Medical Imaging Computed Tomography

CT Image Deconvolution in the Presence of Poisson Noise

YongHwan Lee and Tony Storey

ECE565

Estimation Filtering and Detection

Many diseases and tissue anomalies, such as cancer, can be diagnosed through modern, high-resolution, X-ray Computed Tomography (CT) scans. Unfortunately, with high quality CT images, exposure to X-ray particles (radiation) go up. At higher radiation levels the signal-to-noise ratio is high, producing highly detailed images, but this leads to higher patient radiation exposure. At low level radiation dosages, the recovered image suffers from a phenomenon known as shot noise. This noise is due to the quantum phenomenon of particles and the discrete nature on when they arrive. This can make the image look streaky or as though there are anomalies present, these anomalies can cause false diagnoses. To handle this case, we observe that this “shot noise” follows a Poisson distribution [1]. Because of this we can derive a maximum likelihood estimator to reduce the noise recovered in the CT scan.

I. For a better understanding of how the CT images are captured we will be looking at the problem on a small, 3x3 pixel scale. In practice 512x512, 1024x1024, and 2048x2048 images are used [2]. This means the cross section of the area of interest in a patient would contain up to 4,194,304 voxels. This makes for a very large matrix. This would require an even larger matrix than that to ensure there were enough equations to solve for all the unknown voxel absorption coefficients. For a 3x3 pixel image, there must be at least 16 particle rays (line integrals) to get enough equations to solve for all the unknown absorption coefficients. See fig 1 below. As the particles travel through each voxel, they are absorbed. We can express this as a sum of logs as seen in equation 1. This will be written as a sum of theta values seen in equation 2. With all the paths we have chosen this will give us an mxn (16x9) matrix of 0s and 1s to indicate if the voxel was part of the path. See equation 2, 3 and 4. The matrix describing our model can be seen below (5).

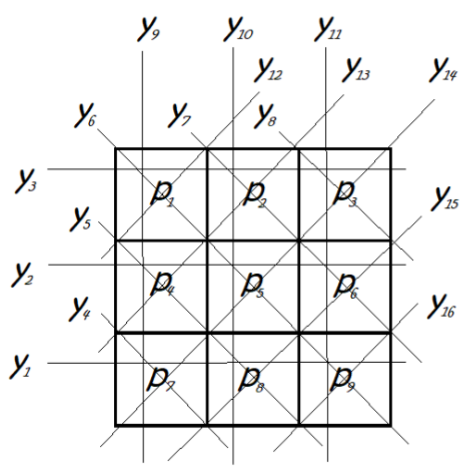

Figure 1—Voxel Cross-section of particle rays

$$\log \frac{P_{out}}{P_{in}} = \log P_7 + \log P_8 + \log P_9 \tag{1}$$

$$Y_1 = \theta_7 + \theta_8 + \theta_9 \tag{2}$$

$$\begin{bmatrix} 0&0&0&0&0&0&1&1&1 \\ 0&0&0&1&1&1&0&0&0 \\ 1&1&1&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&0&0 \\ 0&0&0&1&0&0&0&1&0 \\ 1&0&0&0&1&0&0&0&1 \\ 0&1&0&0&0&1&0&0&0 \\ 0&0&1&0&0&0&0&0&0 \\ 1&0&0&1&0&0&1&0&0 \\ 0&1&0&0&1&0&0&1&0 \\ 0&0&1&0&0&1&0&0&1 \\ 1&0&0&0&0&0&0&0&0 \\ 0&1&0&1&0&0&0&0&0 \\ 0&0&1&0&1&0&1&0&0 \\ 0&0&0&0&0&1&0&1&0 \\ 0&0&0&0&0&0&0&0&1 \end{bmatrix} = \text{A} \tag{4}$$

$$\underline{Y} = A\underline{\theta} \tag{5}$$

The parameter vector is the collection of theta values with observations y. Where m is 16 and n is 9, the Poisson probability mass function, observations and parameter vector can be seen below (6).

$$Poisson(Y_1, Y_2, Y_3, .. Y_m | \theta_1, \theta_2, \theta_3, .. \theta_n) = \frac{((A\theta))^Y e^{-(A\theta)}}{Y!} \tag{6}$$

II. Once we have an idea of the framework for the problem, we will next compute the Fisher Information Matrix (FIM) and the Cramer-Rao Lower Bound (CRLB). This will tell us the best possible mean squared error we can hop to achieve, assuming an efficient estimator. Where e subscript-j is the canonical (one-hot) vector of all 0s and a single 1 to select a single element. The joint of the Poisson distribution for our system can be seen below (7, 8).

$$\prod Poisson(Y_j | \lambda_j) = \prod \frac{(e_j^T (A\theta))^{Y_j} e^{-(e_j^T A\theta)}}{Y_j!} \tag{7}$$

$$\lambda_j = (A\theta)_j \tag{8}$$

We next find the log likelihood for the Poisson distribution (9).

$$\log \prod Poisson(Y_j|\lambda_j) = \sum_{j=1}^{m}(Y_j * \log(e_j^T(A\theta)) - (e_j^T(A\theta)) - \log(Y_j!)) \tag{9}$$

The next step is to find the derivative of the log likelihood for the Poisson distribution (10).

$$\frac{d}{d\theta}\log \prod Poisson(Y_j|\lambda_j) = \sum_{j=1}^{m}(\frac{A^T e_j Y_j}{e_j^T A\theta}) - (e_j^T A) \tag{10}$$

The second derivative of the log likelihood for the Poisson distribution (11).

$$\frac{d^2}{d\theta d\theta^T}\log \prod Poisson(Y_j|\lambda_j) = \sum_{j=1}^{m} -(\frac{A^T e_j e_j^T A * Y_j}{e_j^T A\theta(A\theta)^T e_j}) \tag{11}$$

Finally, the last step is to take the expectation of the negative second derivative of the log likelihood (12).

$$\mathrm{E}[-\frac{d^2}{d\theta d\theta^T}\log \prod Poisson\left(Y_j\middle|\lambda_j\right)] = \sum_{j=1}^{m}(\frac{A^T e_j e_j^T A * \mathrm{E}[Y_j])}{e_j^T A\theta(A\theta)^T e_j}) \tag{12}$$

The expectation with respect to the sample random observation variable y is the mean and variance of the Poisson distribution as seen below (13, 14).

$$\mathrm{E}\left[Y_j\right] = \lambda_j = (e_j^T(A\theta)) \tag{13}$$

$$\mathrm{E}[-\frac{d^2}{d\theta d\theta^T}\log \prod Poisson\left(Y_j\middle|\lambda_j\right)] = \sum_{j=1}^{m}(\frac{A^T e_j e_j^T A * (e_j^T(A\theta))}{e_j^T A\theta(A\theta)^T e_j}) \tag{14}$$

Simplifying the expression gives us the Fisher information Matrix (15,16), inverting this gives is the Cramer-Rao Lower Bound.

$$\mathrm{E}[-\frac{d^2}{d\theta d\theta^T}\log\prod Poisson\left(Y_j\middle|\lambda_j\right)\,] = \sum_{j=1}^{m}(\frac{A^T e_j e_j^T A)}{(A\theta)^T e_j}) \tag{15}$$

$$\mathrm{FIM} = \sum_{j=1}^{m}(A^T e_j\left({e_j}^T A\theta\right)^{-1} e_j^T A)) \tag{15}$$

$$\mathrm{CRLB} = \sum_{j=1}^{m}(A^T e_j\left({e_j}^T A\theta\right)^{-1} e_j^T A))^{-1} \tag{19}$$

III. The next step is showing how the maximization of the objective function for maximum likelihood is equivalent to minimizing the Kullback-Leibler divergence, see equations (20, 21 and 22) below for our objective function to optimize.

$$\log\prod Poisson(Y_j|\lambda_j)|_{\theta=\hat{\theta}} = 0 \tag{20}$$

$$\hat{\theta}_{MLE} = \arg\max_{\theta}\sum_{j=1}^{m}(Y_j\log(\lambda_j) - (\lambda_j) - \log(Y_j!)) \tag{21}$$

$$\hat{\theta}_{MLE} = \arg\min_{\theta}\sum_{j=1}^{m}(-Y_j\log(\lambda_j) + \left(\lambda_j\right) + \log(Y_j!)) \tag{22}$$

The KL divergence can be seen below (23), first we add the integral of the denominator from the KL log fraction and subtract the integral of the numerator (24).

$$D_{KL}(P\|Q) = \int p(x)\log\left(\frac{p(x)}{q(x)}\right)dx \tag{23}$$

$$D_{KL}(P\|Q) = \int p(x)\log\left(\frac{p(x)}{q(x)}\right)dx + \int q(x)dx - \int p(x)dx \tag{24}$$

We can do the same with the objective of the MLE and collect log terms back into a fraction (25, 26).

$$\hat{\theta}_{MLE} = \arg\min_{\theta} \sum_{j=1}^{m}(-Y_j \log(\lambda_j) + (\lambda_j) + \log(Y_j!)) + \sum_{j=1}^{m}(Y_j \log(Y_j)) - \sum_{j=1}^{m}(Y_j) \quad (25)$$

$$\hat{\theta}_{MLE} = \arg\min_{\theta} \sum_{j=1}^{m}(-Y_j \log(\lambda_j) + (\lambda_j)) + \sum_{j=1}^{m}(Y_j \log(Y_j)) - \sum_{j=1}^{m}(Y_j) + constant \quad (26)$$

This ends up taking on the same form as the KL divergence (27, 28).

$$\hat{\theta}_{MLE} = \arg\min_{\theta} \sum_{j=1}^{m}(Y_j \frac{\log(Y_j)}{\log(\lambda_j)}) + \sum_{j=1}^{m}(\lambda_j) - \sum_{j=1}^{m}(Y_j) + constant \quad (27)$$

$$D_{KL}(P\|Q) = \hat{\theta}_{MLE} = \arg\min_{\theta} \sum_{j=1}^{m}(Y_j \frac{\log(Y_j)}{\log(\lambda_j)}) + \sum_{j=1}^{m}(\lambda_j) - \sum_{j=1}^{m}(Y_j) + constant \quad (28)$$

IV. As the particles travel along their respective rays from voxel to voxel, they are absorbed, this absorption follows a Poisson distribution for each voxel. This means the sum of their absorption along the path include hidden latent observations following unique Poisson distributions. See equation (29, 30).

$$Y_1 = Y_{1,7} + Y_{1,8} + Y_{1,9} \quad (29)$$

Where: $$Y_{1,j} = Poisson(A\underline{\theta}_j) \quad (30)$$

Looking back to our objective equation we see by finding the log likelihood of the joint Poisson PMFs, we can see that we have a function that is not analytically optimizable (31).

$$\log \prod \prod Poisson(Y_{ij}|\lambda_j)|_{\theta=\hat{\theta}} = \sum_{i=1}^{n} \sum_{j=1}^{m}(Y_{ij} * \log(e_j^T(A\hat{\theta})) - (e_j^T(A\hat{\theta})) - \log(Y_{ij}!)) = 0 \quad (31)$$

This is where Expectation Maximization can be used to optimize the objective. For the E-step, we first find a surrogate function by finding the expectation of the complete-data log likelihood (32, 33).

$$Q(\theta, \acute{\theta}) = \mathrm{E}[\log P(Y_j, Y_{ji}|\theta) \,|Y_j; \acute{\theta}] \quad (32)$$

$$Q(\theta, \acute{\theta}) = \mathrm{E}[\sum_{i=1}^{n} \sum_{j=1}^{m} Y_{ji} \log\left(A_{ji}\theta_i\right) - A_{ji}\theta_i - \log(Y_{ji}!)] \tag{33}$$

For a Poisson mixture model, we can treat the expectation of the complete-data log likelihood as a binomial distribution and use the expectation of the Binomial distribution to plug in our surrogate (34).

$$Q(\theta, \acute{\theta}) = \sum_{i=1}^{n} \sum_{j=1}^{m} (\mathrm{E}[Y_{ji} | Y_j; \acute{\theta}] \log\left(A_{ji}\theta_i\right) - A_{ji}\theta_i - \mathrm{E}[\log\left(Y_{ji}!\right) | Y_j, \acute{\theta}]) \tag{34}$$

The expectation on the right-hand side of the expression can be seen as a constant because we are optimizing with respect to theta so we can drop it from our surrogate. We have a mixture of Poisson distributions as seen in equation (29) above, so as previously stated we can use a binomial distribution and its expectation in our surrogate (35)

$$\mathrm{E}[Y_{ji} | Y_j; \acute{\theta}] = \frac{Y_j A_{ji} \theta_i}{\sum_{k-1}^{n} A_{jk} \theta_k} \tag{35}$$

The M-step is where we maximize our surrogate to find an update equation to iterate over and find the maximum for our objective function by setting it equal to 0 and solving for theta as seen below in equations (36, 37, 38 and 39).

$$\frac{d}{d\theta_i} Q(\theta, \acute{\theta}) = \sum_{j=1}^{m} \mathrm{E}[\, Y_{ji} | Y_j; \acute{\theta}] \frac{A_{ji}}{A_{ji}\, \theta_i} - \sum_{j=1}^{m} A_{ji} \tag{36}$$

$$\sum_{j=1}^{m} \mathrm{E}[\, Y_{ji} | Y_j; \acute{\theta}] \frac{1}{\theta_i} = \sum_{j=1}^{m} A_{ji} \tag{37}$$

$$\sum_{j=1}^{m} \mathrm{E}[\, Y_{ji} | Y_j; \acute{\theta}] \frac{1}{\sum_{j=1}^{m} A_{ji}} = \theta_i \tag{38}$$

$$\sum_{j=1}^{m} \frac{Y_j A_{ji}}{\sum_{k-1}^{n} A_{jk} \acute{\theta}_k} \frac{\acute{\theta}_i}{\sum_{j=1}^{m} A_{ji}} = \theta_i \tag{39}$$

V. We implemented the expectation maximization algorithm in MATLAB to reduce the noise and estimate the 9 parameters from our 3x3 pixel matrix. We ran 200 Monte Carlo simulations and collected the resulting expectation maximization results to recover the data:

As the EM algorithm converged on a solution, we found a monotonically increasing likelihood. See figure 2.

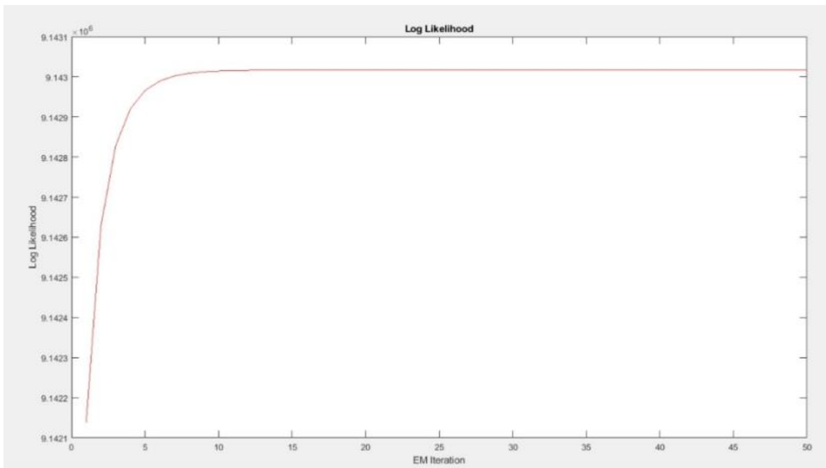

Figure 2. MATLAB Plot of Log Likelihood

After sweeping the signal gain from 0.1 to 100 we found both the Cramer-Rao Lower Bound and the MSE for each parameter. See Figure 3.

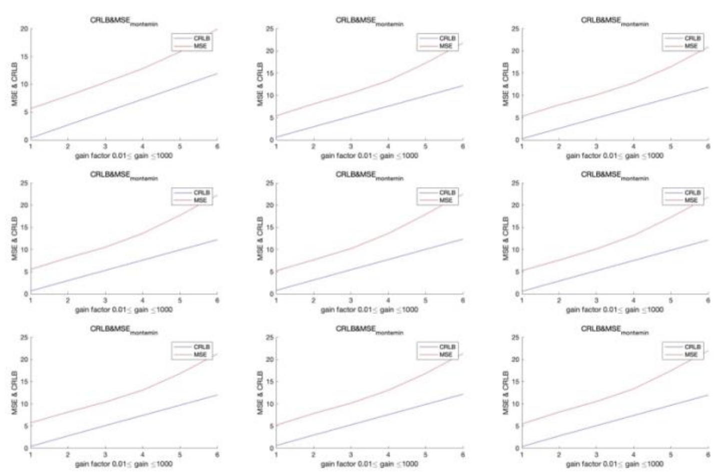

Figure 3. MATLAB plot of Cramer-Rao Lower Bound and MSE

We can see in the plot below the EM algorithm converging on the true value of the parameters. See below figure 4.

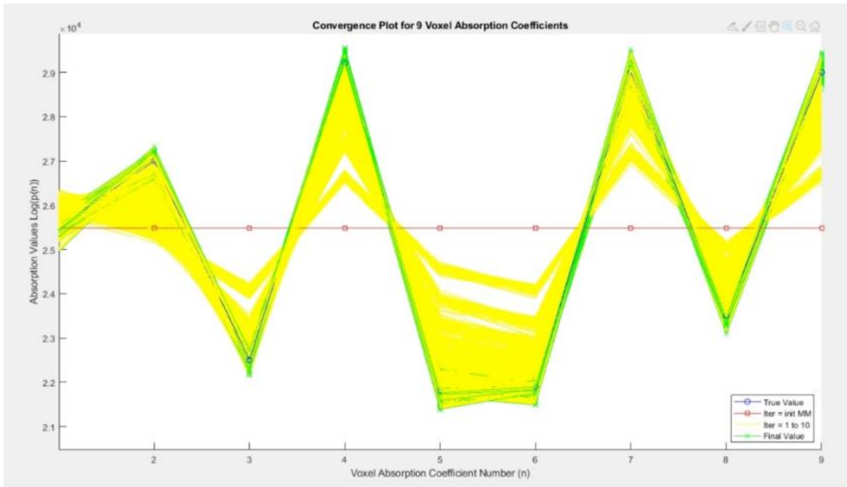

Figure 4. MATLAB Plot Showing EM Data Convergence

We chose a method of moments estimator via a least-squares approximation as a starting point. As one can see, the EM algorithm does not need to travel too far to converge on a final solution. See figure 5.

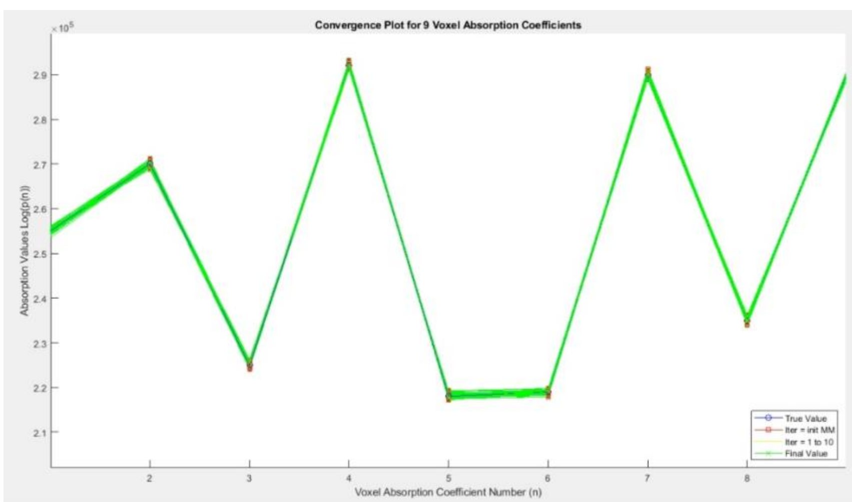

Figure 5. MATLAB Plot Showing EM Data Convergence from MOM Initialization

References

[1] Buzwig, Thorsten. *Computed Tomography from Photon Statistics to Modern Cone-Beam CT*. Springer-Verlag Berlin Heildelberg, 2010.

[2] Hata, Akanori et al. *Effect of Matrix Size on the Image Quality of Ultra-high-resolution CT of the Lung*. Academic Radiology, vol 25 issue 7, 28 July. 2018, https://www.sciencedirect.com/science/article/pii/S1076633217305044 Accessed 24 Dec 2019.