Few Ideas for Counting Glomeruli

1 Problem Setup

Let us assume that the features we extracted for each pixel can be associated to two graphs: (a) the neighborhood graph G, and the (c) spatial neighborhood graph H. The goal is to design an algorithm to count the glomeruli in an image. We denote the features obtained from each pixel as x_i, where $i = \{1, \ldots, T\}$, and the binary class indicator for the glomeruli as y_i. The easiest way is to cluster the features into two classes to get a binary image, extract features from the image and learn a regression model from the features with the count as the response.

Another way is to actually define functions over vertices of the two graphs, where the value of the function is the probability that the particular pixel is a glomerular pixel. We will then define a regression model over these function values and the response now will be the count. Alternatively we can use the training or validation data preferably, to find the best threshold for the function values to result in a good count.

The optimization for determining the function values (logistic probabilities) will be

$$
\begin{aligned}
\hat{\boldsymbol{\theta}} \;=\; \mathrm{argmin}_{\boldsymbol{\theta}} \sum_i \mathcal{L}(\boldsymbol{\theta}^T \mathbf{x}_i, y_i) + \lambda_1 \sum_{i,j} g_{i,j} |\mathrm{logit}^{-1}(\boldsymbol{\theta}^T \mathbf{x}_i) - \mathrm{logit}^{-1}(\boldsymbol{\theta}^T \mathbf{x}_j)| \\
+ \lambda_2 \sum_{i,j} h_{i,j} |\mathrm{logit}^{-1}(\boldsymbol{\theta}^T \mathbf{x}_i) - \mathrm{logit}^{-1}(\boldsymbol{\theta}^T \mathbf{x}_j)|
\end{aligned}
\tag{1}
$$

and the estimated logistic probabilities are

$$
\hat{f}_i = \mathrm{logit}^{-1}(\hat{\boldsymbol{\theta}}^T \mathbf{x}_i). \tag{2}
$$

In (1), $g_{i,j}$ and $h_{i,j}$ are the elements of the adjacency matrices of the graphs G and H, $\mathcal{L}$ is the logistic loss function, and logit^{-1} is the inverse logit function. (*We need to see how easy/hard is it to solve (1) - check convexity etc. Else we may have to solve it as alternating optimization, first compute logistic probabilities, then regularize using the graph and repeat – NRK*).

Now the estimated function value for an image $\hat{\mathbf{f}}$ can be regressed against the count for multiple images. The only problem we will face is that $\hat{\mathbf{f}}$ is extremely high dimensional, and so we will have a severely under-determined system.