Image Processing Techniques applied to Spectroscopy for Nanophotonic Circuit Characterization

Devin Dean

Abstract—Images and videos are an essential part of modern day life. Roughly six billion CMOS image sensors were manufactured in 2020 alone. Computational Imaging describes the algorithms and techniques used to process the often 2-dimensional images taken by these and other imagers. Many algorithms are routinely applied for tasks such as denoising, deconvolution, and data reduction. We show that many of these algorithms are useful in a different area of optical sensing - spectroscopy. Whereas imaging captures the spatial profile of light emanating from a 2D scene, spectroscopy captures the spectral profile of light emanating from a point. Spectroscopy is useful in characterizing nanophotonic waveguides. Nanophotonics is a rapidly growing field that broadly refers to wavelength scale optical devices. Here we show that computational imaging algorithms are useful in characterizing fabricated waveguide circuits.

Index Terms—Computational Imaging, Spectroscopy, Nanophotonics

1 Introduction

There are many similarities and a few key differences between conventional 2D imaging and spectroscopy that affect how well computational imaging techniques from the former can be applied to the latter. The most important similarity between imaging and spectroscopy is that both employ photodetectors and therefore operate by the same physics. This means that the noise sources will be similar - each having contributions from photon noise, electronic noise, and quantization noise. It also means that the limitations will be similar - both have a time-bandwidth tradeoff for acquisition rate and also a fixed dynamic range for readable signal levels. Therefore, computational imaging techniques such as denoising and high-dynamic-range imaging can be directly applied to spectroscopy measurements.

The key difference between imaging and spectroscopy lies not in the hardware, but rather the data itself. The imaging data is inherently two-dimensional and therefore computational imaging algorithms often involve either operations with two-dimensional matrices or workarounds to avoid the computational/memory complexity associated with basic two-dimensional operations. Spectroscopy data, on the other hand, is inherently one-dimensional and therefore may not be limited by the same computational and memory issues faced by image processing.

Furthermore, the spatial features common in 2D images are quite different from the spectral features of 1D spectroscopy data, which implies different priors should be used for post-processing. For example, a common prior for 2D images is that gradients are sparse. However, reasonable priors for spectroscopy data is that the data is smooth and continuous, and features are often periodic. Therefore, minor adjustments may be necessary when applying computational imaging algorithms to spectroscopy data.

To summarize, imaging and spectroscopy have many hardware similarities and only slight differences in data. We therefore conclude that computational imaging techniques are relevant to spectroscopy measurements. In the next section, we focus on a particular kind of spectroscopy measurement: nanophotonic circuit characterization.

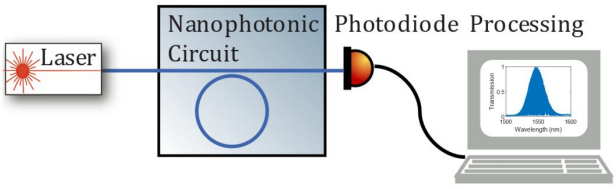

Fig. 1. Nanophotonic Circuit Measurement Setup. The transmission of a tunable laser through the nanophotonic circuit is measured as a function of wavelength.

2 Nanophotonic Circuit Characterization

To characterize fabricated waveguide devices, one typically shines a laser on one end of the waveguide and collects all light output from the other end of the waveguide with a single pixel photodetector [1], [2], [3]. The photodetector voltage is measured as a function of laser wavelength - providing the spectrum of the waveguide device. In this report, we use data from a device fabricated and measured by Luke Qi on the nanophotonic platform of thin-film lithium niobate [2].

A typical spectrum contains both slow and rapidly varying features, as shown in Fig 2. Resonators produce periodic sharp dips in transmission or “resonances”. The spacing between consecutive resonances is known as the free spectral range (FSR), and is equal to one divided by the time it takes light to travel around the resonator. Although the width, minimum, and FSR of the resonances can have some wavelength dependence, the background transmission of an

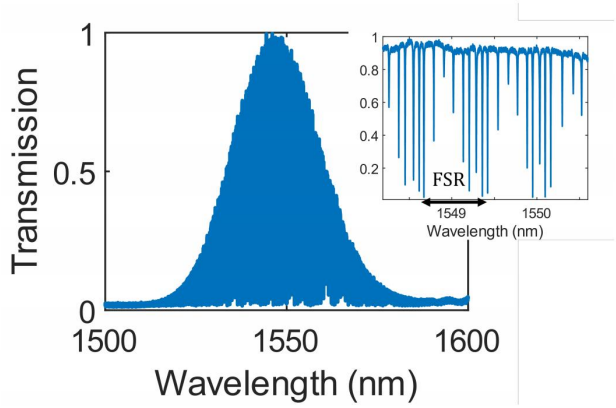

Fig. 2. Transmission Spectrum. The Gaussian-shaped background arises from the strong wavelength dependence of the grating couplers used to get light into the nanophotonic circuit, while the eight periodic dips (visible in inset) arise from the circuit itself - the resonances of the coupled racetrack resonators.

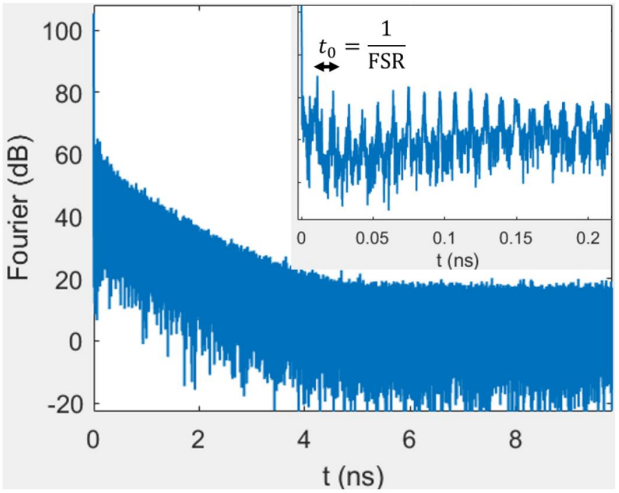

Fig. 3. Discrete Fourier Transform: Time Domain Representation. The Fourier transform of the frequency spectra yields the time domain. Inset: zoomed in on the low-time signal.

ideal resonator is near unity. For example, there is a background envelope to the transmission depending on how the light is coupled onto the nanophotonic chip. If the light is coupled by end-facets, then small reflections off the input and output waveguide will lead to a sinusoidal background ripple superimposed on the actual nanophotonic circuit transmission spectrum. If the light is coupled by a grating, then the wavelength-dependence of the grating leads to a Gaussian window in the transmission spectrum, as shown above.

There are three primary challenges in characterizing nanophotonic circuits. The first challenge is to resolve small spectral features despite the presence of noise. The second challenge is to separate out the spectral features of interest from the background spectral features. The third challenge is to extract useful information from the spectral features of interest.

In the following three sections, we apply computational imaging techniques to address each of the three challenges in nanophotonic circuit characterization. In all of our analysis we utilize the discrete Fourier transform of the frequency data - an alternative representation in the "time" domain. The discrete Fourier transform has been utilized in nanophotonic circuits previously to locate reflections in space [4], but has not to our knowledge been used extensively otherwise.

In the time domain, as shown in Fig. 3, the background spectrum appears near zero time, while the periodic features appear at peaks corresponding to multiples of one divided by their repetition rate, up until the Nyquist time.

Denoising and background extraction are implemented by low-pass filtering the time domain, and circular convolutions are implemented by multiplying vectors in the time domain.

3 Denoising

The total noise in a photodetector signal is a combination of photon noise inherent to the light, electronic noise from thermal fluctuations within the detection circuit, and quantization noise from the ADC. In a typical nanophotonic circuit measurement setup, electronic amplifiers boost signals above the quantization noise floor so that the quantization noise becomes negligible. Photon noise is power dependent (scaling linearly with power by shot noise and quadratically with power by technical noise), while electronic noise is independent of power. For high powers, therefore, the electronic noise becomes negligible compared to the photon noise, but both of these in fact become small compared to the high-power signal - the signal-to-noise-ratio (SNR) is high at high powers. At low powers, electronic noise dominates over photon noise, and the SNR is low. Because electronic noise is the dominant noise source when SNR is low, it is the only noise source we consider here. We can model the electronic noise as Gaussian-distributed zero-mean voltage noise on the photodiode signal. We can directly measure the variance of the voltage noise (without any incident optical power), and therefore calculate the electronic-noise-bounded SNR as a function of power.

There are numerous methods to denoise photodetector data. One could use a linear local averaging technique such as a moving average filter or a low-pass filter, a nonlinear local averaging technique such as max-pool operation, a nonlocal averaging technique such as nonlocal-means, a denoising neural network, or simply take multiple measurements and average them all pointwise together. Some of these methods are not feasible here. Taking multiple measurements poses the challenge of aligning the spectra correctly between different scans - the laser scan is not very repeatable due to vibrations and fluctuations. A neural network requires training data and operates as a "black box" and therefore requires extensive testing to ensure good performance. We cannot use a pre-trained denoiser for images because of the prior differences between images and spectra discussed in the previous section. The nonlocal averaging technique is attractive because of the repetition that often occurs in spectra, but the nonlocal nature casts doubt on

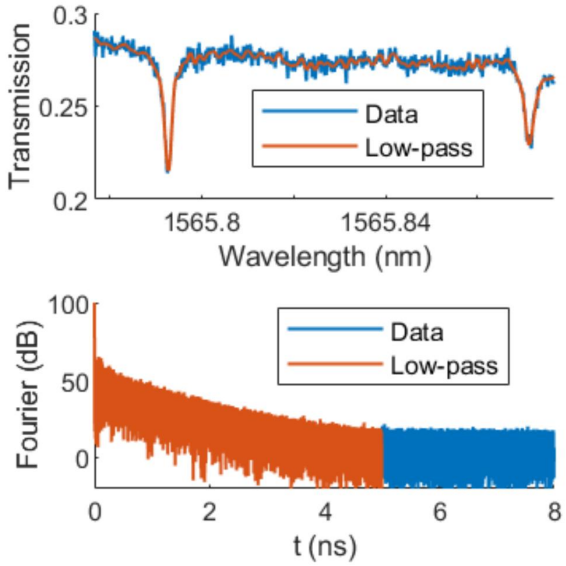

Fig. 4. Low-pass filtering for noise reduction. Bottom: Time-domain representation, showing the low-pass is identical to the data up until the cutoff time $\tau_{cutoff} = 5$ ns. Top: Resulting low-pass signal is visibly less noisy compared to the original data.

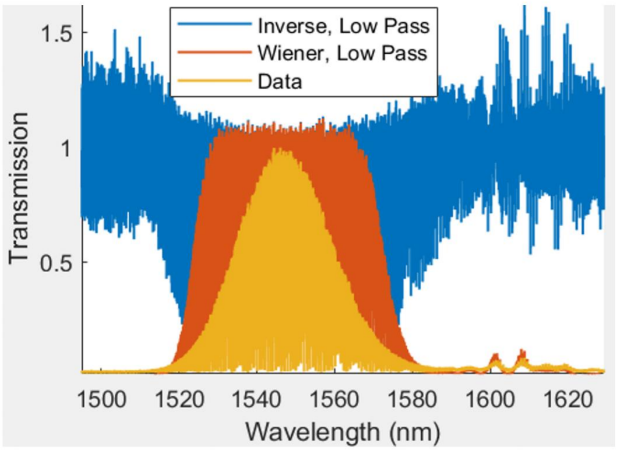

Fig. 5. Inverse Filtering for Feature Extraction. Measured power (yellow) has background go to zero at edges. Resulting normalized transmission (blue) amplifies noise at edges. Including an SNR term (red) successfully normalizes transmission where possible, but does not amplify noise at edges.

the scientific validity of the result since this may obscure local features superimposed on a periodic feature. Given the above challenges, here we employ only local averaging techniques to remove noise.

The conventional method to denoise data is with a simple moving average filter, which amounts to multiplication by a sinc in the time domain. Because of the peaks of the sinc, this method is sub-optimal at denoising and could distort data.

Instead, here we remove high-frequency noise with a simple low-pass filter. The cutoff time of the filter was carefully chosen to smooth the signal without significantly affecting its sharp features. It was found empirically that the optimal cutoff time matches that where the signal drops below the electronic noise floor in the time domain. The time domain spectra clearly shows a decaying signal with time up until about 5 ns, after which the spectra becomes flat - this is the time regime dominated by electronic noise. It therefore is quite logical to filter out the noise-dominated portion of the time domain spectra. This is an important realization, since in practice datasets are taken at different wavelength scan speeds, data sampling rates, and nanophotonic circuits have different characteristic time scales. With this knowledge we can deterministically low-pass filter any nanophotonic circuit data without compromising the signal, simply by observing the location of the electronic noise floor in the time domain.

4 Inverse Filtering for Feature Extraction

When multiple waveguide components are cascaded, their transfer functions become multiplied. This is nothing but the convolution theorem, except we are directly measuring and working within the frequency domain. Therefore, to isolate the transfer function of a resonator, we need to divide by the transfer function of the other components on the chip. In this section, we successfully perform this separation using a prior and Wiener deconvolution.

We can use the prior information that the transfer function of a resonator $T_{resonator}(\lambda)$ has unity transmission off resonance to isolate its transfer function from the rest of the chip's $T(\lambda)$. The slowly varying background $T_{background}(\lambda)$ can easily be separated from the sharp resonance dips using the low-pass filtering techniques described in the previous section, only with a much smaller cutoff time less than the resonator round trip time. We compared the linear local low-pass filtering method to a nonlinear max filtering method to extract the background, and found that low-pass filtering works best to obtain the background transfer function. The resonator transfer function can then be obtained by dividing the total spectrum by the background transfer function, as shown in Eq. 2.

$$T(\lambda) = T_{background}(\lambda) * T_{resonator}(\lambda) \tag{1}$$

$$T_{resonator}(\lambda) = \frac{T(\lambda)}{T_{background}(\lambda)} \tag{2}$$

$$T_{resonator}(\lambda) = \frac{T(\lambda)}{T_{background}(\lambda) + \delta V} \tag{3}$$

However, as in inverse filtering, the division amplifies noise where the background goes to zero. To highlight only the regions with high SNR, we include a noise term δV similar to Wiener filtering, as shown in Eq. 3. The result is a "window" of the resonator transfer function, only displayed where the SNR is high.

In Fig. 5, the data shows a strong background due to the wavelength-dependent transmission of the grating couplers on the nanophotonic chip. Simply dividing by the slowly-varying curvature amounts to inverse filtering (shown in blue), but has the downside of amplifying noise where the background transmission goes to zero. Indeed, we

observe that the transmission becomes nonsensical in these regions. The only reason it doesn't blow up to infinity is because there is a dark current in the photodetector data that leads to a small offset in the data. In practice, this would be subtracted before analyzing the data. We also plot the result of Wiener filtering the data, with SNR calculated based on the background level compared to the electronic noise (measured without any optical power). Note that the background compared to the noise is used rather than the true signal compared to the noise since we are dividing by the background and not the signal - therefore the SNR term is only needed to prevent noise amplification when the background goes to zero.

5 Extracting Quality Factor from Data

While denoising and deconvolutions are important first steps, the main goal of nanophotonic circuit characterization is to extract parameters from the data. The quality factor of resonances (related to the linewidth), for example, is a key metric in circuit performance. Conventionally, resonances are identified using a find-peaks function and then a Lorentzian fit is performed around each resonance to extract the linewidth. Drawbacks to this direct fitting approach include the fact that runtime scales linearly with number of resonances, a substantial fraction of resonances cannot be properly fit, and that the approximate width of the resonance is required in order to determine the fitting window width. In this section, we propose instead to use the wavelet transform to estimate the linewidths of all modes simultaneously, without needing to perform Lorentzian fits.

The wavelet transform is a technique very similar to the windowed fourier transform, and is commonly applied in signal and image processing (such as in JPEG 2000) [5], [6]. In it, basis functions (wavelets) with different widths are convolved with the target function. Peaks appear in the convolution wherever the wavelet width matches the local features of the target function. The linewidth of each resonance can then be easily extracted by reading off the width of the wavelet that resulted in the highest peak.

Here, we create a wavelet by combining two lorentzians, one positive and one negative, with equal areas and slightly different widths. Empirically, the best results were achieved for pairs of Lorentzians with nearly equal widths and zero relative offset. Zero net area is important to ensure the wavelet transform does not scale with the background of the data.

To demonstrate the operation of the wavelet transform, we perform it on a simple simulated resonance. The resonance consists of a narrow dip with a linewidth that relates to the Q. Wavelets with different widths are then convolved with the data. Peaks in the wavelet transform occur wherever the wavelet width matches the resonance width, indicating the Q. The data, wavelets, and wavelet transform are shown in the figure below on the left.

Once we validated that the wavelet transform correctly identifies the quality factor of a resonance, we performed it on all 441 modes in the data simultaneously, for 40 different wavelet widths. We compared the performance of our wavelet linewidth extraction to that of fitting all peaks to Lorentzians with experimental data, as shown in Fig. 7.

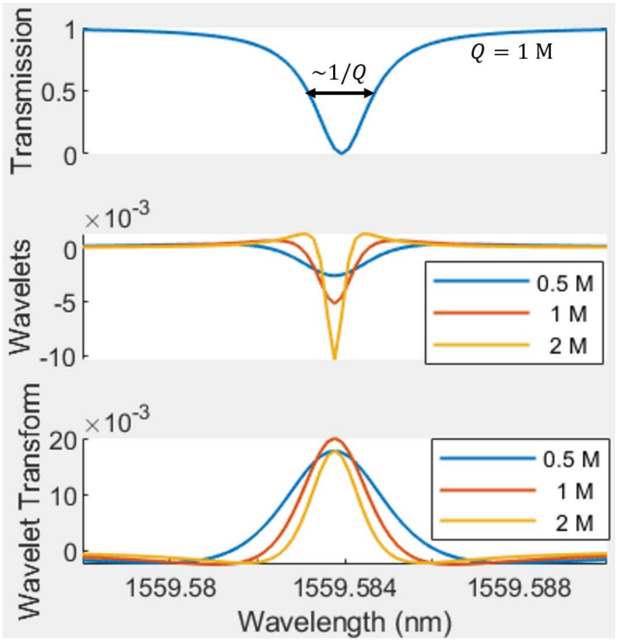

Fig. 6. Example Quality Factor Extraction. Top: Simulated resonance with quality factor $Q = 1$ M. Middle: Three wavelets constructed to have widths corresponding to $Q = 0.5$ M, 1 M, 2 M. Bottom: Resulting convolution between data and wavelets, clearly showing a peak at the correct $Q = 1$ M wavelet.

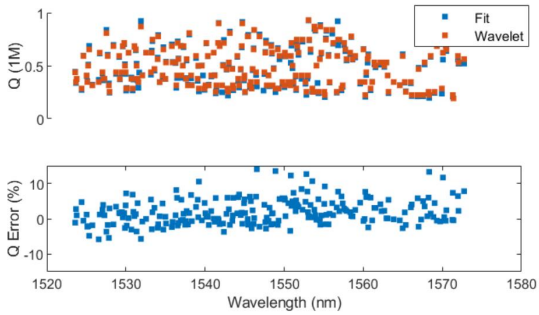

Fig. 7. Quality Factor Extraction from Data. Top: Quality factors as measured with (blue) direct fitting and (red) our wavelet transform approach. Bottom: Difference in quality factor between the two approaches.

The performance is within 5%, which is accurate enough for most purposes and could be increased simply by increasing the number of wavelets used.

One benefit of using wavelets is that the scale is not predetermined - width features from very low quality factor of Q $200,000$ up to very high quality factor Q $1,000,000$ were successfully extracted from the data. Another benefit is that it is more robust than direct fitting. While the wavelet transform produced an estimate for all 441 modes, direct fitting was only able to successfully fit 240 of them. The rest were either not fit at all or produced low-quality fits that had to be discarded. However, this warrants closer investigation. It is entirely possible that the data contains complex mode shapes or other features that should not be considered for analysis - in which case they shouldn't be fit at all.

6 Conclusion

To summarize, we successfully applied techniques from computational imaging, such as denoising, deconvolution, and convolution, to characterize nanophotonic circuits. Furthermore, because of the similarities between spectroscopy and imaging, we believe that many more techniques can be carried over, such as high dynamic range imaging and neural network analysis.

However, one key difference is that while the goal of everyday imaging is often to produce images that look "nice" and "realistic", the goal of spectroscopy is much more quantitative and therefore more similar to scientific imaging. It is often a best practice to show plots with raw data, as opposed to smoothed or otherwise filtered data, in order to minimize the potential that any physical effects be unintentionally obscured in the results. It is therefore important to carefully validate and justify the application of the different computational imaging algorithms to spectroscopy data. This report provides a starting point for that, and hopefully in the future more algorithms from computational imaging will be examined and implemented for use in spectroscopy.

Acknowledgments

The author would like to thank Luke Qi for providing the transmission data for the nanophotonic circuit known as ARB03.

Part of this work was performed at the Stanford Nano Shared Facilities (SNSF), supported by the National Science Foundation under award ECCS-2026822. Work was performed in part in the nano@Stanford labs, which are supported by the National Science Foundation as part of the National Nanotechnology Coordinated Infrastructure under award ECCS-2026822.

References

[1] T. Herr, V. Brasch, J. D. Jost, I. Mirgorodskiy, G. Lihachev, M. L. Gorodetsky, and T. J. Kippenberg, "Mode spectrum and temporal soliton formation in optical microresonators," *Phys. Rev. Lett.*, vol. 113, p. 123901, Sep 2014. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.113.123901

[2] M. Zhang, C. Wang, R. Cheng, A. Shams-Ansari, and M. Lončar, "Monolithic ultra-high-q lithium niobate microring resonator," *Optica*, vol. 4, no. 12, pp. 1536–1537, Dec 2017. [Online]. Available: https://opg.optica.org/optica/abstract.cfm?URI=optica-4-12-1536

[3] Y. He, H. Liang, R. Luo, M. Li, and Q. Lin, "Dispersion engineered high quality lithium niobate microring resonators," *Opt. Express*, vol. 26, no. 13, pp. 16 315–16 322, Jun 2018. [Online]. Available: https://opg.optica.org/oe/abstract.cfm?URI=oe-26-13-16315

[4] B. J. Soller, D. K. Gifford, M. S. Wolfe, and M. E. Froggatt, "High resolution optical frequency domain reflectometry for characterization of components and assemblies," *Opt. Express*, vol. 13, no. 2, pp. 666–674, Jan 2005. [Online]. Available: https://opg.optica.org/oe/abstract.cfm?URI=oe-13-2-666

[5] C. Torrence and G. P. Compo, "A practical guide to wavelet analysis," *Bulletin of the American Meteorological Society*, vol. 79, no. 1, pp. 61 – 78, 1998. [Online]. Available: https://journals.ametsoc.org/view/journals/bams/79/1/1520-0477_1998_079_0061_apgtwa_2_0_co_2.xml

[6] M. Taubman, David; Marcellin, *JPEG2000 Image Compression Fundamentals, Standards and Practice: Image Compression Fundamentals, Standards and Practice*. Springer Science Business Media, 2012.