Machine Learning-Driven Process of Alumina Ceramics Laser Machining

Razyeh Behbahani[1,2], Hamidreza Yazdani Sarvestani[3], Erfan Fatehi[4], Elham Kiani[2,5], Behnam Ashrafi[3*], Mikko Karttunen[1,2,6*] and Meysam Rahmat[7]

[1]Department of Physics and Astronomy, The University of Western Ontario, 1151 Richmond Street, London, ON N6A 3K7 Canada.

What is the problem?

Laser parameters :
- Beam frequency
- Beam amplitude
- Number of passes
- Linear speed
- Focal position

Engraved channel's depth
- top width
- bottom width

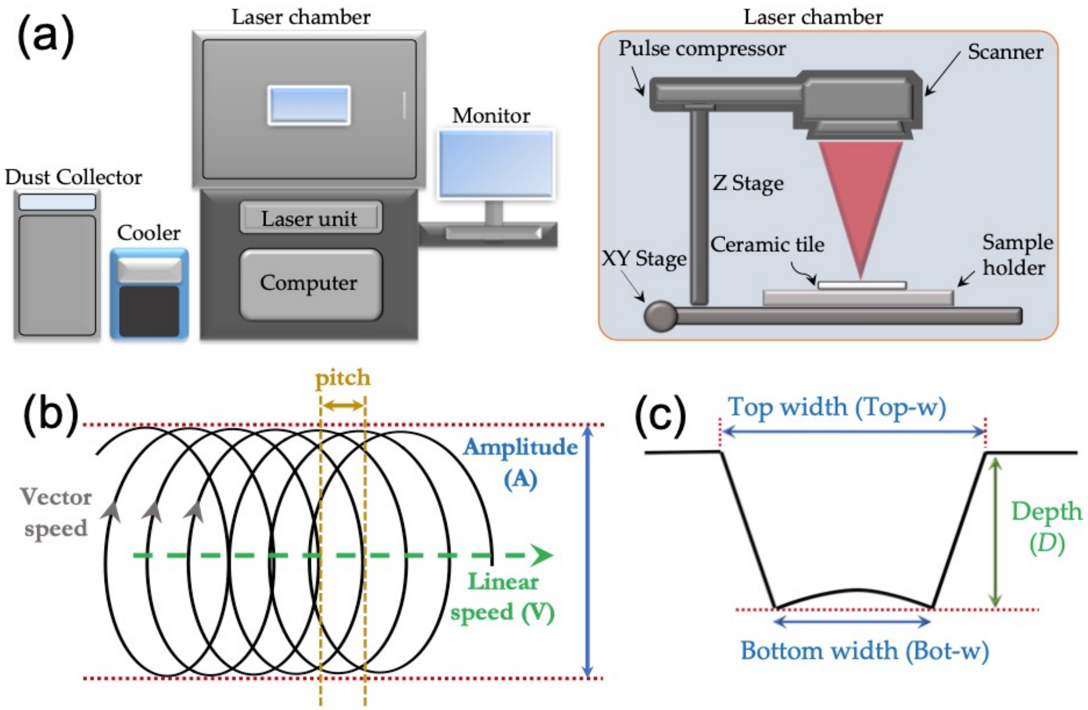
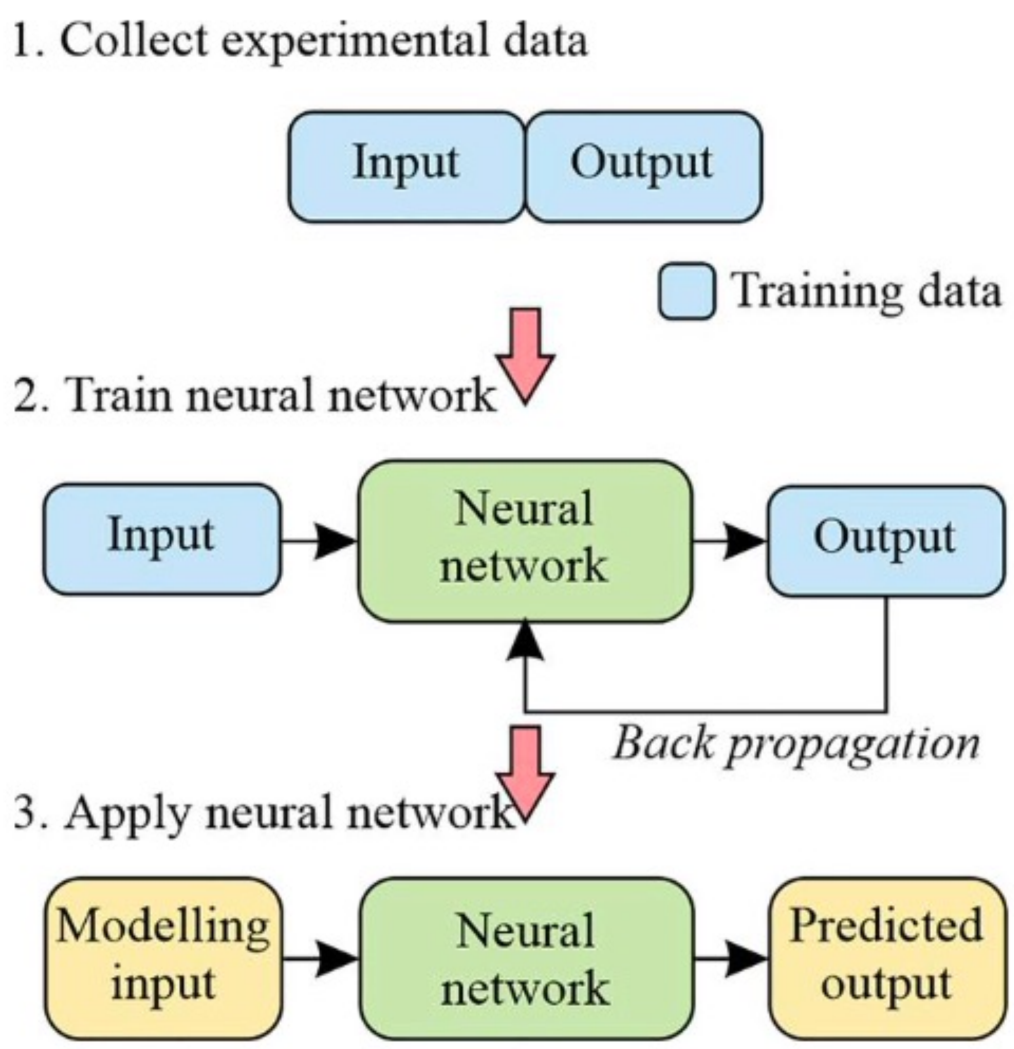


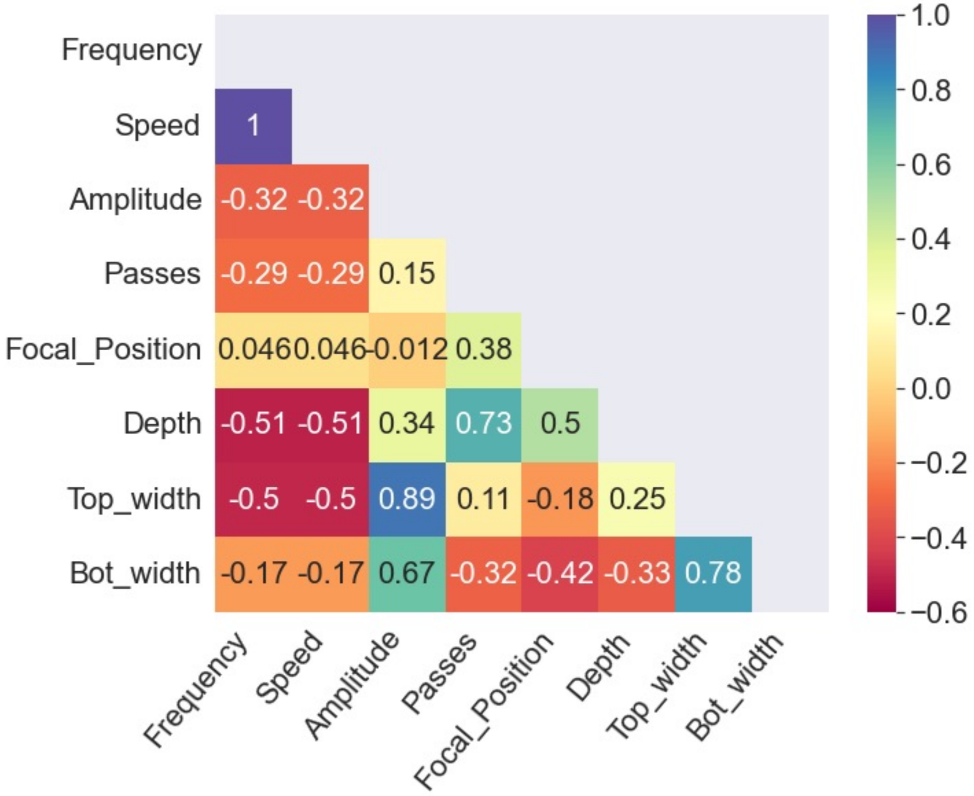
Fig. 1 (a) Schematic of the laser system and the equipment. (b) Schematic of the circular wobble pattern illustrating the laser pulses and direction, wobble amplitude, wobble pitch, linear speed, and vector speed. (c) Schematic of a trapezoidal cut. The channel cross-section is triangular when the bottom width is negligible.

Application of Machine Learning (Neural Networks here) for laser machining:



Stage one: collecting data

- Determine the inputs and outputs
- Clean the data, decide about the missing values, outliers and duplicates
- Look at the correlation between parameters



Speed = Frequency / 40

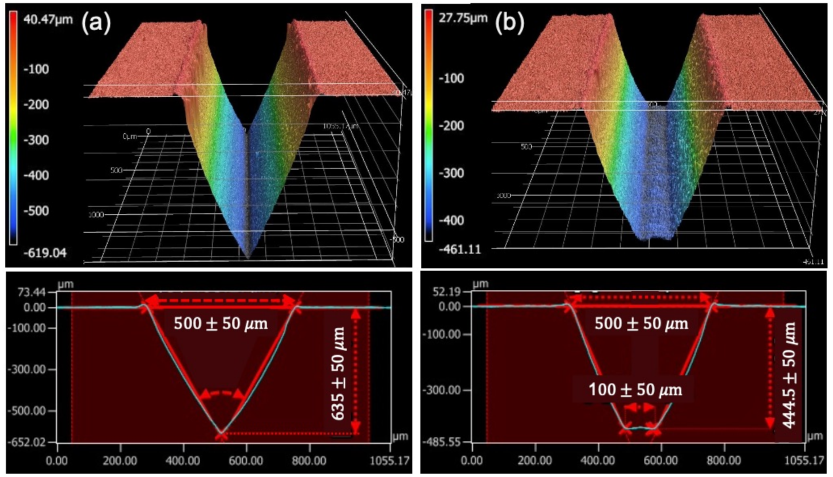


Fig. 11 Three and two dimensional representations of typical engraved channels with target dimensions printed having (a) triangular and (b) trapezoidal cross sections.

Frequency	Speed	Amplitude	Passes	Focal_Position	Depth
800.0	20.0	0.5	1.0	93.2	49.81
800.0	20.0	0.5	1.0	93.2	46.99
800.0	20.0	0.5	1.0	93.2	46.77
800.0	20.0	0.5	1.0	93.2	49.97
800.0	20.0	0.5	1.0	93.2	51.65
800.0	20.0	0.5	1.0	93.2	55.29

Frequency	Speed	Amplitude	Passes	Focal_Position	Depth
800.0	20.0	0.4	6.0	93.2	268.01
800.0	20.0	0.4	6.0	93.2	240.62
800.0	20.0	0.4	6.0	93.2	235.33

Frequency	Speed	Amplitude	Passes	Focal_Position	Depth
800.0	20.0	0.2	40.0	93.2	438.11
800.0	20.0	0.2	40.0	93.2	507.12
800.0	20.0	0.2	40.0	93.2	487.41

Frequency	Speed	Amplitude	Passes	Focal_Position	Depth
400.0	10.0	0.5	6.0	93.2	297.21
400.0	10.0	0.5	6.0	93.2	263.03
400.0	10.0	0.5	6.0	93.2	261.80

$$\text{Exp\_error} = \frac{|\,\text{min\_depth} - \text{max\_depth}\,|}{\text{ave\_depth}}$$

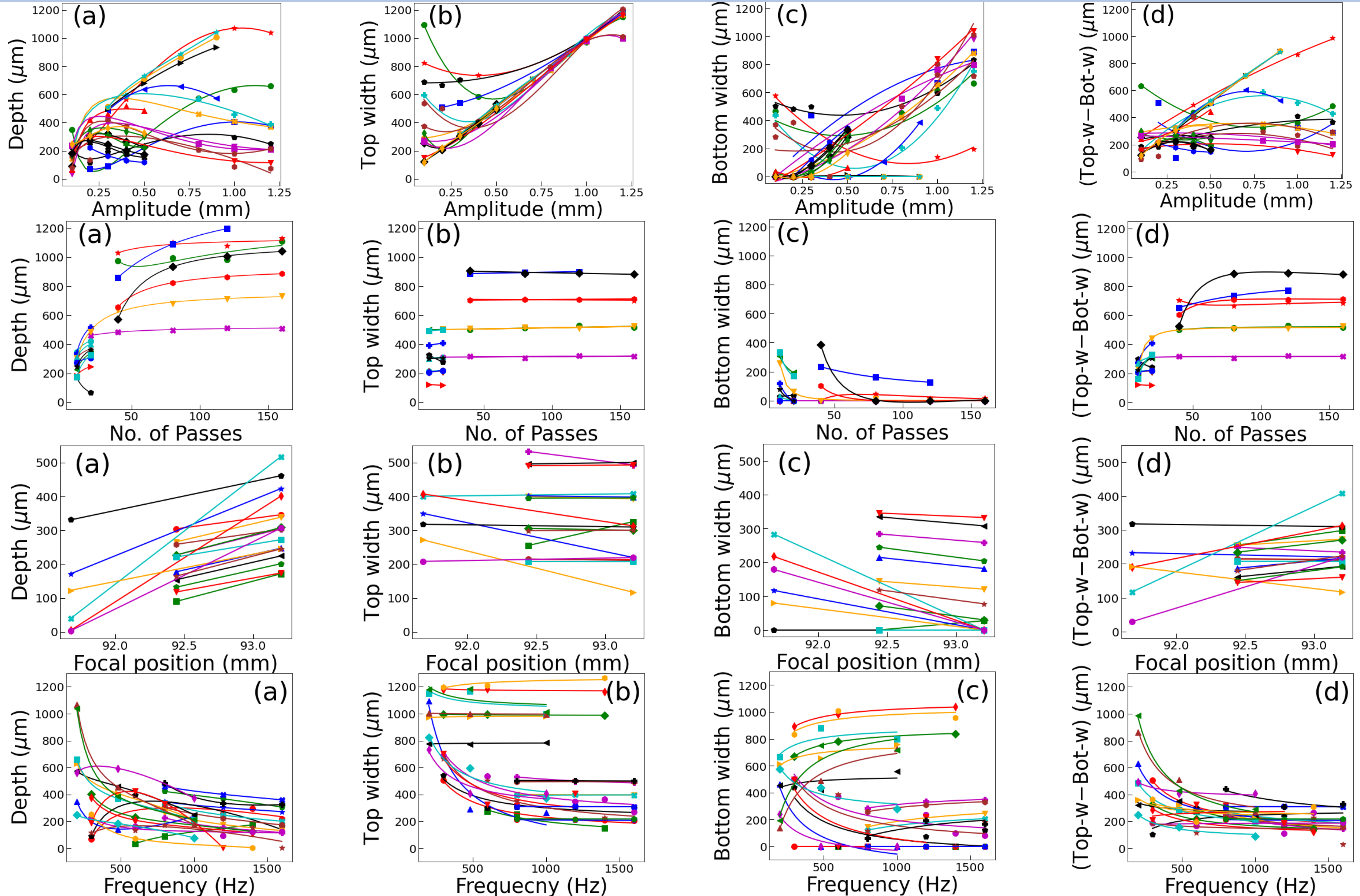
18%

13%

14%

12.9%

Stage one: Interconnection between parameters



Stage two: Model selection

Preprocessing technique:

To be able to perform comparisons, we renormalize and non-dimensionalize the parameter values to the range [0, 1]. For example, feature X in the range $[X_{\min}, X_{\max}]$ is scaled using

$$X_{\text{standard}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}. \quad (5)$$

After evaluating model performances, predictions can be scaled back to their original range.

Model Evaluation

$$R^2 = 1 - \frac{\text{RSS}}{\text{TSS}}, \text{ where RSS} = \sum_i (y_i - y_i')^2 \text{ and TSS} = \sum_i \left(y_i - \frac{\sum_i y_i}{n} \right)^2. \quad (4)$$

$$\text{MSE} = \frac{\sum_i (y_i - y_i')^2}{n}.$$

When the predicted values are shown with y_i'

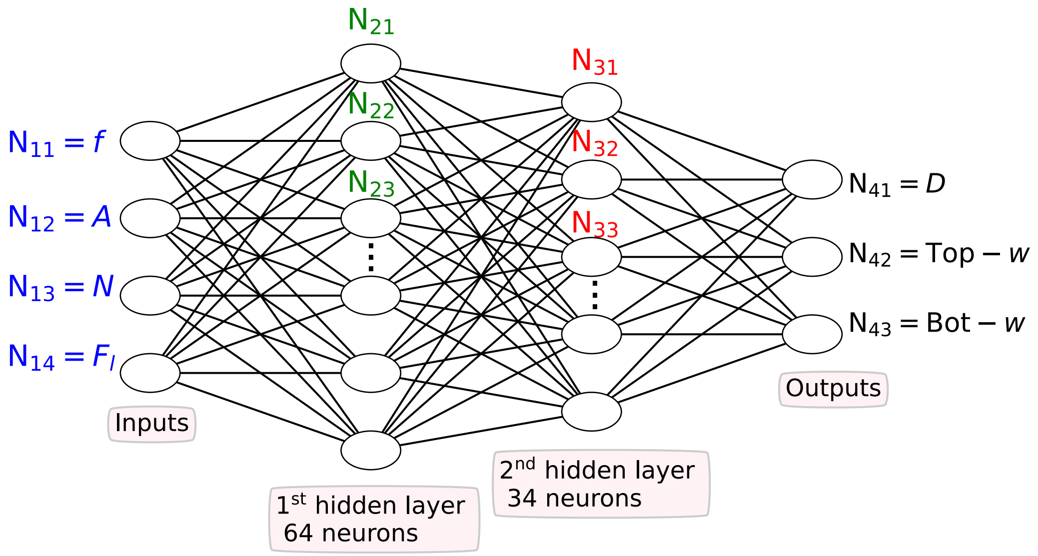
Stage two: Model selection

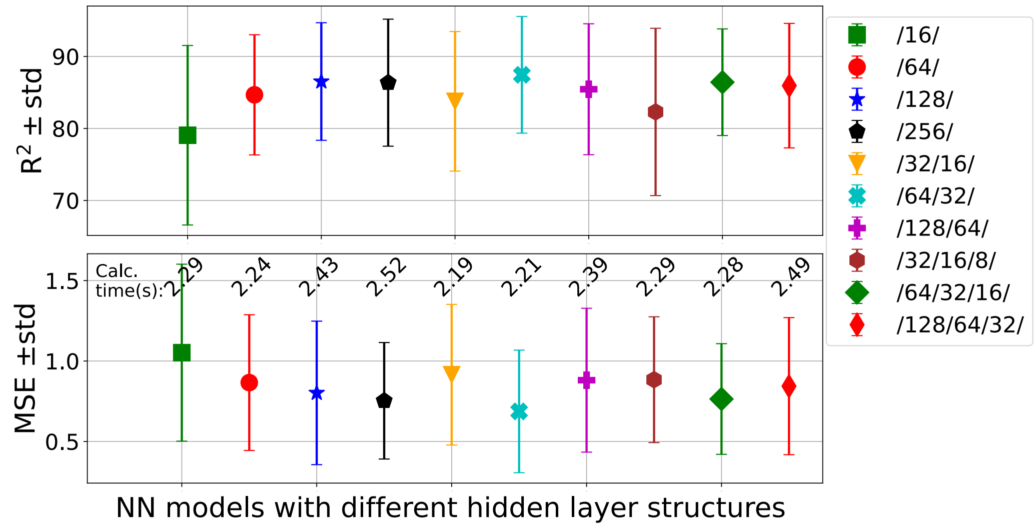
Linear regression:

$$
\begin{aligned}
\text{LinearReg.}(Y) &= w_1X_1 + w_2X_2 + w_3X_3 + w_4X_1X_2 + w_5X_1X_3 + w_6X_2X_3 + b \\
\mathcal{O}(2)\text{Poly.Reg.}(Y) &= \text{LinearReg.}(Y) + w_7X_1^2 + w_8X_2^2 + w_9X_3^2 \\
\mathcal{O}(3)\text{Poly.Reg.}(Y) &= \mathcal{O}(2)\text{Poly.Reg.}(Y) + w_{10}X_1^3 + w_{11}X_2^3 + w_{12}X_3^3,
\end{aligned}
\tag{1}
$$

XGBoosting
Neural Networks

$$
\begin{aligned}
N_{lk} &= f(z_l) \\
z_l &= \sum_m w_m N_{l-1m} + b,
\end{aligned}
\tag{2}
$$



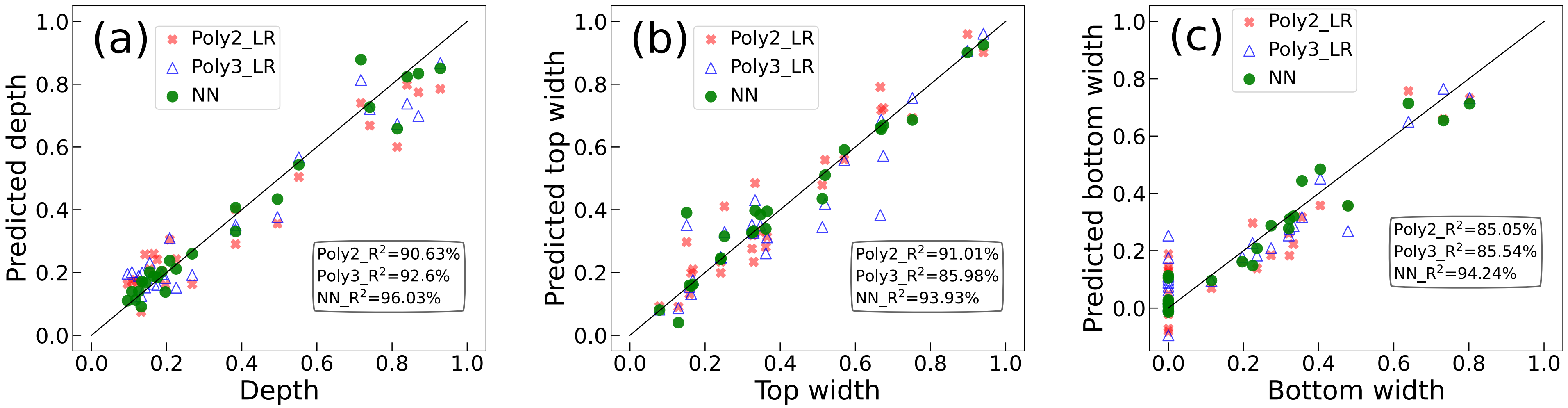


Stage two: Model selection

- 124 experimental data sets (observations).
- Cross-validation to avoid over fitting.
- Bootstrapping to check robustness of the models for different train-test splits.

ML model	10-fold cv-MSE ×100	test-MSE ×100	Bootstrapping MSE×100, R^2	calc. time (s)
Linear Regression (LR)	2.007	1.982	1.925, 67.60%	0.003
2^{nd} order Poly.R.	1.136	1.479	1.111, 81.59%	0.004
3^{rd} order Poly.R.	0.676	1.769	1.096, 80.80%	0.009
4^{th} order Poly.R.	3.624	3.511	5.078 , 21.02%	0.022
XGBoosting	1.559	1.164	1.418, 75.74%	0.084
Neural Networks	-	-	0.687, 87.44 %	2.213

Table 2 Performance of the different ML algorithms for predicting the channel's depth.



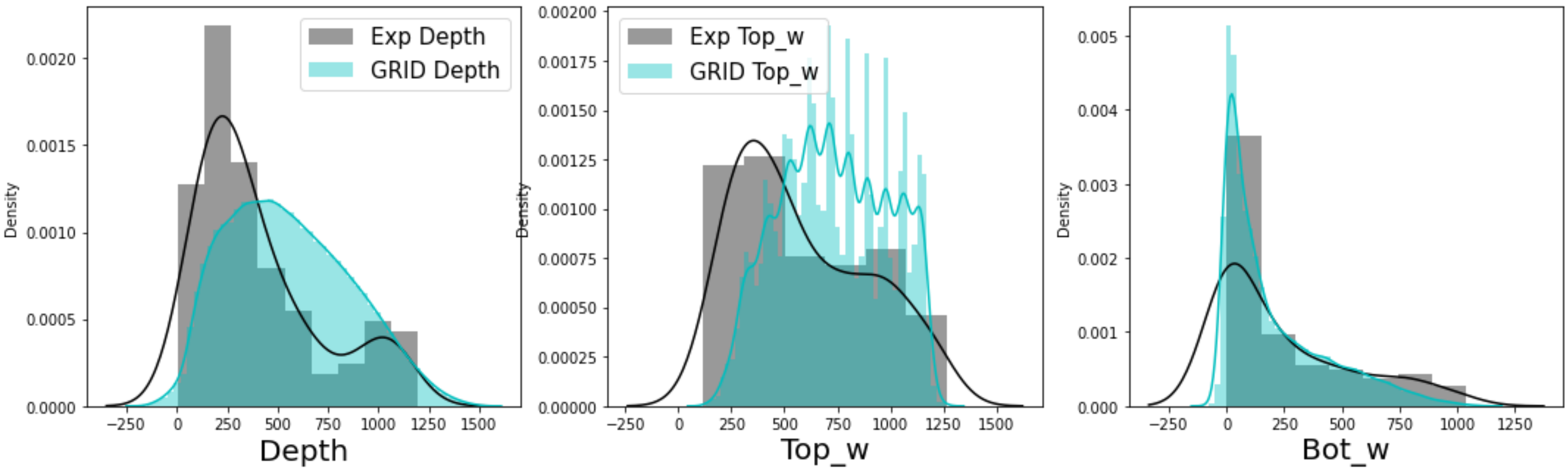
Stage three: Making unseen laser combinations to feed to the trained model

```
Used frequencies are :
 [ 150.   200.   250.   267.   300.   320.   400.   500.   533.   600.   800. 1000.
 1200. 1400. 1600.]

Used speeds are :
 [ 3.   4.   4.   5.   6.   7.   8.   8.   9. 10. 10. 12. 14. 15. 16. 18. 18. 20.
 21. 24. 25. 28. 30. 32. 35. 40.]

Used Amplitudes are :
 [0.3 0.5 0.7 1.   1.3 1.5 2. ]

Used no. of passes are :
 [ 20   40   60   80 100 120]

Used focal positions are :
 [93.2   93.8   93.87 94.4   94.71 95.    95.6 ]
```

```python
DATA = []
for f in freq:
    for s in speed:
        for a in Ampl:
            for np in no_pass:
                for p in pos:
                    data_point = [f, s, a, np, p]
                    DATA.append(data_point)
```

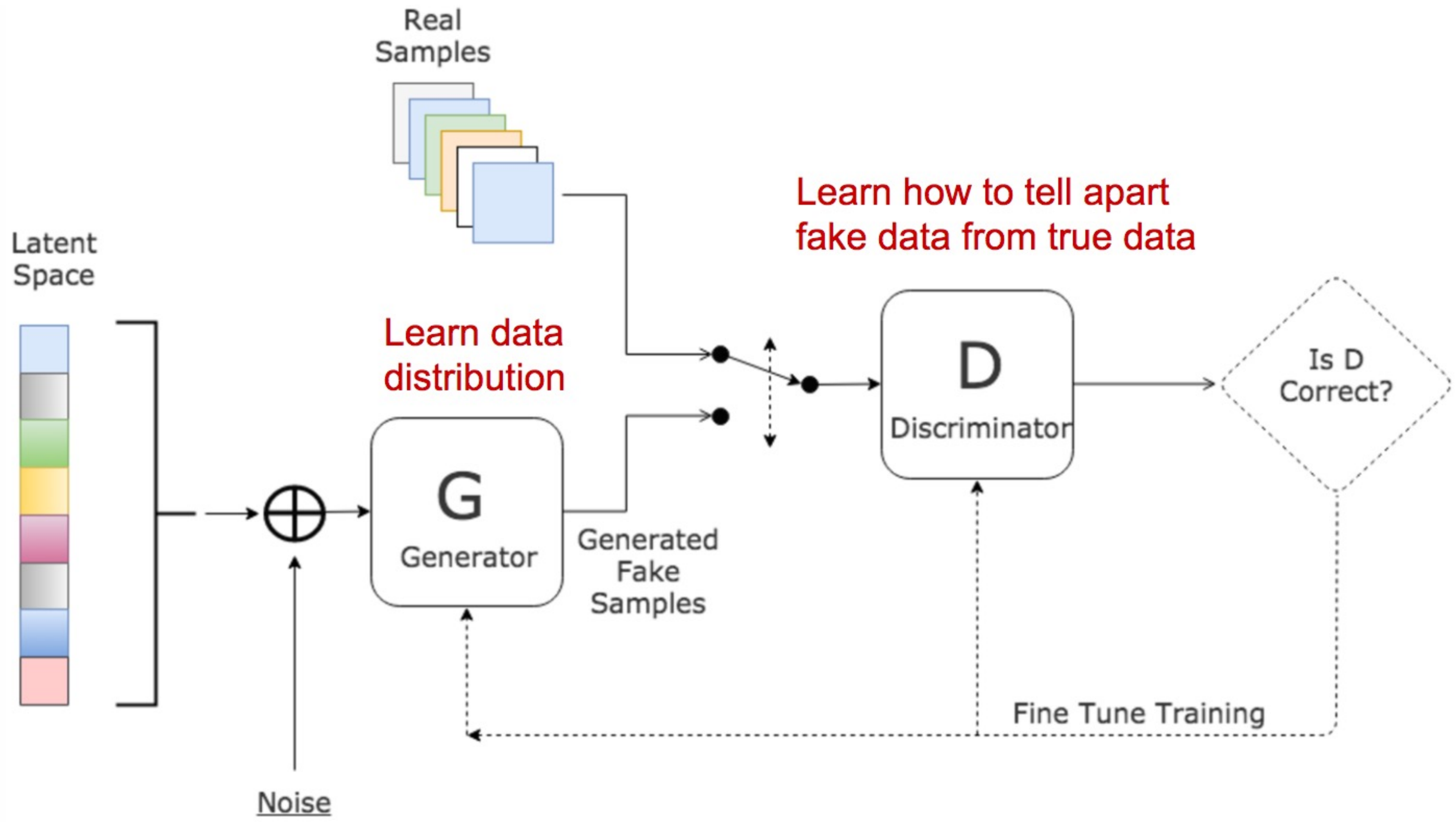
Data frame with 114,660 data points made of all the used values in the experiments

Stage three: Making unseen laser combinations to feed to the trained model

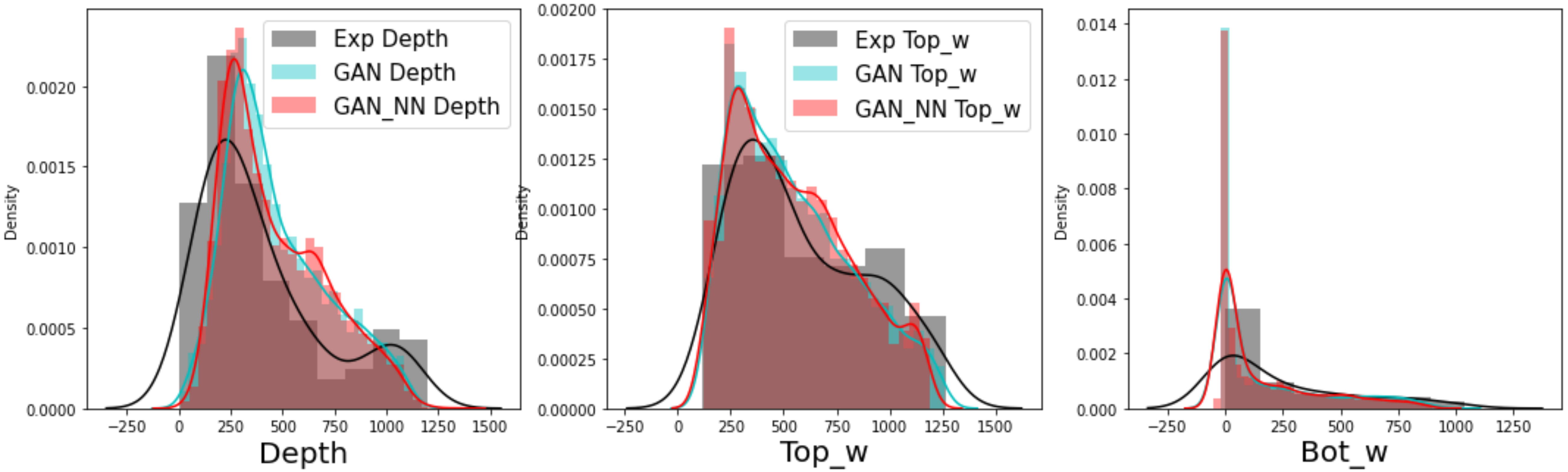


Stage three: Making unseen laser combinations to feed to the trained model

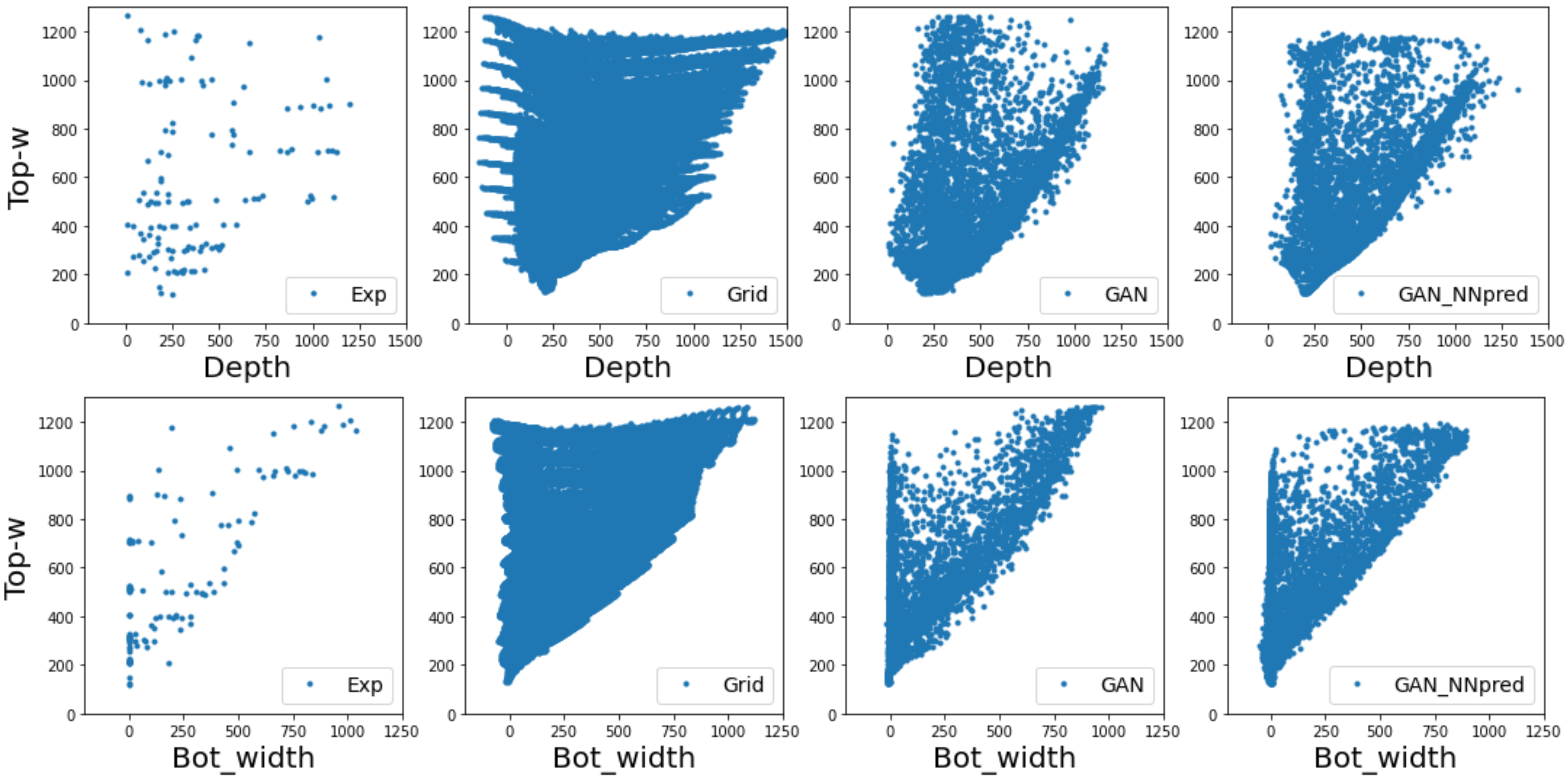
Generative Adversarial Networks (GAN):



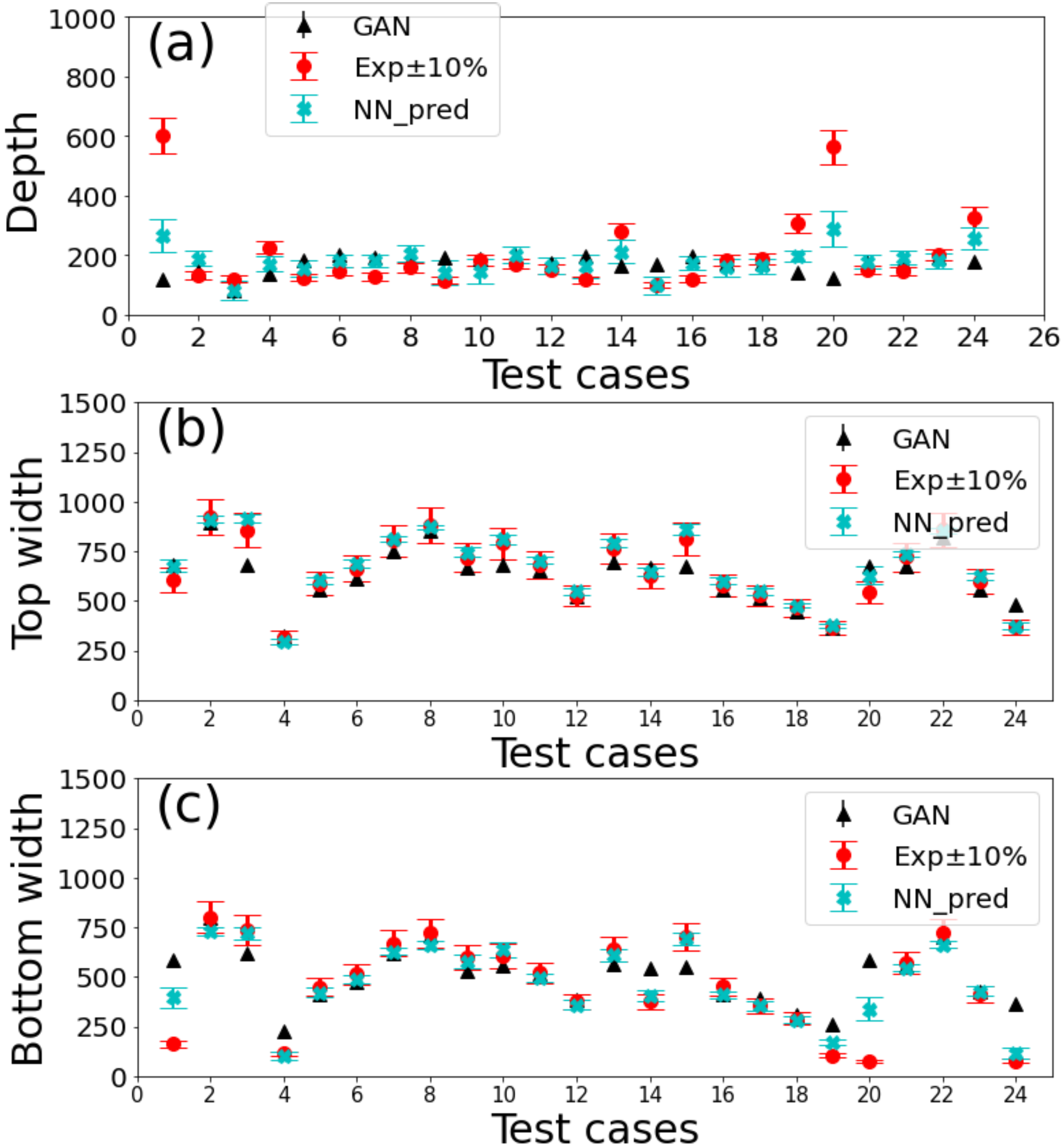
Stage three: Making unseen laser combinations to feed to the trained model



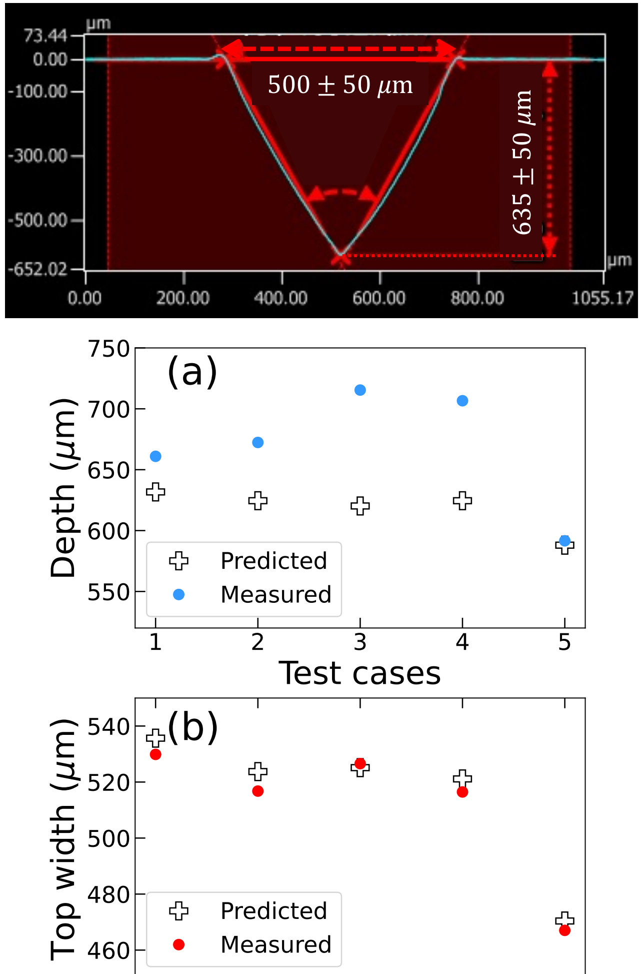
Comparing the grid and GAN data with the experiments



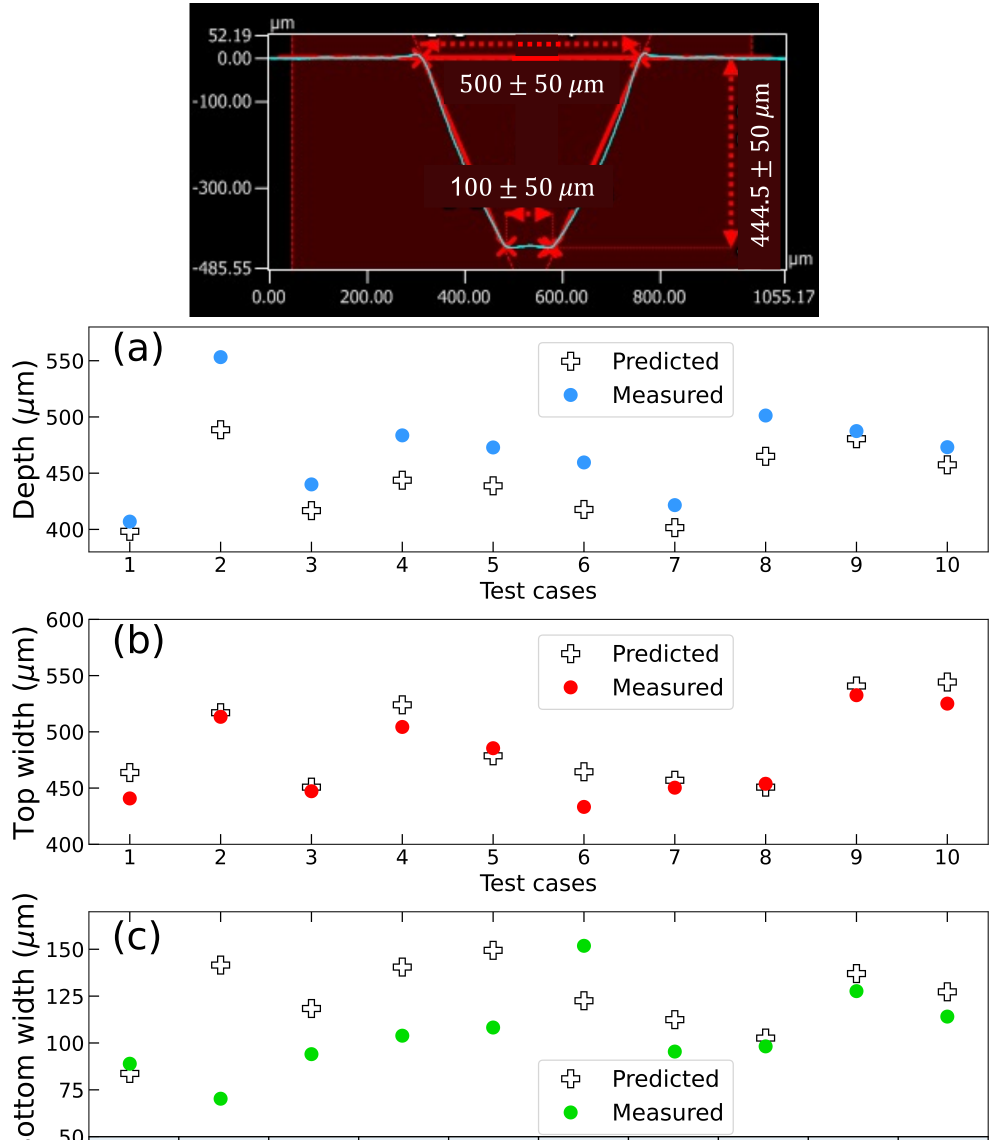
Stage three: Feed unseen data to the trained model



Stage three: Prescribed laser parameter combinations for engraving target channels



	#1	#2	#3	#4	#5
V (mm/s)	21.2	25.2	22.0	14.1	21.1
f (Hz)	847	1007	879	563	842
A (mm)	0.531	0.504	0.519	0.513	0.469
N	38	77	38	33	32
$\mathbf{F}_l$ (mm)	93.2	93.2	93.2	92.67	93.2



	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
V (mm/s)	39.7	15.1	20.0	23.1	12.9	9.0	26.2	11.2	20.2	27.4
f (Hz)	1587	604	800	925	516	362	1048	450	809	1097
A (mm)	0.459	0.53	0.455	0.519	0.49	0.437	0.451	0.458	0.543	0.532
N	29	18	15	22	14	14	20	16	22	28
$\mathbf{F}_l$ (mm)	93.2	93.17	93.19	93.2	93.06	93.05	93.2	92.98	93.2	93.2