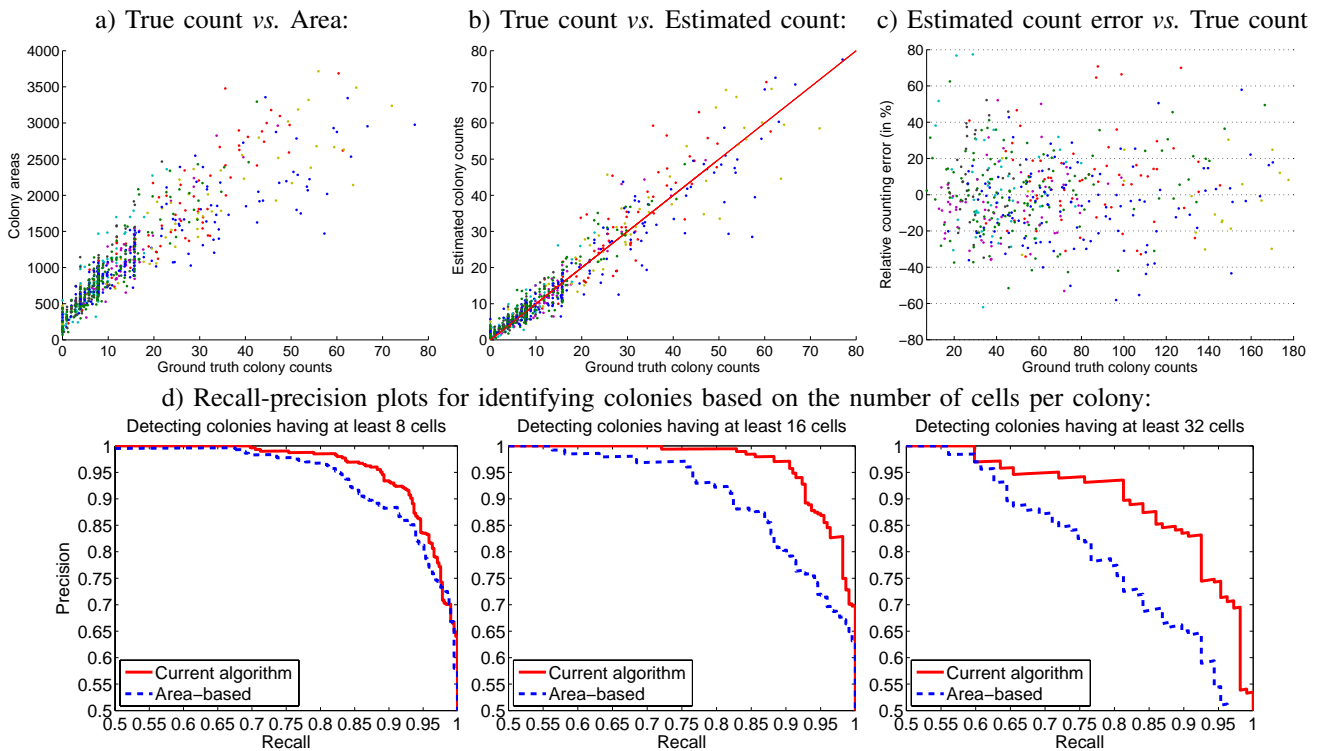


Fig. 6. Counting accuracy for colonies of cells – comparison with the area-based baseline. a) – areas of the colonies plotted against the true counts. b) – prediction of our system plotted against the true counts (notice a smaller vertical spread than in a). Colour of the dots encodes the number of the test CyMap image (1 to 9), from which a colony is taken. c) – relative error plotted against the true counts. d) – Recall-precision curves for the task of the identification of the colonies that have at least a certain number of cells. The curves are obtained by varying the threshold on the area (dashed, lower) or the threshold on the count predicted by our system (solid, higher).

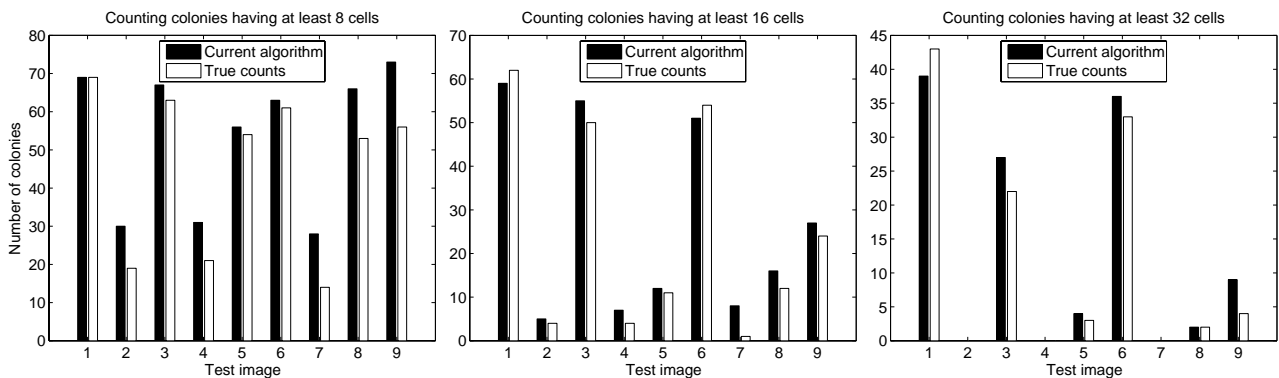


Fig. 7. Comparison of the number of colonies counted for a specified cell size using the current algorithm and the true count. These plots show the number of colonies for each of the 9 test images used in this work. Summing the total number of colonies over 8, 16, and 32 cells from all of the images, we find 483/410, 240/222 and 117/107 colonies are counted for each of the three cases. The algorithm, in general, determines a larger number of colonies because no temporal information is used here and because of overcounting of cells within colonies.

REFERENCES

- [1] B. Vojnovic, P. R. Barber, I. Pope, P. J. Smith, and R. J. Errington, “Detecting objects,” Great Britain Patent WO/2008/090 330, July 31, July 31 2008.
- [2] Essen BioScience Ltd., “Essen incucyte: product documentation,” 2011. [Online]. Available: <http://www.essenbioscience.com>
- [3] BioLogics Inc., “Biologics automated colony counters,” 2011. [Online]. Available: <http://www.biologics-inc.com>
- [4] N. A. Franken, H. M. Rodermond, J. Stap, J. Haveman, and C. van Bree, “Clonogenic assay of cells in vitro.” *Nature protocols*, vol. 1, no. 5, pp. 2315–2319, 2006.
- [5] V. Lempitsky and A. Zisserman, “Learning to count objects in images,” in *Advances in Neural Information Processing Systems (NIPS) 23*, 2010, pp. 1324–1332.
- [6] T. K. Leung and J. Malik, “Representing and recognizing the visual appearance of materials using three-dimensional textons,” *International Journal of Computer Vision*, vol. 43, no. 1, pp. 29–44, 2001.
- [7] C. Schmid, “Constructing models for content-based image retrieval,” in *CVPR (2)*, 2001, pp. 39–45.
- [8] M. Varma and A. Zisserman, “Classifying images of materials: Achieving viewpoint and illumination independence,” in *ECCV*, vol. 3, 2002, pp. 255–271.
- [9] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
- [10] D. Greig, B. Porteous, and A. Seheult, “Exact MAP estimation for binary images,” *Journal of the Royal Statistical Society, Series B*, vol. 51, no. 2, pp. 271–279, 1989.