



Attosecond streaking phase retrieval with neural network

JONATHON WHITE* AND ZENGHU CHANG

*Institute for the Frontier of Attosecond Science and Technology, CREOL and Department of Physics,
University of Central Florida, 4111 Libra Drive, PS430, Orlando, FL 32816, USA*

**jonathonwhite5@gmail.com*

Abstract: A new method for retrieving the spectral phase of isolated attosecond X-ray pulses from streaking traces is explored. The neural network method shows the potential for nearly instantaneous attosecond streaking phase retrieval, without use of the central momentum approximation. A neural network is trained with computer generated data that contain statistical noise and shown to correctly retrieve the phase of both computer generated and experimental attosecond streaking traces.

© 2019 Optical Society of America under the terms of the [OSA Open Access Publishing Agreement](#)

1. Introduction

The generation of single isolated attosecond pulses allows for investigation of electron dynamics through attosecond pump probe experiments [1]. Characterization of extreme ultraviolet (XUV)/soft X-ray attosecond pulses is necessary for determining their temporal profile and pulse duration [2]. The commonly-used Frequency-resolved optical gating for complete reconstruction of attosecond bursts (FROG-CRAB) method is an iterative minimization algorithm used to retrieve the spectral phase of attosecond pulses with the measurement of an attosecond streaking trace [3]. Since the power spectrum of a pulse can be easily measured with a grating spectrometer, the XUV pulse is fully characterized once the spectral phase is known.

The FROG-CRAB method is adapted from phase retrieval with femtosecond pulses using FROG (Frequency Resolved Optical Gating) [4] to retrieve phase from an attosecond streaking trace. This method, when applied to attosecond streaking traces requires the implementation of the central momentum approximation (CMA), which reduces accuracy of the retrieval [5]. Several other alternative methods have been developed that do not need to make such an approximation [5-7]. However, the genetic algorithm and other minimization schemes used in them are time consuming. In this paper, an alternative method to attosecond streaking trace phase retrieval is demonstrated using a neural network. A neural network for phase retrieval of femtosecond pulses has been shown to be successful for femtosecond FROG [8].

For attosecond phase retrieval, the neural network is used to find the mapping function between the streaking trace and the XUV field, which is achieved by training the network with streaking traces with known XUV fields. Once the mapping function is identified, the network can predict XUV pulses from streaking traces it has never seen. The intensity values and the coordinates of each pixel in a streaking trace serve as the inputs to the network whereas the XUV pulse is the output. A neural network may have more than one layer [9]. There are many artificial neurons in each layer. A neuron is connected to many inputs, which generates an output signal when the inputs exceed a certain threshold. The training of the network is the adjustment of the sensitivity of the neuron to each of its inputs. The neuron is mathematically a summation of all the input signals multiplied by a weight matrix, and passed through an activation function. The weight matrix determines the sensitivity of the neuron to each input. As the network is trained, the weights for each neuron in the network are adjusted to change the output of the neural network to match it with the true output (the known XUV electric field). The mapping function is described by the weight matrix. Training neural networks is a lengthy process which demands a large

amount of computational power and speed. In recent years, advanced GPUs have been used to speed up the training process. When the network is fully trained, using the network demands minimal computational power, and an output can typically be calculated in milliseconds. This learning process is applied by us to attosecond streaking phase retrieval by numerically generating a training set of attosecond streaking traces paired with their corresponding XUV phases.

2. Attosecond streaking trace

Attosecond streaking traces are measurements used to determine the phase of attosecond XUV pulses [10]. The trace is generated with the interaction of the attosecond XUV pulse and an infrared laser pulse in a gaseous atomic medium. Photoionization occurs in the medium by single XUV photon absorption, in the presence of an infrared dressing laser field. The photoelectron spectrum varies as a function of time delay between the infrared laser pulse and the attosecond XUV pulse. This delay τ_d is incremented and the electron energy K distribution is measured for each increment, resulting in a 2-dimensional image $s(K, \tau_d)$, the attosecond streaking trace. The trace is mathematically described in Eq. (1) in atomic units by making the Strong Field Approximation [3]. Each point of the image is calculated by performing an integral along the time axis of the product of the XUV electric field at a specific delay value $\vec{E}_X(t - \tau_d)$, the dipole matrix element $\vec{d}[\vec{v} + \vec{A}_L(t)]$, the quantum phase term $e^{i\Phi_G(\vec{v}, t)}$ and $e^{-i(K+I_p)t}$. $\vec{A}_L(t)$ is the vector potential of the infrared laser.

$$s(K, \tau_d) = \left| \int_{-\infty}^{\infty} \vec{E}_X(t - \tau_d) \cdot \vec{d}[\vec{v} + \vec{A}_L(t)] e^{i\Phi_G(\vec{v}, t)} e^{-i(K+I_p)t} dt \right|^2 \quad (1)$$

$$\Phi_G(\vec{v}, t) = - \int_t^{\infty} \vec{v} \cdot \vec{A}_L(t') dt' \quad (2)$$

$$K = \frac{1}{2} \vec{v}^2 \quad (3)$$

where I_p is the ionization potential of the atom.

The FROG-CRAB assumes the dipole moment is a constant and the right hand side of Eq. (1) is the Fourier transform of the product of the XUV pulse and the phase gate, $e^{i\Phi_G(t)}$. For assumptions to be valid, \vec{v} , in the quantum phase term (Eq.(2)), is approximated as the central momentum \vec{v}_0 [3]. The phase retrieval is based on the iterative generalized projection algorithm, which applies two types of constraints at each iteration. The $E_{sig}(K, \tau_d)$ is first calculated for a pair of guessed XUV and gating function. Then its amplitude is replaced by the measured value. The new $E_{sig}(K, \tau_d)$ is inversely Fourier transformed to the time domain, which can be expressed as $E_{sig}(t, \tau_d)$. A new generation XUV and gating function are obtained by minimizing the difference between their product and $E_{sig}(t, \tau_d)$. This process continues until the RMS of the calculated and the measured $s(K, \tau_d)$ reaches an accepted error value. The central momentum approximation is not a restriction in the neural network method, and the quantum phase term remains a function of momentum \vec{v} .

3. Neural network

Neural networks commonly use both dense and convolutional layers for image processing tasks [8]. In a series of dense layers, each neuron is connected to every neuron in the previous layer, and weight values for each connection are stored in computer memory while training. In image processing tasks, if only dense layers are used, the amount of weights stored in memory becomes very large because the input (an image) has large dimensions. This problem is solved with convolutional layers. The convolutional layer output is calculated by multiplying the input by

weight matrices called filters, and summing the product. Identical filters are applied along the input image, which greatly reduces the number of weights stored. This also means that identical patterns can be recognized at different locations of the input image. The number of filters and filter size in each convolutional layer are parameters which are set to optimize the accuracy and training time of the neural network. A common network architecture for image processing is a series of convolutional layers followed by one or more densely connected layers.

The neural network used for attosecond streaking phase retrieval consists of blocks of convolutional layers with different filter sizes, similar to the network used for neural network FROG phase retrieval [8]. The stride parameter of each filter is set to 1 so the convolutional block outputs a series of feature maps with dimensions identical to the input. In a convolutional block, each filter size is set to output the same number of feature maps as the number of input channels. For example, a 2 channel input through a convolutional block with filter sizes (10, 5, 3) will output 6 feature maps. The output of this convolutional block is then input to a standard convolutional layer with a stride of 2 which reduces the size of the output feature maps. The input to the neural network is the attosecond streaking trace $s(K, \tau)$. Our network consists of three sets of these multiple filter size convolutional blocks followed by standard convolutional layers followed by two densely connected layers with the output representing the real and complex part of both the XUV and infrared pulse spectra. The first dense layer has 1024 neurons and the output matches the dimensions of the XUV and infrared spectra. The output XUV spectrum is a complex grid of 125 points, and the output IR spectrum is a complex grid of 20 points. The network output is a vector of length 290 to represent the real and complex parts of both the XUV and IR.

4. Training

The training and testing were done with a computer generated data set to mimic experimental data in the range of electron energy and time delay. The pixel number in each dimension is $301(\text{energy}) \times 58(\text{time delay})$, ranging from 50 to 350 eV and -8 to 8 fs, respectively. The XUV power spectrum used in the training data is obtained by using the experimentally measured XUV spectrum. The central wavelength of the infrared pulse is set to the wavelength obtained from the experimentally measured streaking trace (1.7 μm). Random variation is applied to both the infrared laser intensity and the infrared laser pulse carrier envelope phase. For application of artificial shot noise, the pixel signal levels in the streaking traces are set in the range close to the measured data. Statistic noise is introduced to the spectrogram for each pixel based on its signal count. The 2nd, 3rd, 4th and 5th order dispersion coefficient values are set so that the isolated XUV pulse does not extend over one IR optical period in the domain, which is a limit set by the high harmonic generation physical mechanism. Each training data sample consists of the attosecond streaking image and the complex XUV and IR field amplitudes in the frequency domain used to generate the streaking trace. The latter is the y vector in Eq. (4). The process of training a network with a pre-made data set consisting of the input and output to the network is called supervised learning. The network is trained with this data by first defining a cost function [8]. The cost function is used to evaluate the accuracy of the network, and the goal of training the network is to minimize the cost function (Eq. (6)). The cost function is defined as the mean square error of the network output and the true XUV field and IR amplitude from the generated data set.

To use the network for phase retrieval of an experimentally measured trace, a second stage of learning is used. The experimentally measured streaking trace is the input to the neural network previously trained with the supervised learning procedure on the computer generated data set, and the output is the initial predicted XUV and IR spectra. The accuracy of this output depends on how closely the training parameters have matched with the actual parameters of the measured trace (Dispersion coefficient values, IR intensity, etc.). To increase the accuracy of the neural network output, a second phase of learning is used to again adjust the weights of the network which was previously trained with supervised learning. A streaking trace is generated from the network output, and a new cost function is defined, this time set to minimize the error between the input streaking trace and the generated streaking trace, as expressed by Eq. (5).

$$\text{Out}_i^{\text{Supervised}} = \begin{pmatrix} E_{real\ XUV}^{out} \\ \vdots \\ E_{imag\ XUV}^{out} \\ \vdots \\ E_{real\ IR}^{out} \\ \vdots \\ E_{imag\ IR}^{out} \\ \vdots \end{pmatrix} \quad y_i = \begin{pmatrix} E_{real\ XUV}^y \\ \vdots \\ E_{imag\ XUV}^y \\ \vdots \\ E_{real\ IR}^y \\ \vdots \\ E_{imag\ IR}^y \\ \vdots \end{pmatrix} \quad (4)$$

$$\text{Out}_i^{\text{Unsupervised}} = \begin{pmatrix} S^{out}(K, \tau_d) \\ \vdots \end{pmatrix} \quad y_i = \begin{pmatrix} S^y(K, \tau_d) \\ \vdots \end{pmatrix} \quad (5)$$

$$C = \frac{1}{n} \sum_{i=1}^n (\text{Out}_i - y_i)^2 \quad (6)$$

The cost function is optimized using a form of gradient descent. The gradient descent method uses the partial derivative of the cost function with respect to each network parameter to reduce the cost function iteratively. With gradient descent (Eq. (7)), the next set of network weights θ^{next} is calculated with a constant learning rate η .

$$\theta^{next} = \theta - \eta \nabla_{\theta} C \quad (7)$$

The network is trained using Adam optimization [13], which is a variation of gradient descent. Adam (Adaptive Moment Estimation) uses a dynamic learning rate algorithm to make large adjustments to the weights as the network is early in the training stage, and make fine adjustments as the cost function becomes very low and to converge toward the global minimum rather than local minima.

The network was trained with a data set of 80,000 noisy samples with random IR intensity, IR phase, and random XUV dispersion coefficients (2nd - 5th order). When training the network with the supervised learning procedure it is necessary to remove ambiguities in the data set. The streaking trace does not depend on the carrier envelope phase (CEP) of the XUV, which causes ambiguity in finding the mapping function. Another ambiguity originated in the time shifts of the infrared wave and the XUV pulse. Because the streaking is a spectrally and infrared cycle-resolved crosscorrelation trace, it cannot distinguish the effects of absolute time shifts of the infrared wave and the XUV pulse. Instead, the trace changes with the relative time delay between

them. We chose to use the XUV pulse as the delay time reference and allow the carrier envelope phase of the infrared pulse to change. It is done by expressing the XUV spectral phase as a 5th order Taylor series and setting the carrier-envelope phase and linear phase term to zero. We found this is an effective way to remove ambiguities caused by the CEP of the XUV and the absolute time shifts of the infrared wave and XUV pulse. The dipole matrix element is approximated as 1 and its effects on the XUV pulse duration can be accounted after the spectral phase is obtained [11]. The network is trained using the Tensorflow Python library running on a graphics card for increased speed. The supervised learning converges to a solution after it optimizes weights for all the 80,000 samples 40 times (40 epochs). The optimization is performed with a batch size of 10, meaning the gradients (Eq. (7)) used to optimize the cost function are the average of 10 samples. This happens for the 80,000 samples in batches of 10 for every step (Fig. 1). The cost function reaches a minimum at approximately 40 epochs, which took approximately 3.5 hours running in an Nvidia Titan X Graphics card.

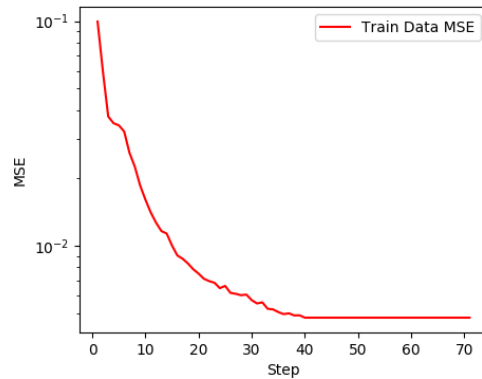


Fig. 1. The MSE cost function is evaluated on 500 samples from the training set after each epoch to evaluate the accuracy of the network on the training data set of 80,000 samples.

5. Results

After training the network with the set of 80,000 samples, the output of the network is tested with computer generated streaking traces generated with the same algorithm used to construct the training data. The output XUV spectrum (Fig. 2) is compared with the true XUV spectrum, and a streaking trace is generated from the output XUV and Infrared spectra.

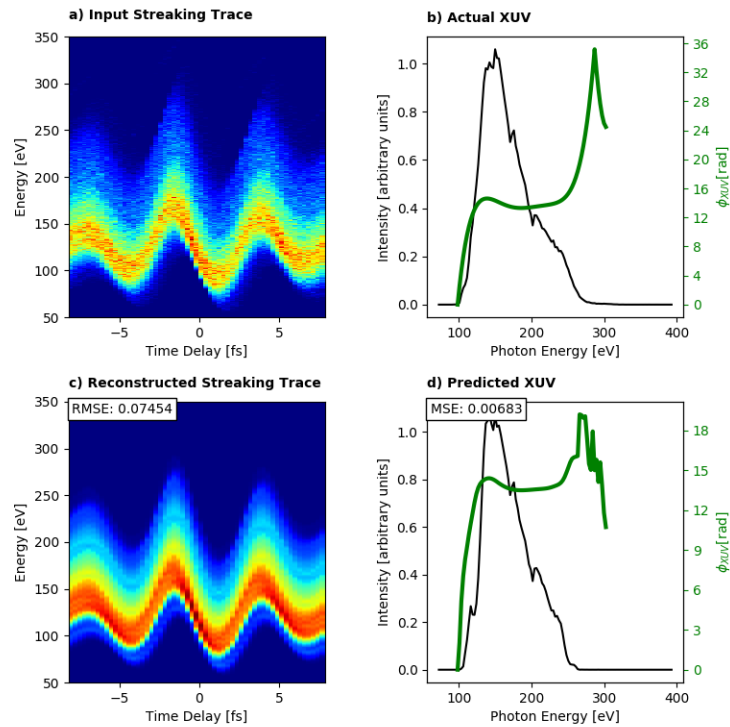


Fig. 2. (a) Input streaking trace. (b) True XUV spectrum and phase corresponding to the streaking trace. (c) Streaking trace from the predicted XUV field (d) Predicted XUV from the neural network output.

The average MSE of the training samples is $5.58 \cdot 10^{-3}$ after training with supervised learning (Fig. 1). The network is tested with computer generated streaking traces generated with the same algorithm as the training data (Fig. 3). These tests show a similar MSE to the average error of the training samples.

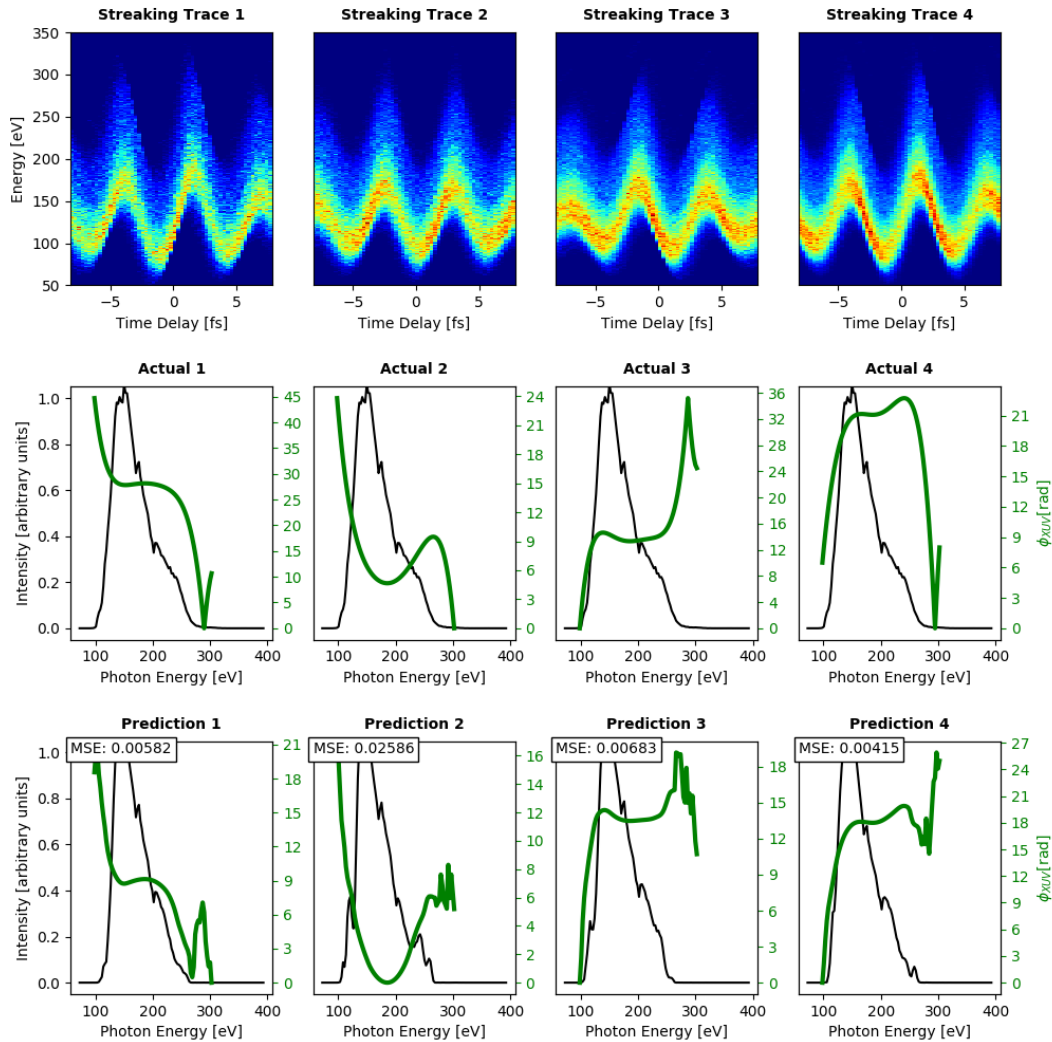


Fig. 3. Several streaking traces are generated and input to the network to test the accuracy of the network with computer generated data. The MSE values of the output field vectors for these traces are similar to the training error shown in Fig. 1.

To characterize an experimentally measured streaking, the unsupervised learning procedure is used to minimize the generated streaking trace from the network output and the input streaking trace. The unsupervised learning procedure takes significantly less time to converge, approximately 5-10 minutes because the pattern recognition is already trained into the network from the supervised learning, only small adjustments are needed. Figure 4 shows the input streaking trace and the reconstructed trace as well as the predicted XUV after using the unsupervised learning procedure.

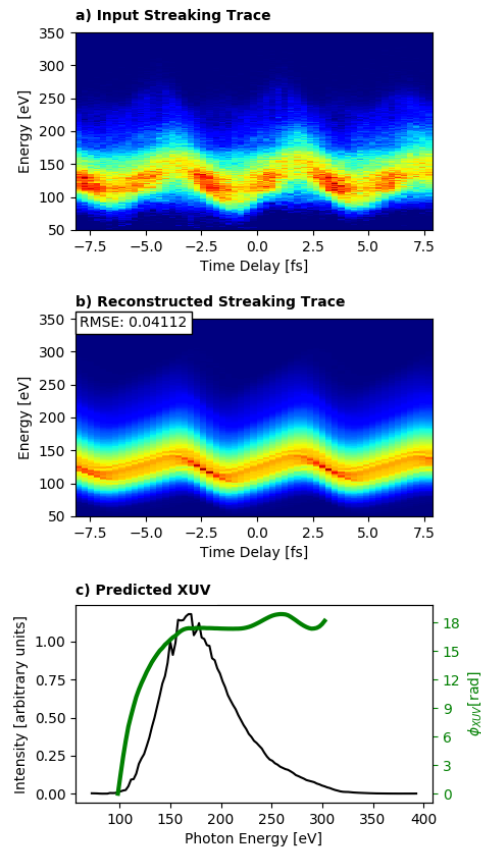


Fig. 4. (a) Input measured streaking trace. (b) Reconstructed Streaking Trace. (c) Predicted XUV spectral phase.

The RMSE between the measured and reconstructed FROG traces is a major measure to validate the FROG phase retrieval. With the neural network, the RMSE of the measured and reconstructed streaking spectrograms is close to that achieved by FROG-CRAB for narrow band XUV pulses [14] and can be further improved.

The PROOF method in refs. [5] and [12] converges in more than one hour for processing data similar to the one in Fig. 4. It is mentioned in [6] that the PROBP typically takes about 30 minutes to converge when the XUV spectrum is known and is fixed. It takes 1.5 hours on a standard desktop computer using the VTGPA to retrieve phase of the XUV pulses with a 100 eV bandwidth [7]. The main advantage of the neuron network is that once it is trained properly, it can retrieve the phases from other streaking traces almost instantly (in milliseconds).

A network may be trained with many variables using computer simulated data, as the computer simulated data matches closely with a real measured trace, the output of the network will have a high degree of accuracy without using unsupervised learning. Ideally, the network would be trained with many real measured streaking traces using unsupervised learning, and then used to predict unknown streaking traces, but because it takes a long time to measure many streaking traces, we can rely on random computer generated streaking traces to train the network.

6. Conclusion

The neural network proves to be a promising method for attosecond phase retrieval from streaking traces. The method does not require the central momentum approximation. The network is able to accurately retrieve the phase of computer simulated streaking traces with supervised learning and experimentally measured streaking traces with unsupervised learning. High phase retrieval speed is important for tuning attosecond pulse duration during experiments and for characterizing X-ray Free Electron Laser pulses that may change from shot to shot.

Funding

Air Force Office of Scientific Research (AFOSR) (FA9550-15-1-0037, FA9550-16-1-0013, FA9550-17-1-0499); Army Research Office (ARO) (W911NF-14-1-0383); Defense Advanced Research Projects Agency (DARPA), Topological Excitations in Electronics (TEE) (D18AC00011); National Science Foundation (NSF) (1806584).

References

1. S. R. Leone, C. W. McCurdy, J. Burgdörfer, L. S. Cederbaum, Z. Chang, N. Dudovich, J. Feist, C. H. Greene, M. Ivanov, and R. Kienberger, "What will it take to observe processes in 'real time'?", *Nature Photonics* **8**, 162-166 (2014).
2. M. Chini, K. Zhao, and Z. Chang, "The generation, characterization and applications of broadband isolated attosecond pulses," *Nature Photonics* **8**, 178-186 (2014).
3. Y. Mairese and F. Quéré, "Frequency-resolved optical gating for complete reconstruction of attosecond bursts," *Physical Review A* **71**, 011401 (2005).
4. D. J. Kane and R. Trebino, "Characterization of arbitrary femtosecond pulses using frequency-resolved optical gating," *IEEE Journal of Quantum Electronics* **29**, 571-579 (1993).
5. M. Chini, S. Gilbertson, S. D. Khan, and Z. Chang, "Characterizing ultrabroadband attosecond lasers," *Opt. Express* **18**, 13006-13016 (2010).
6. X. Zhao, H. Wei, Y. Wu, and C. D. Lin, "Phase-retrieval algorithm for the characterization of broadband single attosecond pulses," *Physical Review A* **95**, 043407 (2017).
7. P. Keathley, S. Bhardwaj, J. Moses, G. Laurent, and F. Kärtner, "Volkov transform generalized projection algorithm for attosecond pulse characterization," *New Journal of Physics* **18**, 073009 (2016).
8. T. Zahavy, A. Dikopoltsev, D. Moss, G. I. Haham, O. Cohen, S. Mannor, and M. Segev, "Deep learning reconstruction of ultrashort pulses," *Optica* **5**, 666-673 (2018).
9. A. Géron, *Hands-On Machine Learning with Scikit-Learn and TensorFlow* (O'Reilly Media, 2017).
10. J. Itatani, F. Quéré, G. L. Yudin, M. Y. Ivanov, F. Krausz, and P. B. Corkum, "Attosecond streak camera," *Physical Review Letters* **88**, 173903-173903 (2002).
11. J. Li, X. Ren, Y. Yin, K. Zhao, A. Chew, Y. Cheng, E. Cunningham, Y. Wang, S. Hu, Y. Wu, M. Chini, and Z. Chang, "53-attosecond X-ray pulses reach the carbon K-edge," *Nature Communications* **8**, 186 (2017).
12. Jie Li, "Generation and characterization of isolated attosecond pulse in the soft X-ray region," PhD thesis, University of Central Florida (2017). <http://ifast.ucf.edu/Publications.aspx?Type=8&Header=Dissertations>
13. D. Kingma and J. Ba, "Adam: a method for stochastic optimization (2014). arXiv preprint," arXiv preprint arXiv:1412.6980.
14. G. Sansone, E. Benedetti, F. Calegari, C. Vozzi, L. Avaldi, R. Flammini, L. Poletto, P. Villoresi, C. Altucci, R. Velotta, S. Stagira, S. De Silvestri, and M. Nisoli, "Isolated single-cycle attosecond pulses," *Science* **314**, 443-446 (2006).