SILIA: Software Implementation of a Multi-Channel, Multi-Frequency Lock-in Amplifier for Spectroscopy and Imaging Applications

Amrut Nadgir,[1] Richard Thurston,[1] Kirk A. Larsen,[1,2] Niranjan Shivaram,[1,3] Matthew M. Brister,[1] Daniel S. Slaughter[1]

[1] Chemical Sciences Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

[2]Graduate Group in Applied Science and Technology, University of California, Berkeley, CA 94720, USA

[3]Department of Physics and Astronomy, Purdue University, West Lafayette, IN 47907 USA

E-mail: DSSlaughter@lbl.gov

Abstract. We describe a software implementation of a multi-channel, multi-frequency Lock-in Amplifier (SILIA) to extract modulated signals from noisy data distributed over multiple channels of arbitrary number and size. This software implementation emulates the functionality of a multi-channel, multi-frequency lock-in amplifier in a post-processing step following data acquisition. Unlike most traditional lock-in amplifiers, SILIA can work with any number of input channels and is especially useful to analyze data distributed over many channels. We demonstrate the versatility and performance for extracting weak signals in spectroscopy and fluorescence microscopy. We also discuss more general applications and exhibit a method to automatically estimate error from a lock-in result.

Keywords: lock-in, phase-sensitive detection, optical spectroscopy

1. Introduction

Lock-in amplifiers are widely used to extract weak signals from noisy data by applying a modulation at a known frequency [1]. This is accomplished by using phase-sensitive techniques to remove the signal components that are not modulated at the desired frequency. Early lock-in amplifiers were analog devices, while many modern lock-in amplifiers include digital components or implemented on a field programmable gate array. Digital lock-in amplifiers tend to be simpler than their analog counterparts, while also achieving wider bandwidth, greater adjustability, higher dynamic reserve and improved accuracy [2, 3, 4, 5, 6]. Software implementations are readily extended to multiple measurement channels and reference frequencies. For example, Lu *et al.*[7] demonstrated a virtual multi-channel lock-in amplifier and applied it to the

measurement of magnetoelectric response in materials. Despite the versatility and broad applications for digital lock-in amplifiers, there are relatively few examples of software implementations of lock-in amplifiers, particularly with highly parallel measurement capabilities for multiple channels and multiple reference frequencies. By implementing a software lock-in amplifier, we adopt an approach where the amplification is a post-processing step after the data acquisition is completed. A software implementation increases the adaptability of the software to the task at hand and simplifies the lock-in process so it can be easily modified to different specifications.

Here we describe a Software Implementation of a Lock-In Amplifier (SILIA) that can be used to analyze data from an amplitude modulated system, having any number of signal channels and modulation frequency references. As a result, it can serve as a simple yet flexible and cost effective alternative for hardware lock-in amplifiers that are commonly used for multi-channel and multi-frequency applications. Many of the experiments and techniques involving lock-in amplification, such as lock-in imaging [8, 9, 10, 11], pump-probe spectroscopy [12, 13, 14, 15] and pump-probe microscopy [16, 17], require the use of specialty hardware that can perform the necessary computations during the course of the experiment. SILIA is compatible with any instrumentation that has a sampling rate above twice the maximum reference frequency and can store the unprocessed data for post-processing. We have designed SILIA for multi-dimensional data, stored locally, having many channels and one or more reference frequencies.

SILIA offers several benefits not available from traditional lock-in amplification. It can interface with any time series in a specified format. It has the capability to repeatedly retrieve the signal amplitude and relative phase of a weak signal after the initial data acquisition, facilitating the adjustment or optimization of input parameters by post-processing analysis. The software can be customized and is readily adapted to a wide variety of spectroscopic or imaging instrumentation. Since SILIA is applied as a post-processing step, the speed of the data acquisition is only limited by the digital instrumentation and there are no additional hardware optimizations required for execution. The software can also easily be manipulated to extract weak signals from varied forms of measurement, such as frequency sweeps [18]. SILIA is also able to estimate the error in the results by partitioning the input signal into overlapping intervals and locking into each of these subintervals. In addition, SILIA is advantageous to using simpler computational error reduction methods such as background subtraction and averaging because it has the ability to extract phase information and can lock into multiple overlayed signals simultaneously.

In this article, we will first discuss the principles behind lock-in amplification and how SILIA implements them in software in Section 2. In Section 3, we perform a basic multi-signal functionality test for the software showcasing the capabilities of SILIA to perform lock-in on multiple channels and references simultaneously, and produce an accurate estimation of the error for its result. Then, we benchmark the runtime and noise suppression abilities of SILIA in Section 4. Two applications of SILIA, optical

spectroscopy and fluorescence microscopy simulations, are demonstrated in Section 5. Finally, we discuss the benefits and limitations in Section 6, as well as potential areas for future improvement. Future revisions, further application examples and tutorials can be found on our Git repository[19].

2. Implementation

A Lock-in amplifier generally performs signal mixing followed by low-pass filtering [1]. The signal mixing step multiplies the input signal by a sinusoidal reference. A sinusoidal reference ensures that the Fourier components of the input signal that oscillate at the reference frequency are shifted to zero frequency in the frequency space representation of the mixed signal. A low-pass filter suppresses the frequency components of the mixed signal that are far from zero frequency. After the lock-in process is completed, the filtered output will be proportional to the amplitude of the frequency components of the input signal within a narrow band centered at the initial reference frequency, thus suppressing noise. The bandwidth is determined by the lock-in time-constant which depends on the time duration of the measurement. The phase of the signal relative to the reference can be extracted by comparing output between the in-phase and quadrature components of the reference [1, 20, 21]. SILIA fits a sinusoidal function to the references and locks into each input channel with each reference frequency (Fig. 1). If there are n input channels and k desired frequency references, then the software will output nk lock-in results.

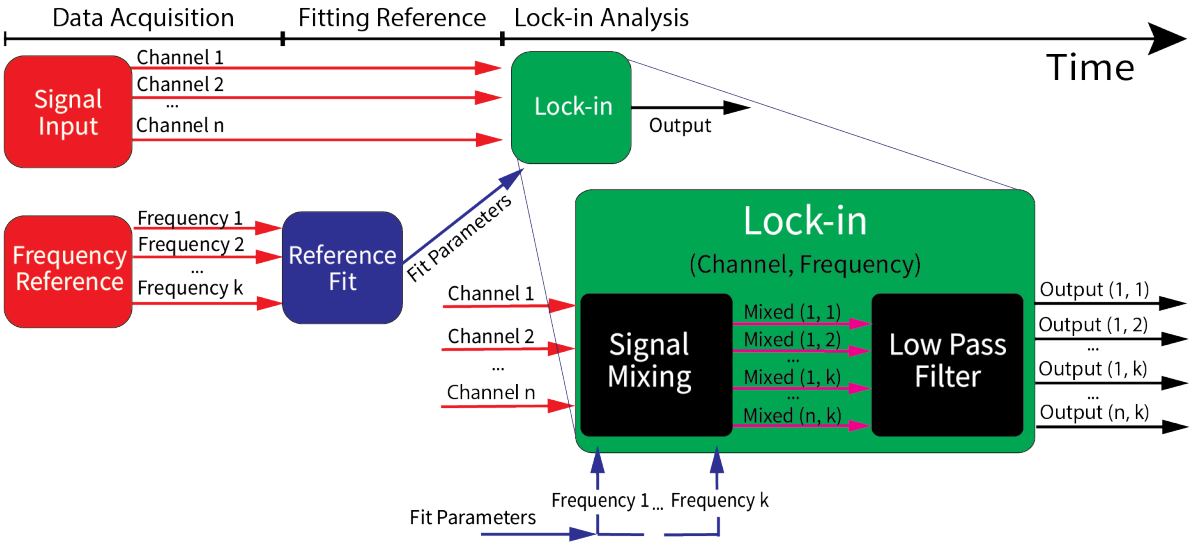

Figure 1. An overview of a single data acquisition cycle and lock-in process, showing sequence of the software lock-in analysis. The final lock-in amplification step requires the signal data over all n channels and the k fitted frequency references as inputs. The signal mixing and low-pass filter steps are run on the input channels and references where every channel and reference pair can lock-in separately. SILIA also has the option to skip the reference fit step.

SILIA requires the raw input signal for each channel, frequency references

corresponding to each amplitude modulation of the raw input signals, and a cutoff frequency for the low-pass filter. The raw input and frequency references are not passed into the software in real time and are processed after the data are acquired and recorded to local memory with timestamps. The data are then provided to SILIA in the form of a python dictionary which contains arrays of the signal data, reference data and their respective timestamps with the appropriate labels [19].

For each reference, the in-phase and quadrature mixed components are computed for each signal channel. Nonuniform sampling rates can cause fluctuations in the extracted signal. In such cases the fluctuations can be reduced or eliminated by performing a linear interpolation on the signal to reassign the signal into regular time intervals. Regularly timed samples are necessary because the Fast-Fourier Transform (FFT) algorithm used in subsequent analysis assumes the samples are uniformly sampled. The software then applies a low-pass filter to each of the mixed signals. To implement a low-pass filter in software, we applied the real FFT with a Hanning window to the mixed signal, supressing spectral components outside a symmetric interval centered on 0 by $\sim$18 dB/octave [22], with the width specified by a frequency cutoff parameter. This acts as a low-pass filter with a response curve that is essentially a step function, thus minimizing any error caused by the filtering step in conventional lock-in amplifiers, but also assumes that samples are periodic (uniform sampling). Depending on the data acquisition system, discrete sampling of the signal can limit the capability of the filter to distinguish adjacent frequency components of the signal. Therefore, the cutoff frequency is limited by the sample rate of the measured signal. Assuming nearly uniform sampling, the frequency resolution tends to improve with longer acquisition times. Most importantly, the reference frequency should not exceed the Nyquist frequency $f/2$ where f is the sampling frequency. Lock-in amplitudes and phases are computed from the measured in-phase and quadrature components of the filtered signals [23]. Each data acquisition cycle produces an array of output values indexed by their respective channel and frequency reference (Fig. 1).

Scaling factors are readily determined for applications where the signal is not expected to be sinusoidal, e.g. a square wave modulation due to an optical chopper, and a corresponding square wave reference. The Fourier series expansion of an even square wave with a peak to peak amplitude of A and a duty cycle d is

$$\frac{2A}{\pi}\sum_{n=1}^{\infty}\frac{1}{n}\sin\left(n\pi d\right)\sin\left(2n\pi ft\right)$$

where t is time and f is the frequency of the square wave [24]. As long as the cutoff frequency of the lock-in is less than f, only the $n = 1$ Fourier component will remain after the low-pass filter. Therefore, the ideal lock-in output amplitude for a standard square wave reference with a duty cycle of 1/2 would be $2A/\pi$. Such scaling is predictable and readily included in the analysis that follows.

To perform lock-in on non-sinusoidal references, we can estimate the inner products in frequency space between the reference and the input signal. To compute these

projections, SILIA has the option to skip the reference fitting step and mix the signal input with a scaled raw reference. The raw reference is scaled by a factor of 2 so the inner product between two purely sinusoidal signals that oscillate at the same frequency is the product of their amplitudes. In this approach we cannot extract phase information due to a lack of a well defined orthogonal quadrature reference input for an arbitrary periodic signal. To estimate uncertainties in the lock-in result, SILIA can automatically partition the input signal into partially overlapping segments of equal size and lock into each segment separately. The standard deviation of the various results provides an estimate of the error in the result. The size and number of segments are specified by the user.

3. Error Estimation Analysis

3.1. Methodology

To verify the accuracy of the error estimation functionality of SILIA, a series of simulated data, consisting of multiple channels with multiple references, were acquired simultaneously. The simulated data were used to compare the output lock-in signal and phase with the input signals and references. The simulation had 100 input channels. Channel groups A, C and E in in Fig. 2 contained only Gaussian noise while groups B and D had additional periodic signal modulated at 80 Hz and 120 Hz respectively. The signal was sinusoidal and had a power of $0.5\sigma^2$, where $\sigma^2 = 1$ is the variance of the Gaussian noise. Here we adopt the convention of signal to noise ratio (SNR) defined as the ratio of their powers. Therefore, the SNR in this simulation was $1 : 2$ for the channels that contained the input signals.

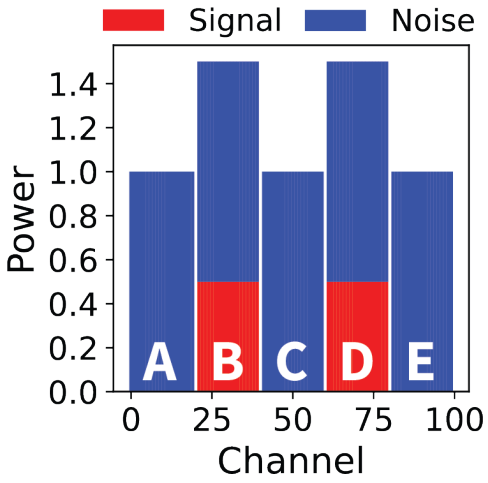

Figure 2. A graphical representation of the SNR for each channel of the input signal. 100 channels are split into groups of 20 channels each. Each group of 20 channels contain Gaussian noise with power equal to the variance $\sigma^2 = 1$. The second and fourth groups contain additional synthetic signals that are sine waves with a power of $0.5\sigma^2 = 0.5$.

The data acquisition is for a simulated time of 5 seconds at a sampling rate of 5000 Hz and a frequency resolution of 0.2 Hz. The cutoff frequency is also set to 0.2 Hz. The references are 80 and 120 Hz square waves. Output values are generated by using SILIA to lock into the 80 Hz and 120 Hz references. Uncertainties are estimated by separately analyzing a sample of 33% of the entire dataset, spanning four evenly spaced time-intervals and computing the standard deviation of the output lock-in amplitude and phase across all intervals.

3.2. Analysis

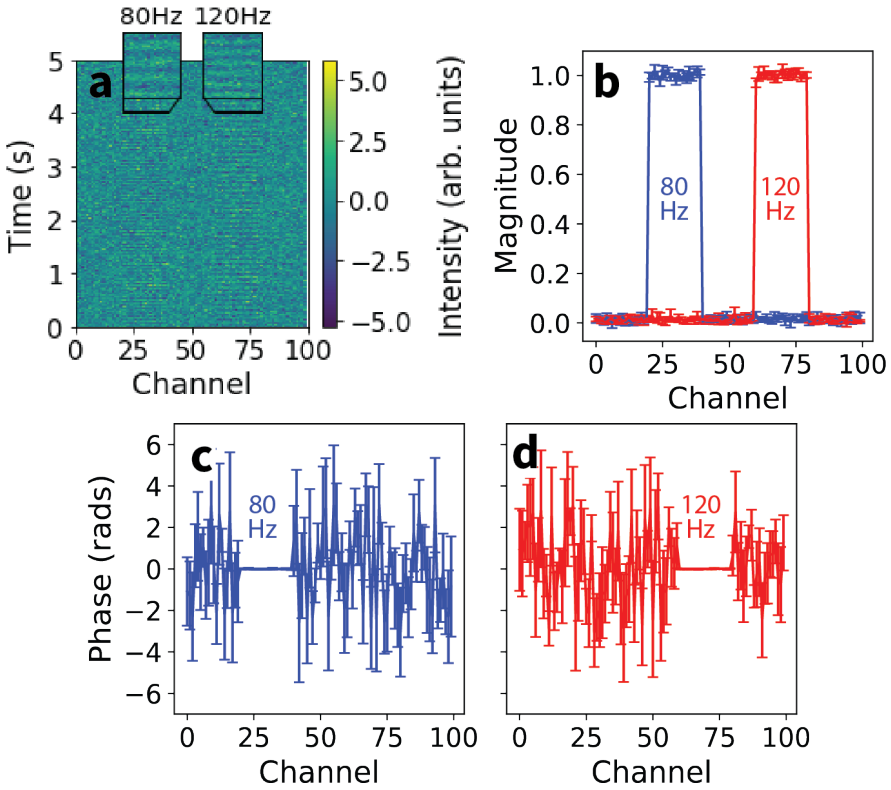

Figure 3. (a) Spectrogram of the simulated signal. The two zoomed insets show the noisy oscillating signals in channels 20-40 and 60-80, both signals oscillate at different frequencies. (b) The multi-frequency SILIA outputs. Both curves correctly identify the amplitudes of the 80 and 120 Hz signals at each channel. The output for each channel at the peaks are equal, within the estimated uncertainties, and the uncertainties are greater when there is no signal present. (c, d) Phase output when locking into both frequencies. We see that the phase error is very small in the presence of signal and large for all noisy channels.

The input consisted of sinusoidal signals with an amplitude of 1 oscillating at 80 Hz and 120 Hz, each in one of two groups of channels, and higher-amplitude noise distributed over all channels (Fig. 3a). SILIA suppresses noise and signals not oscillating at the reference frequency. When locking into the 80 or 120 Hz signal, we expect an output amplitude near 1 for channels 20-40 or 60-80 respectively, and near 0 everywhere

	Amplitude Error		Phase Error	
Frequency (Hz)	**Estimated**	**Actual**	**Estimated**	**Actual**
80	0.016	0.008	0.011	0.006
120	0.017	0.012	0.011	0.008

Table 1. A comparison of the errors estimated by SILIA with the actual standard deviation of SILIA output post-analyzed over the relevant channels.

else. When locking into the 80 and 120 Hz signals, we find mean peak values of 1.000 with standard deviations of 0.008 and 0.012 as well as mean outputs in channels without signal of 0.021 and 0.017 respectively, as expected (Fig. 3b). The error bars at the peak values for the 80 and 120 Hz results are, on average, 0.016 and 0.017 which are somewhat larger than the standard deviation of the output at peak signal (Table 1). The precision of the error estimates can be improved by increasing the sample size, which is presently set to 33% to achieve a reasonable estimate in a short computing time.

For the 80 and 120 Hz outputs in the channels having signal, we observe a phase of 0.004 and -0.001 respectively with standard deviations of 0.006 and 0.008 and mean estimated errors of 0.011 (Fig. 3c, d, Table 1). The phase of the results closely match an expected phase of 0 in the presence of signal and the error bars provide a reasonable estimate of the standard deviation in phase results. We also observe fluctuations in phase where signal is not present since SILIA can only extract noise in those channels (Fig. 3c, d).

4. Benchmarking

4.1. Methodology

For the initial benchmark, the amount of time is measured for the software to run for a variable number of input samples, channels and references by varying each of those three parameters individually while holding the other two constant. We use constant values of 4096 samples, 10 channels and 1 reference. This benchmarking allowed us to confirm the linear runtime of SILIA with respect to the data acquisition time, size of each sample and number of signals (Fig. 4).

We quantified the error of the output amplitude with respect to data acquisition time given a signal frequency of 100 Hz and a sampling rate of 2000 Hz. The input signal was a sinusoidal wave with an amplitude of 1 and additional Gaussian white noise with varying standard deviations. The results were averaged over 100 runs and fit to a function of the form, a/x^n where a and n were the fit parameters and x was the total number of samples collected (Fig. 5a, b). The fit was justified through additional simulations, where we verified that error was generally a function of the total number of samples and not of the sampling rate or data acquisition time individually.

When conducting experiments, noise tends to be time-correlated. To showcase how the lock-in method can take advantage of these time correlations for better noise

suppression, we simulated a sinusoidal signal with an amplitude of 1 and variable frequency. Gaussian pink noise[25] was added to the signal to simulate time-correlated noise. The results were averaged over 100 runs and fit to a function of the form, a/f^n where a and n were the fit parameters and f is the frequency of the input signal (Fig. 5c, d).

When locking into non-sinusoidal periodic inputs SILIA output should match the primary Fourier component of the input signal (Section 2). SILIA was challenged to lock into 100 Hz square waves with peak to peak height of 1 and varying duty cycles. After generating the signals and running the lock-in, we proceeded to confirm that our results matched the primary Fourier component of the signal (Fig. 5e).

To determine the response curve of the low-pass filter, a simulated signal frequency of 100 ± 5 Hz was varied however the reference frequency was constant at 100 Hz. The input signal was a sinusoidal wave with an amplitude of 1 and had no additional noise added to it.The lock-in output amplitude was plotted with respect to the input signal frequency with effective cutoffs at 0.06Hz and 1.02Hz (Fig. 5f).

4.2. Analysis

4.2.1. 4.2.1. Runtime Benchmarking The total runtime of the software rises linearly with increasing input channels (Fig. 4a). We see a similar trend with the number of frequency references, where adding an additional reference frequency to the SILIA input will increase the runtime of the software proportionally (Fig. 4b). These trends are attributed to the software performing the analysis separately and in sequential order for each channel and frequency reference. We also observe a general linear increase in runtime with respect to the number of input samples but see nonlinear fluctuations which result in increased runtimes when the number of input samples is not close to a power of 2 (Fig. 4c). This is likely due to our use of the radix-2 FFT algorithm, which has a runtime that is optimized for input sample sizes that are a power of 2 [26, 27]. As expected, we see that SILIA is slowest when asked to fit the reference signal and interpolate the input, and is significantly faster when the user skips those steps. It is also apparent that using a fitted reference is more costly than applying interpolation to the input signal (Fig. 4). This is due to the additional requirement of computing phase when the reference is fitted.

It is important to note that these benchmark plots do not necessarily reflect the exact runtime of the software in all applications, since those runtimes are dependent on the hardware used to run SILIA, future optimizations in the code, and the scale of the user's run. The observed linear trends imply that the post-processing time will generally be proportional to the data acquisition time.

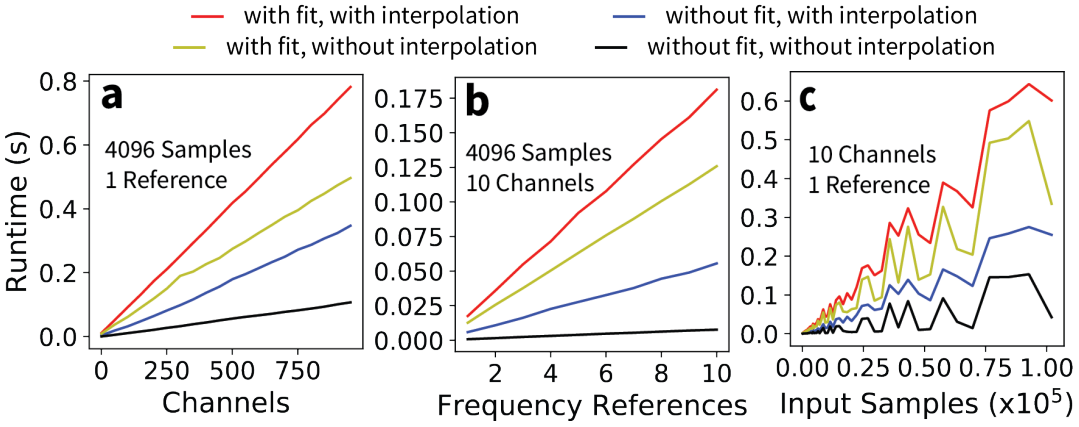

Figure 4. Dependence of the runtime of SILIA on the number of channels, input samples, and frequency references. (a, b) There is a linear dependence of runtime on the number of input channels and frequency references. (c) Shows a general linear relationship between the runtime and data acquisition time, but there is nonlinearity introduced by the slower runtime of the Fourier Transform when the number of input samples is not a power of 2.

4.2.2. 4.2.2. Error Benchmarking When benchmarking the effects of experimental errors, we focused on three potential sources of uncertainties - Gaussian or oscillatory noise in the input signal, the response of each low-pass filter, and non-sinusoidal signal. A low-pass filter response with slow roll-off can introduce unwanted artifacts from frequencies outside the cutoff range into the result of the lock-in amplification, and, as mentioned in Section 2, non-sinusoidal input can introduce bias into the result.

Fit Parameters	a		n	
SNR	0.25	0.01	0.25	0.01
Amplitude Error	5.60	118	1.00	0.98
Phase Error	5.83	161	1.00	1.01

Table 2. Fit parameters for the plots in Figures 5a (Amplitude Error), and 5b (Phase Error), for each of the set SNRs of 0.25 and 0.01. Data were fit to curves of the form a/x^n where a and n are the fit parameters. x is number of samples.

Fit Parameters	a		n	
SNR	0.25	0.01	0.25	0.01
Amplitude Error	0.025	0.55	0.96	0.95
Phase Error	0.025	0.61	0.99	0.98

Table 3. Fit parameters for the plots in Figure 5c (Amplitude Error), and 5d (Phase Error), for each of the set SNRs of 0.25 and 0.01. Data were fit to curves of the form a/f^n where a and n are the fit parameters. f is the frequency of the true signal.

Output error of lock-in amplifiers tends to decrease sharply with respect to data acquisition time and input SNR [23, 2, 28, 29]. We see a similar significant improvement

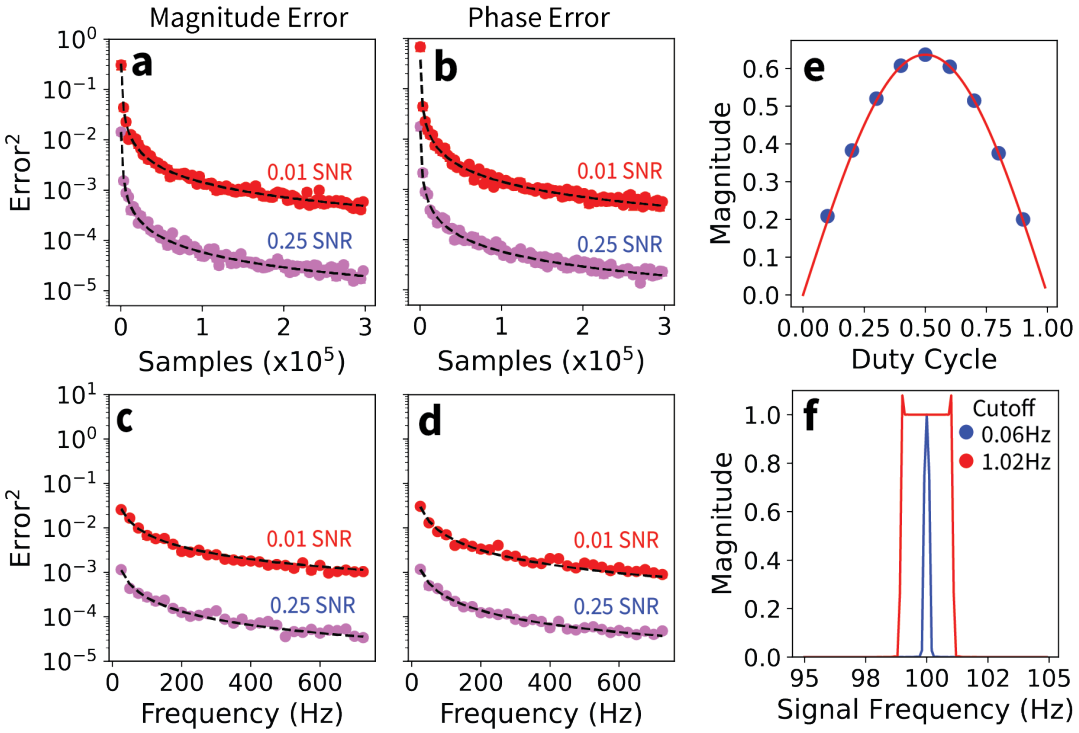

Figure 5. (a, b) Dependence of Mean Square Error of the output with respect number of samples of input signal. The black dashed lines are fitted curves. There is a steep decrease in amplitude error (Fig. 5a) as well as phase error (Fig. 5b). (c, d) SILIA output amplitude (Fig. 5c) and phase (Fig. 5d) increases in accuracy with respect to the frequency of the true signal when the noise is time-correlated. The dashed lines are fitted curves. (e) Output as a function of the duty cycle of a square wave input signal. SILIA outputs are fit to the red curve, which is the first term in the Fourier series expansion of a square wave with varied duty cycles. We calculated the coefficient of determination ($R^2 = 1.0$) using the residuals of a fit to the curve $2\pi \sin(\pi d)$, where d is the duty cycle. (f) Low-pass response curve for a constant reference frequency of 100 Hz.

in the SILIA output error with respect to a the number of samples of the input signal (Fig. 5a, b), and can conclude the necessity of longer data acquisition times and higher sampling rates for more accurate results, especially from data with low SNR. By fitting the error reduction to curves of the form a/x^n where x is variable and equal to the number of samples acquired, we find that the variance in the amplitude and phase of the lock-in result is approximately proportional to $1/x$ with proportionality constants that are a function of signal to noise ratio. (Table 2, Fig. 5a,b).

In addition, in the presence of more realistic time-correlated pink noise, output error decreases with respect to the frequency of the true signal (Fig. 5c,d). This result indicates that increasing the frequency at which an experiment is being done will likely decrease variance in SILIA output. Pink noise has a $1/f$ power spectrum, so we expect the variance in output to decrease at the same rate. That is consistent with what we observe in the fitted parameters (Table 3).

Furthermore, the shape of the input signal waveform affects the output in an

easily quantifiable manner. We can easily observe this by locking into square waves with different duty cycles using a fitted reference, the output closely matches the primary Fourier component of the input signal, as expected (Fig. 5e). Furthermore, the significant decrease in the output amplitude when the input oscillates outside the cutoff frequency range (Fig. 5f) demonstrates the effectiveness of SILIA at filtering out periodic noise.

5. Applications of SILIA

5.1. Spectral Analysis

In our first example application of SILIA, we measured light from a 532 nm diode-pumped solid state laser (ThorLabs CPS532) with high ambient background lighting, and measured the fluorescence spectrum emitted from a solution of rhodamine 6G (R6G) in ethanol excited at 532 nm.

5.1.1. 5.1.1. Methodology An Ocean Optics OCEAN-FX-XR1-ES spectrometer with a wavelength resolution of 0.42 nm on average and a standard deviation of 0.02 nm (0.42 ± 0.02 nm) was used to measure amplitude-modulated laser light, and the fluorescence spectrum emitted from R6G (95% purity) in ethanol ($\geq$ 99.5% purity, Sigma-Aldrich) at different concentrations (0.013, 0.13 and 1.3 mM, respectively). The laser pulses were generated from a laser (ThorLabs CPS532) with a continuous-wave power of 4.5 mW. To modulate the amplitude of the signal and measure a frequency reference, a Thorlabs MC2000B chopper that modulated the excitation laser at 50 or 100 Hz and was connected to a National Instruments USB6001 data acquisition device. The experimental setup is depicted in Fig. 6.

During the data acquisition, the reference input was measured with a 2000 Hz sampling rate from the chopper, which modulated the laser amplitude at 100 Hz for the laser spectrum measurement. The sampling rate of the spectrometer was 230 ± 49 Hz, with a 2 s runtime per data acquisition cycle for three cycles.

Fluorescence spectrum measurements were performed for 3 different concentrations (0.013, 0.13 and 1.3 mM) of R6G in ethanol. The fluorescence signal was coupled into an optical fiber and the spectrometer by a convex lens (focal length 250 mm) near the sample. The laser was modulated at 50 Hz and the reference input was measured from the chopper with a 2000 Hz sampling rate. For this measurement the sampling rate of the spectrometer was 249 ± 58 Hz, with a 3 s runtime per data acquisition cycle for three cycles.

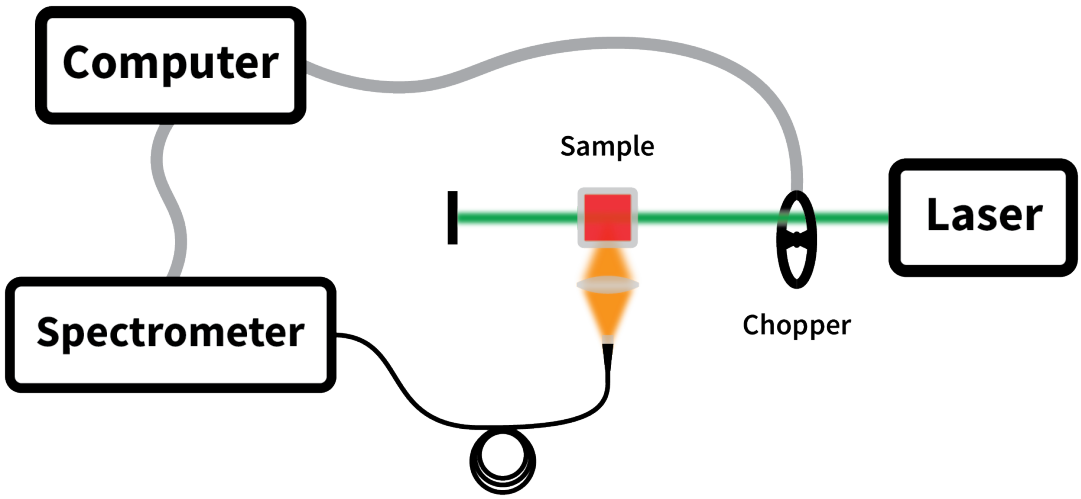

Figure 6. Experimental setup to measure the fluorescence spectrum of a R6G sample. A chopper wheel is used to modulate the laser which excites the R6G sample while a lens is used to focus the resulting fluorescence signal into the spectrometer. The chopper controller outputs a reference waveform, and both the spectrometer data and reference are acquired (National Instruments USB-6001) and recorded to a personal computer. To measure the spectrum of the excitation laser the R6G sample was replaced with a reflective diffuser.

5.1.2. 5.1.2. Results Figure 7a depicts the spectrum of the modulated (100 ± 1 Hz) laser light and ambient background measured with the spectrometer. Over the three data acquisition cycles, the 100 ± 1 Hz reference signal was a square wave with a peak to peak height of 1 that was fitted to sinusoidal functions by SILIA. The output amplitudes fluctuated with a standard deviation of 5.7% (Fig. 7c) for all 3 cycles, possibly due to irregular sampling by the spectrometer data acquisition (Fig. 7b). Figure 7c shows the retrieved laser spectrum, and the ambient white light background effectively removed from the measured spectrum, by SILIA. The retrieved phase difference between the reference and the measured spectrum is small around the 532 nm peak in the spectrum and much higher fluctuations in the other regions of the measured spectrum which are dominated by ambient white light background (Fig. 7d).

Typically, in fluorescence spectroscopy the use of photomultiplier tube is required to obtain sufficient signal from fluorescent samples. Here we apply SILIA to measure the fluorescence of solutions of R6G in ethanol for optical excitation at 532 nm without a photomultiplier, but with a much less sensitive spectrometer (Ocean Optics OCEAN-FX-XR1-ES). Since a photomultipler tube and/or low bandpass filter and/or monochromator was not used, the observed spectrum (Fig. 8a) contains spectral information from scatter photons of the pulsed 532 nm laser light and fluorescent signal of R6G onto the spectrometer. This pollution renders the fluorescent signal indistinguishable from the background signal. Running SILIA allows the fluorescent

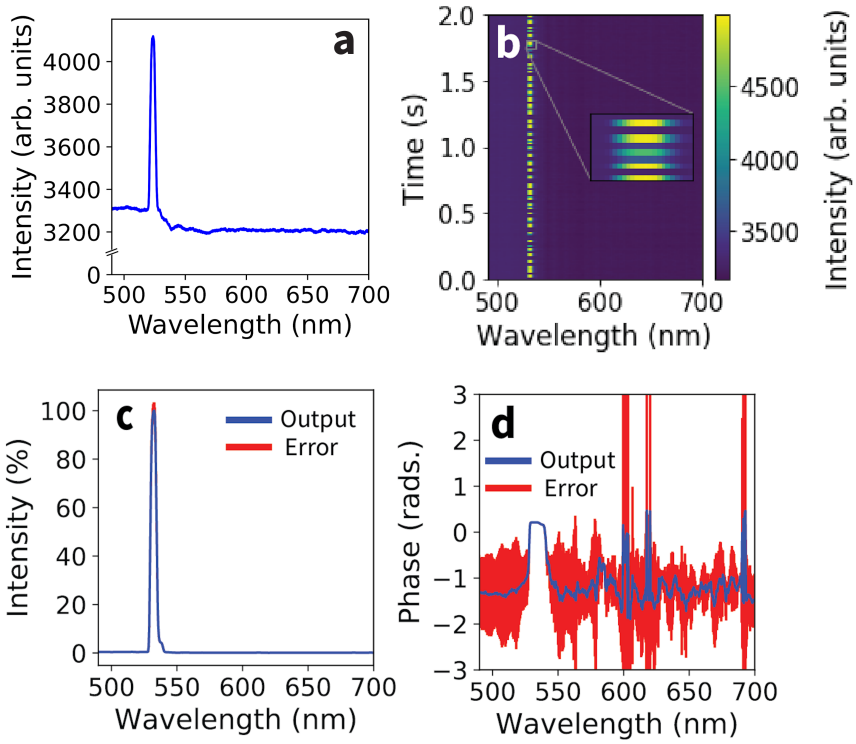

Figure 7. (a) The time-averaged signal of the spectrometer. The peak from the chopped laser signal can be seen at 532 nm. (b) A spectrogram of the chopped laser signal from a single data acquisition cycle of the spectrometer, showing the 100Hz oscillation of the signal input. Irregular sampling of light by the spectrometer leads to intensity variations of ±33% in laser peaks. (c) SILIA output amplitude averaged over three data acquisition cycles showcasing a clear laser signal. (d) SILIA output phase averaged over three data acquisition cycles demonstrating phase convergence in the presence of signal.

signal to be observed (Fig. 8b). Here we fit the reference signal to sinusoidal functions but did not use interpolation. The output closely matches a R6G fluorescence spectrum from the PhotoChem database [30, 31], shown as the green curve, and we observe that higher 1.3 mM concentration of R6G emits fluorescence at longer wavelengths due to the well-known primary inner filter effect[32]. The difference between the fluorescence spectra for the solutions having lower concentrations (0.013 mM and 0.13 mM) is likely due to the secondary inner filter effect.

In Fig. 8c, a simple background subtraction, being the difference between the 0.13 mM-R6G fluorescence spectrum with the chopper open, and the background with the chopper closed (lower black curve), is compared with the SILIA output (upper blue dash dot curve). Both are the average of 5 data acquisition cycles. The background subtraction has a standard deviation of 0.44, and the SILIA output has a standard deviation of 0.37, in the same arbitrary units for amplitude shown in Fig. 8c, confirming the accuracy and precision of SILIA. Unlike lock-in techniques such as SILIA, the simple background subtraction is prone to amplitude variations due to drifts in chopper phase

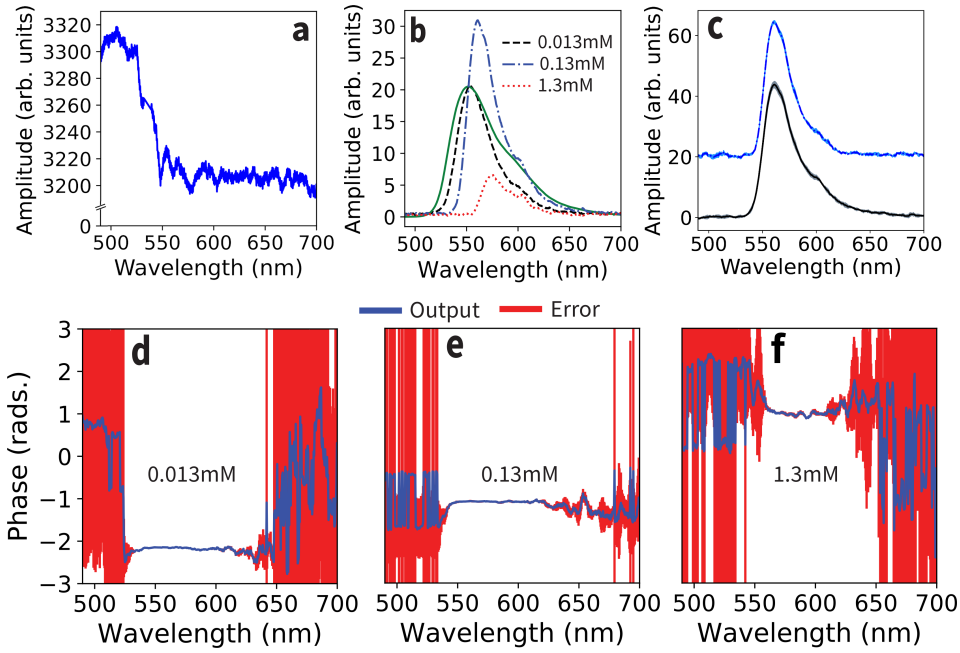

Figure 8. (a) Typical time-averaged signal of rhoadmine-6G measured by the Ocean-FC-XR1-ES spectrometer. The R6G fluorescence spectrum is indistinguishable from the laboratory background lighting and spectrometer dark noise. (b) SILIA output for different concentrations of R6G averaged over 3-5 data acquisition cycles. The spectral dependence on R6G concentration are ascribed to the primary and secondary inner-filter effects. The spectra are plotted against the green curve, a rescaled reference R6G fluorescence measurement taken using a Spex FluoroMax with an excitation wavelength of 480 nm and spectral bandwidth of 4.25 nm [30, 31]. The 0.013 mM results are in good agreement with the reference. (c) Comparison between SILIA output (upper blue dash-dot curve, offset for clarity) and the average chopper open-closed difference signal (lower black curve) for the R6G concentration of 0.13 mM. The standard deviation in the amplitude is shown as the shaded region bounding each curve, and is on the order of 1% for each peak. (d,e,f) SILIA phase output for the various concentrations of R6G. We observe a consistent phase convergence across all three concentrations in the presence of signal.

and laser performance, which were negligible in the present demonstration.

The phase difference between the signal and reference retrieved by SILIA (Fig. 8d,e,f) is almost constant over the wavelength channels displaying significant fluorescence signal, due to the laser modulation by the chopper at the reference frequency, and very noisy for the channels having no fluorescence. As expected, we also see a consistent phase convergence in the presence of signal and note that the convergence is stronger at lower R6G concentrations. The stonger phase convergence at lower concentrations of R6G is consistent with our observation those spectra are in better agreement with the reference spectrum.

5.2. Imaging Simulation

5.2.1. 5.2.1. Methodology SILIA is designed to work with signal inputs with an arbitrary number of channels. To demonstrate these capabilities for imaging applications, we used public domain images produced by fluorescence microscopy of bovine pulmonary artery endothelial (BPAE) cells (Fig. 9a) and human lymph node cells stained with DAPI (Fig. 9b) to simulate a series of identical images with fluctuating noise in lieu of a fluorescence microscopy measurement. Here we demonstrate that lock-in amplification by SILIA can be used on microscopy data to retrieve simultaneous fluorescence signals from noise [11], assuming the excitation source is modulated.

To simulate a simultaneous fluorescence signal, we multiplied the images by square waves with a peak to peak height of 1 so they would periodically flash on and off. The BPAE cells were modulated with a frequency of 100 Hz while the human lymph node cells were modulated at 75 Hz. We then overlayed both signals and added a significant amount of Gaussian noise with a standard deviation of 75 (RGB value) to each pixel. We took 500 total samples of this noisy, combined signal at 10 samples per cycle. The images were in a RGB format and had a size of 512x512 pixels which resulted in an input signal of 786432 channels. Each RGB value in the image can range from 0 to 255, where the values of (0, 0, 0) represents a black color and (255, 255, 255) represents a white color.

5.2.2. 5.2.2. Results We seek to recover the original microscopy images (Fig. 9a,b) from the combined and noisy signal (Fig. 9c,d) by applying SILIA. Since the input signals are square waves with a 50% duty cycle, the raw SILIA output was approximately $2/\pi$ times the original RGB values and was then scaled by $\pi/2$ to recover the original image (Fig. 9c). Since RGB values range from 0 to 255, the scaled output was capped at 255. The difference between the original and amplified images and found to be small, with an average value of 8.56 ± 6.00 (RGB value) for signal 1 (Fig. 9a) and 7.36 ± 5.85 (RGB Value) for signal 2 (Fig. 9b). SILIA took about 10 minutes to compute this output on a laptop computer with a 2.7 GHz Intel i7 CPU and 16.0 GB RAM.

6. Discussion

SILIA is an inexpensive and general alternative to multi-channel and multi-frequency lock-in amplifiers. Some advantages in the present implementation include convenient capabilities to estimate error in lock-in results, simplicity compared to other digital lock-in amplifiers, and the flexibility to extract periodic signals from data sources having an arbitrary number of channels. For example, the code is readily adaptable to a frequency sweep measurement [18] by using a frequency modulated reference signal. SILIA is also advantageous to using simpler computational error reduction methods such as background subtraction and averaging because it has the ability to extract phase information and can lock into multiple overlayed signals (Fig. 9).

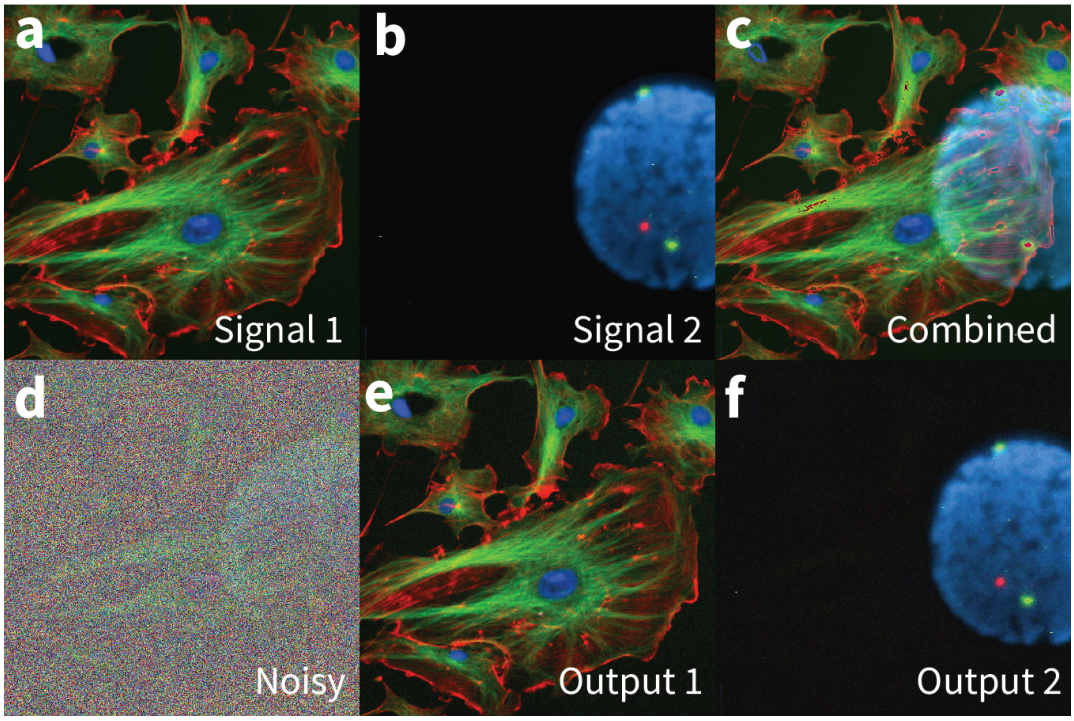

Figure 9. Simulated clean fluorescence signal of (a) a public domain image of BPAE cells with stained nuclei and marked microtubules, (b) a public domain image of a human lymphnode nucleus stained with DAPI, and (c) when both images are fluorescing. (d) Combined fluorescence signal with noise. (e,f) SILIA recovered the original fluorescence images despite significant noise. The result is scaled by $\pi/2$ to compensate for diminished output when locking into a square wave signal with a fitted, sinusoidal reference.

In addition, since the the lock-in procedure is implemented as a post-processing step, measured data can be preprocessed prior to performing the lock-in operation, and SILIA can be run multiple times, e.g. with adjusted input parameters, on the same data.For example, by artificially scaling the time axis of the reference prior to lock-in, harmonics of the reference frequency can also be applied to recreate a Fourier series approximation to the exact signal waveform in each channel. SILIA could also be employed to extract periodic features from video data without a separate reference input by using a reference pixel [33].

Some limitations of using this implementation include the potentially significant runtime of the software when analyzing large datasets. The software currently has a linear runtime with respect to the number of channels and frequency references (Fig. 4a, b), which can be reduced through the parallelization of SILIA with respect to each channel and frequency reference. In addition, the Python packages Numba or Dask could be used to speed up numerical calculations in the future. There are also superior techniques to the optional interpolation procedures for the reference input, which can be implemented to mitigate error from the FFT due to inconsistent sampling rates [34, 35, 36]. Future developments could avoid FFT altogether using, for example, lock-

in frequency measurement[37]. SILIA is also limited in the size of the dataset it can process since it loads all the input data into memory. However, this can be circumvented in future implementations by including methods which take advantage of big data tools.

7. Conclusions

We have developed and demonstrated the software package SILIA that effectively suppresses background noise from periodic features in datasets with an arbitrary number of channels and frequency references. By adopting a paradigm where lock-in amplification is implemented as a post-processing step, we find a higher degree of flexibility than available in other methods while providing accurate results for modulated multi-channel measurements collected with common data acquisition approaches. We anticipate broad potential applications in spectroscopy, microscopy and imaging technologies. Editing SILIA is straightforward, and we encourage users to apply, improve and customize the code to suit their needs.

Acknowledgements

This material is based upon work performed at the Lawrence Berkeley National Laboratory and was supported by the U.S. Department of Energy, Office of Science, Division of Chemical Sciences of the Office of Basic Energy Sciences under Contract DE-AC02-05CH11231.

References

[1] Kishore K and Akbar S A 2020 *IEEE Sensors Journal* **20** 10345–10354
[2] Mandelis A 1994 *Review of Scientific Instruments* **65** 3309–3323
[3] Wang X 1990 *Review of Scientific Instruments* **61** 1999–2001
[4] Stimpson G A, Skilbeck M S, Patel R L, Green B L and Morley G W 2019 *Review of Scientific Instruments* **90** 094701
[5] Giaconia G C, Greco G, Mistretta L and Rizzo R 2017 *Electronics* **6** 18
[6] Carminati M, Gervasoni G, Sampietro M and Ferrari G 2016 *Review of Scientific Instruments* **87** 026102
[7] Lu J, Pan D A, Yang B and Qiao L 2008 *Measurement Science and Technology* **19** 045702
[8] Liu Y, Shen Y, Ma C, Shi J and Wang L V 2016 *Applied Physics Letters* **108** 231106
[9] Probst P and Jaquier A 1994 *Review of Scientific Instruments* **65** 747–750
[10] Foix S, Alenya G and Torras C 2011 *IEEE Sensors Journal* **11** 1917–1926
[11] Marriott G, Mao S, Sakata T, Ran J, Jackson D K, Petchprayoon C, Gomez T J, Warp E, Tulyathan O, Aaron H L, Isacoff E Y and Yan Y 2008 *Proceedings of the National Academy of Sciences* **105** 17789–17794
[12] Bourquin S, Prasankumar R P, Kärtner F X, Fujimoto J G, Lasser T and Salathé R P 2003 *Optics Letters* **28** 1588
[13] Kolarczik M, Ulbrich C, Geiregat P, Zhu Y, Sagar L K, Singh A, Herzog B, Achtstein A W, Li X, van Thourhout D, Hens Z, Owschimikow N and Woggon U 2017 *APL Photonics* **3** 016101
[14] Gilburd L, Xu X G, Bando Y, Golberg D and Walker G C 2016 *The Journal of Physical Chemistry Letters* **7** 289–294

[15] Fushitani M 2008 *Annu. Rep. Prog. Chem., Sect. C: Phys. Chem.* **104** 272–297
[16] Fischer M C, Wilson J W, Robles F E and Warren W S 2016 *Review of Scientific Instruments* **87** 031101
[17] Dong P T and Cheng J X 2017 *Spectroscopy (Santa Monica)* **32** 24–36
[18] Sonnaillon M O and Bonetto F J 2007 *Review of Scientific Instruments* **78** 014701
[19] Amrut Nadgir, Richard Thurston 2021 SILIA URL `https://github.com/AMOS-experiment/SILIA`
[20] Scofield J H 1994 *American Journal of Physics* **62** 129–133
[21] Temple P A 1975 *American Journal of Physics* **43** 801–807
[22] Smith J O (accessed 7 August, 2021) *Spectral Audio Signal Processing* (`http:`http://ccrma.stanford.edu/ jos/sasp/`//ccrma.stanford.edu/~jos/sasp/`) online book, 2011 edition
[23] Bhattacharyya S, Ahmed R N, Purkayastha B B and Bhattacharyya K 2016 *Journal of Physics: Conference Series* **759** 012096
[24] Nasser M Abbasi 2011 Fourier Series Coefficients of a Rectangular Pulse Signal
[25] Timmer J and Koenig M 1995 *Astronomy and Astrophysics* **300** 707
[26] Muqri M R, Wilson E J and Shakib J 2015 A Taste of Python – Discrete and Fast Fourier Transforms *2015 ASEE Annual Conference and Exposition* pp 26.123.1–26.123.11 ISBN 978-0-692-50180-1
[27] Amirfattahi R 2013 *Journal of Medical Signals & Sensors* **3** 217
[28] Van Baak D A and Herold G 2014 *American Journal of Physics* **82** 785–797
[29] Neelakantan K, Dattagupta S and Rajappan K P 1980 *Review of Scientific Instruments* **51** 251–252
[30] Taniguchi M and Lindsey J S 2018 *Photochemistry and Photobiology* **94** 290–327
[31] Taniguchi M, Du H and Lindsey J S 2018 *Photochemistry and Photobiology* **94** 277–289
[32] Kumar Panigrahi S and Kumar Mishra A 2019 *Journal of Photochemistry and Photobiology C: Photochemistry Reviews* **41** 100318
[33] Alinovi D 2018 *ELCVIA Electronic Letters on Computer Vision and Image Analysis* **16** 9
[34] Barnett A H, Magland J and af Klinteberg L 2019 *SIAM Journal on Scientific Computing* **41** C479–C504
[35] Ying L 2009 *SIAM Journal on Scientific Computing* **31** 1678–1694
[36] Candès E, Demanet L and Ying L 2009 *Multiscale Modeling and Simulation* **7** 1727–1750
[37] Lu J 2020 *Review of Scientific Instruments* **91** 075106