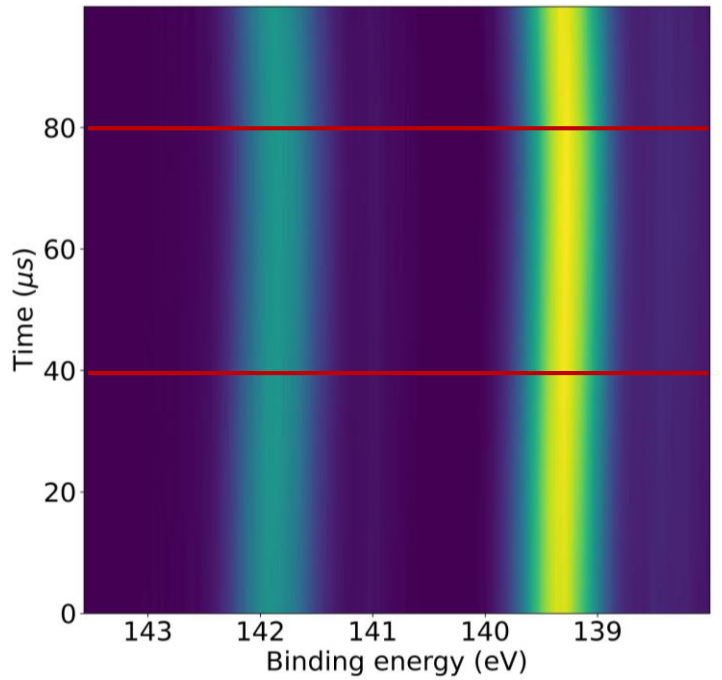
Machine learning techniques for high-throughout analyses of TR-PES measurements

Convolutional Neural Networks employed to fit XPS spectra

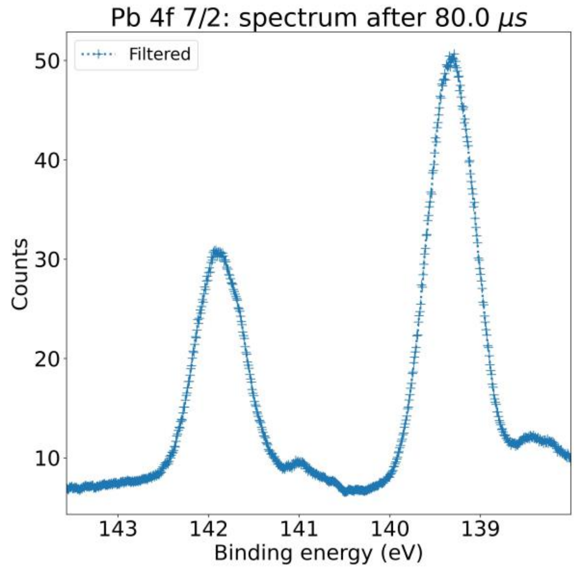
Arthur Julien, Jean-Baptiste Puel, Philip Schulz

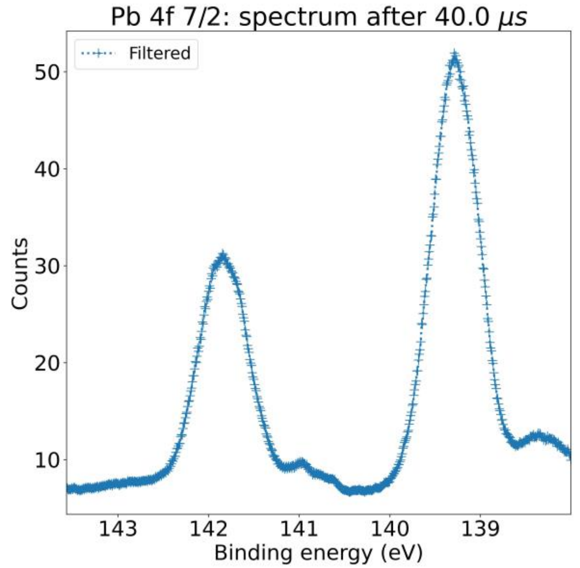
- 18/03/2024

Context: time resolved XPS measurements

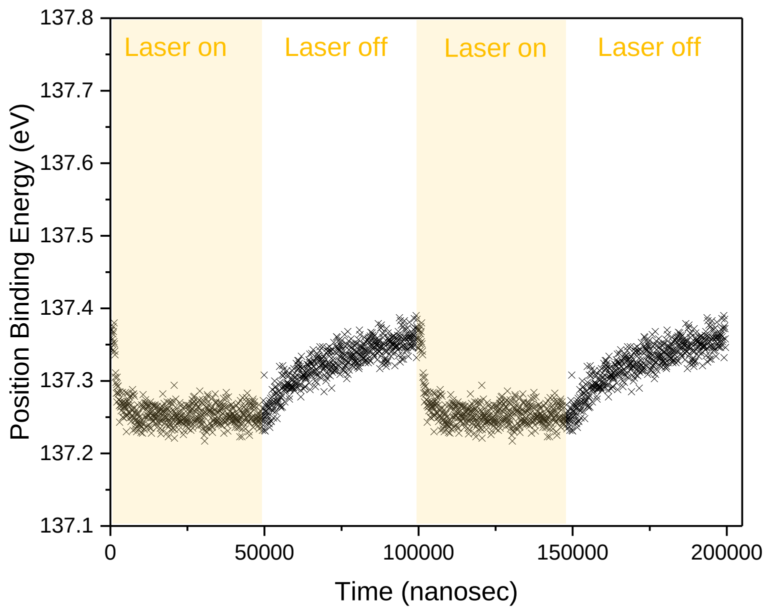


1000 XPS spectra
to analyze





Objective of the experiment: quantify changes of material properties upon illumination

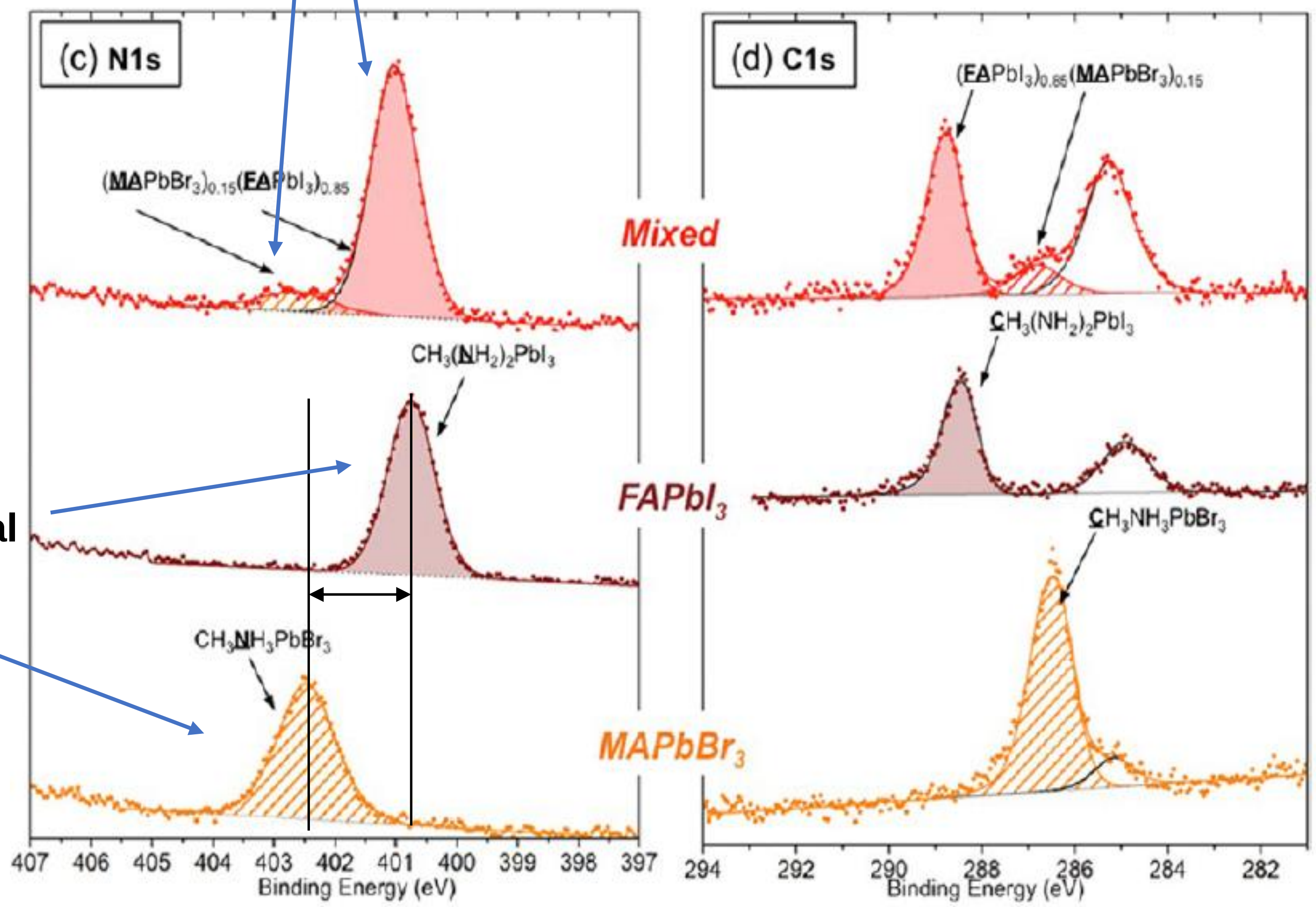


High throughput analysis methods are necessary

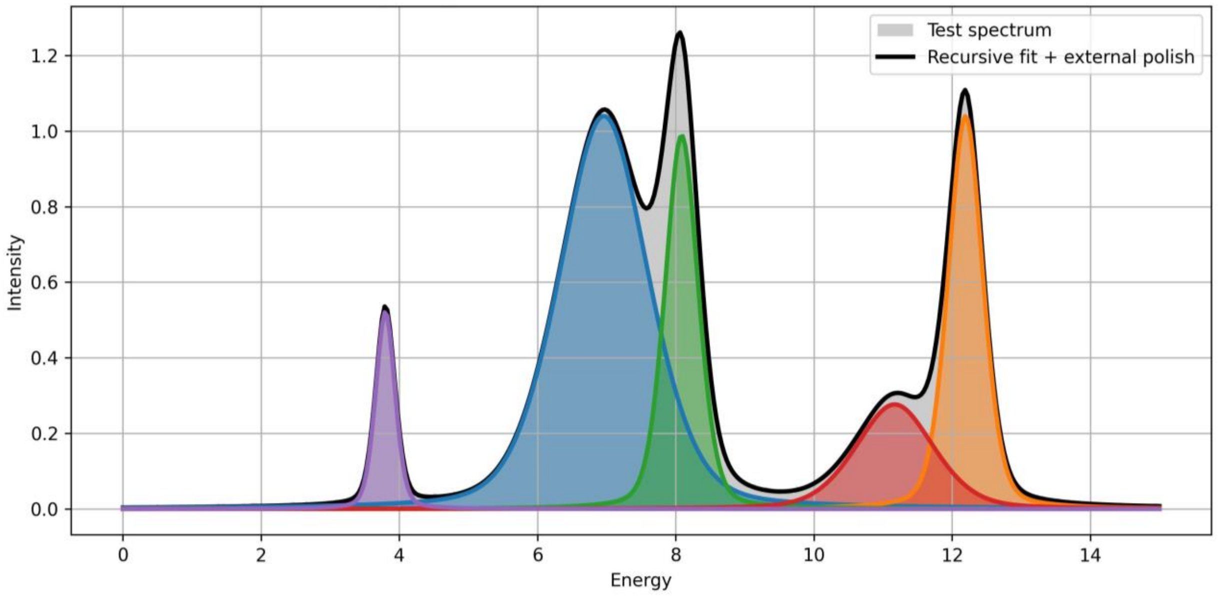
Objective: high throughput fits of XPS spectra

Two different **contributions** in mixed composition

Two different **chemical environments** for nitrogen atoms

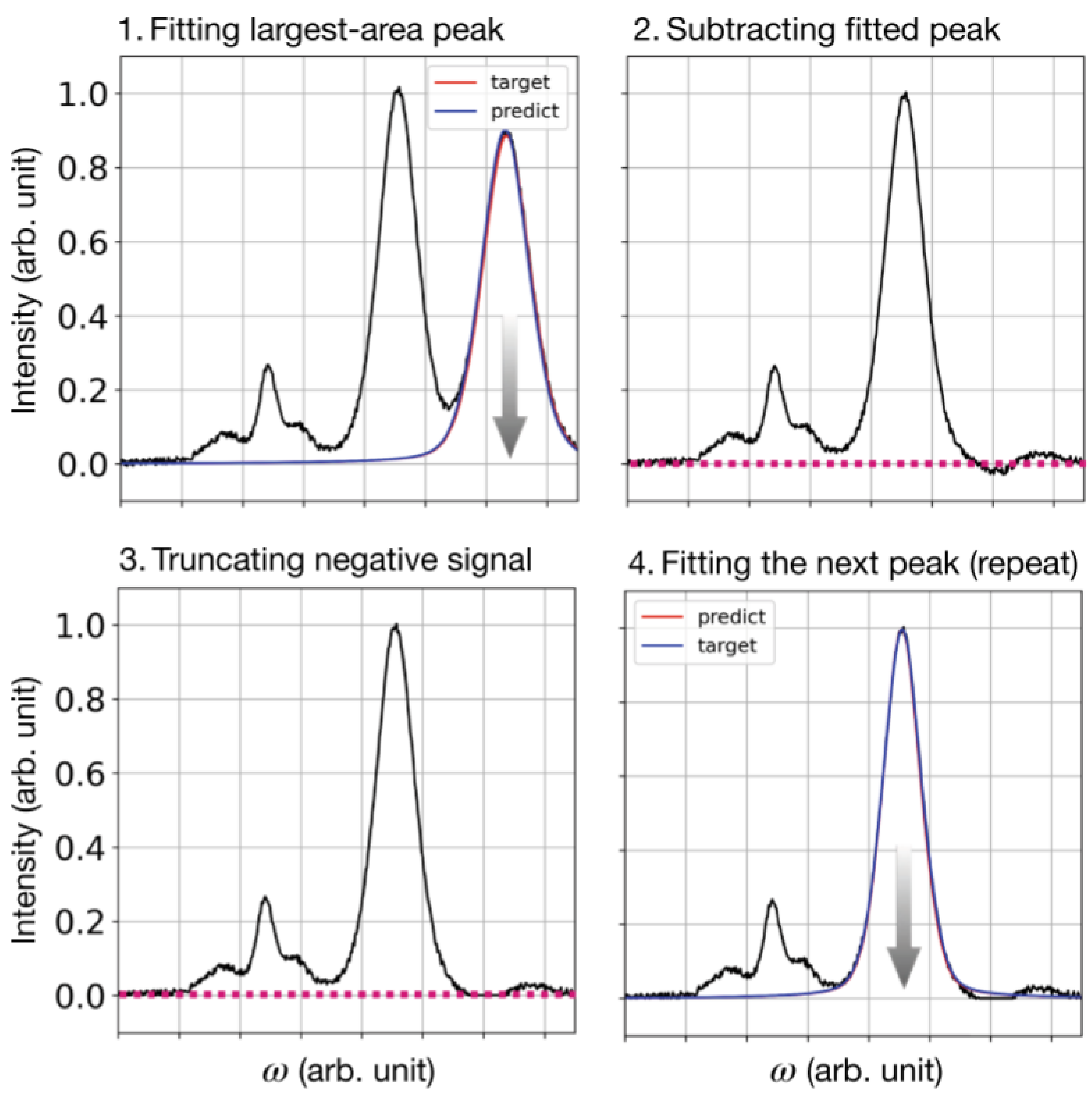


Jacobsson, T. J., Correa-Baena, J. P., Halvani Anaraki, E., Philippe, B., Stranks, S. D., Bouduban, M. E. F., Tress, W., Schenk, K., Teuscher, J., Moser, J. E., Rensmo, H., & Hagfeldt, A. (2016). Unreacted PbI2 as a Double-Edged Sword for Enhancing the Performance of Perovskite Solar Cells. Journal of the American Chemical Society, 138(32), 10331–10343.



Fitting procedure must allow to distinguish peak contributions in XPS spectra

Iterative identification of major peak



Park, S. H., Park, H., Lee, H., & Kim, H. S. (2021). Iterative peak-fitting of frequency-domain data via deep convolution neural networks. Journal of the Korean Physical Society, 79(12), 1199–1208.

Model to fit each peak contribution: pseudo-Voigt profile (sum of Gaussian and Lorentzian functions):

$$y = c\left(\alpha \exp\left(-\frac{(a-x)^2}{2\,(\beta b)^2}\right) + (1-\alpha)\frac{(\Gamma b)^2}{1+(a-x)^2}\right)$$

Ratios between both functions are fixed (α, β and Γ)

Fitting parameters:

- a: peak position
- b: peak width
- c: peak amplitude

Principle of the procedure: iterative identification of a, b and c for each peak contribution.

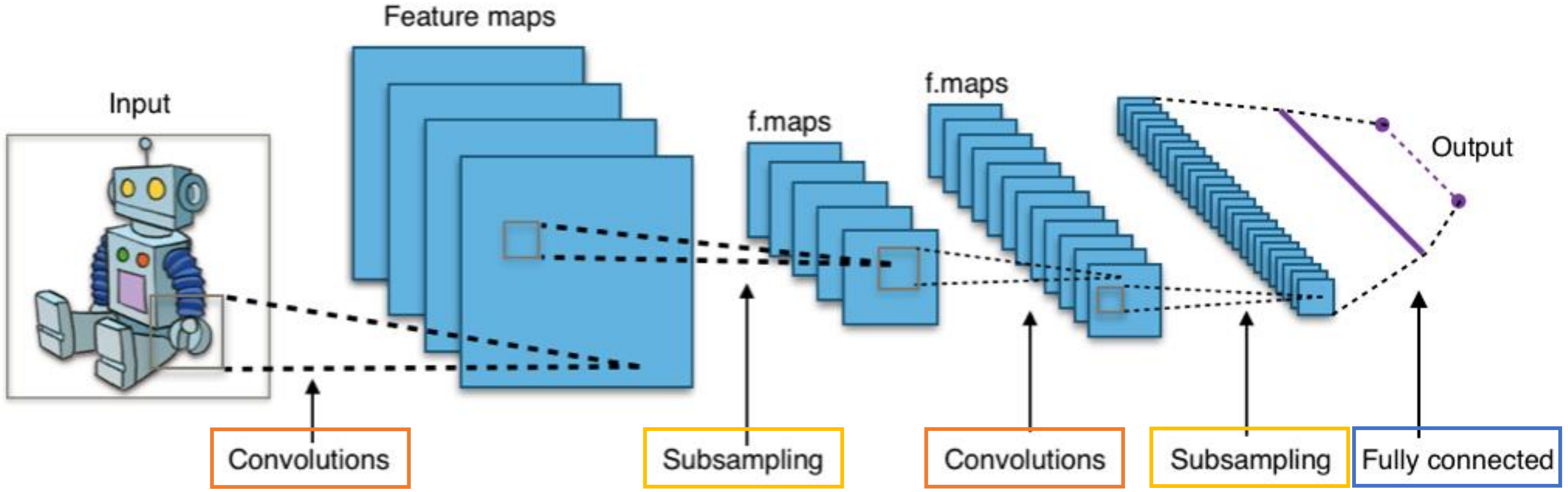
↓

Convolutional neural networks

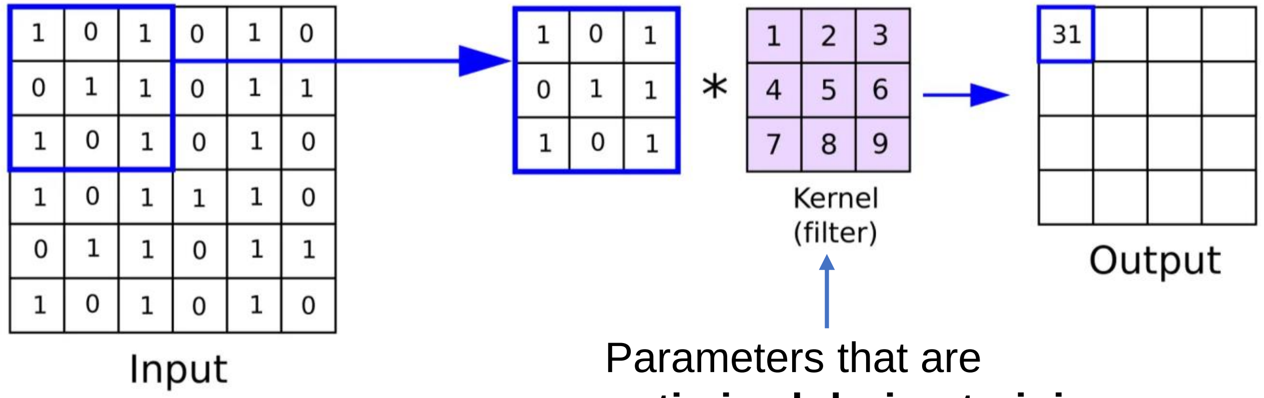
Procedure proposed by Park et al. in: Iterative peak-fitting of frequency-domain data via deep convolution neural networks. Journal of the Korean Physical Society, 79(12), 1199–1208.

Convolutional neural networks

Principle of convolution neural networks: **serial connection** of **convolution** products and **filters**



Convolution layer: convolution with a kernel



Parameters that are
optimized during training

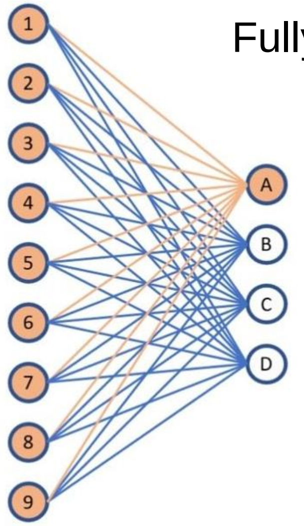
Filtering / subsampling layer:

1	0	2	3
4	6	6	8
3	1	1	0
1	2	2	4

→

6	8
3	4

Max pooling

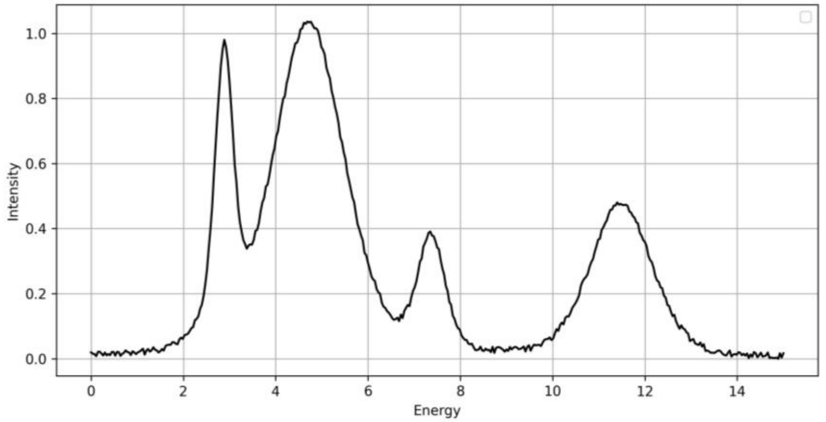


Fully connected layer:

- All inputs are connected to outputs
- Weights of the connections are **optimized during training**

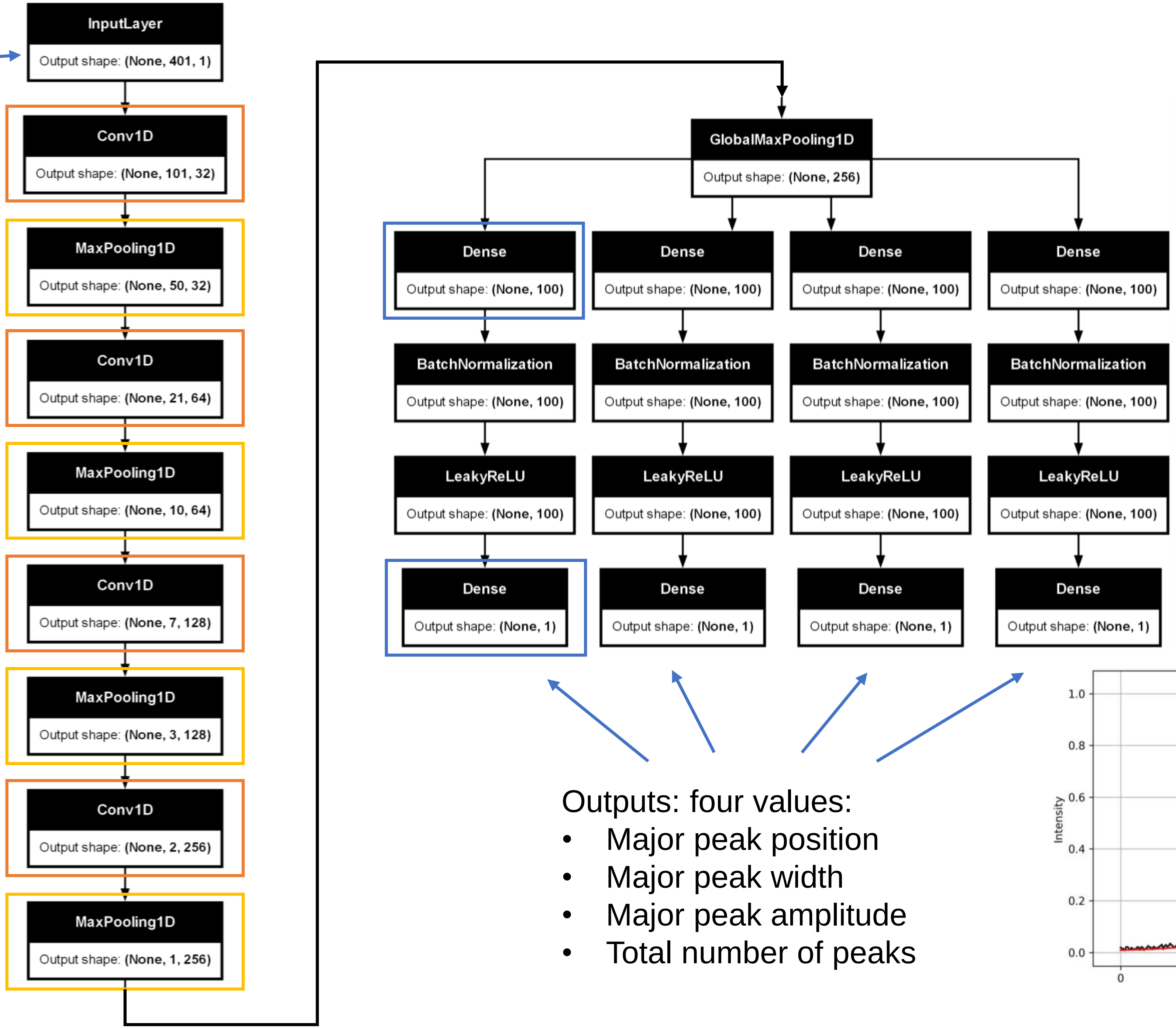
Convolutional neural networks for peak identification

Input: 1D vector containing one XPS spectrum



Convolution layer

Filtering layer

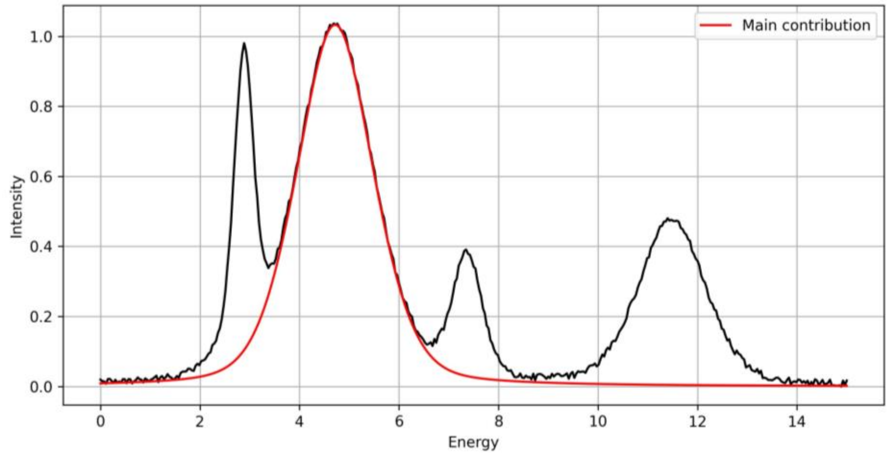


Contemporary CNNs include now significantly more layers:

- Activations functions
- Normalizations
- Random perturbations
- …

Outputs: four values:

- Major peak position
- Major peak width
- Major peak amplitude
- Total number of peaks

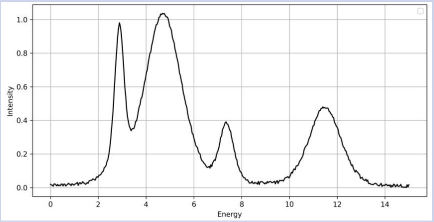


Training of CNN is necessary to adjust internal parameters and obtain desired outputs

Training CNNs

Training database:

- Each entry is synthetized with the sum of 1 to 5 contributions

Total spectrum	Major peak position	Major peak width	Major peak amplitude	Total number of peaks
. . .				
	4.51	0.87	0.94	4
. . .				

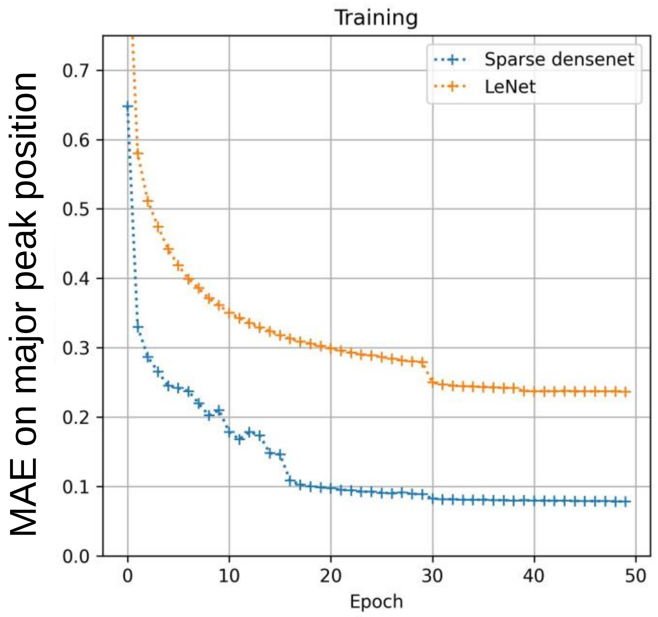
~ 1 000 000 entries

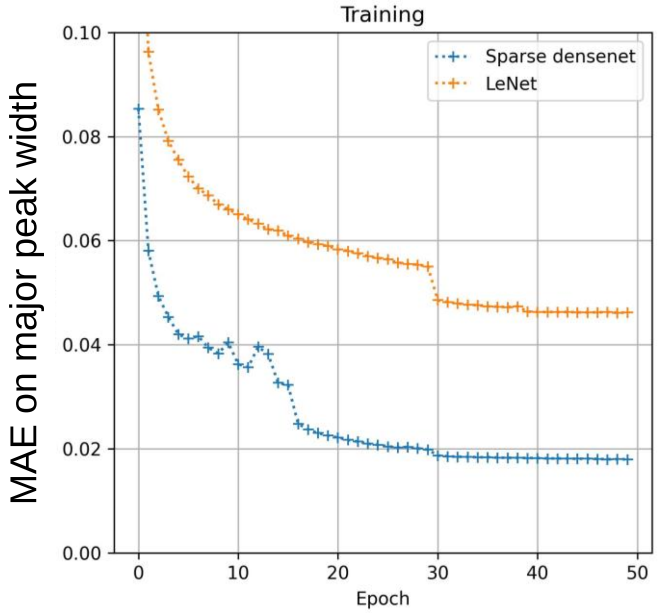
Input of CNN

Outputs of CNN

- Internal parameters of CNN are optimized to minimize error between CNN outputs and true values.
- Exploration of CNN internal parameters is automatized.
- Error is quantified with Mean Absolute Error: MAE

- Sparse densenet: model developed by Park et al.
- LeNet: first simple CNN





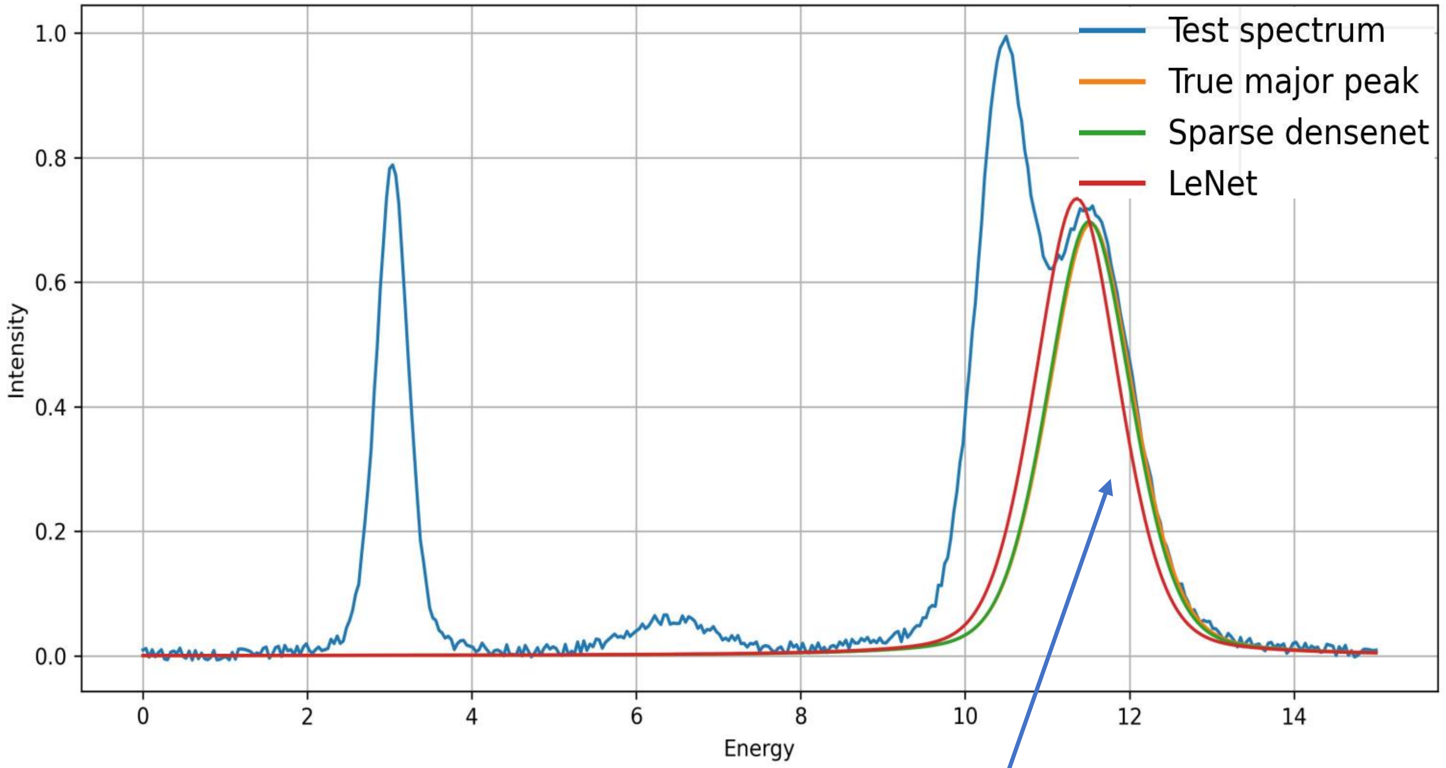
Optimization of CNN internal parameters

Testing CNNs

Test database:

- Similar database to training
- Distinct entries employed to evaluate performances on new cases
- ~ 150 000 entries

Example of test result

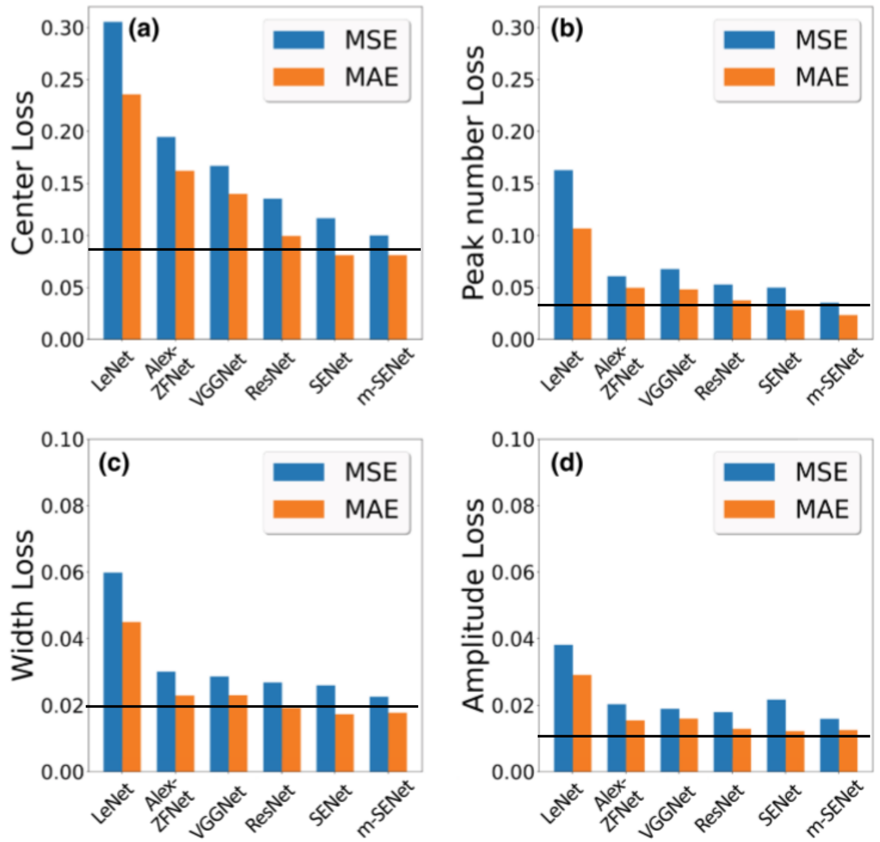


Almost perfect superposition of CNN result and true major peak

Statistical error over test database:

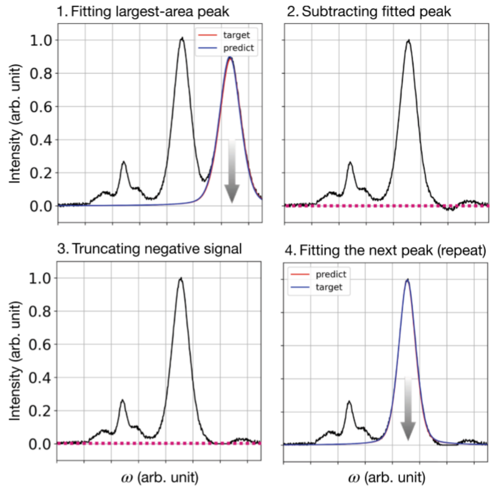
	Sparse densenet	LeNet
MAE on major peak position	0.09	0.25
MAE major peak width	0.02	0.05
MAE major peak amplitude	0.01	0.03
MAE on total number of peaks	0.03	0.1
Average fit time	1.4×10^{-3} s	1.1×10^{-4} s

Similar results to Park et al. (1 200 000 entries training database)



Results obtained here (1 000 000 entries) training database

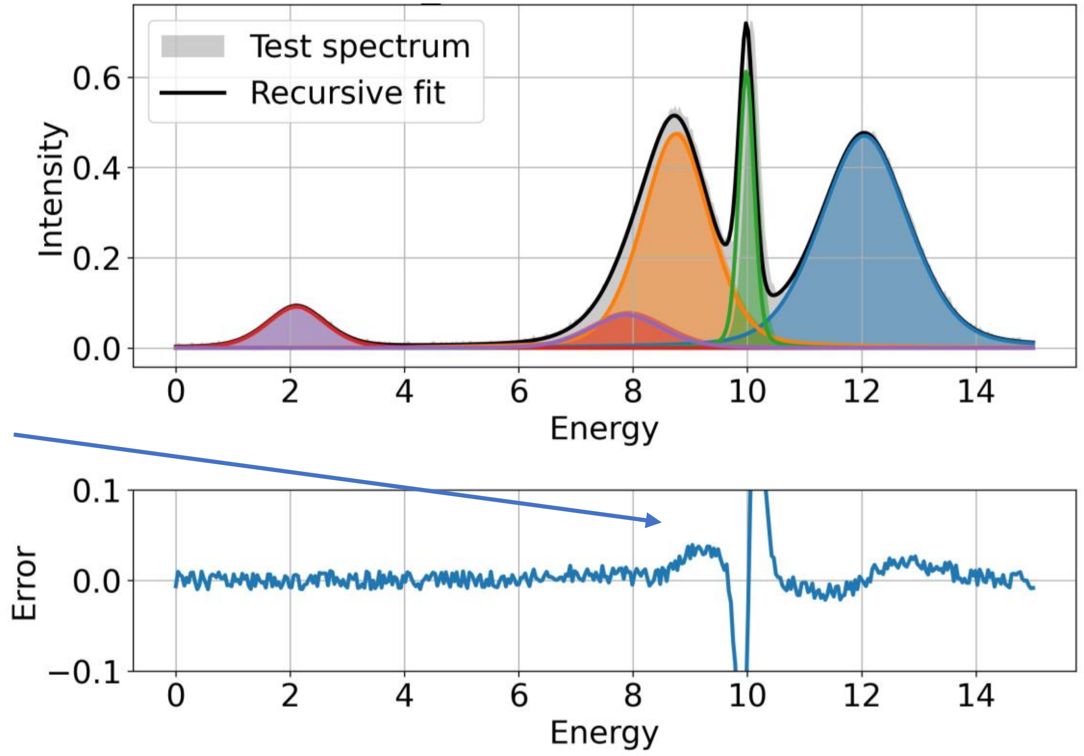
Iterative application of CNN



Neural network as been developed and trained on the identification of **one peak**

It is applied iteratively to find several contributions in a spectrum

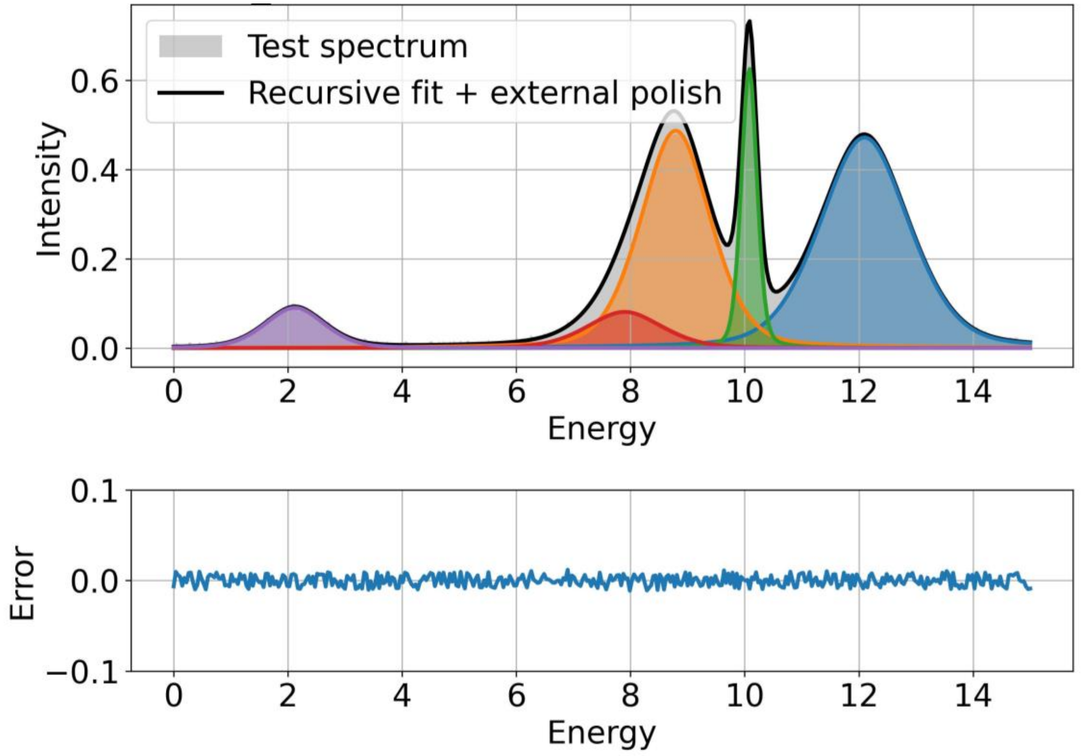
CNN is applied 5 times to iteratively find the remaining major peak



Good results but still errors

Average processing time: 0.21 s per spectrum

CNN is applied 5 times to iteratively find the remaining major peak
+
Least square reduction fit initialized by CNN results to “polish” results



Almost perfect fit

Average processing time: 2.22 s per spectrum

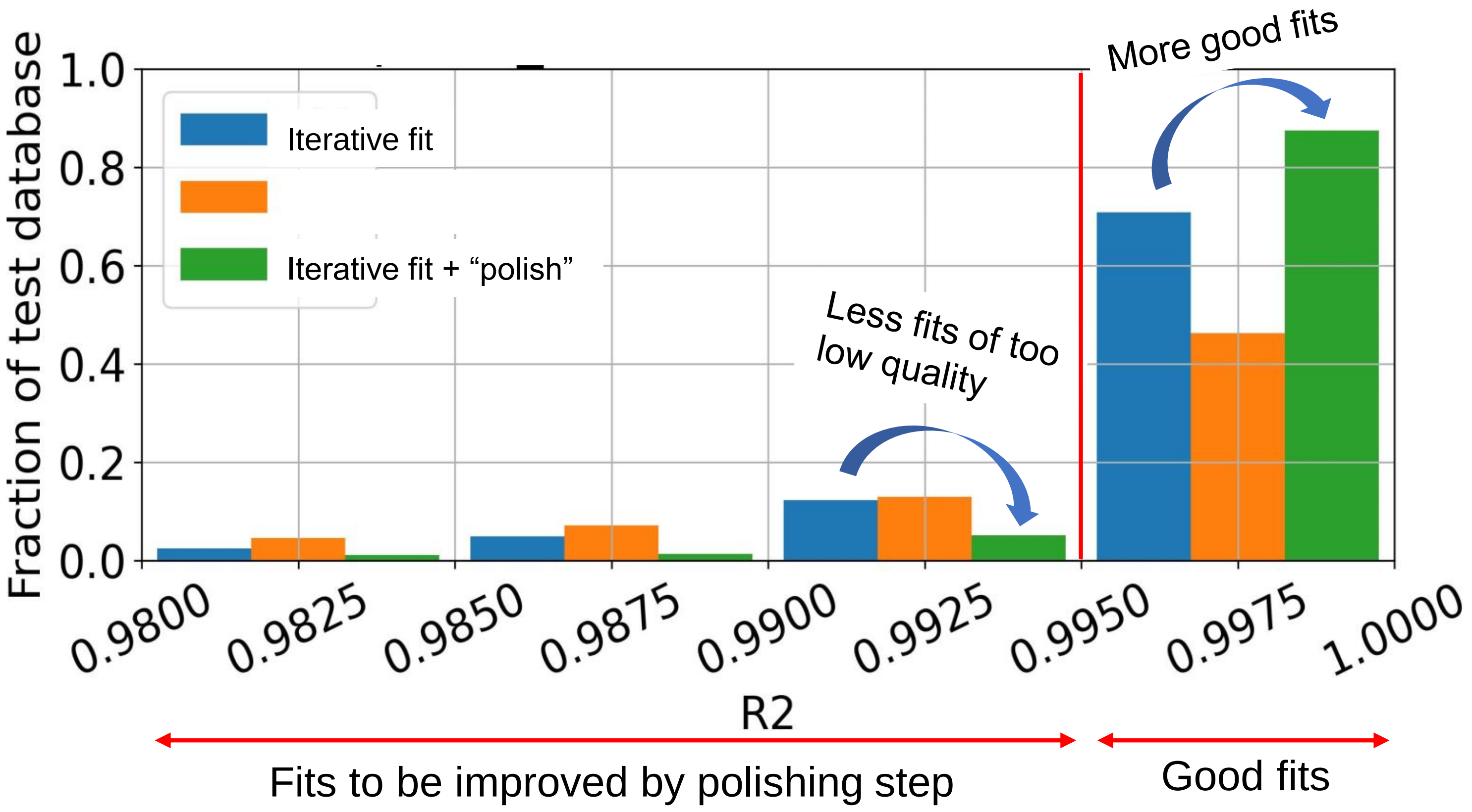
Iterative application of CNN: statistical results

Overall fit quality is evaluated through R^2 coefficient for **total spectrum**:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

Fit quality is sufficient when $R^2 > 0{,}995$ (empirical threshold)

Distribution of R^2 obtained over 500 test spectra:



Mean Average Error for each **peak parameter**:
(only cases with two or five contributions)

		Sub peak 1	Sub peak 2	Sub peak 3	Sub peak 4	Sub peak 5	Avg all peaks
Two sub-peaks							
MAE on position	Iterative fit	0.018	0.024				0.021
	Iterative fit + "polish"	0.014	0.020				0.017
MAE on width	Iterative fit	0.011	0.023				0.017
	Iterative fit + "polish"	0.011	0.020				0.015
MAE on amplitude	Iterative fit	0.006	0.010				0.008
	Iterative fit + "polish"	0.006	0.008				0.007
Five sub-peaks							
MAE on position	Iterative fit	0.173	0.496	0.845	1.055	0.887	0.691
	Iterative fit + "polish"	0.233	0.572	0.789	0.871	0.774	0.648
MAE on width	Iterative fit	0.031	0.072	0.104	0.160	0.156	0.105
	Iterative fit + "polish"	0.044	0.080	0.098	0.144	0.142	0.102
MAE on amplitude	Iterative fit	0.028	0.043	0.061	0.067	0.050	0.050
	Iterative fit + "polish"	0.035	0.046	0.058	0.060	0.045	0.049

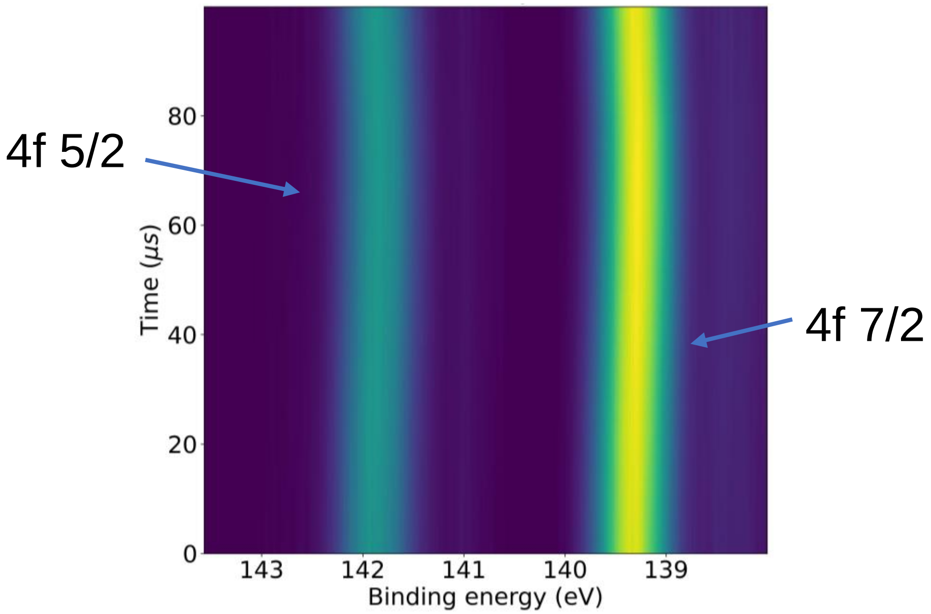
Lower average error after polishing

Remaining contributions are more noisy and more sensitive to accuracy of previous fits

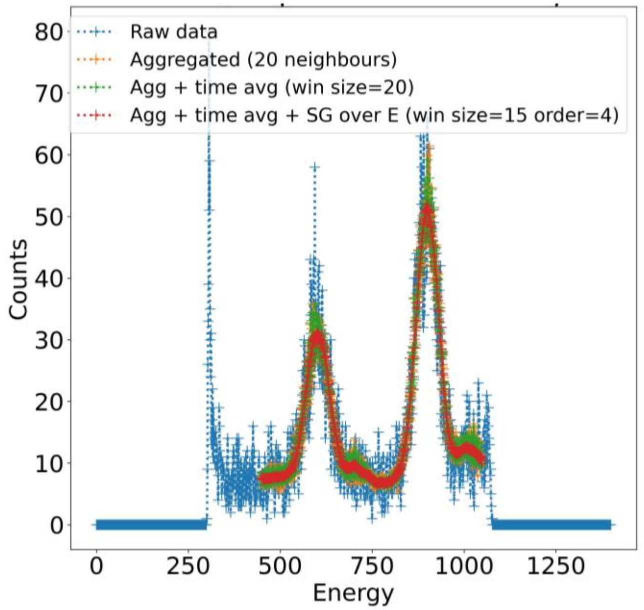
- Polishing step **improves fit quality**
- Even in most difficult cases (five contributions), **average error remains low**
- Statistics over iterative fit quality were not evaluated by Park et al.

Results on experimental measurements

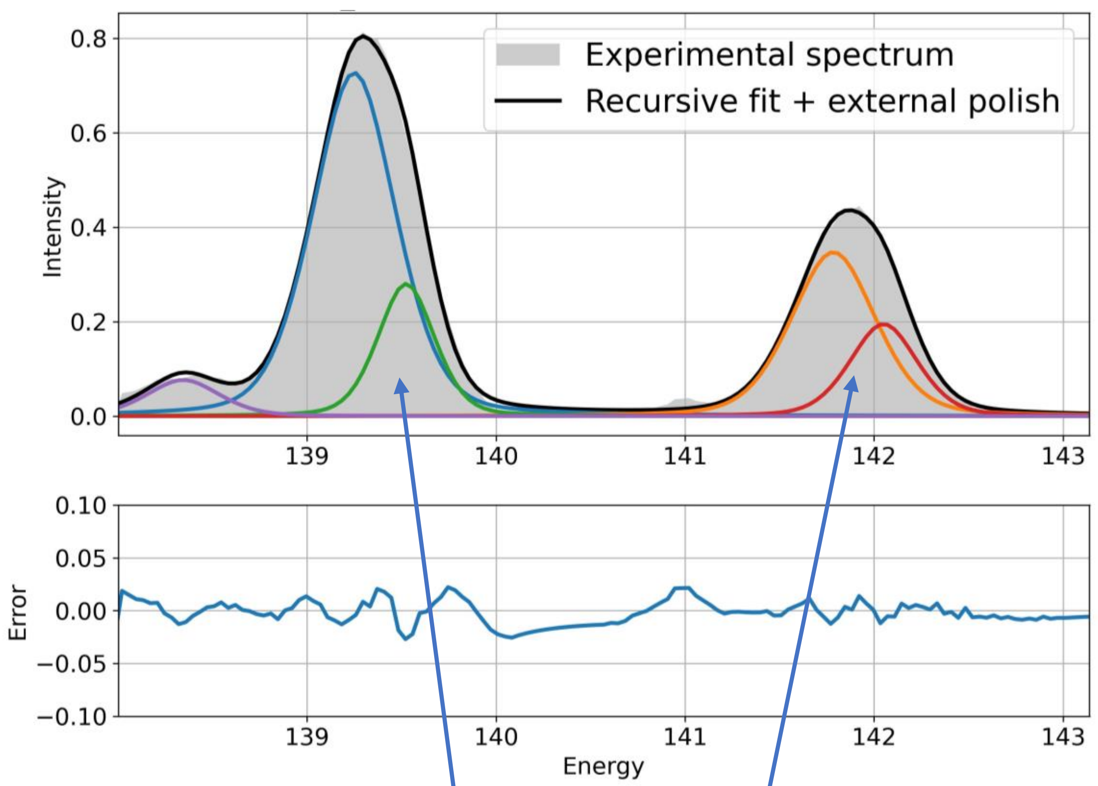
Measurement of Pb 4f:



Filtering noisy experimental data

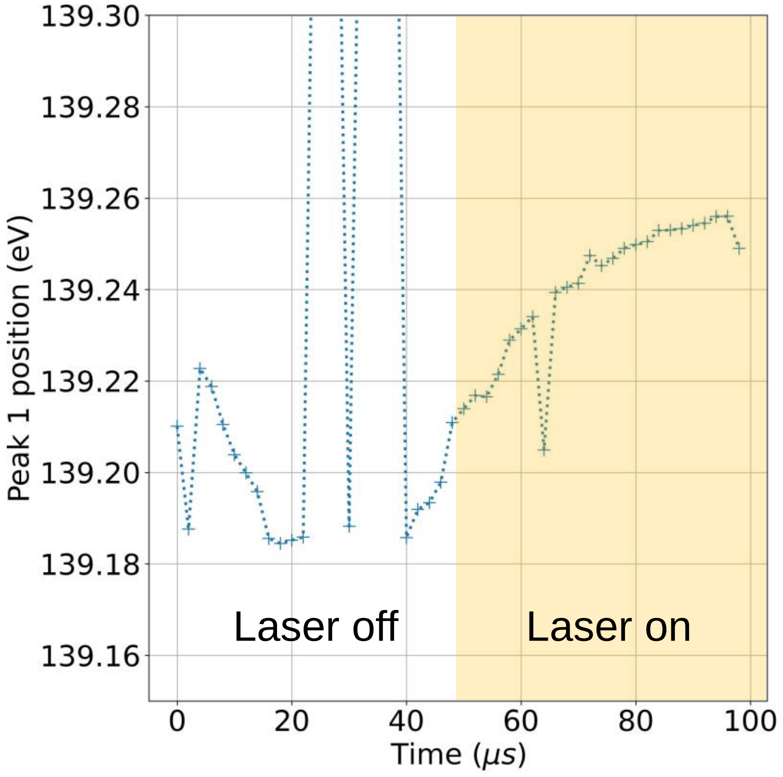


Total fit time (50 spectra): 4 min



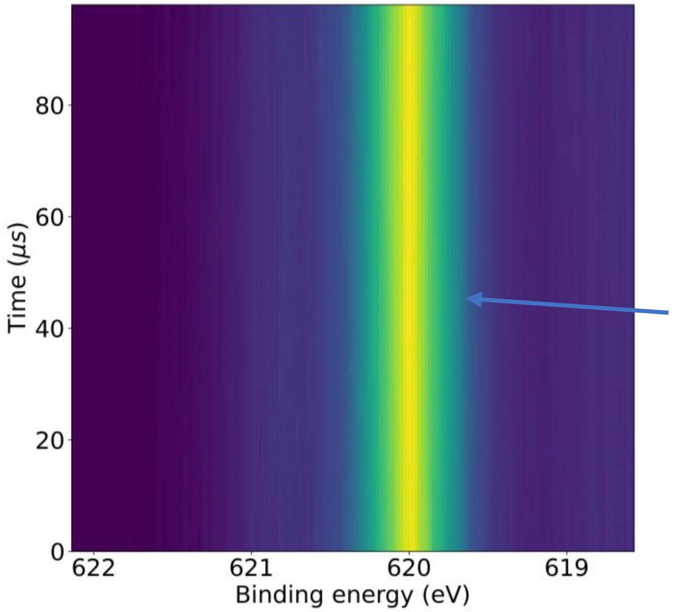
Two contributions found for both peaks: need of further optimization ?

Evolution of first peak contribution:



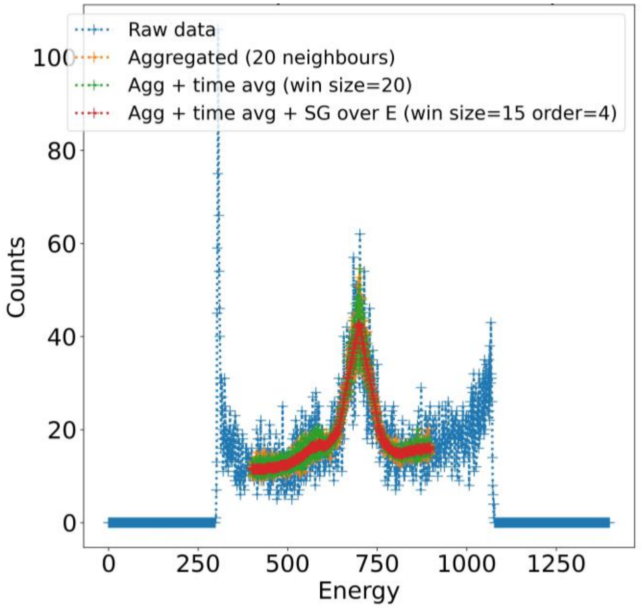
Results on experimental measurements

Measurement of I 3d

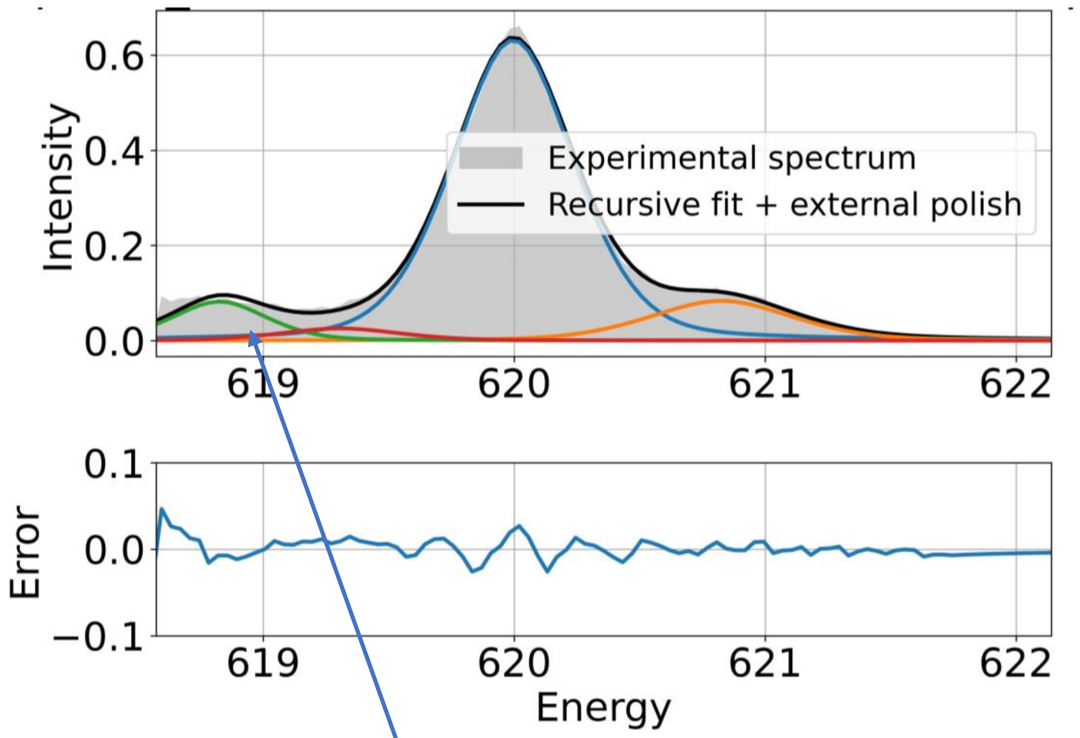


3d 5/2

Filtering noisy experimental data

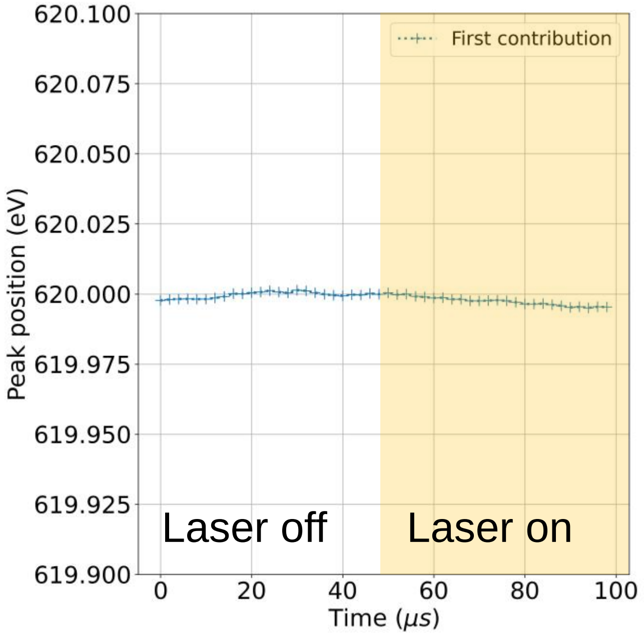


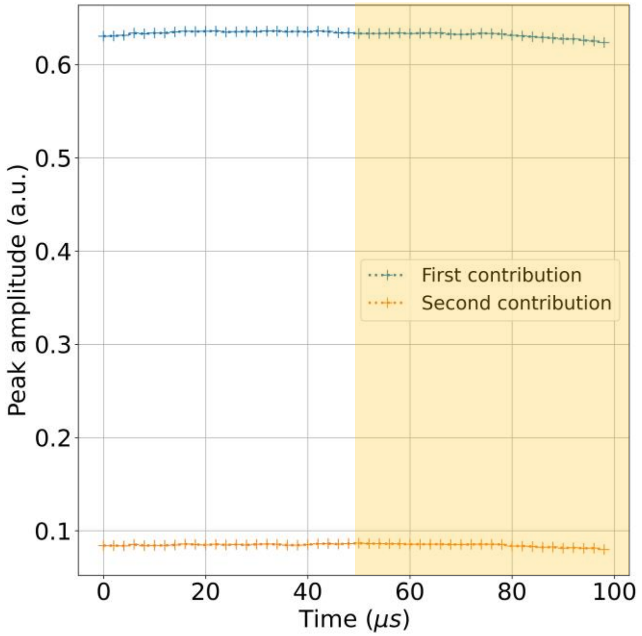
Total fit time (50 spectra): 4 min



Need of better defined background ?

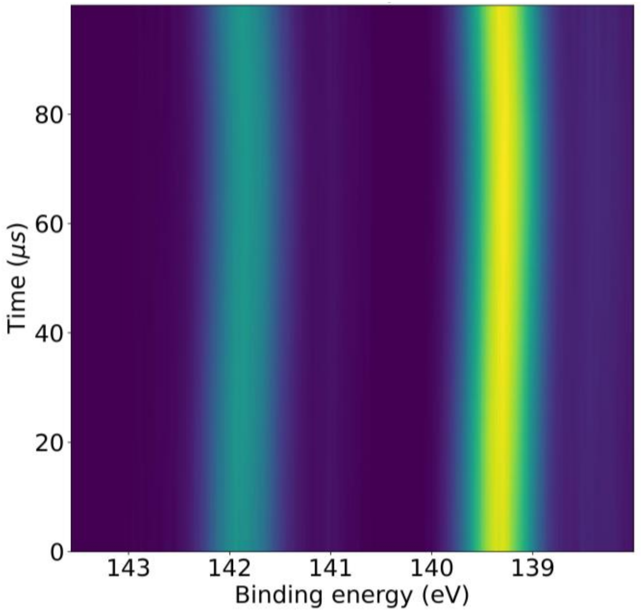
Evolution of two first peak contributions:



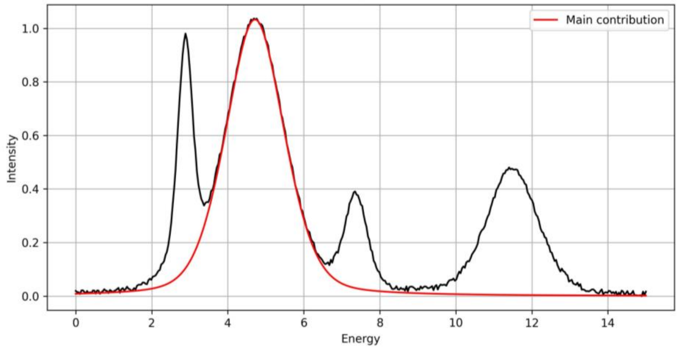


Conclusion

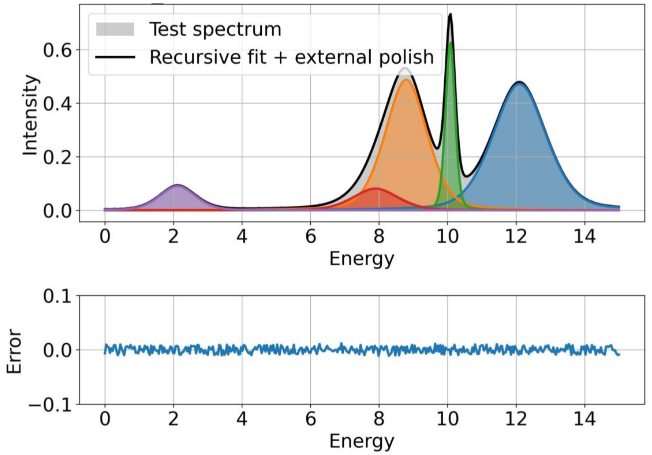
Analysis of time resolved XPS measurements



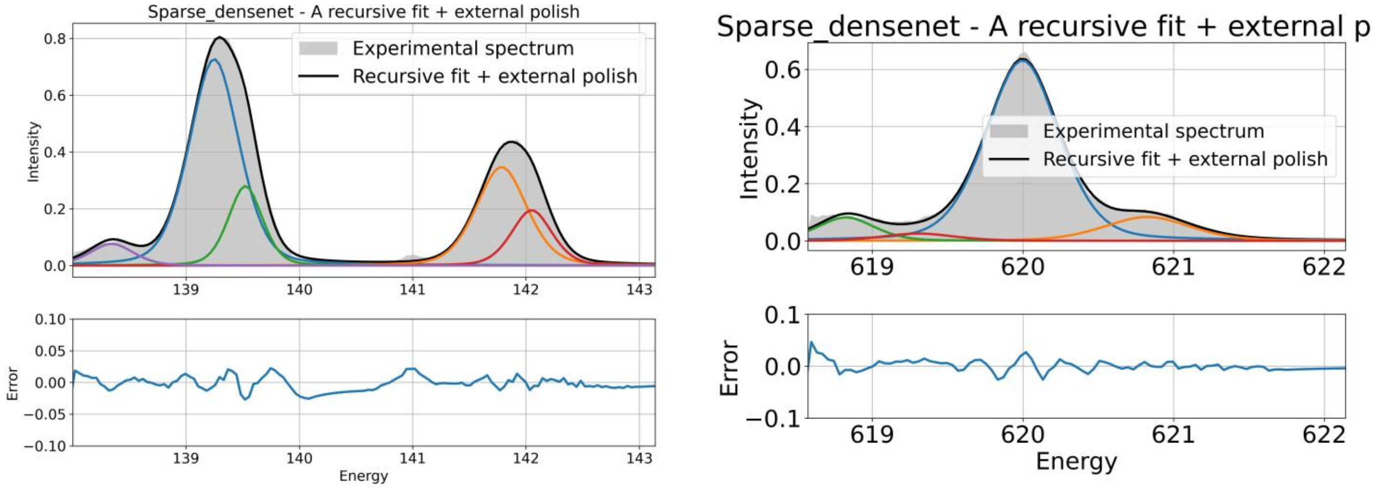
Training of convolutional neural network to find the major contribution



Recursive procedure to find all contributions



Application to experimental measurements



Tracking of contribution properties over time