Uncertainty Analysis in SPECT Reconstruction based on Probabilistic Programming

Manu Francis¹, Muhammed Tarek¹, Mark Pickering¹ and Murat Tahtali¹

¹SEIT, UNSW Canberra, Australia
E-mail: manu.francis@student.adfa.edu.au

Abstract

Single Photon Emission Computed Tomography (SPECT) is one of the nuclear medicine imaging modalities used for functional analysis of animal or human organs. Gamma rays emitted from the scanned body are filtered with collimators and detected by the SPECT head. The conventional reconstruction algorithms do not consider the uncertainty in the process introduced by the field of view of the collimators. In this paper, we incorporate the probabilistic programming approach for SPECT image reconstruction. No-U-Turn Sampler (NUTS) is used to estimate the scanned object system by considering uncertainty. Results indicate that the current work can include uncertainty in reconstruction compared to conventional approaches like MLEM and MAP. However, reconstruction time need to be improved for phantom sizes of 128x128x128 voxels and higher.

Keywords: Image Reconstruction, Probabilistic Programming, Uncertainty Analysis, SPECT

1. Introduction

Single Photon Emission Computed Tomography (SPECT) systems use emitted gamma radiation. This gamma radiation is generated by the infused radiolabelled medicines into the living objects that need to be scanned. The emitted gamma radiation is filtered through collimators and detected by gamma detectors made up of a scintillator and an array of photomultipliers. Projection images represent the distribution of radionuclide from 3D to a 2D plane for a specific view angle [1]. The projection for one view angle is not enough to estimate the distribution of the gamma sources inside the object. Therefore, multiple projection images at different view angles are obtained to reconstruct the gamma emission distribution. This is achieved by rotating the SPECT camera at discrete angles around the scanned body.

The mathematical representation that models forward projection SPECT imaging is:

\[ g = Af \]  

(1)

Where \( g \) is the projection, \( A \) is the system matrix, and \( f \) is the scanned phantom. The image reconstruction aims to find \( f \) by solving equation 1.

The distribution of the radionuclide can be identified from projection images by using certain reconstruction algorithms. The main idea behind the reconstruction is to find the radionuclide distribution by using the back-projection operator and projection images, as discussed next. Analytical and iterative methods are the two main reconstruction algorithm categories.

Analytical methods are simple and mainly based on back-projection. In these methods, the scanned object is reconstructed slice by slice. The projection of each sample is arranged row-wise, called a sinogram. This sinogram is used in the reconstruction algorithm. Each element is back-projected to the area of interest, based on the direction of radiation. The shape and orientation of the collimator provides information about the direction of the radiation. Each voxel in the cube of interest is updated by summing the contribution of the sinogram elements. An advance method of back projection is called filtered back-projection (FBP). Each row of the sinogram is filtered to reduce the amplitude of low frequency components during reconstruction by using FBP algorithm [2]. The back-projection based reconstruction is mainly employed for the Parallel Beam and Fan Beam SPECT systems. The back-projection based reconstruction is mathematically described in:

\[ f(x, y) = \int_0^\pi g(s, \theta) d(\theta) \]  

(2)

Where, \( f(x,y) \) is the reconstructed voxels by integrating the projection pixels \( g(s, \theta) \) after filtering the sinogram rowise for all view angles. \( s \) and \( \theta \) are detector location and view angle respectively.
The main iterative algorithms are Algebraic Reconstruction Technique (ART), Maximum Likelihood Expectation Maximization (MLEM), Ordered Subset Expectation Maximization (OSEM) and Maximum A Posteriori (MAP). These methods iteratively estimate the value $f(x, y)$ of radionuclide distribution. These iterative methods start their estimation from initialized values. Normally, all voxel values are initialized to 1.

In ART, the value of $f$ at $(x, y)$ is updated at every iteration based on the recursive equation [1]:

$$ f_{j}^{k+1} = f_{j}^{k} + \frac{g_{i} - \sum_{j=1}^{N} f_{ji}^{k}}{N} $$

(3)

$f_{j}$ is the current value and $f_{j}^{k+1}$ is the new estimate. $g_{i}$ is the value of the projection value due to the $i^{th}$ ray. Moreover, $N$ is the total number of voxels that contribute to the projection pixel $g_{i}$. The new estimate, $f_{j}^{k+1}$ will be the sum of previous value and correction term. The correction term is the normalized error between the actual projection and the projection calculated from the estimated distribution.

MLEM and MAP are based on probabilistic machine learning models. The parametric model in equation 1 is reformulated into a probabilistic model for MAP and MLEM, as in the following equation:

$$ g(f) \sim \text{Normal}(Af, \sigma) $$

Equation 4 infers that the projection for a given reconstruction will be normally distributed with a mean of $Af$ and a standard deviation of $\sigma$. The variables $A$ and $\sigma$ are the parameters of the system. In SPECT reconstruction, instead of calculating the system matrix, we have programmed a forward projection model by using ray tracing.

MLEM algorithm aims to estimate a reconstructed object that can generate a likelihood projection to the actual projection [3]. This algorithm is implemented by approximating the distribution type of the radiation source as a Poisson’s distribution. MLEM is composed of two steps: the expectation step and the maximization step. In the expectation step, the algorithm calculates any reconstructed image that provides a similar projection. The estimation of the reconstructed image with a good likelihood is calculated in the maximization step. The algorithm is given in:

$$ f_{j}^{k+1} = \frac{f_{j}^{k}}{\sum_{i=1}^{n} A_{ij} \sum_{j=1}^{m} a_{ij}^{K} f_{j}^{k}} a_{ij} $$

(5)

OSEM is similar to MLEM. In this method, projections are grouped into ordered subsets and are executed the same way as in MLEM [4]. This algorithm is introduced to improve the convergence of the estimation.

MAP tries to find the likelihood of the reconstructed image that gives the same projection whilst trying to reduce the noise in the reconstructed image [5]. The additional feature of noise reduction is introduced in MAP by adding a regularization term $R$, as in:

$$ f_{j}^{k+1} = f_{j}^{k} + \frac{g_{i}}{\sum_{i=1}^{n} a_{ij} \sum_{j=1}^{m} a_{ij}^{K} f_{j}^{k}} a_{ij} $$

(6)

The regularization term $R$ is proportional to the derivative of the energy function, as calculated in:

$$ R = \frac{\partial}{\partial f_{j}} U(f^{k}) $$

where $U$ is the energy of the system. In the above methods, FBP is faster; however, due to smoothing, filter resolution will be wrong. Whereas, MLEM helps to calculate the estimate of the scanned object iteratively. However, as the number of iterations increases, the noise in the reconstructed phantom rises. The estimated noisy reconstructions at different iterations can also provide likelihood projections. Moreover, the speed of convergence is also lower in MLEM. The speed can be improved using OSEM, with the same drawbacks of MLEM. MAP introduces to include the likelihood of the radiotracer distribution for a given projection as well as to reduce noise.

These iterative reconstruction methods all estimate a single reconstructed phantom that can generate a likelihood of the projection to the actual one. This implies that voxel values are deterministic and that the above methods don’t consider any uncertainty in the reconstruction.

Some works applied Bayesian based uncertainty analysis in medical imaging for activity and attenuation estimation. In the work done by Leynes et al. [6], they used Bayesian deep learning to estimate the uncertainty while generating a pseudo-CT from MRI data. This method is not used for SPECT image reconstruction Fui et al. [7] tried to estimate the uncertainty in SPECT reconstruction by using the Fisher Information Matrix (FIM). The FIM is used to calculate the uncertainty in terms of the variance of the reconstructed voxel values analytically. Moreover, this work calculates uncertainties in the reconstruction by using the MAP estimation in terms of the FIM matrix.

The probabilistic programming concept can provide multiple values of each reconstructed voxel, which gives likelihood results of the projection with a specific distribution. This means that probabilistic programming considers the uncertainty in the reconstruction based on a particular distribution. The different reconstructed images within a certain uncertainty level can provide the likelihood of the actual position. The uncertainty calculation makes it possible to represent the reconstructed voxel value as a distribution instead of a fixed value. Normally, the reconstructed image is represented as $R = r_{1}, r_{2}, r_{3}, ..., r_{M}$, where $M$ is the total number of voxels in the reconstructed phantom. The probabilistic programming identifies the uncertainty in the reconstruction as a normal distribution of each with means $\mu_{1}, \mu_{2}, ... \mu_{M}$ and variances $\sigma_{1}, \sigma_{2}, ..., \sigma_{M}$. Therefore, the reconstruction by using probabilistic programming can be represented as:
\[ R_p = \{ \text{Normal}(\mu_1, \sigma_1), \text{Normal}(\mu_2, \sigma_2), \ldots, \text{Normal}(\mu_m, \sigma_m) \} \]

This work focuses on generating uncertainties in the reconstruction for parallel hole SPECT by using the probabilistic programming concept. The details about the probabilistic programming reconstruction (PPR) algorithm are presented in section 2 with the technical details of Markov Chain Monte Carlo (MCMC) sampling and the forward projection implementation. The experiments and results are discussed in Section 3. Finally, conclusions are in section 4.

2. Probabilistic Programming for SPECT

Reconstruction

Initially, the patterns of reconstructed images is defined by using probability distributions. The probabilistic programming is based on the Bayesian inference [8]. The posterior value from the prior and likelihood of observations is calculated by:

\[ P(f|g) = \frac{P(g|f)P(f)}{P(g)} \quad (8) \]

where \( P(f|g) \) is the posterior distribution, \( P(f) \) is the prior distribution, and \( P(g|f) \) is the observation. In terms of image reconstruction, the posterior distribution is the conditional probability of the reconstruction for a given projection. Moreover, the observation is the conditional probability of the projection for a given reconstruction. The prior will be the probability of the reconstruction and \( P(g) \) is the probability of the projection. Therefore, the conditional probability of the voxel values given their projection values will be proportional to the product of the likelihood value of the projection for a given reconstruction and the prior probability of reconstruction. The posterior value of the reconstruction is updated for every iteration. The probability of the projection is the integral of the joint probability of the projection and the reconstruction concerning the change in the projection probability. However, the calculation of \( P(g) \) becomes computationally complex as the dimension of the reconstruction is increased. \( P(g) \) is calculated as in:

\[ P(g) = \int P(f, g)df \quad (9) \]

This means that the \( P(g) \) is the integral of the joint probability \( P(f, g) \) with respect to all possible values reconstruction. If the number of voxel values increases, then the complexity of integration increases.

2.1 Approximate bayesian inference

The computational complexity to calculate \( P(g) \) is overcome by using approximate Bayesian inference. In this method, it is possible to sample the posterior distribution of reconstruction with respect to the projection by using joint probability. Therefore, it is not necessary to calculate the solution for the posterior probability by using a closed-form Bayesian method. Then the Bayesian equation for reconstruction will be updated as:

\[ P(f|g) \propto P(g|f)P(f) \quad (10) \]

This posterior probability is generated by using MCMC sampling. These samplers output different posterior values of reconstruction with certain distributions that give likelihood values of projection. MCMC aims to ensure that the sample we get at the end will be similar to independent samples from the exact posterior distribution. There are different types of MCMC sampling available. In these algorithms, the transition from the current sample is by proposing a new sample using some tuning parameters. The newly proposed sample will be considered if the proposal is accepted with known probability, otherwise it will be rejected. Then the next sample will be the same as the previous sample. In this work, we used the No-U-Turn Sampler (NUTS).

2.2 NUTS Sampler

The NUTS sampling is an advanced version of the Hamiltonian Monte Carlo Sampling (HMC). The HMC is an MCMC sampling method that was developed to avoid random walk behaviour in other sampling methods [9]. In HMC, model samples are generated by simulating the behaviour of a physical system. In this system, particles in multidimensional space are subjected to potential and kinetic energies. So, the energies of these particles at specific energy levels can be defined by using potential and kinetic energies, or in terms of position and momentum. The Hamiltonian term is the sum of the potential and kinetic energies. The Leap-Frog algorithm is used to find the position and momentum of the next sample during the HMC sampling. In this algorithm, the important step is to calculate the gradient of the potential energy, which is with respect to the position of the particle. The chain rule is applied to calculate the gradient. It is possible to use forward differentiation and reverse differentiation to calculate the derivatives by using the chain rule. However, in the reconstruction context, the number of parameters is very high, so even though the reverse differentiation is fast, it takes a vast amount of memory [10]. The chain rule for reverse mode automatic differentiation is performed using:

\[
\frac{\partial y}{\partial x} = \frac{\partial y}{\partial w_1} \frac{\partial w_1}{\partial x} = \left( \frac{\partial y}{\partial w_2} \frac{\partial w_2}{\partial w_1} \right) \frac{\partial w_1}{\partial x} = \left( \frac{\partial y}{\partial w_3} \frac{\partial w_3}{\partial w_2} \frac{\partial w_2}{\partial w_1} \right) \frac{\partial w_1}{\partial x} = \ldots \quad (11)
\]

In the reverse mode differentiation, the derivative of the dependent variable \( y \) is calculated with respect to its subexpressions \( w_1, w_2, \ldots \) etc. This derivative calculation helps the HMC to converge quickly. However, the HMC mainly depends on two parameters, namely the step size and the discretization time. If the step size is too small, the HMC starts exhibiting random walk behaviour, and if it is too high, it takes too long to reach convergence. So, the HMC requires some
tuning. However, the NUTS uses a recursive algorithm to generate a set of likely sample points with a given distribution. Moreover, the sampler stops automatically when it starts to double back and retrace its steps [11]. The NUTS is implemented with algorithms to autotune the step size and the discretization time during sampling.

Apart from this, all samples are drawn with a specific distribution. So, prior knowledge about the system is helpful to find the best reconstructions with the highest likelihood. If the selected distribution is normal, then specifying the mean and variance of the distribution is essential. In the model-based reconstruction, it is required to specify the observations. In medical imaging reconstruction, projections are the observations. These observations can be specified with normal distribution with mean value as the projection, calculated by using the prior reconstructions and a standard deviation as a constant value.

2.3 Algorithm for Probabilistic Programming based Reconstruction

The pseudo algorithm for probabilistic programming based reconstruction consists of the following steps:

1. Define the probabilistic model
   a. Assign distribution to voxel values
   b. Find the forward projection based on the voxels’s values with the assigned distribution.
   c. Assign distribution to the projection by setting forward projection value from step b as mean and a fixed variance.
2. Sampling using a specified sampling algorithm.

The prior of the reconstruction can be a distribution of any type. The one necessary condition is that the values of the prior should be greater than zero. Similarly, the distribution of the observation, that is of the projection, is assigned based on the forward projection generated with the prior values of the reconstruction. The forward projection algorithm is explained in section 2.4. This model is used for sampling with the HMC algorithm to find the posterior distribution through different iterations.

2.4 Forward Projection

The algorithm for forward projection is implemented based on the fast voxel ray traversal algorithm, tracking a ray through a cube. It outputs the co-ordinate of each voxel through which the ray has passed, and the entry and exit time in that voxel. Using the entry and exit time, we can calculate the intersection length inside that voxel. For the forward projection, first, the centre of the detector pixel is calculated and a vertical ray towards the cube of interest is generated. Then, the ray is traced through the cube, and each voxel intersection is identified [12-13]. Then, the detector pixel location, from where the ray is generated, is updated with the sum of the intersected voxels. Repeating this step for all detector pixels. After that, the detector base is rotated one step angle and the above procedure repeated. These projections for each view angle are vectorized and concatenated into a single matrix. This concatenated projection matrix is used for the reconstruction.

2.5 Julia programming and Turing toolbox

The probabilistic programming based SPECT image reconstruction is implemented in the Julia programming platform with the help of the Turing Toolbox package [14]. This software is a high-performance computing language with similar performance as C programming and similar ease of use as the Python language. Moreover, the LLVM compiler used in this software package makes programs run faster compared to other languages. The Turing toolbox was originally developed for probabilistic programming and used for uncertainty generation. This package contains functions for MLEM and MAP and variational inference also. The Turing toolbox has built-in functions for different MCMC samplers. In this work, we use the NUTS sampler only. The output after the sampling contains values of each reconstructed voxel at different sampling, with the mean of these reconstruction samples.

2.6 Distributed Programming

The time to reconstruct by using probabilistic method goes higher when the number of voxels in the phantom is higher. Moreover, when the number of iteration increases, it is possible to find more probable reconstruction that gives more likelihood projection. The distributed programming capability of Julia programming language is incorporated into probabilistic programming to speed up the whole process. The Distributed package in Julia to implement this facility for this work. This modification speed up the whole process with a higher number of iterations.

3. Results

This work focuses on evaluating the probabilistic programming concept in SPECT image reconstruction. Further, uncertainty analysis based on probabilistic programming concept was studied. The experiments were conducted using NUTS sampler with reverse differentiation backend and scanned objects were analyzed using 4x4x4, 8x8x8 and 16x16x16 to measure the point source response. Moreover, uncertainty in the reconstruction of these point sources were inspected in terms of variance. Finally, for studying the real case scenario, Shepp Logan phantom was scanned and reconstructed by applying the procedure discussed above.
3.1 Reconstruction for different cube sizes

The quality of probabilistic programming based reconstruction algorithm was analyzed with phantoms of voxel sizes 4x4x4, 8x8x8, and 16x16x16. The forward projections were acquired based on algorithm mentioned in section 2.4.

3.2 Point Source Reconstruction

![Sampling values of central voxel of phantom with 8x8x8 voxels at different step angles](image1)

**Figure 1: Sampling values of central voxel of phantom with 8x8x8 voxels at different step angles**

Point source reconstruction was performed using a phantom with the central voxel having a certain non-zero value and other voxels having value zero. Later, the projection images for each step angle were obtained on rotating the detector around the phantom. The scanned object was then reconstructed probabilistically after many iterations. The final reconstructed phantom was obtained by calculating the mean of all uncertainties using probabilistic programming. The uncertainties pertaining to the central voxel after reconstruction is shown in figure below: These uncertainties in reconstruction can generate projections with values similar to actual scanned values. The sampling values at different iterations for step angles- 2.0 degree, 5.0 degree and 10.0 degree are drawn in blue, red and green colours respectively. The pink represents the actual phantom value. Figure 1 illustrates that the sampling values reach closer to the actual value when the number of projections increases.

3.2.1 Histogram of Sampling: The histograms can provide information about the number of data points that fall within the range of values of each voxel, after probabilistic programming based uncertainty analysis. The histogram is plotted using samples of each voxel, at the end of a specified number of iterations. Moreover, this histogram also provides information about the distribution of uncertainties in reconstruction. The mean value of distribution of uncertainty is obtained at the peak of histogram. The histograms of reconstructions with different step angles are shown in figure 2. It is evident from figure 2 that the histogram comes close to the actual value if the step angle is small.

3.2.2 Point Spread Analysis: Point spread analysis is the impulse response analysis of an optical system. For pointed spread analysis, the central voxel value of the phantom is usually set to a value higher than zero. Here, for the purpose of this analysis, the central voxel value and other voxel values were taken as 10 and zero respectively. The projection of the phantom was generated for each view angle using ray tracing, and it was used for the reconstruction. The mean reconstructed image calculated from all samples were used for point spread analysis. The central slice of the mean reconstructed image is shown in figure 3.
3.2.3 Full Width at Half Maximum (FWHM) Analysis:

![Figure 4: Gaussian fitted point source response of reconstructed image with 32x32x32 voxels](image)

**Figure 4:** Gaussian fitted point source response of reconstructed image with 32x32x32 voxels

**Table 1: FWHM of reconstructed objects with 32x32x32 voxels by using projections with different step angles**

| Step Angle (degree) | 2     | 5     | 10    |
|---------------------|-------|-------|-------|
| FWHM (mm)           | 0.2581| 0.2582| 0.2582|

FWHM is used to calculate the spatial resolution of an optical system. The point source reconstructed image was used for FWHM analysis. Initially, the mean reconstructed image is calculated using the samples generated from probabilistic reconstruction and the central row of the central slice was extracted from the mean reconstructed image plotted against row location. Further, this central row plot was fitted to a gaussian curve.

![Figure 5: Voxel value comparison by using different types of reconstruction algorithm](image)

**Figure 5:** Voxel value comparison by using different types of reconstruction algorithm

The fitted Gaussian curve is shown in figure 4. The width of this gaussian fitted curve at the half of maximum gives the FWHM value. The FWHM of point source response at different iterations is given in Table 1.

3.3 Comparison between MLEM and MAP

MLEM and MAP are the two main iterative reconstruction techniques. These algorithms generate optimized values of the reconstructed phantom. However, these algorithms do not provide any uncertainty information. The mean reconstructed image generated using probabilistic programming was compared with the reconstructed images using MLEM and MAP. The comparison analysis with MLEM and MAP was conducted both visually and analytically. The reconstruction images in figure 5 show how close the probabilistic reconstruction is with conventional reconstruction methods. Relative norm was used for the numerical analysis of reconstruction quality. The equation 11 explains the method for the calculation of relative norm, $R_{\text{norm}}$.

$$R_{\text{norm}} = \sqrt{(x_1 - x_1')^2 + (x_2 - x_2')^2 + \ldots + (x_n - x_n')^2}$$  \hspace{1cm} (11)

Where, $x_1, x_2, x_3 \ldots x_n$ are the actual phantom voxel values and $\{x_1', x_2', x_3', \ldots, x_n'\}$ are the reconstructed phantom values.

**Table 2: Relative norm of reconstructions by applying different algorithms**

| Number of Voxels | Relative Norm | MAP   | MLEM | PPR  |
|------------------|---------------|-------|------|------|
| 4x4x4            | 0.00557       | 0.00557| 0.009|
| 8x8x8            | 0.0304        | 0.0304| 0.01215|
| 16x16x16         | 0.0827        | 0.155 | 0.025607|

The relative norms of reconstruction using probabilistic and conventional algorithms including MAP and MLEM were discussed in Table 2. This table shows that the norm of mean reconstructed image is in reasonable level with the norm of reconstructed image obtained using techniques like MLEM and MAP.

3.4 Uncertainty Analysis

Different methods were employed for analysing the uncertainty in reconstruction. Initially, the variance of each voxel was calculated from the samples obtained using probabilistic programming. The variance has provided information about the uncertainty in reconstruction. If the variances of voxels are very low, then the values obtained from forward projection model used for reconstruction is close to the actual model and the reconstruction values will be close to reality. If the variance of voxels are too high, then the forward projection model used in reconstruction will be inaccurate. Moreover, the chance of error in the reconstruction values can be higher.

3.4.1 Variance of each voxels at different iterations: In this work, the effect of variance between samples of each
voxel were studied and analysed for different step angles. No significant effect of view angles in the variance was found when the number of voxels was low. However, the effect of step angle in variance, calculated from the samples generated using probabilistic reconstruction, was noticeable. It was found to decrease for small step angles. The variance of central voxels at different number of iterations are shown in table 3.

Table 3: Variance of central voxel after reconstructed by using probabilistic programming

| Step Angle (in degree) | 2.0 | 5.0 | 10.0 |
|------------------------|-----|-----|------|
| Number of Voxels       |     |     |      |
| 4x4x4                  | 0.0225 | 0.0225 | 0.0225 |
| 8x8x8                  | 0.0162 | 0.0252 | 0.0325 |
| 16x16x16               | 0.0176 | 0.0265 | 0.0321 |

3.4.2 Mean of Variances: The mean of variance calculated from probabilistic programming based reconstruction, also helps, in numerical analysis of uncertainty in reconstruction.

Table 4: Mean of variances of phantoms of different sizes scanned at different view angles

| Step Angle (in degree) | 2.0 | 5.0 | 10.0 |
|------------------------|-----|-----|------|
| Number of Voxels       |     |     |      |
| 4x4x4                  | 0.008 | 0.008 | 0.008 |
| 8x8x8                  | 0.004 | 0.006 | 0.009 |
| 16x16x16               | 0.003 | 0.005 | 0.006 |

Initially, the variances in reconstruction were generated by using the method described in section 3.4.1. After that, the mean of variances of all voxel values are calculated by using simple statistical methods. The mean of variances corresponding to different specifications are discussed in Table 4. When the number of voxels were low, there were no significant effect of view angles in the mean of variance. However, the effect of step angle, in mean of variance calculated from the samples generated using probabilistic reconstruction, was noticeable. It was found to decrease for small step angles.

3.4.3 Histogram of variances: This histogram of variances is also used to visualize uncertainty in reconstruction. The variance of each sample was calculated by using simple statistical techniques. Then the histogram of voxel’s variance was plotted. Figure 6 shows the histogram for different SPECT settings. This histogram clearly indicates that the variance approaches the value zero, when the step angle approaches the zero value. This shows that the variance reduces whenever the step angle reduces.

3.5 Shepp Logan phantom reconstruction

Shepp Logan phantom is a standard phantom used for medical image reconstruction studies. This phantom was generated using MATLAB software and the output exported to a data file. Then, these phantoms were then imported into Julia program and used to generate projections. The projection of the Shepp Logan phantom was calculated using the algorithm discussed in section 2.4. Further, these projections of Shepp Logan phantoms were used as input for probabilistic reconstruction and analysis. The samples were generated using probabilistic reconstruction and they represent the uncertainties in reconstruction. The mean reconstructed image obtained as a result of probabilistic reconstruction was used.
for visual comparison with original phantom. The central slice of original phantom and probabilistically reconstructed phantom are shown in figure 7.

4. Conclusion

There are different reconstruction methods used for reconstruction in medical imaging. These reconstruction methods include analytical and iterative methods. The objects reconstructed using these methods gives us an idea of radiation distribution inside the scanned body. However, these reconstruction algorithms do not consider the uncertainty involved in the reconstruction. The probabilistic programming based reconstruction is employed in this work. Here, the uncertainty in reconstruction is calculated using Approximate Bayesian concept, which uses MCMC based NUTS sampler. This sampler generates multiple samples of reconstructed objects within the specified distribution. Therefore, probabilistic methods make it possible to represent reconstructed voxel with specific distribution and its parameters instead of a simple value. In this work, the uncertainty of the voxel is defined as a normal distribution with a mean and variance. The mean and variance for the distribution were calculated from samples generated using a NUTS sampler. The accuracies of the forward projection model and the reconstructed phantom were calculated using uncertainty analysis. However, the probabilistic programming based reconstruction is a time-consuming process if the number of voxels is higher in the reconstructed object. The problem with time consumption can be improved by using distributed programming, or by optimizing the sampling algorithms.

References

[1] Bruyant, P. P. (2002). Analytic and iterative reconstruction algorithms in SPECT. *Journal of Nuclear Medicine, 43*(10), 1343-1358.
[2] Tsui, B. M. W., & Frey, E. C. (2006). Analytic Image Reconstruction Methods in Emission Computed Tomography. In H. Zaidi (Ed.), *Quantitative Analysis in Nuclear Medicine Imaging* (pp. 82-106). Boston, MA: Springer US.
[3] Shepp, L. A., & Vardi, Y. (1982). Maximum likelihood reconstruction for emission tomography. *IEEE Trans Med Imaging, 1*(2), 113-122.
[4] Hudson, H. M., & Larkin, R. S. (1994). Accelerated Image Reconstruction Using Ordered Subsets of Projection Data. *IEEE Transactions on Medical Imaging, 13*(4), 601-609.
[5] Green, P. J. (1990). Bayesian Reconstructions from Emission Tomography Data Using a Modified Em Algorithm. *IEEE Transactions on Medical Imaging, 9*(1), 84-93.
[6] Leynes, A. P., Ahn, S. P., Wangerin, K. A., Kaushik, S. S., Wiesinger, F., Hope, T. A., & Larson, P. E. (2020). Bayesian deep learning Uncertainty estimation and pseudo-CT prior for robust Maximum Likelihood estimation of Activity and Attenuation (UpCT-MLAA) in the presence of metal implants for simultaneous PET/MRI in the pelvis. *arXiv preprint arXiv:2001.03414*.
[7] Fuin, N., Pedemonte, S., Arridge, S., Ourselin, S., & Hutton, B. F. (2014). Efficient Determination of the Uncertainty for the Optimization of SPECT System Design: A Subsampled Fisher Information Matrix. *IEEE Transactions on Medical Imaging, 33*(3), 618-635.
[8] Gamerman, D., & Lopes, H. F. (2006). Markov chain Monte Carlo: stochastic simulation for Bayesian inference: CRC Press.
[9] Neal, R. M. (2011). MCMC Using Hamiltonian Dynamics. *Handbook of Markov Chain Monte Carlo, 113-162*.
[10] Christianson, D. B., Davies, A. J., Dixon, L. C. W., Roy, R., & VanderZee, P. (1997). Giving reverse differentiation a helping hand. *Optimization Methods & Software, 8*(1), 53-67.
[11] Hoffman, M. D., & Gelman, A. (2014). The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research, 15*, 1593-1623.
[12] Amanatides, J., & Woo, A. (1987). A fast voxel traversal algorithm for ray tracing. Paper presented at the Eurographics.
[13] Siddon, R. L. (1985). Fast calculation of the exact radiological path for a three-dimensional CT array. *Medical Physics, 12*(2), 252-255.
[14] Ge, H., Xu, K., & Gharahmani, Z. (2018). Turing: a language for flexible probabilistic inference. *International Conference on Artificial Intelligence and Statistics, Vol 84*, 84.