Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *http://www.jstatsoft.org/*

An Interactive Java Statistical Image Segmentation System: GEMIDENT

Susan Holmes
Stanford University *

Adam Kapelner
Stanford Medical School

Peter P. Lee
Stanford Medical School †

Abstract

Supervised learning can be used to segment / identify regions of interest in images using both color and morphological information. A novel object identification algorithm was developed in Java to locate immune and cancer cells in images of immunohistochemically-stained lymph node tissue from a recent study published by Kohrt, Nouri, Nowels, Johnson, Holmes, and Lee (2005). The algorithms are also showing promise in other domains. The success of the method depends heavily on the use of color, the relative homogeneity of object appearance and on interactivity. As is often the case in segmentation, an algorithm specifically tailored to the application works better than using broader methods that work passably well on any problem. Our main innovation is the interactive feature extraction from color images. We also enable the user to improve the classification with an interactive visualization system. This is then coupled with the statistical learning algorithms and intensive feedback from the user over many classification-correction iterations, resulting in a highly accurate and user-friendly solution. The system ultimately provides the locations of every cell recognized in the entire tissue in a text file tailored to be easily imported into R (Ihaka and Gentleman 1996) for further statistical analyses. This data is invaluable in the study of spatial and multidimensional relationships between cell populations and tumor structure. This system is available at `www.GemIdent.com` together with three demonstration videos and a manual.

Keywords: interactive boosting, cell recognition, image segmentation, Java.

1. Introduction

We start with an overview of current practices in image recognition and a short presentation of the clinical context that motivated this research, we then describe the software and the complete workflow involved, finally the last two sections present technical details and potential improvements. The interactive algorithm, although developed to solve a specific problem in histology, works on a

*Funded by NSF-DMS award 02-41246
†Funded by a DOD Era Hope Scholar grant

wide variety of images. For instance, locating of oranges in a photograph of an orange grove (see Fig.1).

Figure 1: The original image (left), a mask superimposed on the original image showing the results of pixel classification (center), the original image marked with the centers of the oranges (right)

Any image that has few relevant colors, such as green and orange in the above example, where the objects of interest vary little in shape, size, and color, can be analyzed using our algorithm. First, we will describe the application to cell recognition in microscopic images.

1.1. Previous research

As emphasized in recent reviews(Ortiz de Solirzano, Garcea Rodriguez, Jones, Pinkel, Gray, Sudar, and Lockett (1999), Mahalanobis, Kumar, and Sims (1996), Wu, Gauthier, and Levine (1995), Wu, Barba, and Gil (1998), Yang and Parvin (2003), Gil, Wu, and Wang (2002), Fang, Hsu, and Lee (2003), Kovalev, Harder, Neumann, Held, Liebel, Erfle, Ellenberg, Neumann, Eils, and Rohr (2006a)), automated computer vision techniques applied to microscopy images are transforming the field of pathology. Not only can computerized vision techniques automate cell type recognition but they enable a more objective approach to cell classification providing at the same time a hierarchy of quantitative features measured on the images. Recent work on character recognition (Chakraborty 2003) shows how efficient interactivity can be in image recognition problems, with the user pointing out mistakes in real time, thus providing online improvement. In modern jargon, we call this interactive boosting (Freund and Schapire 1997). Current cell image analysis systems such as `EBimage` (Skylar and Huber 2006) and `Midas` (Levenson 2006) Collins (2007) do not provide these interactive visualization and correction features.

1.2. Specific context: Breast Cancer Prognosis

Kohrt *et al.* (2005) showed that breast cancer prognosis could be greatly improved by using immune population information from immunohistochemically-stained lymph nodes. To take this analysis a step further, we would like to detect and pinpoint the location of each and every cancer and immune cell in the high-resolution full-mount images of lymph nodes acquired via automated microscopy. This task is harder than classification of an entire slide as normal or abnormal as done in Maggioni, Warner, Davis, Coifman, Geshwind, Coppi, and DeVerse (2004) for instance.

A typical tissue contains a variety of regions characteristic of cancer, immune cells, or both. It would not be possible for a histopathologist to identify and count all the cells of each type on such a slide. Even if a whole team of cell counters were available, the problems of subjectivity and bias on such a scale would discredit the results. It is very useful to have an automated system to identify

and count cells objectively.

Kohrt *et al.* (2005) also showed that T-cell and dendritic cell populations within axillary lymph nodes of patients with breast cancer are significantly decreased in patients who relapsed. No study thus far has examined the spatial variability in lymphocyte populations and phenotypes as related to lymph node-infiltrating tumor cells and clinical outcome. The location of tumor-dependent immune modulation has significant sequelæ, given the critical role of lymph nodes in activation of the immune response. This suggests that tissue architecture could yield clues to the workings of the immune system in breast cancer. This can be investigated through spatial analysis of different cell populations. The limiting step to date has been locating every cell.

1.3. GEMIDENT

This algorithm has been engineered into the software package named GEMIDENT (Kapelner, Holmes, and Lee 2007). The distribution, implemented in Java, includes an easy-to-use GUI with four panels - color (or stain) training, phenotype training / retraining (see Fig. 6), classification, and data analysis (see Fig. 7) with a final data output into a text file which is easy to input and analyse in R. The Java implementation ensures that full platform-independence is supported. The distribution also includes support for image sets derived from the BacusLabs Bliss Imager(Bacus 2007a), the BacusLabs NanoZoomer(Bacus 2007b), and the CRI Nuance Multispectral Imager(Nuance 2007).

In this paper, we focus on the software itself, its use in conjunction with R and the developments which were engineered to analyze multispectral images where the chromatic markers (called chromagens) are separated a priori .
Figure 2 shows GEMIDENT employed in the localization of cancer nuclei in a typical Kohrt(Kohrt *et al.* 2005) image. The algorithm internals are detailed in section 2 but we give a brief summary

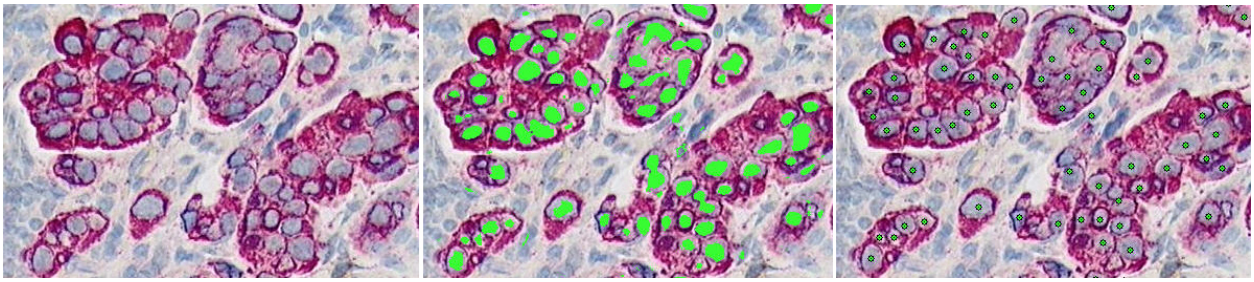

Figure 2: The original image (left), a mask superimposed on the original image showing the results of pixel classification (center), the original image marked with the centers of the nuclei (right)

here for potential users who do not need to know the technical details.

Our procedure requires interactive training: the user defines the target objects sought from the images. In the breast cancer study, this would be the cancer nuclei themselves. The user must also provide examples of the "opposite" of the cancer nuclei, i.e. the "non-object." Fig. 3 shows an excerpt from the training step.

New images can be classified into cancer nuclei and non-cancer nuclei regions using a statistical learning classifierHastie, Tibshirani, and Friedman (2001). Using simple geometric techniques, the centers of the cancer nuclei can then be located and an accurate count tabulated.

The user can then return and examine the mistakes, correct them, and retrain the classifier. In this way a more accurate classifier is iteratively constructed. The process is repeated until the user is

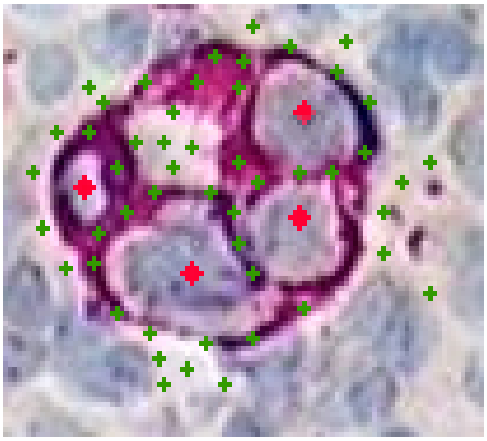

Figure 3: An example of training in a microscopic image of immunohistochemically stained lymph node. The cancer membranes are stained for Fast Red (displays maroon). There is a background nuclei counterstain that appears blue. The phenotype of interest are cancer cell nuclei. Training points for the nuclei appear as red diamonds and training points for the "non-nuclei" appear as green crosses.

satisfied with the results.

2. Algorithm

Image Acquisition

Any multiple wavelength set of images can be used, as long as they are completely aligned. In this example, the images were acquired via a modified CRI Nuance Multispectral ImagerNuance (2007). The acquisition was completely automated via an electronically controlled microscopy with an image decomposition into subimages. Multiple slides were sequentially scanned via an electronically controlled cartridge that feeds and retracts slides to and from the stage. The setup enables scanning of multiple slides with large histological samples (some being $10mm+$ in diameter and spanning as many as $5,000+$ subimages or stages) at 200X magnification.

Each raw exposure is a collection of eight monochromatic images corresponding to the eight unique wavelength filters scanned (for information on dimension reduction in spectral imaging see(Lee, Woodyatt, and Berman 1990)). The choice of eight is to ensure a complete span of the range of the visual spectrum but small enough to ensure a reasonable speed of image acquisition and avoid unwieldly amounts of data. These multispectral images are composed of 15-bit pixels whose values correspond to the optical density at that location. The files for each wavelength for each subimage or stage are typically TIF files of about 3MB each.

Prior to whole-tissue scanning, a small representative image is taken that contains lucid examples of each of the S chromagens. The user selects multiple examples of each of the S chromagens. This information is used to compute a "spectral library." The same spectral library can then be used for every histological sample that is stained with the same chromagens.

After whole-tissue scanning, the intermediate images are combined using proprietary spectral unmixing algorithms(Nuance 2007) to yield S orthogonal chromagen channel images. These "spectrally unmixed" images are also monochromatic and composed of 15-bit pixels whose values correspond

to a pseudo-optical density.

We use the spectrally unmixed images *directly* as score matrices, which we call F_s where s is the chromagen of interest. The program has been used with various scoring generation mechanisms and the quality of the statistical learning output has proved quite robust to these changes. Multispectral microscopy has the advantage of providing a clean separation of the chromagen signals.

Training Phase

1 **Interactive acquisition of the Training Sets for Objects of Interest**

For each of the P phenotypes or categories of objects of interest, a training set is built: the user interactively chooses example points, $\mathbf{t} = (i, j)$, in the images where the phenotype *is* located forming the lists: $T_1^+, T_2^+, \ldots, T_P^+$. In addition, the user interactively chooses example points where *none* of the P phenotypes are located, i.e. the "non-object," forming the list T^-.

Learning Phase

2 **Feature Definition:**

a) Define a "ring", $\mathbf{c}_r$, to be the 0/1 mask of the locus of pixels, $\mathbf{t}$, $r + \epsilon$ distance away from the center point $\mathbf{t}_o$ where ϵ refers to the error inherent in discretizing a circle (see Fig. 4). Generate a set of rings, for $r \in \{0, 1, 2, \ldots, R\}$ where R is the maximum radius of the largest phenotype to be identified and it can be modified by a multiplicative factor which controls the amount of additonal information beyond R to be included: $C \equiv \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_R\}$

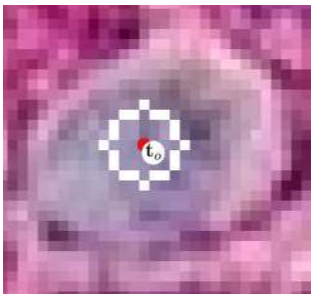

Figure 4: Example of a ring mask: $\mathbf{c}_4$ superimposed onto a typical cancer nuclei from the Kohrt images, white pixels designate the points which participate in the ring's score.

b) For a point $\mathbf{t}_o = (i_o, j_o)$ in the training set and for a ring $\mathbf{c}_r$, create a "Ring Score," $\ell_{q,r}$, by summing the scores for all points in the mask:

$$\ell_{s,r}(\mathbf{t}_o) = \sum_{\mathbf{t} \in \mathbf{c}_r} F_s(\mathbf{t}_o + \mathbf{t})$$

c) Repeat for all rings in C and for all S score matrices to generate the observation record, $\mathbf{L}_i$ of length $S \times R$ by vectorizing $\ell_{s,r}$:

$$\mathbf{L}(\mathbf{t}_o) = \left(\ell_{1,1}, \ell_{1,2}, \ldots, \ell_{1,R}, \ell_{2,1}, \ell_{2,2}, \ldots, \ell_{2,R}, \ldots\ldots, \ell_{S,1}, \ell_{S,2}, \ldots, \ell_{S,R}\right)$$

d) Now compute an observation record for each point $\mathbf{t}$ in the training sets for all phenotypes and for the 'non' category. We append a categorical variable recording the phenotype to all observation records $\mathbf{L}_i$.

3 Creation of a Classifier

All observation vectors are concatenated row-wise into a training data matrix, and a supervised learning classifier that will be used to classify all phenotypes is created.

In this implementation we use the statistical learning technique known as "Random Forests" developed by Breiman (2001) to automatically rank features among a large collection of candidates. This technique has been compared to a suite of other learning techniques in a cell recognition problem in Kovalev, Harder, Neumann, Held, Liebel, Erfle, Ellenberg, Neumann, Eils, and Rohr (2006b) who found it to be the best technique providing both the most accurate and the least variable of all the techniques compared. As all supervised learning techniques (Hastie *et al.* 2001), the method depends on the availability of a good training set of images where the pixels have already been classified into several groups.

At this point, all previous data generated can be discarded. Most machine learning classifiers, including our version of Random Forests, provide information on which scores $\ell_{q,r}$ are important in the classification (see an example in Figure 12).

Classification Phase

4 Pixel Classification

For each image I to be classified, an observation record is created for each pixel (steps 3 & 4), $\mathbf{L}(\mathbf{t}_o)$, and then evaluated by the classifier. The result is then stored in a binary matrix, with 1 coding for the phenotype and 0 for the opposite. There are P binary matrices, $B_1, B_2, \ldots, B_P$, (one for each phenotype):

$$I \xrightarrow{\textit{supervised learning classifier}} B_1, B_2, \ldots, B_P$$

To enhance speed, k pixels can be skipped at a time and the B_p's can be closed k times (dilated then eroded using a 4N mask(Gonzalez, Woods, and Eddins 2004)) to fill in holes.

5 Post-processing

Each pixel is now classified. Contiguously classified regions, called "blobs" hereon are post-processed in order to locate centers of the relevant objects, such as the middle of an orange or the cell nucleus.

Define the matrices C to hold centroid information:

$$B_1, B_2, \ldots, B_P \xrightarrow{\textit{blob analysis}} C_1, C_2, \ldots, C_P$$

There are many such algorithms for blob analysis. We used a simple one which we summarize below. For a more detailed description, see Appendix A. For an example of the results it produces, see Fig. 5

A sample distribution of the training blob sizes is created by reconciling the user's training points then counting the number of pixels in each blob using the floodfill algorithm. We calculated the 2nd percentile, 95th percentile, and the mean. For each blob obtained from the classification, we used those sample statistics to formulate a heuristic rule that discards those that are too small and quarantines those that are too large. Those that are too large are split up based on the mean blob size. To locate each blob's centroid, the blob's coordinates were averaged.

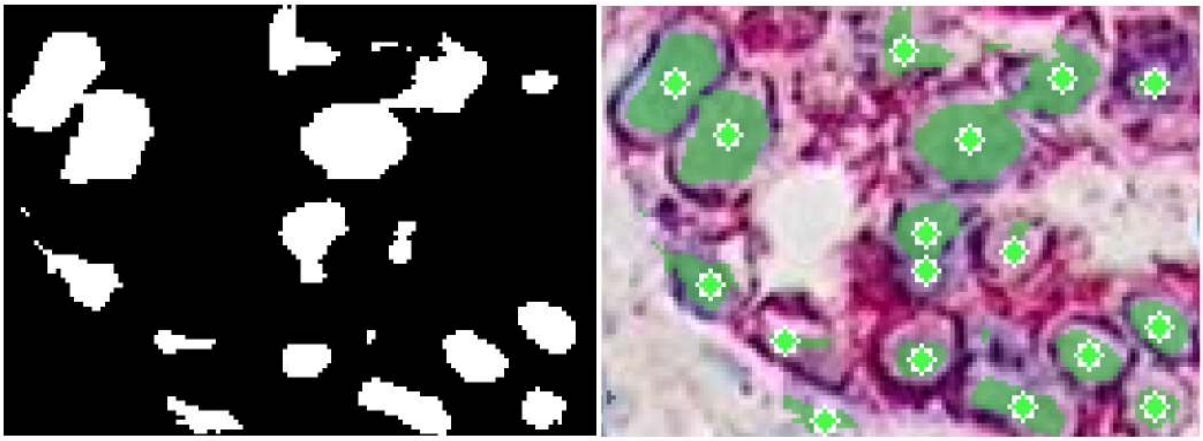

Figure 5: Left - excerpt of B_{cancer} matrix, right - the results of the centroid-finding algorithm superimposed on the marked image

Additional Training Phase(s)

6 Retraining-Reclassification Iterations

After classification (and post-processing if desired), the results from the B's and the C's can be superimposed onto the original training image. The user can add false negatives (hence adding points to the T_p^+'s) and add false positives (hence adding points to T^-). The observation records can then be regenerated, the supervised learning classifier can be relearned, and classification can be rerun with greater accuracy. This retraining-reclassification process can be repeated until the user is satisfied with the results.

3. Visualization Features

In the early stages of the workflow, there are several graphical helpers showing where the training points are. An overview window shows where the individual subimages are situated and how they are labeled (see section 4 for an example), there is also a rare event finder that only displays certain colors when searching for rare cell types or colocalization of two colors.

The user can also interactively identify points by turning them on and off with a mouse click. A magnification window improves the accuracy with which phenotype points are chosen.

In the retraining panel, varying the opacity of the phenotype binary marker matrix B has proven quite useful. (see Fig. 6) The user has an interactive view of the points where there are type I errors and can add points to improve the error rates of that type.

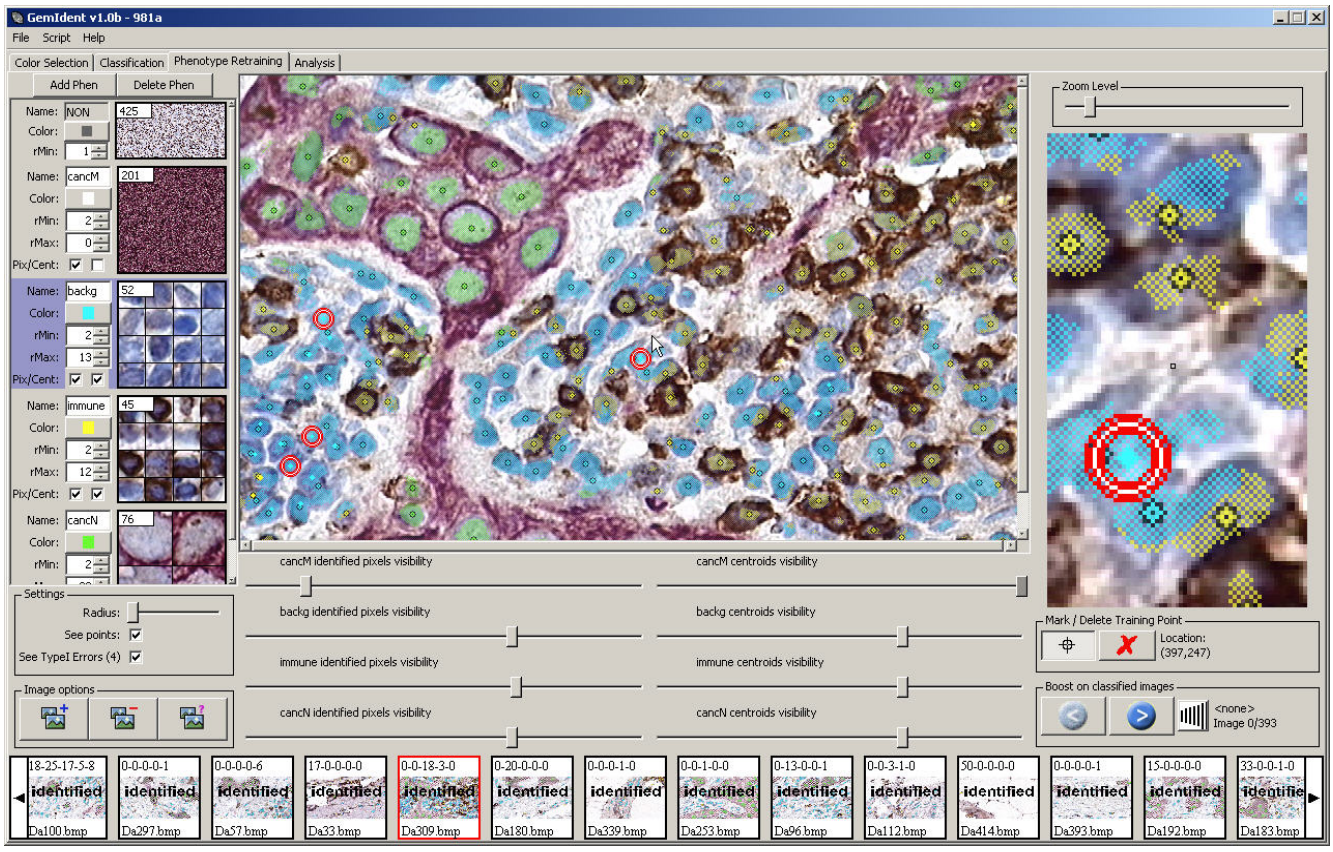

Figure 6: Screenshot of the retraining tool which allows the user to vary the opacity of the B_p's (the sliders in the center bottom) and highlights classification mistakes (the red circles) for easy correction. The example image displayed was scanned at 40x by the optical BacusLabs Bliss Imager(Bacus 2007a)

3.1. First Order Statistics and Output Overview

The data analysis panel (Fig 7) generates histograms and summary reports. Most importantly, the program also allows the user to save the cell center data as a text file that can be used by any specialized statistical package, in the Example section below we show the use of the `spatstat` package in R.

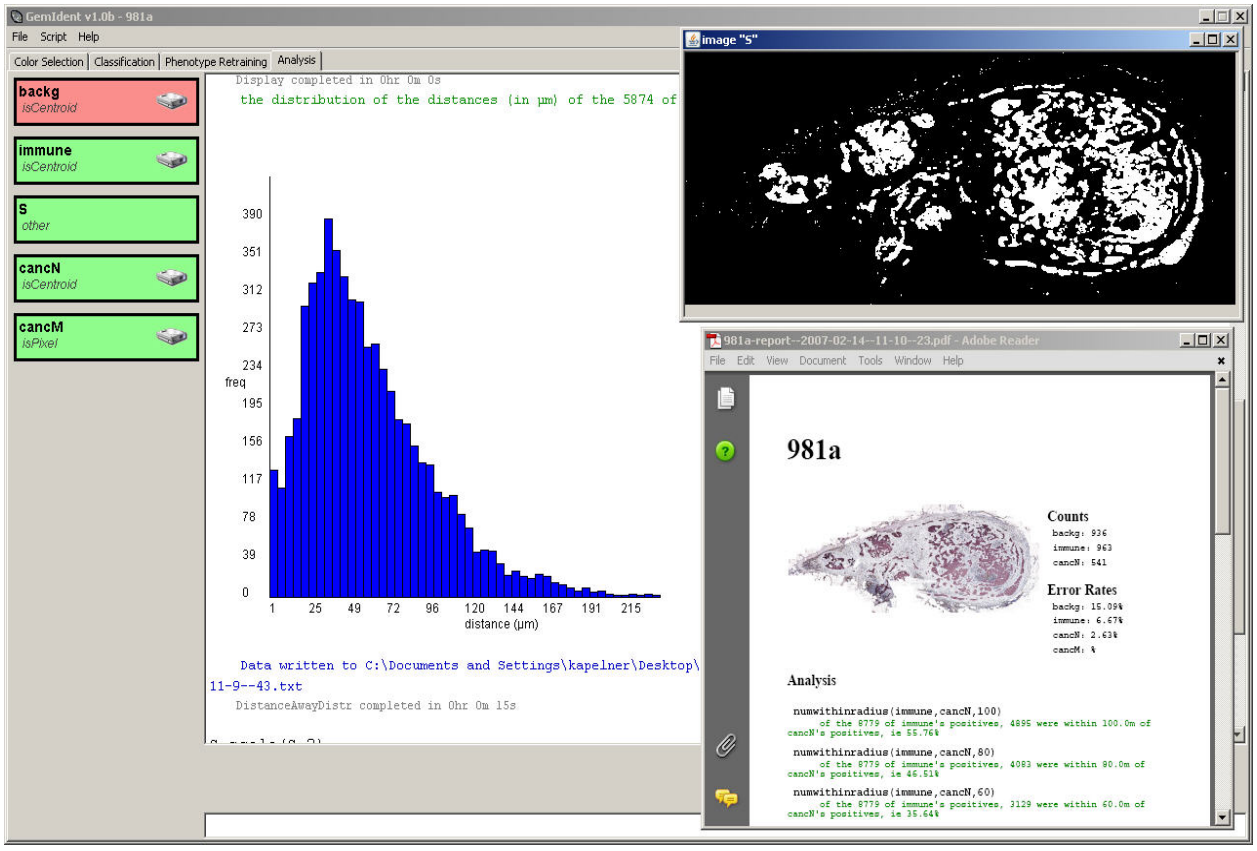

Figure 7: An example of the command-line data analysis panel. The histogram displayed within the panel is the result of a function that finds the distances from all phenotype A centroids to their closest phenotype B neighbors. The binary image is the $B_{CancerMembrane}$. The open PDF document is the autogenerated report which includes a thumbnail view of the entire image set, counts and Type I error rates for all phenotypes, as well as a transcript of the analyses performed.

4. A complete example

Immunohistochemical staining (IHC) refers to the process of identifying proteins in cells of a slice of tissue through the specificity with which certain bind to special antigens present on the cell.

Combining a stain called a chromagen with antibodies allows visualization and reveals their localization within the field. IHC is thus widely used to discover in situ distributions of different types of cells. Until recently most of the data collection was done by manual cell counting. In this example we will show how different types of immune cells as well as cancer cells were detected and localized in large numbers using statistical learning for image segmentation.

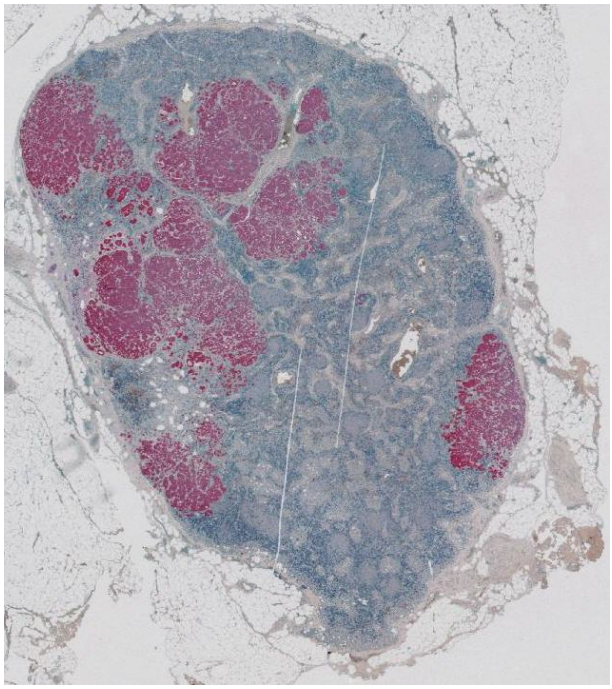

Figure 8: A tumor invaded lymph node as it appears through the microscope, at this resolution we can see the red zones which are Tumor invaded.

4.1. Training Set Helper

When the image set is first opened, a training helper appears. Each number represents the snapshots (called stages) captured individually by the system.

The target phenotypes in this case were `Dendritic cell` nuclei, `T-cell nuclei`, `Tumor` cell nuclei, and unspecific cell nuclei (called `other_cell`).

4.2. Classified Zones

After specifying a set of training pixel phenotypes by clicking on them, the training set is completely classified. Every pixel in the images is assigned a phenotype or put in the 'Non' group. After this, the output is zones or blobs of different phenotypes.

4.3. Interpreting the Classifiers

In order to do the classification, GemIdent creates a random forest (see Section 2 Step 4). When the

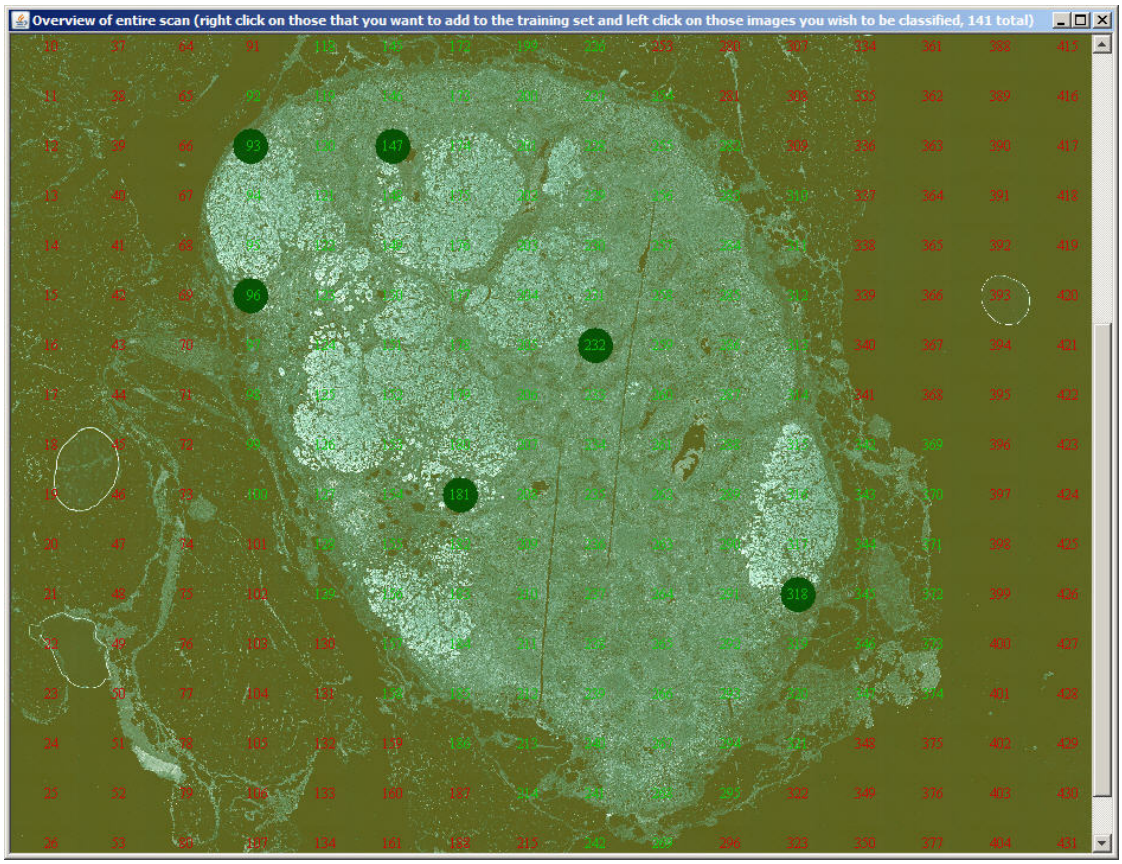

Figure 9: An assembly of all the individual stages into the complete scan, the parts of the images that we want to train and classify are indicated in green, the ones that will be discarded have red numbers, the red disks show the training set.

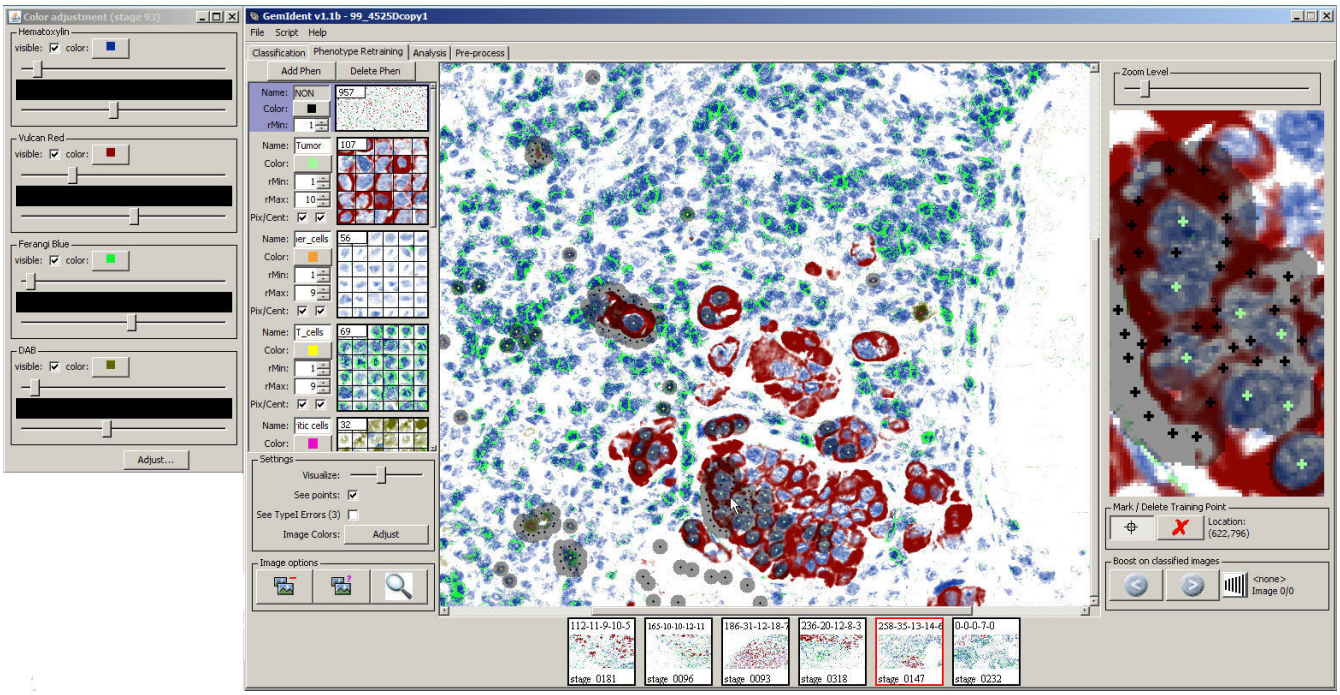

Figure 10: Slides of axillary lymph nodes were imaged and loaded as a new project into `Gemident`. Each slide had the same chromagen profile where CD1a targeting dendritic cells were stained with $3,3'-$Diaminobenzidine (DAB) (these appear as brown in the screenshots), CD4 targeting T-cells were stained with Ferangi Blue (which appear bright green in the screenshot above), AE1/AE3 targeting cytokeratin within breast cancer cells were stained Vulcan Red. In addition, slides were stained with blue Hematoxylin to reveal all nuclei. The intensity at which each of the chromagens is displayed can be adjusted using the "Color Adjustment" dialog box (shown on the left). The pixels chosen as training points are shown as crosses surrounded by grey disks. The color of the crosses indicates the training phenotype chosen (Green for `Tumor`, Black for `Non`).

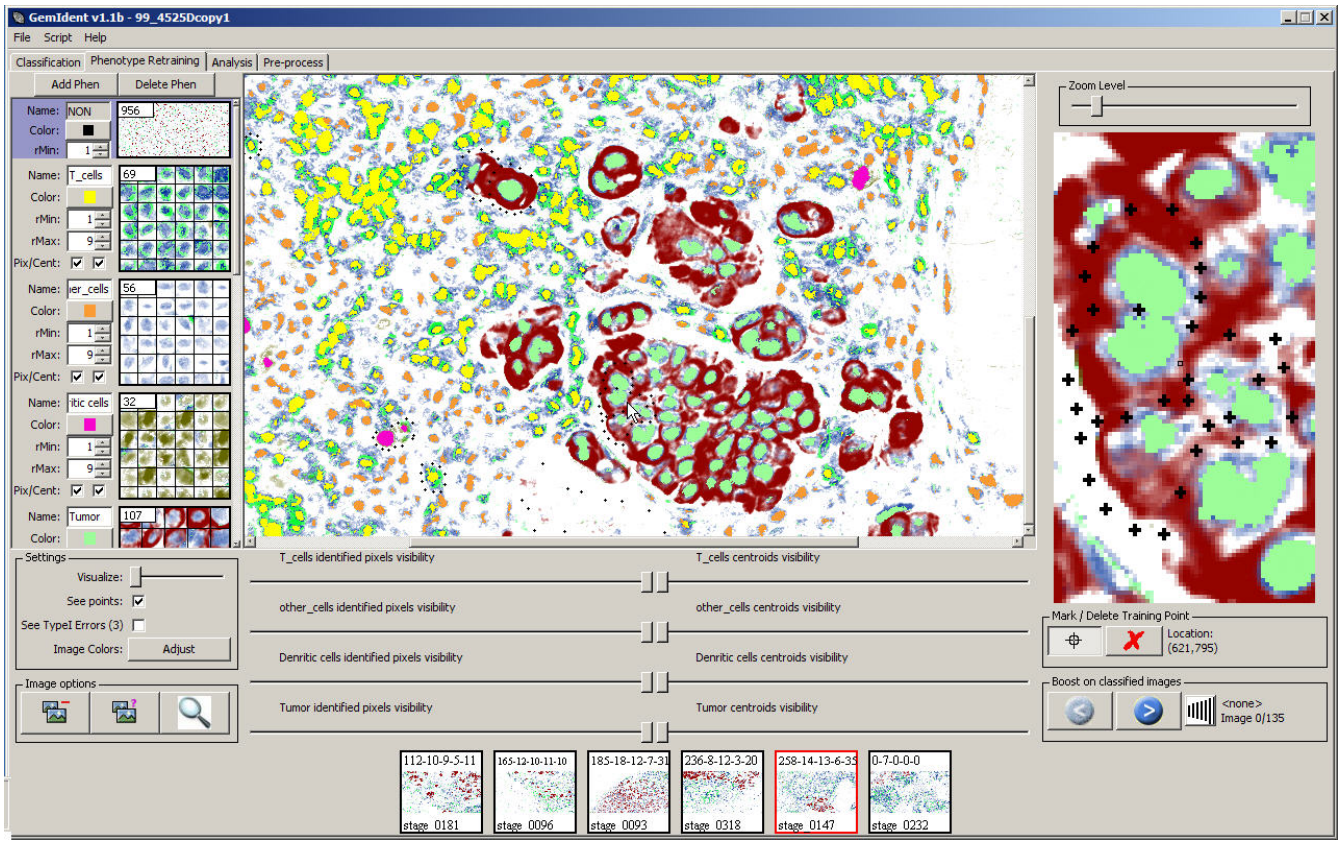

Figure 11: After the training sets have been classified, the pixels in the different phenotypes appear as blobs.

random forest classifier has been created a graphic is displayed illustrating the importances of each feature in the classifier (see Figure 12). Each bar graph represents a chromagen, the x-axes indicate the ring score's radius, and the y-axes indicate relative importance (with the largest bar being the most important). This illustration is important to the user because it serves as an additional check to spot mistakes.

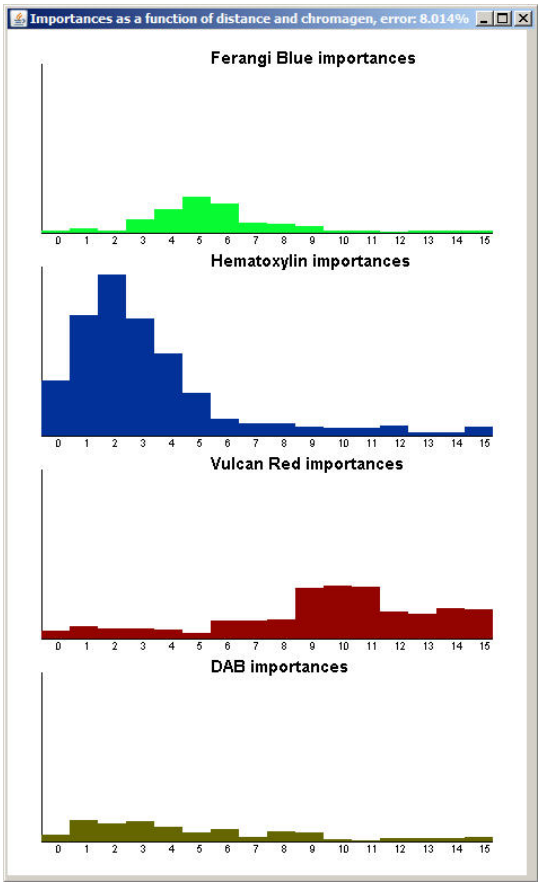

Figure 12: Bar charts for interpreting the distance at which the chromagens influence the classifier. The hematoxylin was found to be very important at $r \in [0, 5]$. The forest learned that cancer cell nuclei, T-cell nuclei, and unspecific cell nuclei all have hematoxylin-rich centers. Ferangi Blue was found to be most important at $r \in [3, 6]$ indicating that the classifier learned that the T-cells are positive at their membranes. Vulcan Red was found to be most important at $r \in [6, 15]$ indicating that cancer cells are positive on their membranes and their diameter varies dramatically. DAB was found to be important at $r \in [1, 6]$ indicating that dendritic cells appear devoid of a nucleus and vary in size.

4.4. Centroid Output

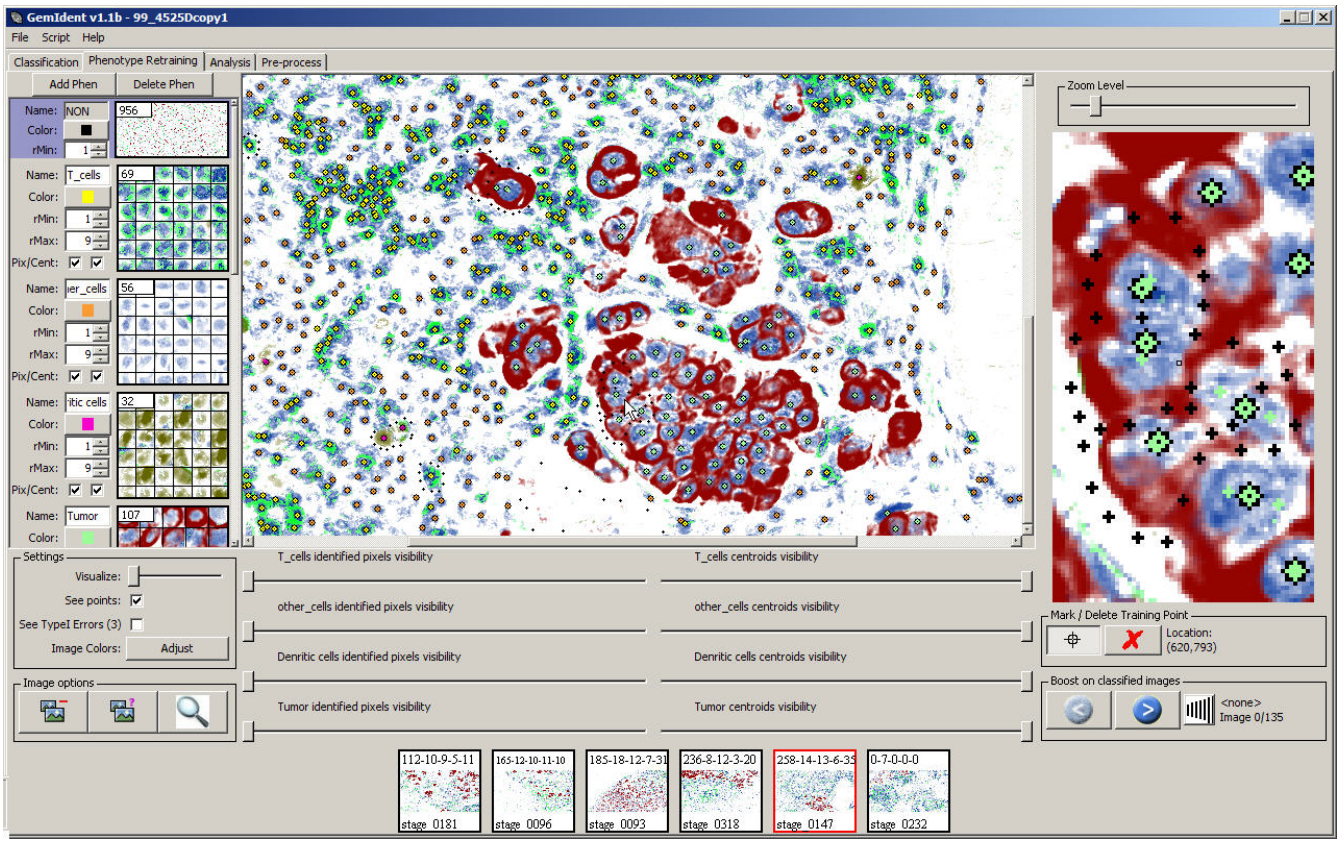

Figure 13: The next step is performed by requesting the program to `Find Centers`, this starts the centroid finding algorithm which designates one pixel as the cell center, thus enabling cell counting and cell localization. Here the cell centroids are marked by black diamonds filled with the relevant phenotype's color code.

After the centroids have been determined the program can evaluate its errors, the type I errors are show in a small window and written to the summary file, in the output directory, which in our case showed:

```
filename,Num_Dendritic_cells,Num_T_cells,Num_other_cells,Num_Tumor
stage_0096,23,482,479,44
stage_0093,29,230,259,615
stage_0126,3,46,166,816
stage_0147,18,1203,907,81
..................
Error Rates
Dendritic_cells         2.78
T_cells                 4.35
other_cells             0.0
Tumor                   7.34
Totals,922,85108,75457,28710
```

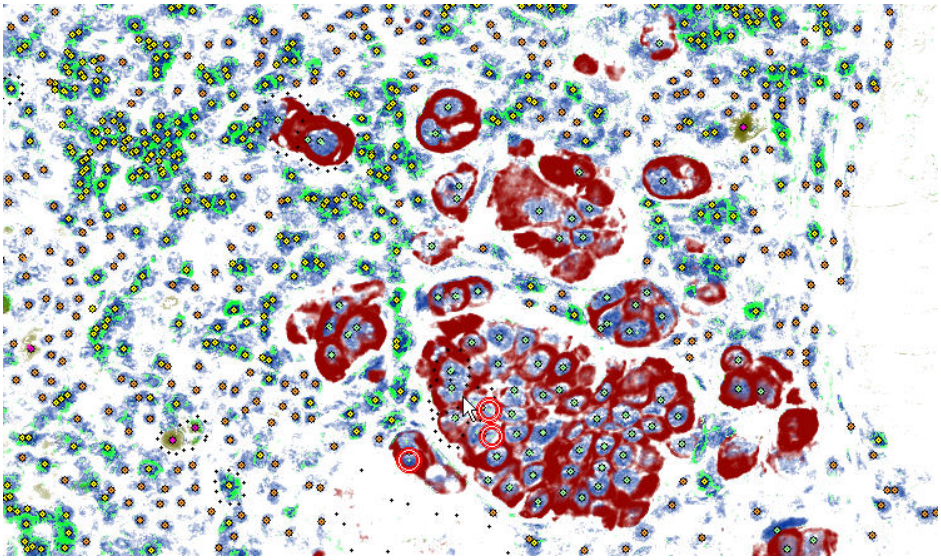

Figure 14: Type I errors are displayed surrounded by red circles, consulting similarities among these errors allows the user to choose which points to add to the training set to improve the accuracy.

4.5. Data Analysis with R

The centroid data is loaded into R as a text file named after the particular phenotype they belong to with 5 columns, the stage number (which is the subimage to which the cell belonged), the local and the global X and Y coordinates, for instance the file `99_4525D-Tumor.txt` contains:

```
filename,locX,locY,globalX,globalY
stage_0096,201,51,4040,13037
stage_0096,214,91,4053,13077
stage_0096,220,76,4059,13062
stage_0096,230,107,4069,13093
.......................
```

This data is easily read into R packages such as spatstat and transformed into an object of the ppp class.

The analysis performed on these data show the spatial landscape of the lymph node immune cells quite clearly.

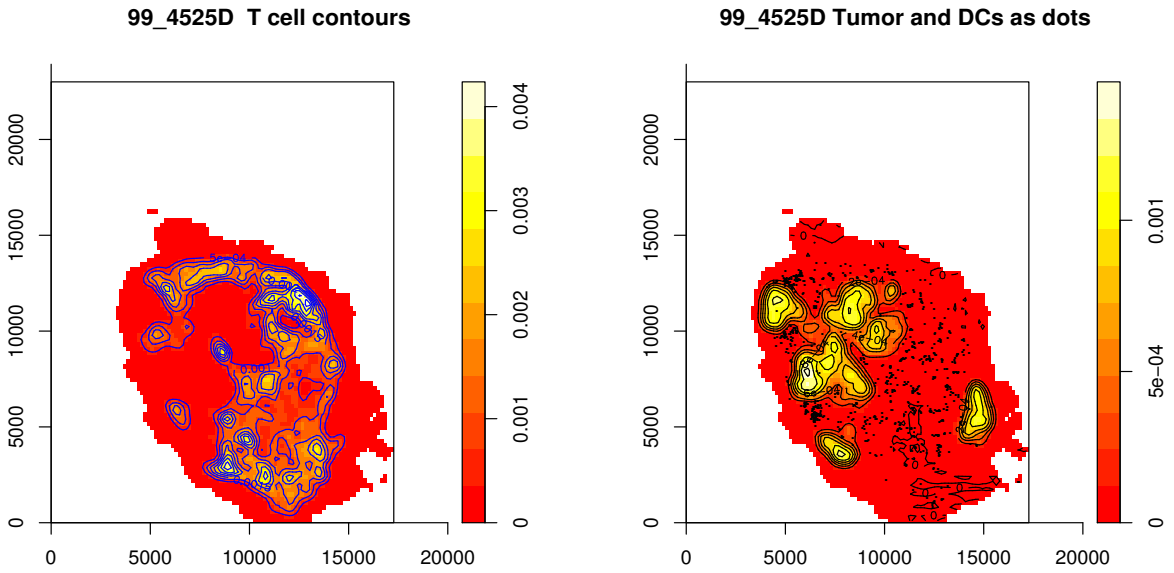

Figure 15: Output from the estimation of spatial densities using kernel density estimates from spatstat.

5. Conclusions and Future Uses

The success of the method resides on the collection of all the data features in a neighborhood of a pixel, and the selection by random forests of the pertinent features for each particular phenotype. The iterative boosting enables the user to increase accuracy to a satisfactory level.

Although we have concentrated here on static multispectral images, fluorescent images can be classified in a similar way. Instead of using distributions in colorspace to obtain scores, density estimation can be used to compute scores in the unidimsensional space of the fluorescent layer intensity images.

Furthermore, the algorithm is not only restricted to static images: Film is a composition of images called "frames" displayed over time. Identification can be done in moving images using the changing frames as the "z-axis" and instead of scores computed via sums of rings, it can be sums of sphere-surfaces. The algorithm can also be generalized to phenotype identification in n-dimensions. The algorithm also may be applied to identification of objects in satellite imagery, face recognition, automatic fruit harvesting and countless other fields.

There is no doubt that images will continue to provide data to solve many biological mysteries. The GEMIDENT project is a step towards combining human expertise and statistical learning to supply a feasible and objective data collection process in the presence of high case to case variability.

Acknowledgements

We thank Holbrook Kohrt for many useful insights and recommendations, Adam Guetz for a careful reading of the manuscript. We thank Kyle Woodward for help designing and implementing a GUI

for GEMIDENT and Francesca Setiadi for the Data Collection and helpful suggestions. We thank CVSDude.com for providing a home for GemIdent's source code. The referees for JSS provided comments which helped improve the paper. This work was funded by a DOD Era Hope Scholar grant to PPL and by NSF-DMS award 02-41246 to SH.

Appendix

Simple Blob Analysis Algorithm

The first step of the blob analysis is the creation of heuristic rules built from the training data:

Step 1 For all points $\mathbf{t} \in T$ (the training set for a phenotype), we verify if they are inside a blob corresponding to this phenotype. If so, use the floodfill algorithm(Wikipedia 2007) to extract the containing blob's coordinates. The collection of these containing blobs is represented by: $\Omega = \{\omega_1, \omega_2, \ldots, \omega_N\}$ These are our reference blobs.

Step 2 For each of the reference blobs, count the number of pixels contained within and record the sizes in the vector $\mathbf{v}$. Now, record reference statistics about: the low threshold value, v_L, (GEMIDENT uses the 2^{nd} percentile), the median, v_M, and the upper threshold value, v_H (GEMIDENT uses the 95^{th} percentile).

We are now going to use the insight into blob sizes in the reference statistics to find centroids for the blobs obtained from the classification:

Step 1 A floodfill algorithm(Wikipedia 2007) is used to extract all blobs, to create the collection Ω.

Step 2 For each blob in the collection, ω, measure its size, v. If $v < v_L$, ignore it — the blob is too small and is most likely noise. If $v_L \leq v \leq v_H$, find the (x, y) center of ω and set $C(x, y)$ true (indicating the presence of a centroid). In a normal application, where the phenotypes in the image are fairly spaced apart, these two tests will cover greater than 90% of the blobs. If $v > v_H$, ie the blob is large, it must be split and multiple centroids must be marked. Proceed to Step A.

Step A Define $n = floor(v/v_M)$. Cut the large blob into n sub-regions using a square mask of semiperimeter $s = \sqrt{\frac{v}{n\pi}}$, the radius of the average circle if the large blob's pixels were split into discs. Mark the centers of each of these cut squares in C.

In fact we use several levels of such statistics to improve the centroid calculations.

References

Bacus (2007a). "Bacus Bliss Imager." URL http://www.bacuslabs.com/bliss.html.

Bacus (2007b). "Nanozoomer." URL www.bacuslabs.com.

Breiman L (2001). "Random Forests." *Machine Learning*, **45**(1), 5–32. URL citeseer.ist.psu.edu/breiman01random.html.

Chakraborty A (2003). *An attempt to perform Bengali optical character recognition*. Ph.D. thesis, Stanford University.

Collins TJ (2007). "ImageJ for microscopy." *BioTechniques*, **43**(1 Suppl), 25–30. URL http://www.ncbi.nlm.nih.gov/pubmed/17936939?dopt=abstract.

Fang B, Hsu W, Lee ML (2003). "On the accurate counting of tumor cells." *Nanobioscience, IEEE Transactions on*, **2(2)**, 94–103.

Freund Y, Schapire R (1997). "A decision-theoretic generalization of on-line learning and an application to boosting." *Journal of Computer and System Sciences*, **55**(1).

Gil J, Wu H, Wang BY (2002). "Image analysis and morphometry in the diagnosis of breast cancer." *Microsc. Res. Tech.*, **59**, 109–118.

Gonzalez RC, Woods RE, Eddins SL (2004). *Digital Image Processing Using Matlab*, pp. 245–255, 352–364. Pearson Education.

Hastie T, Tibshirani R, Friedman J (2001). *The Elements of Statistical Learning*. Springer, NY.

Ihaka R, Gentleman R (1996). "R: A Language for Data Analysis and Graphics." *Journal of Computational and Graphical Statistics*, **5**(3), 299–314.

Kapelner A, Holmes S, Lee P (2007). "Gemident." URL http://gemIdent.com.

Kohrt HE, Nouri N, Nowels K, Johnson D, Holmes S, Lee PP (2005). "Profile of Immune Cells in Axillary Lymph Nodes Predicts Disease-Free Survival in Breast Cancer." *LoS Med*, **2(9)**, e284.

Kovalev V, Harder N, Neumann B, Held M, Liebel U, Erfle H, Ellenberg J, Neumann B, Eils R, Rohr K (2006a). "Feature Selection for Evaluating Fluorescence Microscopy Images in Genome-Wide Cell Screens." *cvpr*, **1**, 276–283. ISSN 1063-6919. doi:http://doi.ieeecomputersociety.org/10.1109/CVPR.2006.121.

Kovalev V, Harder N, Neumann B, Held M, Liebel U, Erfle H, Ellenberg J, Neumann B, Eils R, Rohr K (2006b). "Feature Selection for Evaluating Fluorescence Microscopy Images in Genome-Wide Cell Screens." *cvpr*, **1**, 276–283. URL http://doi.ieeecomputersociety.org/10.1109/CVPR.2006.121.

Lee J, Woodyatt A, Berman M (1990). "Enhancement of high spectral resolution remote sensing data by a noise-adjusted principal components transform." *Geoscience and Remote Sensing*, **28**, 295–304.

Levenson RM (2006). "Spectral Imaging Perspective on Cytomics." *Cytometry*, **69A**(7), 592–600. URL http://www.cri-inc.com/.

Maggioni M, Warner FJ, Davis GL, Coifman RR, Geshwind FB, Coppi AC, DeVerse RA (2004). "Algorithms from Signal and Data Processing Applied to Hyperspectral Analysis: Application to Discriminating Normal and Malignant Microarray Colon Tissue Sections." *submitted*.

Mahalanobis A, Kumar BVKV, Sims SRF (1996). "Distance-classifier correlation filters for multi-class target recognition." *Appl. Opt.*, **35**, 3127.

Nuance (2007). "CRI." URL http://www.cri-inc.com/products/nuance.asp.

Ortiz de Solirzano C, Garcea Rodriguez E, Jones A, Pinkel D, Gray JW, Sudar D, Lockett SJ (1999). "Segmentation of confocal microscope images of cell nuclei in thick tissue sections." *Journal of Microscopy*, **193(3)**, 212–226.

Skylar O, Huber W (2006). "Image Analysis for microscopy screens: Image Analysis and processing with EBimage." *R News*, **6**(5), 12–15. URL http://cran.r-project.org/doc/Rnews/Rnews_2006-5.pdf.

Wikipedia (2007). "Wikipedia Entry for Flood fill." URL http://en.wikipedia.org/wiki/Flood_fill.

Wu HS, Barba J, Gil J (1998). "A parametric fitting algorithm for segmentation of cell images." *Biomedical Engineering, IEEE Transactions*, **45(3)**, 400–407.

Wu K, Gauthier D, Levine MD (1995). "Live Cell Image Segmentation." *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING BME*, **42(1)**.

Yang Q, Parvin B (2003). "Harmonic cut and regularized centroid transform for localization of subcellular structures." *Biomedical Engineering, IEEE Transactions*, **50(4)**, 469–475.

Affiliation:

Susan Holmes
Department of Statistics
Sequoia Hall
Stanford
CA 94305
E-mail: susan@stat.stanford.edu
URL: http://www-stat.stanford.edu/~susan/