Per-Pixel Time Constant Measurement of Bolometer Cameras

Dennis L. Waldron and Dieter J. Lohrmann

***Abstract*— A novel system to measure the time constant of microbolometer cameras per-pixel to create a 2-D time constant map is presented. The measurements take place at the camera level and do not require any special control of the camera beyond the ability to capture raw data. Detailed simulations are run for a broad range of variables and values to demonstrate the usefulness of the system in extracting even extremely fast time constants with high accuracy and repeatability. Measurements are taken on a proof of concept system with an example camera, and a 2-D time constant map is reported. These results are compared with measurements from a linear motion stage system.**

***Index Terms*— Bolometer, camera, focal plane array (FPA), long-wave infrared (LWIR), microbolometer, per-pixel, sensor, time constant.**

I. Introduction

MICROBOLOMETER sensor performance is frequently quantified by a figure of merit (FOM). FOM is the product of the noise equivalent difference temperature (NEΔT) and its thermal time constant (τ); it is expressed in units of mK·ms. NEΔT is a measure of sensitivity, while τ is a measure of response time or blur. One can typically achieve a given FOM with certain process technology, and NEΔT and τ can be traded against each other to get a more sensitive or faster responding detector. (Since FOM roughly scales with the area of the pixel, an alternate FOM would additionally include pixel area in its product [1].) Different applications may put a premium on either NEΔT or τ; a lower (faster) τ detector is useful for on-the-move applications, such as an unmanned aerial system payload, while greater sensitivity is useful for finding low-contrast targets at greater range.

It is important to know both NEΔT and τ not only for system applications but also for research, development, and process troubleshooting when fabricating detectors. This article will focus on a novel method of measuring time constant for each pixel individually in the focal plane array (FPA) at the system level; to the best of our knowledge, this has never before been reported.

There are many reported methods of measuring τ depending on the level of control one has over the system or component. One common method for test structures is to electrically excite the bolometer array with a controlled current to raise the temperature through Joule heating and simultaneously measure the voltage across the device [2]–[5]. The device resistance is then used to calculate a temperature change over time based on the knowledge of the thermal coefficient of resistance (TCR) of the device material. This method has the advantage that τ for each pixel can be recovered though the result is typically averaged across the array to reduce error. The temperature of the bolometers can alternatively be measured directly after Joule heating with a fast thermal camera [6]. In either case, τ is found by fitting the rise or decay of the bolometer temperature to an exponential equation

$$y = a + b \exp\left[\frac{c - t}{\tau}\right] \tag{1}$$

typically with some least square method, where y is the value measured (counts, voltage, temperature, and so on), a is an offset or asymptotic level, b is related to the amplitude, c is an offset in time, and t is the time.

In another method, the phase difference between the current and voltage under a sinusoidal input can be used. The phase lag will be zero as the frequency approaches dc or infinity and maximum at a frequency related to τ [7].

Without direct electrical control, as assumed earlier, the pixel must be excited optically, typically with a large blackbody target and shutter. There are many variations on this method, including a slow chopper [8], [9], an aperture mounted on a linear motion stage (LMS) [10], [11], and a "fast" shutter (transition $< 25\%$ of τ) [9], [12]–[14]. In all cases, the pixels are excited by the blackbody when the aperture is open and cool when it is closed. These methods all fundamentally suffer from the same problem: they all are attempting to gather enough data to fit a curve to (1), but the frame rate of the camera is too slow to populate more than a point or few along the characteristic exponential curve.

The current industry-standard techniques, the slow chopper (2–3 Hz), and LMS (300–500 mm/s) get around this limitation by taking advantage of the spatial profile of the response to a moving aperture. As the aperture moves, the counts increase rapidly where it is open and then fall off in a characteristic blur tail where it is closed. Thus, in a single image frame, one observes that it can fit to the exponential ramp or decay. This occurs over hundreds of pixels in a row, dependent on the details of the FPA, lens, and exact geometry of the measurement. One analysis of the expected error for a slow

Manuscript received January 6, 2020; revised February 5, 2020; accepted March 20, 2020. Date of publication April 2, 2020; date of current version September 15, 2020. The Associate Editor coordinating the review process was Christoph Baer. *(Corresponding author: Dennis L. Waldron.)*

The authors are with the U.S. Army CCDC C5ISR Night Vision and Electronic Sensors Directorate, Fort Belvoir, VA 22060-5806 USA (e-mail: dennis.l.waldron2.civ@mail.mil).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIM.2020.2985138

chopper method predicted ±3.3%, assuming that each error source is uncorrelated [9].

On the other hand, the fast shutter approach must take many frames of imagery. A shutter is thrown with a different delay Δt before the collection of each frame. The bolometer then has a different amount of time to heat or cool before each frame is collected. By carefully tracking Δt for each frame, a single curve can be reconstructed from many frames of data. A related method using a fast chopper (up to 500 Hz) and a laser instead of a blackbody eschews the timing constraints and instead examines response at various chopper frequencies [15]. In this approach, chopper frequency and bolometer response are inversely proportional; the strength of this relationship—how quickly the response falls with frequency—is related to τ.

While methods that require direct electrical biasing of the bolometer are capable of providing per-pixel τ measurements, it is generally impossible to perform these measurements outside of the facility in which the FPAs are fabricated. In addition, these measurements tend not to be reflective of their values in use in a system and under bias and suffer from relying on approximations of the material TCR.

Current system-level approaches also have drawbacks. Any system with a moving aperture will suffer deleterious effects from optics like distortion, defocus and other aberrations, and nonuniform relative illumination. The FPA and readout integrated circuit (ROIC) also introduce many sources of error, such as the effects of responsivity nonuniformity and spatial and temporal noises. As pixels get smaller, diffraction near the aperture edges may also become problematic.

All known current system-level measurement apparatuses report τ for the entire array, typically based on some subsets of the pixels that are actually measured. This is a fundamental limitation for slow aperture systems that require a spatial fit. However, it is only a practical limit of how fast-shutter systems have been constructed to date. This article presents a novel evolution [16] of the fast-shutter approach, which we here call the per-pixel folded-time (PPFT) method, to provide extremely accurate and precise τ results for each pixel. Since NEΔT is already reported per-pixel, this will allow a per-pixel FOM map to be generated for the entire FPA.

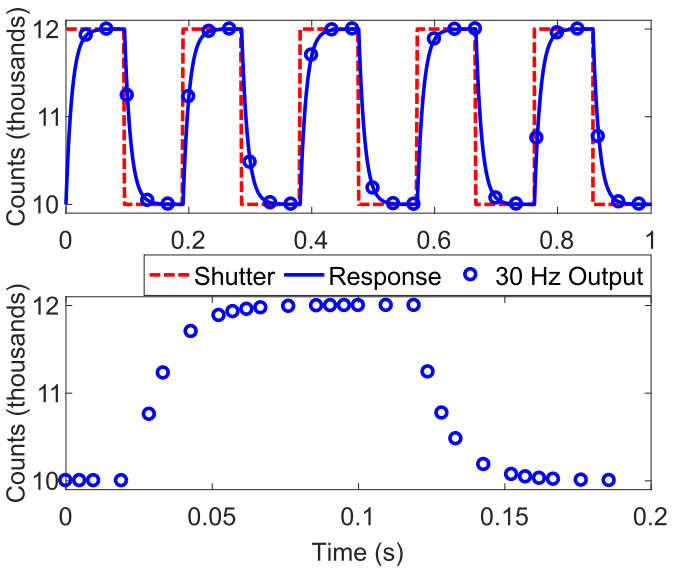

Fig. 1. Top: example simulated data collection with $f_{\text{fr}} = 30$ Hz and 5.25 Hz shutter. Bottom: collected data can be folded back into one period of the forcing function making a curve suitable for extracting τ. Conceptually similar to plots by Waddle *et al.* [13].

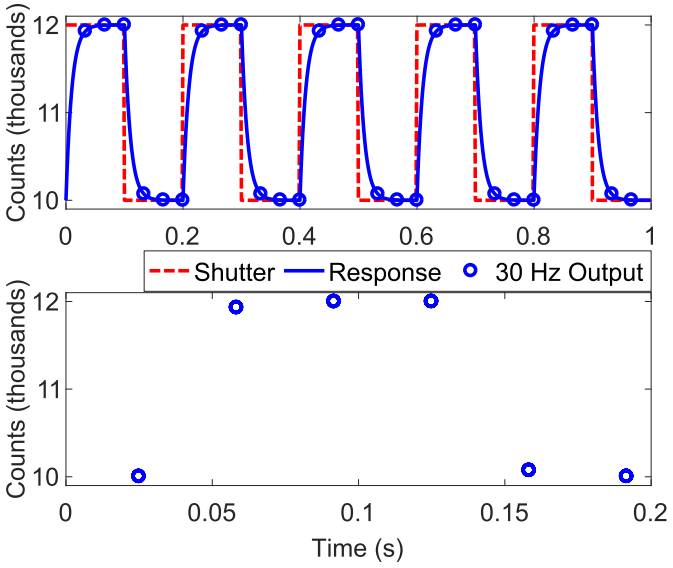

Fig. 2. Top: example simulated data collection with $f_{\text{fr}} = 30$ Hz and 5 Hz shutter. Bottom: forcing function is synchronized to f_{fr}, so a poorly reconstructed curve results when the data are folded back into the period of the forcing function. Conceptually similar to plots by Waddle *et al.* [13].

II. Experimental

A. Concept

The PPFT method is based on the fast shutter approach, whereby τ can be extracted from camera response data collected over many periods of a stimulus that we call the forcing function. Collected data are "folded" back into one period, as shown in Fig. 1, allowing data collected at the camera's frame rate to populate a curve. Either a shutter must be thrown with a set of specific time differences in relation to the camera frame rate f_{fr} [12] or a shutter must be thrown periodically with a period that is not phase-synchronized with f_{fr} [13]. The consequences of synchronization can be seen in Fig. 2; data are captured at the same time relative to the camera frame time and overlaps when folded back, not allowing for enough unique points to fit a curve. If either the camera data or shutter is not carefully timestamped, the shutter period must be long enough to precisely resolve it in the camera response data.

Our system, which is laid out fully in Section II-B, improves upon existing reported systems in key ways, enabling per-pixel results to be calculated for nearly arbitrarily fast time constants. The source type is one key enabler. A laser with a fast acousto-optic modulator (AOM) is used in place of the typical blackbody and mechanical shutter, allowing for fast transition times and a high degree of tuning of various source parameters independently of each other. A mechanical shutter precludes the testing of low-τ cameras because of the large

errors introduced to the response curve from the opening time (at least 1 ms [17]) and possibly diffractive effects near the edge of the shutter blades. Depending on the configuration of the shutter blades, different pixels in the array may have a different exposure time and duty cycle. For example, a pixel in the center of an iris-type shutter will open first and close last, so the observed duty cycle will be longer than for other pixels. Unlike mechanical shutters, a laser beam can additionally be made arbitrarily large, enabling the testing of full large FPAs.

The second key enabler is much more precise timing for all components. Since data over many periods must be folded back into one period, jitter and drift will translate to noisier data and less accurate τ extraction. This can cause both systemic errors in mean estimation and higher random errors characterized by the greater standard deviation of measurements. By timestamping the camera data and precisely controlling the source period, the need to reconstruct the shape of the forcing function with the camera response data is eliminated, effectively reducing the number of fit parameters. Essentially, all of the data can be used in the fit instead of throwing away the highest and lowest 5%–10% as is typical; supplying the fit algorithm with a more complete curve allows more accurate and robust results.

A per-pixel approach additionally has many inherent benefits over approaches that depend in some way on averaging across many pixels. An LMS or other slow aperture technique requires a clear view of the aperture. Distortion, diffraction, defocus, relative illumination, and the modulation transfer function of the lens across the entire field of interest will act as strong sources of error for spatially averaged techniques. A per-pixel approach is much less affected by the lens, basically only to the extent that less flux will create a modestly lower signal to noise near the edge of the FPA. In fact, no lens is required to be used. Only the temporal component of NEΔT causes error in the per-pixel approach; the spatial component, including responsivity differences between pixels, can be ignored.

B. System Implementation

The PPFT τ measurement system that we implemented can be broken into "source" and "data acquisition" (DAQ) paths. The source path, as shown in Fig. 3, provides highly controllable illumination to excite the FPA. An Access Laser L5-S-WCCL CO_2 laser provides up to 5 W of peak continuous wave power with $\pm 1\%$ stability (with closed-loop beam tracker enabled), 2.4 mm beam waist, 5.5 mrad full-angle divergence, and a quality factor of $M < 1.1$ [18]. This is coupled with an AOM with 117 ns/mm optical rise time to provide fast switching and low transition times [19], which is controlled by an IntraAction Corp. GE-4030 radio frequency (RF) source and a Keysight 33611A waveform generator. Additional control of intensity is provided by a Lasnix free-standing wire-grid -10 dB neutral density (ND) filter if needed. A thermopile power meter (Thorlabs S401C head, PM400 console) can be flipped into the beam path to monitor optical power.

The rest of the source path is designed to provide a broad uniform-intensity beam. Here, a series of refractive collimating beam expanders are used although this wastes much of the source beam power. We anticipate further refining this portion of the path to waste less power and realize increased illumination spatial uniformity.

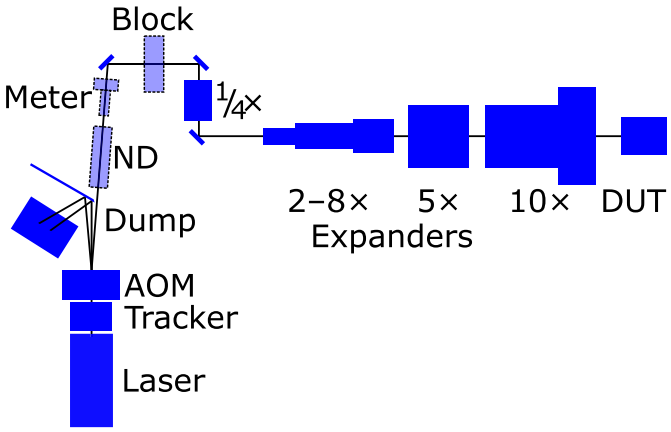

Fig. 3. Source path diagram: components and spacings are mostly to scale. Starting from the lower left, the beam originates in the laser and is split into three beams by the AOM (orders -1, 0, and 1). The dump blocks the two unwanted beams, passing the order 1 beam. Mirrors guide the beam through a set of four expanders (1/4×, 2–8×, 5×, and 10×); the beam passes cleanly through the clear aperture of each expander except going from 5× to 10×. The expanded beam illuminates the camera, i.e., the device under test (DUT). The lighter shaded components with a dotted outline can be flipped in and out of the beam path as needed. They include a -10 dB ND filter, a power meter head, and a beam block. The tracker provides closed-loop feedback to stabilize the laser.

The DAQ path, as shown in Fig. 4, is primarily devoted to collecting precision-timestamped camera response data but also ensures the waveform generator—and, thus, the AOM—is well disciplined and does not drift in time. Raw data from the camera are transmitted via Camera Link base/medium/full to splitter then breakout boxes (Vivid Engineering CLV-402 and CLB-501B). The breakout box provides physical access to the "frame valid" (FVAL) signal, and the splitter prevents any possible signal degradation via the loading of the FVAL output. A time tagger (Swabian Instruments Time Tagger 20) is used to record the time of the leading edge of FVAL with 1 ps resolution and 34 ps rms jitter [20]. This timestamp is used as the frame time when folding data. While the time tagger excels at precise timing over short periods, it is not designed to record data over long periods with low drift without an external stable clock [21]. A clock signal with a 6 ns period conditioned by a 10 MHz rubidium frequency standard (Stanford Research Systems CG635 and FS725/1C) is provided to the time tagger to reduce drift to 0.005 ppm over 20 years [22].

Data frames are recorded on a digital video recorder (DVR) (IO Industries CORE2CL); frames are timestamped by the DVR, but the maximum precision (± 4 μs [23]) may degrade the τ extraction. An alternative to using the splitter and breakout boxes would be to source FVAL from the DVR and feed this to the time tagger but at the penalty of an estimated 200 ns jitter [24]. A "camera busy" signal sourced from the DVR gates the collection of frame timestamp information by the time tagger; timestamps are only generated for recorded frames. The DVR is manually triggered to collect for a

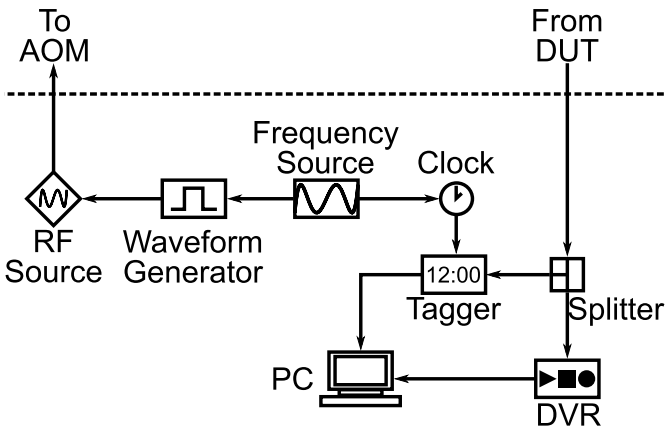

Fig. 4. DAQ path diagram; above the dotted line are connections to the source path (see Fig. 3). Data from the camera (DUT) are split out to the DVR and tagger; FVAL is made physically available to the tagger with a breakout box (not shown). The DVR records video and exports it to a PC for processing. The tagger generates timestamps based on the FVAL signal and exports them to the PC. A clock generator provides stable timing information to the tagger; it is disciplined by an external 10 MHz time base provided by a frequency source. The frequency source also disciplines the waveform generator that controls the AOM in the beam path via an RF source.

predetermined amount of time. All processing takes place on a PC after collection. A slightly modified setup could support other camera data protocols, such as CoaXPress and GigE.

C. Sources of Noise and Error

Timing errors are a primary source of error and were carefully reduced or eliminated to ensure that the system error is dominated by the camera under test. When looking at a plot of bolometer response versus time, timing error will move individual data points left or right, reducing fit accuracy. Note that constant latency through the signal chain does not matter, only jitter and drift.

Estimated jitter for each relevant component is shown in Table I. Jitter for the camera has been broken into clock and engine jitter. Camera clock jitter is a variation of when the camera actually outputs a frame. This is unambiguously clocked by the time tagger and, thus, does not contribute to τ error except in extreme cases. Camera engine jitter is the jitter between when the camera collects data and when it outputs the data; this is estimated from a combination of a number of blocks in the camera video pipeline, which are assumed to be entirely correlated for a worst case analysis. The total jitter on the DAQ path is 337 ps (including the splitter, breakout, tagger, and clock), assuming jitters are uncorrelated (and, thus, is computed via a root-sum-squared method). However, the camera engine jitter must be included, so the total jitter is dominated by and approximately equal to the camera engine jitter, about 80 ns. There is also negligible jitter on the AOM, about 1.4 ps (including the clock and waveform generator).

Drift is also a concern when collecting over long periods. Standard crystal oscillators can accumulate error over time, usually dependent on the temperature and the material's quality. In this setup, the drift has been virtually eliminated as a concern by using a frequency standard. There will still exist drift on the output frame time based on the camera's internal oscillator, estimated to be about 20 ppm based on standard oscillator performance, but this is unambiguously clocked by the tagger and so is not expected to contribute to τ extraction error except in extreme cases.

TABLE I
TIMING JITTER OF VARIOUS RELEVANT COMPONENTS

Parameter	Jitter	Basis of Estimation
camera clock	5 ns	estimated from standard oscillator
camera engine	80 ns	correspondence [25]
breakout box	150 ps	from de-serializer chip [26, 27]
splitter	300 ps	estimated based on breakout box
time tagger	34 ps	data sheet [20]
clock generator	1 ps	data sheet [28]
waveform generator	1 ps	data sheet [29]

Finally, transition time of the source (e.g., ON→OFF and OFF→ON) must be considered. The bolometer response rate of change will be different during the transition period, so longer transition times will introduce error into the τ fit. For transition times that are not negligible, one could attempt to derive and fit to an equation that includes this effect, but the increased number of variables and complexity would also likely degrade accuracy, especially if the transition is not very well characterized. The AOM in this setup has a transition time of 117 ns/mm of beam diameter [19]. Assuming a 2.4 mm $1/e^2$ beam waist diameter [18], $\leq$4 ns rise time from the waveform generator [29], and a rise time of 30 ns for the RF driver [30], a transition time of $<$500 ns was estimated.

Data points in a bolometer response versus time plot can also be moved up or down by errors in the response amplitude. The primary source of this error is characterized by the NEΔT of the pixel. Luckily, only the temporal component of NEΔT contributes to the error with the PPFT method. This is a characteristic of the camera under test and, thus, cannot be eliminated. However, it can be compensated for to an extent by averaging the results of many fits and gathering data longer or perhaps by certain filters in postprocessing. Laser instability can also add error, especially over longer periods of data collection. Here, optical power instability is limited to $\pm 1\%$ with the use of a CO_2 closed-loop line tracker.

Other low-frequency sources of error are also possible, such as $(1/f)$ noise or changes in pixel responsivity from changing thermal conditions. However, these can be largely mitigated by limiting the collection time window for each fit and so are ignored here.

III. MATHEMATICAL DERIVATIONS

As the foundation of our analysis and simulations, we derived the equations describing both the forcing functions and bolometer response solutions. While this analysis derives and implements analytical and semianalytical solutions, a valid alternative would be a fully numerical approach based on the

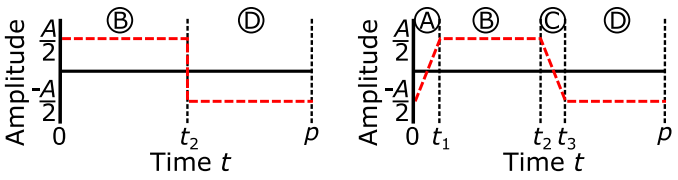

Fig. 5. Square (left) and ramped (right) forcing functions with amplitude A and no dc component. The forcing function and system response can be viewed piecemeal in time with regions Ⓐ–Ⓓ with boundaries set by t_0 to t_4, where $t_0 = 0$ and $t_4 = p$. t_2 represents the center transition and sets the duty cycle of the waveform, and t_1 and t_3 represent the amount of time it takes to transition low→high and high→low, respectively. $0 < t_1 < t_2 < t_3 < p$.

fast Fourier transforms or similar and useful for more arbitrary digital signals.

To solve for the sensor response to the square and ramped forcing functions shown in Fig. 5, we considered boundary conditions. Each segment of the forcing function is assumed to result in a piecewise continuous solution taking the form of a differential equation

$$y(t) = y_{\mathrm{h}}(t) + y_{\mathrm{p}}(t) \tag{2}$$

where y_{h} and y_{p} are the homogeneous and particular solutions, respectively. Critically, the solution must be continuous on either side of a change in forcing function region and, since the forcing function is periodic, must also be equal at the start and end of the period.

y_{p} can be found in general using the Laplace transform for these cases by considering

$$Y(s) = H(s)F(s). \tag{3}$$

$H(s)$ is the bolometer transfer function in the Laplace domain, a single-pole filter

$$H(s) = \frac{1}{1 + s\tau} \tag{4}$$

and $F(S)$ is the Laplace transform of the forcing function.

A. Forcing Functions

Three forcing functions were considered: square, "ramped," and Dirac delta (δ). A "ramped" wave is a square wave that transitions (low→high and high→low) linearly in nonzero time. Both are shown in Fig. 5. All transition times are independently variable for both square and ramped forcing functions so long as $t_0 < t_1 < t_2 < t_3 < t_4$, where $t_0 = 0$ and $t_4 = p$.

These two forcing functions provide the flexibility to model the nonidealities realized when implementing a real system. For example, a shutter or AOM takes some nonzero time to transition, perhaps with different opening and closing times. The times bound the regions Ⓐ–Ⓓ shown in Fig. 5. In the form of $mt + b$, the representative piecemeal equations are

$$\text{Region A} \quad m = \frac{A}{t_1}, \quad b = \frac{-A}{2} \tag{5}$$

$$\text{Region B} \quad m = 0, \quad b = \frac{A}{2} \tag{6}$$

$$\text{Region C} \quad m = \frac{-A}{t_3 - t_2}, \quad b = \frac{At_2}{t_3 - t_2} + \frac{A}{2} \tag{7}$$

TABLE II
ASSUMED VALUES FOR PLOTS IN THIS SECTION (SEE FIGS. 6–8)

Parameter		Value	Unit
amplitude	A	1	arb
period	p	50	ms
time constant	τ	10	ms
duty cycle	d	0.5	frac
rise/fall transition time	t_{tr}	5	ms
time	t_0	0	ms
	t_1	t_{tr}	ms
	t_2	$p \cdot d$	ms
	t_3	$t_2 + t_{\mathrm{tr}}$	ms
	t_4	p	ms

$$\text{Region D} \quad m = 0, \quad b = \frac{-A}{2} \tag{8}$$

where m is the slope and b is the offset of the line.

Assuming that the forcing function in a given region is a line, the forcing function can be expressed in the Laplace domain as

$$f(t) = mt + b \Leftrightarrow F(s) = \frac{m}{s^2} + \frac{b}{s}. \tag{9}$$

An impulse forcing function $\delta(t)$ will also be considered. The Laplace transform of a periodic δ function is given as [31]

$$\begin{aligned} \mathscr{L}\{\delta(t)\} &= \frac{\int_0^p \delta(t) e^{-st}\,\mathrm{d}t}{1 - e^{-sp}} \\ &= \frac{1}{1 - e^{-sp}}. \end{aligned} \tag{10}$$

B. Bolometer Response Solutions

We solve for $y_{\mathrm{p}}(t)$ in a given region using (2) and (3)

$$\begin{aligned} Y(s) &= \frac{1}{1 + s\tau}\left(\frac{m}{s^2} + \frac{b}{s}\right) \\ &= \frac{m}{s^2} + (b - m\tau)\left(\frac{1}{s} + \frac{1}{s - \frac{1}{\tau}}\right) \\ &\Leftrightarrow mt + (b - m\tau)\left(1 + e^{\frac{t}{\tau}}\right) \\ y_{\mathrm{p}}(t) &= m(t - \tau) + b, \quad \text{ignoring transient terms.} \end{aligned} \tag{11}$$

Note that there is no particular solution for a δ forcing function: $y_{\mathrm{p}} = 0$.

Plots of the bolometer response for each of the three forcing functions will be provided. The assumed values for these plots are given in Table II.

All solutions in this section are valid only for one period; replace t with $t - p\lfloor t/p \rfloor$ for a solution which applies over multiple periods, where $\lfloor x \rfloor$ denotes the floor of x.

1) Square Wave: The first case considered is a system with a single τ forced by a square wave, as shown in Fig. 5. There are two regions, Ⓑ and Ⓓ, so a piecemeal solution of two equations is necessary

$$y_{\mathrm{B}}(t) = c_1 e^{-t/\tau} + \frac{A}{2} \tag{12}$$

$$y_{\mathrm{D}}(t) = c_2 e^{-t/\tau} - \frac{A}{2} \tag{13}$$

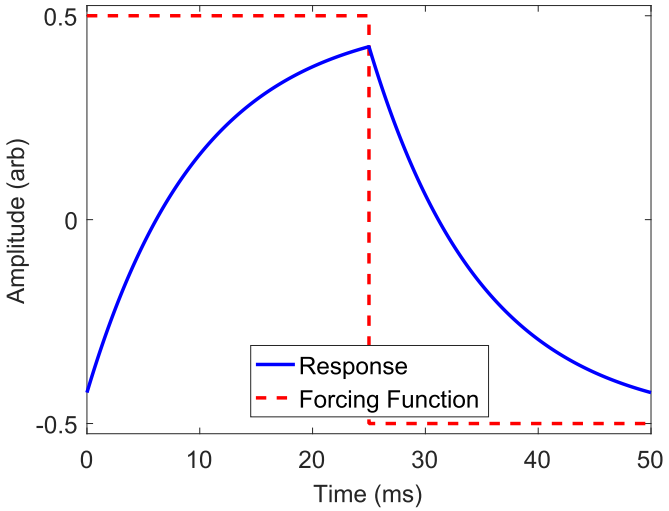

Fig. 6. One period of the solution for a square wave forcing function [plot of (18)].

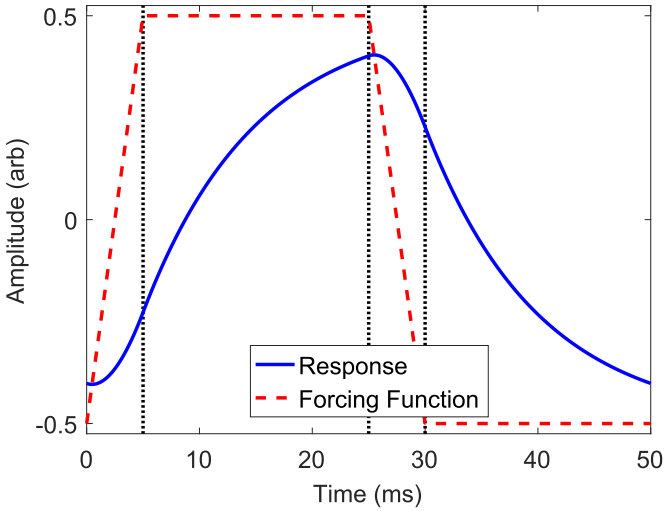

Fig. 7. One period of the solution for a ramped wave forcing function. The "rounding" of the response will negatively impact τ extraction accuracy.

where $\pm A/2$ is the particular solution found by combining (11) with (6) or (8), respectively. There are now two unknowns, c_1 and c_2, which can be solved with a system of two equations derived from the boundary conditions. In this case, the value at the beginning and end of the period must be equal ($y_B(0) = y_D(p)$), and the value at the boundary of regions Ⓑ and Ⓓ must be equal ($y_B(t_2^-) = y_D(t_2^+)$). Incorporating these conditions into (12) and (13) yields the system

$$c_1 e^{0/\tau} + \frac{A}{2} = c_2 e^{-p/\tau} - \frac{A}{2} \tag{14}$$

$$c_1 e^{-t_2/\tau} + \frac{A}{2} = c_2 e^{-t_2/\tau} - \frac{A}{2}. \tag{15}$$

Solving the system for c_1 and c_2 gives

$$c_1 = \frac{A(e^{t_2/\tau} - e^{p/\tau})}{e^{p/\tau} - 1} \tag{16}$$

$$c_2 = \frac{Ae^{p/\tau}(e^{t_2/\tau} - 1)}{e^{p/\tau} - 1}. \tag{17}$$

Finally, (12), (13), (16), and (17) can be combined to yield a complete piecewise solution

$$y(t) = \begin{cases} \dfrac{Ae^{-t/\tau}(e^{t_2/\tau} - e^{p/\tau})}{e^{p/\tau} - 1} + \dfrac{A}{2}, & 0 \le t \le t_2 \\ \dfrac{Ae^{\frac{p-t}{\tau}}(e^{t_2/\tau} - 1)}{e^{p/\tau} - 1} - \dfrac{A}{2}, & t_2 \le t \le p. \end{cases} \tag{18}$$

One period of the solution is plotted in Fig. 6. The response does not have time to converge fully to the half amplitude of the forcing function before the pulse inverts, achieving a maximum value of 0.424.

Any dc offset of the forcing can be added directly to this solution since there is no gain from the transfer function.

2) Ramped Wave: Using a process similar to that for the square wave, a solution can be found for a system being forced by a ramped wave, as shown in Fig. 5. This time, there are four regions, Ⓐ–Ⓓ, and, thus, four unknowns, c_1–c_4

$$y_A(t) = c_1 e^{-t/\tau} + \frac{A}{t_1}(t - \tau) - \frac{A}{2} \tag{19}$$

$$y_B(t) = c_2 e^{-t/\tau} + \frac{A}{2} \tag{20}$$

$$y_C(t) = c_3 e^{-t/\tau} + \frac{A(t - t_2 - \tau)}{t_3 - t_2} + \frac{A}{2} \tag{21}$$

$$y_D(t) = c_4 e^{-t/\tau} - \frac{A}{2}. \tag{22}$$

Again using the boundary conditions for values at the interface of each region and setting the start and end of the period equal, the unknowns can be found

$$c_0 = \frac{A\tau}{t_1(t_3 - t_2)(e^{p/\tau} - 1)} \tag{23}$$

$$c_1 = c_0\left[t_1\left(e^{\frac{t_3}{\tau}} - e^{\frac{t_2}{\tau}}\right) + (t_3 - t_2)\left(e^{\frac{p}{\tau}} - e^{\frac{t_1}{\tau}}\right)\right] \tag{24}$$

$$c_2 = c_0\left[t_1\left(e^{\frac{t_3}{\tau}} - e^{\frac{t_2}{\tau}}\right) + (t_3 - t_2)\left(e^{\frac{p}{\tau}} - e^{\frac{p+t_1}{\tau}}\right)\right] \tag{25}$$

$$c_3 = c_0\left[t_1\left(e^{\frac{t_3}{\tau}} - e^{\frac{p+t_2}{\tau}}\right) + (t_3 - t_2)\left(e^{\frac{p}{\tau}} - e^{\frac{p+t_1}{\tau}}\right)\right] \tag{26}$$

$$c_4 = c_0\left[t_1\left(e^{\frac{p+t_3}{\tau}} - e^{\frac{p+t_2}{\tau}}\right) + (t_3 - t_2)\left(e^{\frac{p}{\tau}} - e^{\frac{p+t_1}{\tau}}\right)\right] \tag{27}$$

where c_0 is a common factor between each of the other c_n.

Although omitted for brevity, it is now possible to combine (24)–(27) into (19)–(22) to form a complete piecewise solution as in (18). One period is shown in Fig. 7. The maximum value this time is slightly less than the square case, 0.403 compared with 0.424, and shifts slightly beyond the time the forcing function begins to invert.

3) δ Function: Solutions are found for a single pulse applied to a system at rest and for a pulse train with period p. For a single pulse when the system is at rest, one need only solve $Y(s) = H(s)F(s)$, where $F(s) = 1$

$$\frac{1}{1 + s\tau}(1) \Leftrightarrow \frac{1}{\tau}e^{-t/\tau}. \tag{28}$$

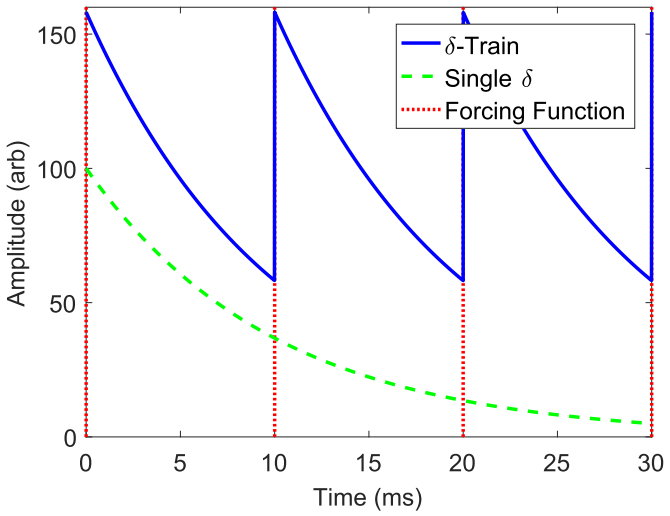

Fig. 8. Single δ pulse and three periods ($p = 10$ ms) of a train of δ pulses.

When considering a periodic train of δ pulses, $Y(s) = H(s)F(s)$ must again be solved, but, now, $F(s)$ is given by (10)

$$Y(s) = \frac{1}{1+s\tau}\frac{1}{1-e^{-sp}} = \frac{1}{\tau}\frac{1}{s+\frac{1}{\tau}}(1+e^{-sp}+e^{-2sp}+\cdots+e^{-nsp}) \quad (29)$$

where the forcing function is represented by a power series. Consider that each term in this power series will shift the response function in time because $F(s)e^{-st_0} \Leftrightarrow f(t-t_0)u(t-t_0)$ [32]. Thus, the Laplace transform of (29) can be found

$$y(t) = \frac{1}{\tau}e^{-t/\tau}(1+e^{p/\tau}+e^{2p/\tau}+\ldots+e^{np/\tau}) \quad (30)$$

$$= \frac{1}{\tau}e^{-t/\tau}\left[\frac{1-e^{p(n+1)/\tau}}{1-e^{p/\tau}}\right] \quad (31)$$

$$= \frac{\exp\left[\frac{p+np-t}{\tau}\right]}{\tau(e^{p/\tau}-1)} - \frac{e^{-t/\tau}}{\tau(e^{-t/\tau}-1)} \quad (32)$$

$$= \frac{\exp\left[\frac{p+np-t}{\tau}\right]}{\tau(e^{p/\tau}-1)} \quad \text{ignoring transients} \quad (33)$$

where the series has been condensed using the identity

$$\sum_{k=0}^{n}\frac{1-r^{n+1}}{1-r}, \quad r = e^{p/\tau}.$$

For computational purposes, n is simply the integer number of periods from t_0

$$n = \left\lfloor \frac{t}{p} \right\rfloor. \quad (34)$$

The response of a single δ pulse and a train of δ pulses is shown in Fig. 8. Note that the period of the pulse train is 10 ms and so the system cannot fully relax between pulses.

4) "Noisy" Solutions: The solutions shown in Sections III-B1 and III-B2 are limited in that they describe the steady-state behavior of the system, unperturbed by changes in conditions or noise. While some sources of noise, such as NEΔT, can simply be added after the calculation of the detector solution, others will require new values of c_n to be calculated each period; equations for $y(t)$ [see (12), (13), and (19–22)] are still valid.

When it is necessary to solve for a new c_n, a similar procedure is followed as before, but with two differences. The times t_0–t_4 are now not calculated just based on the period, duty cycle, and rise time, but also considering various timing jitter and drift. More importantly, the boundary conditions change. We still assume that the solution must be continuous between regions of the forcing function, but it is no longer periodic because the noise on timing forces new values of t_n each period. Thus, we instead set the start of the current period equal to the end of the previous period y_{prev} and recompute all values of c_n every period. For the first period computed, the noiseless equations are used to seed the noisy solution with the "previous" value.

For the square case

$$c_1 = \left(y_{\text{prev}} - \frac{A}{2}\right)e^{t_0/\tau} \quad (35)$$

$$c_2 = c_1 + Ae^{t_2/\tau}. \quad (36)$$

For the ramped case

$$c_1 = \left(\frac{A(\tau - t_0)}{t_1} + y_{\text{prev}} + \frac{A}{2}\right)e^{t_0/\tau} \quad (37)$$

$$c_2 = c_1 - \frac{A\tau}{t_1}e^{t_1/\tau} \quad (38)$$

$$c_3 = c_2 + \frac{A\tau}{t_2 - t_3}e^{t_2/\tau} \quad (39)$$

$$c_4 = c_3 - \frac{A\tau}{t_2 - t_3}e^{t_3/\tau}. \quad (40)$$

C. Fit Methods

In this work, we evaluated three methods to fit data and extract τ from the PPFT data. The first method is to perform a nonlinear least squares (NLLS) fit to (1). We used the ezfit function in the MATLAB EzyFit toolbox [33], but any equivalent method would suffice. Goodness of fit for this method depends strongly on good estimates of initial conditions. Initial values for a, b, and τ are chosen based on an integral noniterative nonlinear regression of the exponential function [34] shown in (1) assuming that $c = 0$

$$S_k = \begin{cases} 0, & k = 1 \\ S_{k-1} + \frac{1}{2}(y_k + y_{k-1})(t_k - t_{k-1}), & 2 \le k \le n \end{cases} \quad (41)$$

$$\begin{bmatrix} \sim \\ \frac{-1}{\tau} \end{bmatrix} = \begin{bmatrix} \sum_{k=1}^{n}(t_k - t_1)^2 & \sum_{k=1}^{n}(t_k - t_1)S_k \\ \sum_{k=1}^{n}(t_k - t_1)S_k & \sum_{k=1}^{n}(S_k)^2 \end{bmatrix}^{-1}$$

$$* \begin{bmatrix} \sum_{k=1}^{n} (t_k - t_1)(y_k - y_1) \\ \sum_{k=1}^{n} (y_k - y_1) S_k \end{bmatrix} \tag{42}$$

$$\begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} n & \sum_{k=1}^{n} e^{-t_k/\tau} \\ \sum_{k=1}^{n} e^{-t_k/\tau} & \sum_{k=1}^{n} e^{-2t_k/\tau} \end{bmatrix}^{-1} \begin{bmatrix} \sum_{k=1}^{n} y_k \\ \sum_{k=1}^{n} y_k e^{-t_k/\tau} \end{bmatrix}. \tag{43}$$

The other two methods that we consider are corrected successive integration (CSI) [35], [36] based on trapezoidal integration and a related linear regression of the sum (LRS) [36] based on rectangular integration; both use a solution of linear least squares. The LRS method was shown by Everest and Atkinson [36] to run about an order of magnitude faster than a traditional Levenberg–Marquardt least-squares method while maintaining near-equal accuracy. During their analysis, all least-squares methods seemed to perform best with a record length (i.e., forcing function period) of five to seven times τ. Other methods that could be considered in the future are those based on the Fourier transform [36]–[38] or the Legendre transform [39].

IV. Simulation

We wrote a simulation in MATLAB to both act as proof of concept and also to explore the limits of the accuracy of this system. It first generates data as we would expect to receive from the PPFT measurement hardware and then attempts to extract τ. Depending on the parameters being modeled, it uses the appropriate bolometer response solutions discussed in Section III-B and the fit methods described in Section III-C.

A. Description

The simulation generates a counts value for a single pixel based on the bolometer response equations and an assumed linear responsivity of the pixel in counts per Kelvin (cnt/K) at the time each frame is collected. There are two primary cases, "pristine" and "noisy." In the pristine case, the simulation simply calculates the response for all times $t = 0$ to t_{end} in increments of one over frame rate f_{fr} using the appropriate equation (for example, (18) for a pristine square wave). NEΔT (temporal noise) can be added directly to this result and is assumed to be Gaussian noise with a mean of zero and standard deviation equal to the NEΔT.

However, to model timing jitter or drift or a change in forcing function amplitude (e.g., if the laser source is unstable over time), c_n must be recomputed each period and so the noisy functions must be used. The DAQ jitter is applied to each frame timestamp as a Gaussian random variable with mean zero and standard deviation equal to the total jitter, while AOM jitter is likewise applied to the time of the start of the transition of the forcing function from low$\rightarrow$high (t_0) and independently to high$\rightarrow$low (t_2). Camera clock jitter is applied to the time the response is calculated, i.e., the frame time. Thus, there are two timings that are important within the simulation: the time the response is calculated (frame time) and the timestamp is applied to this response (the time the DAQ thinks the response occurred).

Drift, typically measured in parts per million, is applied to each clock, respectively, as a "time dilation." A linearly spaced vector with the same length as the clock vector is generated from 1 to (1+drift), and the clock vector is multiplied by this dilation vector.

Laser instability is modeled by changing the amplitude of the forcing function each period. A moving average filter is applied to Gaussian noise with mean zero and standard deviation equal to one-third of the given percentage of the amplitude so that values within three standard deviations will be within the given stability percentage. The filter window length was chosen such that the result looked "reasonable," i.e., provided a realistic-looking result when 10 min of values are plotted and viewed. In this scheme, amplitude A is computed once for each period. Thus, if the length of the period is changed, the noise statistics (e.g., power spectral density) will change. To compensate, the window length is scaled linearly with the length of the period. The window chosen was $5000 * 50 \times 10^{-3}/p$, so the window is 5000 samples wide when the period is 50 ms. Finally, the output of the moving average filter is scaled by the square root of the window length to restore the proper standard deviation after being suppressed by the filter. This scheme generates random amplitude with proper standard deviation but with high-frequency changes suppressed.

Data are generated in a loop for each frame for the duration of the desired collection time, and all outputs are then wrapped back to one period. Because this data always start with the rising edge aligned with $t = 0$, the data can optionally be randomly phase-shifted to simulate real data that will have an arbitrary phase based on the exact pixel read time by the ROIC.

With the simulated data now generated, the fit routine starts by removing the minimum value and then removing the phase shift. Central to the phase shift removal algorithm is MATLAB's findchangepts, a method to find abrupt changes in a signal. In this case, it looks for changes in both the mean and slope of the numerical derivative of the response. To assist the function, the data are first smoothed using a Savitzky–Golay smoothing filter (the unsmoothed data are still used to extract τ). This is used to find the indices of the array values at which the rising edge transitions to a falling edge, and vice versa. Fits to find τ are then performed on the selected falling edge data after setting the start of the selected data to $t = 0$.

Some known sources of noise and variability have been ignored in this model. Of particular importance is the fact that a real FPA will be both nonlinear in response and has pixel-to-pixel variability in responsivity that will likely be the ultimate limit to the reduction of measurement standard deviation in practice. Many low-frequency noise sources are ignored, such as $(1/f)$ and drift with temperature fluctuation.

B. Simulation Results

A number of simulation cases were run to explore the best conditions for performing the experiment and perform

TABLE III
BASELINE CASE SIMULATION VARIABLES, SYMBOLS, AND VALUES

Variable		Value	Unit
time constant	τ	10	ms
responsivity		100	cnt/K
room temperature counts		10 000	cnt
room temperature		20	°C
blackbody temperature		80	°C
frame rate	f_{fr}	30	Hz
forcing function period	p	50.9	ms
duty cycle	d	0.35	frac
rise/fall transition time	t_{tr}	500	ns
collection time		30	s
camera clock drift		20	ppm
camera clock jitter		5	ns
camera engine jitter		80	ns
DAQ clock drift		5	ppb
DAQ clock jitter		337	ps
AOM jitter		1.4	ps
laser stability		1	%
temporal noise	NEΔT	50	mK
random phase		1	bool.

a sensitivity analysis. Four cases were run to find ideal experimental conditions, including a sweep through forcing function period, duty cycle, data collection time, and equivalent blackbody temperature. A list of simulation variables and their values reflecting choices made during the four initial sweeps for the baseline case is given in Table III. Variables not optimized by these four sweeps were taken from collected values (see Table I) or reasonable estimates; many of these were later swept for a sensitivity analysis. In each case, 2000 simulations were run for each value of the variable being swept, except for the baseline case, for which 10 000 simulations were run.

The results of the first baseline-setting decision sweep, varying forcing function period 2.036–152.7 ms, can be seen in Fig. 9. The standard deviation drops as desired as the period length increases, leveling off near 50.9 ms; this matches prior results predicting best performance with $p = (5\times\text{–}7\times)\tau$ [36]. NLLS slightly outperforms the other methods, with CSI and LRS being about equal. The mean also stabilizes and converges to the correct value as the period length increases though much more quickly for CSI and NLLS methods. Again, NLLS does best, with CSI close behind and LRS performing noticeably worse. It is expected that an increased period length will give better results because the bolometer is allowed to relax more fully and more of the tail is available for fitting; at extremely long period lengths, we would anticipate results to get worse as the tail is overweighted while fitting. This ranking of goodness, NLLS best and LRS worst, will become a general trend.

Next, the duty cycle "high" fraction was swept 0.01–0.9; results are shown in Fig. 10. There is a minima in standard deviation around 0.3–0.35, and the mean is very near the true value for CSI and NLLS. LRS is again seen to oscillate but is

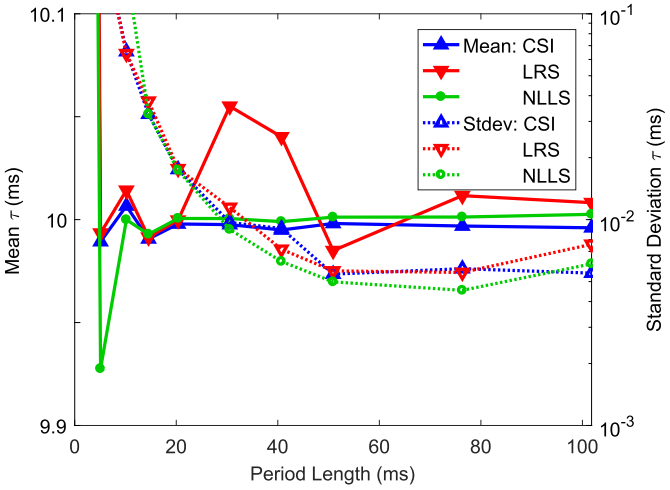

Fig. 9. Mean and standard deviation of predicted τ values versus period length p. Markers are data points; lines are guides to the eye.

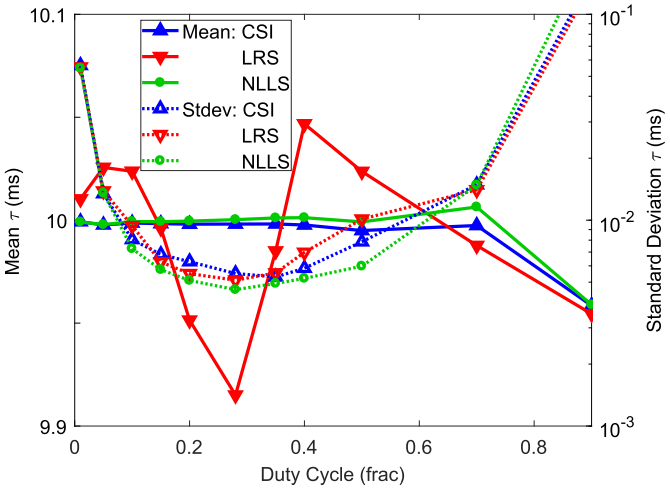

Fig. 10. Mean and standard deviation of predicted τ values versus duty cycle d. Markers are data points; lines are guides to the eye.

near the true value at 0.35, so this value of d was chosen for the baseline case. Performance in terms of both mean and standard deviation suffers as the duty cycle is increased beyond this range because there is very little time in the falling edge curve with which to fit. However, as the duty cycle is reduced, the signal to noise suffers because there is less time for the pixel to heat up; this can be, somewhat, ameliorated by increasing the source intensity, but this may be impractical.

The length of time that the collection is run is another parameter that can easily be varied. Obviously, it is desirable to run the experiment for shorter lengths of time, especially in terms of how long it takes to postprocess the data. A shorter collection would also limit the effects of low-frequency noise, which was not considered in this simulation. As shown in Fig. 11, running the experiment longer allows the predicted mean value to stabilize near the true value for CSI and NLLS fits; 30 s was chosen to minimize time and system resources to postprocess and immunity to low-frequency noise while reducing systemic error. The fact that gains in standard

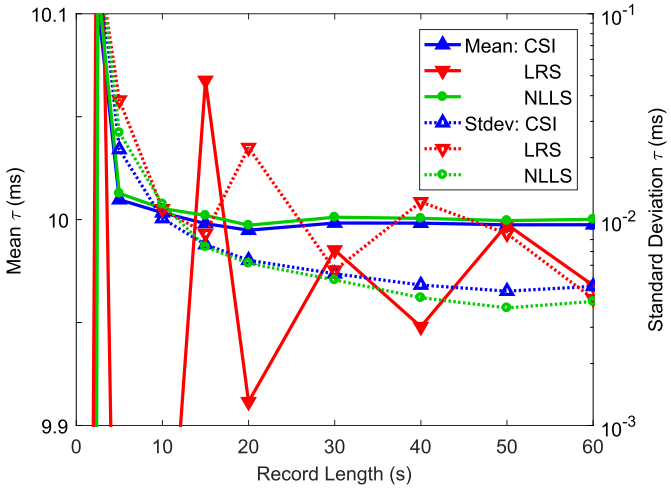

Fig. 11. Mean and standard deviation of predicted τ values versus record length (the amount of time data is recorded). Markers are data points; lines are guides to the eye.

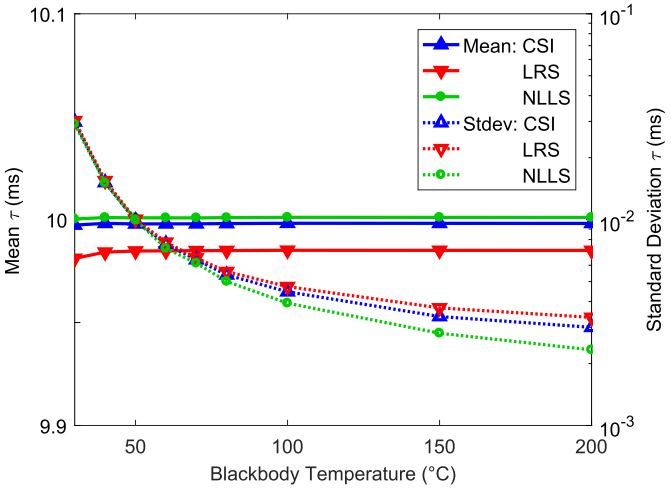

Fig. 12. Mean and standard deviation of predicted τ values versus source equivalent blackbody temperature. Markers are data points; lines are guides to the eye.

deviation begin to level off after 30–40 s implies that one is better off averaging the results of two 30-s experiments rather than running a longer single experiment to reduce random fit error.

Finally, source intensity in terms of equivalent blackbody temperature was swept 30 °C–200 °C (Fig. 12). Again, as expected, standard deviation goes down as temperature goes up due to the increased signal to noise, but the mean is surprisingly robust to even the lowest temperatures; 80 °C was chosen as a realistic temperature while maximizing simulation performance, but a lower temperature may be better when taking experimental data so as to limit nonlinear effects with temperature not considered in this model. We note that at 30 °C, even though the mean was well predicted, there were a significant number of fit failures as a result of the phase-recovery portion of the algorithm failing. In general, fit "failures" are due to failure of the phase recovery algorithm, not because the fit algorithms fail to converge.

With parameters thus "optimized" and set as in Table III, the baseline case was run. Aggregate results are shown in Table IV, and a histogram of the 10 000 runs are shown in Fig. 13 (solid traces). Again, we see that NLLS provides the best fit, followed closely by CSI. However, NLLS has a significant penalty in time per fit compared with CSI and LRS, about three orders of magnitude slower. This makes NLLS impractical for a large array; for a 1920×1200, NLLS will take about 40.6 h to fit every pixel, compared with 3.35 min for CSI and 1.47 min for LRS on a single 3.4 GHz processor core. Given this, we view CSI as the best fit method to use experimentally in terms of goodness of fit and processing time. Fit times for the CSI and LRS cases are unfortunately limited by the phase recovery algorithm that takes 1.21 ms per record or 46.6 min for the aforementioned array.

To assess the potential impact to fit the accuracy of the phase recovery algorithm, the simulation was run with a known (zero) phase offset, and the phase recovery algorithm was bypassed. A histogram of the results can be seen in Fig. 13 (dotted traces). Both mean accuracy and standard deviation improve slightly for CSI and NLLS as expected. However, the goodness of fit for LRS actually degraded, likely coincidentally due to its oscillatory behavior observed in Figs. 9–11. In all cases, the impact is present, but relatively minor.

TABLE IV
BASELINE CASE SIMULATION RESULTS FOR RANDOMIZED PHASE AND KNOWN ZERO PHASE

Rand. phase	CSI	LRS	NLLS	Unit
Mean τ	9.99016	9.98504	10.0011	ms
% error	−0.01834	−0.14960	0.01068	%
St. dev. τ	5.49940	5.68525	5.07802	µs
Run time	0.0873	0.0383	63.4	ms
Known phase	**CSI**	**LRS**	**NLLS**	**Unit**
Mean τ	10.0001	9.99859	10.0000	ms
% error	0.00089	−0.21414	−0.00036	%
St. dev. τ	4.76577	4.68997	4.66222	µs
Run time	0.1014	0.0403	70.5	ms

After establishing the baseline case, a sensitivity analysis was run by varying each of the other parameters in Table III individually. While many of these parameters are either set by the choice of experimental hardware or entirely outside of the operator's control (i.e., a parameter of the camera under test), this was done to establish limits within which an accurate result can be expected when performing the experiment. For brevity, figures are omitted in the discussion of these results.

The camera clock drift was varied from 0 to 1000 ppm from its baseline value of 20 ppm. There is a slight degradation of mean and standard deviation starting at 20 ppm, but all perform well out to 1000 ppm, with CSI and NLLS performing best; NLLS has essentially no change in standard deviation in the simulated range. At 1000 ppm, percent errors of the predicted mean for CSI, LRS, and NLLS are $<-0.1\%$, $<-0.35\%$, and $<-0.02\%$, respectively.

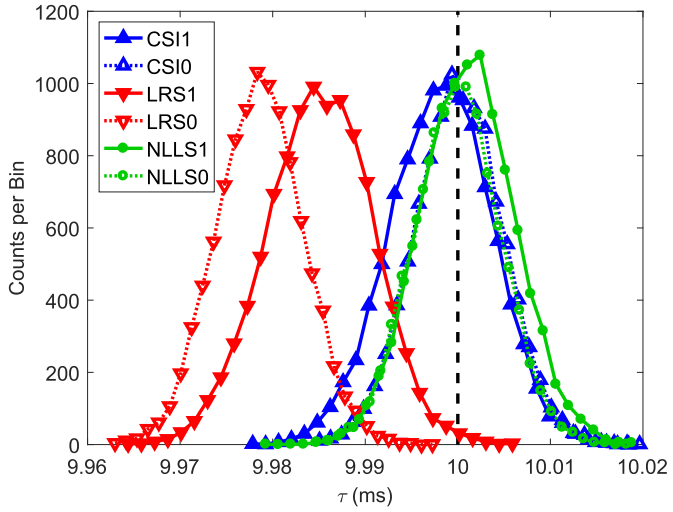

Fig. 13. Histogram of predicted τ values for each fit type for the baseline case using 2000 simulation runs in 30 bins. Fits with "1" notation have their phase randomized and must be run through the phase-detection algorithm; "0" denotes a known zero phase shift. Both systemic error in prediction of τ and standard deviation are reduced for the CSI and NLLS fits for the known-phase case as expected, demonstrating the effect of imperfect phase detection. For the LRS case, we believe that the oscillatory behavior seen in Figs. 9–11 explains why the systemic error is apparently increased, even as the standard deviation is improved.

TABLE V
SIMULATION RESULTS FOR TWO EXTREME VALUES OF τ

$\tau = 100$ µs	CSI	LRS	NLLS	Unit
Mean τ	99.872	100.69	99.976	µs
% error	−0.128	0.689	−0.024	%
St. dev. τ	191.34	65.184	61.118	ns
$\tau = 100$ ms	CSI	LRS	NLLS	Unit
Mean τ	100.07	99.912	99.995	ms
% error	0.074	−0.088	−0.005	%
St. dev. τ	50.304	54.520	48.176	µs

TABLE VI
SIMULATION RESULTS FOR TWO "PERFECT" NO-NOISE FORCING FUNCTION INPUTS, δ-FUNCTION AND SQUARE WAVE

δ-function	CSI	LRS	NLLS	Unit
Mean τ	10.0001	9.99957	10.0000	ms
% error	0.001	−0.004	−0.000	%
Square wave	CSI	LRS	NLLS	Unit
Mean τ	10.0001	9.99106	10.0000	ms
% error	0.001	−0.089	−0.000	%

The camera clock jitter was varied 1 ps–1 ms from its baseline value of 5 ns. All fit methods maintain nearly exactly their baseline values out to >1 μs. CSI and NLLS maintain performance to >10 μs, with the predicted mean unchanged for NLLS within the simulated range.

The DAQ clock jitter was varied from 1 ps to 10 ms from its baseline value of 337 ps. For this simulation, the camera engine jitter was set to 0 because DAQ and engine jitter add directly; a separate simulation sweeping camera engine jitter was omitted. The results are essentially the same as for the camera clock jitter sweep, except that all fit methods begin to perform very poorly or fail at 0.1 ms.

The DAQ clock drift was varied from 0.1 to 1000 ppm from its baseline value of 5 ppb. All perform equally well to their baseline value out to $>$ 0.1 ppm with only very minor degradation to $>$ 1 ppm and tolerable degradation out to about 10 ppm. For a drift of $\geq$ 50 ppm, all fits fail in nearly every trial.

The laser stability was varied from 0% to 50% from its baseline value of 1%. Impressively, the mean is nearly unchanged across the entire simulated range although 1 of the 2000 trials failed to be fit at 50%. However, the standard deviation increases near-linearly with instability with a 2–4.5× overall increase depending on the fit type, with a lower slope below about 5%.

The NEΔT was varied 0 mK–1000 mK from its baseline value of 50 mK. The mean is essentially unchanged to 100 mK, with only minor degradation to 750 mK. However, the standard deviation increases about three orders of magnitude with an increasing rate of change as NEΔT increases, especially above 35 mK. Limited fit failures begin to occur near 750 mK.

The AOM timing jitter was varied from 1 ps to 1 ms from its baseline value of 1.4 ps. The results are basically equivalent to the results of the DAQ clock jitter although, at 1 ms, more than two in three trials fail to find a fit.

The transition time t_{tr} was varied from 1 ps to 10 ms from its baseline value of 500 ns. The mean and standard deviation of each fit method are essentially unchanged to 100 μs. At 1 ms, there is a significant error in the mean ($>$10%) and degradation of standard deviation performance; results at 10 ms do not fail to fit, but the systemic error in prediction of τ is so great as to make the results unusable. Assuming that t_{tr} is sufficiently stable, it is conceivable that a fit technique that took it into account would be successful to larger values of t_{tr}.

To evaluate the usefulness of the technique for varying τ, the model was run at $\tau = 100$ μs, $p = 509$ μs and $\tau = 100$ ms, $p = 509$ ms without any reoptimization of other parameters. The results are shown in Table V and demonstrate the power of such a system to predict a wide range of τ values. It is odd that a histogram of the CSI fit results for the $\tau = 100$ μs case (not shown) shows a clear binomial shape, unlike any other case that was run. This explains the larger than expected standard deviation seen for the CSI fit in this case and merits future investigation. It should be noted that although the sensitivity analysis shows very wide bounds for successfully predicting τ near 10 ms, it is likely that the bounds of timing parameters would be narrowed significantly for τ on the order of 100 μs.

Finally, the fit techniques were evaluated under "perfect" no-noise conditions and known phase with both a δ-function and ideal square wave with $d = 0.35$. The results can be seen in Table VI. All methods predict τ with very high accuracy though, again, we see the trend of NLLS and CSI performing the best. Since there is no noise or other variations between iterations, the standard deviation is zero.

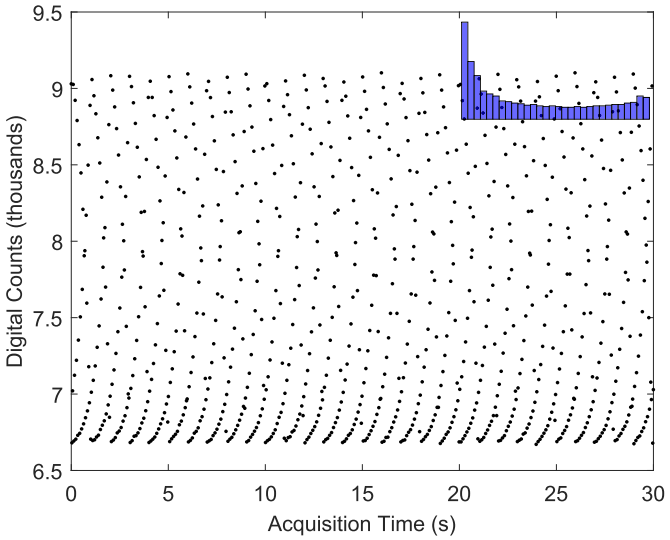

Fig. 14. Raw data for a single pixel through 30 s of DAQ. Upper right inset: histogram of the frequency of digital count levels (30 bins).

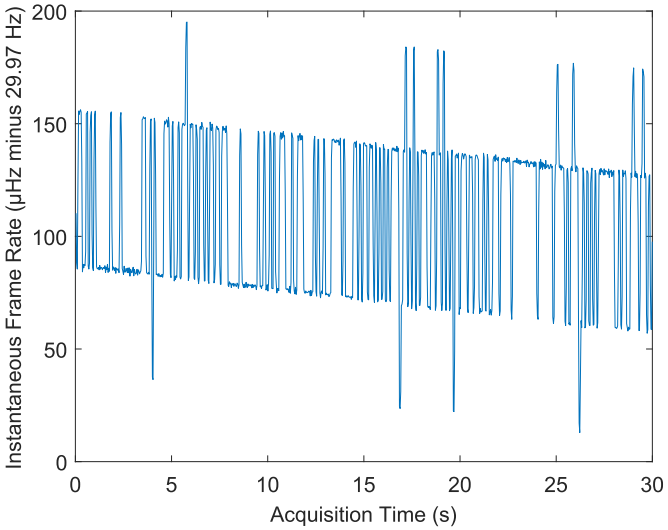

Fig. 15. Instantaneous frame rate generated from the gathered timing data through 30 s of DAQ. Note that the y scale goes from 29.97 to 29.9702 Hz, near the nominal frame rate of 30 Hz.

V. Results

The proof-of-concept experimental results are given using the system implementation discussed in Section II-B. These results are compared with measurements taken on an LMS [10], [11].

A. Experimental Validation

A FLIR model Tau camera was used without lens as a camera under test. It features a 640 × 512 format microbolometer FPA, 17-μm pixel pitch, and an expected $\tau \approx 9$ ms. Data were collected in 14-bit raw mode, i.e., without bad pixel replacement or noise filtering. A shutter-based nonuniformity correction (NUC) was performed immediately prior to data collection. Fig. 14 shows raw data in digital counts as collected over 30 s for a single pixel. A histogram in the inset of Fig. 14 shows the frequency of digital counts. Overall, this shows a very good spread of values, one indicator which implies that we have chosen p well. Lower count levels are observed more frequently because $d < 0.5$.

Fig. 15 shows the instantaneous frame rate generated from the time tags collected by the tagger corresponding to each data point in Fig. 14. It is interesting to observe that there is indeed drift involved in the camera frame clock, apparent here as a low-frequency change in frame rate. Also, the timing jitter appears to be better characterized primarily as so-called "popcorn" or random telegraph noise rather than the Gaussian assumed in the simulation.

Fig. 16 shows the data from Fig. 14 folded back into a length of time equal to the source period p with the help of the gathered timing data. The data very nicely demonstrate the expected characteristic exponential shape. Coincidentally, the phase is nearly aligned as needed to fit the data, with just a small bit of the falling edge data at the end of the sequence.

Fitting proceeds in the same manner as dealing with simulated data but now looping over pixels in the array instead of individual simulation runs. The falling edge data used for the

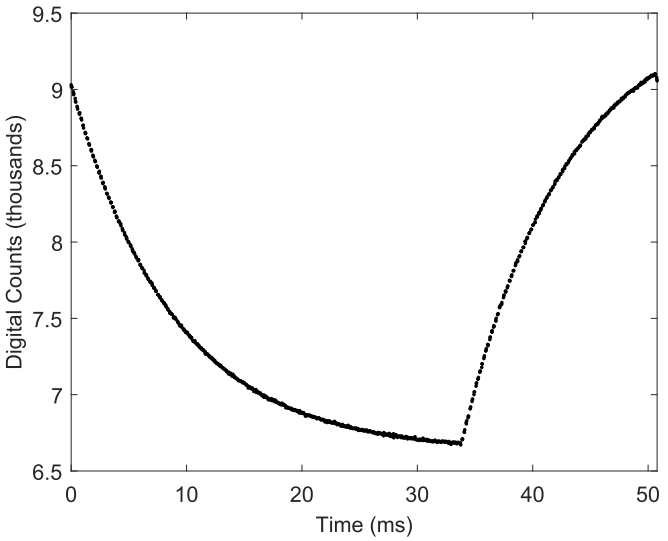

Fig. 16. Raw data from Fig. 14 folded back into a length of time equal to source period p. Coincidentally, the phase is nearly aligned as needed to fit the data, with just a small bit of the falling edge data at the end of the sequence.

fit and the resulting fit curve from each algorithm are shown in Fig. 17. Just as in the simulation, all fits perform well, with CSI and NLLS providing a superior fit. The extracted τ's for this example pixel are 8.9048, 8.8484, and 8.8936 ms for CSI, LRS, and NLLS, respectively.

Repeating this process for each pixel results in the 2-D pixel map shown in Fig. 18. The map shown was generated using the CSI algorithm; maps using the LRS and NLLS maps look very similar, so it has been omitted. Fig. 19 shows the histogram of the frequency of predicted values of τ. The bimodal shape of the histogram is reflected in the pixel map—roughly, the top half of the FPA shows a faster τ than the bottom half. The

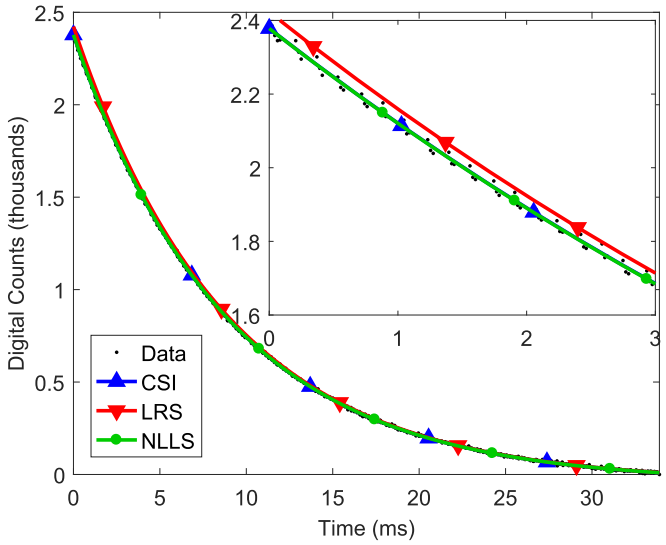

Fig. 17. Selected falling edge data and fits using extracted τ. As expected from modeling, all three fits are very representative of the data, but CSI and NLLS are clearly superior. Upper right inset: a zoomed-in view of the first 3 ms from the 33.9 ms falling edge.

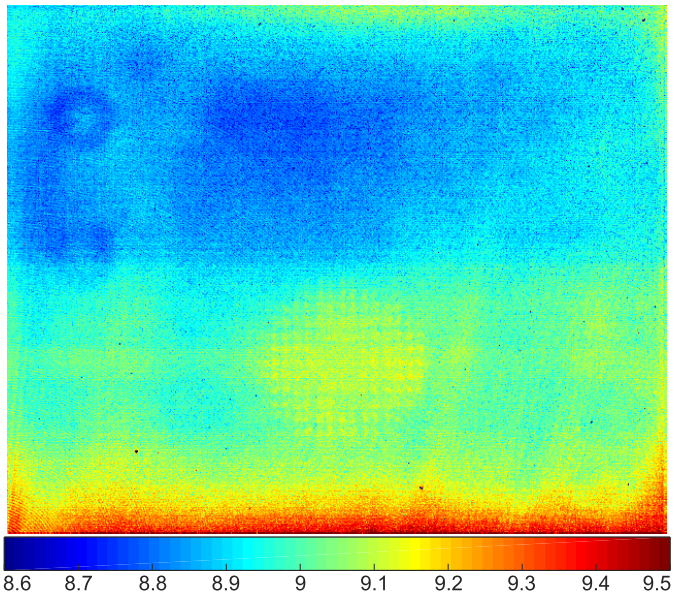

Fig. 18. 2-D per-pixel map of τ values extracted by the CSI algorithm for a FLIR Tau camera. Values are in milliseconds. Note the appearance of a few clusters of apparent dead pixels.

long tail on the trailing edge of the histogram is the band of slow pixels along the bottom of the map (see Fig. 18).

Per-pixel run times of fits were virtually identical to those observed in simulation. In this example, 64 pixels failed to fit because the phase recovery algorithm failed. These can be seen in small clusters in Fig. 18. We take these to be "dead" pixels. (There are many definitions of dead pixels, which are typically considered those which have a responsivity value far from the observed mean; the exact definition is not important here.)

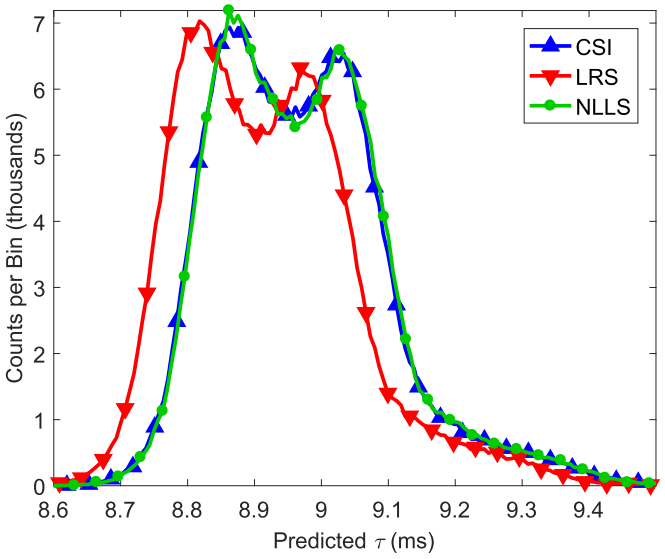

Fig. 19. Histogram of the frequency of extracted τ values from each fit algorithm. Bins are about 6.6 μs wide; markers appear every fifth bin.

TABLE VII
COMPARISON OF PER-PIXEL RESULTS TO LMS RESULTS FOR TWO RECTANGULAR REGIONS OF INTEREST

τ	PPFT	Linear Motion	Δ	Unit
Top	8.79	9.65	0.86	ms
Bottom	9.08	9.91	0.83	ms
Δ	0.29	0.26		ms

B. Comparison to Linear Motion Stage

Our current established practice for measurement of τ is to use an LMS technique using a 6.5 mm slit traveling at 500 mm/s in front of a 50 °C blackbody. In order to compare the methods, the same camera was measured on this LMS. Two areas of the FPA were examined, each with a three row by 150 column region of interest: a "top" rectangular region bounded by points (210, 99) and (359, 101) and a "bottom" region bounded by points (250, 339) and (399, 341), where (0, 0) is the top left corner of Fig. 18. These regions correspond roughly to the area of darker blue in the top left and yellow in the bottom middle. Extracted τ's from ten total frames in each region were averaged for the LMS measurement, and all pixels in each region of interest were averaged for the PPFT measurement. Results are shown in Table VII.

The results from the LMS show a slower time constant in the bottom region, as expected from the PPFT 2-D map. The difference between the top and bottom regions are very similar for both methods, about 0.3 ms. However, the PPFT method results report a τ almost 9% faster than the LMS in both regions. We have observed a similar magnitude of difference between the LMS and rotating chopper methods in the past.

Preliminary simulations of the LMS technique suggest that at least 0.14 ms of the difference is due to bias in the LMS technique. Rounding of the response similar to that observed in Fig. 7 is caused by the sliding of the slit across the pixel (i.e., the source does not switch ON/OFF instantaneously). The

amount of rounding and, therefore, the error in τ extraction are dependent on many factors, such as measurement geometry, lens parameters, and stage velocity. We have also observed that for the LMS technique, the selection of data to fit is important and somewhat arbitrary in terms of how close to the peak response to select and how much of the tail to include. For a given data set, selecting a different subset to fit can easily alter the extracted τ by 0.2 ms or more. Different effective source temperatures will also modify the operating or quiescent point of the bolometers, altering the actual time constant of the pixel. However, these sources of bias and error do not fully account for the differences observed here, and further investigation is ongoing.

VI. Conclusion

We have demonstrated a novel PPFT system and algorithm capable of extracting τ on a per-pixel basis on the camera level, without needing direct control of the ROIC. To the best of our knowledge, this is both the first system reported capable of doing so and also the first time a per-pixel τ map has been reported. This exciting new capability will allow a more complete characterization of microbolometer cameras. Paired with current per-pixel NEΔT measurements, we can now generate per-pixel FOM maps.

This additional information will aid bolometer manufacturers in troubleshooting and refining fabrication processes, enabling more rapid and cost-effective research and development. It will also assist system integrators in setting component specifications for camera cores and delivery compliance testing.

As a part of the proof of principle testing of a FLIR Tau camera, we performed a preliminary comparison to our current method of record, i.e., the LMS. These limited results seem to indicate a correlation between the two measurement techniques in that both indicate regions of faster and slower τ in the FPA. However, further investigation is required into why there is a nearly 9% discrepancy between the methods.

In the near future, there is still work to be done in refining the system, especially the uniformity of the illumination reaching the camera. Alternate approaches to beam management, including optimized expander placement, an integrating sphere, or a beam shaper [40], [41], are under consideration. Improvements to source uniformity are our first priority, as they will provide more accurate and precise results, allowing us to make better comparisons to the LMS technique as we investigate differences. Currently, we have only demonstrated the ability to test cameras without optics (i.e., bare FPA window), but we believe that the technique should be easily extensible for use with a lens assembly. There also remains some minor hardware work on the DAQ chain to allow the collection of raw data from cameras using a protocol other than Camera Link.

Further in the future, we hope to demonstrate the ability to independently extract thermal time constants and information about the ROIC transfer function characteristics, e.g., the electrical time constant associated with a switch capacitor network or integrator of a continuous bias system. However, extraction of two time constants from a data set is nontrivial and may only succeed under certain conditions [39], [42]. We believe that our hardware provides enough timing precision if appropriate algorithms can be found to perform the parameter extraction.

References

[1] G. D. Skidmore, C. J. Han, and C. Li, "Uncooled microbolometers at DRS and elsewhere through 2013," *Proc. SPIE*, vol. 9100, May 2014, Art. no. 910003, doi: 10.1117/12.2054135.

[2] X. Gu, G. Karunasiri, G. Chen, U. Sridhar, and B. Xu, "Determination of thermal parameters of microbolometers using a single electrical measurement," *Appl. Phys. Lett.*, vol. 72, no. 15, pp. 1881–1883, Apr. 1998, doi: 10.1063/1.121214.

[3] R. P. G. Karunasiri, G. Xu, G. X. Chen, and U. Sridhar, "Extraction of thermal parameters of microbolometer infrared detectors using electrical measurement," *Proc. SPIE*, vol. 3436, pp. 668–674, Oct. 1998, doi: 10.1117/12.328067.

[4] K. M. Muckensturm *et al.*, "Measurement result of a 12 μm pixel array based on a novel thermally isolating structure using a 17 μm ROIC," *Proc. SPIE*, vol. 9819, May 2016, Art. no. 98191N, doi: 10.1117/12.2223608.

[5] F. Utermöhlen and I. Herrmann, "Model and measurement technique for temperature dependent electrothermal parameters of microbolometer structures," in *Proc. Symp. Design, Test, Integr., Packag. MEMS/MOEMS (DTIP)*, 2013, pp. 1–6. [Online]. Available: https://ieeexplore.ieee.org/document/6559403

[6] A. J. Syllaios, M. J. Ha, W. L. McCardel, and T. R. Schimert, "Measurement of thermal time constant of microbolometer arrays," *Proc. SPIE*, vol. 5783, pp. 625–630, May 2005, doi: 10.1117/12.603153.

[7] P. Lambkin, N. Folan, and B. Lane, "Simple technique for the measurement of thermal time constants of microbolometer structures," in *Proc. IEEE 1999 Int. Conf. Microelectronic Test Struct.*, vol. 12, 1999, pp. 179–183. [Online]. Available: https://ieeexplore.ieee.org/document/766239

[8] R. Blackwell *et al.*, "17 μm FPA technology at BAE Systems," *Proc. SPIE*, vol. 7298, May 2009, Art. no. 72980P, doi: 10.1117/12.819389.

[9] D. P. Lacroix, "Microbolometer FPA time constant measurement based on a slow chopper," BAE Syst., Nashua, NH, USA, Tech. Rep., 2017.

[10] J. G. Zeibel, C. Marshall, J. Pelligrino, and P. Perconti, "The development of short time constant uncooled LWIR cameras for mobile operations," in *Proc. Mil. Sens. Symp. Passive Sensors*, 2006, pp. 1–25.

[11] J. Zeibel, "Standard laboratory testing description," U.S. Army RDECOM CERDEC Night Vis. Electron. Sensors Directorate, Fort Belvoir, VA, USA, Tech. Rep., 2007.

[12] F. Généreux *et al.*, "On the figure of merit of uncooled bolometers fabricated at INO," *Proc. SPIE*, vol. 9819, May 2016, Art. no. 98191U, doi: 10.1117/12.2228863.

[13] C. Waddle, T. Pitt, S. Wood, W. Edwards, and B. Yeske, "Advanced 17 μm uncooled infrared development for small unmanned aerial systems," in *Proc. Mil. Sens. Symp., Passive Sensors*, 2010, pp. 1–20.

[14] J. Dawson, M. Chambliss, S. Anacker, S. Wood, and K. Armstrong, "Methodologies and results for correlation of field and laboratory performance of uncooled IR missile seekers," in *Proc. Mil. Sens. Symp., Passive Sensors*, 2005.

[15] Z. Xu, L. N. Phong, and T. D. Pope, "Performance prediction and characterization of highly insulated microbolometers for space applications," *Proc. SPIE*, vol. 8250, Feb. 2012, Art. no. 82500K, doi: 10.1117/12.909573.

[16] D. Waldron and R. Lohrmann, "Apparatus and method to perform per-pixel microbolometer camera time constant measurements," U.S. Patent 1 616 690,163, Nov. 21, 2019.

[17] Timing comparison chart. *Vincent Associates Uniblitz Shutter Systems, Data Sheet*. Accessed: Jul. 2019. [Online]. Available: https://www.uniblitz.com/wp-content/uploads/2016/02/timingcomparisonchart.pdf

[18] L5-S-WCCL. *Access Laser*. Accessed: Jul. 2019. [Online]. Available: https://www.accesslaser.com/wp-content/uploads/l5-s-wccl-rev_a3.pdf

[19] *Model AGM-406B1 IR Acousto-Optic Modulator Frequency Shifter, Lambda Photo*. Accessed: Jul. 2019. [Online]. Available: https://www.lambdaphoto.co.uk/pdfs/intraaction/model%20agm-406b1%20pg9.pdf

[20] Swabian Instruments GmbH. *Time Tagger Series*. Accessed: Jul. 2019. [Online]. Available: https://www.swabianinstruments.com/static/downloads/timetagger.pdf

[21] M. Schlagmüller, "Time tagger quotation," Swabian Instrum. GmbH, Stuttgart, Germany, Dec. 2018.
[22] Stanford Research Systems. *Frequency Standards FS725—Benchtop Rubidium Frequency Standard*. Accessed: Dec. 2019. [Online]. Available: https://www.thinksrs.com/downloads/pdfs/catalog/fs725c.pdf
[23] S. Frehe, "Core2max timestamp accuracy with external time source," IO Industries, London, ON, Canada, Tech. Rep., Dec. 2018.
[24] E. Longley, "Core2max timestamp accuracy with external time source," IO Industries, London, ON, Canada, Tech. Rep., Jan. 2019.
[25] R. Blackwell, "Camera frame time jitter," BAE Syst., Nashua, NH, USA, Tech. Rep., Nov. 2018.
[26] D. Johnson, "Frame timestamping," Vivid Eng., Shrewsbury, MA, USA, Tech. Rep., Dec. 2018.
[27] Texas Instruments. *DS90CR287/DS90CR288A +3.3V Rising Edge Data Strobe LVDS 28-Bit Channel Link—85 MHz*. Accessed: Jul. 2019. [Online]. Available: https://www.ti.com/lit/ds/symlink/ds90cr288a.pdf
[28] Stanford Research Systems. *Synthesized Clock Generator*. Accessed: Jul. 2019. [Online]. Available: https://www.thinksrs.com/downloads/pdfs/catalog/cg635c.pdf
[29] Keysight Technologies. *Keysight Technologies 33500B and 33600A Series Trueform Waveform Generators 20, 30, 80, 120 MHz*. Accessed: Jul. 2019. [Online]. Available: https://literature.cdn.keysight.com/litweb/pdf/5992-2572en.pdf
[30] AInnoTech. *Model GE Series High Power Acousto-Optic Modulator Driver*. Accessed: Jul. 2019. [Online]. Available: http://ainnotech.com/ainnotech/pdf/02/4_2/2IA-GE%20series-high%20power%20power%20modulator%20driver.pdf
[31] R. Gabels and R. Roberts, *Signals Linear System*, 3rd ed. Hoboken, NJ, USA: Wiley, 1987.
[32] B. Lathi, *Signal Processing and Linear Systems*, 1st ed. Oxford, U.K.: Oxford Univ. Press, 1998.
[33] F. Moisy. (2016). *Ezyfit a Free Curve Fitting Toolbox for MATLAB*. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/10176-ezyfit-2-44
[34] J. Jacquelin. *Regressions et Equations Integrales, Regressions Non Lineaires Des Genres: Puissance, Exponentielle, Logarithme*. Weibull. Accessed: Jul. 2019. [Online]. Available: https://fr.scribd.com/doc/14674814/regressions-et-equations-integrales
[35] D. Halmer, G. von Basum, P. Hering, and M. Mürtz, "Fast exponential fitting algorithm for real-time instrumental use," *Rev. Sci. Instrum.*, vol. 75, no. 6, pp. 2187–2191, Jun. 2004, doi: 10.1063/1.1711189.
[36] M. A. Everest and D. B. Atkinson, "Discrete sums for the rapid determination of exponential decay constants," *Rev. Sci. Instrum.*, vol. 79, no. 2, 2008, Art. no. 023108, doi: 10.1063/1.2839918.
[37] P. D. Kirchner, W. J. Schaff, G. N. Maracas, L. F. Eastman, T. I. Chappell, and C. M. Ransom, "The analysis of exponential and nonexponential transients in deep-level transient spectroscopy," *J. Appl. Phys.*, vol. 52, no. 11, pp. 6462–6470, Nov. 1981, doi: 10.1063/1.328595.
[38] M. Mazurenka, R. Wada, A. J. L. Shillings, T. J. A. Butler, J. M. Beames, and A. J. Orr-Ewing, "Fast Fourier transform analysis in cavity ring-down spectroscopy: Application to an optical detector for atmospheric NO_2," *Appl. Phys. B, Lasers Opt.*, vol. 81, no. 1, pp. 135–141, Jul. 2005, doi: 10.1007/s00340-005-1834-1.
[39] G. Bao and D. Schild, "Fast and accurate fitting and filtering of noisy exponentials in Legendre space," *PLoS ONE*, vol. 9, no. 3, Mar. 2014, Art. no. e90500, doi: 10.1371/journal.pone.0090500.
[40] A. Laskin and V. Laskin, "Imaging techniques with refractive beam shaping optics," *Proc. SPIE*, vol. 8490, Oct. 2012, Art. no. 84900J, doi: 10.1117/12.930284.
[41] A. Laskin, V. Laskin, and A. Ostrun, "Beam shaping optics to enhance performance of interferometry techniques in grating manufacture," *Proc. SPIE*, vol. 10513, Feb. 2018, Art. no. 105130F, doi: 10.1117/12.2287528.
[42] L. J. Curtis, H. G. Berry, and J. Bromander, "Analysis of multi-exponential decay curves," *Phys. Scripta*, vol. 2, nos. 4–5, pp. 216–220, Oct. 1970, doi: 10.1088/0031-8949/2/4-5/015.

Dennis L. Waldron was born in MD, USA, in 1988. He received the B.S. degree (Hons.) in electrical and computer engineering from Lafayette College, Easton, PA, USA, in 2010, and the Ph.D. degree in electrical and computer engineering from the University of Virginia, Charlottesville, VA, USA, in 2015.

Since 2015, he has been with the U.S. Army Night Vision and Electronic Sensors Directorate, Fort Belvoir, VA, USA, leading research, development, and test efforts for uncooled long-wave infrared sensors and related systems.

Dieter J. Lohrmann was born in Weisshorn, Germany, in 1960. He received the B.S. and M.S. degrees in electrical engineering and the Ph.D. degree in physics from The Catholic University of America, Washington, DC, USA.

He is currently with the U.S. Army Night Vision and Electronic Sensors Directorate, Fort Belvoir, VA, USA, leading research, development, and test efforts for uncooled long-wave infrared sensors and related systems.