

BSc Project

**ACCELERATING LASER-DRIVEN ION
SOURCE INSTRUMENTATION USING
MACHINE LEARNING**

Tik Tsoi
CID: 01701209

Supervisor: Nicholas P. Dover
Assessor: Oliver C. Ettlinger

Word count: 5993

May 2022

Abstract

In this paper, we demonstrated a machine learning method for extracting spectral beam parameters from a laser-driven ion source using a convolutional neural network (CNN). The training and test data for the model were synthetically generated using our simulation model based on the Thomson parabola spectrometer. We first applied the model to generate a 64x64 dataset consisting of multiple ion species with experimental noise added. Subsequently, we proceeded to compare the measurement accuracy and speed of the CNN with conventional analysis techniques, targeting the protons. By considering the absolute percentage error of our results, we showed that the CNN is 5 times, 10 times and 12 times more accurate in predicting the effective temperature T , maximum energy E_{max} and particle count N . The same set of measurements was executed 1.5 times faster using a local computer. Real-time output is feasible. However, that might require high performance computing or compromises on measurement accuracy.

Declaration of Work Undertaken

I declare that the design and construction of the code for this paper were jointly undertaken by myself and my project partner. Subsequent data analysis and evaluation are entirely my own work.

Contents

| | | |
|----------|---|-----------|
| 1 | Introduction | 4 |
| 2 | Data Synthesis | 5 |
| 2.1 | Thomson Parabola Spectrometer | 5 |
| 2.1.1 | Working Principle and Setup | 5 |
| 2.1.2 | Modelling | 6 |
| 2.1.3 | Validation and Optimisation | 7 |
| 2.2 | Charge-coupled Device (CCD) Camera | 8 |
| 2.2.1 | Working Principle | 8 |
| 2.2.2 | Image Formation and Noise Modelling | 8 |
| 2.2.3 | Dataset Creation | 8 |
| 3 | Machine Learning Techniques | 10 |
| 3.1 | Convolutional Neural Network (CNN) | 10 |
| 3.1.1 | Algorithm | 10 |
| 3.1.2 | Data Preprocessing | 10 |
| 3.1.3 | Architecture | 10 |
| 3.1.4 | Optimisation and Validation | 11 |
| 4 | Traditional Analysis Techniques | 12 |
| 4.1 | Median Filtering | 12 |
| 4.2 | Thermal Distribution Reconstruction | 13 |
| 5 | Flow of Evaluation | 14 |
| 6 | Results | 14 |
| 6.1 | Accuracy of Measurements | 14 |
| 6.1.1 | Summary | 14 |
| 6.1.2 | Asymmetry | 14 |
| 6.1.3 | Outliers | 15 |
| 6.2 | Speed of Measurements | 16 |
| 6.2.1 | Summary | 16 |
| 7 | Discussions | 16 |
| 7.1 | Energy Dependency | 16 |
| 7.2 | Limitations | 17 |
| 7.2.1 | Traditional Analysis | 17 |
| 7.2.2 | Machine Learning | 17 |
| 7.3 | Real-time Measurements | 18 |
| 7.4 | Practical implementation | 18 |
| 7.4.1 | What's Missing | 18 |
| 7.4.2 | Future Extensions | 19 |
| 8 | Conclusion | 20 |

1 Introduction

Plasma acceleration is a novel technique for accelerating charged particles to high energies by utilising the high electromagnetic fields sustained by a plasma, typically a thousand to a million times stronger than conventional accelerators [1]. As a result, plasma-based accelerators can be manufactured compactly and cheaply. The numerous advantages of these accelerators have enabled a wide range of applications which can be implemented on a much smaller scale. This technology may be applied to radiotherapy: current use of matter particles in the field is limited owing to their shallow penetration depth. Yet with relativistic electrons from a plasma accelerator, the deep-seated tumours can now be reached and destroyed in a short burst, leading to more effective treatment with fewer side effects.

The technology behind this advancement is enabled by ultrashort, high-intensity lasers, capable of generating laser intensities up to 10^{22} W/cm^2 [2], which are aimed at a target foil in a plasma accelerator (see Fig 1). At such intensities, bound electrons are ripped from their parent atoms and accelerated to relativistic speeds, transforming the target body into a hot plasma. As the fast electrons propagate and eventually leave the material, an internal electric field is set up at the rear surface. The main area of interest in this paper is to study the spectral parameters of the light ions such as protons and carbon ions, which are released from atoms at the rear surface by the strong field. These properties contain information about the energy spectrum of the ion beams, hence are crucial for understanding the characteristics of the ion beams and devising their appropriate applications.

Previously, high-intensity ultrashort lasers were restricted by low-repetition rates of $< 10 \text{ Hz}$. Measurements of the particle beams were possible using self-developing films, where different species were analysed separately. With the recent breakthrough of high-repetition-rate lasers ($10 - 100 \text{ Hz}$), a charge-coupled device (CCD) camera system was put in place instead. However, the extra convenience also comes with a cost. First, the quality of digital photographs heavily depends on pixel counts. As images are captured in finite-sized “bins”, it is inevitable that final results are spatially limited. Second, various types of noise are inherent to a camera system, which can significantly undermine image quality. For example, electric current and thermal heat associated with the camera’s electronic circuits can result in non-existing counts by so-called dark current (see Sec 2.2.2 for further discussion). With regards to the latter issue, traditional image denoising and analysis techniques are deemed slow and inaccurate. Henceforth, to achieve real-time output, a more robust and efficient method is demanded.

In recent years, machine learning (ML) has gained importance in many fields of Physics, thanks to the ability of ML models to recognise patterns in data and to make subsequent predictions. A type of model, known as the convolutional neural network (CNN), is widely applied in image analysis such as object identification and classification. Similar research has shown that neural networks are effective at denoising images from a laser-driven ion source [3,4,5]. Here, we propose using CNN to extract beam parameters of ions from noisy images based on the Thomson parabola spectrometer setup. This technique requires first generating synthetic data using Monte Carlo simulations, which were used in model training and analysis. The predictions from the CNN were then compared to the results from traditional analysis techniques in both speed and accuracy. As a proof of concept, we focused on the measurement of three spectral parameters of the proton beam including its effective temperature T , maximum energy E_{max} and particle count N .

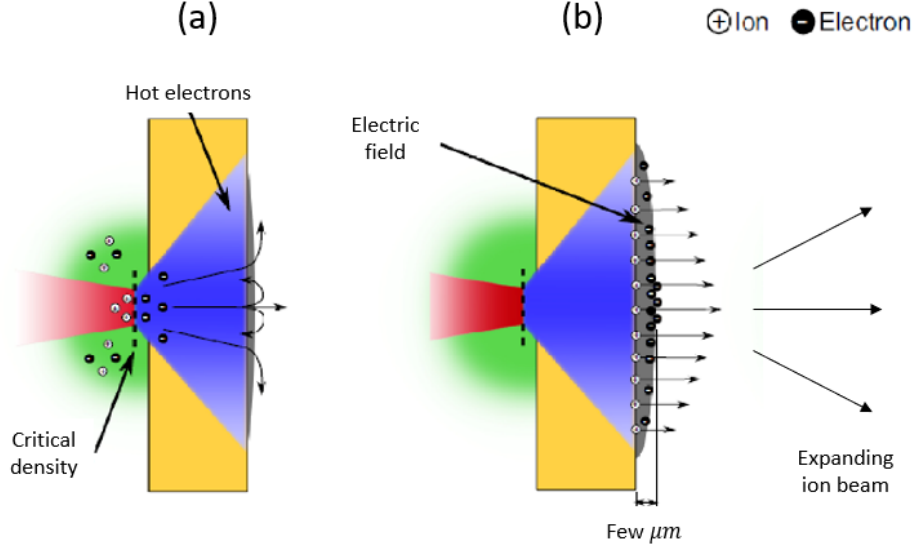


Figure 1: (a) The high-intensity laser pulse interacts with the front surface of the target. At the critical density, electrons are accelerated in the forward direction. They propagate through the material and eventually escape from the rear surface. (b) The charge separation induces an electric field and accelerates the lighter ions which proceed to expand in a range of directions [6].

2 Data Synthesis

2.1 Thomson Parabola Spectrometer

2.1.1 Working Principle and Setup

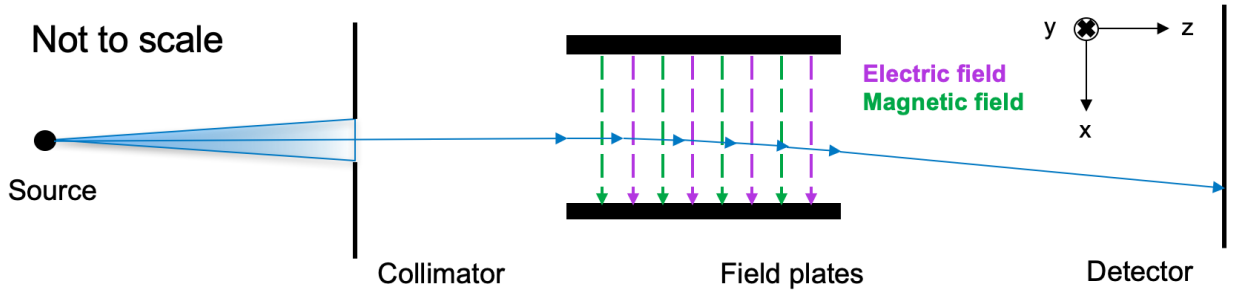


Figure 2: Schematic diagram of the Thomson parabola spectrometer. An example ray (blue) travelling in the $+z$ direction is drawn as an illustration.

What makes the Thomson parabola spectrometer a powerful diagnostic tool is that it allows precise and accurate measurements of multiple ion species simultaneously (see Eq 4). The spectrometer originates from the laser-driven ion beam produced at the back of the target foil, which is then narrowed by a pinhole collimator. The collimated beam then enters a region of electric and magnetic fields which are both aligned perpendicular to the beam propagation direction. Inside this region, the motion of the beam ions is governed by the Lorentz force F :

$$F = q(\vec{E} + \vec{v} \times \vec{B}), \quad (1)$$

where q and \vec{v} are the charge and velocity of the ions, \vec{E} and \vec{B} are the electric and magnetic fields. As a result, the ion beam is deflected from their straight trajectories. Beyond the fields,

ions that arrive at the detector plane are recorded by a CCD camera. Assuming uniform fields and non-relativistic particles, the deflections of ions in the x-axis and y-axis are described by the two equations (note z-axis is defined in the ion propagation direction):

$$x = \frac{qElD}{2E_{kin}}, \quad (2) \quad y = \frac{qBlD}{\sqrt{2mE_{kin}}}, \quad (3)$$

where E_{kin} is the kinetic energy of the ions. Combining Eq (2) and (3) results in a parabolic equation:

$$y^2 = \frac{q}{m} \frac{B^2 l D}{E} x. \quad (4)$$

Here we see that ion species conveniently separate into different parabolic tracks according to their charge-to-mass ratio q/m . Particularly, Eq (2) and (3) show two important results. First, deflections in x are caused by the electric field whereas deflections in y are caused by the magnetic field. Second, deflections are inversely proportional to some power of E_{kin} , meaning slower travelling ions are more deflected from the origin.

The experimental setup evaluated in this paper was based on the above. Additionally, two minor modifications were made to the setup discussed above in the actual implementation. First, instead of a finite-sized ion source, a point source was considered, which would reduce the beam divergence at the collimator. However, considering that the size of the pinhole \ll size of the source, this impact is negligible. Second, a non-uniform experimental magnetic field was used [7], resulting in an edge effect in the field (see Fig 3). This means that the actual deflections will differ to some degree from the above ideal case.

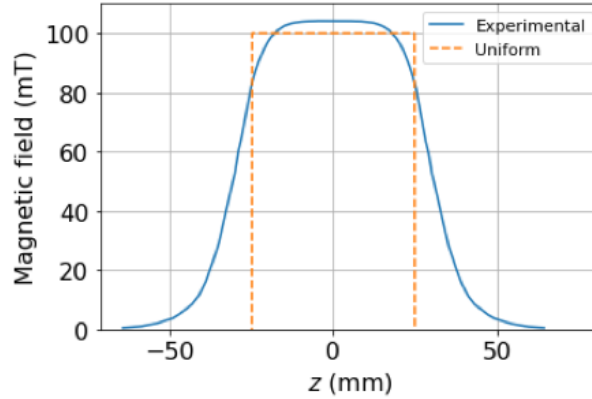


Figure 3: Comparison of the experimental (blue) and uniform (orange) magnetic field profiles. The experimental field shows a long tail beyond the end of the field plates, indicating edge effects.

2.1.2 Modelling

The simulation model was written with object-oriented programming. On a high level, the *Beam* class takes in three parameters: effective temperature T , maximum energy E_{max} and particle number N , which in turn calls the *Particle* class and generates N number of ions. Each ion is then individually propagated in the Thomson parabola spectrometer.

Monte Carlo Simulations To save computation time, we chose to generate the ions at the collimator. This requires two different samplings to be made: the particle position \vec{r} inside the pinhole and its associated energy E , which was achieved by employing the Monte Carlo method.

The Monte Carlo method relies on repeated uniform samplings from a probability distribution function (PDF) to simulate events that are stochastic in nature. In our implementation, we first randomly sample an x coordinate and a y coordinate from the diameter of the pinhole. If $x^2 + y^2 < r_{pin}^2$, the coordinates are assigned to a new particle, where r_{pin} is the radius of the pinhole. For a collisionless plasma, it turns out that the energy distribution $p(E, T)$ of ions is given by the Boltzmann equation with a cutoff energy E_{max} :

$$p(E, T) = \frac{1}{T} \exp^{-\frac{E}{T}}. \quad (5)$$

A similar random sampling process takes place along Eq (5) to determine an energy value. In addition, a velocity vector is also needed to enable propagation. This is found by tracing the particle back to the point source. The above procedure repeats for all N particles.

Interpolation An experimentally measured magnetic field essentially means that data points are discrete. To obtain a continuous output, interpolation was applied to reconstruct the original function. We adopted the cubic spline interpolation which uses a third-order polynomial as the interpolant.

Finite Difference Method Outside the fields, particles are propagated in single steps. The emergence of the Lorentz force inside the fields means we have to propagate the particles step by step. The finite difference method is commonly applied to numerically solve differential equations using discrete (often small) time steps Δt . Here, by redefining Eq (1), we arrive at a second-order differential equation:

$$\frac{d^2 \vec{r}}{dt^2} - \frac{q}{m} \left(\frac{d\vec{r}}{dt} \times \vec{B} + \vec{E} \right) = 0. \quad (6)$$

Using the Euler method, a first-order finite difference scheme (up to $O(\Delta t)$ accuracy), we can compute the motion of the particles from one instance i to the next instance $i + 1$:

$$\vec{r}_{i+1} = \vec{r}_i + \vec{v}_i \cdot \Delta t, \quad (7) \quad \vec{v}_{i+1} = \vec{v}_i + \vec{a}_i \cdot \Delta t, \quad (8)$$

where \vec{a} is the Lorentz acceleration.

As an aside, ions beyond the reach of a contact surface or intercepted by the field plates become invalid and will not be recorded by the detector plane. This can result in a significant discrepancy between the synthesised count and detected count.

2.1.3 Validation and Optimisation

The model was validated by comparing our numerical solutions to the analytical solutions, Eq (2) and (3), using uniform fields. An absolute percentage error (APE) was computed for all x and y coordinates which have energy between 0.1 MeV and 10 MeV. Figure 4 (a) shows that the simulation accuracy is energy-dependent: the APE asymptotically increases as $E \rightarrow 0$. This is not a surprising result since we know that slower particles experience larger deflections (recall Eq (2) and (3)), meaning the error is magnified when arriving at the detector plane. Overall, we measured the mean APE across the energy range to be 0.05% and 0.03% in x and y respectively, indicating very accurate results.

The model was optimised by varying Δt with a strategy similar to the above. 20 regular time steps between 10^{-13} s and 10^{-9} s were examined in the same energy range. A compromise was made between computation time and overall accuracy (mean APE in x and y combined). This led us to choose $\Delta t = 9 \times 10^{-11}$ s (the sixth last data point from Fig 4 (b)), where a steep decline in computation time occurs.

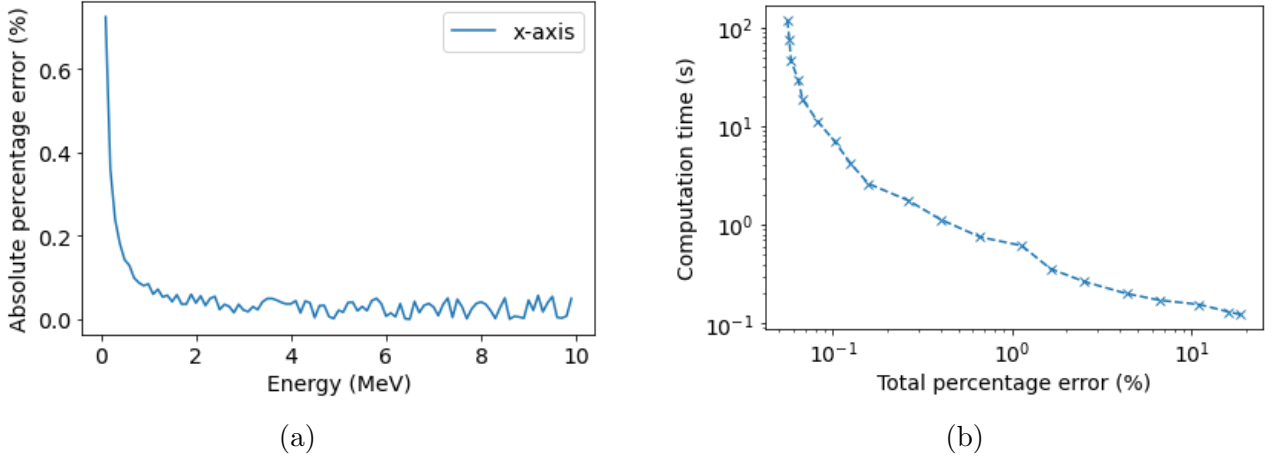


Figure 4: (a) APE of the simulation results against energy in x . y displays a similar trend so is not shown again. (b) Computation time against the combined mean APE.

2.2 Charge-coupled Device (CCD) Camera

2.2.1 Working Principle

A charge-coupled device (CCD) is a microelectronic circuit containing a grid of photosensitive pixels. When an incoming photon falls within a pixel, one or more electrons are ejected due to the photoelectric effect. After a photoshoot is completed, charges bound in each pixel are shifted along with the columns to the readout register and counted up. The camera ultimately stores an intensity value for every pixel.

Note in an actual system, ions are first converted to photons using a scintillator. We neglected this step in the implementation and continued with counting ions.

2.2.2 Image Formation and Noise Modelling

A 2D histogram is computed over the ion track that resided on the detector plane. The histogram has a set number of equal-sized bins which count the number of particles fallen within their areas, much like a CCD. A weight is applied to the histogram to account for the fact that intensity is measured in reality (\propto number of charges). Also, assuming an 8-bit camera, the count is thresholded at $2^8 = 255$.

Other modifications are added sequentially to mimic a noisy experimental image. First, the pixel that is coaxial to the pinhole collimator (corresponding to the bottom-left pixel) is (near) saturated, which is a direct consequence of neutral particles passing straight into the CCD. Second, a minor background noise is added uniformly to mimic the presence of small electric currents in a CCD even when there is no incoming radiation. Third, the image is convolved by a Gaussian function (commonly known as Gaussian blur) to model the optical aberrations found in a camera system. For example, the defocus of light rays after passing through a spherical lens. Fourth, a fraction of pixels are randomly saturated, which is caused by x-rays that are emitted from relativistic electrons (Bremsstrahlung radiation) hitting the CCD.

2.2.3 Dataset Creation

Using the college high performance computing cluster, we constructed a 32x32 dataset and a 64x64 dataset, each with 39925 images. Based on our preliminary results, a 10,000 sample size is a basic requirement to train the CNN to a reasonable accuracy, the more the better. The images contain five ion species, including protons p^+ and four other carbon ions C^+ , C^{2+} , C^{3+}

and C^{4+} . Each ion beam was separately propagated in the spectrometer and added linearly to form a final image (see Fig 5 for example).

Table 1 summarises the parameters used in generating the datasets for this paper. To allow randomness when forming the images, the beam parameters T , E_{max} and N can be varied and are specified by a range. A data multiplier was introduced to enlarge the datasets. Here, a multiplier of two means every image in the dataset is duplicated, after which noise is independently added. Ultimately, there are a clean dataset and a noisy dataset for the two image sizes, totalling 78950 images now. Although the clean images are fundamentally unchanged, the randomness of the noise means that the duplicated noisy images are largely uncorrelated to their parents (apart from the underlying tracks but their intensity is a lot weaker than that of the noise), thus appearing as new data to the machine learning model.

Table 1: Parameters used in the dataset creation. The choice of beam parameters was advised by our supervisor based on typical experimental conditions. The beam parameters here are based on the protons, carbon ions only have 1/100 of the proton number N but share the same T and E_{max} .

(a) Image configuration.

(b) Modifications.

| Description | | Description | |
|---------------------------|----------------|-------------------------------------|-----|
| Image size (pixel) | 32x32, 64x64 | Weight | 0.1 |
| Detector size (mm^2) | 35x35 | Zero point error (count) | 250 |
| T range (MeV) | [0.1, 6.0] | Detector threshold (count) | 255 |
| E_{max} range (MeV) | [4.0, 10.0] | Uniform background (count) | 10 |
| N range (particles) | [18000, 24000] | σ of Gaussian filter (pixel) | 0.2 |
| | | Hard-hit noise fraction | 0.1 |

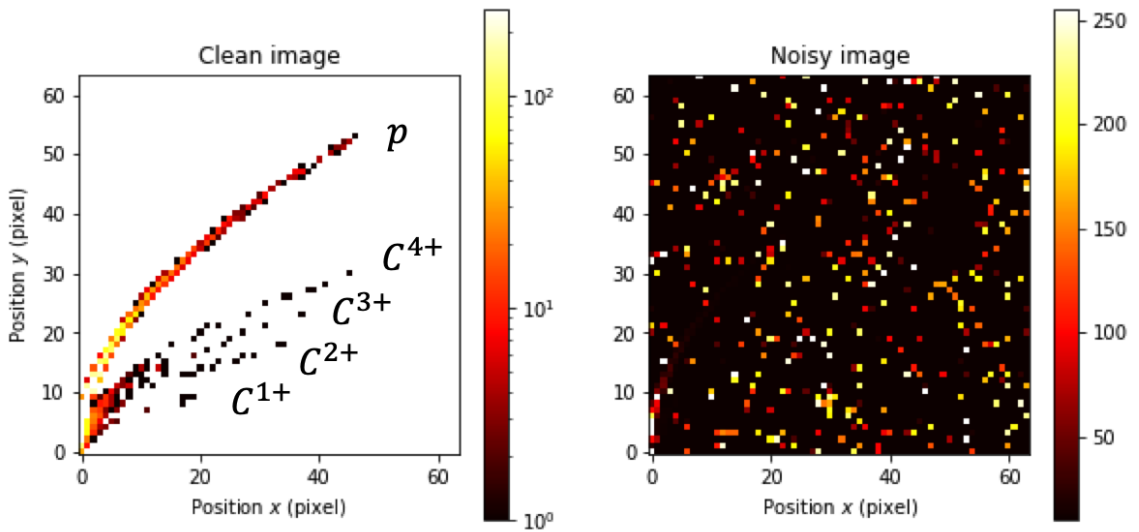


Figure 5: Clean and noisy images (64x64) created based on the parameters in Tab 1. The clean image was plotted on logarithmic scale to accentuate the weak carbon ion tracks.

3 Machine Learning Techniques

3.1 Convolutional Neural Network (CNN)

3.1.1 Algorithm

At heart, a CNN is an algorithm belonging to supervised machine learning. In supervised learning, the model is trained and tested using labelled input. For example, in our case, all images are paired with three defined beam parameters T , E_{max} and N . The aim of the algorithm is then to extract a mapping function from the training data and predict the outcome of unseen data using the function. For its best performance, these function is optimised based on a loss function, similar to the process of minimising a negative log-likelihood for estimating parameters in statistics, by iterating over the same dataset many times. Ideally, when the loss is zero, the model gives perfect predictions.

3.1.2 Data Preprocessing

Two types of data, images \mathbf{X} and the associated variables \mathbf{y} are required as input. Here, \mathbf{X} is the 64x64 noisy image dataset and \mathbf{y} is an array that contains the beam parameters of the images in the dataset. To help feature extraction, both \mathbf{X} and \mathbf{y} are normalised to values $\in \{0, 1\}$.

3.1.3 Architecture

A basic CNN consists of convolutional layers, pooling layers and dense layers. To extract features with higher complexity, convolutional layers and pooling layers are often repeated.

Convolutional Layer The convolutional layer extract features by convolving training images with small kernels (filters). A 3x3 kernel was chosen here. In a convolution, the kernel slides across the image and computes an elementwise multiplication with a stride of 1. After the product, values inside the kernel are summed together and output to a position on a new plane called the feature map (see Fig 6 (a)). By using many filters, we can obtain a set of feature maps which can be used to describe the various visual properties of an image. Note zero padding is added around the edges to keep in-plane dimensions the same. Otherwise, feature maps will reduce in size after a convolution and we risk losing important information near the ends. The activation function selected is the nonlinear Rectified Linear Unit (ReLU) function, which has the following form:

$$a(x) = \begin{cases} 0 & x < 0 \\ x & x \geq 0, \end{cases} \quad (9)$$

where x is the weighted sum of the features. Thus, an activation function generally serves the purpose of weighing the participation of a particular feature map in the next layer. For example, a feature map makes no contribution when x is less than zero, with increasing involvement for more positive x .

Pooling Layer Often a convolutional layer contains an excessive number of parameters, for instance, a 64x64 image with 64 kernels means $64^3 = 262144$ parameters need optimisation. This can significantly hinder the learning process. Thus, a pooling layer is used to downsample the feature maps. We have chosen the max pooling operation and a 2x2 kernel with a stride of 2. Different from convolution, the kernel moves forward in steps of two and outputs the maximum number inside the kernel (see Fig 6 (b)). As a result, the height and width of the feature maps are reduced by a factor of 2.

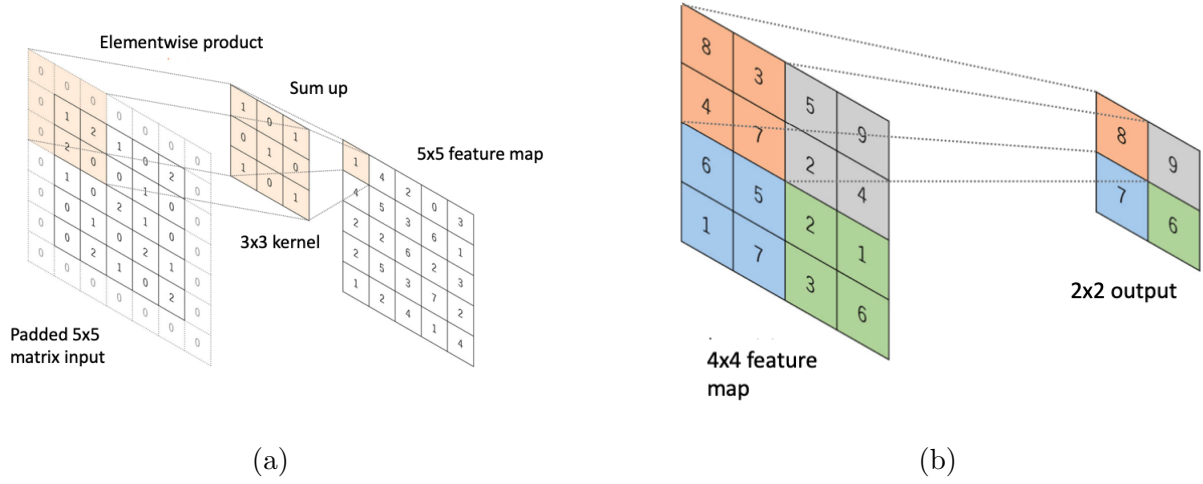


Figure 6: (a) Convolution operation of a 3x3 kernel on a 5x5 input which is zero-padded. The output from the elementwise product goes to a feature map. (b) Max pooling operation of a 2x2 kernel on a 4x4 feature map. The maximum value inside the kernel is output. [8].

Dense Layer Finally, the set of feature maps from the last pooling layer is joined together as a 1D array to form the dense layer. This acts to relate separate pieces of information from earlier parts and classify them. Note since we are handling a regression problem, a linear activation function was chosen in the last layer to predict the three continuous variables \mathbf{y}^{pred} .

3.1.4 Optimisation and Validation

Table 2: Architecture and hyperparameters of the optimised model for the 64x64 dataset. Note the shape of a typical layer is described by (map height, map width, number of maps). The optimised hyperparameters are included for the interest of the reader.

(a) Optimised architecture.

| Layer | Shape | Activation |
|---------------|---------------|------------|
| Convolutional | (64, 64, 64) | ReLU |
| Max pooling | (32, 32, 64) | |
| Convolutional | (32, 32, 128) | ReLU |
| Max pooling | (16, 16, 128) | |
| Convolutional | (16, 16, 64) | ReLU |
| Max pooling | (8, 8, 64) | |
| Dropout | (8, 8, 64) | |
| Flatten | (4096) | |
| Dense | (64) | ReLU |
| Dense | (3) | Linear |

(b) Optimised hyperparameters.

| Hyperparameter | Description |
|------------------------------|-------------|
| Kernel size in convolutional | 3x3 |
| stride in convolutional | 1 |
| kernel size in max pooling | 2x2 |
| Stride in max pooling | 2 |
| Epochs | 100 |
| Learning rate | 0.0001 |
| Batch size | 128 |
| Dropout ratio | 0.25 |

Table 2 shows the architecture of the optimised model for the 64x64 images. Initially, a

standard architecture was adopted from the internet. A process of trial and error followed, including arbitrarily adding and removing layers, testing different types of layers (such as a dropout layer) and altering their arrangement. Parameters which require assignment (known as hyperparameters) were fine-tuned using a built-in Python module (called Keras Tuner). The Tuner works by testing a set of randomly chosen hyperparameter values based on the loss, ones that lead to the minimum loss are retained.

A cross-validation technique was used throughout to test the effectiveness of the model, which requires separating the data into a training set, a validation set and a test set with a 72 : 8 : 20 ratio. The training set supplies the data for training, after which an in-training evaluation is done on the validation set. An overall evaluation can be performed on the test set to examine the accuracy of the model in the end. Part of the validation can be done by monitoring the evolution of the training loss and the validation loss over epochs (number of repeats that the algorithm has trained on the training set). Modelling error such as underfitting and overfitting can be looked out for. For example, a consistently greater validation loss indicates underfitting, and vice versa for overfitting, wherein both cases suggest that the model is inadequate to represent the dataset. Ideally, we would want the two losses to agree throughout the training as shown in Fig 7.

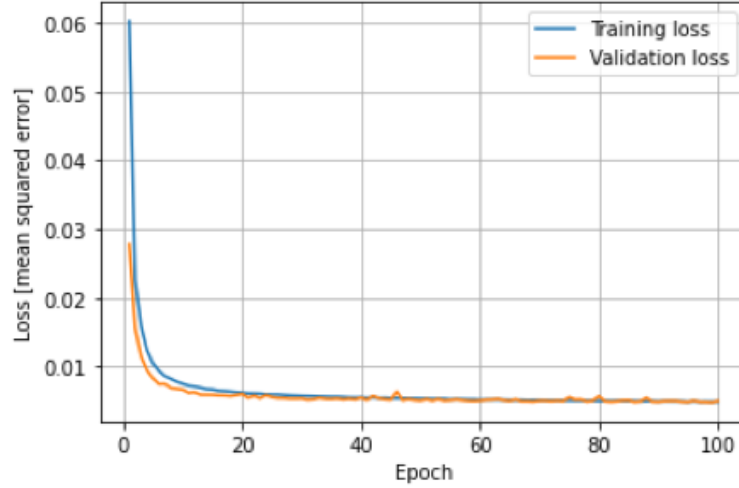


Figure 7: Changes of training loss and validation loss over 100 epochs completed by the optimised model. A mean squared error loss was chosen. Both losses agree throughout the training and flatten out near the end.

4 Traditional Analysis Techniques

4.1 Median Filtering

For traditional analysis, noisy multi-track images have to be first transformed to clean single proton track images via manual noise reduction. A filtering technique well-suited for removing the dominant hard-hit noise (noise with oddly large amplitude) in our images is median filtering. Like convolution, a window slides across the image but this time, a median value is selected from the numbers within the window. It is also favoured for the reason that sharp features are preserved. With other choices, the mean filtering for example, “non-existing” pixel values are created, which can result in smudging around edges. The optimal kernel size and step size were decided through trial and error, where we found a 3x3 kernel with a stride of 1 results in the best performance. Two metrics, the peak signal-to-noise ratio (PSNR) and Structural

Similarity Index Measure (SSIM), were implemented as guidance [3]. PNSR measures the maximum power of a signal relative to its noise and SSIM measures the likeness between the filtered image and the original image. A perfect reconstruction is defined by $PNSR = \infty$ and $SSIM = 1$.

Additional steps are required to minimise other noise. Since the ion tracks only cover a small fraction of the image, most pixels are populated by the uniform background noise. Thus, its magnitude is surveyed by finding the modal intensity in the image and subtracted from every pixel. The carbon residue is minimised by thresholding weak signals in the bottom half of the image (where the carbon tracks lie). The effect on the proton track is minimal because it possesses a much stronger intensity. Lastly, intensity is converted back into count by undoing the particle weight.

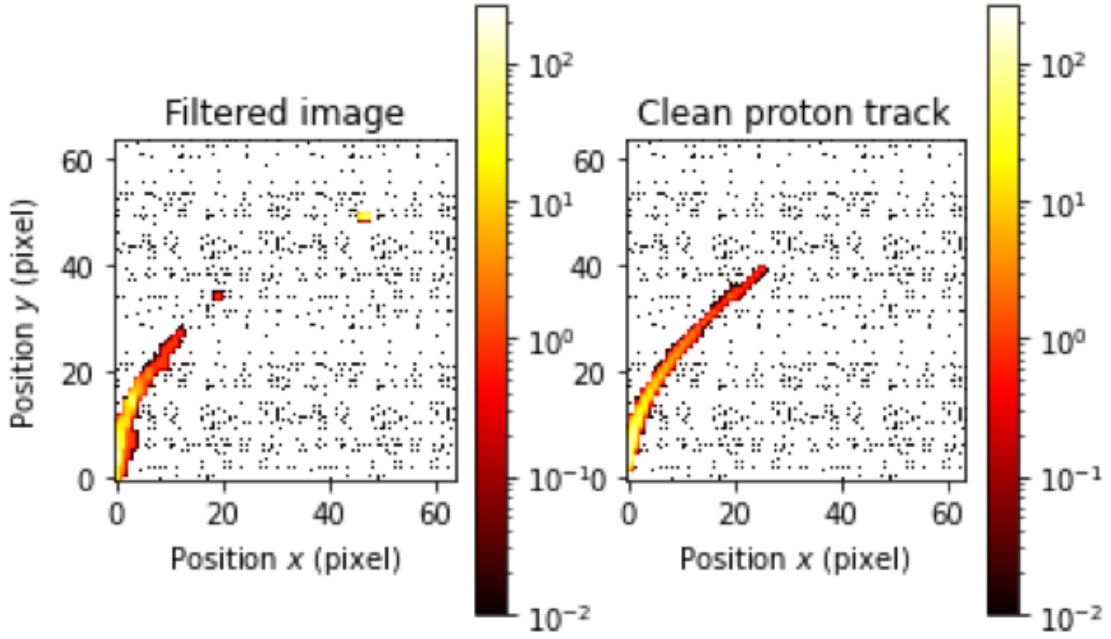


Figure 8: Manually filtered image and its corresponding proton track image. The accuracy of the traditional analysis depends heavily on how well the proton track is reconstructed.

4.2 Thermal Distribution Reconstruction

The original thermal distribution of the proton beam is needed to make measurements. In Sec 2.1.1, we discussed the amount of deflection experienced by a particle and the implications on its energy. Specifically, assuming uniform electric and magnetic fields, the deflection at any point in x and y is given by Eq (2) and (3). Thus, our first step is to convert the pixel edges to energy values. We have chosen to transform rows (corresponding to Eq (3)) instead of columns because a much larger energy range can be covered in y . After that, counts on the same row are summed up. Knowing the total count and energy interval in every row, we can generate a number / probability density plot against energy (see Fig 9). This is the reconstructed thermal distribution of the protons. Here, we estimate the mean energy of a bin using its bin centre and the uncertainty from its bin edges. Due to the poor statistics of the low-energy bins (caused by the relatively low count and small energy interval), bins with $E < 0.1 \text{ MeV}$ were removed from the plot.

Equation (5) is fitted to the reconstructed thermal distribution to estimate a value for T directly. Finding N is also trivial, this is done by summing all the counts in an image. Finally, We estimate E_{max} by looking for the most energetic bin that is being occupied.

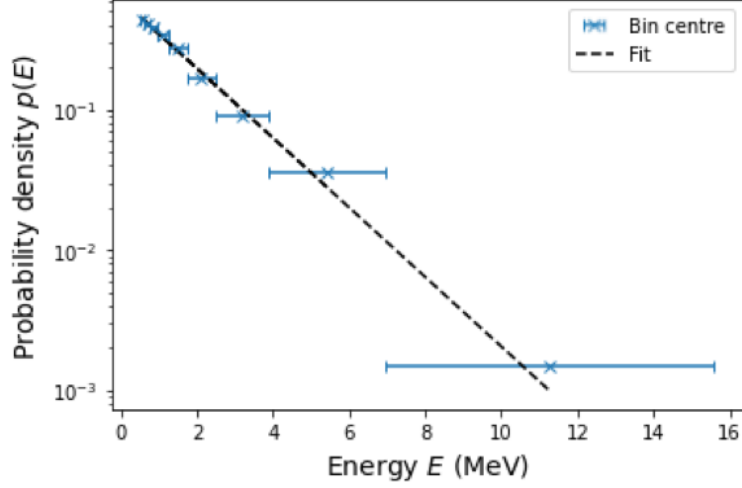


Figure 9: The reconstructed thermal distribution of the proton beam. Note that bins with $E < 0.1 \text{ MeV}$ are removed. Equation (5) is fitted to the thermal distribution to get T .

5 Flow of Evaluation

To summarise, an initial dataset containing 39475 images was synthesised using the developed simulation model. The dataset was doubled in size, totalling 78950 images, after which noise was added. The noisy data were then split into a training set, validation set and test set with a 72:8:20 ratio. The training set and validation set were used to train the optimised CNN, where predictions were made on the test set. On the other end, traditional analysis was performed on the same test set: first filtering the images, then reconstructing individual thermal distribution to measure the beam parameters. Finally, the results from each method were compared to the true beam parameters, evaluating the accuracy and feasibility of high-repetition-rate diagnostics.

6 Results

6.1 Accuracy of Measurements

6.1.1 Summary

Figures 10 (a) and 10 (b) provide a visual summary of the result accuracy for the machine learning and traditional analysis techniques across T , E_{max} and N . The y-axis corresponds to the absolute percentage error (APE) of the measurements relative to the true values. Machine learning is superior in all three variables judging from the apparent difference in the median (red line) and spread (interquartile range IQR) of the data. Specifically, the median is 5.2 times, 10.1 times and 12.1 times smaller respectively, and the IQR is 5.6 times, 3.3 times and 7.6 times narrower. Table 3 tabulates a more detailed summary of the raw values.

6.1.2 Asymmetry

Asymmetry is observed across the two techniques. The Bowley skewness B_{skew} (also known as the quartile skewness coefficient) is used to quantify our findings, it has the following form:

$$B_{skew} = \frac{Q_1 - 2Q_2 + Q_3}{Q_3 - Q_1}, \quad (10)$$

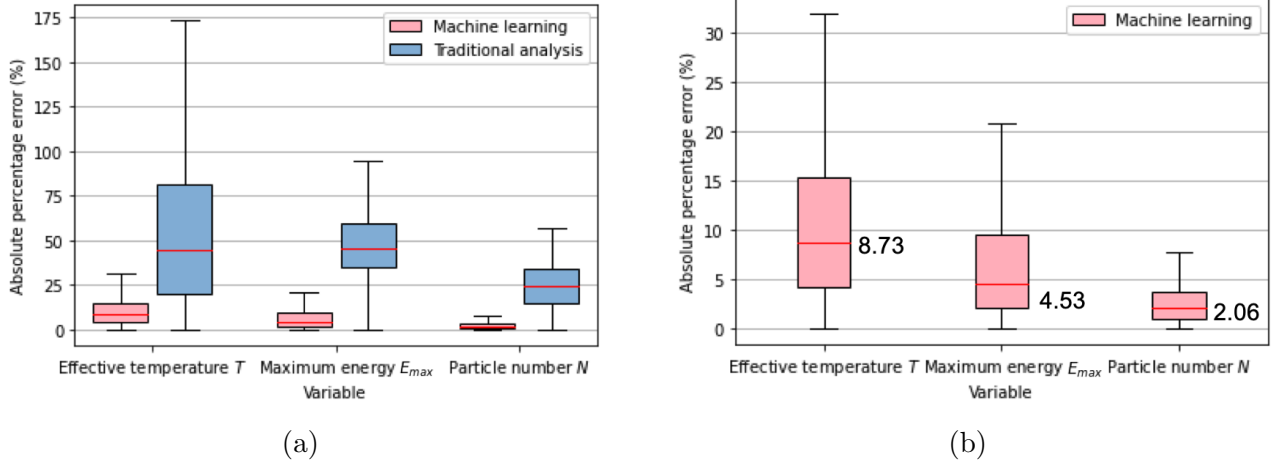


Figure 10: (a) APE of machine learning against traditional analysis. The median value is denoted by the red line inside each box. Machine learning demonstrates smaller median and IQR consistently. (b) APE of machine learning only. Similar to traditional analysis, T is the most difficult to predict, followed by E_{max} and N . Outliers are not shown here.

where Q_1 , Q_2 and Q_3 represent the first, second and third quartiles. Here, positive B_{skew} indicates a positively skewed distribution and vice versa for negative B_{skew} . The larger the magnitude, the stronger the skewness. From Tab 3, the majority of the variables exhibit positive skewness (apart from N by traditional analysis), meaning inaccurate measurements are more common. As with the median, this tendency is more prevailing with the energy quantities, T and E_{max} .

Table 3: Statistical measures used to describe the results of the machine learning and traditional analysis.

| Parameter | Technique | Median | IQR | B_{skew} | Anomalies |
|---------------------------|----------------------|--------|-------|------------|-----------|
| Effective temperature T | Machine learning | 8.54 | 10.97 | 0.18 | 112 |
| | Traditional analysis | 44.01 | 60.89 | 0.22 | 1339 |
| Maximum energy E_{max} | Machine learning | 4.53 | 7.49 | 0.32 | 10 |
| | Traditional analysis | 45.55 | 24.50 | 0.12 | 21 |
| Particle number N | Machine learning | 2.06 | 2.58 | 0.16 | 0 |
| | Traditional analysis | 24.93 | 19.64 | -0.06 | 0 |

6.1.3 Outliers

Conventionally, an outlier is classified as any data point that is located outside the whiskers (outside $Q_1 - 1.5 \text{ IQR}$ or $Q_3 + 1.5 \text{ IQR}$). Using this definition, we recorded 810, 1629 and 592 observations respectively for machine learning, and 223, 22 and 0 observations for traditional analysis. By comparing the number of outliers, we can easily be led to the conclusion that machine learning performs less consistently. However, bear in mind that an outlier here is defined to the IQR of a particular distribution, the number of outliers thus is not meaningful outside this context. To allow for a fair comparison, we instead define an outlier to be an observation that has an absolute percentage error of $> 100\%$. With the standard definition, the number of outliers from machine learning is restricted to only 112, 10 and 0, less than that from traditional analysis in all aspects (see Tab 3).

6.2 Speed of Measurements

6.2.1 Summary

The time required for machine learning and traditional analysis to evaluate the test set was measured on a local computer. Table 4 documents a detailed breakdown of the measurement time to evaluate a single image. Here, the measurement time and the associated uncertainty were estimated using the mean time and standard deviation of the population. An additional set of measurements was made for the 32x32 noisy images to examine the effect of image size on measurement time. A separate CNN optimised for the smaller image size was used. In short, the new model has a less complex structure, i.e. fewer number of kernels in the convolutional layers. From Tab 4, it is evident that machine learning once again outperforms traditional analysis, the prior is 3.0 times and 1.5 times faster at processing the 32x32 and 64x64 image sets.

Table 4: Breakdown of the measurement time of machine learning and traditional analysis. Measurements were made on a 2016 MacBook Pro (2.6 GHz Intel Core i7 and 16 GB RAM).

| Technique | Breakdown | Time (ms / image) | |
|----------------------|-----------------------------|-------------------|-------------------|
| | | 32x32 noisy image | 64x64 noisy image |
| Machine learning | Total | 0.36 ± 0.01 | 1.36 ± 0.06 |
| | Median filtering | 0.27 ± 0.01 | 0.88 ± 0.02 |
| Traditional analysis | Distribution reconstruction | 0.84 ± 0.02 | 1.18 ± 0.03 |
| | Total | 1.11 ± 0.03 | 2.06 ± 0.05 |

7 Discussions

As first demonstrated in Sec 6.1.3, the measurement accuracy is observed to be energy-dependent. Such behaviour is evident in Fig 10 (b), where the energy quantities T and E_{max} exhibit a larger median error and spread. To draw closer attention to the relationship between accuracy and energy, an error distribution was computed on a 2D histogram for all three variables (see Fig 11). Note the histograms were normalised between 0 and 1 so that the intensity of the colour is indicative of the likelihood of finding an observation.

7.1 Energy Dependency

For T and E_{max} , it is observed that the lower energy range is prone to anomalies. This effect is more outstanding for T as the lower bound approaches 0 *MeV*. This series of observations is found to be strongly correlated to the fact that the simulation results are inherently energy-constrained. An almost identical trend is displayed in Fig 4 (a) and the T distribution here. Recall Eq (5), that a lower T represents a larger decay constant, meaning lower energy particles are more likely to exist. As discussed in Sec 2.1.1, the slower particles experience larger deflections by the fields. As a result, the uncertainty resulting from the finite difference method is naturally magnified. This means that images with a low-energy profile contain higher-than-normal fluctuations and are more difficult for the CNN to interpret.

On the other hand, the N distribution is free of extreme values and more uniform. This is unsurprising because N is independent of energy. However, such behaviour is later realised to be an oversimplification of the system. In the simulation, an arbitrary constant is used as

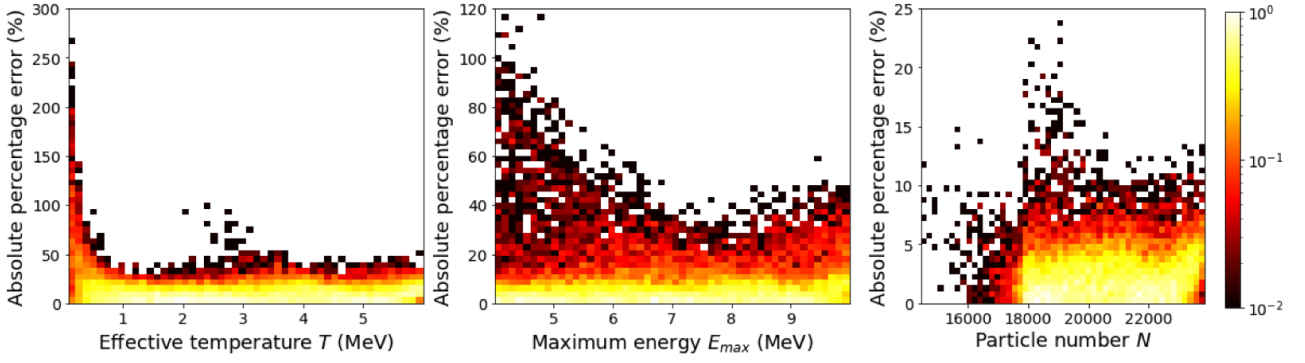


Figure 11: Error distributions of all three beam parameters. Both T and E_{max} show a surge of anomalies at low energy. On the other hand, a relatively uniform distribution is seen with N .

the particle weight which acts to mimic the conversion of particle count to intensity. A similar quantity in real life is the quantum efficiency of a CCD, which measures the ratio between the number of photoelectrons collected and the number of photons incident on the CCD surface. This quantity is, however, energy-dependent. For a standard CCD, the spectral sensitivity is highest between 550 nm and 900 nm [9]. Note in an experiment, the conversion actually happens in two steps: first from ions to photons in the scintillator and later from photons to photoelectrons in the CCD. The efficiency of both processes should be measured to compute a final scaling function.

7.2 Limitations

7.2.1 Traditional Analysis

The poor measurement accuracy of the traditional analysis can be attributed partly to the artificial filtering technique. First, it tends to remove image details which have weak signals. This is evident in Fig 8, where the tail of the proton track is broken up and almost eliminated. Second, it is incapable of distinguishing signal and noise that share a similar magnitude. An obvious example is the carbon residue close to the bottom of the proton track. The combining effect leads to the discrepancy in N .

The heavy reliance on the reconstruction of the thermal distribution is another key factor. Despite using an experimental magnetic field to synthesise the data, we continued to use Eq (3) to calculate energies from the coordinates, which assumes the use of uniform fields, to begin with. This results in fundamentally different distributions, causing inaccurate fits and estimations of T . As a project extension, a new coordinate-energy mapping function can be determined by measuring the coordinates of particles over a range of energies.

A majority of the low-energy pixels are never occupied. The poor statistics prompted the use of energy threshold (see Sec 4.2) and caused around 36 data points to be removed on average, further worsening the fits. On the other end, high-energy pixels also have their shortcomings. They suffer from large energy bin widths, so estimating E_{max} using bin centres is very unreliable. For instance, we can end up with energy values which are outside the synthesis range.

7.2.2 Machine Learning

It is important to realise that machine learning also has its ceiling. The same analysis was performed on the original test set (without noise) to obtain a benchmark. From Fig 12, the median and IQR of the error is seen to roughly halve. At the same time, this marks the limitation of the current model.

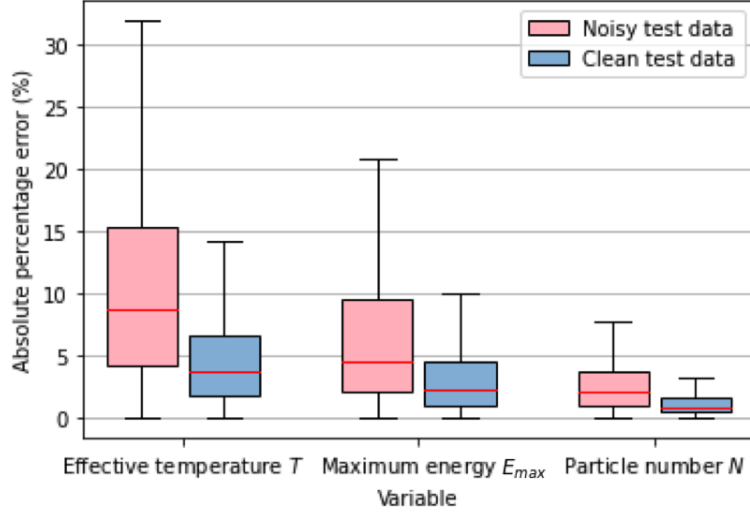


Figure 12: APE of noisy test data against clean test data. The median and IQR roughly halve.

7.3 Real-time Measurements

A major objective of this project is to accomplish real-time measurements of the beam parameters. For a 100 Hz laser, it means that an image needs to be analysed under 10 ms . This goal has been more than achieved with the 32x32 and 64x64 image sets. A potential n^2 dependence on the image size is observed in the machine learning prediction: the measurement speed on the 64x64 image set is 3.8 times slower than the 32x32 image set. This means that achieving real-time measurements on real experimental images is impossible on a local computer: $(1024/32)^2 \cdot 0.36 \approx 370$ ms . However, the situation can be overturned by analysing the images on a high performance computing cluster. Optionally, we can make a trade-off between accuracy and speed by reducing the model’s complexity.

7.4 Practical implementation

7.4.1 What’s Missing

The current model is not currently ready for use in an experiment. First, the CNN is only reliable in predicting the variable ranges that it has learned. Outside the “seen” domain, we might risk having large errors. Second, the model is specific to the experimental setup. This means that it is not easily adaptable to changes in the environment. If this does happen, we will need to adjust the simulation model and synthesise a new image set to train the CNN. Fortunately, the above problems have been accounted for since the start of the project, in that the entire setup, image configuration and noise are free to parameterise.

Unrealistic signals / noise is another major issue. Figure 13 shows an example of a real experimental image. We can see that the zero point error occupies not only a single pixel but spans over a large area. Paying closer attention to the image details, streak noise (highlighted in the red box) is also visible. The long streaks arise due to the movement of charges during a readout and are often associated with overloaded pixels. In addition, our images are synthesised from set beam parameters and noise. To fully recreate the experimental results, we shall consider modelling the system from the laser source and numerically simulating the laser-matter interactions. This can be done by extending our developed algorithm or potentially using VSim, a plasma-accelerator simulation software [10].

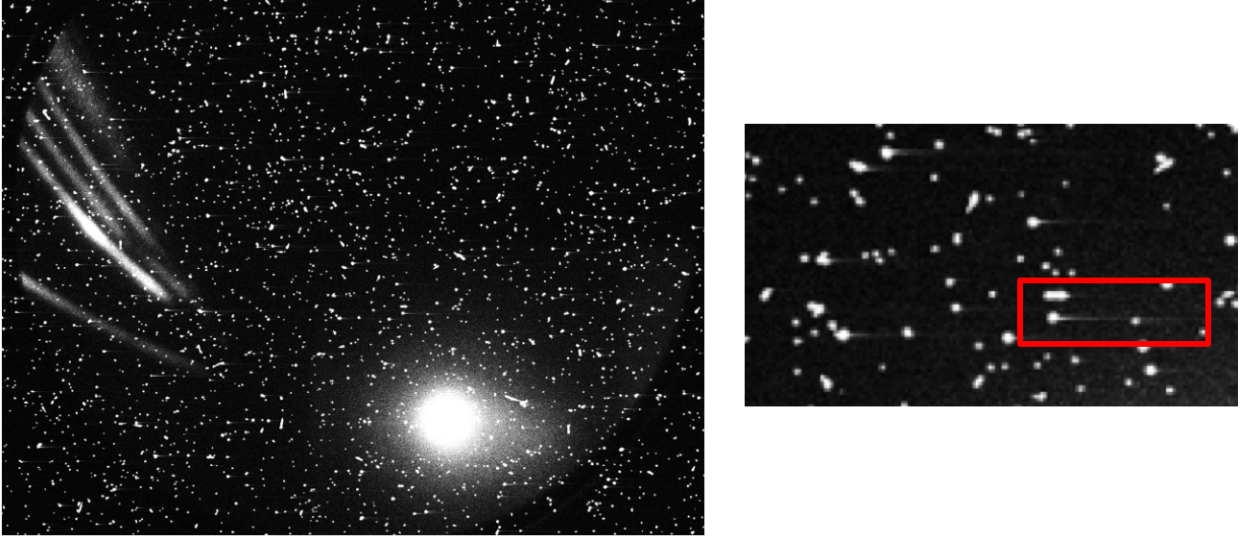


Figure 13: A section of the 1024x1024 experimental image. The larger image size means more details to the signals and noise. The appearance of streak noise is highlighted in the red box (right).

7.4.2 Future Extensions

Other modifications are planned to complete the data analysis pipeline. A good starting point will be to experiment with larger image sizes. This will allow us to verify the potential n^2 dependence of the measurement time and conclude whether real-time measurements are feasible. At the same time, we can compare the prediction accuracy amongst a range of image sizes and identify the optimal one to be put in place in an experiment. Data augmentation, such as random rotation and shifts, can be incorporated to expand the existing dataset. The purpose of this technique is to recreate the random / systematic variations in a real-life setup, which may be caused by the unintended tilting or thermal effects of the imaging device. Overall, it can reduce overfitting when training the CNN.

Shortly before the project was finished, we trialled a neural network-based image denoising method, known as the convolutional denoising autoencoder (CDAE) [3]. The CDAE works broadly on the same principle as the CNN, apart from the fact that the model is trained using noisy images \mathbf{X} and corresponding noise-free images \mathbf{X}' . A CDAE is advantageous because it presents an automated image denoising process, which allows instant track inspection during an experiment. The results of our preliminary investigation are summarised in Tab 5. The CDAE outperforms median filtering by 5.1 times in PSNR and has achieved an SSIM score much closer to 1.

Table 5: PSNR and SSIM scores of the CDAE and median filtering.

| Metrics | Median filtering | CDAE |
|--|------------------|--------|
| Peak signal-to-noise ratio (PSNR) | 17.3 | 88.9 |
| Structural similarity index measure (SSIM) | 0.9639 | 0.9998 |

8 Conclusion

The use of machine learning method, specifically a convolutional neural network (CNN), is shown to be effective in measuring the spectral beam parameters of a laser-driven ion source. A simulation model based on the Thomson parabola spectrometer was designed to synthetically generate training and test data for the machine learning model. Comparisons of measurement accuracy and speed were made between the CNN and conventional analysis techniques. The final results showed that the CNN is 5 times, 10 times and 12 times more accurate in predicting the effective temperature T , maximum energy E_{max} and particle count N on a 64x64 dataset. With regards to measurement speed, the same measurements were executed 1.5 times faster on a local computer. Real-time output proves to be possible for small image sizes, to fulfil the same standard with an experimental image (1024x1024), the use of high performance computing and other compromises on prediction accuracy may be required. Future work is needed to validate the model using larger image sizes and more realistic noise.

References

- [1] References Albert, F., Couprie, M., Debus, A., Downer, M., Faure, J., Flacco, A., Gizzi, L., Grismayer, T., Huebl, A., Joshi, C., Labat, M., Leemans, W., Maier, A., Mangles, S., Mason, P., Mathieu, F., Muggli, P., Nishiuchi, M., Osterhoff, J., Rajeev, P., Schramm, U., Schreiber, J., Thomas, A., Vay, J., Vranic, M. and Zeil, K., 2021. 2020 roadmap on plasma accelerators. *New Journal of Physics*, 23(3), p.031101.
- [2] Harres, K., Schollmeier, M., Brambrink, E., Audebert, P., Blazevic, A., Flippo, K., Gautier, D., Geißel, M., Hegelich, B., Nurnberg, F., Schreiber, J., Wahl, H. and Roth, M., 2008. Development and calibration of a Thomson parabola with microchannel plate for the detection of laser-accelerated MeV ions. *Review of Scientific Instruments*, 79(9), p.093306.
- [3] Miyatake, T., Shiokawa, K., Sakaki, H., Dover, N., Nishiuchi, M., Lowe, H., Kondo, K., Kon, A., Kando, M., Kondo, K. and Watanabe, Y., 2021. Denoising application for electron spectrometer in laser-driven ion acceleration using a Simulation-supervised Learning based CDAE. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 999, p.165227.
- [4] Gondara, L., 2016. Medical Image Denoising Using Convolutional Denoising Autoencoders. 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW),.
- [5] Lee, D., Choi, S. and Kim, H., 2018. Performance evaluation of image denoising developed using convolutional denoising autoencoders in chest radiography. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 884, pp.97-104.
- [6] Hoffmeister, G., 2014. Influencing laser-accelerated ions by femtosecond-laser desorption.
- [7] n.d. circular_magnet.mat.
- [8] Yamashita, R., Nishio, M., Do, R. and Togashi, K., 2018. Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, 9(4), pp.611-629.
- [9] Spring, K. and Davidson, M., n.d. Hamamatsu Learning Center: Quantum Efficiency. [online] Hamamatsu.magnet.fsu.edu. [Accessed 6 May 2022].
- [10] Hanahoe, K., 2017. Simulation studies of plasma wakefield acceleration.

Acknowledgements

I would like to thank my project partner for his continuous devotion to this project. I would also like to thank our supervisor, Nicholas Dover, for his help at all stages and time spared to discuss interesting Physics with us.