Trainable segmentation of phase contrast microscopy images based on local Basic Image Features histograms

Nicolas Jaccard[1]
n.jaccard@ucl.ac.uk
Nicolas Szita[2]
n.szita@ucl.ac.uk
Lewis D. Griffin[1]
l.griffin@cs.ucl.ac.uk

[1] Department of Computer Science / CoMPLEX
University College London

[2] Department of Biochemical Engineering
University College London

Abstract

Phase contrast microscopy (PCM) is routinely used for the inspection of adherent cell cultures in all fields of biology and biomedicine. Key decisions for experimental protocols are often taken by an operator based on typically qualitative observations. However, automated processing and analysis of PCM images remain challenging due to the low contrast between foreground objects (cells) and background as well as various imaging artefacts. We propose a trainable pixel-wise segmentation approach whereby image structures and symmetries are encoded in the form of multi-scale Basic Image Features local histograms and classification of them is learned by random decision trees. This approach was validated for segmentation of cell versus background, and discrimination between two different cell types. Performance close to that of state-of-the-art specialised algorithms was achieved despite the general nature of the method. The low processing time (<4s per 1280×960 pixel images) is suitable for batch processing of experimental data as well as for interactive segmentation applications.

1 Introduction

Phase contrast microscopy (PCM) is widely used as the *de facto* light microscopy modality for the inspection of adherent cell cultures. Segmentation of PCM images is challenging due to the low contrast between the cell objects and the image background: generic thresholding approaches (e.g. Otsu's [9]) do not usually produce satisfactory results. Specialised approaches for PCM image segmentation that rely on *a priori* knowledge of the structure and properties of the images have been developed, including methods based on contrast filters [2, 6, 7, 14], active contours [1, 12], weak watershed assemblies [3], and image formation models [16]. More recently, trainable segmentation methods for microscopy images based on statistical learning of image features (e.g. intensity, texture) have been gaining traction [8, 13, 15]. Random forest classifiers were found to be suitable for segmentation tasks due to low computational complexity and their ability to accommodate large datasets such as images [11, 13]. Typically, trainable segmentation involves using the responses to a bank

of linear and non-linear filters computed at multiple scales as feature vectors for pixel-wise classification. However, the vector for a given pixel typically contains only a single value per scale for a given feature and thus does not encode for potentially valuable local information and context.

In this contribution, we describe a framework for PCM image segmentation whereby local histograms encoding image features at multiple scales were used as the input to random decision trees classifiers. This was achieved by computing Basic Image Features (BIFs), an image representation whose pixels take one of seven values depending on local features and symmetries [4]. This small range of possible pixel values allowed efficient construction of local histograms and classifier training was computationally tractable even in the case where multiple scales were considered. The segmentation performance is assessed using two separate PCM images dataset with distinct challenges. It is also compared to specialised PCM segmentation algorithms.

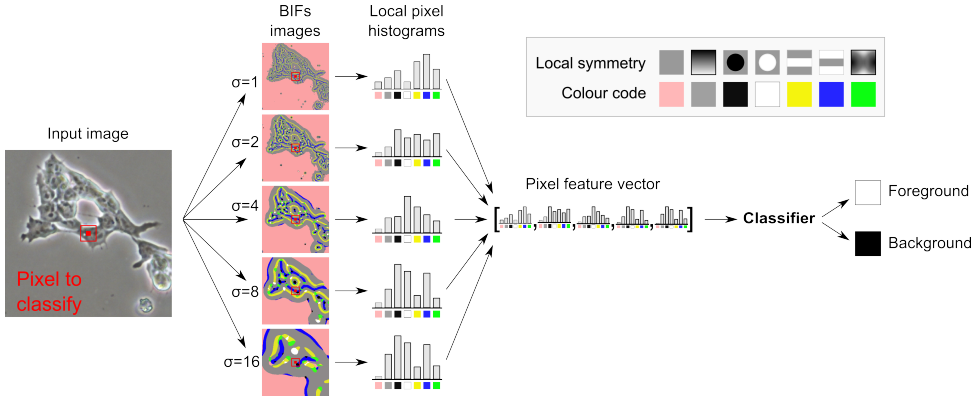

Figure 1: PCM pixel classification based on local histograms of BIFs.

2 Trainable segmentation

2.1 General approach

PCM images were segmented based on local histograms of Basic Image Features (BIFs) (Fig. 1). First, BIFs of the input image were computed at various scales. Local BIFs histograms were then computed for windows centred at each pixel of the image. The feature vector for classification was constructed by concatenation of the local BIFs histograms obtained for a given pixel of the input across all scales considered. The dimensions of the pixel feature vectors were thus $M \times 7$ where M is the number of scales considered. For comparison purposes, the situation where a single value per scale per pixel was considered, effectively corresponding to a window width of 0 pixel. Pixel feature vectors for classification were then of dimensions $M \times 1$.

The classifier used was a random forest with 20 trees and $\sqrt{F}$ features sampled at each split where F is the total number of features. The output of the classifier was a binary label, with 1 for foreground objects (i.e. cells) and 0 for image background. This output was used as is for segmentation without further processing or refinement.

2.2 Local Basic Image Features histograms computation

The computation of Basic Image Features (BIFs) consisted in classifying the output obtained from convolution of an image with a bank of derivative-of-Gaussian (DtG) filters into one of seven categories. These categories corresponded to distinct local image structures, as defined by local symmetries [4]. The response of the convolution of the image I with one of the DtG filter was denoted c_{ij} where i and j represented the order in the x and y directions, respectively. Scale normalised response s_{ij} was then computed as shown in equation 1.

$$s_{ij} = \sigma_B^{i+j} c_{ij} \tag{1}$$

Based on the scale normalised response, an intermediate calculation is carried out as shown in equations 2 and 3.

$$\lambda = s_{20} + s_{02} \tag{2}$$

$$\gamma = \sqrt{(s_{20} + s_{02})^2 + 4s_{11}^2} \tag{3}$$

Both λ and γ were computed for each pixel of the input image I. Pixels were then classified in one of seven categories based on the largest of $\{\varepsilon c_{00}, \sqrt{c_{10}^2 + c_{01}^2}, \lambda, -\lambda, \frac{\gamma+\lambda}{\sqrt{2}}, \frac{\gamma-\lambda}{\sqrt{2}}, \gamma\}$, resulting in a BIFs image I_B. BIFs computation was thus controlled by two parameters: the scale (standard deviation) σ_B of the DtG filters and a value ε that controls when a pixel should be considered flat (i.e. no specific structure). For this work, ε was kept at a constant value of 0.03, which was empirically found to produce good results regardless of the feature scale considered.

Soft-edged local BIFs histograms were computed by convolution [5]. First, seven binary masks $b^{(k)}$ were generated as shown in equation 4, one per Basic Image Feature.

$$b^{(k)}(x,y) = \begin{cases} 1 & \text{if } I_B(x,y) = k \\ 0 & \text{otherwise} \end{cases} \quad \text{for },k = 1,2,\ldots,7 \tag{4}$$

Images $C^{(k)}$ were obtained by convolution of each binary mask $b^{(k)}$ with a Gaussian kernel G_{σ_w} of standard deviation σ_w equal to the desired window size as shown in equation 5.

$$C^{(k)}(x,y) = G_{\sigma_w} * b^{(k)} \text{ for } k = 1,2,\ldots,7 \tag{5}$$

The histogram at location (x,y) was then constructed by concatenating the values of obtained across the seven $C^{(k)}$ images for that location, as shown in equation 6.

$$H(x,y) = [C^{(1)}(x,y), C^{(2)}(x,y), \ldots, C^{(7)}(x,y)] \tag{6}$$

2.3 Datasets and segmentation performance evaluation

Two datasets were used for segmentation performance evaluation (Fig.2). The first one was a set of 50 250×250 pixel mouse embryonic stem cells (mESC) PCM images [6]. This dataset was used to evaluate the performance of the algorithm for a simple foreground versus background segmentation task. The second dataset comprised 20 500×500 pixel PCM images of human embryonic stem cells (hESC) co-cultured with mouse embryonic fibroblasts (MEFs). This dataset was used to evaluate algorithm performance for the discrimination between two

foreground object types with similar visual features. This second dataset was used in a previous study [10] where a preliminary, unoptimised implementation of the approach described here-in resulted in promising results but at the cost of long processing times (∼40s per images). Due to the nature of the cells imaged, it was not possible to segment individual cells. Instead, the goal was the classification of pixels as either foreground or background.

Segmentation performance was evaluated by comparison of the algorithm output with ground truth images annotated by human experts. The agreement between the two was calculated using the F-score (i.e. Dice's coefficient). A leave-one-out cross-validation (LOOCV) approach was taken whereby the classifier was trained using 50000 pixels randomly sampled across $N-1$ images before being used to predict the labels for each pixel of the left out image. This was repeated N times so that all images were left out once. The reported LOOCV F-score was thus the mean F-score across the N images.

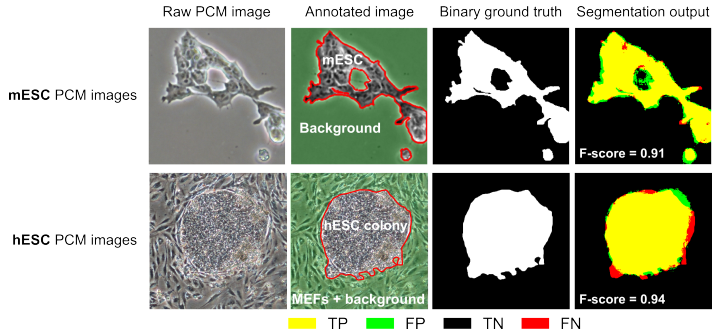

Figure 2: Datasets used for segmentation performance evaluation: mouse embryonic stem cells (mESC) and human embryonic stem cells (hESC) PCM images. The last column shows the agreement between the segmentation output using optimal parameters and the ground truth.

3 Segmentation performance

Segmentation performance was evaluated for both the mESC and hESC PCM images datasets over a range of parameter values. The width of the local histogram window (σ_w) was varied between 5 and 400 pixels. Five base BIFs scales (σ_B) were considered: 1, 2, 4, 8, and 16. All 31 permutations of these scales were evaluated for each window size σ_w. For both datasets, the best performance was obtained using local BIFs histograms computed at three scales: $\sigma_{B_1} = 1$, $\sigma_{B_2} = 2$, $\sigma_{B_3} = 8$ (Table 1). The optimal window width was found to be 15 and 100 pixels, for the mESC and hESC PCM datasets respectively. This difference was most likely due to the nature of the textures to discriminate between in both cases.

These optimal results were compared to those obtained by using different feature types and encoding methods (only the best results obtained for each condition after parameter values optimisation are reported). For both datasets, using single-scale BIFs as features resulted in a slight drop in performance. Likewise, using single BIFs values instead of local BIFs histograms resulted in marked decreases in mean LOOCV F-score of 15% and 51% for the mESC and hESC PCM datasets, respectively. Finally, alternative feature types were considered: raw pixel intensity values and pixel intensity values after application of a contrast (i.e. standard deviation) filter. Both feature types were encoded using 10 bin local histograms. The same scales permutations and window widths were investigated as in the case of BIFs.

Using intensity pixel values as features resulted in a pronounced decrease in segmentation performance for both datasets. In contrast, standard deviation features fared well, especially for the hESC datasets where it approached the performance of single-scale BIFs.

Our results were then compared with those obtained using previously described PCM image segmentation algorithms for the same mESC PCM images dataset (Table 1). Our trainable segmentation approach outperformed two of the three specialised algorithms it was compared to and produced results approaching those obtained using the third (best performing) one. It is important to note that those algorithms were specifically devised based on *a priori* knowledge of PCM images properties and specificities whereas trainable segmentation employed a generic framework solely based on BIFs and the user-set hard constraints. Interestingly, the classifier learned how to properly label halo artefacts around foreground objects as background pixels without being explicitly designed to do so (Fig.2).

Table 1: Optimal segmentation results compared with those obtained by alteration of various components of the algorithm: single BIFs scales in place of multiple scales, a single pixel value instead of local histograms, and intensity or contrast features instead of BIFs. Our results for the mESC dataset were also compared to specialised PCM segmentation algorithms. All results shown as mean F-score $\pm$ standard deviation after leave-one-out cross-validation (LOOCV).

	mESC dataset	hESC dataset
Optimal	0.92 ± 0.05	0.90 ± 0.07
Single-scale scheme	0.91 ± 0.05	0.88 ± 0.10
No histogram	0.78 ± 0.15	0.44 ± 0.22
Intensity features	0.83 ± 0.12	0.71 ± 0.24
Contrast features	0.85 ± 0.15	0.87 ± 0.10
Jaccard et al [6]	0.95 ± 0.04	-
Juneau et al [7]	0.85 ± 0.10	-
Topman et al [14]	0.84 ± 0.11	-

4 Summary and conclusion

In this work, we described a trainable segmentation algorithm for PCM images based on multi-scale local BIFs histograms. It performed well in foreground versus background segmentation tasks, approaching performance of state-of-the-art specialised algorithms. Indeed, the random forest classifier implicitly learned how to correct halo artefacts, which is usually done as an extra post-processing step in said algorithms [2, 6]. It also produced good results for a more complex segmentation task consisting in differentiating between two types of foreground objects with similar visual attributes. The fact that two significantly different problems could be suitably addressed using the same algorithm demonstrated the versatility of trainable segmentation approaches in general, and that of the proposed method in particular.

Processing a standard microscopy image (1280×960 pixels) took less than 4 seconds using a single thread on an 3.7 Ghz E5-1620 CPU, including the computations of BIFs at three scales and the construction of histograms for each pixel of the image. Using BIFs as features had the advantage of requiring only 7 bin histograms per scale, which allowed their rapid computation for each pixel. It also significantly reduced the computational complexity

of the offline phase (i.e. classifier training) as memory requirements and training time both increase with the number of features. In contrast, when using local 256-bin intensity feature histograms, computation time soared to more than 45 seconds for the same image and conditions. Specialised algorithms took on average about a second to process the same images [6].

These low processing times using BIFs make this method suitable for batch segmentation of large number of PCM images or that of time-lapse movie frames. To generate the results presented in this paper, the classifier was trained based on 50000 pixels sampled across the entire dataset (minus the left out image), or less than 1.6% and 1% of the total number of pixels for the mESC and hESC datasets, respectively. Combined with the low processing times, the ability to handle sparse annotations could enable the use of the proposed approach for interactive segmentation of PCM images.

5 Acknowledgment

The authors gratefully acknowledge the British Heart Foundation for funding Nicolas Jaccard's PhD studentship (Grant no. SP/08/004)

References

[1] M.E. Ambühl, C. Brepsant, J.-J. Meister, A.B. Verkhovsky, and I.F. Sbalzarini. High-resolution cell outline segmentation and tracking from phase-contrast microscopy images. *Journal of microscopy*, 245(2):161–70, 2012.

[2] C.J. Bradhurst, W Boles, and Y Xiao. Segmentation of bone marrow stromal cells in phase contrast microscopy images. In *23rd IVCNZ*, pages 1–6. IEEE, 2008.

[3] O. Debeir, I. Adanja, N. Warzee, P. Van Ham, and C. Decaestecker. Phase contrast image segmentation by weak watershed transform assembly. In *Biomedical Imaging: From Nano to Macro, 2008. ISBI 2008. 5th IEEE International Symposium on*, pages 724–727. IEEE, 2008.

[4] L.D. Griffin, M. Lillholm, M. Crosier, and J. van Sande. Basic image features (bifs) arising from approximate symmetry type. In *Scale Space and Variational Methods in Computer Vision*, pages 343–355. Springer, 2009.

[5] L.D. Griffin, P. Elangovan, A. Mundell, and D.C. Hezel. Improved segmentation of meteorite micro-ct images using local histograms. *Computers & Geosciences*, 39:129–134, 2012.

[6] N. Jaccard, L.D. Griffin, A. Keser, R.J. Macown, A. Super, F.S. Veraitch, and N. Szita. Automated method for the rapid and precise estimation of adherent cell culture characteristics from phase contrast microscopy images. *Biotechnology and bioengineering*, 111 (3):504–17, 2014.

[7] P.-M. Juneau, A. Garnier, and C. Duchesne. Selection and tuning of a fast and simple phase-contrast microscopy image segmentation algorithm for measuring myoblast growth kinetics in an automated manner. *Microscopy and microanalysis*, 19(4):855–66, 2013.

[8] T. Kazmar, M. Smid, M. Fuchs, B. Luber, and J. Mattes. Learning cellular texture features in microscopic cancer cell images for automated cell-detection. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2010:49–52, 2010.

[9] N OTSU. Threshold Selection Method From Gray-Level Histograms. *IEE Transactions on Systems Man and Cybernetics*, 9(1): 62–66, 1979.

[10] M. Reichen, R.J. Macown, N. Jaccard, A. Super, L. Ruban, L.D. Griffin, F.S. Veraitch, and N. Szita. Microfabricated Modular Scale-Down Device for Regenerative Medicine Process Development. *PLoS ONE*, 7(12):e52246, 2012.

[11] F. Schroff, A. Criminisi, and A. Zisserman. Object Class Segmentation using Random Forests. In *Procedings of the British Machine Vision Conference 2008*, pages 54.1–54.10, 2008.

[12] I. Seroussi, D. Veikherman, N. Ofer, S. Yehudai-Resheff, and K. Keren. Segmentation and tracking of live cells in phase-contrast images using directional gradient vector flow for snakes. *Journal of microscopy*, 247(2):137–46, 2012.

[13] C. Sommer, C. Straehle, U. Kothe, and F.A. Hamprecht. Ilastik: Interactive learning and segmentation toolkit. In *2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 230–233. IEEE, 2011.

[14] G. Topman, O. Sharabani-Yosef, and A. Gefen. A Method for Quick, Low-Cost Automated Confluency Measurements. *Microscopy and Microanalysis*, 17(06):915–922, 2011.

[15] Z. Yin, R. Bise, M. Chen, and T. Kanade. Cell segmentation in microscopy imagery using a bag of local Bayesian classifiers. In *2010 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 125–128. IEEE, 2010.

[16] Z. Yin, T. Kanade, and M. Chen. Understanding the phase contrast optics to restore artifact-free microscopy images for segmentation. *Medical image analysis*, 16(5):1047–62, 2012.