The use of physical optics and machine learning in modelling the QUBIC beam pattern.

A.B. Flood[1,2], C. O'Sullivan[1], M.L. Gradziel[1], D. Gayer[1], S Marwede[1], J-CH. Hamilton[3], S. Torchinsky[3,6], T. Laclavere[3], L. Kardum[3], M. Regnier[3], E.S. Battistelli[5], P. de Bernardis[5], B. Costanza[7], S. Ferazzoli[5], M. Gervasi, S. Masi[5], N. Mirón-Granese[7], L. Mousset[9], C. Scóccola[7,8], M. Zannoni[4]

[1]Physics Department, Maynooth University, Maynooth Co. Kildare, Ireland;
[2]Hamilton Institute, Maynooth University, Maynooth Co. Kildare, Ireland;
[3]Université Paris Cité, CNRS, Astroparticule et Cosmologie, Paris, F-75006 Paris, France;
[4]Physics Department "G. Occhialini" University of Milano Bicocca, Italy;
[5]Physics Department, Sapienza University of Rome, Italy;
[6]Université PSL, Observatoire de Paris, Astroparticule et Cosmologie, Paris, France;
[7]Facultad de Ciencias Astronómicas y Geofísicas, Universidad Nacional de La Plata, Argentina; [8]Departamento de Física, FCFM, Universidad de Chile, Santiago, Chile;
[9]Laboratoire de Physique de l'École Normale Supérieure, ENS, Université PSL, CNRS, Sorbonne Université, Université de Paris, Paris 75005, France;

ABSTRACT

The Q and U Bolometric Interferometer for Cosmology (QUBIC) is a ground-based telescope which will observe the cosmic microwave background (CMB) with the goal of detecting its extremely faint primordial B-mode polarization pattern as observational evidence for the theory of inflation in the early Universe.

QUBIC has a novel bolometric interferometer design that allows it to have precise control over instrument systematics and to remove astronomical foregrounds which would otherwise obscure the CMB polarization signal. Due to its interferometric design, QUBIC has a complex multi-peaked antenna beam pattern known as the synthesized beam which must be well known in order to correctly process the QUBIC data. We have modelled the QUBIC beam pattern at multiple frequencies and for detectors located at different positions in the focal plane using precise electromagnetic and physical optics simulations. For the complex setup of the QUBIC telescope these simulations are computationally expensive.

In this work we use machine learning to interpolate the beam at multiple frequencies and detector locations in order to reduce the amount of computation required to model it. For our purposes, two types of neural networks were used: a Multilayer Perceptron (MLP) model and a Long Short-Term Memory (LSTM) model. For the amount of data generated, only the LSTM model was successful in replicating the synthesized beam to a satisfactory degree. We used the LSTM machine learning (ML) model to generate the synthesized beam for all 248 detectors at 150 GHz and compared its use in our analysis pipeline to a fully simulated PO beam and an approximate analytical model. From this it can be seen that although the ML prediction did not replicate the PO synthesized beam perfectly, it was closer than the beam produced with the analytical formula. Finally, we show the effect of the differences in beam pattern prediction on the recovered B-mode spectrum of the CMB.

Keywords: Physical Optics, Beam Pattern, Cosmic Microwave Background, Bolometric Interferometer, Machine Learning

UV/Optical/IR Space Telescopes and Instruments: Innovative Technologies and Concepts XII,
edited by Jonathan W. Arenberg, H. Philip Stahl, Proc. of SPIE Vol. 13623, 136230J
2025 Published by SPIE · 0277-786X · doi: 10.1117/12.3064328

1. INTRODUCTION

The Q and U Bolometric Interferometer for Cosmology (QUBIC) is a ground-based telescope which will observe the cosmic microwave background (CMB) with the goal of detecting its extremely faint primordial B-mode polarization pattern as observational evidence for the theory of inflation[1]. A QUBIC technical demonstrator (TD) is currently undergoing commissioning at the observing site near Salta in Argentina.

QUBIC has a novel bolometric interferometer design that allows it to have precise control over instrument systematics and to remove astronomical foregrounds which would otherwise obscure the CMB polarization signal. For QUBIC, the re-emitted beams from the aperture feedhorns are superimposed by a cold reflective optical combiner onto the focal plane. The first observations will be made in a frequency band centered on 150 GHz. Due to its interferometric design, QUBIC has a complex multi-peaked antenna beam pattern known as the synthesized beam [2]. This must be well known in order to correctly process the data that QUBIC collects. We have modelled the QUBIC beam pattern at multiple frequencies, including optical aberrations, using precise electromagnetic and physical optics (PO) simulations.

PO simulations are computationally expensive, especially for the QUBIC TD instrument which has 248 detectors and 64 horns (with more planned in a future upgrade to the full instrument). This means that simulations can take days of computation time to produce the synthesized beam at a single frequency for each detector, even with the use of a cluster computer and parallel processing. For this reason, the use of machine learning techniques and neural networks are proposed to interpolate the beam at different frequencies and to reduce the number of PO simulations required.

Another reason to use neural networks is that once a fully trained model has been created it can be used to generate a beam at any frequency on demand. This is convenient as QUBIC is a worldwide collaboration and instead of sending large amounts of data, a pre-trained machine learning model can be easily shared with collaborators.

Other work with the same goal of reducing computation for generating antenna beam patterns exists within the field of communications, particularly for the development of 5G and 6G mobile networks where ML techniques are often leveraged to reduce the computation involved with simulating and optimizing antenna beam patterns[3]. However, this work differs from such research as it is used to model the antenna beam itself rather than metrics associated with antenna performance such as Signal to Interference plus Noise Ratio as is the case for [4]. Emulators for antenna beam patterns, such as Medea[5], exist, however neural networks are not used in this case.

For our purposes, two types of neural networks are used to interpolate the QUBIC synthesized beam: a Multilayer Perceptron (MLP) model and a Long Short-Term Memory (LSTM) model. Two different parameters, frequency and detector position on the focal plane, are investigated. The results from the interpolations are then compared to actual PO simulations using general metrics such as the structural similarity index measure (SSIM) and mean squared error (MSE), as well as metrics specific to the performance of QUBIC such as peak intensity and peak location. Finally, the outputs of the neural networks are compared to the PO simulations and analytical formula using the Qubicsoft[6] pipeline that we use for data analysis.

2. MODELING THE QUBIC SYNTHESIZED BEAM

PO is a full vector method for determining the propagation of electromagnetic (EM) radiation through an optical system and it can be used to model QUBIC's complex synthesized beam. We use in-house software, MODAL [7], to implement PO. Using MODAL, we can define each element in a system: the source, reflectors, apertures, and an output plane. Then we can propagate radiation in a chain from element to element. Maxwell's equations are used to calculate the equivalent currents induced on reflectors (we assume each is a perfect electrical conductor) and then radiation integrals are used to calculate the electric and magnetic fields at an observation point due to these currents [8]. The PO approximation assumes that the surface current at a particular point on a curved surface is the same as the surface current at a point on an infinite planar surface which is tangential to the curved surface at that point. Any element in the optical system is subdivided into many such tangent surfaces and PO is applied to each. The number of these points depends on the sampling chosen and determines the resolution of the output.

When observing the sky, radiation enters through QUBIC's cryostat window, past filters, a rotating half-wave plate and polarizer, then through a back-to-back horn array. In traditional aperture synthesis interferometry, the correlations of signals from the different horns build up a sampling of fringe visibilities in *uv* (focal) plane and the Fourier transform of

this signal on the *uv* plane is an image of the sky. In QUBIC, however, the radiation from all horns will be simultaneously superimposed on the focal plane[9] using an optical beam combiner. This is equivalent to imaging the sky with a standard imager whose beam is the synthesized beam of the bolometric interferometer i.e. the beam formed by the combination of all interference patterns of all possible pairs of horns in the aperture array [1]. The beam combiner uses a primary and secondary mirror in an off-axis Gregorian setup to focus the horn beams on the focal plane (figure 1). The off-axis nature of this fast optical set-up is expected to introduce optical aberrations into the synthesized beam. The detectors used for QUBIC are transition edge sensors (TES), which can detect low energy photons.

We model the synthesized beam of the instrument by starting at a TES detector location on the focal plane and propagating a beam out through the optics and onto the sky. In practice we model the beam through each individual horn of the 64-element array separately and combine them on the sky. Each TES 'sees' a different part of the sky depending on its location in the focal plane, in effect, the interference pattern from the 64 horns shifts underneath the antenna primary beam envelope (see reference [9] for details).

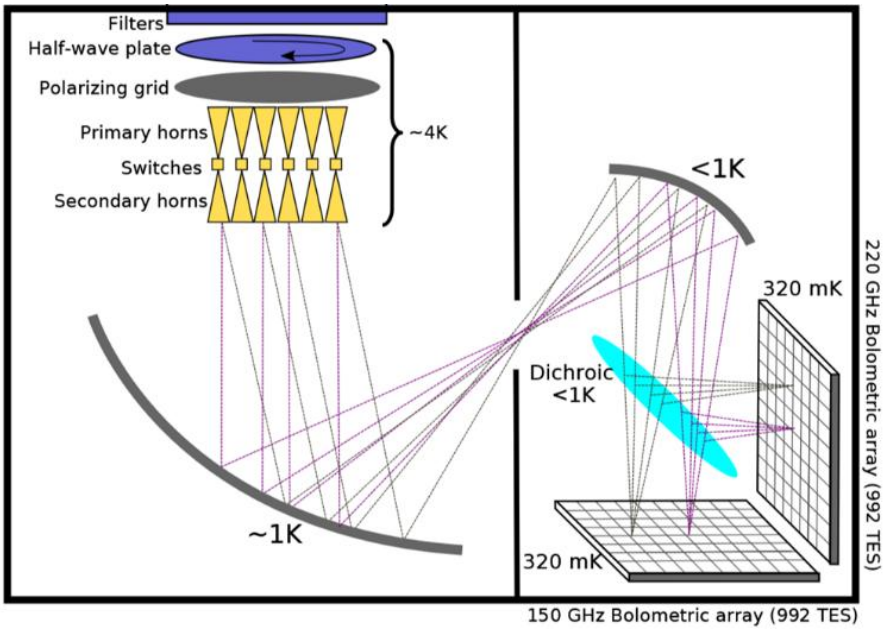

Figure 1. A diagram of the set-up of the QUBIC telescope. Source: [1] (The design of the full instrument is under review, the TD has only one focal plane).

The synthesized beam can also be modelled using an analytical formula that does not take account of optical aberrations. The equation for the intensity of the synthesized beam for a TES location $\vec{r}$ on the focal plane is given by:

$$B_{synth}(\vec{r},\vec{n},\lambda) = F(\lambda)B_{prim}(\vec{n})B_{sec}(\vec{r})\frac{\sin^2\left(n_h\frac{\pi}{\lambda}\Delta h\left(\frac{r_x}{D}-n_x\right)\right)}{\sin^2\left(\frac{\pi}{\lambda}\Delta h\left(\frac{r_x}{D}-n_x\right)\right)}\times\frac{\sin^2\left(n_h\frac{\pi}{\lambda}\Delta h\left(\frac{r_y}{D}-n_y\right)\right)}{\sin^2\left(\frac{\pi}{\lambda}\Delta h\left(\frac{r_y}{D}-n_y\right)\right)} \quad (1)$$

where $\vec{n}$ is the off-axis direction of the point source, $\vec{r}$ is position on the focal plane, λ is the wavelength of the electromagnetic radiation, $F(\lambda)$ describes the shape of the filter, B_{prim} is the intensity of the primary lobe of the antenna pattern, B_{sec} is the intensity of the secondary lobe of the antenna pattern, n_h is the number of horns along one side of the horn array, Δh is the spacing between the horns and D is the focal length of the QUBIC beam combiner ($D \sim 300$ mm). The performance of the ML models will be benchmarked against the analytical (simplest) and PO (most complex) models.

3. DATA AND INTERPOLATION PROBLEMS

3.1 Test Cases

Initially the model was tested on some simpler test cases to investigate whether or not a good model could be developed for interpolating diffraction patterns produced using PO. Since the beam from a central TES detector in QUBIC is expected to be of similar form to the far field pattern of a square array of 8×8 holes in an illuminated aperture, the two initial cases chosen were a pair of slits (Young's slits) and a square array of 2×2 holes in an illuminated aperture. These patterns were produced using MODAL for a range of frequencies from 100 GHz to 300 GHz. For these two test cases, the data consist of the electric field intensity on an output plane in the far field of the aperture. The test cases were sampled at 5 GHz intervals to assess whether ML could interpolate the data in this manner, and to potentially optimize models for the problem where possible.

3.1.1 Young's Slits

A MODAL model of an aperture with two slits was created as shown in figure 2. The model comprises of a gaussian beam at the origin which illuminates an aperture located 50 mm from the source (along the z axis) which has two 2 mm wide slits separated by a distance of 10 mm. The fields are then propagated from the source to the aperture and from the aperture to an output plane located 1 m (along the z axis) from the source. The images generated were of the electric field intensity at the output plane for a given frequency.

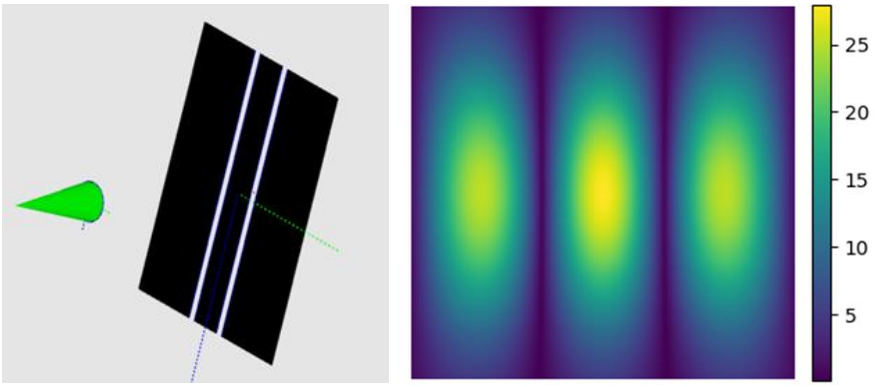

Figure 2. The figure on the left shows a MODAL model of Young's Slits while the figure on the right shows the output plane result at 150 GHz (arbitrary magnitude units).

3.1.2 2 × 2 Square Array

This MODAL model, shown in figure 3, was similar to Young's slits, but the aperture was altered so that it had four holes arranged in a 2 × 2 grid around the centre. The model comprises of a gaussian beam at the origin which illuminates an aperture located 50 mm from the source (along the z axis) which has four holes arranged in a 2 × 2 square array around the centre. These holes were 2 mm wide and 10 mm apart. The fields, one from each hole, are then propagated from the source to the aperture and from the aperture to an output plane located 1 m (along the z axis) from the source. The images generated were the electric field intensity at the output plane for a given frequency.

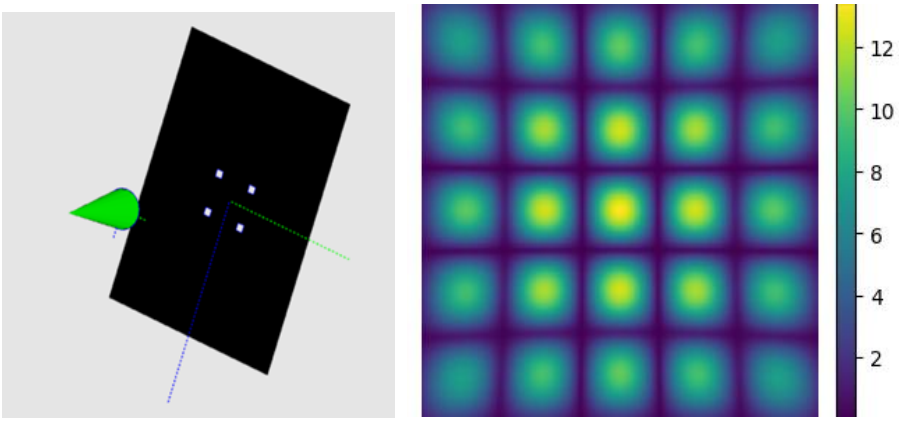

Figure 3. The figure on the left shows a MODAL model of the 2 × 2 grid while the figure on the right shows the output plane result at 250 GHz.

3.2 QUBIC Cases

The QUBIC TD synthesized beam was previously modeled using MODAL for all 248 detectors at 130 GHz, 150 GHz, and 170 GHz. The input data used for the ML model were the synthesized beams formed by propagating the electric field to an output plane in the far field of the instrument (25 m away). These data were in the form of images, 2D arrays of electric field intensity at each point.

The ML model would be used to predict the QUBIC synthesized beam at a particular frequency and for a particular position on the focal plane. The synthesized beam was modelled for a variety of different frequencies and detector positions in order to have sufficient training data and to determine how many frequencies and detector positions need to be provided for training to form a sufficient reconstruction.

In order to use the data from the test cases and QUBIC with an artificial neural network, the data were converted from a 2-dimensional array with values for the electric field intensity at each grid point to a set of data with x and y coordinates, electric field intensity and frequency, so that the problem was reframed as a 4D interpolation problem, with the dependent variable being the electric field intensity. For interpolation based on detector position the model was changed to include two additional independent variables; the x and y spatial positions of the detector on the focal plane.

4. ML MODELS

Within this work, two different ML approaches were taken in order to determine which method was the most effective.

4.1 Feed-Forward Neural Network

Feed-forward neural networks are the most common deep learning models. Their goal is to approximate some function $f(x)$. Neural networks are so called because they are loosely based on neurons in the brain. In feed-forward neural networks the flow of information is in one direction only: from the dependent variables through the connections and layers to the output [10]. Neural networks typically consist of three datasets: training, validation and testing. A training dataset is used to adjust the weights within neural network so that it produces better outputs. A validation dataset is used to get a current metric representing the neural network's accuracy on inputs it hasn't been trained on. A trained neural network is then used on a testing dataset to determine the neural network's perfomance.

Within a neural network there are input nodes which take in the data for the model. Hidden nodes, which make up the hidden layers, pass the weights and data from the input layer (or from another hidden layer) to the next hidden layer (or output layer) and apply the activation function to the data. Nodes within hidden layers are not directly interacted with, only receiving inputs from and sending outputs to other layers. The connections in the network go from node to node and each

have a weight assigned to them. The weights are a numerical value which the data is multiplied by. During the learning process the weights are adjusted to achieve the final output. The activation function defines the output of a node. The output nodes map the data to the desired output shape [11].

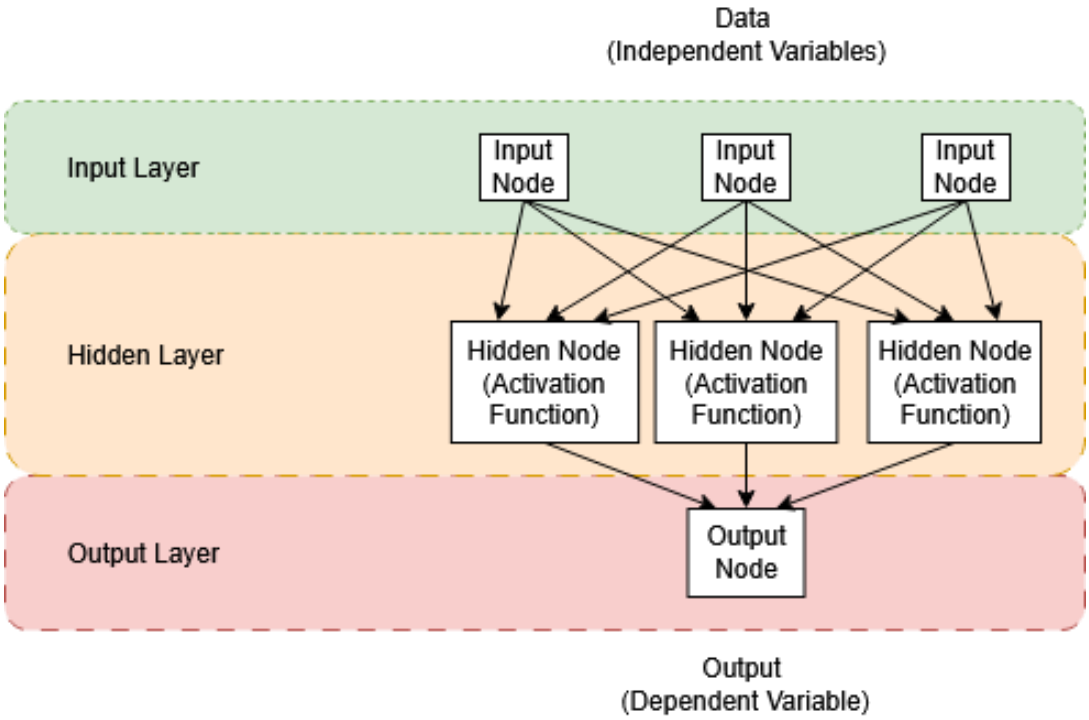

Figure 4. The figure shows information flow through an MLP.

The weights are adjusted through an iterative process to achieve the smallest difference between the output predicted by the model and the actual value. Usually, this difference is represented by a loss or objective function. The goal is to minimise the objective function. Common ways to achieve this are by using gradient descent or stochastic gradient descent. In the gradient descent method, the gradient of the objective function is found in order to determine the direction to move in order to minimise the objective function. The change is then backpropagated through the model to update all the weights [12]. This process repeats over many iterations called epochs.

4.2 Multilayer Perceptron

One of the models used for this work was a multilayer perceptron (MLP); this a type of simple feedforward neural network (figure 4). In this case the ADAM optimizer [13] is used alongside the Keras library in python [14]. The loss function is Mean Squared Error (MSE):

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(y_i - \hat{f}(x_i)\right)^2 \tag{2}$$

where $\hat{f}(x_i)$ is the predicted value that the model gives for the ith observation, while y_i is the actual value, and N is the total number of observations. The activation function used for the model was a Rectified Linear Units (ReLU) activation function, which is defined by the function, $g(z) = max\,(0, z)$. ReLUs are similar to linear units except they are zero across half their domain [10]. Within this work the MLP had 4 dense layers with ReLU activation functions.

4.3 LSTM

Recurrent Neural Networks (RNNs) are a type of neural network where information can persist from one training example to the next through the use of feedback loops, as opposed to feedforward neural networks where information propagates in only one direction [15]. This means that RNNs are particularly useful when previous data points impact the next ones, i.e. sequential data. For this reason, RNNs were identified as being well suited to this problem since it can be thought of as sequential data that evolves with frequency and focal plane position. Long Short Term Memory (LSTM) models are a type of RNN which can learn long term dependencies.

Like a feedforward neural network, RNNs are composed of artificial neurons and weights, where the weights are adjusted to minimise some loss function, typically gradient descent is used to do this. However, the artificial neurons in an RNN will have one or more feedback loops with a hidden state which contains information from previous training examples. A simple RNN has three layers, an input layer, a hidden layer, and an output layer [16]. Figure 5 (a) shows a simple diagram of how information flows through a simple RNN, with feedback loops occurring on the hidden layer. It is convenient to think of the operations performed on the hidden layer as an RNN cell and represent it as such on a diagram as the same operations are repeated for each loop.

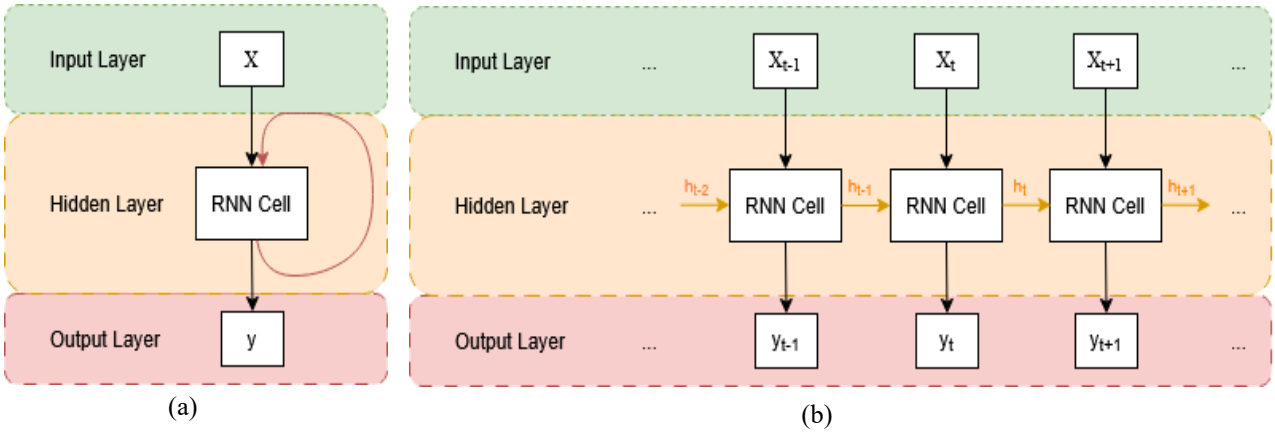

Figure 5. (a) Shows information flow through an RNN. (b) Shows information flow through an unrolled RNN.

In order to compute the result, we can look at our recurrent network as a finite series of steps, this is called truncating and unrolling our network and an example is shown in Figure 5 (b). Truncated unrolled RNN systems are usually trained using Back Propagation Through Time (BPTT), which is a Back Propagation technique for sequences[17]. Back Propagation involves the repeated application of the chain rule of differentiation on the gradients. Back Propagation is applied to recurrent networks by exploiting the fact that for every recurrent network, for a finite number of steps, there exists an equivalent feedforward network with the same behaviour [17].

So, for one timestep, t, of the RNN, the training data, X_t, are input. The input layer is connected to the hidden layer where we have the RNN cell. Typically, inside the RNN cell a tanh() function is applied to the previous hidden state, h_{t-1}, and the input to give the current hidden state, h_t. The current hidden state is used to compute the cell output, y_t. The hidden layer then has two outputs; an updated hidden state to be passed on to the next iteration and the output for these data, both of these outputs are informed by the current data and previous data [18].

But these RNNs have their limitations. Although each hidden layer contains information from the previous hidden layer, as the gap between a previous step and the current step grows, the repeated operations performed can cause a loss of the information from the previous step. This is known as the vanishing gradient problem, and it means that basic RNNs have trouble learning long term dependencies. This problem was addressed with LSTMs, which are capable of handling long term dependencies [19].

LSTMs have a similar overall architecture to basic RNNs, but the operations inside the cell are different. An LSTM cell has a cell state which persists throughout the whole hidden layer and can allow information to flow through undisturbed,

allowing the network to learn long-term dependencies. The cell state can only be changed by gates within the LSTM cell. There are three types of gates to allow the cell state to be altered: forget, update, and output [20].

1. The forget gate is a sigmoid layer that is used to forget the irrelevant information from past timesteps and remove it from the cell state.
2. The update gate selectively updates the cell state value using input data from that timestep.
3. The output gate provides the value of the hidden state, which is obtained by applying some function to the cell state.

LSTMs are one of the most commonly used RNNs, which is why they were chosen for this work. The model used was built using the Keras library [14]. Within this work the LSTM had 2 LSTM layers with 401 LSTM units each, as well as a dense layer with ReLU activation function.

5. EVALUATION

To determine the effectiveness of the models for recreating the synthesized beam for different frequencies and detector positions, they were used to generate a predicted beam intensity at each x and y coordinate of a far field image. The ML predicted image was then directly compared to the PO image simulated in MODAL.

5.1 General Metrics

Two metrics were also used to analyse the differences between the original and generated images, they are MSE and structural similarity index measure (SSIM). In this case the MSE was applied to compare the values between corresponding pixels for the two images. The SSIM is a method for measuring similarity between two images by comparing local patterns of pixel intensities[21]. A low MSE and a high SSIM indicate a better performance.

5.2 QUBIC Specific Metrics

Since the Qubicsoft pipeline doesn't use the full image of a beam and instead uses the position and intensity of the zeroth and first order peaks, the differences between the position and intensity of these peaks were also used as a metric to understand the performance of the ML models.

Additionally, the Qubicsoft pipeline can be used to evaluate the effect of these differences on the final cosmological data analysis. Using Qubicsoft we can observe a simulated sky using the PO beam and obtain the time ordered data QUBIC would have produced. This is then treated like real data in QUBIC's data processing pipeline [22]. However, during data processing, we do not have to use the same beam, as is the case in practise when we do not have an accurate model. Thus, we can see how the observations would be affected if the beam used to observe the sky was different to the beam used in the data reduction pipeline. The effectiveness of ML prediction was compared to that of the analytical formula.

6. RESULTS

6.1 Test Cases

The MLP model was first applied to the simpler test cases to determine whether or not ML could be used to interpolate diffraction patterns produced by PO. Additionally, the effect of using a different number of frequencies to train the model was investigated to estimate how much data would be required for an adequate model.

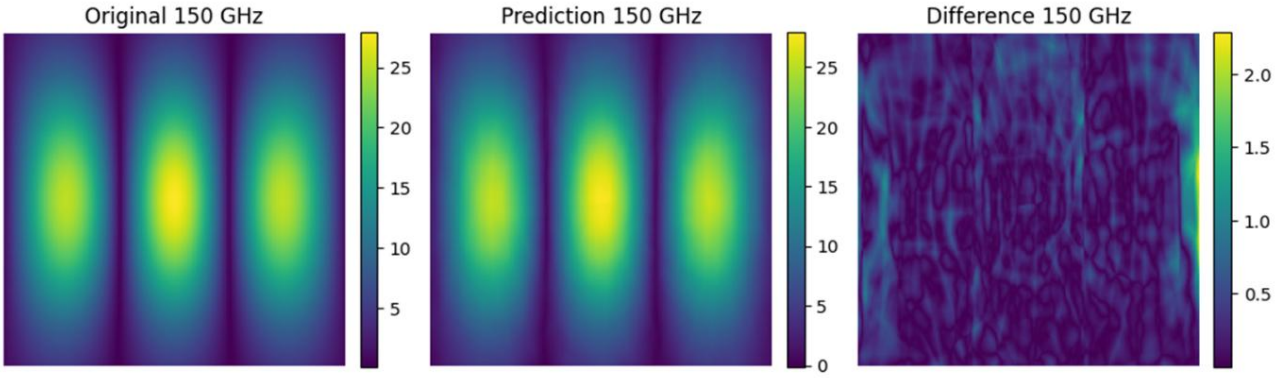

Figure 6. Young's slits model output at 150 GHz.

The first case tested with the model was Young's slits. Initially the model was trained with 21 frequencies with 10 GHz spacing. The model performed well with this amount of training data with only small differences between the prediction and the actual diffraction pattern. Figure 6 shows the output of the MLP for test case 1; Young's Slits, at 150 GHz. This shows that an ML model can replicate the far field diffraction pattern. For the 2 × 2 grid the same tests were performed as for Young's slits. The model appears to perform similarly well on these data; figure 7 shows the output of the MLP for test case 2; 2 × 2 grid, at 150 GHz.

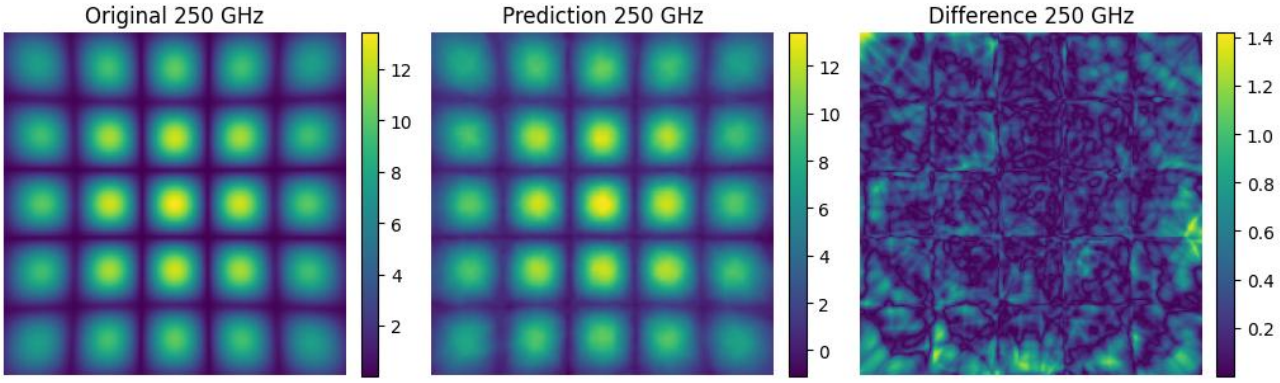

Figure 7. 2 × 2 Grid model output at 250 GHz.

The MLP was then trained using differing amounts of data to get an estimate of how much data would be required to properly train the model. The same range of 100 GHz -300 GHz was used but fewer frequencies were selected, this corresponded to wider gaps between frequencies.

The SSIM and MSE for the differing amounts of data for both test cases are shown in figure 8, where it's clear that the MLP performs similarly for both test cases and it can be seen that the MLP performs best when it is trained on at least 8 different frequencies. Its performance steadily decreases once fewer than 8 frequencies are used for training. A low MSE and an SSIM close to 1 indicate a better performance.

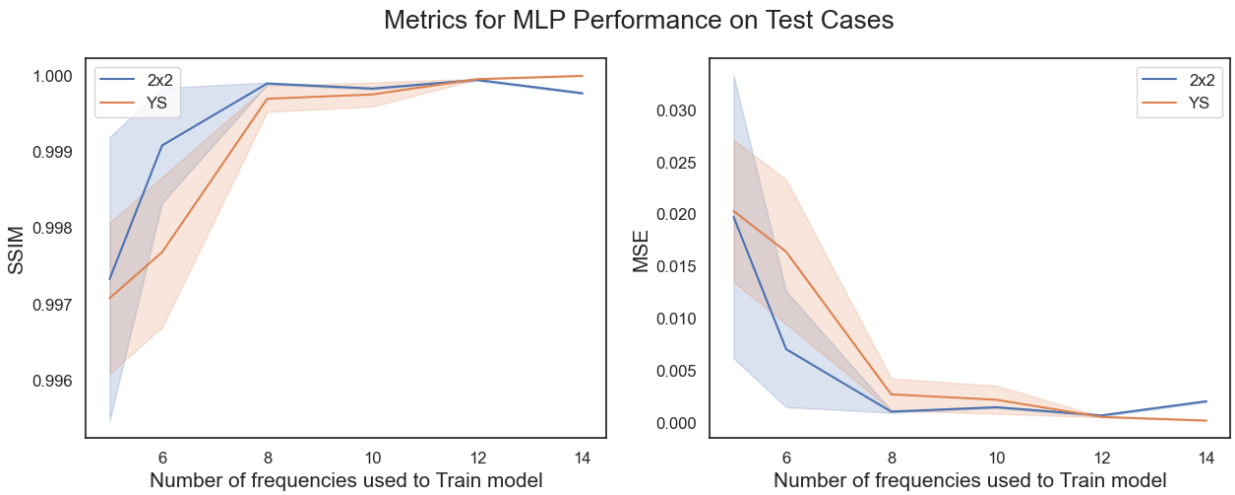

Figure 8. Young's slits model and 2 × 2 grid model SSIM and MSE plots for different amounts of training data.

6.2 Interpolating QUBIC Data

There are three ways of approaching the interpolation of QUBIC's synthesized beam; interpolating the pattern at different detectors but for a single frequency, interpolating the pattern at different frequencies but for a single detector, and interpolating the pattern at both different frequencies and different detectors. The third way would allow for the most flexibility and require the least amount of data to be simulated for its training.

The synthesized beam was first interpolated for different detectors at a single frequency, since these data were the most readily available with 248 detectors having already been simulated at one frequency. The MLP model was adjusted to remove the frequency variable from the input layer and add two more independent variables; the x and y positions of the detector on the focal plane. The differing positions of the detectors change how the synthesized beam appears; the position of the peaks and their intensities will be different at each detector location and in addition, the detectors are affected by optical aberrations which are more prominent for some detector locations than others, particularly those further from the centre of the detector plane. Despite this, the MLP is capable of replicating the beam when enough data are provided for training; an example of the beam reconstruction can be seen in figure 9.

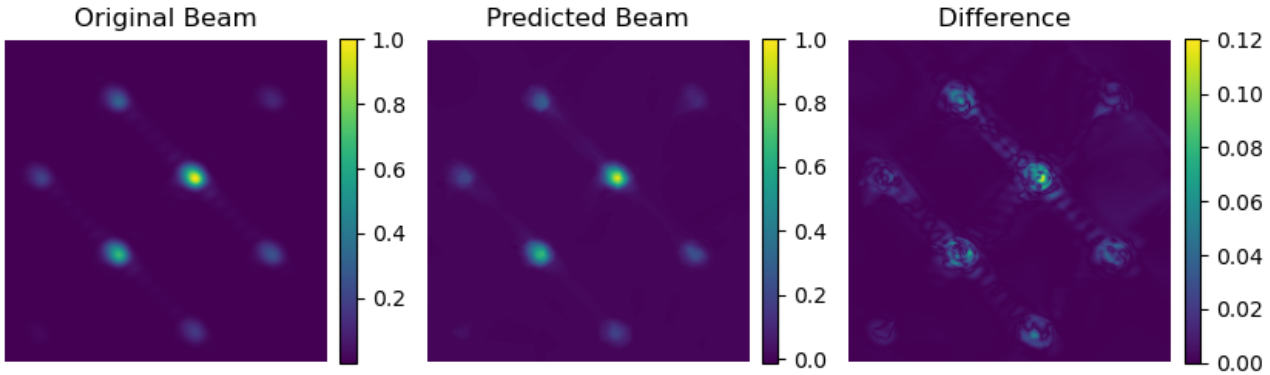

Figure 9. QUBIC detector position interpolation MLP model output trained on 20% of the detectors. (This was for a TES detector at $x = -2.4$ mm, $y = -41.4$ mm from the center of the focal plane.)

In order to determine how much data is necessary for optimal performance, the MLP was trained on several different amounts of data. This was represented as a percentage of the total number of detectors (248). Graphs showing the SSIM. MSE, peak position difference, and peak intensity difference are shown in figure 10. Results in performance are consistent across each metric, showing that at least 20% of the detectors, or approximately 50 detectors, should be used for training to achieve the best performance.

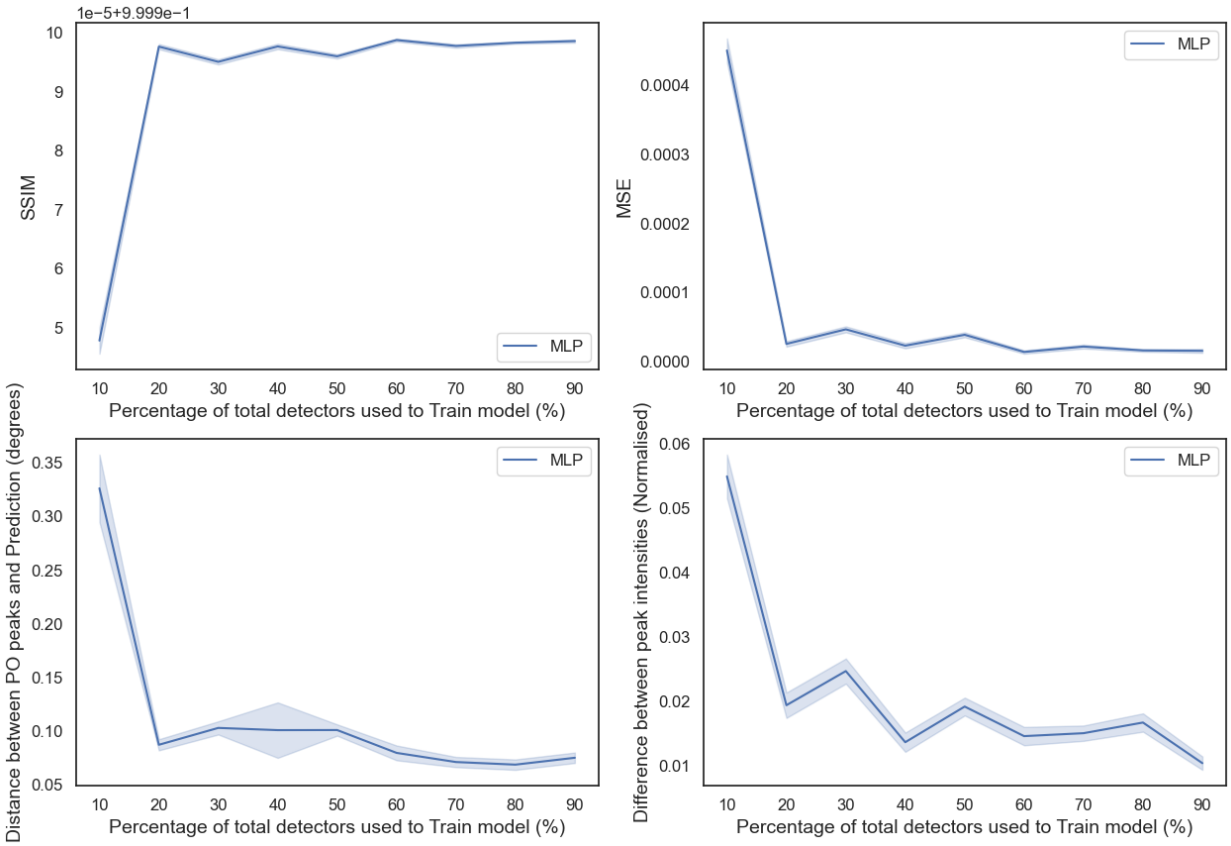

Figure 10. QUBIC detector position interpolation MLP model results, with plots showing SSIM, MSE, peak position differences, and peak intensity differences for different amounts of training data.

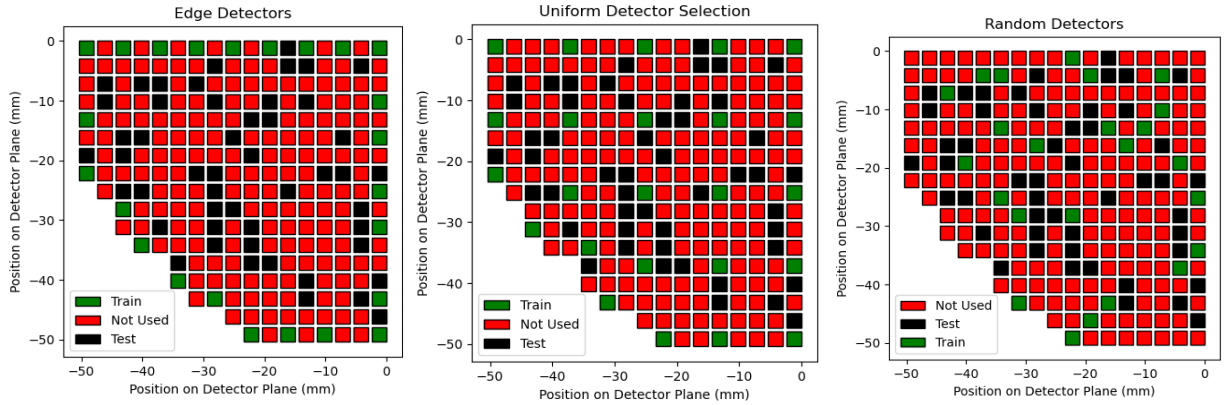

Figure 11. The QUBIC detector positions and each different combination of detectors used for the training set, which were: edge detectors (left), a uniform distribution (center), and a random selection of detectors (right). The green represents the selected detectors which were used to train the model, and the black represents the test set used (which never included a detector in the training set). The TD focal plane is a single quadrant of a larger full focal plane to be implemented in future.

The combination of detectors that gives the best results was also investigated. To do this, different combinations of 24 detectors, or approximately 10% of the total number of detectors, were used to train MLP models to see if any of them would perform better. The first combination was a uniform distribution of the detectors, throughout the detector plane. The second was every other detector around the edges. Finally random selections were also used to act as a baseline. A diagram

of the detector combinations is shown in figure 11. Table 1 shows the results for the different model types; it can be seen that the uniform selection and the random selection have similar performance, whereas the selection of edge detectors have the worst performance.

Table 1. Comparison of models trained on different detector combinations for the differences between the original PO synthesized beam and the models , ± indicates one standard deviation.

	SSIM	MSE	Peak distance (degrees)	Peak Intensity Difference
Uniform	1.000 ± 7.51E-06	5.94E-05 ± 6.76E-05	0.149 ± 0.188	0.043 ± 0.035
Edge	1.000 ± 1.14E-05	1.61E-04 ± 9.02E-05	0.170 ± 0.153	0.062 ± 0.031
Random	1.000 ± 1.14E-05	7.25E-05 ± 1.07E-04	0.118 ± 0.132	0.038 ± 0.033

The performance of an LSTM was also investigated, it was found to take much longer to train but could produce similar results to the MLP, an example of its output is shown in figure 12.

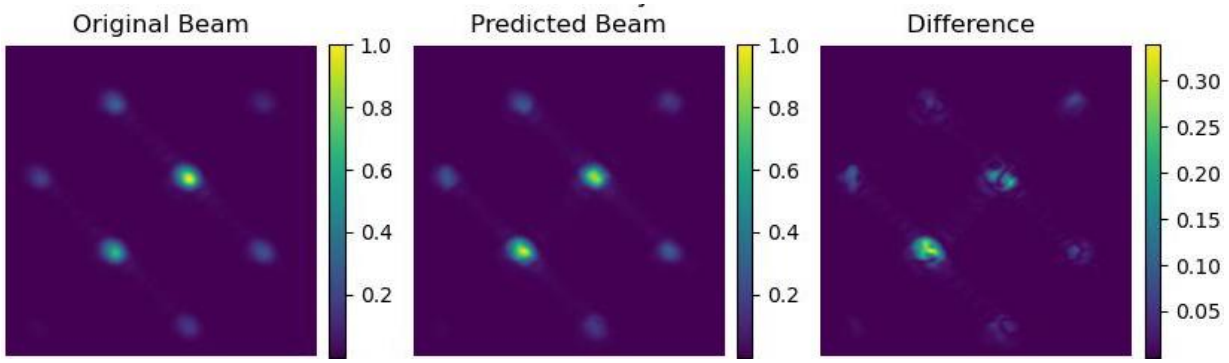

Figure 12. QUBIC detector position interpolation LSTM model output trained on 20% of the detectors. (This was for a TES detector at $x = -2.4$ mm, $y = -41.4$ mm from the center of the focal plane.)

The QUBIC synthesized beam was then interpolated at different frequencies using ML techniques. An MLP and an LSTM were used. Beams were simulated at frequencies from 130 to 195 GHz, at 5 GHz intervals. Before the model was trained, the natural log of the data was taken, as it was found the ML models performed better when this was done. The output was then converted back when the ML prediction was compared to the PO beam. Both MLP and LSTM models were able to replicate the synthesized beam once trained on enough frequencies. An example of the outputs for both models are shown in figures 13 and 14.

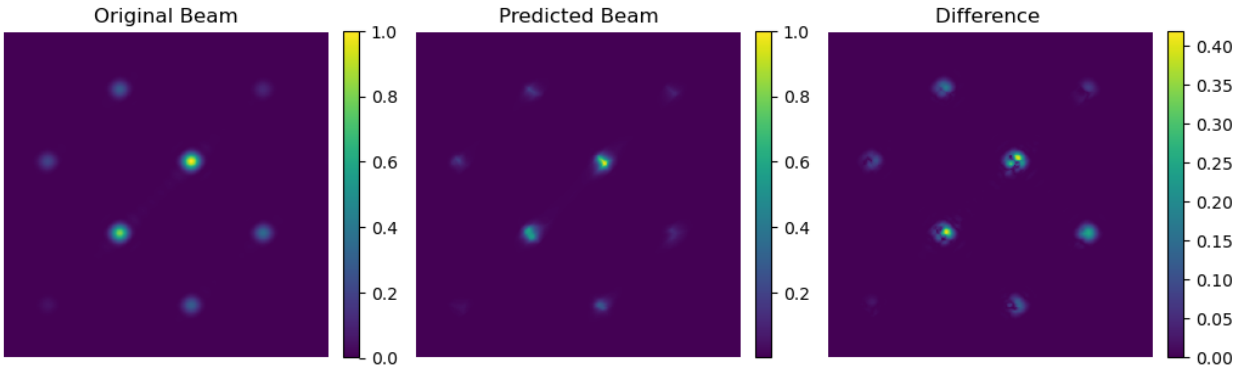

Figure 13. QUBIC frequency interpolation model output at 130 GHz for the MLP model trained on 14 frequencies. (This was for a TES detector at $x = -23.4$ mm, $y = -23.4$ mm from the center of the focal plane.)

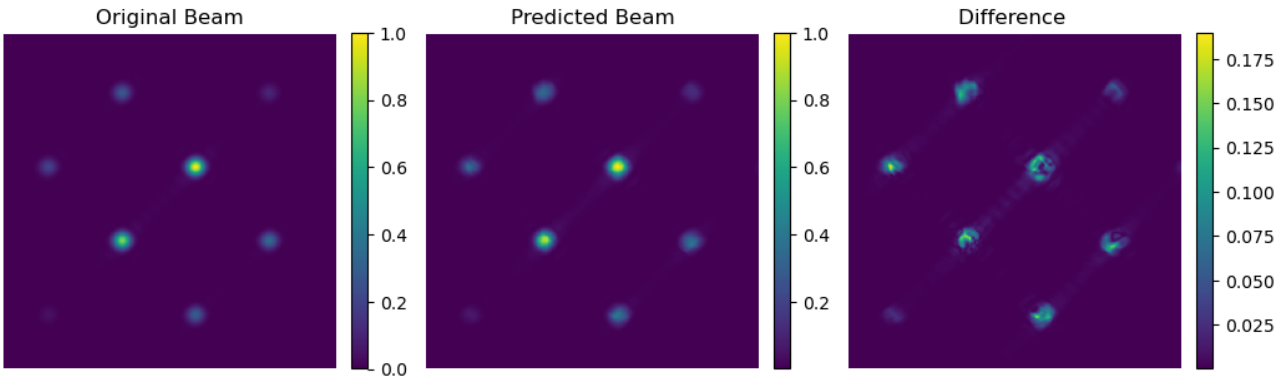

Figure 14. QUBIC frequency interpolation model output at 130 GHz for the LSTM model trained on 9 frequencies. (This was for a TES detector at $x = -23.4$ mm, $y = -23.4$ mm from the center of the focal plane.)

As shown in figure 15, the metrics for both the MLP and the LSTM were calculated for models trained on varying amounts of data. It can be seen that the LSTM model performed much better than the MLP for smaller amounts of training data. However, the training times for the LSTM were much longer than the MLP, particularly as more data are added to the training set. The LSTM needs just 5 frequencies to perform best, and still has a similar performance at 3 frequences, however the MLP requires at least 9 frequencies for the optimal performance.

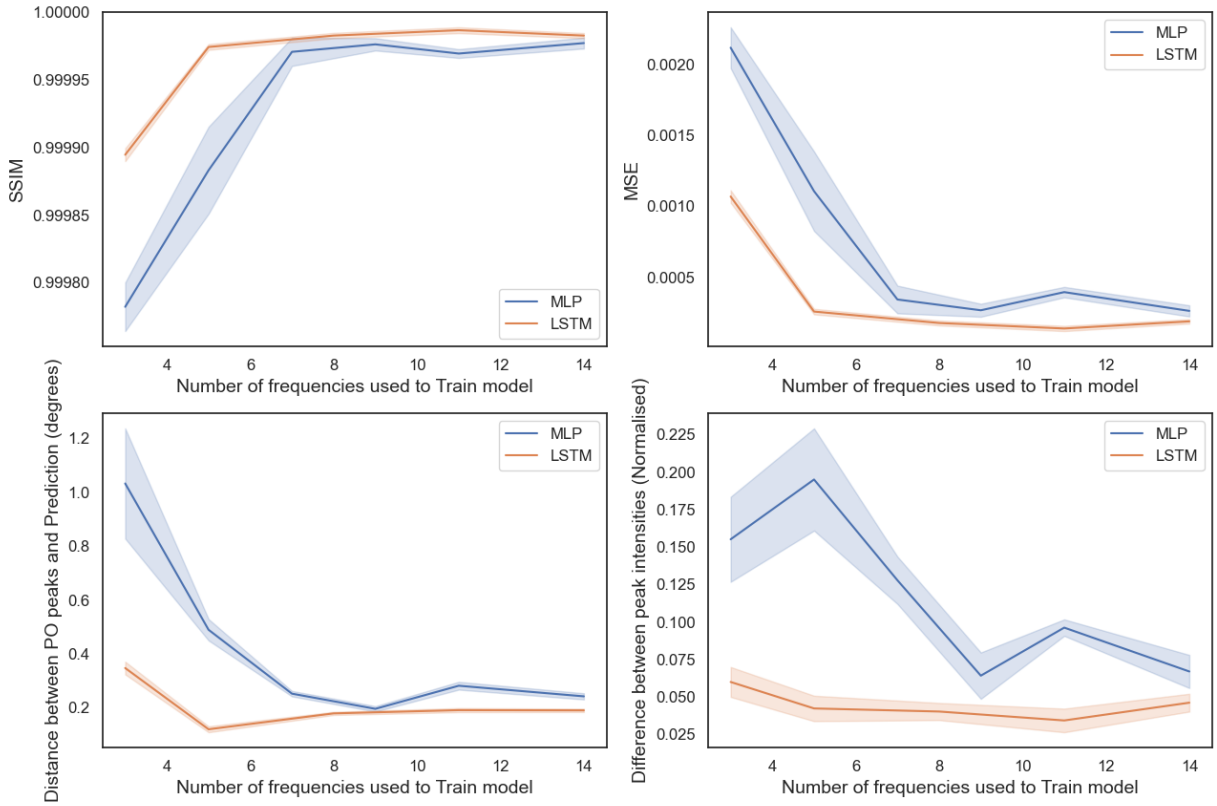

Figure 15. QUBIC frequency interpolation results for both MLP and LSTM, with plots showing SSIM, MSE, peak position differences, and peak intensity differences for different amounts of training data.

Finally, the models were edited to include the frequency and x and y positions of the detectors as independent variables. Simulations of the synthesized beam were carried out with PO in order to train the models, and the detectors chosen for simulation were informed by the results from previous sections. 24 detectors were simulated at 5 frequencies which were 130, 140, 160, 180 and 190 GHz. The data simulated with PO are represented in figure 16. . In order to simulate these, the cluster computer in the Maynooth Physics department was used. It took approximately 83 hours of computing time to simulate these data with MODAL.

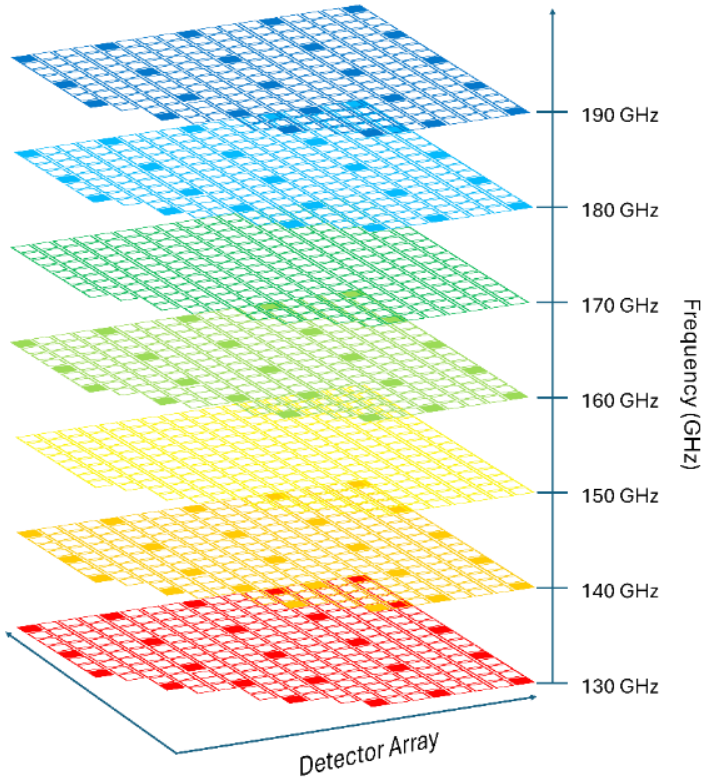

Figure 16. Training data simulated with PO for QUBIC frequency and detector position interpolation. The solid squares represent the detectors which were simulated with PO and used for training.

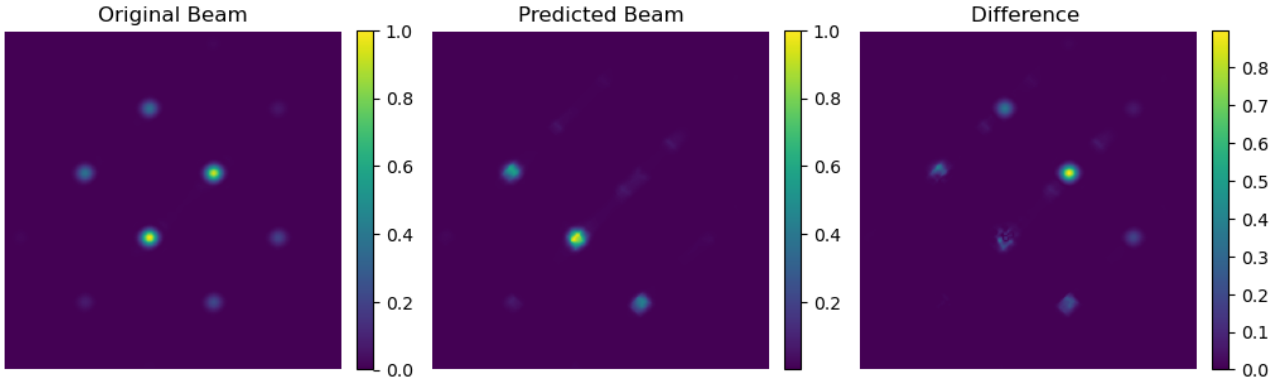

Figure 17. QUBIC frequency and detector position interpolation model output for the MLP model at 150 GHz. (This was for a TES detector at $x = -20.4$ mm, $y = -11.4$ mm from the center of the focal plane.)

This training data were then used with both the MLP model and LSTM model. The output for the MLP can be seen for a detector at 150 GHz in figure 17. None of the 150 GHz detectors were included in the training data, and it's clear that the MLP model cannot replicate the synthesised beam at 150 GHz when trained on this amount of data. However, when

interpolating unsimulated detectors at a simulated frequency the model performs better. An LSTM model was also trained on the data, and an example of its output can be seen in figure 18, where the model was successful in replicating the synthesized beam at unseen detectors and frequencies. The LSTM was then tested on a selection of detectors at 150 GHz. The results for the performance on these detectors are given in table 2.

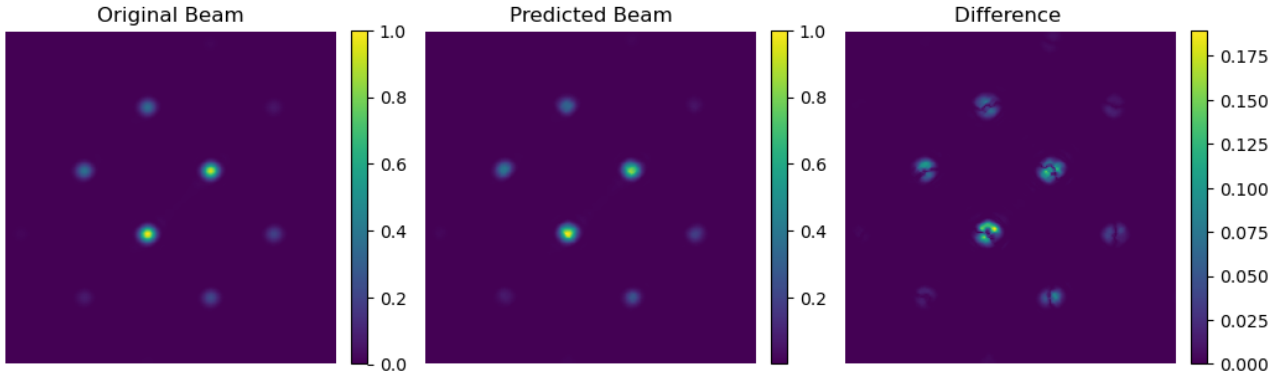

Figure 18. QUBIC frequency and detector position interpolation model output for the LSTM model at 150 GHz. (This was for a TES detector at $x = -20.4$ mm, $y = -11.4$ mm from the center of the focal plane.)

Table 2. The normalized differences between the original PO synthesized beam and the LSTM model interpolation with frequency and detector position as parameters, ± indicates one standard deviation.

SSIM	MSE	Peak distance (degrees)	Peak Intensity Difference
1.000 ± 7.245E-06	7.192E-05 ± 7.474E-05	0.133 ± 0.033	0.023 ± 0.012

In order to use the Qubicsoft pipeline to determine how differences in the ML prediction will affect the overall output of QUBIC, the LSTM model was used to predict the synthesized beam for all 248 detectors at 150 GHz. The peak positions and intensities were then found so they could be used within the Qubicsoft pipeline. Figure 19 shows the B-mode power spectra (the object of interest for cosmologists) for the input sky, which was "observed" with the fully simulated PO beam as well as the reconstructed beam for both the ML prediction and the analytical formula for the synthesized beam.

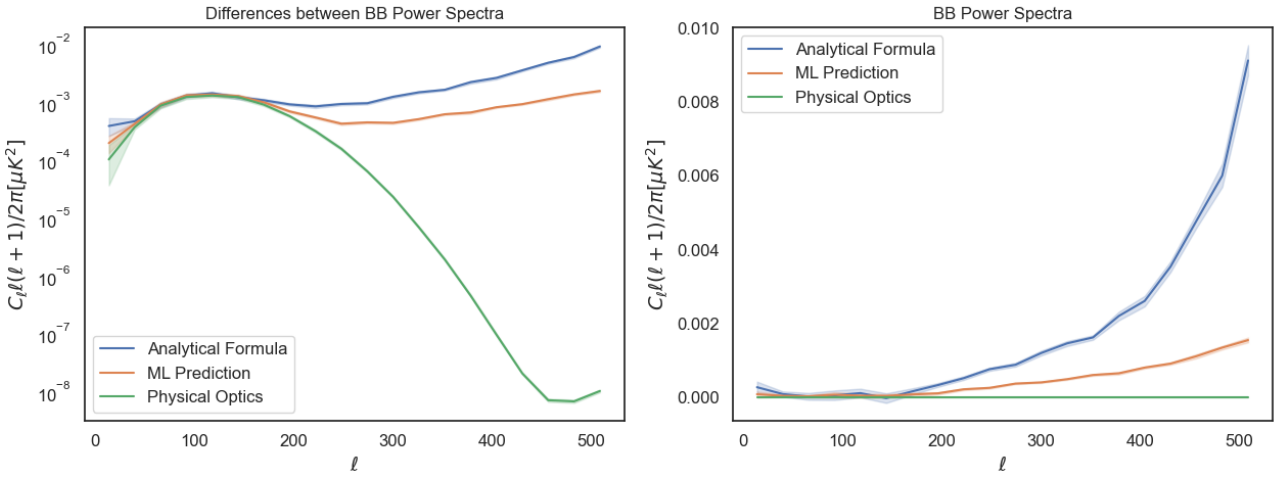

Figure 19. Plot of the B-mode power spectrum for a sky observed with the PO synthesized beam and reconstructed with the LSTM beam prediction and the analytical synthesized beam. The shaded areas represent ± one standard deviation, as the simulation was performed on 30 different sky patches. Differences become most notable at multipole higher than about $l = 150$ where angle on the sky $\theta \approx 180°/\pi$.

7. CONCLUSION

The aim of this work was to investigate whether or not the synthesized beam for the QUBIC instrument could be interpolated using machine learning. This was done using two different models, an MLP model and an LSTM model. The interpolation problem was approached in three different ways; interpolating the pattern at different frequencies but for a single detector, interpolating the pattern at different detectors but for a single frequency, and interpolating the pattern at both different frequencies and different detectors. By interpolating for frequency only and for detector position only, it was possible to estimate the amount of data required for the third combined case.

For the amount of data generated, only the LSTM was successful in replicating the synthesized beam. It was used to generate the synthesized beam for all 248 detectors at 150 GHz and compared to a fully simulated PO beam at 150 GHz using the QUBIC collaboration's Qubicsoft data analysis pipeline.

It was shown that that although the LSTM ML prediction did not replicate the PO synthesized beam perfectly, it was closer than the beam produced with the analytical formula. Although more time must be spent producing the training data and training the model itself, once this is completed, it takes a similar amount of time to produce the beam with ML as it does by evaluating the analytical formula. This means that we can produce a more accurate beam with some extra computational cost. In addition to this the size of the fully trained LSTM model is 18 MB, which is much more convenient to share with collaborators than full images, especially when considering maximum upload sizes on sites such as Github.

ACKNOWLEDGEMENTS

This publication has emanated from research supported in part by a grant from Taighde Éireann – Research Ireland under Grant number 18/CRT/6049. For the purpose of Open Access, the author has applied a CC BY public copyright license to any Author Accepted Manuscript version arising from this submission.

REFERENCES

[1] J.-Ch. Hamilton *et al.*, "QUBIC I: Overview and science program," *Journal of Cosmology and Astroparticle Physics*, vol. 2022, no. 04, p. 034, Apr. 2022, doi: 10.1088/1475-7516/2022/04/034.

[2] C. O'Sullivan *et al.*, "QUBIC VIII: Optical design and performance," *Journal of Cosmology and Astroparticle Physics*, vol. 2022, no. 04, p. 041, Apr. 2022, doi: 10.1088/1475-7516/2022/04/041.

[3] M. O. Akinsolu, K. K. Mistry, B. Liu, P. I. Lazaridis, and P. Excell, "Machine Learning-assisted Antenna Design optimization: A Review and the State-of-the-art," *14th European Conference on Antennas and Propagation, EuCAP 2020*, Mar. 2020, doi: 10.23919/EUCAP48036.2020.9135936.

[4] P. Testolina *et al.*, "Enabling Simulation-Based Optimization Through Machine Learning: A Case Study on Antenna Design," *Proceedings - IEEE Global Communications Conference, GLOBECOM*, Aug. 2019, doi: 10.1109/GLOBECOM38437.2019.9013240.

[5] J. J. Hibbard, B. D. Nhan, D. Rapetti, and J. O. Burns, "MEDEA: A New Model for Emulating Radio Antenna Beam Patterns for 21-cm Cosmology and Antenna Design Studies," Aug. 2024, doi: 10.5281/zenodo.11584632.

[6] "GitHub - qubicsoft/qubic: Data analysis tools for the QUBIC experiment." Accessed: Apr. 03, 2024. [Online]. Available: https://github.com/qubicsoft/qubic

[7] M. L. Gradziel *et al.*, "Modelling of the optical performance of millimeter-wave instruments in MODAL," in *Proceedings Volume 6472, Terahertz and Gigahertz Electronics and Photonics VI*, K. J. Linden and L. P. Sadwick, Eds., Feb. 2007, p. 64720D. doi: 10.1117/12.700680.

[8] J. Gutiérrez-Meana, J. A. Martínez-Lorenzo, F. Las-Heras, J. Gutiérrez-Meana, J. A. Martínez-Lorenzo, and F. Las-Heras, "High Frequency Techniques: the Physical Optics Approximation and the Modified Equivalent Current Approximation (MECA)," *Electromagnetic Waves Propagation in Complex Matter*, Jul. 2011, doi: 10.5772/17307.

[9] E. S. Battistelli *et al.*, "QUBIC: The Q & U Bolometric Interferometer for Cosmology," *J Low Temp Phys*, vol. 199, no. 1–2, pp. 482–490, Apr. 2020, doi: 10.1007/S10909-020-02370-0.

[10] I. Goodfellow, Y. Bengio, and A. Courville, “Deep Feedforward Networks,” in *Deep Learning (Adaptive Computation and Machine Learning series)* , MIT Press, 2016, ch. 6, pp. 164–191.
[11] K. Hinkelmann, “Neural Networks,” University of Applied Sciences Northwestern Switzerland.
[12] G. James, D. Witten, T. Hastie, R. Tibshirani, and J. Taylor, “An Introduction to Statistical Learning,” 2023, doi: 10.1007/978-3-031-38747-0.
[13] D. P. Kingma and J. Lei Ba, “ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION”.
[14] F. Chollet, “Keras,” Github. Accessed: May 22, 2025. [Online]. Available: https://github.com/fchollet/keras
[15] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *Nature 1986 323:6088*, vol. 323, no. 6088, pp. 533–536, 1986, doi: 10.1038/323533a0.
[16] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valaee, “Recent Advances in Recurrent Neural Networks,” Feb. 2018, Accessed: Jul. 14, 2025. [Online]. Available: arXiv:1801.01078
[17] A. Sherstinsky, “Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network,” *Physica D*, vol. 404, p. 132306, Mar. 2020, doi: 10.1016/J.PHYSD.2019.132306.
[18] Simeon. Kostadinov, *Recurrent Neural Networks with Python Quick Start Guide : Sequential Learning and Language Modeling with TensorFlow.* Packt Publishing Ltd, 2018.
[19] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/NECO.1997.9.8.1735.
[20] Ovidiu. Calin, “Recurrent Neural Networks: LSTM,” in *Deep learning architectures : a mathematical approach*, Springer, 2020, ch. 17, pp. 554–556.
[21] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image Quality Assessment: From Error Visibility to Structural Similarity,” *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, Apr. 2004, doi: 10.1109/TIP.2003.819861.
[22] L. Mousset *et al.*, “QUBIC II: Spectral polarimetry with bolometric interferometry,” *Journal of Cosmology and Astroparticle Physics*, vol. 2022, no. 04, p. 035, Apr. 2022, doi: 10.1088/1475-7516/2022/04/035.