Identifying Subcellular Locations from Images of Unknown Resolution

Luís Pedro Coelho[1,2] and Robert F. Murphy[1,2,3]

[1] Lane Center for Computational Biology,
[2] Joint Carnegie Mellon University–University of Pittsburgh Ph.D. Program in Computational Biology,
[3] Departments of Biological Sciences, Biomedical Engineering, and Machine Learning, Carnegie Mellon University

Abstract. Our group has previously used machine learning techniques to develop computational systems to automatically analyse fluorescence microscope images and classify the location of the depicted protein. Based on this work, we developed a system, the Subcellular Location Image Finder (SLIF), which mines images from scientific journals for analysis.

For some of the images in journals, the system is able to automatically compute the pixel resolution (the physical space represented by each pixel), by identifying a scale bar and processing the caption text. However, scale bars are not always included. For those images, the pixel resolution is unknown. Blindly feeding these images into the classification pipeline results in unacceptably low accuracy.

We first describe methods that minimise the impact of this problem by training resolution-insensitive classifiers. We show that these techniques are of limited use as classifiers can only be made insensitive to resolutions which are similar to each other. We then approach the problem in a different way by trying to estimate the resolution automatically and processing the image based on this prediction. Testing on digitally down-sampled images shows that the combination of these two approaches gives classification results which are essentially as good as if the resolution had been known.

1 Introduction

Fluorescent microscopy is one of the methods of choice for determining the subcellular location of proteins. Methods for automatically analysing subcellular patterns in fluorescence microscope images have been extensively developed, allowing such determinations to be performed in a high-throughput, comprehensive manner (for reviews see [1,2]).

A very important property of a cell image is its pixel resolution (i.e., how big the space represented by a pixel is). This depends on the imaging approach used to collect the image (e.g., for widefield fluorescence microscopy using a digital camera, it depends on the magnification of the lens(es) used and the pixel spacing of the camera). Higher resolution images carry more information

M. Elloumi et al. (Eds.): BIRD 2008, CCIS 13, pp. 235–242, 2008.

and detail. However, many considerations may lead to acquisition of images with lower resolution.

As an outgrowth of our location proteomics work, we have developed a system, the Subcellular Location Image Finder (SLIF), that analyses images (and their associated captions) from scientific publications [3,4]. This system identifies figure panels likely to contain fluorescence microscope images and then attempts to analyse the subcellular patterns within. Of course, these images vary widely in magnification, and are often annotated with a scale bar from which the pixel resolution can be inferred. This approach has two problems. First, it relies on successful identification of both the scale bar in the image and the caption text that describes its size. We have obtained acceptable, but far from perfect, results for this task [3]. A more fundamental problem is that not all images are so annotated, in which case the pixel resolution is unknown.

This work is focused on classifying images whose resolution is unknown. We start by simply ignoring the problem and feeding images of unknown resolution into the classification pipeline. We next consider approaches to estimating resolution from the images.

2 Methods

For the work described here, we used a publicly-available collection of two-dimensional images of HeLa cells previously obtained by our group using widefield fluorescence microscopy [5]. It consists of immunofluorescence images of 9 proteins often used as markers for particular organelles or structures (one each for the endoplasmic reticulum, lysosomes, endosomes, mitochondria, the actin cytoskeleton, the tubulin cytoskeleton, and nucleoli, and two for the Golgi complex) as well as parallel images of a DNA-binding fluorescent probe to mark the nucleus. The pixel resolution of the images is $0.23\mu m$/pixel, and out-of-focus fluorescence in each image was estimated and removed using nearest neighbor deconvolution.

In order to investigate the effects of lowering the resolution, the images were digitally down-sampled. All data and software used in this paper are available from http://murphylab.web.cmu.edu/software.

2.1 Processing Images of Unknown Resolution

The first approach we used was to make the system insensitive to resolution, either by using features that are insensitive to image resolution or by training classifiers on examples from different resolutions so that they are able to classify any incoming image. Our group has previously pursued this line of reasoning with some success [6].

Some features can be designed in such a way as to make them roughly independent of resolution (e.g., SLF7.5, the ratio of the largest to smallest object in an image, which, discounting quantization effects, has the same value after image resampling). However, some informative features cannot be transformed

so that they become resolution independent. Haralick texture features [7], for example, are both very informative and resolution dependent.

Whether a classifier can be trained to handle multiple resolutions is an empirical question. We measured how gracefully a classifier degrades when tested outside its training resolution and found that its performance drops very fast as the difference between the training and the testing resolutions increases. For example, a classifier trained on images with a high resolution of $0.23\mu m$/pixel achieves only 45% accuracy when classifying images at resolution $1.15\mu m$/pixel. Comparatively, a classifier trained at that resolution can achieve 83%. Thus, ignoring the issue is not a viable procedure.

An alternative to building a resolution-independent classifier is to train it on multiple resolutions by including, for each image in the training set, several down-sampled copies of it. This approach showed better results (data not shown). For classifiers trained in a small set of nearby resolutions, no accuracy is lost when classifying images in any of those resolutions. In fact, there seems to be a small boost from training with multiple copies of the same image, as previously reported [3].

This approach, however, scales badly to a large set of resolutions. When training on resolutions which are very different, there is a performance cost. For example, a classifier trained on images at both 0.23 and $3.68\mu m$/pixel has only 71% accuracy on the low resolution images, while a classifier trained only on those images obtains 79%. The classifiers thus obtained also degrade poorly to resolutions which were not part of their training sets. Furthermore, the increase in size of the problem has huge computational costs (training a classifier goes from minutes to several hours).

2.2 Inferring Resolution

We propose a different approach for handling images of an unknown resolution: infer the resolution, based only on the image. We shall see that this complements the approach above.

If one was approaching this problem manually, without fast computers, one could start by counting how many pixels wide the nucleus of the cell appears to be. Given the knowledge that a real cell has a nucleus of around 20 μm, one can obtain an estimate of how large a pixel is. This idea underlies the approach we outline below.

For predicting resolution, we therefore define numerical features which attempt to capture the size of the nucleus. We start by thresholding the image by retaining only the pixels that are above average. To remove small objects, which are likely to be noise, we smooth the binarized image with a majority filter (implemented in Matlab by the *bwmorph* function). Finally, using the Matlab function *convhull*, we compute the convex hull of the resulting binary image. On the basis of this, we compute:

1. The number of pixels in the hull (the area).
2. The square root of the hull area.

3. Its perimeter (measured as the number of pixels on the edge).
4. Number of pixels across its semi-major and semi-minor axes. This was calculated as illustrated by Prokop and Reeves [8].

We also include the inverse of all of these features as the resolution scales linearly with the inverse of the size. These features can be calculated on either the protein channel or on the DNA channel, if available. We refer to this feature set as SLF28 if calculated on the protein image and SLF29 if calculated (separately) on the protein and DNA channels.

Algorithms. Starting with the description of the image given by the features, we attempt to predict the resolution. There are two possible ways to handle this as a machine learning problem: *classification* (by deciding on a few representative classes, for example) or *regression*. Our initial trials in using classification showed that, for a large number of classes, the learning algorithms took too long to converge. Thus, we focus on regression, namely linear regression. Regression parameters were learned by minimizing the squared error.

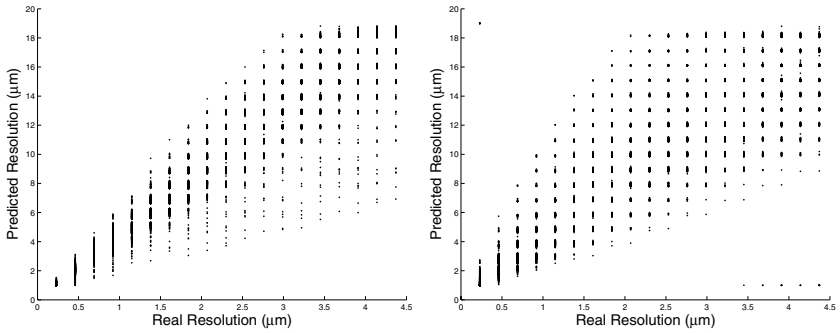
Initial tests revealed two properties:

- The range of of the training set is important. A set of parameters learned on the downsampling values $(1, 2, 3, 4, 5, 6)$ will do very well on test images of those values, but performance downgrades extremely fast outside of it (e.g., an image down-sampled by a factor of 10 will often be predicted to have been down-sampled by 15). On the other hand, inclusion of intermediate values is not as important as the range (i.e., training on $(1, 2, 3, 4, 5, 6)$ will do as well at handling images down-sampled by 3, a class in the training set, as training on $(1, 2, 4, 6)$ which does not include it).
- Breadth of training data (i.e., the difference between the largest and smallest resolution in the set) has a negative effect on accuracy.

This suggested a iterated regression scheme. We call Estimate$(i; \boldsymbol{r})$ the prediction for the resolution of image i given by the estimator trained on the set of resolutions $\boldsymbol{r}$. To process an incoming image, we first predict its accuracy on the whole range of values ($p_1 = \text{Estimate}(i; 1, 2, \ldots, N)$). A second prediction is defined by $p_2 = \text{Estimate}(i; p_1 - 2, p_1 - 1, p_1, p_1 + 1, p_1 + 2))$, where the first prediction is used to lookup the correct parameters for a refined prediction. Finally, we output the value $\hat{p} = \text{Estimate}(i; p_2 - 1, p_2, p_2 + 1)$.

2.3 Evaluation of Resolution Prediction Schemes

Figure 1(a) shows the results for predictions made using both the DNA and protein channels for each image. As we can see, the error is very small for high resolution images, but increases for low resolution ones. This is explained by quantization effects. Even if the nucleus always measured a perfect $20\mu m$, this translates to 87 pixels at $0.23\mu m$/pixel resolution, which can clearly be told apart from the 43 pixels it takes when the images are down-sampled by 2 to $0.46\mu m$/pixel resolution. However, at resolutions lower than $4\mu m$/pixel, a $20\mu m$



(a) With DNA channel (b) Without DNA channel

Fig. 1. Resolution Inference Results. Each dot is the result obtained for one test image, on the *x-axis* the original resolution is shown, on the *y-axis*, we show the output of the system. The results have not been rounded. A perfect result would be on the diagonal.

object takes only 4 pixels. If we consider that the nucleus size has some variation itself, it becomes clear that low resolutions cannot be told apart, even in principle.

When no DNA channel is available, one expects the variation in the size of the hull to be much larger (e.g., in the case of f-actin, the hull will probably contain the whole cell, while a DNA tag will only show the nucleus). To test this, we measured the coefficient of variation (the observed standard deviation divided by the standard mean, expressed as a percentage) of our measured features.

Table 1. Coefficient of Variation. The coefficient of variation of the features introduced in this work, when calculated on the DNA and protein channels.

	$\frac{\sigma}{\mu}$ DNA	$\frac{\sigma}{\mu}$ protein
Area	27%	126%
sqrt(Area)	13%	62%
Perimeter	14%	63%
Semi-Major Axis	16%	64%
Semi-Minor Axis	17%	66%

As Table 1 makes clear, the variation in features calculated on the protein channel is much greater than that calculated on the DNA channel. This explains why the results of inferring resolution based on the protein channel, presented on Figure 1(b), are not as good as those obtained using the DNA channel. We tested introducing SLF7DNA features into the regression model, followed by feature selection with stepwise discriminant analysis and regression on this set of variables. This brought about a small improvement in the results, but not enough to match the results with DNA features.

Looking at whether the iterated linear regression scheme makes a difference, one finds that the errors at the second level are lower than those at the first level, while the third level brings only a very minor improvement.

We tested the effects of removing half the resolutions used for training, while still testing on every resolution. This procedure simulates a situation where the image resolution was not in the training set. Results show that there is no accuracy penalty for this (data not shown).

2.4 Classification Pipeline

In the context of our work, the final goal of image processing is the classification output and the system must be evaluated on its accuracy there. First, we bring together the elements described above into an integrated classification pipeline.

For each image in our training set, we generated copies of it at lowered resolutions. We trained a classifier for each downsampling level, but included images from the level above and below it (to make it partially insensitive to resolution changes). To process an image, we estimate its resolution, and classify it using the classifier that was trained centred on the estimated resolution.

In order to evaluate our results, given that we expect classification accuracy to decrease with resolution due to lowered image quality, we compared our system

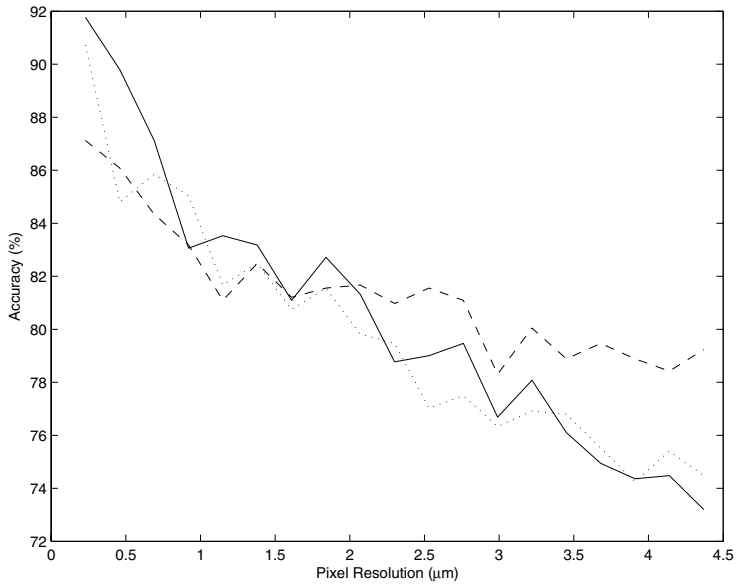


Fig. 2. Final Accuracy Results. The solid line shows the accuracy obtained in the case where the images were processed with resolution known. The dashed line shows the accuracy obtained when the resolution is inferred from the image, using only the protein channel, and this estimate is used for further processing. The dotted line shows the accuracy obtained when the resolution estimate is based on the DNA channel.

against a baseline where the image resolution is known. Figure 2 shows this comparison. We conclude that our system is capable of overcoming the unknown resolution problem.

3 Discussion

The SLIF system (which analyses images from published scientific journals) developed by our group needs to process images of differing and often unknown resolutions. Thus, we needed to adapt the image processing pipeline to handle such images.

We tackled the problem of handling images of unknown resolution by predicting it. Our solution was based on the calculation of simple features, which tried to capture the size of the nucleus, when a DNA channel was present, or the size of the cell, when it was not. These features were used for iterated linear regression. The resulting estimate predicted the resolution very well, if the DNA channel was available as the nucleus provides a known reference point in each cell. On the basis of only the protein channel, the prediction error increases by around two-fold.

For integration into the SLIF system, where images often contain multiple cells, the images will have to be segmented as a preprocessing step. In the case where a DNA channel is available (usually represented by one of the image's color channels), segmentation is easier as nuclei tend to be separable. Since the whole image is at the same resolution, results from different cells can be averaged together to obtain the final prediction.

Acknowledgments

This work was supported in part by NIH grant R01 GM078622. Facilities and infrastructure support were provided by NSF grant EF-0331657, NIH National Technology Center for Networks and Pathways grant U54 RR022241, and by NIH National Center for Biomedical Computing grant National U54 DA021519.

Luís Pedro Coelho was partially supported by a fellowship from the Fulbright Program and by the Portuguese Science and Technology Foundation (fellowship SFRH/BD/37535/2007).

References

1. Chen, X., Velliste, M., Murphy, R.: Automated interpretation of subcellular patterns in fluorescence microscope images for location proteomics. Cytometry 69 A, 631–640 (2006)
2. Glory, E., Murphy, R.: Automated Subcellular Location Determination and High-Throughput Microscopy. Developmental Cell 12(1), 7–16 (2007)
3. Murphy, R.F., Velliste, M., Yao, J., Porreca, G.: Searching online journals for fluorescence microscope images depicting protein subcellular location patterns. In: BIBE 2001: Proceedings of the 2nd IEEE International Symposium on Bioinformatics and Bioengineering, Washington, DC, USA, pp. 119–128. IEEE Computer Society, Los Alamitos (2001)

4. Murphy, R.F., Kou, Z., Hua, J., Joffe, M., Cohen, W.W.: Extracting and structuring subcellular location information from on-line journal articles: The subcellular location image finder. In: IASTED International Conference on Knowledge Sharing and Collaborative Engineering, pp. 109–114 (2004)
5. Boland, M.V., Murphy, R.F.: A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of HeLa cells. Bioinformatics 17(12), 1213–1223 (2001)
6. Chen, X., Murphy, R.: Interpretation of Protein Subcellular Location Patterns in 3D Images Across Cell Types and Resolutions. In: Lecture Notes in Computer Science, pp. 328–342. Springer, Heidelberg (2007)
7. Haralick, R.M.: Statistical and structural approaches to texture. Proceedings of the IEEE 67, 786–804 (1979)
8. Prokop, R.J., Reeves, A.P.: A survey of moment-based techniques for unoccluded object representation and recognition. CVGIP: Graph. Models Image Process. 54(5), 438–460 (1992)