SANDIA REPORT
SAND2007-0311
Unlimited Release
Printed March 2007

Widefield Laser Doppler Velocimeter: Development and Theory

Phillip L. Reu, Bruce D. Hansche, and Jordan E. Massad

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd.
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2007-0311
Unlimited Release
Printed March 2007

Widefield Laser Doppler Velocimeter: Development and Theory

Phillip L. Reu, Bruce D. Hansche and Jordan E. Massad
Applied Mechanics Development
Sandia National Laboratories
P.O. Box 5800
Albuquerque, New Mexico 87185-1070

Abstract

The widefield laser Doppler velocimeter is a new measurement technique that significantly expands the functionality of a traditional scanning system. This new technique allows full-field velocity measurements without scanning, a drawback of traditional measurement techniques. This is particularly important for tests in which the sample is destroyed or the motion of the sample is non-repetitive. The goal of creating "velocity movies" was accomplished during the research, and this report describes the current functionality and operation of the system. The mathematical underpinnings and system setup are thoroughly described. Two prototype experiments are then presented to show the practical use of the current system. Details of the corresponding hardware used to collect the data and the associated software to analyze the data are presented.

CONTENTS

FIGURES

TABLES

NOMENCLATURE

Acronyms

BI2	binary file type for WLDV data storage (previous format)
BI3	binary file type for WLDV data storage (current format)
BS	beam splitter
CIN	Vision research Phantom camera image file format
CCD	charge-coupled device
CMOS	complimentary metal oxide semiconductor
DC	Direct current (i.e. not time-varying)
DDS	direct digital synthesis
DOE	Department of Energy
EOM	Electro-optic modulator
ESPI	Electronic speckle pattern interferometry
FFT	Fast Fourier transform
FPGA	field programmable gate array
JTFA	joint time frequency analysis
LDV	laser Doppler velocimetry
PVDF	Polyvinylidene flouride
SNL	Sandia National Laboratories
STFT	Short-term Fourier transform
TSPI	Temporal speckle pattern interferometry
WLDV	Widefield laser Doppler velocimeter

Variables

A_{pixel}	Pixel area
E	Electric field (of light)
d	Diverging lens factor
f	Frequency (Hz)
$f/\#$	f-number of an optical lens
k	Sensitivity vector
m	Number of speckles
n	Alias integer number
N	Number of samples
r	Radius of turntable
s	Speckle diameter
v	velocity
x	x-dimension, typically mm's or pixels
y	y-dimension, typically mm's or pixels
t	time
T	Time or camera frame rate
α, β, γ	Beam splitter ratios
α	Rotation angle
λ	Wavelength of laser light
μ	Modulation depth
ω	Rotational velocity (rad/s)

Ω	Rotation rate (rev/min)
ϕ	Phase of the light wave
A,B,C	Wave amplitude constants
I	Irradiance (sometimes referred to as intensity)

1. INTRODUCTION

Laser Doppler Velocimetry (LDV) has been an extremely useful tool for measuring the velocity of vibrating surfaces. It has many benefits that make it an ideal measurement solution for a large number of experimental applications. The benefits include the noncontact nature of the measurement, which prevents mass loading, whereby a laser probe beam is the only thing in contact with the surface. This is especially critical as work continues in the microelectromechanical arena, where the item being tested often has dimensions in the micron scale. The small measurement spot size of a typical velocimeter also lends itself to high data density via scanning the measurement point across the surface. The advanced signal processing and high-speed data acquisition also give the LDV a wide velocity measurement range from picometers/s to 20 meters/s.

Additionally, displacement information can be obtained either through integrating the velocity or, more typically, by using fringe-counting concepts. Fringe counting has a significant drawback in that it cannot determine the relative displacement between two points on the surface for a single actuation, as scanning of the probe beam inherently causes the loss of an unknown number of the fringes. This highlights a traditional drawback to LDVs in that they are a *single-point* measurement. With standard LDV techniques, full-field information is built up via scanning the single point over the area of interest. This typically has two drawbacks: it precludes both direct full-field displacement measurement and full-field measurement of transient, nonrepetitive events. Finally, scanning is especially ineffectual when the test item is destroyed during the test.

These drawbacks in traditional LDV measurements have attempted to be alleviated from two directions: by creating faster scanning or more measurement channels in a traditional LDV, or using electronic speckle pattern interferometry (ESPI) to provide full-field displacement results as a function of time. The first approach, which is most similar to traditional LDV, adds more simultaneous measurement locations either by adding more photodiodes and the required decoding for each desired channel, or by using a high-speed camera. The difficulty is in finding a cost-effective means of creating a large number of parallel LDVs. The parallel method has been demonstrated with up to 16 simultaneous channels,[1,2] with one paper proposing up to 625 channels, although no hardware for this extremely high channel count system was shown.[3] The second scheme automates the typical point scanning by using the individual pixels of a high-speed CMOS imaging sensor, as an equivalent to a single photodiode.[4,5] By illuminating the entire surface, different locations of the imaged object can then be quickly scanned electronically. The drawback is that to have useful acquisition rates, only one pixel, or at most a few, can be used. This, in essence, creates an electronic scanning single-point measurement similar to the standard mechanical LDV solution of using mirrors to scan the measurement point over the surface.

The full-field displacement method tends to approach the problem from ESPI methodology rather than LDV methodology. Time speckle pattern interferometry (TSPI) uses a camera to count the fringes going past any location (pixel) in the image. This paper notes important and interesting similarities between “fringe-counting” and Doppler frequency shifting that can help with the development and understanding of both methods. In regards to TSPI, it has typically

been limited to very slow speeds because the camera frame rates used, even for high-speed cameras, are not fast enough for anything other than nearly quasistatic motion. This is especially true when compared to laser Doppler measurements. In the TSPI literature, a number of approaches are used to decode the time-varying irradiance of the pixels. Frequency-based methods are the most similar to Doppler velocimetry,[6-9] but re-curvature methods,[10] scanning phase methods,[11,12] and temporal branch-cut methods have also been utilized.[13,14]

Another unique approach to obtain full-field velocity information is a system termed optically phase-locked ESPI.[15] This system uses the Doppler information from a single measurement location to control a local oscillator to lock and track the frequency and phase of portions of the field of view that have the same temporal Doppler signature as the lock point. This makes the fringes of equal Doppler frequency stationary and allows the velocity distribution to be determined.

This paper demonstrates the theoretical and practical development of a widefield laser Doppler velocimeter (WLDV) that is different than methods found in the literature just outlined. The great advantage of this technique over others is its ability to image the entire velocity surface simultaneously. This is a critical improvement in non-repetitive motion or destructive testing. Additionally, it could lead to a more noise immune ESPI configuration for measuring displacements in noisy environments.

As part of the development, a comparison and reconciliation between TSPI (fringe counting) and Doppler shifting is presented that may spark ideas regarding better approaches for both techniques. The unification of TSPI and Doppler has important consequences on how one arranges the interferometric experiments in terms of laser power, lenses used, and so forth. Of special note is the possibility of having many speckles per pixel (or on a photodiode) and the potential benefits and drawbacks of unresolved speckles.

To demonstrate the functioning of the WLDV hardware, three experiments are presented: the measurement of a swinging pendulum, a block on a rotating turntable, and a mode shape of a vibrating film. The pendulum experiment shows the measurement of a time-varying but nearly spatially constant velocity field. The turntable results show a spatially and temporally varying velocity field ($v(x,y,t)$) measured by the WLDV and processed to create a velocity movie. A velocity movie is defined as an animation of the instantaneous velocity distribution of an object surface (a velocity image) as it varies in time. Both of these experiments highlight the bandwidth limitations of using a digital camera, even a high-speed camera, for sampling the Doppler frequency. The low camera sampling rate leads naturally to aliased velocity data. We present sub-Nyquist sampling methods to exploit the aliased data and "unwrap" them; they significantly extend the bandwidth of the system up to 10 times as is shown in this paper. It is important to note that for the WLDV, the bandwidth limitations are not an absolute maximum-velocity restriction, but only a constraint of the velocity distribution over the imaged area.

1.1. Widefield laser Doppler theory and mathematics

1.1.1. Reconciling TSPI and Doppler

In optical measurement literature, there is a separation between the LDV and the ESPI communities that is mostly a result of the quantities measured and the typical experiments

conducted, rather than a difference in the fundamental physics of the experiment. However, it is not intuitively obvious that the measurements are identical, in that ESPI measures a phase change to determine displacement and Doppler measures the frequency to find the velocity. Therefore, it is interesting to note that Doppler is an optical derivative of the ESPI formulation; that is, velocity is the derivative of displacement, and frequency is the derivative of the instantaneous phase change. This similarity has been noted in the literature by Briers in his paper[16] reconciling ESPI and LDV, as well as being briefly mentioned by Cloud in his book on optical methods[17] and implicitly understood by Joenathan in his papers on TSPI.[6-8] It seems desirable to the current authors to highlight the similarities and differences in the approach, to briefly derive the equations, and to reconcile the approaches in more detail than is contained in any of these references. For simplicity in illustrating both concepts, a Michelson interferometer arrangement, shown in Figure 1, is used for the discussion. However, the actual arrangement of the WLDV is in the Mach-Zehnder format as will be elucidated later. This detail does not change either the mathematics or the principles discussed. Both interferometer types interfere an object beam and a reference beam together before measuring a time-varying irradiance or spatial fringe pattern at a detector, such as a photodiode or digital camera. Traditionally, this is a quasistatic irradiance distribution in the case of ESPI or time-varying Doppler frequency in the LDV paradigm. (In TSPI, this is different in that a time-varying irradiance is used.) To begin the discussion, a useful visualization of fringe counting is in order. The fringes formed by the interference of the reference and object beams create a pattern of dark and light bands in space in front of the detector as illustrated in Figure 1. The motion of the object Δx moves this fringe field across the detector—creating so called "fringe counting."

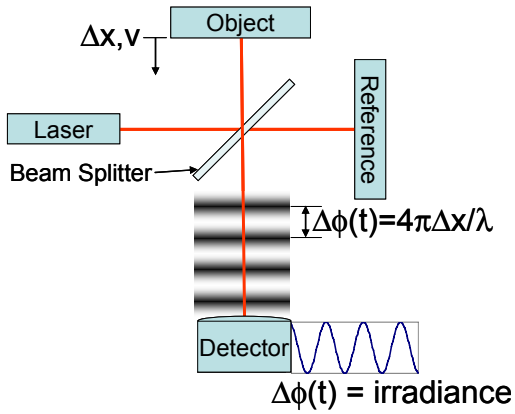

Figure 1. Michelson interferometer arrangement.

This concept is outlined mathematically using complex notation starting with the electromagnetic field equation of light emitted from a laser:

$$E(t) = \mathrm{Re}\left\{Ce^{j\omega_\lambda t}\right\}, \tag{1}$$

where C is the complex amplitude of the signal, $j = \sqrt{-1}$, ω_λ is the frequency of the light entering the system, and t is time. The real operator Re will be omitted from subsequent equations, but it should be known by the reader that only the real part is being examined. This notation is greatly simplified from the entire vector equation shown as a complete description in many sources, [17] but it is correct for the uniform distribution with a planar wavefront in a single direction found in the Michelson interferometer. The beamsplitter (BS) creates an object and reference beam. After

scattering from the surface of the moving object, denoted by subscript *o*, the field of the object light beam is described by

$$E_o(t) = Ae^{j(\omega_o t + \phi_o(t))}, \tag{2}$$

where, A is the field amplitude after reflecting from the object and returning through the beamsplitter, ω_o is the object beam frequency, and $\phi_o(t)$ is the time-dependent phase shift caused by the object. In this case, the frequency is equivalent to the Doppler-shifted frequency

$$\omega_o = \omega_\lambda \pm \omega_d\,, \tag{3}$$

where ω_d is the Doppler frequency in radians per second. Whether the action of the object surface on the electromagnetic wave changes either the phase or the frequency dictates whether the response is thought of as ESPI or LDV. Strictly speaking, the phase shift (or frequency change) is a function of both space and time. This spatial dependence is used by the WLDV to measure the velocity at all points on the surface, but has been dropped in this derivation and can be thought of as being equivalent to looking at the information for a single pixel or photodiode (assuming large speckles). The field equation describing the reference beam, denoted by subscript *r*, after returning through the beamsplitter is

$$E_r(t) = Be^{j(\omega_r t + \phi_r)}\,, \tag{4}$$

where B is the amplitude after returning from the reference surface through the beamsplitter, ω_r is the reference beam frequency, and ϕ_r is the phase shift from the reference surface. In this case, ω_r is equivalent to the optical frequency ω_λ. Upon combining the two beams at the detector, the total electromagnetic field is the sum of the object and reference fields

$$E_T(t) = E_o(t) + E_r(t)\,. \tag{5}$$

Detectors such as photodiodes and digital cameras respond to the irradiance of the signal. The irradiance is defined as the square modulus of the electromagnetic amplitude, which can be calculated as:

$$I(t) = E_T E_T^*\,, \tag{6}$$

where E_T^* represents the complex conjugate of the total field. Using Equations (2), (4), and (5), the resulting irradiance measured at the detector simplifies to

$$I(t) = I_{DC} + 2AB\cos\left[(\omega_r - \omega_o)t + \Delta\phi(t)\right], \tag{7}$$

where I_{DC} represents the DC components of the irradiance and $\Delta\phi(t)=\phi_r-\phi_o(t)$ represents the phase change. This equation is both the Doppler and ESPI description of the light, depending on how the two terms in the cosine function are defined, but if inspected closely, the reader may notice some mathematical legerdemain. That is, a time varying phase is by definition the instantaneous frequency, and therefore the ω terms and the ϕ terms are redundant in this presentation. However, for purposes of illustration, if the quasistatic case is assumed, then $\omega_r=\omega_o$. Therefore, the Doppler frequency is zero and Equation (7) results in the traditional ESPI formulation. Alternatively, if one assumes that $\Delta\phi$ is not a function of time, then Equation (7) results in the typical Doppler formulation. Both descriptions are appropriate and physically identical to the sinusoidal irradiance change experienced as the fringes are seen moving over a pixel in time-varying ESPI or fringe counting in LDVs. This derivation shows the similarity between the speckle pattern techniques (including TSPI) and laser Doppler velocimetry, which is useful in comparing and contrasting the two techniques. Of course, in a practical interferometry experiment, both the quasi-static phase change $\Delta\phi(t)$ and the Doppler term $\omega_o=\omega_\lambda\pm\omega_d$ are functions of space as well as time, and the intensity described by Equation (7) is a function of space and time, $I(x,y,t)$. In a "classical" ESPI experiment, a single interferogram is used to

compare the position of an object at two times, before and after some motion. This results in a fringe pattern image $I(x,y)$ which, in the early years of ESPI (and its predecessor, holographic interferometry) was simply examined for "anomalies", or fringe patterns characteristic of flaws, etc. With the advent of "phase stepping", several interferograms were taken as the reference phase ϕ_r was changed in a controlled fashion. This essentially utilized the time element of Equation (7) as an independent variable, allowing several instances of Equation (7) at different times to be solved for $\Delta\phi(x,y)$. TSPI just applies this concept at a lot more time steps, enabling measurement of $\Delta\phi(x,y,t)$. Speed up the measurement even more so that we measure frequency instead of phase, and we have the Doppler approach. Thus, we are dealing with a closely related family of techniques, all based on the same physical principle. Deciding which one to use depends mainly on practical considerations, such as the speed of available cameras, modulators, and so forth.

From the authors' perspective, two important advantages are to be gained by approaching interferometric velocity *and* displacement measurements from the Doppler formulation. By thinking about the information as a frequency, modulation techniques to heterodyne the high-frequency Doppler information are easily imagined. This is one of the key differentiating ideas used for the development of the WLDV. Additionally, speckle size and decorrelation have not been considered a problem for Doppler velocimetry, and this freed us to investigate having large numbers of speckles on a given pixel without regard for the historical case of having fully resolved speckles. A further conclusion is that having large numbers of speckles on the detector is advantageous in many cases—not just for the WLDV, but for ESPI as well, and will be briefly discussed in a later section.

1.1.2. Widefield Laser Doppler Theory—Derivation of System Equations

Having derived and compared Doppler and fringe counting, the derivation of the WLDV system equations will be done showing the use of amplitude modulation to shift the high-frequency Doppler signal down into the bandwidth of a digital camera. Doppler frequencies for surface-velocity measurement are characteristically in the kHz to MHz range. This frequency is too high for a typical high-speed camera and is the reason fast photodiodes are used in an LDV. Therefore, to create a velocity image of all points on the surface simultaneously, the WLDV uses optical heterodyning to shift the high-frequency Doppler signal down into the frame rate of a high-speed camera. In general, heterodyning relies on the trigonometric property that the *product* of two harmonic signals consists of terms containing the *sum* and *difference* of the frequencies of the original two signals. In an electronic circuit, the two signals are added, and the result passed through a nonlinear element to create product terms. In a basic optical heterodyne system, the object beam and the reference beam frequencies are added at the beam combiner. The camera responds to *irradiance*, which is the square of the amplitude, so this acts as the nonlinear element. If either the reference beam or both beams are amplitude modulated at a frequency, various combinations of sum and difference terms are generated. All terms at the optical frequency and above are too fast for the detector to respond to and are seen as a constant term. Optical heterodyning is accomplished in the WLDV with a Mach-Zehnder interferometer setup using a 532-nm wavelength laser and an electro-optic modulator (EOM) to amplitude modulate the laser signal. A schematic of the system is shown in Figure 2.

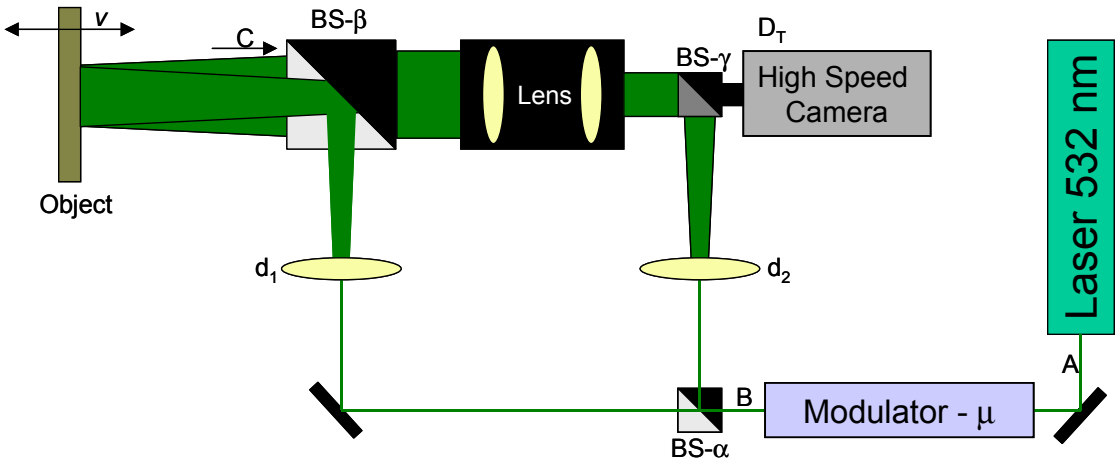

Figure 2. Schematic of the widefield laser Doppler velocimeter.

Using this arrangement, the relationship of the object velocity and modulation frequency is derived as follows. Equation (1) describes the electric field at the laser. The equation describing the light after it passes through the modulator is

$$E_m(t) = Ce^{j\omega_\lambda t}\left(\frac{1}{2} + \mu e^{j\omega_m t}\right), \tag{8}$$

where μ is the modulation depth and ω_m is the modulation frequency. After passing through BS-α, lens d_1, BS-β, and reflecting off of the moving target surface, the Doppler shifted light recombines with the reference beam at BS-γ. Then the equation for the object beam at the camera is

$$E_O(t) = d_1\alpha\beta\gamma(1-\beta)e^{j\omega_d t}E_m(t), \tag{9}$$

where d_1 is a beam divergence coefficient and α, β, and γ are beam ratios ranging from zero to one. Accordingly, the reference beam, which comprises the light that passes lens d_2, is described by

$$E_R(t) = d_2(1-\alpha)(1-\gamma)E_m(t)\ . \tag{10}$$

Finally, the total beam at the camera is represented by the sum of the object and reference fields

$$E_T(t) = E_O(t) + E_R(t)\,. \tag{11}$$

As mentioned in the Section 1.1.1, the camera measures only irradiance, which is computed in Equation (6). Using Equations (9) thru (11), the resulting modulated irradiance measured at the camera simplifies to

$$I(t) = I_{DC} + C_m\cos(\omega_m t) + C_d\cos(\omega_d t) + C_{md}\left[\cos((\omega_m - \omega_d)t) + \cos((\omega_m + \omega_d)t)\right], \tag{12}$$

where the coefficients have the forms

$$
\begin{aligned}
I_{DC} &= \frac{1+4\mu^2}{2\mu} C_m \\
C_m &= \frac{\left(d_1\alpha\beta\gamma(1-\beta)\right)^2\mu + \left(d_2(1-\alpha)(1-\gamma)\right)^2\mu}{2} \\
C_d &= \frac{1+4\mu^2}{\mu} C_{md} \\
C_{md} &= \frac{d_1 d_2 \alpha\beta\gamma(1-\alpha)(1-\beta)(1-\gamma)\mu}{2}.
\end{aligned}
\tag{13}
$$

Note that all terms on the order of the optical frequency (ω_λ) are integrated to apparent DC (i.e. constant irradiance) terms on the detector. There are three frequency-dependent components to Equation (12), all of which comprise measured data. The C_m term represents the laser modulation and the C_d term represents the Doppler signal that is directly proportional to object velocity. The C_{md} term represents the two heterodyned difference components that shift the Doppler signal up and down by the modulation frequency.

For the Mach-Zehnder interferometer setup used in the WLDV, the out-of-plane velocity of the surface is directly proportional to the Doppler frequency

$$\omega_d = \frac{4\pi v}{\lambda} \text{ or } f_d = \frac{2v}{\lambda}, \tag{14}$$

where f_d is the Doppler frequency in Hz, v is the surface velocity, and λ is the wavelength. (Theoretically, the WLDV arrangement can also be used for measuring in-plane velocities by changing the laser illumination arrangement and the above Doppler equation, but this is not done here.) Typically, the maximum velocity able to be measured would be limited by the Nyquist rate of the high-speed camera. Because of the heterodyning in the WLDV, this limit is not an absolute maximum velocity limit, but rather a velocity bandwidth limitation as expressed in the following equation:

$$v_{BW} \le \frac{\lambda f_{SR}}{4}, \tag{15}$$

where v_{BW} is velocity bandwidth and f_{SR} is the sample rate (frame rate) of the camera in Hz. In practice, this means that a reference velocity v_r is chosen, and the Doppler frequency $\omega_m=(4\pi v_r)/\lambda$ is used with the modulator. This has the effect of representing the reference velocity with a shifted Doppler frequency term ($\omega_m - \omega_d$) of zero Hz. The WLDV can then measure velocities without aliasing from the reference velocity value to the sum of the reference velocity value and the velocity bandwidth. For the current system with a frame rate of 76 kHz and a 532-nm laser, this translates into a velocity bandwidth of ~10 mm/s. If the velocity bandwidth is exceeded, alias unwrapping can be used to expand this bandwidth as explained and demonstrated later in this report.

1.1.3. Joint Time Frequency Analysis—Tracking Velocity Changes in Time

An important difference between a single-point LDV and the WLDV is in processing the time-varying irradiance signal that is collected. Typically, a single-point LDV uses phase-lock loop methods to track the changing velocity (frequency) through time. This is not possible with the camera system and the thousands of simultaneous measurement points at different velocities being collected. The WLDV uses only the time-varying irradiance of a given pixel to determine

the velocity via frequency domain methods. If the velocity were constant throughout the entire test, as done in the TSPI work by Joenathan,[6] the Fast Fourier Transform (FFT) would contain only a single frequency peak. This is typically an acceptable constraint for displacement measurements, but unacceptable when taking velocity measurements where the velocity is both a function of location on the object and time. For this reason, joint time-frequency analysis (JTFA) methods were used to track the changing Doppler frequency as a function of time and, thereby directly measure the changing velocity. There exists a number of JTFA techniques in the literature including the Gabor Spectrogram, Choi-Williams, etc.[18] For the analysis of the data, the short-term Fourier transform (STFT) was found to be sufficient. The STFT uses a sliding and overlapping FFT window to divide the time history into short time segments where the frequency is then calculated with a traditional FFT. It should be noted that the uncertainty principle dictates that there is a trade-off between time (t) and frequency (f_{min}) resolution that obeys

$$\Delta t \Delta f_{\min} \geq \tfrac{1}{2} \ . \tag{16}$$

A compromise is chosen that balances the frequency and time resolution to obtain the most meaningful result.

1.1.4. Alias unwrapping for bandwidth extension

Sub-Nyquist data sampling can be used to expand the bandwidth of the camera significantly. Sub-Nyquist sampling refers to techniques where the signal is sampled at less than half of the highest frequency in the signal (i.e. the sample is aliased), but using *a priori* knowledge, useful data can still be obtained. There is a body of literature on using sub-Nyquist sampling to aid in data collection, for example, in the Doppler radar community. A velocity at one point on the surface and the relationship of that point to the camera data must be unambiguously known. The start point for alias unwrapping can either be spatially known or temporally known. For instance, a traditional LDV has been used to measure the velocity of a point in the field of view, and this becomes the known velocity at all times for the data. The other constraint is that the surface must be continuous, or a continuous path from the known points to all other points on the surface must be available. This is similar to the phase unwrapping criteria for ESPI.[22] If these two constraints are met, then starting at the known velocity point, the data can be unwrapped using the following equations:

$$f_R = n\left(\frac{f_{SR}}{2} + \chi_n f_A\right), \tag{17}$$

where

$$\chi_n = 1 + 2\left(n - 2\,\mathrm{ceil}\left(\frac{n}{2}\right)\right), \tag{18}$$

and f_R is the correct unaliased Doppler frequency, f_{SR} is the camera frame rate, n is the integer aliasing order, f_A is the measured apparent frequency, and ceil is the ceiling operation, which rounds the argument to the nearest integer. This unwrapping concept is valid in both the time and the spatial axes. This has the benefit of using pixel data on all sides of the point of interest in the data cube for unwrapping and noise suppression purposes. A data cube in this context is the 3D array containing the irradiance data defined for the x_{pixel}, y_{pixel}, locations and t dimensions yielding $I(x_{pixel},y_{pixel},t)$.

1.1.5. Speckles per Pixel Discussion—Competition of Illumination Power and Lens Aperture

Speckle size has an important effect on a number of parameters of interest for this experiment. The subjective speckle field (that imaged on the camera) in ESPI applications is traditionally optimized to have large speckles (fully resolved), that is, a few pixels for each speckle. This is because of the subtractive nature of the mathematics, where decorrelation of the speckles will severely affect fringe contrast, and hence measurement accuracy. A similar speckle decorrelation concept occurs in laser Doppler velocimetry, where speckle drop-out is seen in the signal; that is, when a dark speckle is illuminating the detector, little or no Doppler signal is able to be measured. Fully resolved speckles have been the traditional thinking on the topic of speckle size. We have found in our own experiments that this is not necessarily the optimum arrangement from balancing two competing constraints: irradiance on the detector and speckle size. The short exposure times associated with high frame rates require a large amount of optical power. This is especially true when, as with the WLDV, large fields of view may be involved or poorly reflective surfaces are imaged. For these reasons especially, it is useful to open up the aperture and capture more of the scattered light. This has the result of creating smaller speckles as related by the equation

$$s \approx \lambda f/\#, \tag{19}$$

where s is the speckle diameter and $f/\#$ is the f-number of the lens, which is defined as the focal length divided by the lens aperture. To create large speckles, a small aperture that limits the light gathering and the power on the detector is required. In a number of articles on speckle-size optimization for ESPI, Lehmann has drawn the same conclusion regarding speckle size and what he terms modulation quality or efficiency. He has shown both theoretically[19,20] and experimentally[21] that other competing factors, such as laser power may outweigh the potentially higher modulation possibility of fully resolved speckles. Lehmann relates the modulation level $\langle I_M \rangle$ to the number of speckles m and the average irradiance $\langle I \rangle$, via[21]

$$\langle I_M \rangle \approx \sqrt{\frac{\pi}{m}} \langle I \rangle, \text{ for large } m. \tag{20}$$

Therefore, because $\langle I \rangle$ is proportional to m, the modulation level $\langle I_M \rangle$ is proportional to the square root of the number of speckles

$$\langle I_M \rangle \propto \sqrt{m}. \tag{21}$$

This is because $\langle I \rangle$ is proportional to $(f/\#)^{-2}$, and m is proportional to s^{-2}. That is, the modulation level increases with increasing numbers of speckles until the pixels are saturated. Since we can digitally remove the DC portion of the signal, the optimum aperture is that which just avoids pixel saturation, regardless of the number of speckles per pixel. In the experimental setup for the WLDV, the optimum illumination and aperture results in hundreds of speckles on each pixel. Lehmann in his papers does not calculate the values for hundreds or thousands of speckles, but experiments conducted here in this work generally confirm Lehmann's results for the Doppler situation. We obtain the approximate value for m by dividing the pixel area A_{pixel} by the speckle area (s^2) calculated with Equation (19):

$$m \approx \frac{A_{pixel}}{(\lambda f/\#)^2}. \tag{22}$$

The result of this is that by opening the lens aperture, more light is available for the digital camera, without eliminating the harmonic content that is the Doppler signal. Therefore, for the experiments, unresolved speckles are used, to minimize the exposure time, without losing the velocity information. This fact has implications beyond WLDV operation, and should be considered in ESPI as outlined by Lehmann.

2. EXPERIMENTAL RESULTS

2.1. Experimental Results – Introduction

The WLDV is designed to measure a full-field time-varying velocity distribution. When analyzed, the irradiance data gathered by the high-speed digital camera will result in a data movie that is a time-varying velocity field over the imaged area. We created a set of experiments to supply a controllable velocity field for measurement; including a pendulum that supplied a time-varying velocity, with little spatial variation; and a block on a rotating turntable that supplied a spatially varying but nominally constant velocity in time. A further successful experiment was conducted with an accelerating turntable where the velocity was a function of both space and time, with the results presented in Section 3. While the implementation of the modulator in the WLDV allows any velocity to be heterodyned to the camera frequency, the camera frame rate still imposes some severe velocity bandwidth limitations as discussed previously. This is illustrated by the high-frequency terms in Equation (12), where, without *optical* anti-aliasing filters, aliased data are endemic. However, the situation is not hopeless, and by using *a priori* knowledge, the aliased data can be manually filtered and unwrapped to extend the bandwidth of the system. Manual filtering in this context means removing known alias peaks from the JTFA data (see Section 3.2). For example, the modulator frequency is known and can be zeroed out in the JTFA to assist in tracking the velocity. The following sections detail the experimental results and alias unwrapping.

Figure 3 shows the equipment used in the WLDV for these experiments. A 150-mW Coherent laser with a wavelength of 532 nm was used for creating the object and reference beams. A ConOptics Model 25A EOM and matched amplifier were used for amplitude modulation of the laser. Frequency of the modulation via the EOM was controlled with a function generator. A large plate beamsplitter was used to direct the laser to the object and return the Doppler shifted light to the telephoto lens. The lens was a standard 105-mm c-mount TV zoom lens imaged through a relay lens. The aperture was set to roughly f/4, which yielded approximately 100 speckles per pixel. Two variable neutral-density filters were used to control the relative intensities of the object and reference beams. Both the object and reference beams were diverged by means of microscope objectives. A Vision Research Phantom V 7.0 high-speed monochrome CMOS digital camera was used to obtain the data. The default resolution is 800×600 pixels at 4800 frames per second. For all the data shown in this paper, a smaller field of view of 256×64 pixels was used with a frame rate of 76.923 kHz. Even at the reduced resolution, this yields 16,384 simultaneously measured data points and is an extremely large number of data points when compared to the other methods discussed in the Introduction.

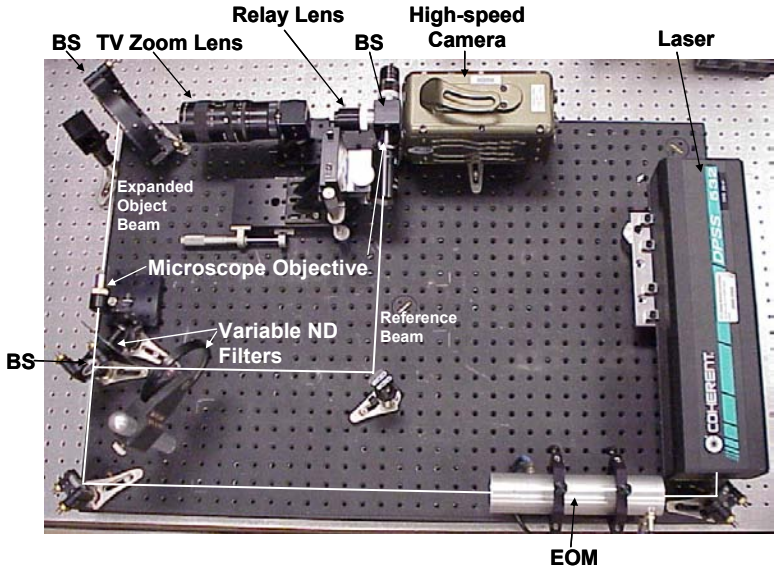

Figure 3. Photograph of laboratory setup and equipment used in the WLDV.

2.2. Pendulum—Time-Varying Velocity Measurement

The first experiment used a 0.56-meter pendulum (1.5-second period) to supply a nominally spatially constant velocity to create a time varying velocity movie. For this case, as the velocity was small enough during the entire period that the modulator frequency (ω_m) was set to zero. The pendulum was pulled back and allowed to swing freely, and data were taken for an entire period. Both the time data and the JTFA results are shown in Figure 4a. The color on the JTFA corresponds to an FFT amplitude at that time slice with arbitrary units related to the pixel amplitude in counts. For both plots, the time-dependent irradiance trace of the data is shown directly below the corresponding JTFA analysis. (The JTFA figures show each windowed FFT plotted vertically, with irradiance representing FFT magnitude.) The velocity bandwidth of the entire pendulum is too large for the frequency bandwidth of the camera, and because of this (and the lack of an optical anti-aliasing filter), the data wrap about the Nyquist frequency (or velocity). This wrapped data is shown in Figure 4a, where the velocity seems to oscillate incorrectly between 0 and 10.2 mm/s with almost all of the data being aliased. With the *a priori* knowledge of where the pendulum reaches the peak of its swing and turns around these data can be unwrapped to obtain the entire velocity profile. To show the time-varying irradiance data more clearly, data for a single pixel at the turnaround point are shown in Figure 4b. The speed decreases to zero and accelerates again as the pendulum starts to swing backwards.

In this report, as we rely heavily on a "spectrogram" display, such as that in the upper portion of Figure 4, a little explanation on how to read these is in order. These plots represent "frequency tracks" or equivalently via Equation 14, "velocity tracks". On the spectrogram (or equivalently the JTFA) one axis represents frequency or velocity as appropriate, and the other represents either time (at a given location) or a space dimension (at a given time). The color or intensity represents FFT magnitude at the corresponding frequency. Typically, the FFT has a definite peak representing the desired velocity data. Sometimes other undesired peaks are present, such as the modulator signal and attempts are made to remove or ignore these spurious peaks in the unwrapping. If the FFT magnitude is thresholded so that all magnitudes below a certain level display as a uniform color (white in Figure 4, blue in various other spectrograms), the FFT peak

(or peaks) define “tracks” in the spectrogram. These tracks represent the velocity data as a function of either time or space.

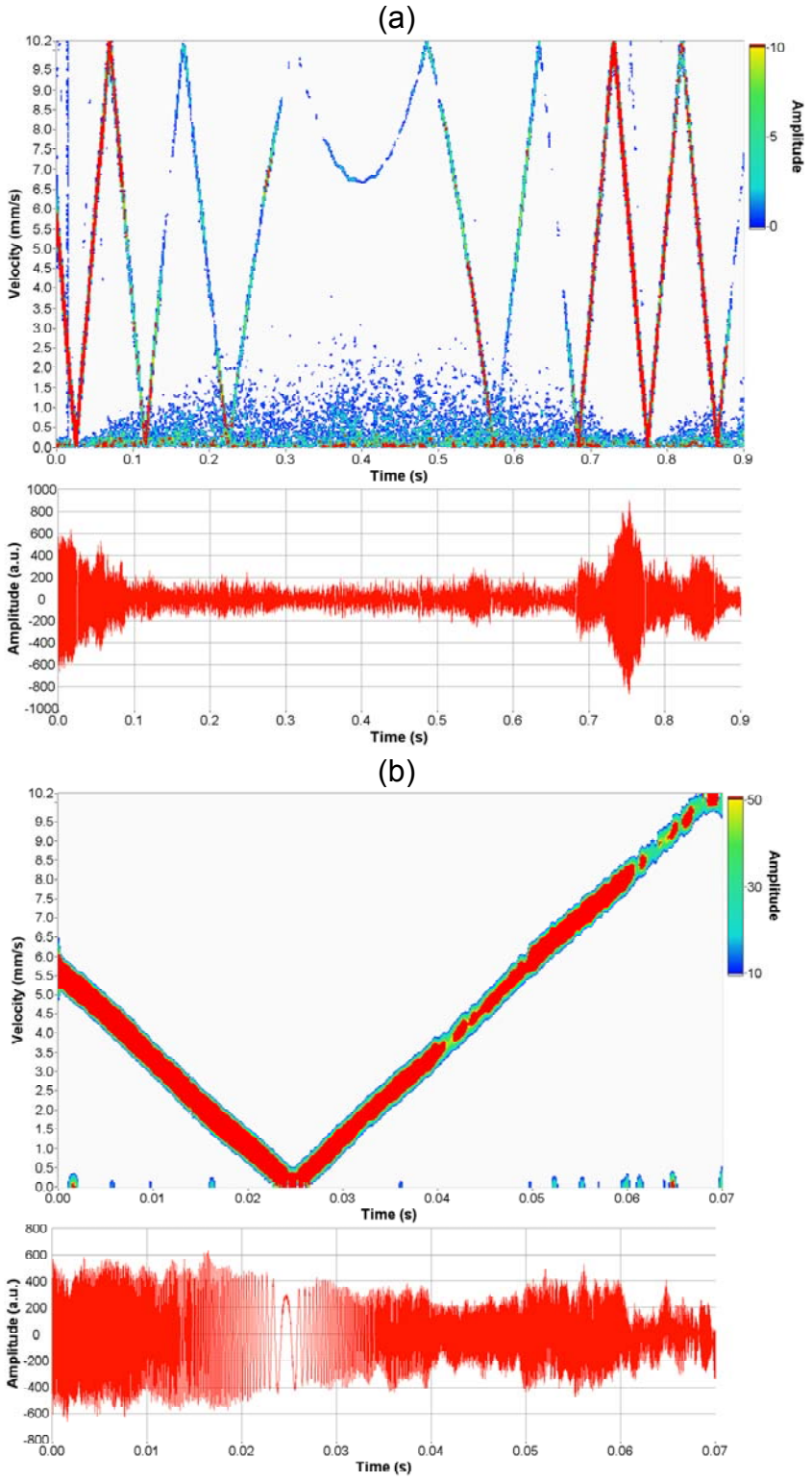

Figure 4. Single-pixel pendulum data with JTFA results above irradiance versus time plots (a) for entire pendulum swing and (b) at the turnaround point.

The aliasing seen in Figure 4a illustrates a potential issue with the WLDV. The limited sampling rate of currently available digital cameras (even high-speed cameras) makes most typical structural dynamics experiments difficult because the velocity ranges create Doppler frequencies exceeding the camera bandwidth. The lack of a good optical anti-aliasing filter means that all the frequencies are measured by the camera. For the pendulum data, the velocity of all the pixels when the pendulum switches directions is known to be zero, thus giving a starting point for unwrapping the aliased data. Using Equation (17), the data for a single pixel were unwrapped and are shown in Figure 5. The inset shows the alias integer numbers and the first fit plane used for unwrapping. Unwrapping is an important detail for making the WLDV practical in a broader range of situations. As can be seen, a bandwidth extension of nearly ten times has been realized by unwrapping the data. That is, a velocity spread of 10 mm/s has been expanded to 100 mm/s. The figure also shows a theoretical velocity profile that visually quantifies the agreement between the unwrapped velocity and the expected velocity profile.

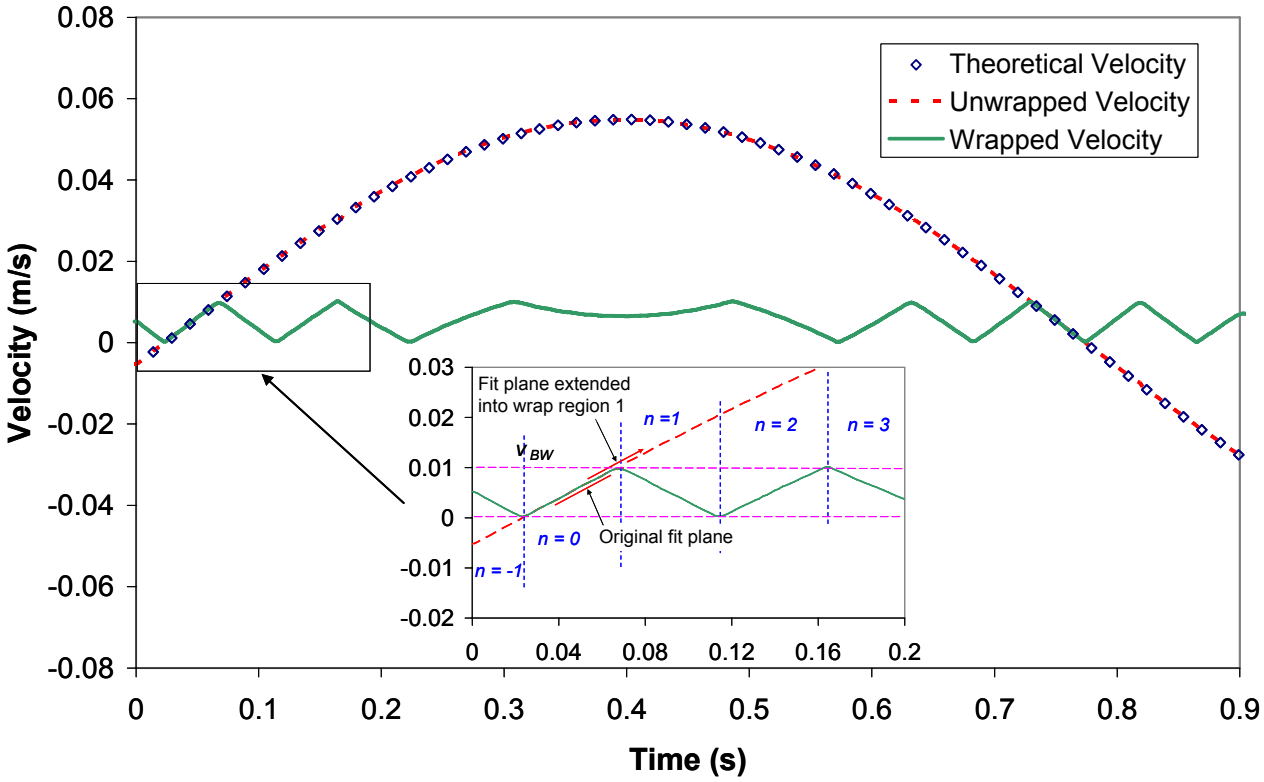

Figure 5. Aliased and unwrapped data from complete pendulum swing in m/s.

2.3. Turntable – Spatially Varying Velocity Measurement

For many applications, the entire velocity bandwidth is outside the detection capabilities of the camera. For these cases, the use of amplitude modulation to heterodyne the signal into the camera bandwidth was demonstrated using a block on a rotating turntable. This supplies a spatially-varying velocity profile to be measured by the WLDV. A sketch of the experimental setup is shown in Figure 6. The viewing angle of the WLDV being perpendicular to the block as it rotates into view has the benefit that any column viewed by the camera remains at a constant velocity. This is because the sensitivity vector of the out-of-plane setup of the WLDV only

measures the portion of the velocity vector directed towards the camera. As illustrated in Figure 6, the change in the angle (α) of the vector is exactly compensated by the changing position being measured (radius r). Therefore, any pixel in a given column in the image has a constant velocity identical to other pixels in the same column. A constant rotation rate (Ω) of 10.2 RPM was used to supply a nominally constant velocity to the block moving toward the WLDV. In Figure 6, r_{OP} is the radial position of a given point on the block that intersects the laser beam at α equals zero.

Using the 105-mm lens and a relay lens, the block was imaged on the camera with a magnification of 0.16. This gives the field of view of 29 mm × 7.3 mm seen in Figure 7a. An arbitrary field of view was chosen centered at a turntable radius of 62 mm, which yields a calculated Doppler frequency at the center of approximately 240 kHz. Without modulation of this signal, it would clearly be outside the bandwidth of the camera, but setting the modulator frequency to this value heterodynes the Doppler frequency at the center of the field of view to zero Hz. The exact pixel column on the camera where the modulator equals the Doppler frequency can be found by plotting the magnitude of column average FFTs in a waterfall plot as shown in Figure 7b. The zero point occurs at column 119, with a modulation frequency value of 240 kHz, which indicates that the velocity at this point is 64 mm/s. The velocity then linearly increases as predicted by theory from column 119 to column 199 and then becomes aliased.

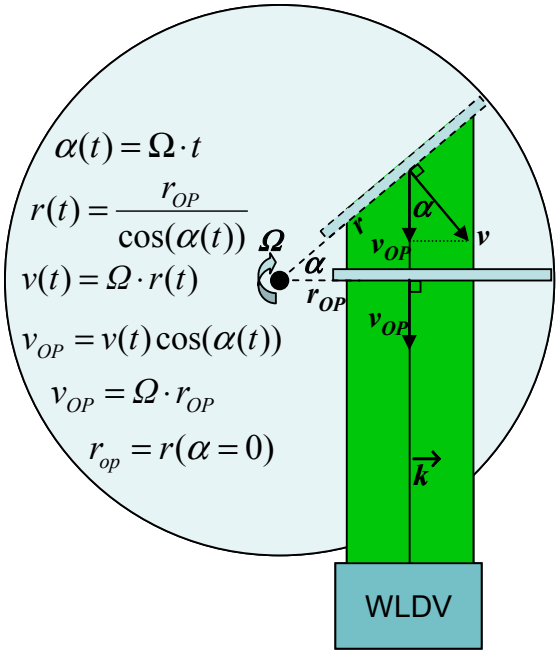

Figure 6. Rotating block on a turntable experimental setup, showing how any given pixel in a column has the same velocity.

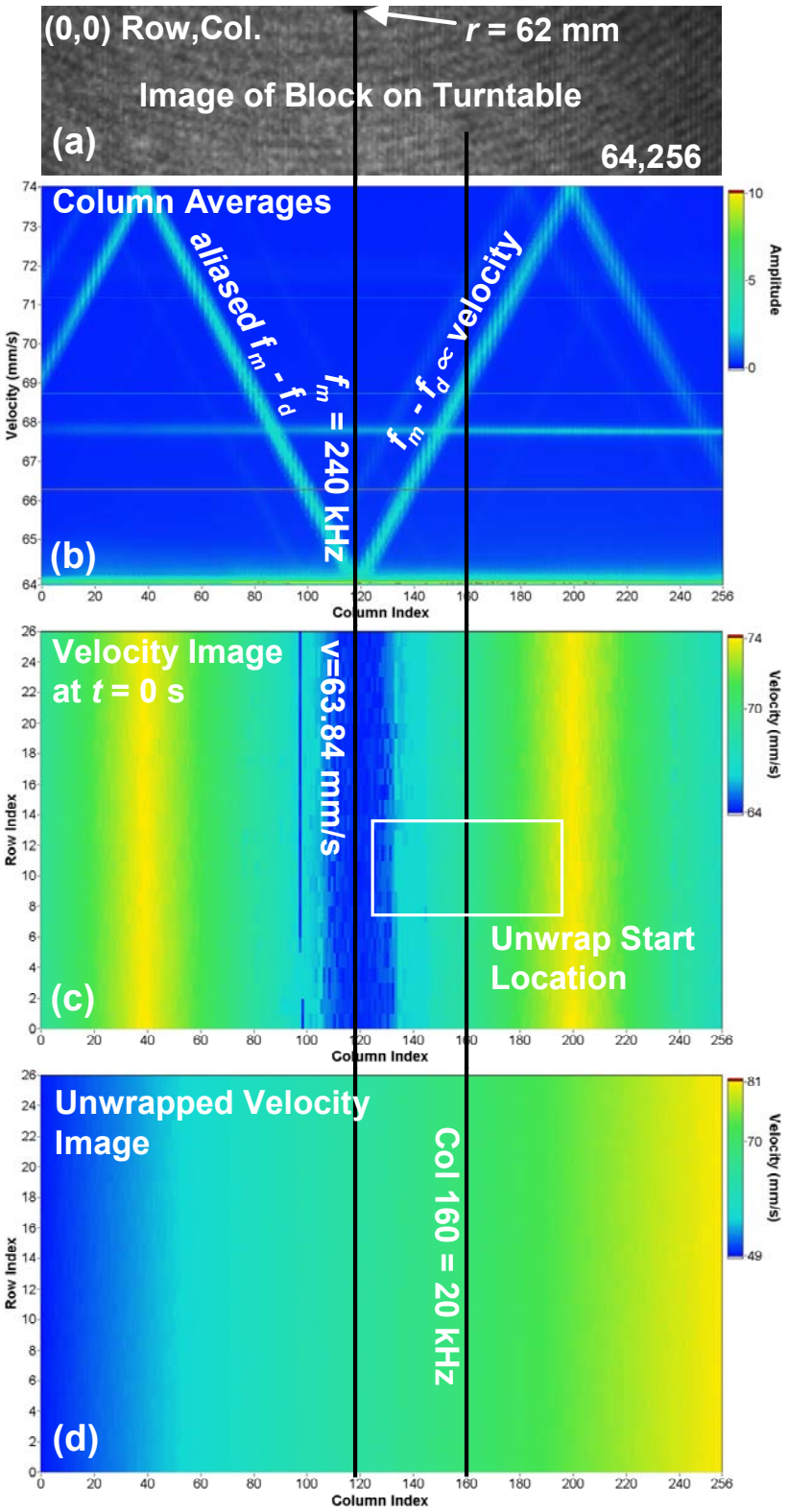

Figure 7. Results from the constant velocity turntable experiment. All columns line up vertically, and are at the same velocity. (a) Image of the block on the turntable, (b) column-averaged frequencies, (c) aliased single frame from the velocity movie, and (f) unwrapped velocity image.

Similar to the temporally aliased data of the pendulum, the spatially aliased data from the turntable can be unwrapped. By performing the JTFA on every pixel and then tracking the peak through time, a velocity matrix can be assembled for each pixel. The velocity array can then be assembled at any given time to yield a velocity image and can be animated to yield a velocity movie. A velocity image from the first frame of the movie is shown in Figure 7c. Again the data are spatially aliased. There are issues with tracking the velocity at points near DC in the JTFA because of noise in the system. Modifications are being pursued to mitigate this problem. Nevertheless, using the alias unwrapping concepts and techniques discussed in detail Section 3.3, the data can be unwrapped and spatially filtered to give a true velocity profile over the entire surface as shown in Figure 7d. While the intention was to have a constant turntable velocity, imperfections in the bearings or control system caused the turntable rotation rate to vary. This can be seen in the JTFA plot shown in Figure 8 of an individual pixel from column 160, where the color represents the magnitude of the JTFA in arbitrary units.

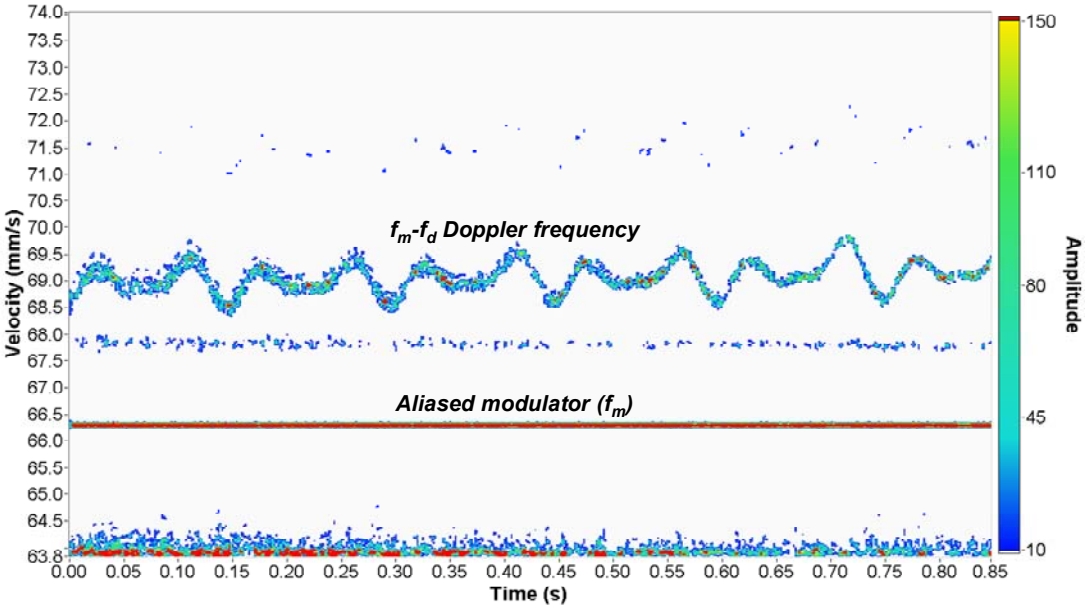

Figure 8. JTFA of a single pixel showing time variation of velocity.

2.4. Vibrating Film—Mode Shape Measurements

A sample of Polyvinylidene fluoride (PVDF) film approximately 100 mm square was supported in a fixture which did not constrain the boundaries, as shown in Figure 9. The available field of view is approximately 80-mm square. The bare film only returned sufficient optical power to image a vertical strip, 512 pixels by 32 pixels. Attaching a square of 3M retroreflective film allowed a full 512×512 pixel dataset. In both cases, the camera frame rate was 10 kHz.

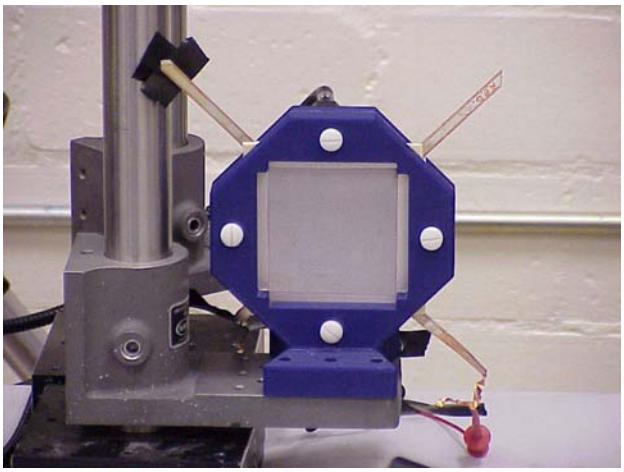

Figure 9. PVDF film in mount, with retro-reflective tape attached.

Electrically driving the film with a 100 Hz sine-wave provided a fast harmonic motion which challenged the capabilities of the WLDV system. Even at a frame rate of 10 kHz, there are very few camera samples per Doppler fringe, as seen in Figure 10a. This requires a very small JTFA window. Analyzing the temporal velocity response of a small set of pixels gives the result shown in Figure 10b, using a JTFA window of 16 samples. This clearly shows the harmonic velocity profile (rectified, since the WLDV responds only to velocity magnitude). The drive level was selected as the highest that did not cause the Doppler frequency to exceed the 5 kHz Nyquist limit for the 10 kHz frame rate.

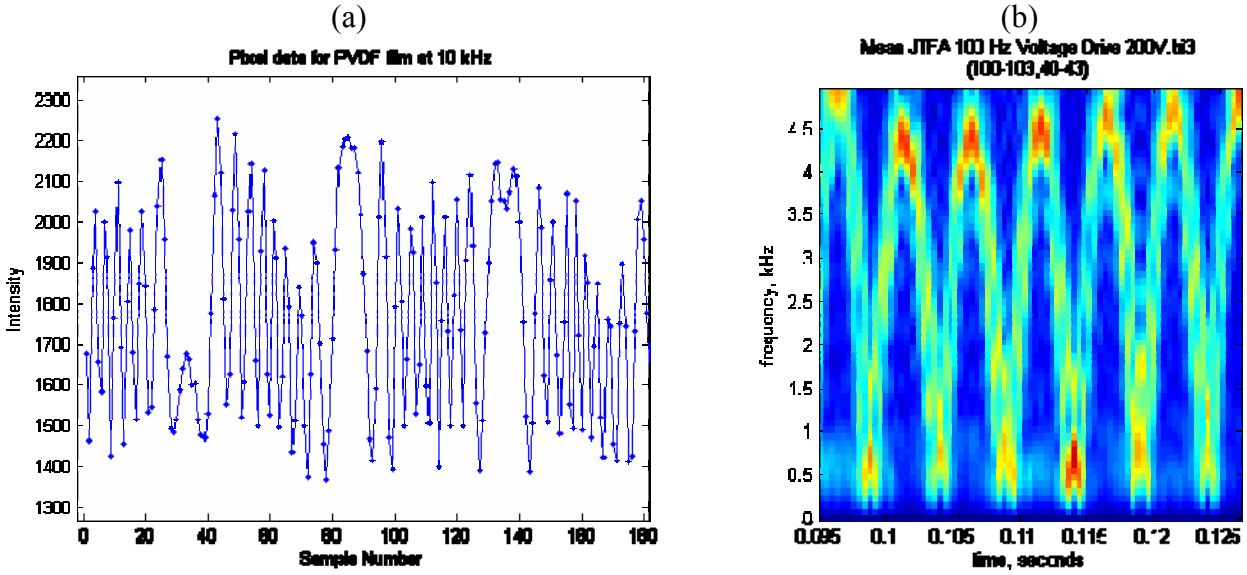

Figure 10. (a) Single pixel intensity response from PVDF film harmonically driven at 100 kHz. (b) JTFA of film driven at 100 Hz, 4 by 4 pixel average.

To demonstrate a measurement of a dynamic event, the film was driven with a slow square wave to generate a step response. Figure 11 shows a position-time plot of velocity for a single step of 100V. Note that Y Pos is vertical position in a strip centered on the sample. The response is

clearly quite complex—this information would be quite time consuming to obtain with a scanning single point LDV, and in fact impossible for that technique if the PVDF response varies, even slowly, with time.

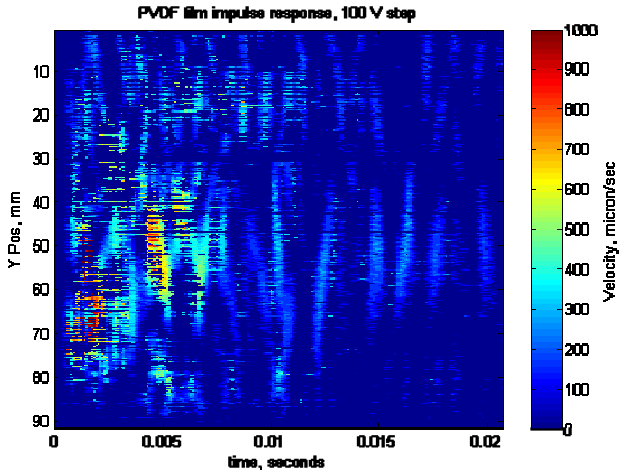

Figure 11. Y-position versus time Doppler frequency plot of PVDF film impulse response.

To analyze velocity shapes, the sample was first driven slowly (4 Hz sine wave). Then it was driven at what appeared to be a resonance at 104 Hz sine wave. Figure 12 and Figure 13 show the peak velocity at each pixel, regardless of phase. A velocity movie of the 104 Hz results shows that one set of peaks is about 180° out of phase relative to the other.

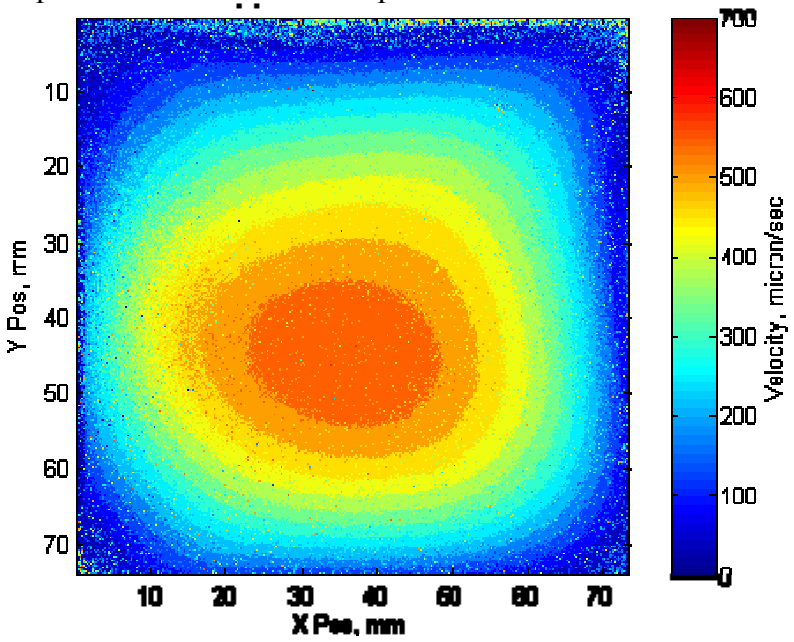

Figure 12. Peak velocity for PVDF driven quasi-statically.

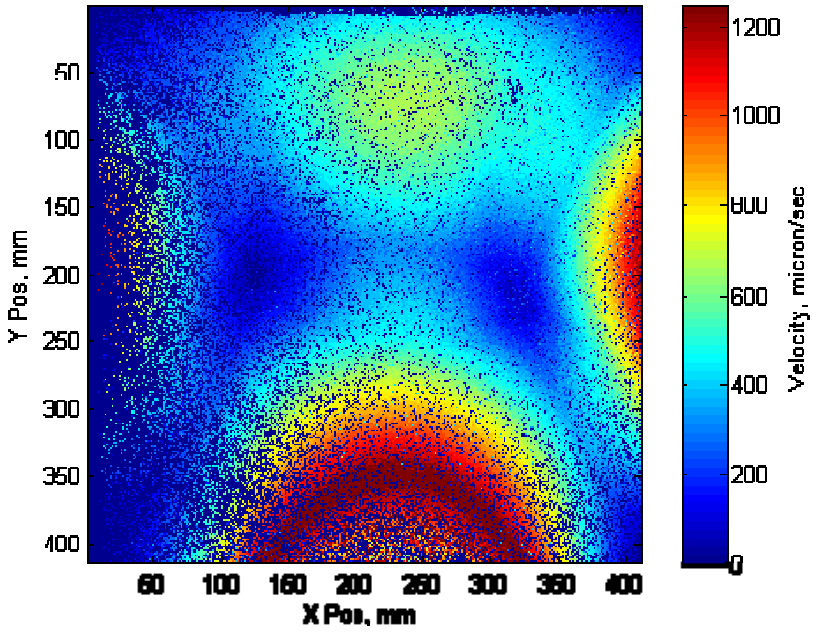

Figure 13. Peak velocity for PVDF driven at 104 Hz.

3. ANALYSIS TECHNIQUES

The details of the data analysis methods presented in Section 2 will be presented in this section. This expands and completes the overview sections contained in Section 1.1.3 and 1.1.4. The flow of this section will track that of the Doppler information from the surface of the object through to the velocity image or movie as outlined in Figure 14.

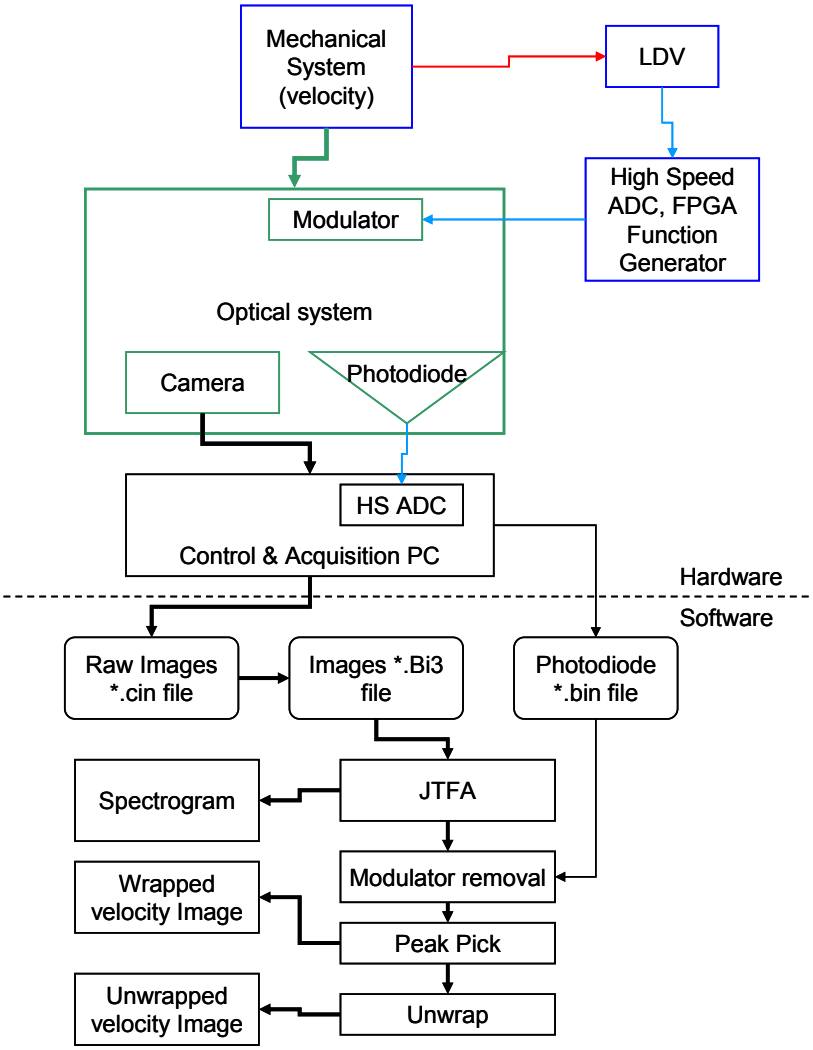

Figure 14. Data flowchart. Red and green lines are single beam and wide-field image optical data, respectively, blue lines are analog signals, and black are digital information.

3.1. Raw Data characteristics and display

The most basic data set for a WLDV experiment (without modulation) is a set of images taken by the high speed camera of the moving part. The raw data is written in Vision Research "CIN" format, and then converted via a LabVIEW program to an intermediate file readable by MATLAB ("BI2" or "BI3" format). The intermediate file also contains a header containing exposure time, camera frame rate, array size, etc. Each image is an array of pixel intensities with

the rows and columns representing spatial location on the object resulting in a "datacube" with dimensions X, Y, and time. At each pixel location, the irradiance as a function of time contains the Doppler frequency information described by Equation (12) and illustrated in Figure 4. The first step in the analysis is to determine the frequency content, as a function of time, at each pixel, by using a JTFA analysis as described in Section 1.1.3. If there are groups of pixels which, by *a priori* knowledge, we know should represent similar velocities, we can average the spectral magnitudes over these pixels to reduce noise. (Note that averaging the pixel intensities first, before frequency analysis, is *not* valid due to the random phases of the speckles.) An example of this is the pendulum experiment in Section 2.2, where all pixels in an image column should represent the same velocity.

As mentioned in Section 1.1.3, the JTFA we are using is a short term Fourier analysis. This analysis consists of taking the FFT of short subsets or "windows" of the data. The subsets must have sufficient samples to give the desired frequency (hence velocity) resolution, but be short enough that the Doppler frequency does not vary significantly during the window time. If the number of samples in a window is N_{WIND}, the camera frame rate is T_C, and the increment between windows is N_{INC} samples, then the time window for each FFT is $\Delta T = T_C N_{WIND}$ and the frequency resolution is $\Delta F = 1/\Delta T$. The JTFA time resolution, or time increment between FFTs, is

$$T_{JTFA} = N_{INC} T_C = \frac{N_{INC}}{N_{WIND}} \Delta T \tag{23}$$

Figure 15 shows a single column average FFT magnitude at one particular time from a pendulum dataset. This is the FFT for $N_{INC} = 256$ samples centered at sample number 28032, which occurs at $T_{JTFA} = 0.364$ s.

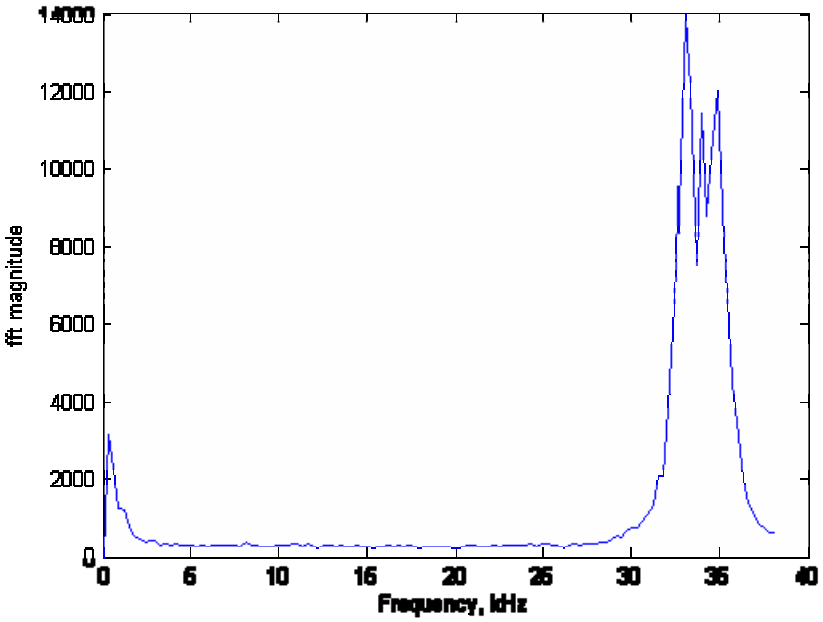

Figure 15. Spectral magnitude for Col 100, average over Row 1-21 at 0.364 s.

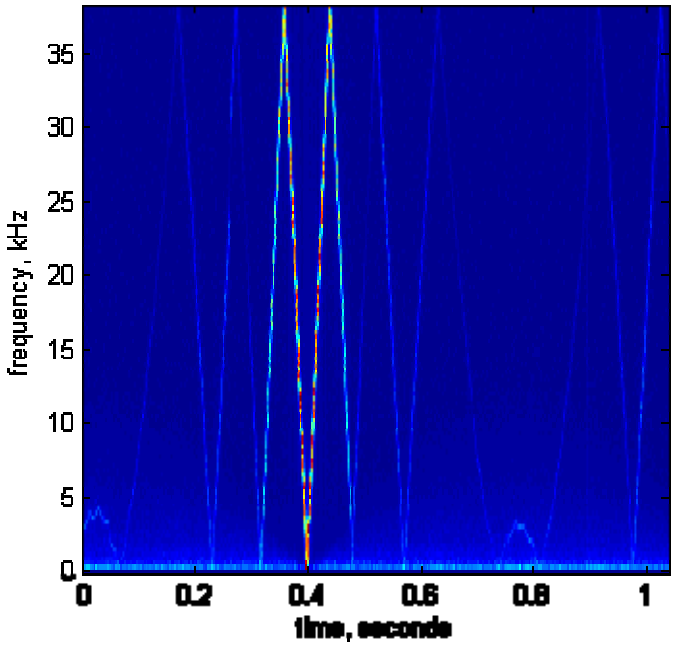

Figure 16. Typical spectrogram from unmodulated data (pendulum).

If we represent spectrum magnitude by color, and plot all of the FFTs in a JTFA as columns in an image, we get a "spectrogram", as shown in Figure 16. This is a time-frequency image; where the spectral peaks (indicated by brighter colors) trace out the Doppler frequency, hence velocity, function of this particular set of pixels as a function of time. The vertical axis on this particular spectrogram is Doppler frequency, due to the relation between Doppler frequency and velocity via Equation (14). Some examples in this section will have spectrograms plotted in frequency and some in velocity. The wavelength for the WLDV laser is 0.532 μm, therefore a frequency of 38.4 kHz corresponds to a velocity of 10.2 mm/s. A typical Nyquist frequency for the experiments presented is 38.4 kHz, dictated by the frame rate of the camera.

A slightly more complex dataset is one where the laser is modulated at a constant frequency. Some data from the turntable experiment (Section 2.3) illustrates this. The modulation frequency was 240 kHz, and the camera frame rate was 76,923 Hz. A typical spectrogram from such a dataset is shown in Figure 17. Here the horizontal line at 9.23 kHz represents the aliased constant modulator frequency, (|240000-6(76923/2)|= 9.23 kHz), while the wavy line between 25 and 30 kHz is the shifted Doppler signal ω_m-ω_d at this particular column. During the experiment, the turntable was decelerating slightly, as indicated by the downward trend, and had a "rumble" indicated by the oscillations.

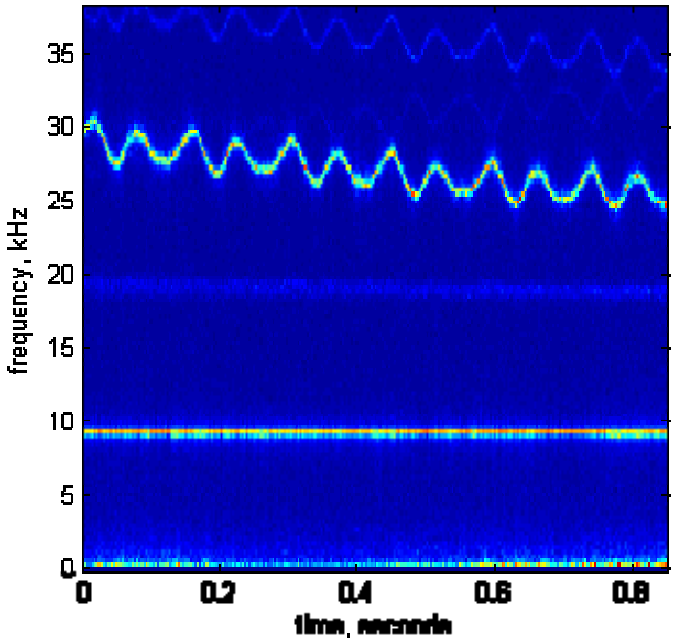

Figure 17. Spectrogram from constant modulator experiment (Col 128).

Another way to present the frequency (velocity) data is via the "zig-zag" plot shown in Figure 18. This is a presentation of the time average of all the FFTs at each column. The vertical (Doppler frequency) width of the line corresponds to the horizontal (time) integration of the data in Figure 17. The zig-zag line represents the time-averaged Doppler frequency of the rotating plate as velocity increases with column number. The modulation is selected to provide a zero Doppler frequency at column 83 for this example. Col 83-147 represent "true" down-modulated Doppler frequency ω_m-ω_d. Col 148-211 are the n=1 aliased frequencies, 21-82 the n = -1 aliased frequencies, and so forth. The "shadows" of the zig-zag line are other aliased signals as discussed in 4.1.

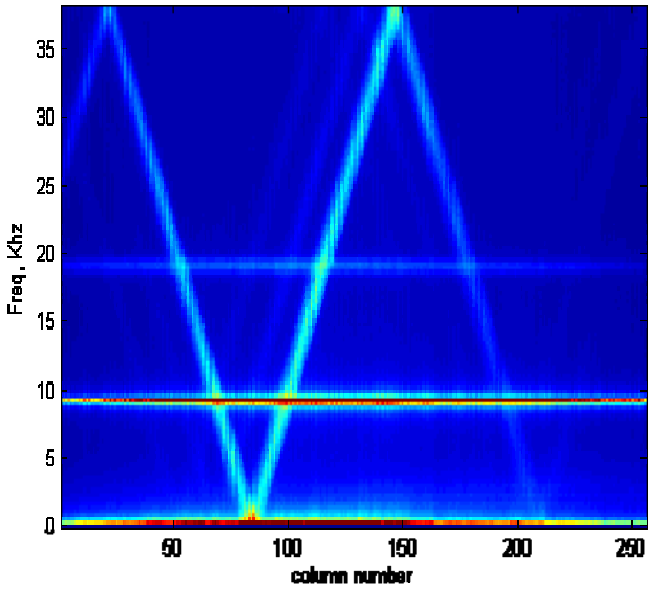

Figure 18. "Zig-Zag" time-averaged JTFA plot from constant modulator experiment.

The most complex WLDV data results from the "tracking modulator", or "time varying modulation" mode. In this mode, the modulator frequency ω_m varies with time to represent the mean or bulk velocity of a surface, while the WLDV data indicates variations from that mean over a wider area. In some cases, the modulator frequency function might be derived from *a priori* information such as the drive signal for a harmonic motion experiment. Alternatively, the mean velocity can be tracked using a conventional single-point LDV to detect the velocity at a point on the surface as depicted in Figure 19. The velocity signal from the LDV is used in real time to generate a modulator frequency which, ideally, shifts the Doppler frequency in the WLDV data to zero at that particular point. This is accomplished using a field-programmable gate array (FPGA) board manufactured by National Instruments. An FPGA board allows a user to "hardware" program a circuit for processing and control functions to be carried out deterministically and typically at high frequencies. This was used to create a cleaner and faster frequency modulation than was able to be obtained via an analog function generator. The FPGA board measured the velocity from the single point interferometer and then calculated a phase angle for output to a direct digital synthesis (DDS) board that would create a sinusoidal signal proportional to the measured velocity. Then this signal was used to control the EOM to modulate the Doppler signal down to DC, or to some user-specified offset. This avoids problems with highly aliased WLDV data and the associated alias unwrapping difficulties. To record the modulator signal we used a high bandwidth photodiode viewing an area-integrated optical signal, read by a high-speed digitizer to record instantaneous irradiance at a rate sufficient to capture the modulation frequency. We typically sampled the photodiode at a rate of 4 MHz, while our typical modulation frequencies were in the range of zero to several hundreds of kHz.

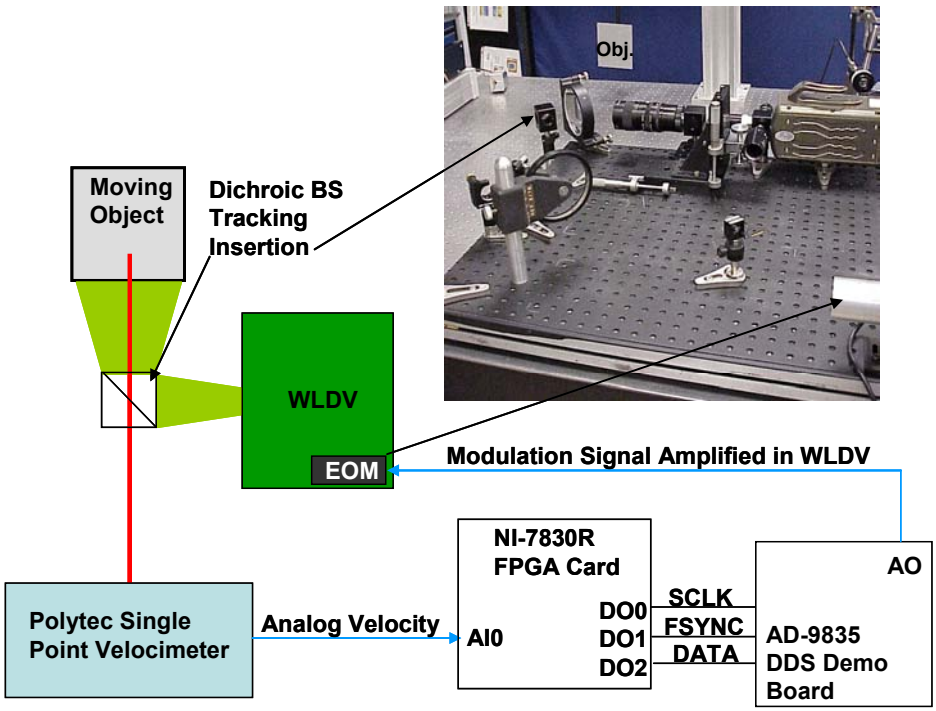

Figure 19. FPGA tracking modulator arrangement. Red and green lines are single beam and image optical data, respectively, blue lines are analog signals, and black are digital information.

A dataset from a WLDV experiment with time-varying modulation consists of two files: a set of images and a single time sequence of photodiode data. The image set contains the desired Doppler data, while the photodiode data contains the unaliased irradiance modulation signal, with typically little or no Doppler data due to spatial averaging. The modulator data is recorded in a binary (*.bin) format directly readable in MATLAB. Figure 20 is a spectrogram from a typical dataset with tracking modulator from a pendulum experiment. The bright zigzag line, centered around 0.35 seconds, is the modulator spectrum (some of its harmonics also appear as dimmer zigzag lines), which is aliased (wrapped) over 6 times in this particular case. The slowly varying data is the down-converted Doppler data, which appears as a hump. Figure 20 shows data from only one column, hence one Doppler trace. Figure 21 is an average over all of the columns, showing the envelope of the Doppler data as it varies within the image. At the maximum velocity time of 0.75 sec, there is a maximum variation in velocity across the image. In this particular dataset, there is about a 20 kHz offset in the modulator frequency, so that at the zero velocity "turn-around" time (0.4 sec), all columns have the same (ω_m-ω_d) Doppler frequency of 20 kHz, as they should, since the mechanical velocity of the pendulum at this time is zero. In some cases, we introduce an intentional offset to bring the zero velocity Doppler information out of the low frequency noise band. The offset is not a problem since the actual modulator frequency is recorded—the purpose of tracking the modulator frequency at all is to reduce the aliasing of the camera data, which is evident in this example. The down-shifted relative velocity data only "wraps" once, as it goes through zero Hz. Figure 22 is a spectrogram of the photodiode data obtained by a JTFA similar to that applied to the camera data, which shows the unaliased modulation frequency of the laser, as explained in more detail in Section 3.2

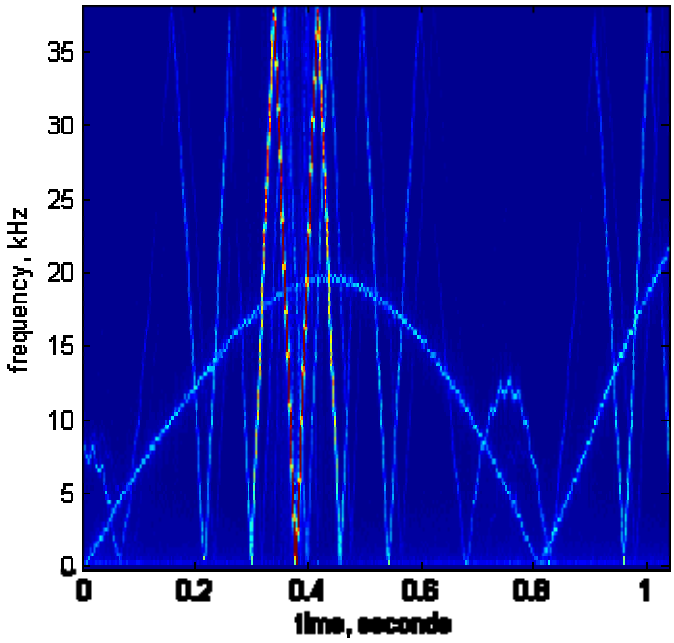

Figure 20. Single column spectrogram from pendulum with tracking modulator.

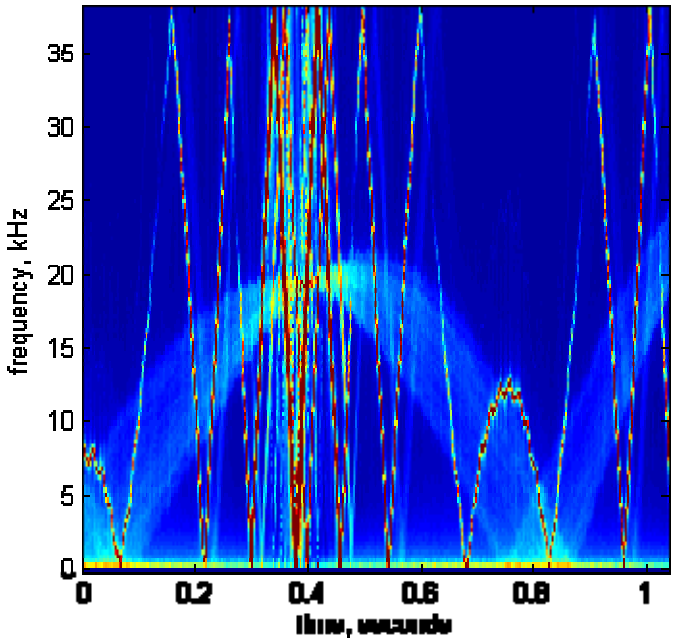

Figure 21. Average spectrogram of all columns from pendulum with tracking modulator.

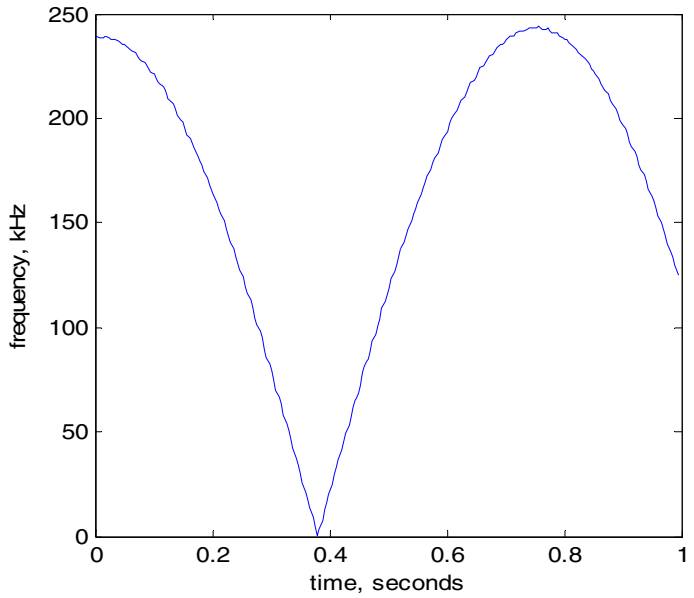

Figure 22. Full-frequency spectrogram of photodiode data.

3.2. Frequency peak identification

The next data analysis step is identifying the Doppler frequency ω_d from the spectral data. For unmodulated data, this is simply a peak-pick operation (possibly with some filtering). For instance, for the data in Figure 21, a peak-pick indicates that the frequency at column 100, at 0.364 sec is 33 kHz, while some smoothing of the FFT magnitude might result in the selection 34 kHz.

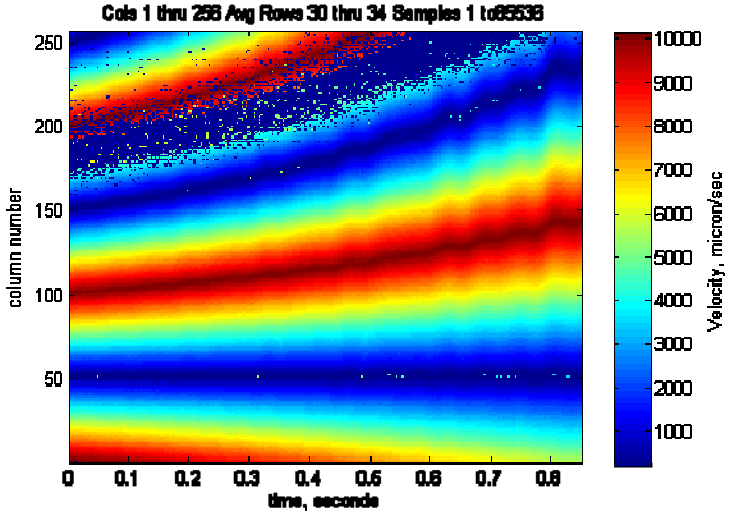

Figure 23. Turntable velocity from unmodulated WLDV data via simple peak-pick.

Figure 23 is a colormap of frequency peaks for an unmodulated turntable experiment using a simple peak-pick on each of the FFTs in the JTFA. The noisy data displayed in the upper part of the figure is unexplained—possibly due to a bad spot on the turntable block.

When the modulator is used, whether it is constant or "tracked" frequency, the modulation frequency is contained in all of the pixels and tends to dominate the frequency information.

Typically, the spectral component at the modulation frequency is larger than that at the desired data frequency. Simply searching for the largest component would find the modulation frequency. If the modulation frequency is constant, the peak-pick algorithm can simply be instructed to ignore that known constant frequency. Peak identification of data with time-varying modulation is more difficult. Using some method, the peak representing the modulator frequency must be removed from the data before attempting to identify ω_d. Our first attempt with time-varying modulation was to estimate the wrapped modulator spectrum by simply averaging all the spectrograms over space. This was not particularly successful for two reasons. First, the mean irradiance, or signal power, at any particular pixel is modulated by the speckle pattern of the object, and by noise in the reference beam (which was particularly noisy since we did not use any spatial filtering in the reference beam). So to successfully remove the modulator spectrum by subtracting an average spectrogram, the spectra would have to be normalized on a pixel-by-pixel basis. Figure 24 illustrates this—it is an image formed from the magnitude of the spectral peak of a modulated dataset. It looks very similar to the speckle image of the same dataset. Another problem with subtracting a mean spectrogram is that it would also remove any Doppler information that happened to be constant over the surface.

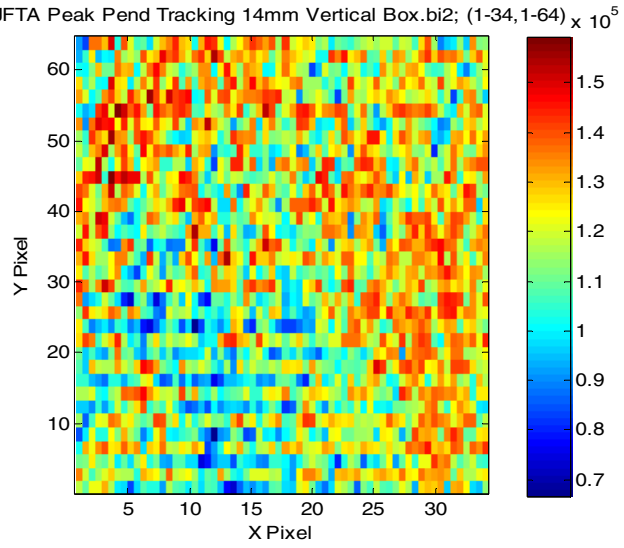

Figure 24. Image of spectral peak magnitude.

A better way to remove the modulator signal is to use the data from the photodiode. However, since it is sampled at a sufficiently high frequency to avoid any aliasing, it must first be wrapped or aliased to match the sampling frequency of the image data. There are two ways to do this: the "wrapped spectrogram" method, and the "resampling" method. In the "wrapped spectrogram" method, we take the full-frequency photodiode spectrogram as shown in Figure 7 and wrap it by applying what amounts to the inverse of Equation (17).

Define $P(t)$ as the photodiode frequency function obtained by performing a JTFA on the photodiode time signal. If $F_C = 1/T_C$ is the camera frame rate, then $F_N = F_C/2$ is the Nyquist

rate at which the frequency function should wrap back toward zero. Given two intermediate modulo functions $t_1 = \mathrm{mod}(P, F_C)$ and $t_2 = \mathrm{mod}(P, F_N)$, the wrapped frequency function is

$$P_W = t_2 - 2\frac{t_2 - t_1}{F_N} t_1 . \tag{24}$$

Figure 25 is an example plot of an unwrapped function and its wrapped version, selected in time and amplitude to be similar to Figure 5 with a "wrap velocity" of 10. Figure 26 is a plot of t_1 and t_2, from which it is obvious that $(t_2 - t_1)/F_N$ is a binary function that is zero on the "upward wrap" and one on the "downward wrap". It allows us to subtract twice t_1 from t_2 at the proper times to form the "downward wrap", while t_2 alone gives us the "upward wrap". (Note that the wrapped frequency follows a continuous, triangle path as the independent or "base" frequency increases, not a *sawtooth* path as that which occurs in typical phase wrapping in interference experiments that we have used as an analogy for alias unwrapping.)

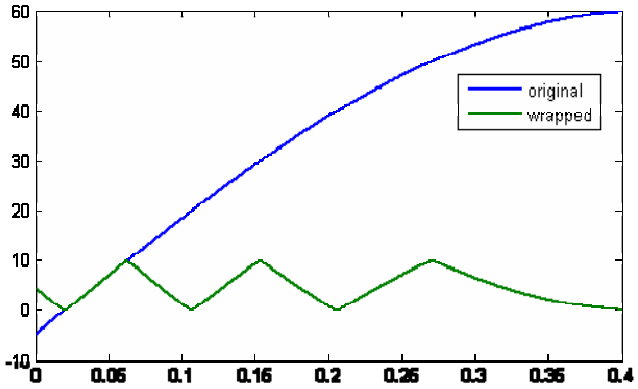

Figure 25. Original and wrapped functions by equation (1).

The "resampling" method of deriving the wrapped modulator spectrum from the photodiode data is to resample the photodiode time series data at the camera rate, then take a JTFA of the resampled data with the same parameters as those used for the camera data. Figure 27 is a spectrogram of a tracked turntable experiment, with the turntable decelerating. The data are dominated by the modulator spectrum. Figure 28 is an overlay of the wrapped photodiode spectra created by both techniques. Note that the frequency resolution of the photodiode spectrum must be similar to that of the camera data for a proper match. Since the photodiode data is sampled at a much higher frequency (4 MHz) than the camera data (76 kHz), the JTFA window must be proportionally larger in sample numbers for the wrapped spectrum technique. In the above example, the JTFA window for the camera data was 256 samples, while that for the photodiode JTFA was 16384 samples, which is the next power of 2 above the proportional size of $(4/0.076)256 = 13474$. Our current implementation uses the resampling technique.

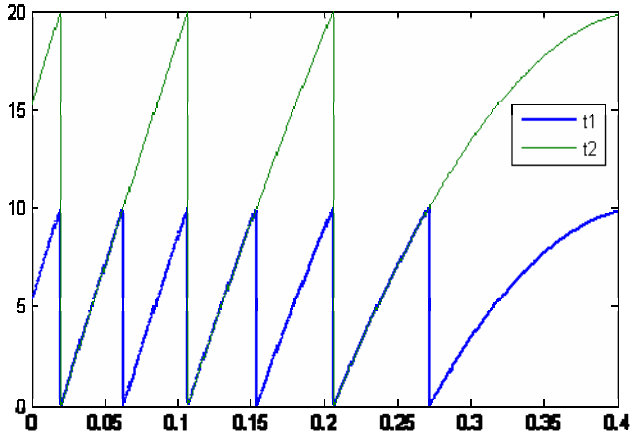

Figure 26. Intermediate modular functions t1 and t2.

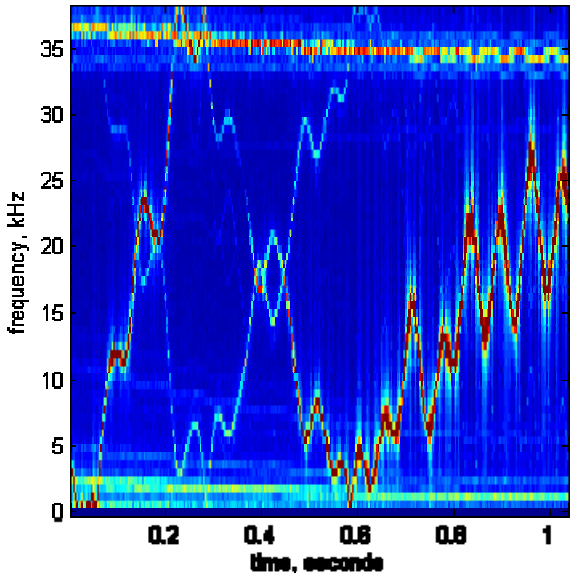

Figure 27. Tracked turntable spectrogram.

Another problem that must be solved before subtracting the photodiode-derived modulator spectrum from the camera derived WLDV data is temporal alignment of the two signals. Despite efforts to trigger the camera and photodiode digitizer simultaneously, some misalignment was always present. Figure 27 and Figure 28 show a rather large misalignment. This problem was solved by correlating the resampled photodiode signal with one of the pixels of camera data. The location of the correlation peak shows the offset between the two signals. From this, the camera data and/or photodiode data are aligned and truncated appropriately. Figure 29 shows a correlation signal of some typical data over the entire time window, and expanded near the correlation peak.

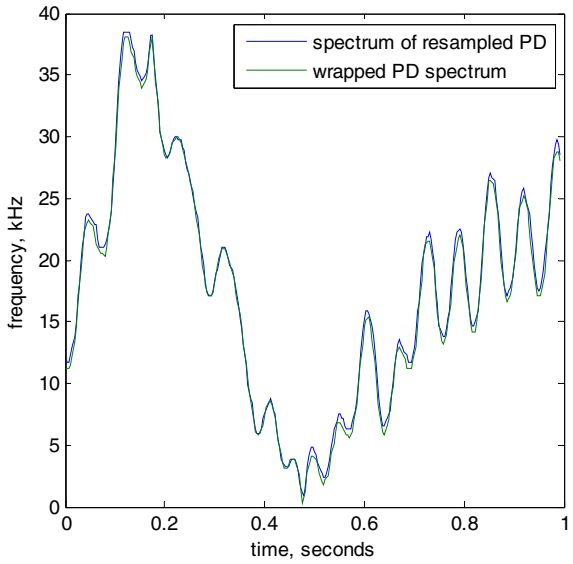

Figure 28. Modulator spectra derived by wrapping and by resampling.

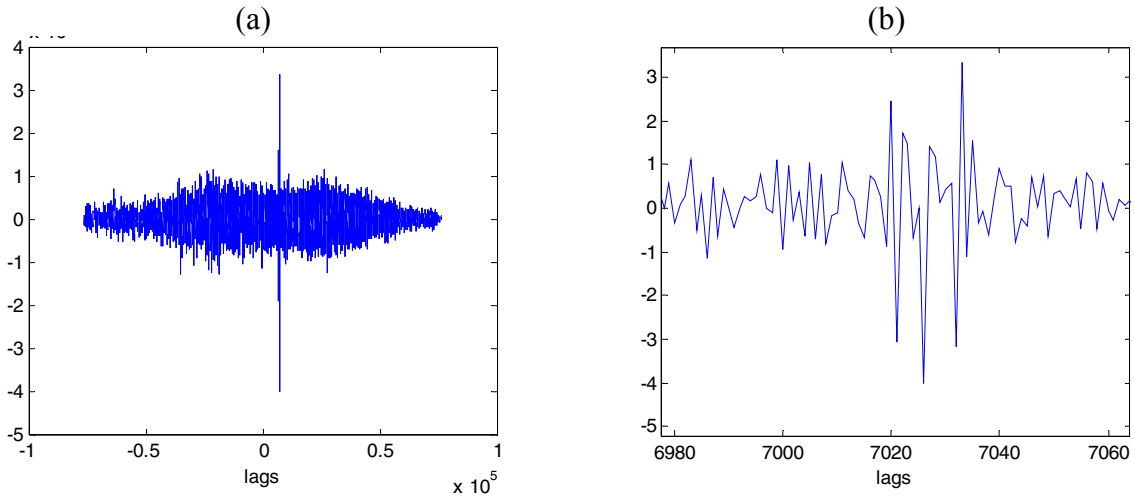

Figure 29. (a) Correlation peak for an entire time window, and (b) expanded near the peak showing the multiple peak nature. Alignment is chosen at the highest peak magnitude.

Another question that can be answered by correlating these two signals is whether there is any "slip," or sample timing error between the camera and photodiode signals. This is unlikely, since both the camera frame time and sample digitizer times are generated by highly stable signal generators. By correlating short windows at various times within the signal, we see that, within noise limits, the delay is in fact constant, as shown in Figure 30. Note that the modulation is the dominant frequency component in both camera and modulator signals, and is represented by several peaks in the correlation signal. The goal here is to correlate variations in the velocity data, which varies much more slowly than the modulation. Hence we simply pick the time of the largest (absolute) peak to indicate the signal delay. This process is accurate to a few cycles of the modulator frequency which, for typical data bandwidths, represents 1/100 to 1/1000 of a cycle of the velocity data. We have not attempted to use tracking modulation techniques on higher frequency velocity signals (hundreds of Hz) like that in the PVDF film experiment.

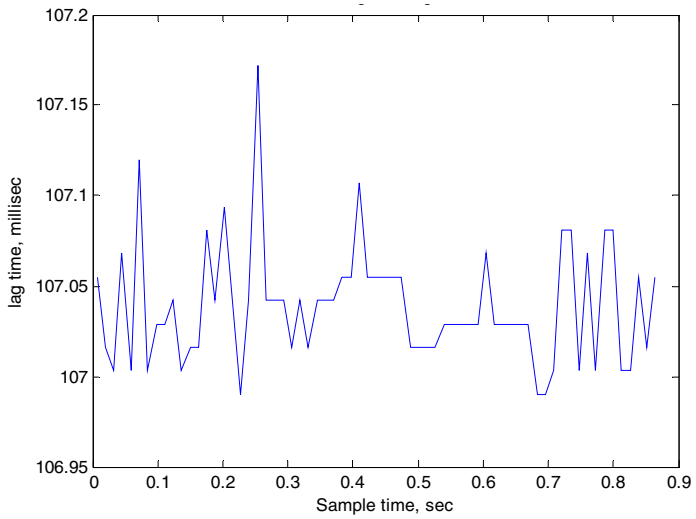

Figure 30. Sliding window correlation between photodiode and camera data.

Figure 31 shows the data from Figure 27 with the modulator spectrum removed from the JTFA by simply zeroing frequency components at or near the local modulator frequency.
Figure 32 shows the result of applying a simple peak-pick to the spectrogram of this data after removing the modulator. Points where the FFT peak is not above a pre-selected threshold are identified by the not-a-number (NaN) flag in MATLAB, and presented as zero (dark blue) in Figure 32. While significant work remains on optimizing the peak picking process and selecting analysis parameters, the frequency (hence velocity) surface is clearly present. In this dataset, the LDV velocity correction point was near column 130. The turntable was decelerating significantly, as evidenced by the wrapped modulator spectrum in Figure 31. Figure 32 shows the (wrapped) velocity deviation, plus an offset, from the LDV point. Figure 33 again shows the wrapped velocity deviation, but in a 3D format rather than the color format of the same data in Figure 32.

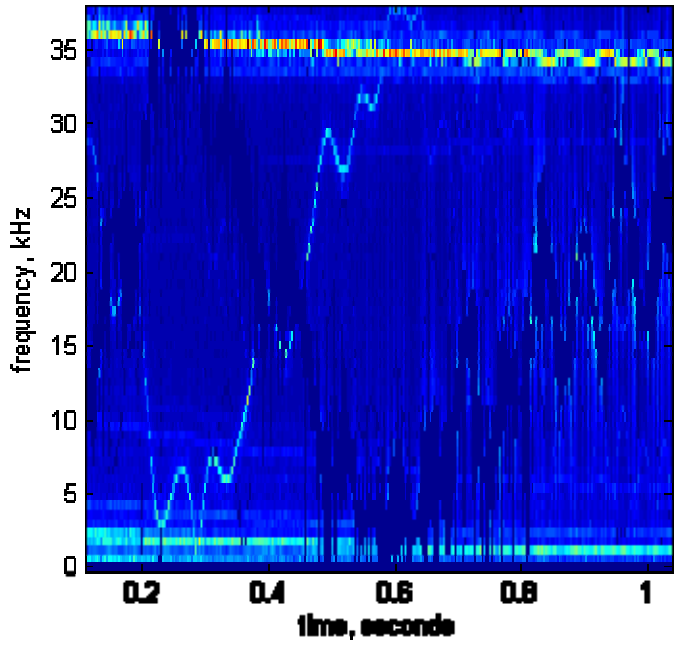

Figure 31. Turntable spectrogram with data at modulator frequencies set to zero.

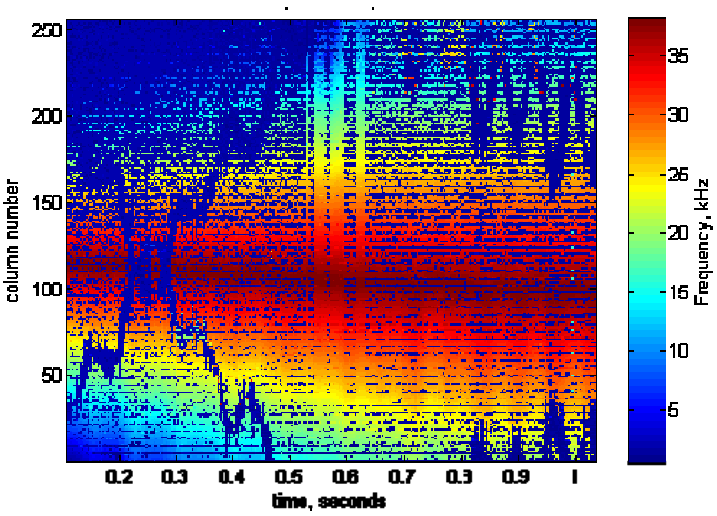

Figure 32. Frequency surface for modulated turntable experiment.

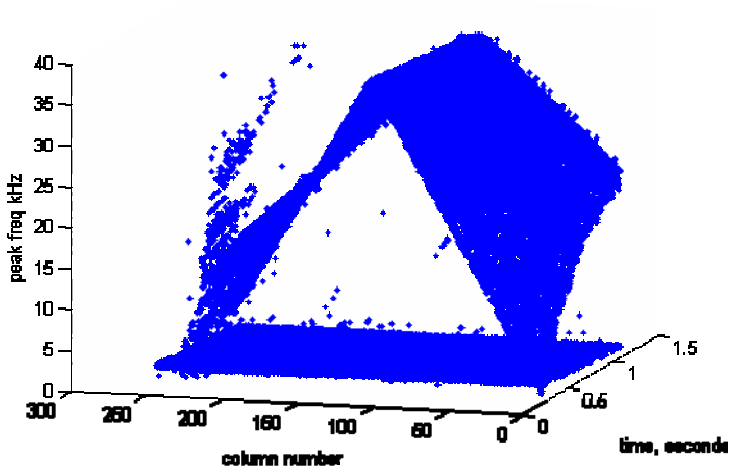

Figure 33. 3D Dot plot of the same frequency surface as in Figure 32

3.3. Unwrapping

3.3.1. Unwrapping regions

The algorithm used to unwrap a single plane of aliased data relies on the continuity of the velocity data. An array of data can be either only spatial (row and column) as in Figure 7 or space-time data. In either case, we refer to the *xy*-plane as the coordinates over which the wrapped velocity data are defined. The unwrapping algorithm is based on identifying "wrap regions," which are *xy*-regions within which the aliasing order (n in Equations (17)-(18)) is constant. A subregion of the plane is selected (by the operator in this implementation—automated techniques are under development) for which the velocity is not wrapped. Figure 7c shows the wrapped velocity data from the turntable, with the initial unwrap start location selected. A simple fitting function (a plane in this case—a quadratic can also be used) is fit to the velocity data within the analysis area. A new subregion is selected that overlaps the first subregion, and the fitting function is evaluated in the new subregion as indicated in the inset to Figure 5. Any portion of the new subregion for which the fitting function extends beyond a wrap boundary is identified and assigned the next region number. Naturally, it is necessary to identify whether the velocity is increasing or decreasing and assign the new region number appropriately. The actual data is then unwrapped in the new subregion. This process continues, extending the unwrapped data subregion by subregion, until all the data are unwrapped.

All subregions must be selected to overlap the preceding subregion. In this case, this was done by a pair of simple boustrophedonic scans (i.e. back and forth), beginning at the operator-selected area and proceeding first up/right, then down/left as indicated in Figure 38, which also shows the aliasing regions. As the wrap regions are identified, the frequency data are unwrapped via Equation (18), where n is the region number (aliasing order) and χ_n is chosen via Equation (19). The process repeats, each time fitting an already unwrapped subregion and then extending the fitting function into the next subregion. Figure 7d shows the result for this particular example.

3.3.2. Initial plane selection

Currently the initial unwrapping plane is selected by the operator. A figure depicting the wrapped frequency peaks is displayed, and the operator selects a rectangular subregion which

presumably lies within the unaliased wrap region. We anticipate automated initial plane selection based on *a priori* knowledge, such as the location of the LDV measurement beam spot. In any case, there is a possibility that the selected subregion might extend a bit into the next region. (Current routines account for such an overlap on only *one* boundary.) An iterative technique is used to select a proper fitting surface. In this example, a plane is used as the fitting surface, but a higher order polynomial would work as well. First, the surface is fit within the selected region. A preliminary wrap boundary is selected based on the initial surface, and data points are mirrored based on this initial boundary, as shown in red in Figure 34a. Using the newly wrapped points, a new surface is fit as shown in Figure 34b. Surfaces are generated above and below the initial surface at a threshold distance, as shown in Figure 34b. Data points falling within the threshold distance are used for the next step. Finally, a plane is fit to the data points within the threshold illustrated in Figure 35.

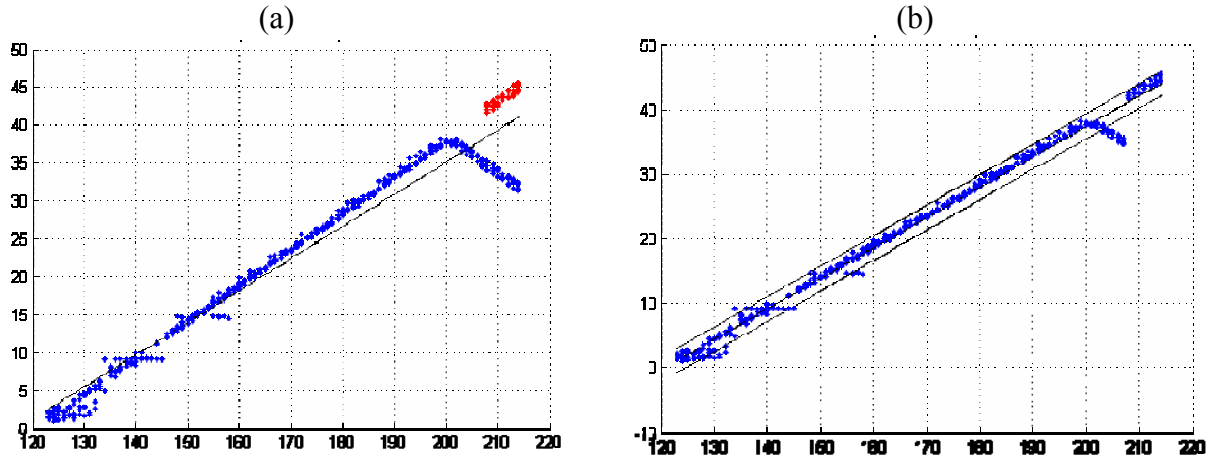

Figure 34. (a) Initial fitting plane with threshold based acceptance region. (b) Second iteration fitting plane.

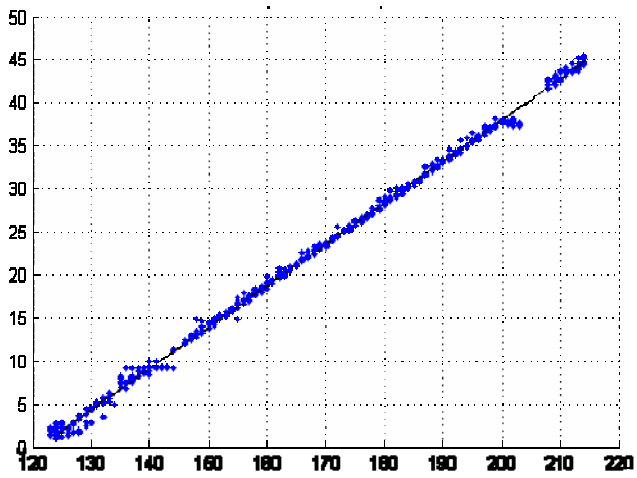

Figure 35. Final fit for initial plane.

During unwrapping, the result can be interpolated by assigning any unidentified points the value of the fitted plane. Figure 36 shows the unwrapped, non-interpolated version of the data in Figure 32, while Figure 37 shows the interpolated result. The unwrapping paths for the above example are shown in Figure 38.

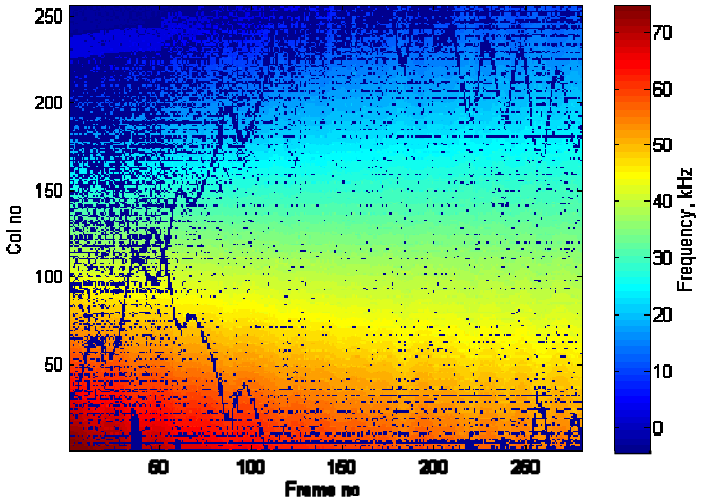

Figure 36. Unwrapped data from decelerating turntable experiment, no interpolation.

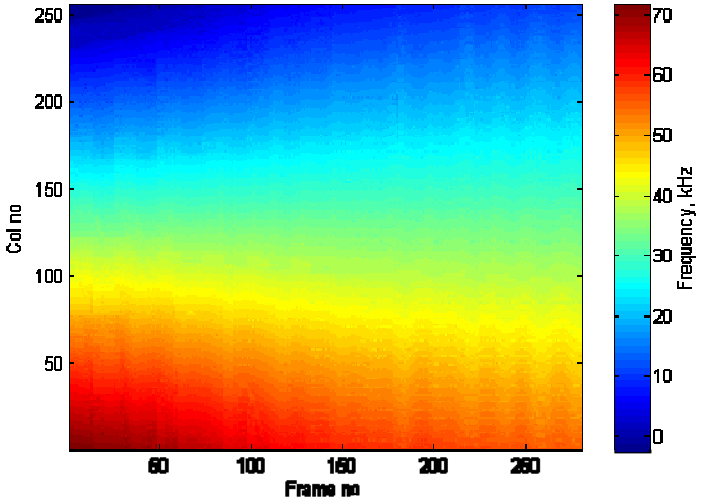

Figure 37. Unwrapped data from decelerating turntable experiment, with interpolation.

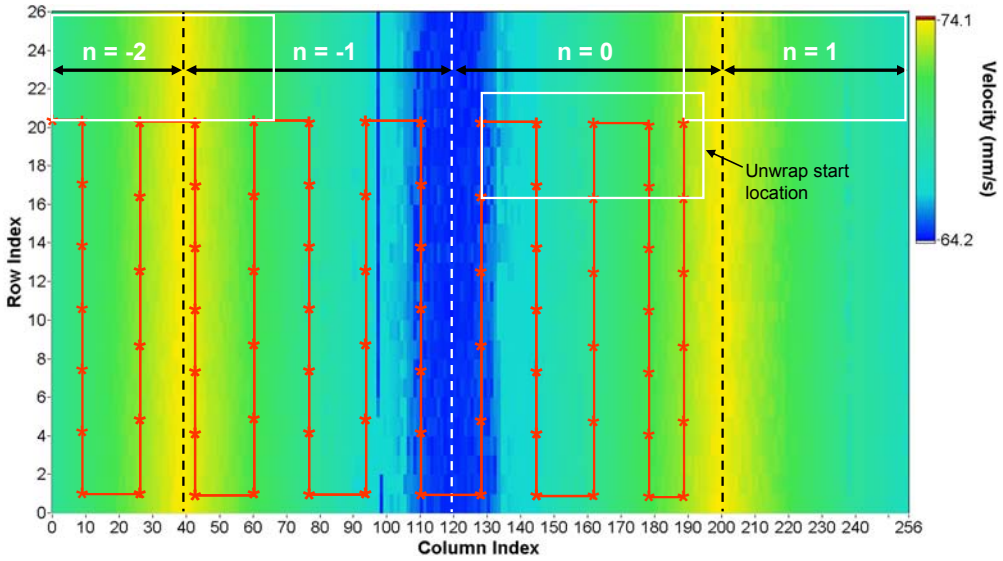

Figure 38. Boustrophedonic scan path is shown with asterisks indicating each step used to unwrap the aliased data. Boxes represent unwrap start and final locations. Unwrap regions are indicated by the integer value n.

4. SIMULATIONS

4.1. Aliasing Simulations

To understand the multiple signals seen in the JTFA results, it is important to determine their source. This is clearly illustrated in Figure 39, where the frequency tracks (FFT peaks) are labeled. Enlightenment on the source of each peak was important in determining how to minimize the data confusion, utilize the extra information, or ignore it. As discussed previously and in the appendix, anti-aliasing for the optical domain is not possible; therefore working with the aliased data was critical.

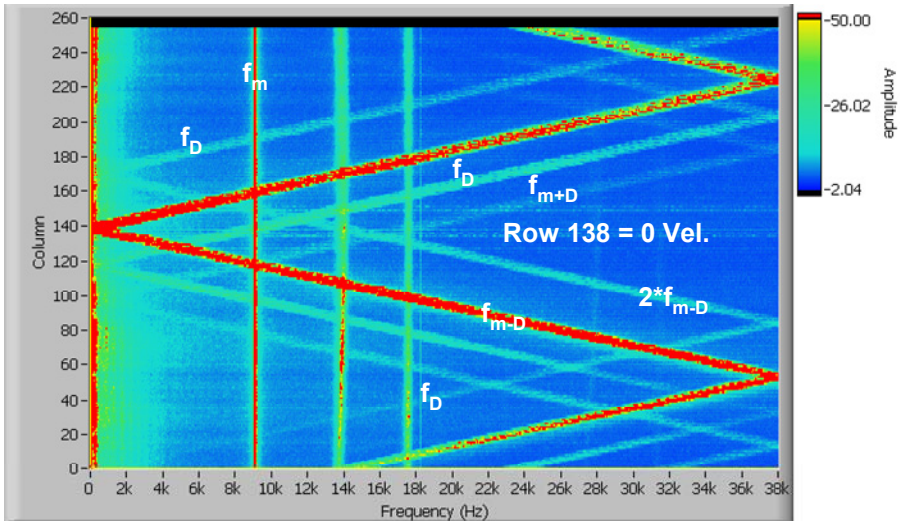

Figure 39. JTFA zig-zag plot showing a number of aliased signals.

For the following examples, we have simulated the time varying irradiance on a WLDV pixel for different Doppler frequencies. To accomplish this, we simply generated the modulated Doppler signal, based on Equation (12), then boxcar integrated to simulate the camera integration time. As a simple example to visualize how the equation works, Figure 40 shows the Doppler, the modulator, and the sum and difference frequencies, all unaliased. This simulates the turntable experiment with a modulator frequency of 4 kHz, near the center of the table, that is, where the Doppler frequency is low, 5.2 kHz in this case.

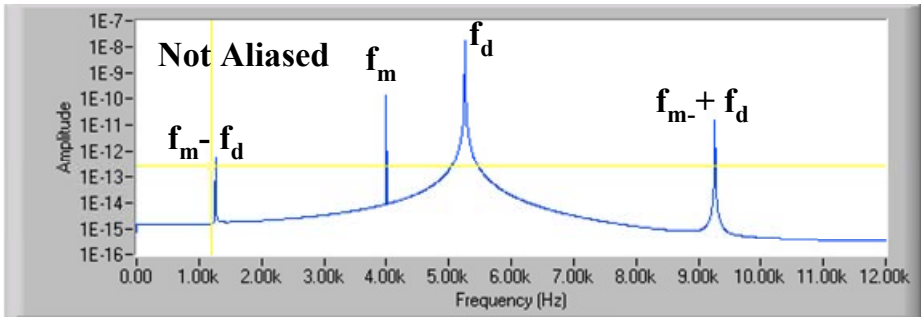

Figure 40. Simulated unaliased data for the WLDV turntable experiment.

To see what happens when a location with Doppler frequencies greater than half the sampling rate is encountered, the same equation was simulated further out on the turntable. The results are shown in Figure 41.

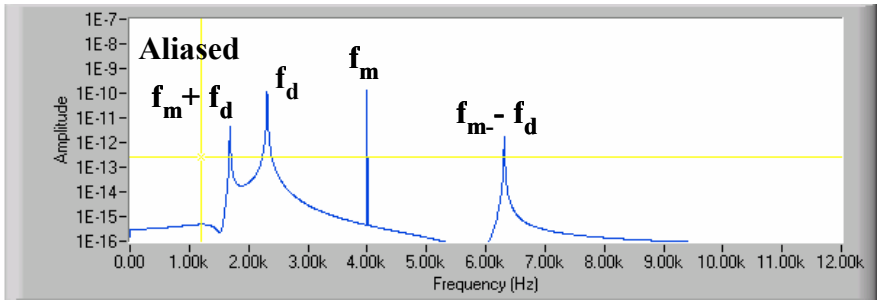

Figure 41. Simulated aliased data for the WLDV turntable experiment.

Using the knowledge gained from the previous simulations, and using Equation (17), it is possible to label the peaks in real data. Figure 39 shows data in a zig-zag plot from the turntable experiment. A slice from column 113 from this zig-zag plot is shown in Figure 42 with some of the peaks labeled with aliased frequency and their aliasing number *n* clarifying the experimental results, the unlabeled peaks are currently unidentified.

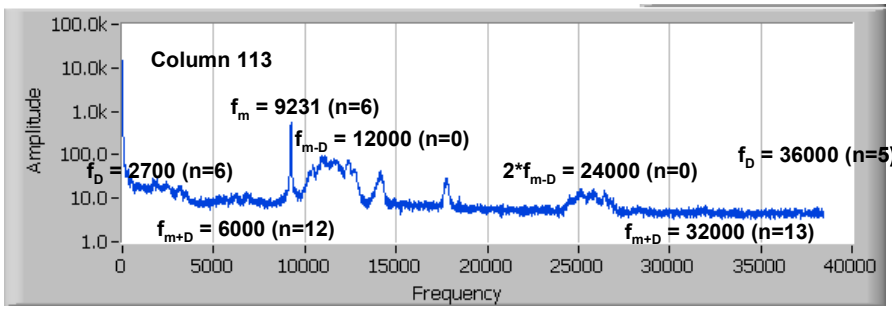

Figure 42. Aliased data from column 113 shown in zig-zag plot of Figure 39.

4.2. Multiple speckles per pixel simulation

Next we study the effect of multiple speckles within a camera pixel. For an object with constant velocity and no modulation of the illuminating light, each speckle is represented mathematically by a Doppler term containing a Doppler frequency with a random amplitude and phase. Therefore, the total irradiance in the pixel has the form

$$\begin{aligned} I(t) &= \sum_{k=1}^{N} \{1 + \mu_k \cos(\omega_d t + \phi_k)\} \\ &= A + B\cos(\omega_d t + \phi), \end{aligned} \tag{25}$$

where μ_k and ϕ_k are the random normalized irradiance and phase, respectively, of the k^{th} speckle and N is the total number of speckles. The constants A, B, and ϕ are the resultant amplitudes and phase of the sum, respectively. Given the identity in Equation (25), the resultant values are

$$A = N,$$
$$B = \sqrt{\left\{\sum_{k=1}^{N} \mu_k \cos(\phi_k)\right\}^2 + \left\{\sum_{k=1}^{N} \mu_k \sin(\phi_k)\right\}^2},$$
and
$$\tan(\phi) = \frac{\sum_{k=1}^{N} \mu_k \sin(\phi_k)}{\sum_{k=1}^{N} \mu_k \cos(\phi_k)}. \quad (26)$$

Identifying "harmonic content" with coefficient B, and contrast with the ratio B/A, we performed the sums in Equation (26) for one to 10000 speckles with μ_k uniformly distributed on [0,1] and ϕ_k uniformly distributed on [-π,π]. The average of 1000 sums for each number of speckles is plotted in Figure 43 and some sample values are listed in Table 1. The harmonic content B increases approximately as $N^{1/2}$ and correspondingly, the contrast decreases as $N^{-1/2}$. Often, attention is given to the fact that as the number of speckles increases, the contrast *decreases*. Nevertheless, since we are processing the irradiance waveform digitally, we can remove the DC component. Ultimately, our signal consists only of the harmonic component whose amplitude *increases* along with the DC component, as discussed in Section 1.1.5, until either the camera pixel or the AD converter saturate. It is observed in Figure 43 that the scatter in the harmonic component increases with the number of speckles in a pixel. In fact, the variance is proportional to N.

Table 1: Results of sums random speckles on harmonic content and contrast.

N	1	10	100	1000	10000
Harmonic Content	0.494	1.615	5.192	16.225	50.964
Contrast	0.494	0.162	0.052	0.016	0.005

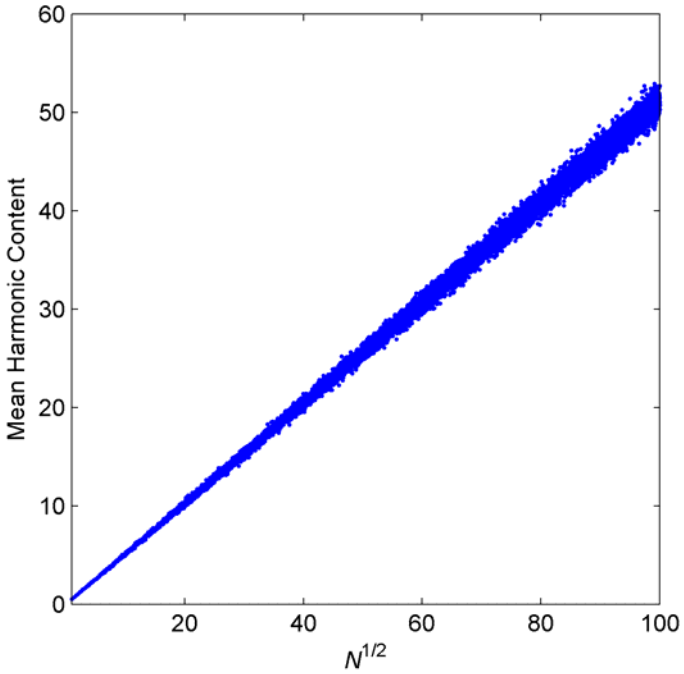

Figure 43. Mean harmonic content values for 1000 sums of random speckles.

Using the irradiance formulation in Equation (25) with a linearly increasing Doppler frequency (increasing velocity) and 2800 speckles, the irradiance from the speckles on a detector was simulated and the DC component was removed. This simulation corresponds to a turntable experiment where a velocity gradient from 76 mm/s to 82 mm/s was measured across the detector. The resulting irradiance variation is plotted in Figure 44 along with the experimental data. Key features of the simulation results are similar to those of the experimental results, including the "Doppler bursts." In this figure, these are not actually bursts, but result from the addition of quasi-monochromatic light, which results in the amplitude modulated waveform that is seen in Figure 44. The quasi-monochromatic nature comes from the velocity spread of the speckles on the detector each with a different Doppler shift, or in other words a frequency shift of the light.

It is noted that the simulation based on Equation (25) deals only with assumed speckle intensities and neglects simulation of the optical system. To simulate the propagation of random amplitude through a simple imaging system and to allow simulation of the interference process, we use an approach based on Fourier optics. Fourier optics hinges upon the fact that the physical phenomenon of light passing through a lens can be represented mathematically by Fourier transforms. Details of Fourier optics can be found in Goodman[23].

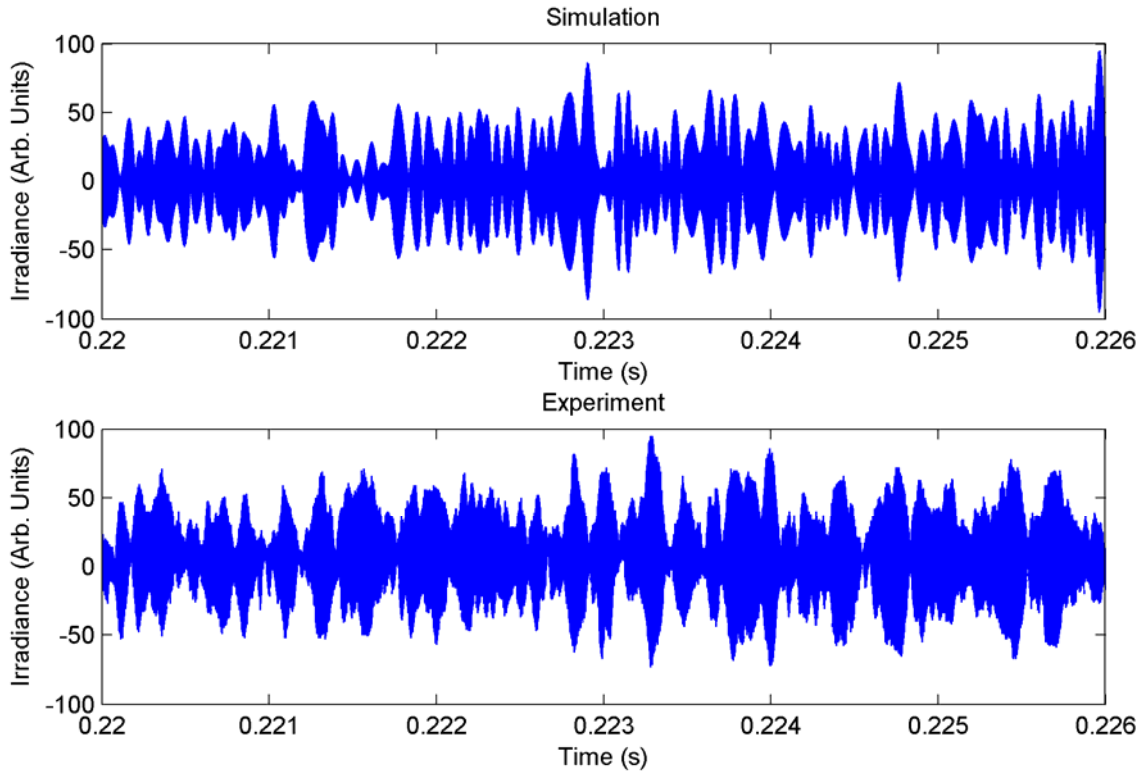

Figure 44. Simulation and experimental measurement of irradiance from 2800 speckles on a detector.

In order to simulate the time varying irradiance at the camera, we modeled a single camera pixel with an array of subpixels and constructed our rough scattering object as an array of constant amplitude, random phase scatterers. First, we take the phase array and add the Doppler phase component representative of the prescribed, out-of-plane object velocity. Second, we compute the object wave field using the object amplitude, summed phase, and an apodization filter. This apodization illustrated in Figure 45a eliminates the "leakage" caused by the implied periodicity of the FFT. Third, we take the Fourier transform of the object field and pass it through a circular pupil filter representing the lens geometry, as illustrated in Figure 45b. Fourth, we take the inverse Fourier transform on the filtered result to reconstruct an image of the object. Finally, we add a reference beam of uniform phase and compute the irradiance distribution within the pixel, which is the square modulus of the complex image field. The irradiance value measured at the detector is taken as the mean of the irradiance distribution.

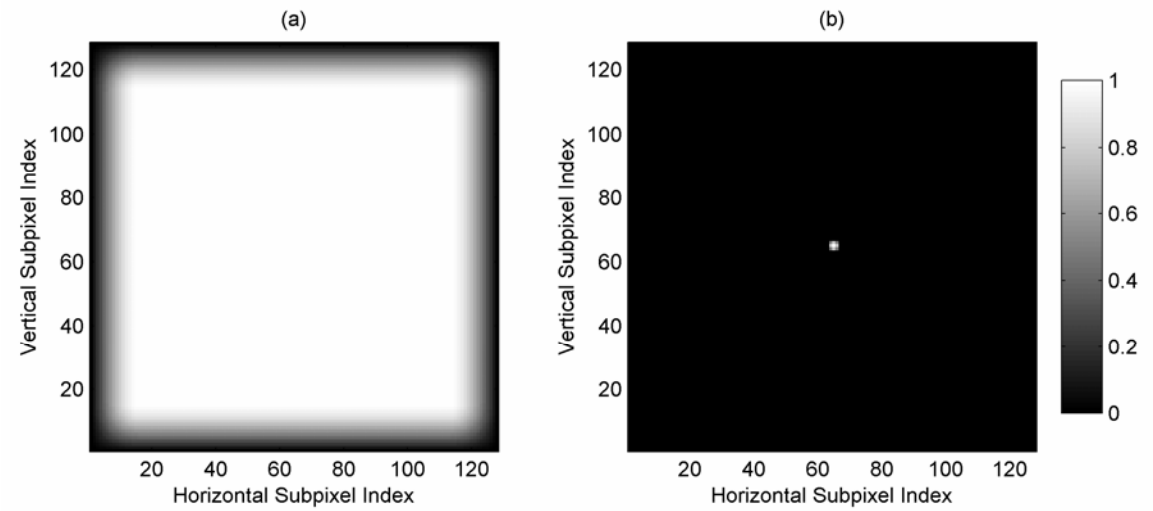

Figure 45. Illustration of numerical filters used in the Fourier optics algorithm; (a) apodization filter and (b) circular pupil filter.

With the Fourier optics approach, we can simulate the case of multiple speckles per pixel. Figure 46 is a set of single pixel speckle patterns at various *f/#* settings for the 22 μm pixel size of our Phantom 7 camera. An identical random phase input field was used for each case. For each of these pixels, a reference beam has been added, with irradiance equal to the mean irradiance of the image speckle field. The images have all been scaled for best display—the irradiances, which decrease with decreasing aperture size, are indicated in the respective scale bars.

Figure 46 demonstrates the core of the simulation program, which can be used to simulate various Doppler experiments by specifying a time- and spatially-varying input phase. For instance, by introducing a spatially constant phase function, linearly increasing with time, a constant velocity is simulated. By repeating this simulation over many instances of the random input function, one can study in more detail the predictions on harmonic content made above and in Section 1.1.5. Figure 47a shows the magnitudes of the DC and harmonic content terms (*A* and *B* respectively in Equation (25)) as a function of aperture. One hundred instances for each *f/#* are calculated and the mean values for each are plotted in Figure 47b. Least squares fits indicate that the mean DC term is proportional to $f/\#^{-2}$ and the harmonic term to $f/\#^{-1}$, which is normal for both coherent and incoherent imaging systems. The statistical variance of both the DC and harmonic components decreases with increasing *f/#*.

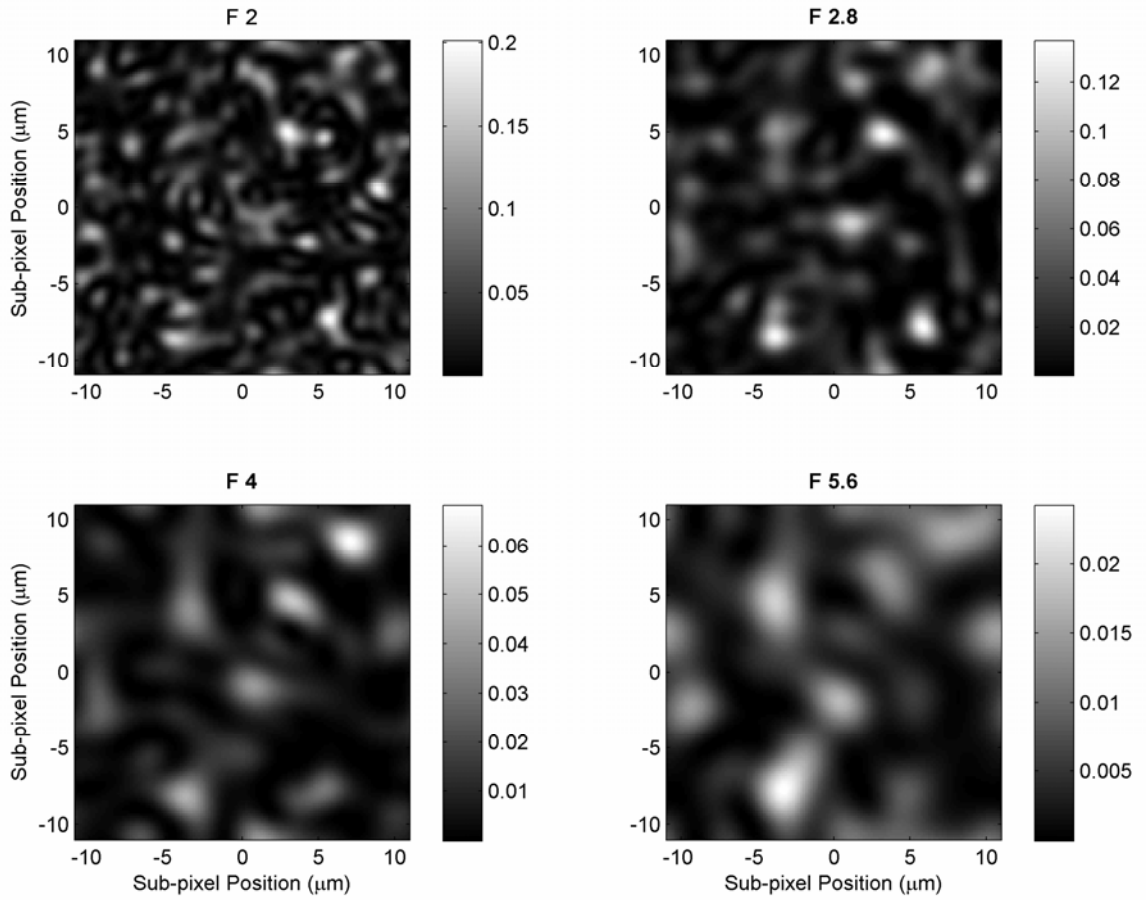

Figure 46. Single pixel speckle patterns for various lens apertures and moderate reference beam intensities. All cases have the same random input field.

As described in Equation (22), the number of speckles per pixel is approximately proportional to $f/\#^{-2}$ for a fixed pixel area and wavelength. Therefore, given the dependence of the DC and harmonic terms on *f/#* illustrated in Figure 47, the DC term (A) is proportional to N and the harmonic term (B) to $N^{1/2}$. This analysis corroborates the results presented at the beginning of this section.

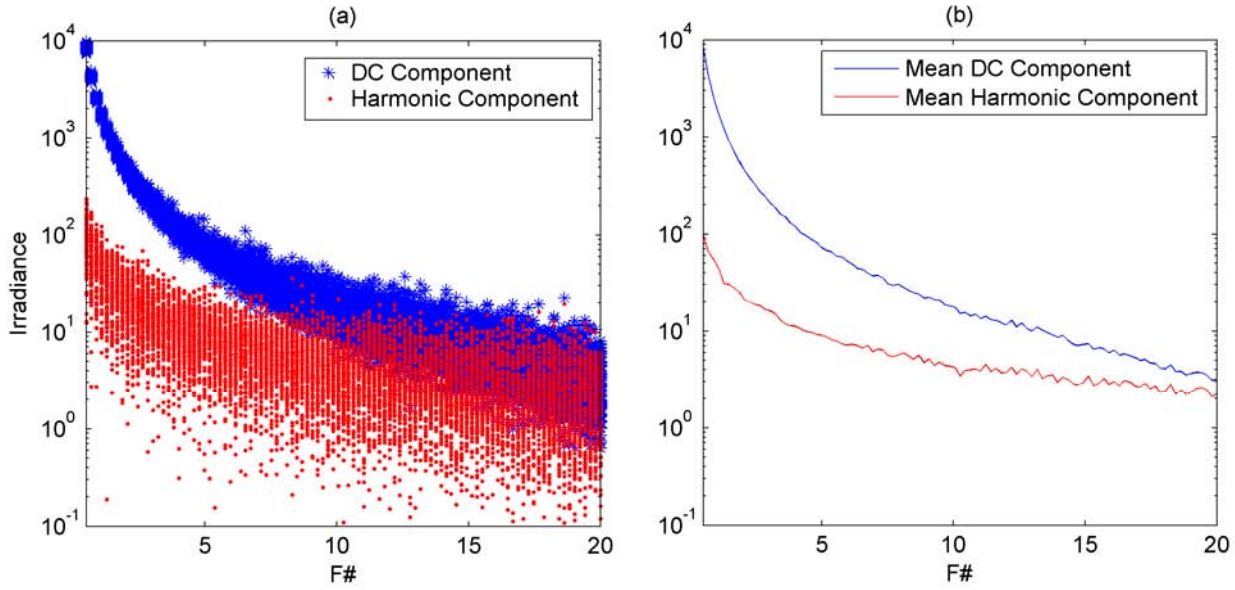

Figure 47: Simulated DC and harmonic terms for a constant velocity input (a) over many instances of random input field and (b) the corresponding mean values.

4.2.1. Speckle Translation

In efforts to simulate the measured speckle field of the rotating block described in Section 2.3, we investigated the effect of pure unidirectional translation across a surface. Figure 48 illustrates the results of a simulation with the block surface moving at 9 mm/s and a camera aperture of *f*/4. This velocity approximates the average speed of the rotating block surface moving across the projected laser beam during a 15° to 0° arc. No significant identifiable harmonic content was found in the calculated speckle irradiance time series. This was confirmed by taking an FFT of the signal.

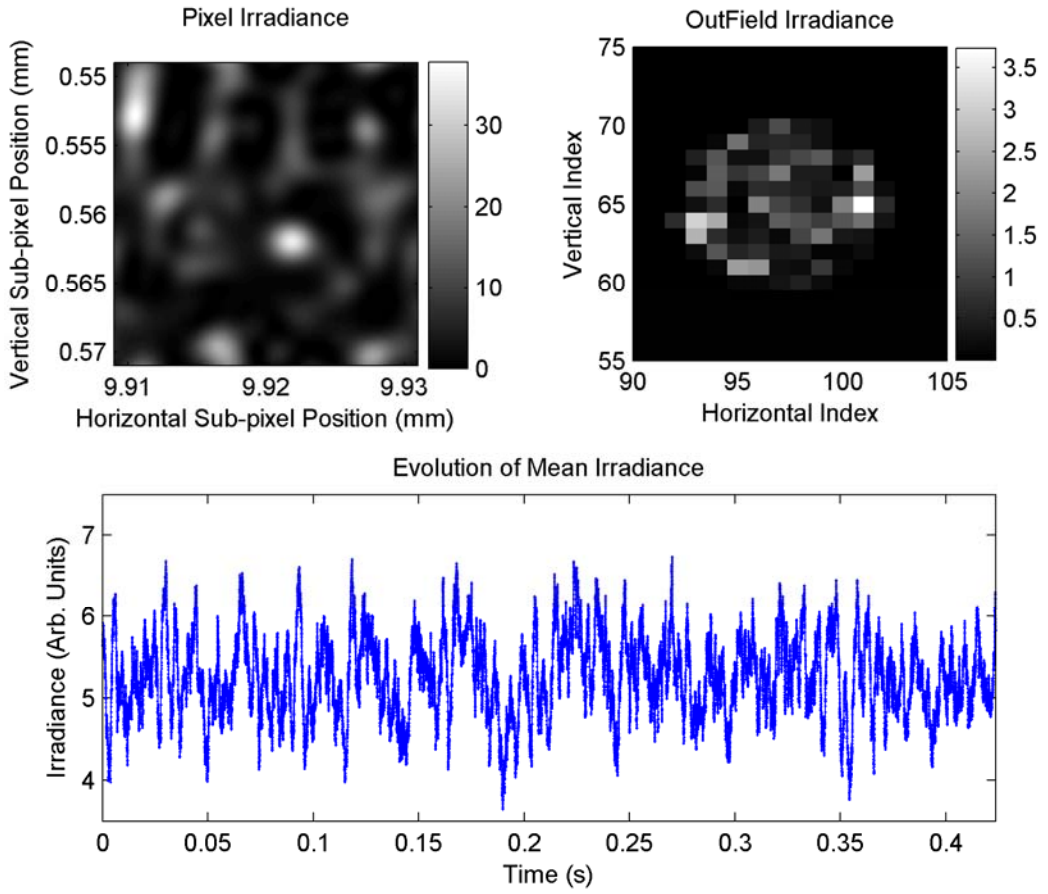

Figure 48. Simulated pixel irradiance for horizontal translation at constant velocity. The pixel irradiance map corresponds to the final timestep in the evolution plot.

To achieve a complete simulation of the rotating block, further considerations are necessary. For example, one must account for the illuminated spot changing shape as a result of the angular motion. Additionally, velocity variations within a pixel should be included, and will create a modulated waveform as seen in Figure 44. As illustrated in Figure 6, the angular position of the block is

$$\alpha(t) = \operatorname{arcsec}\left(\frac{L}{r_{op}}\right) - \Omega t\,, \tag{27}$$

where L is the length of the block and $t = 0$ s corresponds to the instant at which the laser beam first intersects the tip of the block. Therefore, the projected length of a pixel varies with the rotation of the block. The dynamic pixel length is

$$L_{pixel}(t) = L^{0}_{pixel}\sec(\alpha(t)), \tag{28}$$

where L_{pixel}^{0} is the length of the pixel at 0°. Considering an angular displacement of 15°, the length of the pixel changes 3.5%, which may be neglected in preliminary simulations; however, simulating motion across longer arcs will necessitate the investigation of the changing effective pixel size impact on the speckle pattern. The position of the center of the projected pixel is

$$r(t) = r_{op} \sec(\alpha(t)). \tag{29}$$

Therefore, the corresponding velocity of the surface across the projected pixel (or column of pixels) is

$$\dot{r}(t) = -r_{op}\Omega \sec(\alpha(t))\tan(\alpha(t)). \tag{30}$$

This formula is useful in estimating the velocity at which the rotating block surface traverses the laser beam.

5. CONCLUSIONS

We have demonstrated a widefield velocity measurement system capable of creating velocity movies of both spatially and temporally varying velocity profiles without the scanning of a traditional LDV system. This has benefits in the ability to test nonrepetitive transient experiments or experiments in which the test item is destroyed. The WLDV, with the help of the heterodyning modulator, is able to span a large range of practical laboratory velocities (limited only by the speed of the EOM), but still suffers from bandwidth limitations. The successful approach of using aliased data to unwrap the velocity and to increase the bandwidth by factors of ten has been presented and shown to be useful. Another approach under development is using a single-point LDV to control the modulator, tracking the velocity of a single point on the test structure. This assists in removing temporal aliasing issues, potentially simplifying the problem to only spatial aliasing, which can be unwrapped with the techniques discussed in this paper. These two approaches give hope that the WLDV can be developed into more than just a laboratory curiosity and find use in making velocity measurements. The WLDV technique is similar to TSPI as shown in this paper, but differs from it in the approach of the problem from the Doppler perspective rather than the ESPI perspective. We have shown that mathematically these approaches are identical, but being freed of the assumed need to fully resolve the speckles on the camera, we have eliminated the severe velocity and displacement limitations (from decorrelation) experienced with TSPI. As shown by Lehmann and demonstrated by our own work, in either TSPI or Doppler velocimetry, it is often better to have many speckles on a pixel rather than attempting to resolve the speckle field fully. This is especially true in eliminating Doppler dropouts. Another great benefit of a larger camera aperture is that more laser power can be gathered for measuring, and wider fields of view or less reflective objects can be measured for any given maximum laser power.

5.1. Future work

Speckle bands.

Many of our spectrograms contain an unidentified noise band that appears to vary in frequency with object velocity. We have called them “speckle bands” but we do not really know what causes them. In some cases they dominate the actual data. We need to identify and eliminate them from the measurements.

Fiber illumination

We attempted to implement the WLDV using single mode fibers to carry both the reference and object beam illumination. This would have made the instrument much more convenient, but the signal was corrupted with kHz-frequency noise. The noise could be due to optical feedback into the laser cavity from the specular fiber input face; this should be investigated further and eliminated.

Software development.

There are several “obvious” improvements that need to be made to the analysis codes. One is to consolidate the full analysis process—from raw data to unwrapped velocity movie—in a GUI based process.

Peak tracking.

The Doppler frequency representing the actual velocity is represented by *a single* peak in the FFT at that particular time and space. Typically however, there are several peaks in the FFT, sometimes larger than the actual Doppler frequency. The previously described codes use no intelligence in the peak-picking process. As is evident from the data presented, it is possible to generate "clean" data where a majority of the frequency peaks are in fact the correct one. It should be possible to do a "tracking peak pick" where a space-time region of "good" peaks is identified, and the region is used to limit the frequency range for searching for peaks in neighboring FFTs. Note that this is a similar concept to that used in the tracking unwrapping process, where a surface is fit to the "good" data, and that surface and its associated search region is extended into areas not yet unwrapped. This process does tend to eliminate some of the noise. However, this unwrapping currently only operates on peak data that is selected independently at each space-time FFT. The tracking process we are suggesting here would occur *before* any unwrapping, and would give the unwrapping tracker much cleaner data to work with.

3D unwrapping.
Currently the tracking unwrapping process operates on a single plane of frequency peak data, either X-Y or space-time. It requires an initial "start box" located in a region which represents unaliased data. Two innovations are needed to make this practical: first, there needs to be an automated way to select the start box, not requiring operator intervention. Second, the tracking process needs to be extended into the third dimension. An initial attempt at unwrapping multiple frames of data (to produce the final result, a "velocity movie") was only successful in a situation where the unaliased "start box" remained at the same location for all time. This required the velocity to be roughly constant in time, which is unrealistic.

6. REFERENCES

1. Amit K. Lal, C. F. Hess, H. Zhang, E. Hurtado, V. Aranchuk, V. Markov, W. Mayo, "Whole-field laser vibrometer for buried landmine detection*," Detection and Remediation Technologies for Mines and Minelike Targets VII*, SPIE, 4742, 640 (2002).
2. J. M. Sabatier, N. Xiang, A. G. Petculescu, V. Aranchuck, M. Bradley, "Vibration sensors for buried landmine detection," *International Conference on Requirements and Technologies for Detection and Removal and Neutralization of Landmines*, EUDEM2-Scotland (2003).
3. V. Markov, J. Trolinger, J. Webster, G. Pardoen, "Optoacoustical sensor to examine the structural integrity of transportation systems," *Optical Engineering*, 42, 1277 (2003).
4. M. V.Aguanno, F. Lakestani, M. P. Whelan, M. J. Connelly, "Single pixel carrier based approach for full field laser interferometry using a CMOS-DSP camera," *Detectors and Associated Signal Processing*, SPIE, 5251, 304 (2004).
5. M. V. Aguanno, M. J. Connelly, M. P. Whelan, "Full field laser vibrometry employing a novel CMOS-DSP camera," *Optical Engineering*, 4827, 123 (2002).
6. C. Joenathan, B. Franze, P. Haible, H. J. Tiziani, "Speckle interferometry with temporal phase evaluation for measuring large-object deformation," *Applied Optics*, 37, 2608 (1998).
7. C. Joenathan, B. France, P. Haible, H. J. Tiziani, "Novel temporal Fourier transform speckle pattern shearing interferometer," *Optical Engineering*, **37**, 1790 (1998).
8. C. Joenathan, P. Haible, H. J. Tiziani, "Speckle interferometry with temporal phase evaluation: influence of decorrelation, speckle size, and nonlinearity of the camera," *Applied Optics*, 38, 1169 (1999).
9. P. Haible, M. P. Kothiyal, H. J. Tiziani, "Heterodyne temporal speckle-pattern interferometry," *Applied Optics*, 39, 114 (2000).
10. X. Li, G. Tao, "Low-frequency harmonic vibration analysis with temporal speckle pattern interferometry," *Optics and Laser Technology*, 34, 259 (2002).
11. X. Li, K. Nishino, S. Yoda, "Temporal speckle pattern interferometry for measuring micron-order surface motion of a liquid bridge," *Measurement Science and Technology*, 15, 2284 (2004).
12. X. Li, G. Tao, Y. Yang, "Continual deformation analysis with scanning phase method and time sequence phase method in temporal speckle pattern interferometry," *Optics and Laser Technology*, 33, 53 (2001).
13. J. M. Huntley, M., H. Saldner, "Temporal phase-unwrapping algorithm for automated interferogram analysis," *Applied Optics*, 32, 3047 (1993).
14. C. R. Coggrave, J.M. Huntley, "Real-time visualization of deformation fields using speckle interferometry and temporal phase unwrapping," *Optics and Lasers in Engineering*, 41, 601 (2004).
15. S. E. Moran, R. L. Law, P. N. Craig, W. M. Goldberg, "Optically phase-locked electronic speckle pattern interferometer," *Applied Optics*, 26, 475 (1987).
16. J. D. Briers, "Laser Doppler and time-varying speckle: a reconciliation," *J. Opt. Soc. Am. A*, 13, 2 (1996).
17. G. Cloud, Optical Methods of Engineering Analysis, Cambridge University Press, New York, p. 53 (1995).
18. B. Boashash, ed., "Time frequency signal analysis and processing, a comprehensive reference," Elsevier Ltd., 2003, Oxford, UK.

19. M. Lehmann, “Optimization of wavefield intensities in phase-shifting speckle interferometry,” *Optics Communications*, 118, 199 (1995).
20. M. Lehmann, “Phase-shifting speckle inerferometry with unresolved speckles: A theoretical investigation,” *Optics Communications*, 128, 325 (1996).
21. M. Lehmann, M., “Measurement optimization in speckle interferometry: the influence of the imaging lens aperture,” *Optical Engineering*, 36, 1162 (1997).
22. P. Rastogi, ed., “Digital speckle pattern interferometry and related techniques,” John Wiley and Sons, 2001, NY, NY.
23. J.W. Goodman, Introduction to Fourier Optics, McGraw-Hill, 2nd ed.New York, 1996.

7. APPENDIX

Appendix A. Analysis Codes

Figure A1 shows in more detail the information flow in the WLDV analysis process.

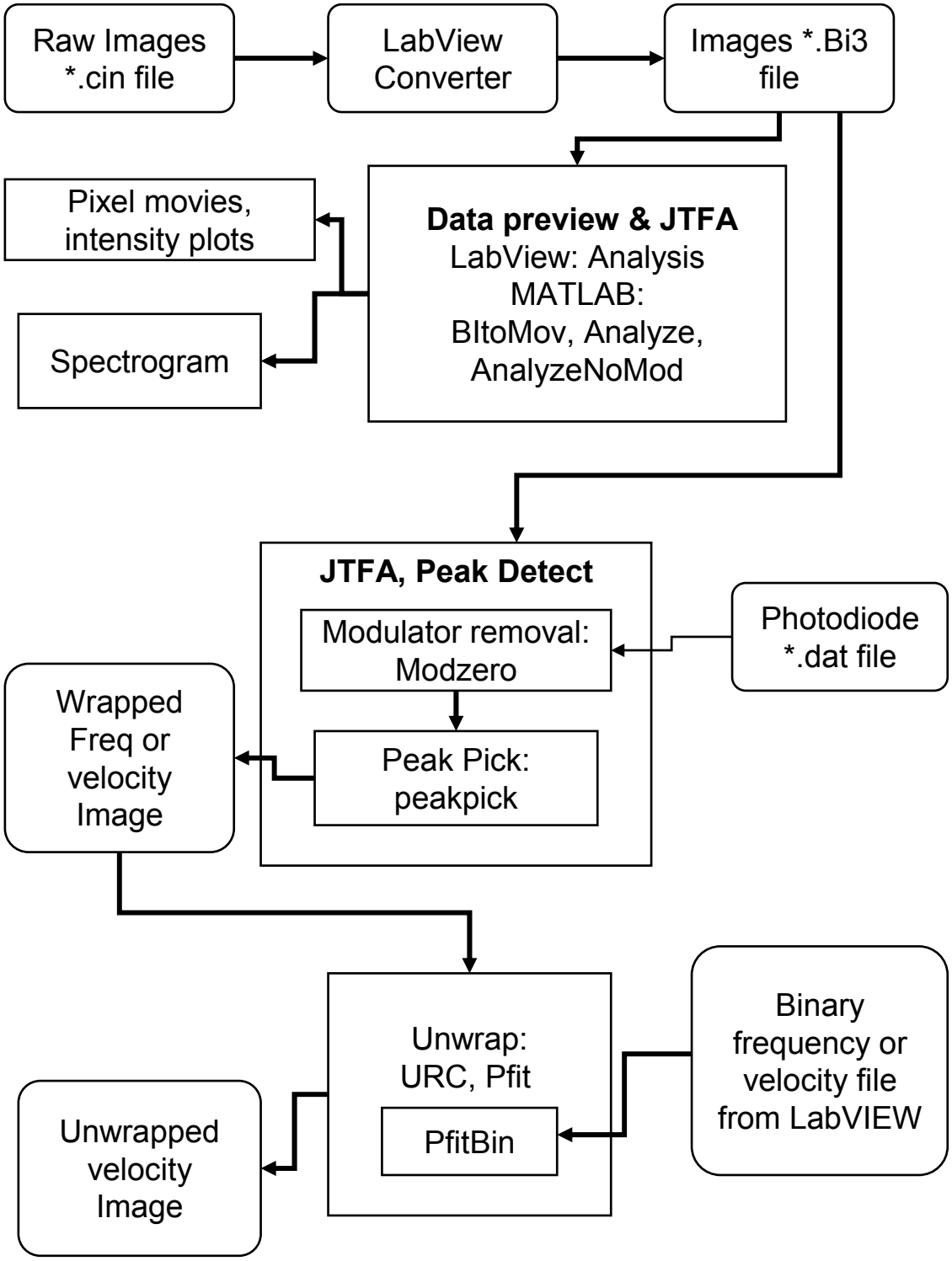

Figure A1. Software data flow in analysis routines

The purpose of this LDRD was to "prove the principle", as opposed to "provide a finished product". As such, the analysis codes supporting this research are in various stages of development. The purpose of this list and explanation is to provide a "snapshot in time" of the status and purpose of these codes. Their current function will be explained, and where appropriate, future development suggestions will be made. This explanation is intended to be sufficient for a person *already skilled in LabVIEW and/or MATLAB* to use them to analyze existing data, or future data in the same format. Presumably, future development of this technique will include development of a single, GUI based analysis suite which accomplishes all that these individual routines do.

The tables list the various current codes. These codes have been frozen and archived at the point of this writing. The text describes the more complex ones, their function and interrelation in more detail, and suggests possible paths for future work.

Data files

The raw data, taken by the Phantom 7 high speed digital camera, is in Vision Research *.CIN format, which is "space primary", that is, written in a row, column, time order. Since our analysis involves frequency extraction via JTFA, efficient execution of our routines requires a "time primary" format, with the data grouped into time sequences. The LabVIEW program *Convert Cine to Binary v1.3.llb* allows the user to select an area of interest, and reorganizes the data into a time primary file, with the file extension "Bi3". These files follow the format in Table A1. Finally, note that all the binary data is "big-endian", which is not the Windows standard, so a byte-swap must be performed for MATLAB use.

Table A1. Data file format.

Bi3 File Format		(all data is "big-endian")
Variable	**Type**	**Content**
Offset	int16	Header size, bytes
Ysize	int16	No of rows
Nframes	int32	No of camera frames, or time steps
Tsamp	int16	Camera frame rate, microseconds
Texp	int16	Camera exposure time, microseconds
Xleft	int16	First column of this dataset relative to CIN file
Ytop	int16	First row of this dataset relative to CIN file
Xsize	int16	No of columns
pixdat	int16	Data in time, row, col order. An X,Y pixel of data is an integral block of Nframes numbers beginning at byte number FramePos = {[X-1+(Y-1)* Xsize]*Nframes}*2+ Offset Where X,Y begin at 1,1 are position in this file (not in the original CIN images)

Appendix B. MATLAB Utility Subroutines

Table B1. MATLAB Subroutines

MATLAB Subroutines			
Name	**Inputs**	**Outputs**	**Comments**
BIopen	Input path to bi2 or bi3	Data structure	Queries operator for file. Note Data does not contain any actual camera data
PDread	Reads .bin photodiode data file	Structure PD, with time sequence	Optionally asks user for input file if it doesn't find it
JTFApad	Single time sequence	Set of FFT's	JTFA of a single time series with option to "pad" short data windows, interpolating FFT's
AreaJTFA	Bi2 or bi3 file	Single average JTFA. Calls JTFApad	Asks user for area, and time window, averages JTFA's for all pixels in window. NOT used in SurfColSum family
ReadGoodPix	Camera data (Data structure)	Single good time sequence	Reads 4 pixels, returns one with highest rms. Use to select ModZero input pixdat.
ModZero	Photodiode and camera data (PD and Data)	Lag time, 2D mask	Resamples PD at camera rate, time correlates, with camera data, does JTFA, thresholds to create mask
Planefit, quadfit	X,Y,Z data	Coefficients	Least squares fit, ignores NaN's
Boustro	Sample rectangle, increments	Scan path (llh corner of rectangles)	Boustrofidonic scan path generator, two directions
GetBox	Image (PFreq) array, titles	First analysis box for unwrapping	Opens an image, gets box via cursor input, leaves image open
UWplane	PFreq or Vel, supporting data	PUnw unwrapped array	Single plane unwrapper, takes box as input. Doesn't care whether it is X-Y or X-time data
BItoMov	Bi2 or bi3 file	Speckle movie	Used to view raw data (speckle images) from bi files. Inverts the conversion from cin to bi3

- **BIopen** queries the user for a Bi3 data file name and path. It accepts the file name and path as an argument, and if they exist, skips the user query. (Many of the datasets consist of matching image (Bi3) and photodiode (Bin) data. If these files have the same name except for extension, either BIopen or PDread can be called with file information so that the user need only be queried once). It also works with the obsolete "Bi2" file format, supplying default parameters for those variables not defined in the Bi2 header. It opens the file (read only, big-endian format), reads the header information and creates structure "Data", containing file and header information. See comments in the code listing for Data structure elements. This routine does not read any actual data. The file is left open.

- **PDread** like BIopen either opens the file in the argument, or queries the user if the file does not exist. It opens a photodiode (Dat) file, creates structure PD (see listing for details), and reads the photodiode data into structure element PD.data. The file is closed on exit.

- **JTFApad** is the "core" JTFA routine, returning a single JTFA for a time sequence of data. It uses structure "Samp" for window settings. It returns a single JTFA array with dimensions (Nfreq,Nfft) where Nfreq is the number of frequency samples from DC to the Nyquist frequency, and Nfft is the number of FFT's (number of time windows within the sample). It will optionally pad the sample windows with zeros to create interpolated FFT's—this is useful for rapidly varying velocities such as harmonic vibrations.

- **AreaJTFA** prompts the user for a set of pixels over which to average FFT's to create a single JTFA. This is useful for datasets where all columns have the same velocity, or for cases where the velocity is relatively constant in an area covered by many pixels. It creates structures Area and Raw, which are the pixels to analyze relative to the (local) Bi3 dataset, and to the (global) CIN data respectively. It calls JTFApad once for each pixel in the area, averages the JTFA magnitudes, normalizes by total number of pixels and returns an average JTFA.

- **ModZero** aligns the photodiode data to the camera data. It resamples the photodiode data at the camera rate (using structures Samp and PD for sample rate information) and correlates the resampled photodiode time sequence with a camera pixdat sequence to determine how much one must be shifted to align in time with the other. It truncates both data sequences to provide a pair of sequences which overlap and align. It then uses JTFApad to create a JTFA of the resampled photodiode sequence, applies a threshold to identify frequencies dominated by modulator data, and creates a mask array which can be used to zero out modulator frequencies in subsequent JTFA analyses. It returns the alignment information and the mask array (as a sparse matrix).

- **UWplane** unwraps a single X-Y plane of data. X and Y can be both spatial, or one can be time. The data can be frequency or velocity. It takes a path (typically created by **boustro**), and an initial "good data" area (typically created by **GetBox**) and returns the unwrapped data and the "region mask" described in the Analysis Techniques section.

Appendix C. MATLAB Main Programs

Table C1. MATLAB analysis programs

MATLAB Analysis Programs			
Name	**Inputs**	**Outputs**	**Comments**
BItoMov	Bi2 or BI3	Pixel Image, movie or average	Inverts Bi3 process. For visualization of raw data.
Analyze	Bi2 or bi3, operator area select	JTFA,	Single average JTFA spectrogram from selected pixel area, modulator removed
AnalyzeNomod	Bi2 or bi3, operator area select	JTFA	Single average JTFA spectrogram from selected pixel area (no modulator removal)
SurfColsumNomod,	Bi2 or bi3	JTFA, Zig-Zag plot, PFreq array and plots	Like SurfColsum2, but for data with no modulator. Has user input time window
SurfRowsumNomod	Bi2 or bi3	JTFA, Zig-Zag plot, PFreq array and plots	Like SurfColsumNomod, but average column data for each row. Has Time WIndow
SurfColsum2 (as of Jul 06 there is no SurfRowSum2)	Bi2 or bi3	JTFA, Zig-Zag plot, PFreq array and plots	Averages over rows for each col. Removes modulator. PFreq is image of frequency peaks on time and column. Has user input time window
SurfNomod	Bi2 or Bi3	Pfreq datacube (X,Y,Time), movie	No unwrapping
Peakpick	Average from Surf…	PFreq, Vel	Select new FirstFreq and Threshold for JTFA to pick peaks
Pfit	PFreq from SurfCol…	Diagnostic. Plots ONE fitted plane.	Must run one of SurfColSum family first. Check mirroring across region boundary.
URC	PFreq	Unwrap a row-column frame	Must run Surf… first, and peakpick

- **Analyze & AnalyzeNomod** are essentially executors for AreaJTFA with and without ModZero. They use AreaJTFA to calculate one JTFA of an area of pixels, and plot the result. Analyze calls ModZero to remove the modulator frequencies, and plots that result as well. After execution, array Average, the resulting average JTFA, is left in the workspace for further analysis. AnalyzeNomod generated Figure 11 in the Analysis Techniques section.

- **Surf…** family. These routines all generate a frequency "surface" in the form of frequency (or velocity, since they are related by a constant) as a function of two other variables, either X-Y or space-time. **SurfColsumNomod** and **SurfRowsumNomod** sum FFTs within columns and rows respectively, without removing the modulator frequencies. Since a lot of our data has a constant frequency within each column, SurfColsum tends to be the one to use. SurfColsumNomod created Figure 27in the Analysis Techniques section. **SurfColsum2** operates similarly, averaging over rows within each column, but then it calls ModZero to remove the modulator frequency components. The output, left in the workspace after execution, is array PFreq, which is an array of the frequency where the peak of each FFT

occurs, as a function of column number and time. This data is selected from each FFT in array Average by a simple peak-pick algorithm, which ignores frequencies below a given lower limit, and ignores FFT peaks below a given threshold. The Surf family typically plots the results in a few formats, including a color image and a 3D dot plot, such as shown in Figures 27 and 28 in Analysis Techniques. **SurfNomod** does not average, but creates a frequency surface for each time step, resulting in a "datacube" of PFreq with dimensions of Row, Column, and Time. It generates a movie of the color images as they evolve with time.

- **peakpick** is simply a repetition of the peak picking algorithm applied to array Average at the end of the Surf…execution. It queries the user for a new first frequency and peak threshold, creates a new PFreq array and plots the result. It overwrites the existing PFreq array, leaving the final selection in the workspace. An example of why we might want to modify the low frequency bound is shown in Figure C1a. This is a frequency spectrogram of the turntable dataset used for the Analysis Techniques section as shown in, Figure 32 and Figure 33. The "peak" in higher frequencies represents the actual turntable velocity as it varies with column number (turntable radius). There is a noise band at about 1.5 kHz, which happens to be the 7th frequency sample. Figure C1b and Figure C2 show two different peakpick results.

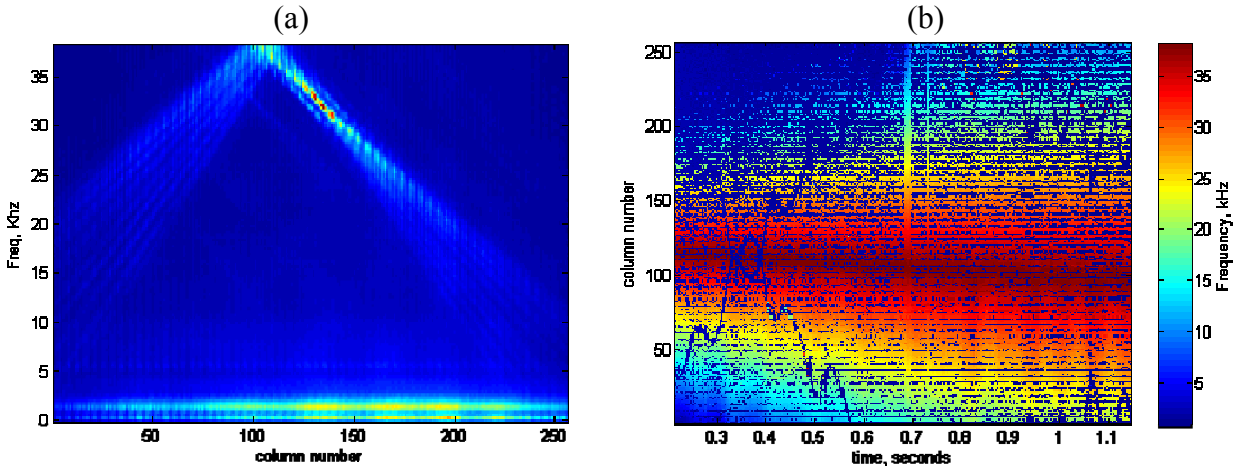

Figure C1. (a) Spectrogram over all time of turntable data. (b) Turntable velocity, First Frequency sample = 4 (the default)

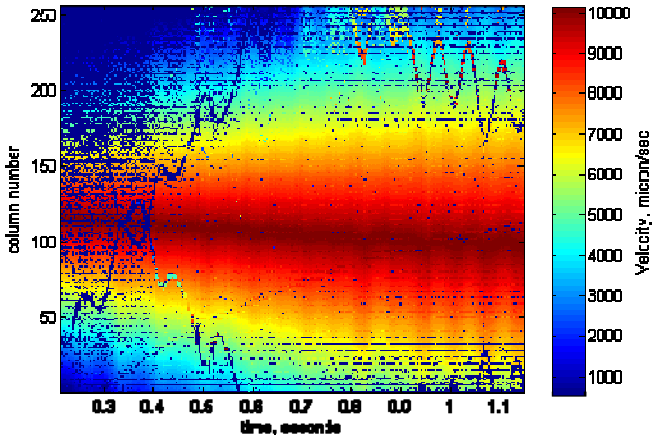

Figure C2. Turntable velocity, First Frequency = sample 7

- **Pfit** is a diagnostic program written to explore the fitting process for the first candidate plane in the unwrapping, as explained in Analysis Techniques section. The generated results are shown in Figure 34 and Figure 35 in that section. The program is meant to be run after the Surf… execution, operating on array PFreq.

- **URC** unwraps a single row-column velocity (or frequency) array, using subroutine UWplane described above.

Appendix D. LabVIEW programs

Table D1. LabVIEW programs.

LabVIEW Programs			
Name	**Inputs**	**Outputs**	**Comments**
Convert Cine to Binary v1.3.llb, .lvproj	BI3	Input is a cine file, selection of area to convert	Creates a *.bi3 file for use in LabVIEW or Matlab routines as previously described.
Velocity Analysis-Binary-Cine v1.4.llb, .lvproj	BI3, cin	Can read in either a cine or bi3 file type for analysis	Used for quick analysis and checking of the data. Does JTFA, zig-zag plot, and simple peak tracking.
Cont Acq to File (binary).vi	Bin	Binary DAQ file	DAQ stream to disk, with triggering. Used for photodiode acquisition.
Display-Analyze LDV data.llb	Bin	Plots bin file, FFT, resample	Reads, plots and analyzes bin files
Frequency Tracking Software.vi	analog velocity	modulator drive signal	Reads single point LDV velocity, generates modulator frequency via FPGA

- **Convert Cine to Binary v1.3** This program converts a selected Vision Research Phantom "cin" file to the bi3 format. The bi3 format is outlined in Appendix A1. The user can select the entire image or a subset of the image for conversion. The header information is created automatically, however, the frame-rate is not automatically read from the cine file. It is contained in the file, but errors in the cine dll for use with LabVIEW do not properly decode the frame-rate. All other parameters are done correctly.

- **Velocity Analysis-Binary-Cine v1.4** Velocity analysis is used for analyzing either the cin or bi3 file types. The functionality contained is similar to a number of the MatLAB programs, but simpler in scope. These include JTFA, plotting irradiance vs time, zig-zag plots, and peak tracking. There is no alias unwrapping functionality. Simple velocity movies can also be created and viewed, however, aliasing is not unwrapped.

- **Cont Acq to File (binary).vi** is an example program from LabVIEW, with some minor additions, including: Triggering, and hardware settings. The program was used to stream the high speed (4 MHz typically) photodiode data to disk.

- **Display-Analyze LDV data.llb** Simple data viewer for the binary files created by Continuous Acquire. Allows the user to view the time history, zoom in on sections of it, and do FFT's of data sets. Also included is a resample function to resample the data into the camera frame rate.

- **Frequency Tracking Software.vi** is written in LabVIEW Compact RIO software to be downloaded onto a Field Programmable Gate Array (FPGA) for hardware execution. This method allows deterministic and faster loop times. The FPGA has a DAQ channel that reads the velocity, and then calculates the phase increment (frequency word). This is then output to a the AD9835 DDS board to control the modulator frequency. To calculate the phase increment from the Analog Input (AI) value, the following equation should be used:

$$\Delta\phi = \frac{2^{32}}{50\times10^{6}}\left[\frac{2}{\lambda(m)}\left\{\frac{1}{LDV(V/mm/s)}\frac{20}{2^{16}}\right\}AI\right]$$

$$\Delta\phi = \frac{98.5503}{LDV(V/mm/s)}AI$$

Appendix E. MATLAB simulation programs

Table E1. MATLAB simulation programs

MATLAB Simulation Programs			
Name	**Inputs**	**Outputs**	**Comments**
LDVsim	*f*/#, pixel size	Speckle image	"core" of simulation routines
LDVsim_Fno	List of *f*/#s, no. statistical instances, no. frames	Speckle movie, integrated irradiance, translation distance,…	Simulate speckle pattern due to sub-pixel uni-directional, in-plane translation of object
LDVsim_TranslateX	*f*/#…	Speckle movie, integrated irradiance, translation distance,…	Simulate speckle pattern due to sub-pixel uni-directional, in-plane translation of object
CircPupil	FFT and pixel array size, freq cutoff	Multiplicative filter array (mask)	generates circular pupil in quadrants appropriate for FFT freq domain filtering
apodizer	No. rows/columns, percentage apodized	Multiplicative filter array (mask)	Eliminate leakage due to FFT periodicity
BigPhaseNYxNX.dat	N/A	N/A	NY-by-NX random phase array for LDVsim-based codes.

- **LDVsim** This script provides the initial core simulation of the speckle image formation process in the WLDV. The imaging process is based on Goodman, "Introduction to Fourier Optics" (McGraw Hill 1968), eq 6-17. This includes the concepts described early in chapter 6: the diffraction effects are calculated on the ideal geometric image. Magnification is applied before any diffraction and filtering. Hence, the "input" image is in the camera domain, and any phase effects due to motion of the object need to be demagnified to camera space to be applied. A demagnification on the order of 10 is assumed, so the image distance is essentially equal to the lens focal length. This makes cutoff freq dependent on *f*/#. The velocity phase is simply added to the random object phase before forward transform. The core code neglects spatially varying velocity fields.

- **LDVsim_Fno** This script calculates a number of samples of a Doppler cycle for a constant out-of-plane velocity over an aperture. This occurs for several apertures and several statistical instances. Input parameters include pixel size and discretization, selected *f*/#s, and number of frames and statistical instances. The primary outputs are data and plots of harmonic content and DC components versus *f*/#.

- **LDVsimTranslate_X** This program provides speckle pattern simulations based on the Fourier optics method. The motion of the simulated surface is confined to in-plane translation along a single direction. Optional input includes the sample array size (column discretization), the length of the traversal path, the number of sub-pixels moved per frame, and the absolute position of the pixel center. Optional output includes irradiance distributions over pixel area at each step and the irradiance representation of the pre-imaged pixel.

- **CircPupil** This program generates a pupil array for transform domain filtering in a Fourier optics system. Pixels cut by the cutoff frequency are assigned a grey level between zero and one proportional to the amount of the pixel included in the pupil. Zero frequency is at pixel

(1,1) and the low frequencies are replicated at the corners ready for direct multiplication. FFT array dimensions should be even (typically powers of two). This routine is most efficient for small pupils as it calculates an OUTER bounding box and sets all pixels outside to zero without calculating radii. It uses a different approach for the INNER box. Inputs include horizontal (column) and vertical (row) fft array sizes, horizontal and vertial frequency pixel sizes, and the cutoff frequency. Output includes the pupil multiplicative array filter.

- **apodizer** This program generates an apodizing multiplicative filter array. The apodizing function is sinusoidal. Input includes the number of rows and columns of the desired array and the fraction of rows and columns to apodize. The sole output is a 2-D array that apodizes a target array when multiplied element-by-element.

- **BigPhaseNYxNX.dat** These are NY-by-NX random phase arrays used in LDVsim-based codes. These arrays are used when a fixed phase array is required to compare speckle simulations for varying F/#s, velocities, etc. The NY dimension corresponds to the discretization of the pixel y-axis and the NX dimension corresponds to the discretization of the pixel x-axis.

Distribution

1	MS0123	D. Chavez, LDRD Office	01011
1	MS0555	Bruce D. Hansche	01522
2	MS0899	Technical Library	04536
1	MS1070	Jordan E. Massad	01526
1	MS1070	Phillip L. Reu	01526
1	MS1454	Mark S. Garrett	02554
1	MS1454	Kevin J. Fleming	02554
2	MS9018	Central Technical Files	08944