Dissertations, Master's Theses and Master's Reports

2019

OPTICAL VORTEX AND POINCARÉ ANALYSIS FOR BIOPHYSICAL DYNAMICS

Anindya Majumdar
Michigan Technological University, amajumda@mtu.edu

www.manaraa.com

OPTICAL VORTEX AND POINCARÉ ANALYSIS FOR BIOPHYSICAL
DYNAMICS

By

Anindya Majumdar

A DISSERTATION

Submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

In Biomedical Engineering

MICHIGAN TECHNOLOGICAL UNIVERSITY

2019

© 2019 Anindya Majumdar

This dissertation has been approved in partial fulfillment of the requirements for the Degree of DOCTOR OF PHILOSOPHY in Biomedical Engineering.

Department of Biomedical Engineering

Dissertation Advisor: *Dr. Sean J. Kirkpatrick*

Committee Member: *Dr. Jeremy P. Bos*

Committee Member: *Dr. Jeremy Goldman.*

Committee Member: *Dr. Jingfeng Jiang*

Department Chair: *Dr. Sean J. Kirkpatrick*

Table of Contents

Preface

The work presented in this dissertation has been completed under supervision of my advisor, Dr. Sean J. Kirkpatrick. Dr. Kirkpatrick currently also serves as the chair of the Dept. of Biomedical Engineering at Michigan Technological University, Houghton, Michigan (USA). Parts of this work have so far been published in 2 papers within the Journal of Biomedical Photonics and Engineering (J-BPE) and 3 papers within the Proceedings of the International Society of Optics and Photonics (SPIE). All of these 5 are peer-reviewed articles. Additionally, parts of this work are in consideration for publication as at least 2 more peer-reviewed journal articles.

Chapter 1 contains excerpts previously published as parts of "Statistical studies on optical vortices in dynamic speckle fields" by A. Majumdar and S.J. Kirkpatrick, in J-BPE in 2018 (doi: 10.18287/JBPE18.04.020301).

Parts of Chapter 2 are under consideration for at least 1 peer-reviewed journal publication in the future.

Chapter 3 contains excerpts previously published as parts of "Statistical studies on optical vortices in dynamic speckle fields" by A. Majumdar and S.J. Kirkpatrick, in J-BPE in 2018 (doi: 10.18287/JBPE18.04.020301), and as parts of "Optical Vortices as potential indicators of biophysical dynamics" by the same authors in the Proceedings of SPIE in 2017 (doi: 10.1117/12.2251026).

Parts of Chapter 4 are under consideration for at least 1 peer-reviewed journal publication in the future.

Chapter 5 contains excerpts previously published as parts of "Spatial Poincaré Plots as Descriptors of Speckle Pattern Second-Order Statistics" by A. Majumdar and S.J. Kirkpatrick, in J-BPE in 2017 (doi: 0.18287/JBPE17.03.030501), and as parts of "Poincaré descriptors of long and short- range correlations in the spatial and temporal fluctuations of coherently scattered light" by the same authors in the Proceedings of SPIE in 2018 (doi: 10.1117/12.2287590) and "Ellipticity imaging for visualizing and quantifying long and short range correlations in laser speckle data II: phantom and animal studies" by Majumdar *et al.* in the Proceedings of SPIE in 2019 (doi: 10.1117/12.2507019). Parts of this chapter are also under consideration for at least 1 peer-reviewed journal publication in the future. The animal studies described in this chapter were performed by the authors Ivan Fedosov, Arkady Abdurashitov, Oleg Grishin, Olga Sindeeva, Valery Tuchin and Sean J. Kirkpatrick at Saratov State University, Russia. The raw data from the animal studies were also obtained by these authors, while I performed the analysis that is presented in this dissertation.

The permission granted for using parts of these publications have been attached in the Appendix section of this dissertation.

Acknowledgements

First and foremost, I would like to thank my advisor, Dr. Sean J. Kirkpatrick, without whose constant support and encouragement, this dissertation would not have been possible. His guidance and willingness to help has shaped most aspects of this di work, and has also been instrumental in smoothening out multiple facets of my graduate student life. For this, I will always be grateful.

Next, I would like to extend my thanks to the committee members, Dr. Jeremy Bos, Dr. Jeremy Goldman, and Dr. Jingfeng Jiang Their timely advice and critical feedback have been essential at various times during my journey to finish the work presented. I would also like to thank Dr. Orhan Soykan for his friendly advice regarding maneuvering through the academic and scientific body at large. I would also thank Dr. Ravindra Pandey and Dr. Miguel Levy, from the Dept. of Physics, who were instrumental in my joining Dr. Kirkpatrick's group in the first place.

Graduate Student Govt. (GSG) at Michigan Tech for funding my travels to various conferences over the years.

I would also like to thank Ms. Coreen Dompier and Ms. Stacey Sedar at the Biomedical Engg. Dept. front office, and the Graduate School office and International Programs & Services at the Admin Building. They have always been kind, patient and helpful in answering the multitudes of questions I have had over the years. They have also been instrumental in smoothening my life at Michigan Tech.

On a more personal front, I owe a lot to my family back home. They have borne the brunt of my range of emotions, and were always present for any need I had. Their contribution in this work and my life beyond, can't be summed up in a few lines. I would also like to thank my long time partner, Ms. Divya Kamath, for her constant support and assistance during my life at Michigan Tech. She has shaped and improved me as a person, over the last few years. I also thank my friends and colleagues at MTU and the Houghton community in general for their valuable companionship. I have had no trouble in finding occupying myself with other activities whenever I needed to take my mind of work. A special mention here is in place for piece of equipment that has brought hundreds of hours of joy over the years- the personal computer of Mr. Srinivas Kannan. I, for sure, have spent more time on it than my own computer. It would have been unfair to not acknowledge the entertainment it has brought in, and I think many individuals would vouch for that.

Abstract

Coherent light - such as that from a laser - on interaction with biological tissues, undergoes scattering. This scattered light undergoes interference and the resultant field has randomly added phases and amplitudes. This random interference pattern is known as speckles, and has been the subject of multiple applications, including imaging techniques. These speckle fields inherently contain optical vortices, or phase singularities. These are locations where the intensity (or amplitude) of the interference pattern is zero, and the phase is undefined.

In the research presented in this dissertation, dynamic speckle patterns were obtained through computer simulations as well as laboratory setups involving scattering from phantoms and animal tissues. Optical vortices were tracked within these patterns. Novel techniques were applied to relate scattering media dynamics with the temporal evolution of the speckle fields and the optical vortex locations. Parameters, such as optical vortex trail lengths, mobility of optical vortices and charge separation between different types of vortices were introduced and calculated. Mathematical formulations, namely Poincaré descriptors, were employed to analyze the statistics of speckle intensity and optical vortex dynamics.

A brief review of the advancements in the understanding and detection of optical vortices is presented. This is followed by the theory behind Poincaré analysis. It is concluded that Poincaré descriptors can be used to characterize the correlation in a data series. Speckle patterns with different dynamic behaviors – such as Brownian and Lorentzian modes of decorrelation between consecutive frames, at varying rates - were studied. It was noted that

measuring the optical vortex parameters in a dynamically evolving field, provided insights into the decorrelation characteristics of the scattering medium. As speckle size is a statistical measure of the intensity variation in the field, the use of Poincaré descriptors to estimate speckle size is demonstrated. Additionally, these descriptors differentiate between short- and long-range orders in data. Laser speckle images obtained from flow in fluid phantoms and animal tissues indicated that Poincaré analysis provides an alternate method of quantifying flow.

1 Introduction to Optical Vortices

In this chapter, we introduce phase singularities in wavefields, and why they are alternatively called "optical vortices". We discuss some of their properties which have found modern applications. Excerpts of Section 1.3 of this chapter have been taken, with permission, from "Statistical studies on optical vortices in dynamic speckle fields" by A. Majumdar and S.J. Kirkpatrick, published in the Journal of Biomedical Photonics and Engineering in 2018. The letter of permission is attached in Appendix A.2.

1.1 Light Waves

Light is a form of electromagnetic radiation. It has been understood to display properties of waves as well as particles, and is thus considered to have a dual nature. As waves, it has properties such as wavelength, amplitude and phase. In vacuum, the speed of light is established to be exactly 299,792, 458 m/s, often approximated to 3×10^8 m/s. The wavelength of visible light is commonly taken to be the range of 400-700 nm, but the exact range varies slightly by individual.

As electromagnetic waves, light is governed by Maxwell's equations. These equations and their implications for the study of phase singularities are discussed in the following section.

1.2 Maxwell's Equations

As we will be dealing only with coherent light, the following discussion assumes monochromaticity. Additionally, the fields explored have no source of charge within them,

www.manaraa.com

and only paraxial effects are considered. The waves are assumed to propagate in vacuum and polarization effects are ignored.

Maxwell's equations relating the electric field $\vec{E}$ and the magnetic field $\vec{B}$ are considered the backbone of electrodynamics. They state that [1]

$$\nabla \bullet \vec{E} = \frac{\rho}{\varepsilon_0} \tag{1.1}$$

$$\nabla \bullet \vec{B} = 0 \tag{1.2}$$

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \tag{1.3}$$

and

$$\nabla \times \vec{B} = \mu_0 \vec{J} + \mu_0 \varepsilon_0 \frac{\partial \vec{E}}{\partial t} \tag{1.4}$$

where ρ is the charge density (volumetric) in the region of interest, $\vec{J}$ is the displacement current introduced by Maxwell, and ε_0 and μ_0 are the permittivity and permeability of vacuum, respectively. If the region of interest has no sources of charge and no current, we have $\rho = 0$ and $\vec{J} = 0$. Under these conditions, we obtain

$$\nabla^2 \vec{E} = \nabla(\nabla \bullet \vec{E}) - \nabla \times (\nabla \times \vec{E}) = 0 - \nabla \times (-\frac{\partial \vec{B}}{\partial t}) = \frac{\partial(\nabla \times \vec{B})}{\partial t} = \mu_0 \varepsilon_0 \frac{\partial^2 \vec{E}}{\partial t^2} \tag{1.5}$$

A monochromatic wave can be represented as

$$\vec{E}(\vec{r},t) = \text{Re}\{\vec{E}(\vec{r})e^{-i\omega t}\} \tag{1.6}$$

2

Substituting into Eq. (1.5) reduces Eq. (1.6) to

$$\nabla^2 \vec{E}(\vec{r}) + k^2 \vec{E}(\vec{r}) = 0 \tag{1.7}$$

where $k = \omega\sqrt{\mu_0 \varepsilon_0}$ is known as the *propagation constant* of the wave. Eq. (1.7) is the vector form of the Helmholtz equation [2]. If the direction of propagation is constant throughout, leads to the scalar Helmholtz equation

$$\nabla^2 U(\vec{r}) + k^2 U(\vec{r}) = 0 \tag{1.8}$$

Paraxial beams have an exponential form

$$U(\vec{r}) \approx u(\vec{r})e^{ikz} \tag{1.9}$$

where z is the direction of propagation. Further adding the assumption of slowly varying envelopes, these paraxial rays reduce the scalar Helmholtz equation (1.8) to the *paraxial wave equation*

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + 2ik\frac{\delta u}{\delta z} = 0 \tag{1.10}$$

1.2.1 Laguerre-Gauss Paraxial Beams

Among the commonly studied solutions to the paraxial wave equations is the Gaussian beam, which has a Gaussian intensity distribution Eq. (1.7) at the z=0 plane.

$$I(x,y,0) = u(x,y,0)u^*(x,y,0) = I_0 e^{-2(x^2+y^2)/w_0^2} \tag{1.11}$$

3

The Laguerre-Gauss (LG) beams, one of the solutions to the paraxial wave equation, are a subset of Gaussian beams dealt with in the cylindrical coordinate system (as opposed to the Hermite-Gauss (HG) beams in the Cartesian coordinates). They form a set of cylindrically symmetric solutions given by the general form [3]

$$u_{pl}(\rho,\phi,z) = \frac{C}{\sqrt{1+(\frac{z}{z_R})^2}} \left\{ \frac{\sqrt{2}\rho}{w(z)} \right\}^l L_p^l \left(\frac{2\rho^2}{w^2(z)} \right)$$

$$\times \exp(il\phi)\exp(-\frac{\rho^2}{w^2(z)})\exp(-\frac{ik\rho^2 z}{2(z^2+z_R^2)})\exp(-i(2p+l+1)\tan^{-1}(\frac{z}{z_R})) \qquad (1.12)$$

where $w(z)$ is the radius of the beam given by $w(z) = w_0\sqrt{1+(\frac{z}{z_R})^2}$ with w_0 being the beam waist and z_R being the Rayleigh range, related by $z_R = \frac{\pi w_0^2}{\lambda}$ for a Gaussian beam of wavelength λ [4]. The Rayleigh range of the beam is defined as the distance at which the width of the beam has increased to √2 times the waist width (at z=0). The numbers l and p are indices of the associated Laguerre polynomials $L_p^l(x)$ given by

$$L_p^{|l|}(x) = (-1)^{|l|} \frac{d^{|l|}}{dx^{|l|}} L_{p+|l|}(x) \qquad (1.13)$$

where $L_n(x)$ are the more commonly known Laguerre polynomials [5]

$$L_n(x) = \frac{e^x}{n!} \frac{d^n}{dx^n}(e^{-x}x^n) \; . \qquad (1.14)$$

Normalizing Eq (1.12) leads to the amplitude distribution [5]

4

$$u_{pl}(\rho,\phi,z) = \sqrt{\frac{2p!}{\pi(p+|l|)!}} \frac{1}{w(z)} \left\{ \frac{\sqrt{2}\rho}{w(z)} \right\}^{|l|} L_p^{|l|}\left(\frac{2\rho^2}{w^2(z)}\right)$$
$$\times \exp(il\phi)\exp(-\frac{\rho^2}{w^2(z)})\exp(-\frac{-ik\rho^2 z}{2(z^2+z_R^2)})\exp(-i(2p+l+1)\tan^{-1}(\frac{z}{z_R})) \qquad (1.15)$$

The normalization factor comes from defining these beams to have a unit cross-sectional intensity

$$\int |u_{pl}(\vec{r})|^2 d^2r = 1. \qquad (1.16)$$

We note that the expression Eq. (1.15) for the amplitude distribution reduces to an ordinary Gaussian beam for $p = l = 0$. The vanishing of the axial amplitude (where $\rho = 0$) is also evident for all $l \neq 0$. Additionally, due to the $e^{il\phi}$ term, the phase rotates by an integral multiple of 2π around the axis, at constant z. Thus, we have an "optical vortex" with zero amplitude at a point and the phase rotating by a multiple of 2π around this point. On including the explicit time dependence term $e^{-i\omega t}$, we have a vortex that is traverses along the direction of propagation og the travelling LG beam.

1.3 Properties of Optical Vortices

1.3.1 Topological Charge and Index

The number l is referred to as the *topological charge* of the vortex, and is a conserved quantity during propagation. This refers to the integer multiple of 2π by which the phase of the field rotates around a vortex, and is mathematically given as [6]

$$l = \frac{1}{2\pi} \oint_C \nabla \phi(\vec{r}) d\vec{r} \qquad (1.17)$$

This quantity dictates the angular momentum of light carried by the beam, whose value is $l\hbar$ per photon. The topological charge is an integer, with its sign denoting whether the phase rotates clockwise or counter-clockwise on propagation. The integral value is a necessary condition to conserve the continuity of the wavefield. To conserve this number, vortex generation and annihilation typically occurs in pairs of opposite and equal topological charges. Additionally, it has been observed that vortices with topological charge $l = 1$ are generally not affected by perturbations in the optical field, while vortices with higher charges eventually break down into vortices with $l = 1$. We also note from Eq. (1.15) that the intensities of LG modes of the orders $|l|$ and $-|l|$, with the same p, are same. Thus, the phase cannot be determined uniquely from the intensity alone. This gives rise to a class of problems in optics, known as *phase retrieval problems*.

Another conserved property of a wave field is what is called its *topological index*. This is mathematically defined as [6]

$$n = \frac{1}{2\pi} \oint_C \nabla \theta(\vec{r}) d\vec{r} \qquad (1.18)$$

where $\theta(\vec{r})$ is any vector field. Physically, the topological index indicates the behavior of the gradient vectors to the surfaces of constant phase, in a wave field. Constructing a vector field from such gradient vectors, the topological index is the number of rotations such a vector field undergoes as a closed path around a feature in the field is traversed. These features may be singularities, sinks, sources or saddle points in the concerned wave field.

6

1.3.2 Angular Momentum of Light

Poynting in 1909 was among the earliest to discuss the possibility of light waves carrying angular momentum with them [7]. Two types of momenta contribute to the total angular momentum of light – *Spin Angular Momentum (SAM)* due to the circular polarization of light and *Orbital Angular Momentum (OAM)* due to the twisting of the wavefield while propagating. While the SAM has been well investigated throughout the 20th century, the detailed study of OAM is fairly recent, with major groundwork done by Allen *et al.* [3]. They proposed that for elliptically polarized light (resulting in a helical phase front) with phase profile corresponding to $e^{il\phi}$, the OAM is equal to $l\hbar$ per photon. Here, l is the topological charge, ϕ is the phase and $\hbar$ is the reduced Planck constant. The OAM has since found applications in multiple forms in the fields of optical trapping and manipulation of microparticles [8-10]. Other applications have been developments in remote sensing and wireless communication [11] as well as exploring the information content of optical vortices [12, 13].

1.3.3 Ubiquity

In section 1.2.1, we have discussed the origin of the term "optical vortices" from the general expression Eq. (1.12) of the amplitude of Laguerre-Gauss (LG) beams. These LG beams have been extensively studied to intrinsically contain optical vortices within them. In other words, it should be noted here that the generation of optical vortices does not require special

circumstances. It is a phenomenon that accompanies any interference of randomly scattered plane waves [14]. However, experimentally, it has been identified that helically phased beams play a vital role in the formation of phase singularities [15]. These beams, particularly the Laguerre- Gaussian mode (LG_{01}) have been shown [16] to be produced from a combination of Hermite-Gaussian modes HG_{01} and HG_{10}. An alternative method of generation was demonstrated [17] in 1990 where a diffractive element such as a grating with a dislocation was used to obtain a helically phased mode from the Gaussian beam of a laser. Subsequently, the generation of optical vortex beams has also been achieved using spiral phase plates [18], hologram [19] and spatial light modulators [20].

1.4 Detection of Optical Vortices

The main idea behind detecting the presence of an optical vortex is to be able to quantify the phase in a coherent light field. A brief review of some of the commonly applied methods is presented in this section.

1.4.1 Spatial Phase Filtering

In the 1980s, Golub [21, 22] demonstrated the possibility of using spatial filters to detect the phase of wavefields. Initially, these filters had pre-designed transmissions [22] to separate specific transverse modes. They were computer-synthesized and fabricated by precision systems. Transparencies were the most common spatial filters used in these experiments. A phase tilt was introduced for each individual mode in multimode beams. It was demonstrated that they accurately determined the transverse modes composition of

8

these beams [22]. Improvements in terms of higher selectivity of the modes as well as better energy efficiency were later made by the same group [23].

1.4.2 Interferometry

The next significant advancement in the detection of phase singularities was based on the observation of interference patterns arising from them. These were first reported for helical modes with topological charges up to 3 [24]. It was noted that the interference patterns arising from each mode corresponds to two possible states of opposite helicity, i.e., same topological charge, but opposite signs.

1.4.3 Shack-Hartmann Sensors

A Shack-Hartmann wavefront sensor is a multi-lens system used to characterize the shape of a wavefront, and has been in use in adaptive optics for a few decades [25]. The most common method is to perform a least-square reconstruction of the phase gradient, from the wavefront data obtained [25, 26]. However, performance issues have been reported in the presence of strong scintillations [27]. These issues have been attributed to optical vortices. As the phase is undefined at these locations, phase gradient calculations produced anomalies in the wavefront measurements. In the early 2000s, Chen *et al.* [28] demonstrated that suitable mathematical modifications of the sensor data can be used to obtain the location and the topological charge of these singularities.

9

1.4.4 Diffraction-based methods

In 2010, Hickmann *et al.*[29] noted that helical beams diffracted by triangular apertures produced triangular lattices in the far field, which could be used to calculate the topological charge of the beam. This work included an experimental demonstration up to a charge of 3. This method was further developed and expanded to a topological charge up to 7 soon [30]. Further work has since been done with a variety of apertures such as annular triangle [31], annular ellipse [32] and diamond-shaped [33] apertures. The essence of all of these methods lies in the observation that the number of diffraction spots (maxima) in the resultant pattern indicated the magnitude of the charge while the orientation of the pattern indicated the sign.

1.4.5 Automated Mode separation

A key aspect of the spatial phase filtering approach mentioned above was that the desired composition of the modes to be extracted had to be encoded within the design of the filter. This requirement was removed by an improvement by Berkhout *et al.* [34] who used spatial light modulators to separate each mode of a helically phased beam to individual lines on a detector.

1.5 Summary

Dislocations in wavefronts were first reported in the early 1970s [35] and have been a subject of investigation for the past few decades, under terms such as phase singularities or optical vortices. This chapter presented a discussion of why they are present in all coherent light fields, without the need of special circumstances. This was followed by a review of how the presence of these singularities are detected. In Chapter 3, we introduce another method for identifying the locations of these singularities, which uses a pseudo-phase representation of the field. This method is used further in this dissertation work, and is thus treated with higher details.

In chapter 2, we take a detour form our discussion of optical vortices, to introduce a mathematical formulation that has been used in the subsequent work presented in this dissertation.

1.6 References

[1] D.J. Griffiths, *Introduction to Electrodynamics*, Prentice Hall [1981].

[2] K.F. Riley, M. P. Hobson, and S. J. Bence, *Mathematical Methods for Physics and Engineering: A Comprehensive Guide*, 3rd edition, Cambridge University Press [2006].

[3] L. Allen, M. W. Beijersbergen, R. J. C. Spreeuw, and J. P. Woerdman, "Orbital angular momentum of light and the transformation of Laguerre-Gaussian laser modes", Physical Review A 45 (11), pages 8185-8190 [1992].

[4] A. Ghatak, *Optics*, 3rd edition, Tata McGraw-Hill Education [2005].

[5] A. M. Yao, and M. J. Padgett, "Orbital angular momentum: origins, behavior and applications", Advances in Optics and Photonics 3, pages 161-204 [2011].

[6] M.V. Berry, and M.R. Dennis, "Phase singularities in isotropic random waves", Proceedings of the Royal Society of London A 456, pages 2059-2079 [2000].

[7] J. H. Poynting, "The Wave Motion of a Revolving Shaft, and a Suggestion as to the Angular Momentum in a Beam of Circularly Polarised Light", Proceedings of the Royal Society of London A 82, pages 560-567 [1909]

[8] H. He, M. E. J. Friese, N. R. Heckenberg, and H. Rubinsztein-Dunlop, "Direct observation of transfer of angular momentum to absorptive particles from a laser beam with a phase singularity", Physical Review Letters 75 (5), pages 826-829 [1995].

[9] N. Simpson, K. Dholakia, L. Allen, and M. Padgett, "Mechanical equivalence of spin and orbital angular momentum of light: an optical spanner", Optics Letters 22 (1), pages 52-54 [1997].

[10] D. Grier, "A revolution in optical manipulation", Nature 42, pages 810-816 [2003].

[11] B. A. Knyazev, Y. Y. Choporova, V. S. Pavelyev, N. D. Osintseva, and B. O. Volodkin, "Transmission of high-power terahertz beams with orbital angular momentum

through atmosphere", International conference on Infrared, Millimeter, and Terahertz Waves, 7758816 [2016].

[12] G. S. Agarwal and J. Banerji, "Spatial coherence and information entropy in optical vortex fields", Optics Letters 27 (10), pages 800-802 [2002].

[13] A. Kumar, S. Prabhakar, P. Vaity, and R. P. Singh, "Information content of optical vortex fields", Optics Letters 36 (7), pages 1161-1163 [2011].

[14] K. O'Holleran, M. Padgett, and M. Dennis, "Topology of optical vortex lines formed by the interference of three, four and five plane waves", Optics Express 14 (7), pages 3039-3044 [2006].

[15] M. Brambilla, M. Cattaneo, L. A. Lugiato, V. Penna, F. Prati, C. Tamm, and C. O. Weiss, "Transverse laser patterns. I. Phase singularity crystals", Physical Review A 43, pages 5090-5113 [1991].

[16] J. M. Vaushan, D. Willetts, "Interference properties of a light-beam having a helical wave surface", Optics Communications 30 (3), pages 263-267 [1979]

[17] V. Bazhenov, M. V. Vasnetsov, and M. S. Soskin, "Laser-beams with screw dislocations in their wave-fronts", Journal Experimental and Theoretical Physics Letters 52, pages 429-431 [1990].

[18] M. W. Beijersbergen, L. Allen, H. van der Veen, and J. P. Woerdman, "Astigmatic laser mode converters and transfer of orbital angular-momentum", Optics Communications 96, pages 123-132 [1993].

13

[19] N. R. Heckenberg, R. McDuff, C. P. Smith, and A. G. White, "Generation of optical phase singularities by computer-generated holograms", Optics Letters 17 (3), pages 221-223 [1992].

[20] J. E. Curtis, B. A. Koss, and D. G. Grier, "Dynamic holographic optical tweezers", Optics Communications 207, pages 169-175 [2002].

[21] M. A. Golub, A. M. Prokhorov, I. N. Sisakyan, and V. A. Soifer, "Synthesis of spatial filters for investigation of the transverse mode composition of coherent radiation", Soviet Journal of Quantum Electronics 12 (9), pages 1208-1209 [1982].

[22] M. A. Golub, S. V. Karpeev, S. G. Krivoshlykov, A. M. Prokhorov, I. N. Sisakyan, and V. A. Soifer, "Experimental investigation of spatial filters separating tranverse modes of optical fields", Soviet Journal of Quantum Electronics 13 (2), pages 1123-1124 [1983].

[23] M. A. Golub, S. V. Karpeev, N. L. Kazanskii, A. V. Mirzov, I. N. Sisakyan, V. A. Soifer, and G. V. Uvarov, "Spatial phase filters matched to transverse modes", Soviet Journal of Quantum Electronics 18 (3), pages 392-393 [1988].

[24] M. Harris, C. A. Hill, P. R. Tapster, and J. M. Vaughan, "Laser modes with helical wave fronts", Physical Review A 49 (4), pages 3119-3122 [1994].

[25] D. L. Fried, "Least-square fitting a wave-front distortion estimate to an array of phase-difference measurements," Journal of the Optical Society of America 67 (3), pages 370-375 [1977].

[26] R. H. Hudgin, "Wave-front reconstruction for compensated imaging," Journal of the Optical Society of America 67 (3), pages 375-378 [1977].

[27] J. D. Barchers, D. L. Fried, and D. J. Link, "Evaluation of the performance of Hartmann sensors in strong scintillation", Applied Optics 41 (6), pages 1012-1021 [2002].

[28] M. Chen, F. S. Roux, and J. C. Olivier, "Detection of phase singularities with a Shack-Hartmann wavefront sensor", Journal of the Optical Society of America A 24 (7), pages 1994-2002 [2007].

[29] J. M. Hickmann, E. J. S. Fonseca, W. C. Soares, and S. Chávez-Cerda, "Unveiling a Truncated Optical Lattice Associated with a Triangular Aperture Using Light's Orbital Angular Momentum", Physical Review Letters 105, 053904 [2010].

[30] L. E. E. de Araujo, and M. E. Anderson, "Measuring vortex charge with a triangular aperture", Optics Letters 36 (6), pages 787-789 [2011].

[31] L. Yongxin, T. Hua, P. Jixiong, and L. Baida, "Detecting the topological charge of vortex beams using an annular triangle aperture", Optics & Laser Technology 43, pages 1233-1236 [2011].

[32] H. Tao, Y. Liu, Z. Chen, and J. Pu, "Measuring the topological charge of vortex beams by using an annular ellipse aperture", Applied Physics B 106, pages 927-932 [2012].

[33] Y. Liu, S. Sun, J. Pu, and B. Lu, "Propagation of an optical vortex beam through a diamond-shaped aperture", Optics & Laser Technology 45, pages 473-479 [2013].

[34] G. C. G. Berkhout, M. P. J. Lavery, J. Courtial, M. W. Beijerbergen, and M. J. Padgett, "Efficient Sorting of Orbital Angular Momentum States of Light", Physical Review Letters 105, 153601 [2010].

15

[35] J. F. Nye, and M. V. Berry, "Dislocations in wave trains", Proceedings of the Royal Society of London A 336, pages 165-190 [1974].

2 Theory of Poincaré Analysis

We introduce a novel approach to using dynamic laser speckle data to quantify biophysical dynamics including ordered flows and random motions. The approach yields images that superficially resemble traditional laser speckle contrast images, but instead of relying on the statistics of the local time integrated intensity values calculated over temporal and sliding spatial windows as is done in laser speckle contrast imaging (LSCI) to create images, this approach – which we term "Ellipticity Imaging (EI)" - directly yields images that quantify the relative dominance of long-range correlations in the temporal dimension of a series of speckle patterns to the short-range correlations in the same dimension. The approach relies on a Poincaré analysis of the speckle data which yields metrics that statistically describe both the short-terms variations in the temporal speckle intensity (i.e., the standard deviation in successive differences) and also the corresponding long term variations. These metrics are plotted against each other (Poincaré plots) and an ellipse fit to the data. The ratio of the semi-major axis to the semi-minor axis of this ellipse for each temporal speckle sequence is then used as the data to form the images (thus the term EI). The theory of EI is presented. The application of this approach to speckle data obtained from simulations, phantom studies and animal studies are presented in chapter 5 of this dissertation. Though not fundamental for understanding the work in this chapter, some of the parameters described here can be calculated in a MATLAB platform using the code in Appendix B.4.

17

2.1 Introduction

When coherent light is scattered from a rough surface or scattering volume (such as biological tissues), granular interference patterns are produced. These are known as speckles [1]. If the scattering sample is dynamic, then the resultant speckle patterns evolve with time. These patterns have been the subject of extensive research that have led to a large number of applications exploiting speckle metrology [2] and interferometry [3]. Prominent among these applications is Laser Speckle Contrast Imaging (LSCI), which has developed over the last two decades from the early work of Briers and Webster [4].

Poincaré descriptors are a set of statistical measures which have been previously used to study heart rate variations [5]. In this paper, the underlying mathematical principles behind the relation between these Poincaré descriptors and correlation functions is first discussed. These correlation functions form the basis of analyzing the spatio-temporal behavior of evolving speckle fields. This is followed by a discussion on the potential for these descriptors to be used for imaging, similar to current LSCI methods.

2.2   Mathematical background

2.2.1 Relating the Standard Descriptors and the Autocorrelation Function

Let us consider a data series X, with elements indicated by X_1, X_2,….etc. For this series X, the autocorrelation function is given by

18

$$\gamma(k) = \overline{X_i X_{i+k}} = \frac{1}{N-k} \sum_{i=1}^{N-k} X_i X_{i+k} \qquad (2.1)$$

where X_i is the ith element of the series X, and N is the total number of elements in the series. The variable k is commonly called a "lag", making the above definition the *lag-k* autocorrelation function. The overline denotes taking the mean.

Clearly,

$$\gamma(0) = \overline{X_i^2} \qquad (2.2)$$

gives the mean-squared value of the data series.

For the same data series, the autocovariance function is given by

$$\phi(k) = \overline{(X_i - \mu)(X_{i+k} - \mu)} = \frac{1}{N-k} \sum_{i=1}^{N-k} (X_i - \mu)(X_{i+k} - \mu) \qquad (2.3)$$

where μ is the mean of the data series. Clearly,

$$\phi(0) = \overline{(X_i - \mu)^2} = \sigma^2 \qquad (2.4)$$

gives the variance of the series. Here, σ represents the standard deviation in the series.

We note that

$$\begin{aligned}
\phi(k) &= \frac{1}{N-k} \sum_{i=1}^{N-k} (X_i - \mu)(X_{i+k} - \mu) \\
&= \frac{1}{N-k} \sum_{i=1}^{N-k} (X_i X_{i+k} - \mu(X_i + X_{i+k}) + \mu^2) \\
&= \frac{1}{N-k} \sum_{i=1}^{N-k} X_i X_{i+k} - \frac{1}{N-k} \mu \sum_{i=1}^{N-k} (X_i + X_{i+k}) + \mu^2
\end{aligned} \qquad (2.5)$$

19

We introduce the "first error function" ε_1 such that

$$\varepsilon_1(k) = 2\mu^2 - \frac{1}{N-k}\mu\sum_{i=1}^{N-k}(X_i + X_{i+k}) \qquad (2.6)$$

This reduces Eq. (2.5) to

$$\phi(k) = \gamma(k) + \varepsilon_1(k) - \mu^2 \qquad (2.7)$$

Now, the standard descriptor *SD1* is defined as [5]

$$SD1(k) = \{\frac{1}{2}Var(X_i - X_{i+k})\}^{\frac{1}{2}} \qquad (2.8)$$

where *Var(X)* denotes the variance in the series *X*.

Thus,

$$\begin{aligned}
2SD1(k)^2 &= Var(X_i - X_{i+k}) \\
&= \overline{(X_i - X_{i+k})^2} - \overline{(X_i - X_{i+k})}^2 \\
&= \frac{1}{N-k}\sum_{i=1}^{N-k}(X_i - X_{i+k})^2 - \varepsilon_2(k)
\end{aligned} \qquad (2.9)$$

where we have introduced the "second error function" ε_2 such that

$$\varepsilon_2(k) = \overline{(X_i - X_{i+k})}^2 \qquad (2.10)$$

We also note that for *k=1*, the first term of the right side in Eq. (2.9) becomes what is known in statistics as the "Mean Squared Successive Differences (MSSD)". We shall henceforth denote the *k-lagged MSSD* as

20

$$MSSD(k) = \frac{1}{N-k} \sum_{i=1}^{N-k} (X_i - X_{i+k})^2 \qquad (2.11)$$

We expand Eq. (2.11) to get

$$MSSD(k) = \frac{1}{N-k} \sum_{i=1}^{N-k} (X_i - X_{i+k})^2 = \frac{1}{N-k} \sum_{i=1}^{N-k} (X_i^2 + X_{i+k}^2) - \frac{2}{N-k} \sum_{i=1}^{N-k} X_i X_{i+k}$$
$$= 2\{\overline{X_i^2} - \gamma(k)\} + \varepsilon_3(k) \qquad (2.12)$$

where the "third error function" ε_3 is introduced such that

$$\varepsilon_3(k) = \frac{1}{N-k} \sum_{i=1}^{N-k} (X_i^2 + X_{i+k}^2) - 2\overline{X_i^2} \qquad (2.13)$$

Substituting Eq. (2.12) into (2.9) gives

$$SD1(k)^2 = \overline{X_i^2} - \gamma(k) + \frac{1}{2}\{\varepsilon_3(k) - \varepsilon_2(k)\} \qquad (2.14)$$

Substituting expressions from Eq. (2.2) and Eq. (2.7) into the above Eq. (2.14) gives

$$SD1(k)^2 = \phi(0) - \phi(k) + \varepsilon_1(k) + \frac{1}{2}\{\varepsilon_3(k) - \varepsilon_2(k)\} \qquad (2.15)$$

The other standard descriptor *SD2* is defined as

$$SD2(k) = \{2Var(X) - SD1(k)^2\}^{\frac{1}{2}} \qquad (2.16)$$

Thus,

$$SD2(k)^2 = 2\sigma^2 - SD1(k)^2 = \phi(0) + \phi(k) - \varepsilon_1(k) - \frac{1}{2}\{\varepsilon_3(k) - \varepsilon_2(k)\} \quad (2.17)$$

21

We define "ellipticity" e as

$$e(k) = \frac{SD2(k)}{SD1(k)} \tag{2.18}$$

Introducing an "overall error function"

$$\varepsilon(k) = \varepsilon_1(k) + \frac{1}{2}\{\varepsilon_3(k) - \varepsilon_2(k)\} \tag{2.19}$$

makes the expression for ellipticity

$$e(k) = \left\{\frac{SD2(k)^2}{SD1(k)^2}\right\}^{\frac{1}{2}} = \left[\frac{\phi(0) + \{\phi(k) - \varepsilon(k)\}}{\phi(0) - \{\phi(k) - \varepsilon(k)\}}\right]^{\frac{1}{2}} \tag{2.20}$$

Dividing the numerator and denominator in Eq. (2.20) by the variance gives

$$e(k) = \left[\frac{1 + \{\frac{\phi(k)}{\sigma^2} - \Delta_k\}}{1 - \{\frac{\phi(k)}{\sigma^2} - \Delta_k\}}\right]^{\frac{1}{2}} \tag{2.21}$$

where

$$\Delta_k = \frac{\varepsilon(k)}{\sigma^2} \tag{2.22}$$

is the ratio of the error function to the variance of the data series. From here on, this ratio

is referred to as the "normalized error function". We also note that

$$\frac{\phi(k)}{\sigma^2} = \frac{\frac{1}{N-k}\sum_{i=1}^{N-k}(X_i - \mu)(X_{i+k} - \mu)}{\frac{1}{N}\sum_{i=1}^{N}(X_i - \mu)^2} = \frac{N}{N-k}r_k \tag{2.23}$$

22

where r_k is the lag-k autocorrelation coefficient. It can be noted that when the ellipticity value is close to 1.0, the autocorrelation coefficient is close to the value of the normalized error function (Δ_k) .In fact, since k is always a positive integer, $r_k < \Delta_k$ if ellipticity is *exactly* 1.0. It has been observed that the error function is in the range of 10^{-4}-10^{-5}. Thus, this condition on the ellipticity corresponds to an extremely small value for the autocorrelation coefficient. In other words, there is no correlation in the data. If the data series in concern is the spatial intensity distribution in a speckle field, this makes the value of lag where ellipticity is close to 1.0, an indicator of the speckle intensity correlation length.

2.2.1.1 Use of the term "Ellipticity"

As described by Brennan *et al* [5], traditionally, Poincaré plots are constructed by plotting the data point X_{i+1} against the preceding data point X_i. This results in a scatterplot as shown by the red points in Fig 2.1.

23
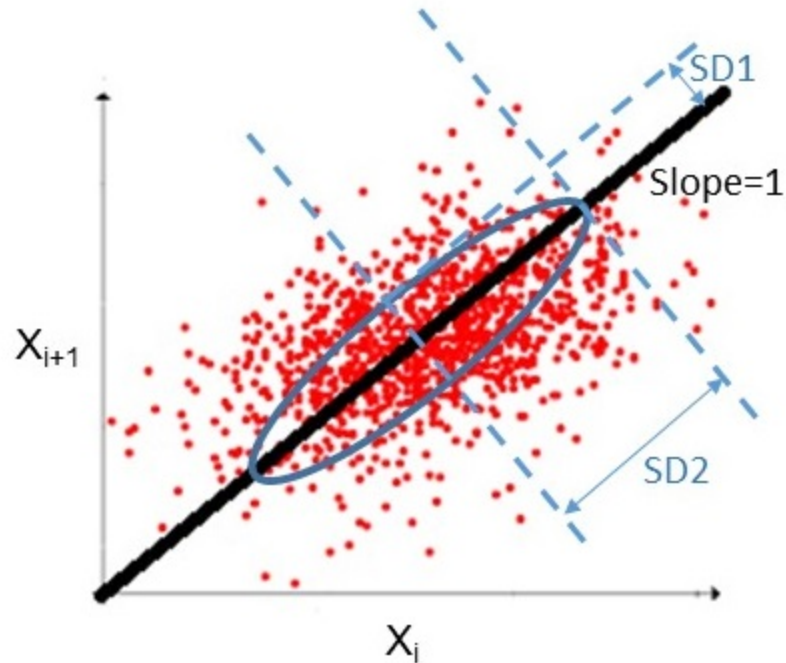
Fig 2.1. A sample Poincaré plot. The scatter plot consists of each data point plotted against its predecessor. The ellipse is drawn centered on the line-of-identity with its semi-axes equal to the values of the Poincaré descriptors

Over this scatter plot, the line-of-identity with slope equal to unity is drawn. Centered on this line, an ellipse is constructed with semi-axis along the line-of-identity equal to SD2 and semi-axis perpendicular to the line-of-identity equal to SD1. Thus, as the ratio between SD2 and SD1 determines the shape of the ellipse in the traditional Poincaré plot, we have used the term "ellipticity" to define this imaging technique. It is to be noted that while dealing with k-lag definitions, the y-axis of Fig 2.1 should be X_{i+k} instead of X_{i+1}.

2.2.1.2 Notes

24

1. The factor of $(N-k)^{-1}$ outside the summations in definitions Eq. (2.1) and Eq. (2.3) leads to what are called the "unbiased" definitions of these functions. Alternate definitions, called the "biased" versions of these functions exist in statistics, in which this factor of $(N-k)^{-1}$ is replaced by N^{-1}. If these biased definitions are used for these functions, the right-hand side of Eq. (2.23) reduces to exactly the autocorrelation function. In other words,

$$\frac{\phi(k)}{\sigma^2} = r_k \qquad (2.24)$$

2. If the data series X is considered "wide-sense stationary" (true for speckle intensity variations as the Wiener-Khinchin Theorem is widely accepted to hold [6]), then the second error function is equal to 0. In practice, this number is observed to range from 10^{-5} to 10^{-12}. Using this condition in Eq. (2.9) and Eq. (2.11) leads to

$$SD1(k)^2 \approx \frac{MSSD(k)}{2} \qquad (2.25)$$

We recall that MSSD(k) is the k-lagged mean squared successive differences. Thus, $SD1(k)$ is a measure of the variation in the data from one point (say, X_i) to the next k-lagged point (X_{i+k}). Thus, $SD1(k)$ gives a sense of the *short-term variations* in the data. Additionally, from Eq. (2.16), we see that the standard descriptor $SD2(k)$ is defined in terms of the difference between the variance and $SD1(k)$. As the variance is a notation of the general spread in the data, and $SD1(k)$, as discussed above, signifies the short-term variations in the data, $SD2(k)$ thus gives a sense of the remaining contribution to the spread in the data.

25

We denote this as the *long-term variations* in the data. Also, including the approximation of Eq. (2.25) into the definitions Eq. (2.16) and Eq. (2.18), leads to

$$SD2(k)^2 \approx 2\sigma^2 - \frac{MSSD(k)}{2}$$

$$e(k)^2 \approx \frac{4\sigma^2}{MSSD(k)} - 1 \tag{2.26}$$

Thus, ellipticity is a direct measure of the ratio of the variance to the lag-k mean squared successive difference.

2.2.2 Relation to Laser Speckle Contrast Imaging (LSCI)

In section 2.2.1, we noticed how the ellipticity values indicate the autocorrelation coefficient values, and thus in case of the spatial distribution of intensity in a speckle field, they give the correlation length (i.e. speckle size). If the data series represents the intensities of a single speckle at different points in time, then Eq. (2.3) gives the autocovariance of the temporal intensity fluctuations in that speckle. This has been related to the variance of speckle field by [6]

$$\sigma(T)^2 = \frac{2}{T} \int_0^T \left(1 - \frac{\tau}{T}\right) \phi(\tau) d\tau \tag{2.27}$$

where T is the integration time of the camera in speckle imaging settings. It is to be noted here that this expression is a corrected form of a previously used expression [7] by the same author. It was noted by Boas and Dunn [8] that the difference between the initial and the updated expressions for biomedical applications had little significance, as integration time

26

is much longer than the coherence time of the observed flow. The *biased* (see point 1 in Section 2.2.1.2) autocorrelation function, from Eq. (2.7) then becomes

$$\gamma(\tau) = \phi(\tau) + \mu^2 \tag{2.28}$$

Dividing by the square of the mean gives the normalized definitions of these functions:

$$g_2(\tau) = 1 + c(\tau) \tag{2.29}$$

where $c(\tau) = \dfrac{\phi(\tau)}{\mu^2}$ is the normalized temporal intensity autocovariance function and

$g_2(\tau) = \dfrac{\gamma(\tau)}{\mu^2}$ is the normalized temporal intensity autocorrelation function. The subscript

'2' indicates that this is a second-order function, related to the first-order temporal electric field autocorrelation function $g_1(\tau)$ using the Siegert relation [9] :

$$g_2(\tau) = 1 + \beta \mid g_1(\tau) \mid^2 \tag{2.30}$$

where β is a geometric parameter accounting for loss of correlation normalized). Various models for $g_1(\tau)$ have been proposed with multiple additions and/or modifications [6] including discussions on different models for Lorentzian (considered appropriate for dense and convoluted circulation such as capillary flow) and Gaussian (large blood vessels) flow models or a combination of both.

Now, from Eq. (2.20), we have

$$\phi(k) - \varepsilon(k) = \frac{e(k)^2 - 1}{e(k)^2 + 1}\phi(0) = \frac{e(k)^2 - 1}{e(k)^2 + 1}\sigma^2 \tag{2.31}$$

27

Considering the data to be the temporal intensity fluctuations in a speckle, all the statistical definitions in Eq. (2.31) are for this temporal variation. To denote this, and to distinguish the variance symbol from the variance in spatial intensity indicated by the left-hand side of Eq. (2.27), we rewrite Eq. (2.31) as

$$\phi(\tau) = \left(\frac{e(\tau)^2 - 1}{e(\tau)^2 + 1} \right) \sigma_\tau^2 + \varepsilon(\tau) \tag{2.32}$$

This substituted into Eq. (2.27) gives the expression for the spatial variance in terms of the ellipticity and temporal variance as

$$\sigma_s(T)^2 = \frac{2}{T} \int_0^T \left(1 - \frac{\tau}{T} \right) \left(\left(\frac{e(\tau)^2 - 1}{e(\tau)^2 + 1} \right) \sigma_\tau^2 + \varepsilon(\tau) \right) d\tau \tag{2.33}$$

where the subscripts s and τ are used to distinguish the spatial and temporal variations in the intensity fluctuations.

2.2.2.1 Potential advantages of these descriptors

One of the most interesting differences in the two methods is the inherent importance of the *MSSD (k)* value in ellipticity measurement. Potentially, this makes any rapid short-term temporal changes in intensity appear more prominently in the ellipticity value, whereas that might be getting blurred out in the temporal autocorrelation (or autocovariance) values used in contrast imaging. We might need to look at actual LSCI data for confirmation/elimination of this hypothesis. We need to be conscious of the fact that at its

28

core, the ellipticity value is a measure of the relative prominence of long and short-term variations in the data.

2.2.3 A discussion on the first and third error functions

We note that the first error function was defined as

$$\varepsilon_1(k) = 2\mu^2 - \frac{1}{N-k}\mu\sum_{i=1}^{N-k}(X_i + X_{i+k}) \tag{2.34}$$

The terms can be re-arranged as below

$$\begin{aligned}
\varepsilon_1(k) &= \mu\left[2\mu - \frac{1}{N-k}\sum_{i=1}^{N-k}(X_i + X_{i+k})\right] \\
&= \mu\left[\left\{\mu - \frac{1}{N-k}\sum_{i=1}^{N-k}X_i\right\} + \left\{\mu - \frac{1}{N-k}\sum_{i=1}^{N-k}X_{i+k}\right\}\right]
\end{aligned} \tag{2.35}$$

We note that the expression in the first curly brackets is simply the difference between the overall mean and the mean of the *first (N-k) terms*, while the expression in the second curly brackets is the difference between the overall mean and the mean of the *last (N-k) terms*.

Similarly, copying from Eq. (2.13) the expression for the third error function:

$$\varepsilon_3(k) = \frac{1}{N-k}\sum_{i=1}^{N-k}(X_i^2 + X_{i+k}^2) - 2\overline{X_i^2} \tag{2.36}$$

The terms can be re-arranged to

$$\varepsilon_3(k) = \left\{\frac{1}{N-k}\sum_{i=1}^{N-k}X_i^2 - \overline{X_i^2}\right\} + \left\{\frac{1}{N-k}\sum_{i=1}^{N-k}X_{i+k}^2 - \overline{X_i^2}\right\} \tag{2.37}$$

29

We note that the expression in the first curly brackets is the difference between the mean square of the *first (N-k) terms* and the overall mean squared value, while the expression in the second curly brackets is the difference between the mean square of the *last (N-k) terms* and the overall mean squared value.

2.3 Summary

In this chapter, the mathematical background for Poincaré analysis is presented. For future consideration, a couple of points to note in this analysis would be:

- We can always choose to select biased estimators instead of the currently used unbiased ones. As in Eq. (2.24), this will give a direct approach for the autocorrelation coefficient.

- In any case, the ellipticity value is seen to be linearly varying with the ratio $\dfrac{\sigma^2}{MSSD(k)}$. Thus, tracking this ratio alone gives just as much information as the ellipticity values.

It is worthwhile to note that most of the material is presented without any specific consideration to speckle phenomena. Thus, this form of analysis can be used in other areas concerned with correlations among measured quantities.

The specific application to speckle phenomena and the results obtained from such applications are discussed in chapter 5. Comparisons with existing methods for quantifying

these phenomena are also made. Application to certain aspects of optical vortex dynamics is also presented in chapters 3 and 4.

2.4 References

[1] J. W. Goodman, *Speckle Phenomena in Optics: Theory and Applications*, Roberts & Company [2007].

[2] R. K. Erf, *Speckle Metrology*, Academic Press, New York [1978].

[3] J. A. Leendertz, "Interferometric displacement measurement on scattering surfaces utilizing speckle effect", Journal of Physics E: Scientific Instruments 3, pages 214-218 [1970].

[4] J. D. Briers, and S. Webster, "Laser Speckle Contrast Analysis (LASCA): A nonscanning, full-field technique for monitoring capillary blood flow", Journal of Biomedical Optics 1 (2), pages 174-179 [1995].

[5] M. Brennan, M. Palaniswami, and P. Kamen, "Do Existing Measures of Poincaré Plot geometry reflect Nonlinear Features of Heart Rate Variability?," IEEE Transactions on Biomedical Engineering 48 (11), pages 1342-1347 [2001].

[6] J. W. Goodman, *Statistical Optics*, 2nd edition, Wiley [1985].

31

[7] J. W. Goodman, "Some effects of target-induced scintillations on optical radar performance," Proceedings of IEEE 53 (11), pages 1688–1700 [1965].

[8] D. A. Boas, and A. K. Dunn, "Laser speckle contrast imaging in biomedical optics," Journal of Biomedical Optics 15 (1), 011109 [2010].

[9] P. A. Lemieux, and D. J. Durian, "Investigating non-Gaussian scattering processes by using nth–order intensity correlation functions," Journal of the Optical Society of America A 16 (7), pages 1651–1654 [1999].

32

3 Statistical studies on Optical Vortices in simulated Speckle fields

In chapter 1, we mentioned some advancements in the detection of optical vortices. In this chapter, we introduce another method that uses a pseudo-phase representation of an intensity field. We detect the presence of optical vortices with this method. We then relate the time evolution of the spatial distribution of these vortices to the dynamic systems that lead to their creation. Excerpts of this chapter have been taken, with permission, from "Statistical studies on optical vortices in dynamic speckle fields" by A. Majumdar and S.J. Kirkpatrick, published in the Journal of Biomedical Photonics and Engineering in 2017. The letter of permission is attached in Appendix A.2. The calculations in this chapter have been performed in a MATLAB environment using the code in Appendix B.1.

3.1 Speckle fields and Optical Vortices

Random optical interference gives rise to a granular pattern with bright and dark regions called speckle. Light scattering techniques, including those involving laser speckle analysis have been used to study dynamic biological systems over the last several decades [1-4]. Speckle based imaging techniques have seen tremendous advancements since the 1990s and are now well-established in the field of biomedical optics [5-7]. As mentioned in chapter 1, optical vortices are ubiquitous in a coherent field and do not require special circumstances to be formed. Within speckle fields are also locations of phase discontinuity where the intensity of the field is zero and the phase is undefined. Such phase singularities were first described by Nye and Berry [8]. A few years later they reported an analysis of

phase singularity lines within fields of multiple beam interference [9]. Although the existence of these phase singularities have been known since the 1970s, their in-depth study in speckle fields [10] has garnered interest much more recently.

While the key properties of optical vortices were highlighted in section 1.3, it would be appropriate here to re-iterate the presence of topological charges, which were introduced in section 1.3.1. Key aspects of topological charges are used in our methods to detect optical vortex locations within speckle fields.

Speckle fields, $\Psi(\mathbf{r},t)$, like other electromagnetic fields, can be written as a general solution of the Helmholtz equation as

$$\psi(\boldsymbol{r},t) = A(\mathbf{r},t)e^{i\phi(\boldsymbol{r},t)}, \tag{3.1}$$

where $A(\mathbf{r},t)$ is the amplitude and $\phi(\boldsymbol{r},t)$ is the phase of the field, both at location *r* and time *t*. The real and imaginary parts of this speckle field can be separated, with phase singularities occurring in locations where both the real and imaginary parts are zero (Fig 3.1).
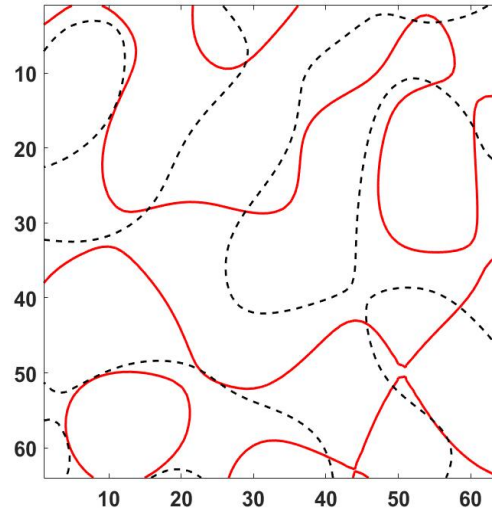
Fig 3.1. Zero-contours of real and imaginary field components. The red (solid) lines represent where $\text{Re}(\psi) = 0$ and the black (broken) lines represent where $\text{Im}(\psi) = 0$. The points of their intersection are the locations of phase singularities.

In the immediate vicinity of and surrounding these singular points, the optical phase rotates in either a clockwise or counter-clockwise direction over at least a full 2π radians and the structures are subsequently referred to as either negatively charged or positively charged optical vortices, respectively [11]. The number of 2π rotations of the phase is known as the topological charge of the vortex, expressed mathematically as

$$n_t = \frac{1}{2\pi} \oint_c \nabla \phi(x,y) dl, \tag{3.2}$$

where the line integral is along a closed loop, l, around the vortex. Optical vortices always occur in pairs of opposite topological charge, and as a consequence of conservation of

35

charge, there must always be an equal number of positively and negatively charged vortices in the scattered field.
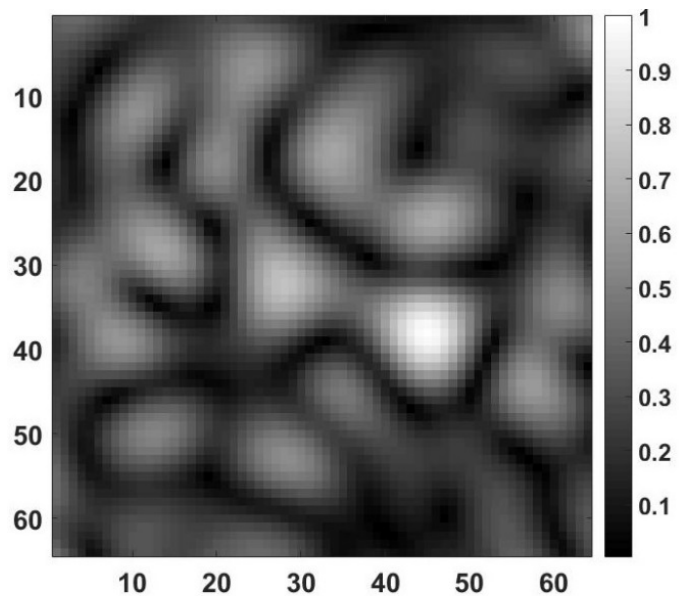
3.2 Tracking Optical Vortices

3.2.1 Simulation of Speckle fields

Sequences of 50 speckle patterns with user-defined correlation behaviors were simulated using the concept of a copula [12] in the MATLAB environment. Decorrelation rates are defined in terms of the number of frames it takes for the overall speckle pattern autocorrelation coefficient to reduce to a value of $1/e$. Three different rates of decorrelation behavior between the frames were studied, with their autocorrelation coefficients falling to $1/e$ in 11, 6 and 3 frames, during the 50 frame simulation. Herein, they shall be referred to as the slow, medium and fast rates of decorrelation, respectively. For each of the rates, three different decorrelation behaviors (or modes) were studied. One mode was of constant sequential autocorrelation coefficient, Γ ($\Gamma = 0.9984$ for slow decorrelation rate; $\Gamma = 0.99$ for medium decorrelation rate and $\Gamma = 0.96$ for fast decorrelation rate) between frames. The second behavior was one that displayed a Gaussian decorrelation line shape. This decorrelation behavior has generally been understood to be caused by ordered dynamics [13]. The third behavior examined, Lorentzian decorrelation, is linked to Brownian dynamics. In general, biological systems likely exhibit a combination of ordered and Brownian (or disordered) dynamics.

36

3.2.2 Detecting Vortex locations in Speckle fields

Once the 50 speckle patterns were generated for each decorrelation behavior studied, we created a pseudo-phase representation for each of the patterns using a two-dimensional Hilbert filter (Fig 3.2(b)) [10]. The location of optical vortices in the speckle were identified and tracked on a frame-by-frame basis through these sequences (Fig 3.2(c)).
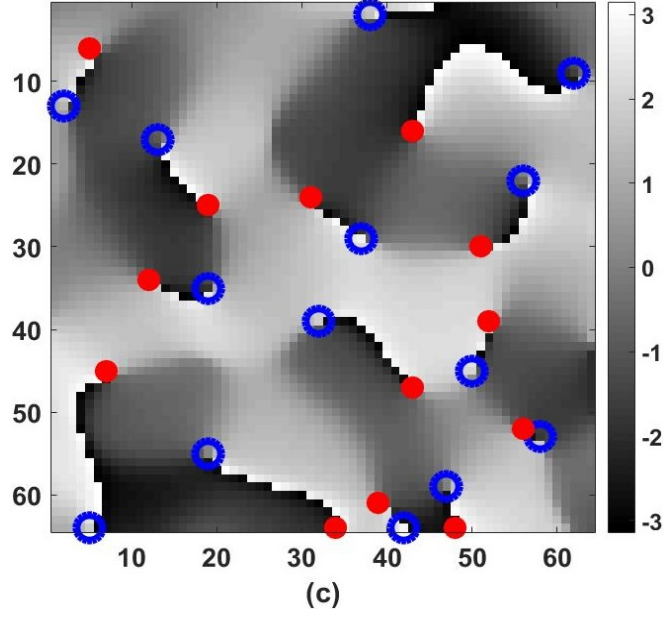

(a)

(b)

37
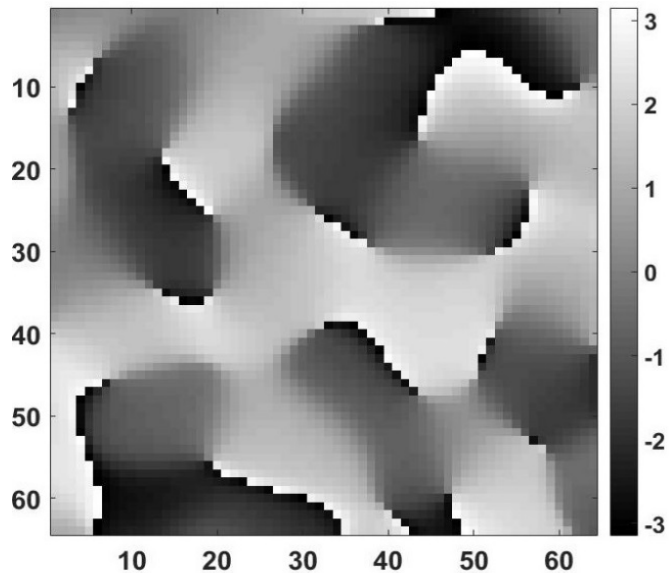
Fig 3.2. Locating optical vortices in a speckle field. (a) A speckle image. (b) Its pseudo-phase representation. (c) Location of positive (blue circles) and negative (red points) vortices. The colorbar in (a) indicates relative intensity while those in (b) and (c) indicate phase. The axes represent spatial coordinates.

This was done using a series of convolution operations on the phase distribution over the speckle fields as shown in Eq. (3.3) and Eq. (3.4) [10]. This series of convolution operations gives non-zero values only at the locations of phase singularities. These values are the topological charge n_t at those locations.

$$n_t = \phi(x,y) \otimes D_1 + \phi(x,y) \otimes D_2 + \phi(x,y) \otimes D_3 + \phi(x,y) \otimes D_4, \qquad (3.3)$$

where $\phi(x,y)$ represents the pseudo-phase of the speckle field, $\otimes$ is the convolution operator and

38

$$D_1 = \begin{pmatrix} 0 & 1 \\ 0 & -1 \end{pmatrix}; D_2 = \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}; D_3 = \begin{pmatrix} -1 & 0 \\ 1 & 0 \end{pmatrix}; D_4 = \begin{pmatrix} 0 & 0 \\ -1 & 1 \end{pmatrix}. \qquad (3.4)$$

The location of a positive (negative) topological charge is called a positive (negative) vortex. Physically, the two different charges represent the two opposing senses of rotation, clockwise and anti-clockwise, of the phase of the speckle field in the complex plane. Tracking any particular vortex, as it traverses through the frames, results in what has been termed a vortex trail [10].

Physically, coherent light scattering from dynamic biological systems over a period of time leads to temporal decorrelation of the speckle pattern in the observation plane. In our simulations, the individual frames can be considered as snapshots of the speckle field as it decorrelates over time. Thus, traversing through frames is equivalent to the temporal movement of a vortex. In this context, frame numbers can be considered as discrete points in time.
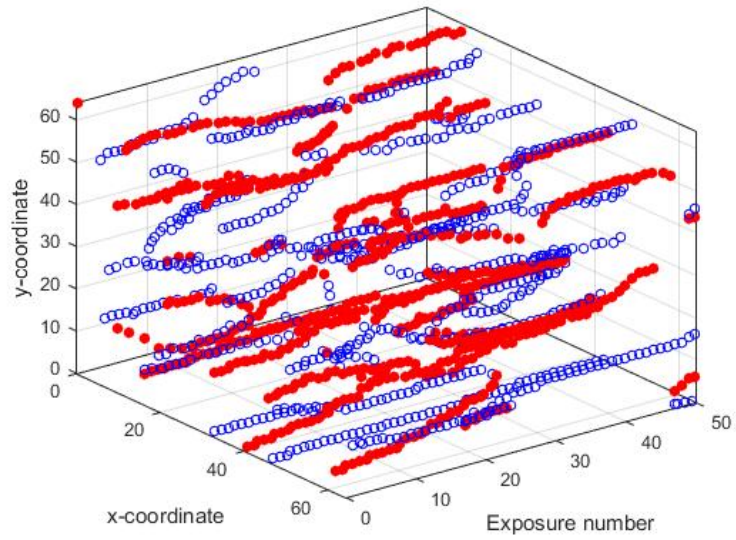
(a)
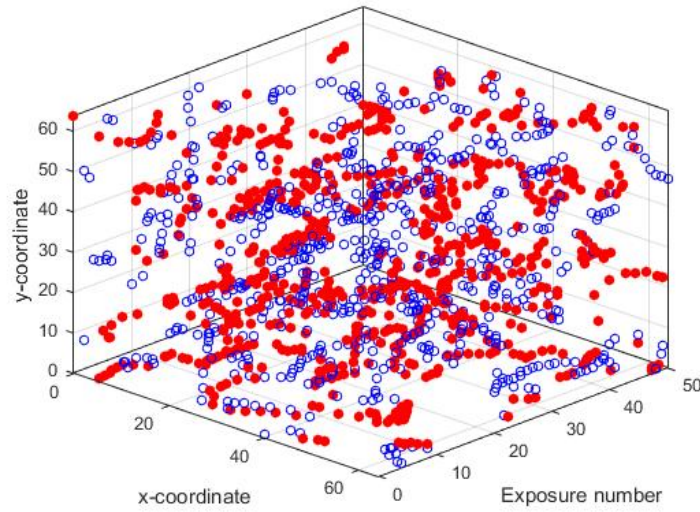
(b)

Fig 3.3. Typical displays of vortex trails, as the vortices (red points: negative,

blue circles: positive) are tracked through the 50 frames (a). Slow

decorrelation with autocorrelation coefficient falling to $1/e$ in 11 frames (b). A faster decorrelation with autocorrelation coefficient falling to $1/e$ in 6 frames.

Identifying the x and y coordinates of each optical vortex as it passes through the frames, gives rise to a trail from its point of generation to its point of annihilation. Occasionally, a trail also ends due the vortex leaving the field of observation. A trail is called either a positive or a negative trail, according to the sign of the topological charge contained in the vortex that created it. When a negative and a positive vortex intersect at a single spatial and temporal location, they annihilate each other and both trails that they were forming end [10].

A challenge in the simulations discussed herein was to draw the boundary between a continuous trail and a new trail starting close to where a trail ended in the previous frame. As the locations of the vortices were defined in the terms of discrete pixels, it was essential to decide how close two vortices in adjacent frames must be to be considered the part of the same trail. Based on empirical observations of multiple simulation results, it was determined that a change of four or more pixels in any direction would be considered to be the beginning of a new trail.

 To create the vortex trail displays like the ones shown in Fig 3.3, the vortices in each frame were located as described above, using the series of convolution operators. The location of each vortex, found in this fashion, was then stored on a frame-by-frame basis. The coordinates of the vortices found in each frame formed an individual list. This was followed

41

by locating the first vortex in the first frame (or equivalently, the first list) and searching for a vortex in the second frame such that it was within four pixels in any direction of the coordinates of the previous vortex found. If such a vortex did not exist, then the trail ended and a new trail began with the next vortex in the first frame. If a vortex was found within the pre-defined four pixel radius, we moved to the third frame and searched for a vortex within the same radius with the coordinates of the second frame as the center location. In this manner, each trail continued until no vortex was found within the set radius of 4 pixels, or all the frames were exhausted. After all the trails starting from the first frame were tracked, the algorithm searched for trails starting from second frame using vortices that were not already considered to constitute any of the previously formed trails. This process continued through the final frame. A pictorial description of this algorithm is shown as a flowchart in Fig 3.4.
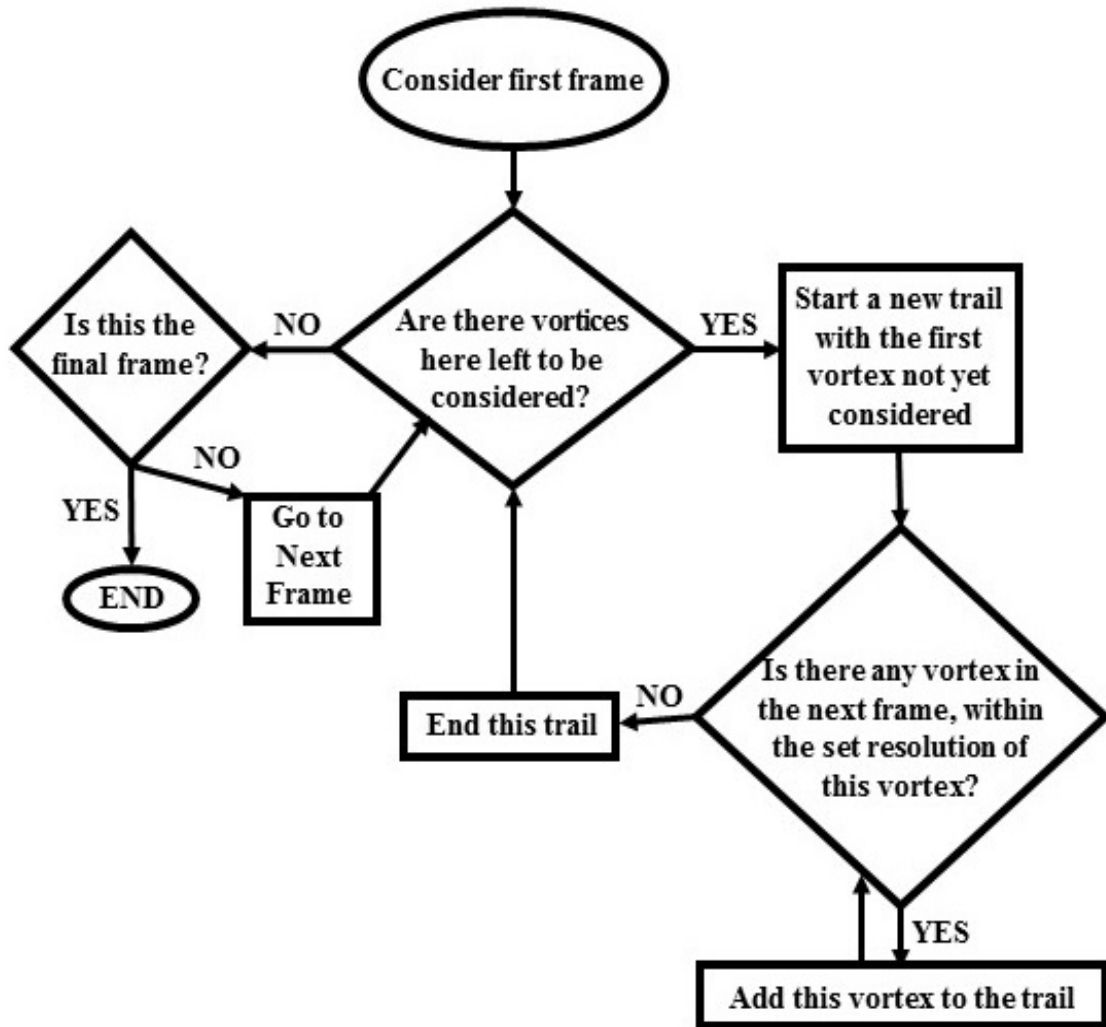
Fig 3.4. Flowchart representing the algorithm followed to track individual optical vortex

trails.

3.3 Measurements on Optical Vortices

3.3.1 Definitions

Once the individual trails were tracked, various parameters of the motion of the vortices

were investigated. One parameter investigated was the average length of the positive and

43

negative trails, as well as the overall average length of all trails regardless of charge. This was defined in terms of the number of frames the average trail survived. Another parameter studied was the average displacement of the vortices, as they traversed through the frames (equivalently, through time). For this, the positive and negative centers of the frames were determined. A positive (negative) center (X_c, Y_c) of a frame is defined as the mean location of all the positive (negative) vortices in the frame:

$$X_c = \frac{\sum_{i=1}^{n} x_i}{n}; Y_c = \frac{\sum_{i=1}^{n} y_i}{n}, \tag{3.5}$$

where (x_i, y_i) is the coordinates of the ith vortex in the frame and n is the total number of positive (negative) vortices in the frame. The overall center of the frame (X_{co}, Y_{co}) is the weighed mean location of the positive and negative centers:

$$X_{co} = \frac{n_p X_{cp} + n_n X_{cn}}{n_p + n_n}; Y_{co} = \frac{n_p Y_{cp} + n_n Y_{cn}}{n_p + n_n}, \tag{3.6}$$

where (X_{cp}, Y_{cp}) is the location of the positive center and (X_{cn}, Y_{cn}) is the location of the negative center; n_p and n_n represent the number of positive and negative vortices in the frame, respectively. We defined the mobility, M, for each type of vortex as the mean distance moved by their center, per frame:

$$M_{cp,cn,co} = \frac{1}{N-1} \sum_{i=2}^{N} \sqrt{({}^i X_{cp,cn,co} - {}^{i-1} X_{cp,cn,co})^2 + ({}^i Y_{cp,cn,co} - {}^{i-1} Y_{cp,cn,co})^2}, \tag{3.7}$$

44

where M_{cq} represents the mobility of the qth type of vortex and $\left(^{i}X_{cq}, ^{i}Y_{cq} \right)$ is the location of the center of qth type of vortex in the ith frame. As indicated, q can represent positive (p), negative (n) or overall (o). N is the total number of frames in the simulation.

Additionally, the distance between the positive and negative centers of each the frame was studied. We have termed this distance as the charge separation, $G(i)$. Poincaré plots were used to analyze the charge separation. The mathematical basis for these plots is detailed in chapter 2. Poincaré plots are used to display consecutive measurements against each other, i.e., the ith element of a series X, $X(i)$ is plotted against the $(i-1)^{th}$ element, $X(i-1)$. This is continued for all set of consecutive data points on X.

Once all the data points are plotted as above, standardized metrics $SD1$ and $SD2$ for the data set are measured with the following definitions

$$SD1 = Var(\frac{1}{\sqrt{2}}X(i-1) - \frac{1}{\sqrt{2}}X(i))^{\frac{1}{2}};$$
$$SD2 = \left\{ 2Var\left(X \right) - SD1^{2} \right\}^{\frac{1}{2}},$$

(3.8)

where $Var(X)$ stands for the variance of the data series X.

An ellipse fitting process is then employed over the Poincaré plots. This ellipse has semi-major axis length of $SD2$ and semi-minor axis length of $SD1$ with the major axis inclined at 45° to the horizontal axis of the Poincaré plot (Fig 3.7). The metric $SD1$ represents the short-term variations of the quantity measured (such as standard deviation in successive differences), while $SD2$ is what remains when contribution of $SD1$ is removed from the variance (Eq. (3.7)) and thus indicates the long-term variations. The shape of the obtained

45

ellipse indicates the comparative effects of each type of variation. The more dominant long-term variations are, the higher is the magnitude of the semi-major axis (given by *SD2*), resulting in a more elongated ellipse.

3.3.2 Results and discussion

The average trail lengths and number of trails for each type of vortex is noted in Table 3.1.

	Slow Decorrelation			Medium Decorrelation			Fast Decorrelation		
	C	G	L	C	G	L	C	G	L
Positive Trail Length (frames)	9.85	8.71	10.09	3.93	4.04	4.42	2.02	2.14	2.48
Negative Trail Length (frames)	9.78	8.64	10.39	3.73	4.21	4.51	1.94	2.05	2.45
Positive Trail Nos.	71	80	64	169	163	154	323	304	282
Negative Trail Nos.	68	83	61	176	164	148	331	314	274

Table 3.1. Average trail lengths and number of trails of each type (C: Constant; G: Gaussian; L: Lorentzian). The trail lengths are given as the number of frames the trail lasted. Total number of frames in the simulation = 50.

46

The effects of decorrelation line shapes and rates were investigated (see above). By weighting the trail length of each type (positive and negative) by the corresponding number of trails, the overall average trail length of each type and rate of decorrelation was obtained. These results are displayed in Fig 3.5.
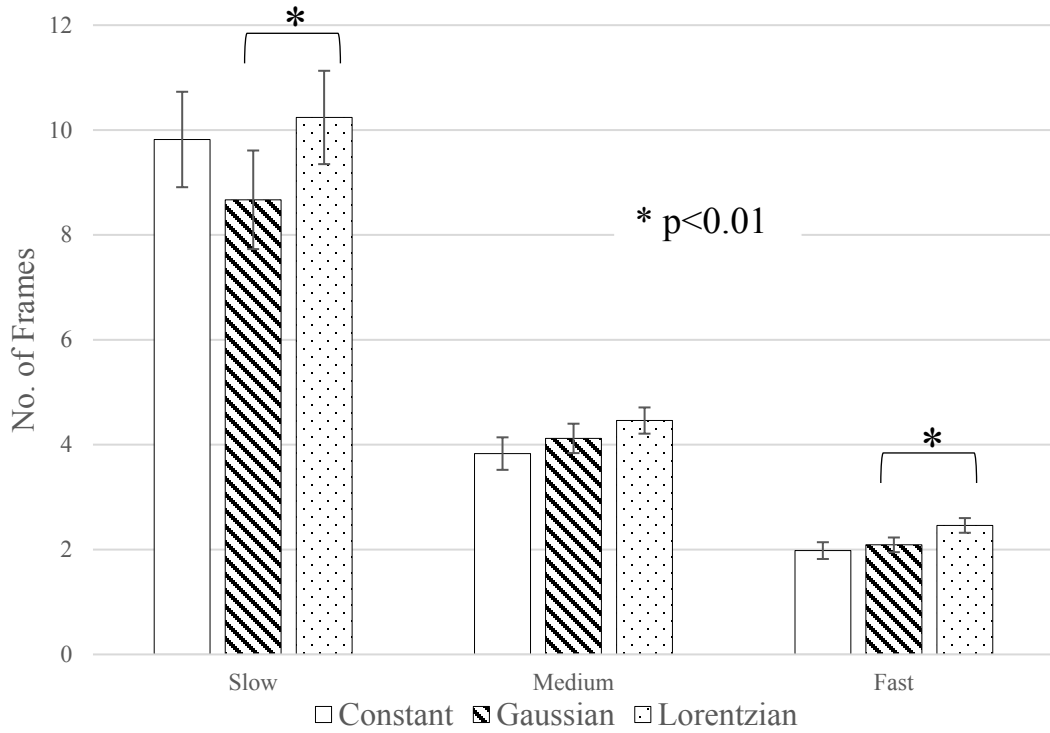

Fig 3.5. Average trail length for each decorrelation type, under three different rates of decorrelation. (Slow: $(1/e)$ = 11 frames; Medium: $(1/e)$ =6 frames; Fast: $(1/e)$ =3 frames; Total frames in simulation = 50. No. of repeated measurements for each behavior = 20; Error bars represent 1 standard deviation

47

The Lorentzian decorrelation behavior typically resulted in longer and fewer trails compared to Gaussian decorrelation, given the same rate of decorrelation ($p < 0.01$ for both the slow and fast rates of decorrelation shown in Fig 3.5). Student's t-test for difference of means was used for all statistical significance testing in this study. Additionally, it was also observed that as the rate of decorrelation increases, the average trail length of each mode became shorter. For each type of decorrelation, the number of vortices in each frame roughly remained independent of all other simulation conditions. The number of vortices identified was on the order of half of the number of coherence areas in each speckle field [14]. Thus, longer trails directly resulted in fewer number of trails. In addition to elongated trail lengths, slower decorrelation was also found to be associated with lesser number of individual trails. The third mode, constant rate of decorrelation ($\Gamma = 0.9984$ for slow, $\Gamma = 0.99$ for medium and $\Gamma = 0.96$ for fast decorrelation, respectively), formed the shortest, and by extension, the most number of trails at the medium and fast decorrelation modes. At the slowest decorrelation rate, the constant decorrelation mode formed trails longer than the Gaussian mode, but shorter than the Lorentzian mode.
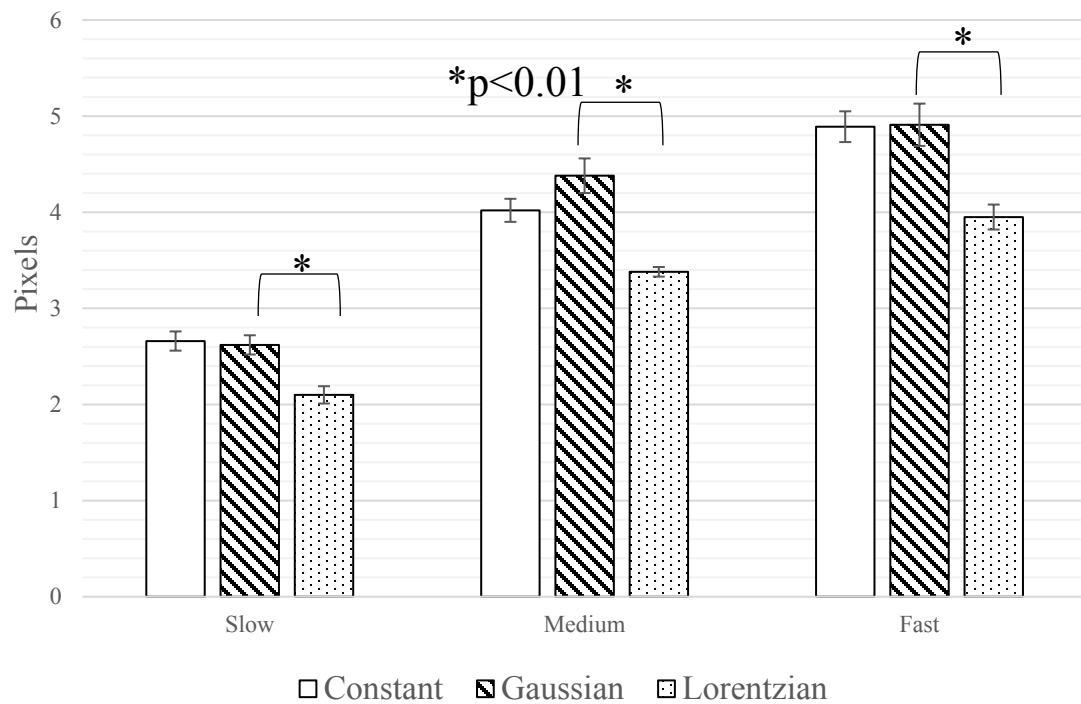
Fig 3.6. Average vortex displacement (or mobility) for each mode, under three different rates of decorrelation (same as Fig 3.5). Total measurements = 20; Error bars represent one standard deviation

	Slow Decorrelation			Medium Decorrelation			Fast Decorrelation		
	C	G	L	C	G	L	C	G	L
Positive Displacement	2.63	2.58	2.14	4.30	4.67	3.37	5.07	5.14	4.40

49

Negative Displacement	2.96	2.97	2.29	4.26	4.50	3.71	5.17	5.56	4.57
Overall Displacement	2.66	2.62	2.10	4.02	4.38	3.38	4.89	4.91	3.95

Table 3.2. The average displacement of each type of vortex center as they travel through the frames (C: Constant; G: Gaussian; L: Lorentzian). All distances are given in terms of coordinate points. Each frame was a square of 64×64 coordinate points.

The next parameter that was investigated was the mobility, M, of the vortices. Mobility was defined in terms of the average displacement incurred by the center of each type of vortex field, while moving between consecutive frames. From Table 3.2, we can observe that the vortices experienced higher mobility under Gaussian decorrelation, as compared to Lorentzian decorrelation ($p < 0.01$). Additionally, increasing the rate of decorrelation resulted in higher motion of the vortices ($p < 0.01$). Combining the results from both the tables above, it can be seen that the longer trails were associated smaller displacements.

To study the variations in charge separation, Poincaré plots [7] were employed. As noted above, these plots are used to investigate the long-term and short-term variations in a variable. In this case, the variable in question was the charge separation over the 50 frames. A typical Poincaré plot from this set of measurements is shown in Fig 3.7.
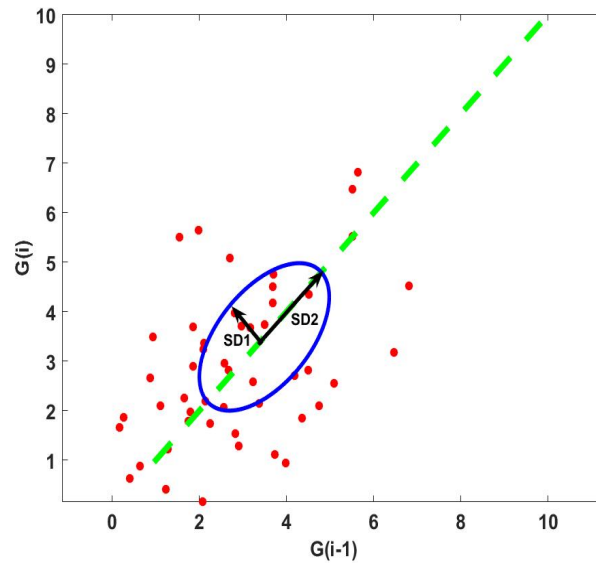
Fig 3.7 . A typical Poincaré plot obtained for charge separation $G(i)$. The magnitude of descriptor SD1 indicates short-term variability in the value of the separation, while SD2 indicates the long-term variability. In this plot, SD1 = 1.01 and SD2 = 1.85. The lengths are in terms of coordinate points. This simulation was done for a 64 X 64 pixel speckle field.

It was observed that slow decorrelation resulted in elongated ellipses on the Poincaré plot, which is a result of higher *SD2/SD1* ratio. As mentioned earlier, this is the ratio of the long-term variability to the short-term variability. Thus, we notice that as the decorrelation rate increases, the short-term variations tend to become more prominent as compared to the long-term variations. This ratio tends towards a value of 1.0 at higher decorrelation rates. The *SD2/SD1* ratios can be visualized by placing the obtained ellipses in overlap with each other, as shown in Fig 3.8.

51

(a)

(b)

52
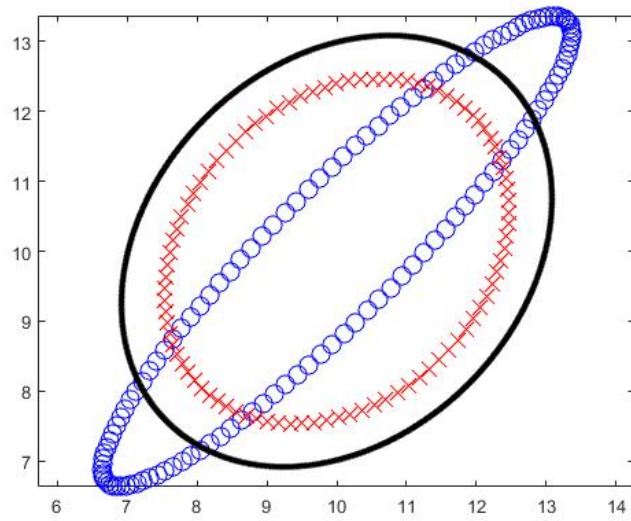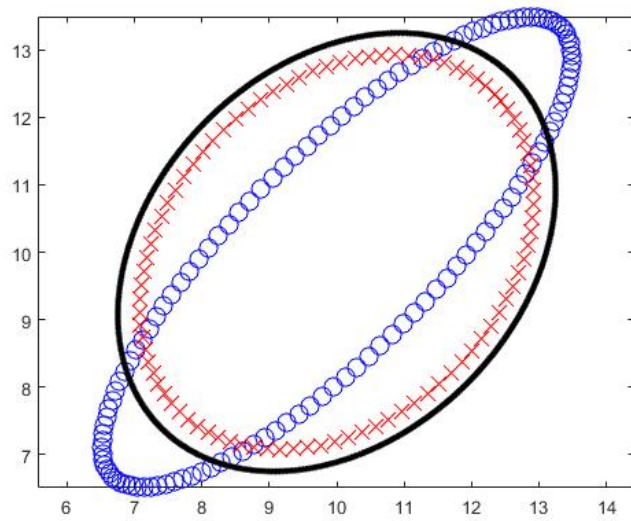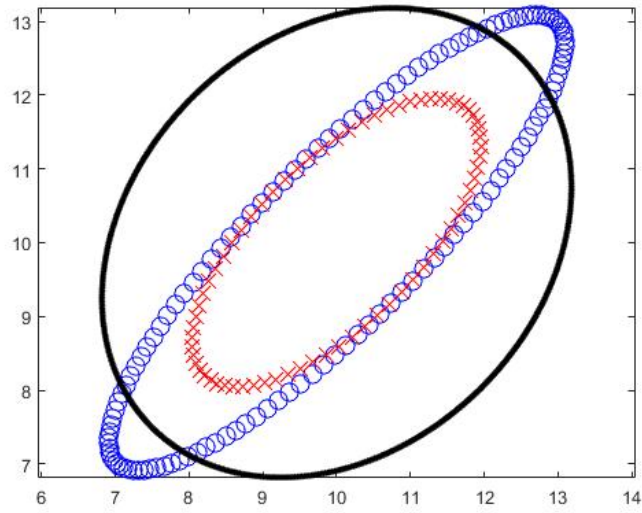
(c)

Fig 3.8 . Ellipses from the Poincaré plots of charge separation *G(i)* for (a) Gaussian(b)
Constant and (c) Lorentzian decorrelation behavior. (O: Slow decorrelation; X : Medium
decorrelation; - :Fast decorrelation). It can be noted that the ellipses are prominently
elongated for the slowest rate of decorrelation, indicating a high *SD2/SD1* ratio. This
elongation of the ellipse indicates a greater effect of long-term variations on the
conditions under which the measurement has been made.

It can be seen that the use of Poincaré plots of charge separation gives a stronger indication
of the rate of decorrelation, rather than the lineshape of the decorrelation. Thus, this can be
used as an indicator of the rate of dynamics in a system as is discussed below.

53

3.4 Analysis

A new approach to studying dynamic systems by tracking phase singularities (or optical vortices) has been presented. Speckle patterns for biological systems inherently contain phase singularities, or optical vortices, points around which the phase of the field rotates. In this work we have related the observed dynamics of the optical vortices in a simulated speckle field, to the type and rate of decorrelation in the field. This decorrelation is a representative of the dynamic behavior of scatterers that, under coherent illumination, produced the speckle field under observation [13].

Analyzing the location of vortices at discrete time points in a speckle field simulated using a pre-defined rate of decorrelation, we note the expected result that as the rate increased, the lengths of the vortex trails decreased. In this study, the rate of speckle decorrelation was assumed to correspond to the rate of activity in the scattering object. Thus, in a scattering medium with fast motion among particles, the individual phase singularities tend to survive for shorter periods of time. Faster activity was also shown to result in a higher degree of vortex displacement per frame in the location of the singularities. Additionally, we compared different modes of activity; a Lorentzian decorrelation relationship between frames, corresponding to Brownian or unordered form of activity among the scatterers and a Gaussian decorrelation relationship, corresponding to an ordered mode of activity among the scatterers. The ordered motion resulted in more mobile phase singularities as they traversed through time. This also corresponds to shorter durations for which the trails existed. We remind the readers that the mobility, M, is defined as the average displacement

54

per frame of the mean location (center) of the vortices. A possible reason for low vortex mobility for Brownian motion could be the homogeneous line profile of the scatterer distribution in a Lorentzian flow model [15]. This homogeneity results in no preferential direction for the vortices to move. This results in the average summed motion to be less than the Gaussian flow model, which has an inhomogeneous line profile of the scatterers [15]. We also examined a technique of analyzing sequential vortex data using Poincaré plots, by separating the long and short-term variations and calculating their relative prominence with regards to each other. When applied to the average charge separation per frame, this technique did not indicate any significant difference between the different modes of decorrelation. However, reducing the rate of decorrelation did result in the Poincaré ellipses elongating, indicating a higher prominence of the long-term variations. A possible reason could be that faster decorrelation did not allow sufficient time for the long-term trends in the data to dominate over the short-term trends. We note here that at no point at all do the short-term variations $SD1$ dominate over the long-term variations $SD2$. In the case in which $SD1 > SD2$ the ellipse would be more "broad" than "long', i.e., the major and minor axes would have reversed. Thus, the use of Poincaré plots was more useful in comparing the different rates of activity, rather than the different modes of activity.

Note that actual phase information is not available in the intensity speckle patterns, and thus we took the approach of calculating pseudo-phase estimate maps using convolution operators [10]. While this approach will not necessarily re-create the true phase of an imaged speckle pattern, the behavior of the vortices is not affected. Also, the simulation only takes into account a limited spatial window of the speckle field. Thus, there are

instances of minor topological charge imbalance (about 1 extra vortex in 4000 total vortices) in some of the simulations. This usually occurred in situations where a vortex was close to the edge of the observation window, and the opposite topological charge corresponding to this lone vortex was just outside the window. This oppositely charged vortex typically appeared within the window in the very next frame, thus balancing out the charges. This can be thought of as a case where a pair production happens right at the edge of the window, but only half the pair is within the window at that instant. It is also to be noted we are currently not in a position to speculate on the physical significance of some of the parameters noted here (such as the charge separation, $G(i)$). As such, they are introduced only as a measureable feature with some sensitivity to the simulation dynamics (in the case of $G(i)$, sensitivity to the rate of activity).

3.5 Summary

In this chapter, methods to analyze vortex dynamics in simulated dynamic speckle fields have been introduced. These methods can be used to add to our current understanding of the field of singular optics. This can, in turn be used to improve existing microscopy and imaging techniques, particularly those aimed at quantifying biophysical dynamics. Simulating additional biological features such as tumors and other complex systems containing dynamics scattering elements, and identifying them using the dynamics of optical vortices as done here, is among the possible directions in which this research can progress. For now, we turn to mimicking the systems simulated in this section, in a laboratory. In the next chapter, fluid phantoms and metal blocks are used as scattering

systems with varying degree and mode of orderliness in their dynamics. The vortex dynamics obtained in their speckle fields are analyzed similar to how it is done for simulated fields in this chapter.

3.6 References

[1] A. F. Fercher, and J. D. Briers, "Flow visualization by means of single-exposure speckle photography", Optics Communications 37 (5), pages 326-330 [1981].

[2] S. E. Murialdo, G.H. Sendra, L. I. Passoni, R. Arizaga, and J. F. Gonzalez, "Analysis of bacterial chemotactic response using dynamic laser speckle", Journal of Biomedical Optics 14 (6), 064015 [2009].

[3] H. Sendra, S. Murialdo, and L. Passoni, "Dynamic laser speckle to detect motile bacterial response of *Pseudomonas aeruginosa*", Journal of Physics: Conference Series 90, 012064 [2007].

[4] A. Oulamara, A. Tribillon, and G. Doverney, "Biological activity measurements on botanical specimen surfaces using a temporal decorrelation effect of laser speckle", Journal of Modern Optics 36 (2), pages 165-179 [1989].

[5] J. D. Briers, and S. Webster, "Laser Speckle Contrast Analysis (LASCA): A nonscanning, full-field technique for monitoring capillary blood flow", Journal of Biomedical Optics 1 (2), pages 174-179 [1996].

[6] D. A. Boas, and A. K. Dunn, "Laser speckle contrast imaging in biomedical optics", Journal of Biomedical Optics 15 (1), 011109 [2010].

[7] A. K. Dunn, H. Bolay, M. A. Moskowitz, and D. A. Boas, "Dynamic imaging of cerebral blood flow using laser speckle", Journal of Cerebral Blood Flow and Metabolism 21 (3), pages 195-201 [2001].

[8] J. F. Nye, and M. V. Berry," Dislocations in Wave Trains", Proceedings of the Royal Society of London A 336, pages 165-190 [1974].

[9] M. Berry, J. Nye, and F. Wright, "The elliptic umbilic diffraction catastrophe", Philosophical Transactions of the Royal Society 291 (1382), pages 453-484 [1979].

[10] S. J. Kirkpatrick, K. Khaksari, D. Thomas, and D. D. Duncan, "Optical vortex behavior in dynamic speckle fields", Journal of Biomedical Optics 17 (5), 050504 [2012].

[11] M. S. Soskin, V. N. Gorshkov, M. V. Vasnetsov, J. T. Malos, and N. R. Heckenberg, "Topological charge and angular momentum of light beams carrying optical vortices", Physical Review A 56 (5), pages 4064-4075 [1997].

[12] D. D. Duncan, and S. J. Kirkpatrick, "The copula: a tool for simulating speckle dynamics", Journal of the Optical Society of America A 25 (1), pages 231-237 [2008].

[13] D. D. Duncan, and S. J. Kirkpatrick, "Can laser speckle flowmetry be made a quantitative tool?", Journal of the Optical Society of America A 25 (8), pages 2088-2094 [2008].

[14] J. W. Goodman, *Speckle Phenomena in Optics Theory and Applications*, Roberts & Co., Englewood, CO, USA [2007].

58

[15] M. E. Thomas, and D. D. Duncan, "Atmospheric transmission," *Atmospheric Propagation of Radiation*, Vol. 2 , Infrared & Electro-Optical Systems Handbook [1993].

4 Optical Vortices from Dynamic Scattering Samples

In the last chapter, correlations were drawn between dynamic scatterer system behavior and evolution of optical vortex fields, through simulated speckles. In this chapter, similar correlations are drawn between physical systems and their resultant vortex fields. The calculations in this chapter have been performed in a MATLAB environment using the code in Appendix B.2.

4.1 Introduction

When coherent waves- such as those from a laser source- are scattered from biological tissues, granular structures appear in the scattered field. These structures, called speckle patterns, are due to the random interference of the incident coherent waves. These patterns have been the subject of extensive research and are a subject of large number of applications exploiting speckle metrology and interferometry [1]. Within these speckle fields are locations where the phase of the resultant wave is undefined and the electric field amplitude - and by extension, the intensity- is zero [2]. These phase singularities are known as optical vortices. The word "vortex" is used as the phase undergoes a *rotation* around the singularity. This rotation can have one of two senses- clockwise and counter-clockwise. Depending on this sense of rotation, optical vortices are denoted to have either a positive or a negative *topological* charge [3]. The value of this charge n_t indicates the number of rotations of 2π radians the phase goes through, around the singularity:

$$n_t = \frac{1}{2\pi} \oint_c \nabla \phi(x, y) dl, \qquad (4.1)$$

where ϕ is the phase and the line integral is along a closed loop, l, around the vortex.

As in chapter 3, Majumdar and Kirkpatrick [4] have demonstrated that in speckle fields simulated using computer algorithms, the evolution of optical vortex distribution can be correlated to the type and rate of activity in the scattering source. In this work, solutions of microspheres works as proxy for biological tissues. Laser light scattered from these solutions give rise to speckle patterns. These patterns and the singularities in a pseudo-phase mask obtained from these patterns are then analyzed. The range of motions studies vary from the completely ordered motion with decorrelation, to completely disordered motion. Additionally, effect of temperature in the motion of the optical vortices is also analyzed. Comparisons are drawn between the previously demonstrated results from simulated speckle fields, and the analysis of these systems that represent biophysical tissues.

4.2 Materials and Methods

4.2.1 Source of scatterers

Fluid phantoms comprising of hollow glass microspheres (Potters Industries, Inc.) suspended in DI-water were used to represent Brownian motion (i.e., Lorentzian decorrelation in the time-evolving speckle field). The solutions were held in transparent colorless cuvettes of internal cross-section 10 mm X 10 mm. The concentrations of microsphere solution used in this study ranged from 0.060 mg/ml to 0.157 mg/ml. To

represent orderly motion (i.e. Gaussian decorrelation), the same cuvettes without the solutions were used. Additionally, to observe a state of zero decorrelation, the roughened side of a block of aluminum was used to scatter light.

4.2.2 Obtaining Speckle Images

A polarized 532-nm laser (Melles Griot, Inc.) was used as the source of illumination for this experiment. The light from this source was passed through a variable neutral density filter and focused at the center of the scattering sample using a 10X microscope objective (Newport, Inc.). To obtain the scattering coefficients of the samples, ballistic transmission measurements were done and Mie theory was used . For this, the scattered light was incident on to a visible femtowatt photoreceiver (New Focus, Inc.) through a 150 micron pinhole. It was assumed that the absorption coefficient, μ_a, was negligible compared to the scattering coefficient, μ_s. Thus, the Beer-Lambert law of decay in propagating light intensity

$$I(z) = I_0 e^{-(\mu_s + \mu_a)z} \tag{4.2}$$

changes to

$$I(z) = I_0 e^{-\mu_s z} \tag{4.3}$$

giving the scattering coefficient as

$$\mu_s = \frac{\ln\left(\dfrac{I_0}{I(z)}\right)}{z} \tag{4.4}$$

62

Here, I_0 is the incident intensity and $I(z)$ is the intensity after the light travels a linear distance z in the scattering medium. the reduced scattering coefficient, $\mu_s{}'$ was calculated by assuming a scattering anisotropy of g=0.9 and

$$\mu_s{}' = \mu_s(1-g) \ . \tag{4.5}$$

To obtain the speckle images (Fig 4.1), a CCD camera (Point grey, Dragonfly) was used. The images were collected near real-time at 200 frames per second using a firewire adapter (StarTech, Inc.). The scattering samples were placed on a movable base controlled by a picometer control pad (New Focus, Inc.). The samples were studied under speeds ranging from 0.33 to 0.98 mm/min.
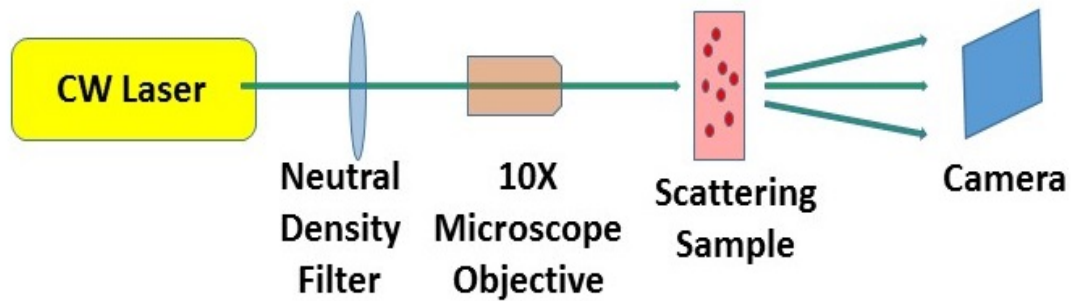

Fig 4.1. Experimental setup to capture speckle images from scattering samples consisting of solutions of hollow microspheres

63

For the fluid samples and empty cuvette samples, the speckles were recorded in the transmitted scattered light. For the aluminum block, the speckles were recorded in the diffusively reflected light.

4.2.3 Locating Optical Vortices

Once the speckle images were obtained, all further computations were done in a MATLAB environment. First, to remove noise artifacts, the images were smoothened by passing through a Gaussian image filter [5]. The smooth speckle images were then passed through a Hilbert filter to obtain their pseudo-phase map. A series of convolution operations were then performed on these pseudo-phase mappings to obtain the locations of phase singularities, or optical vortices [6]. (Fig 4.2)
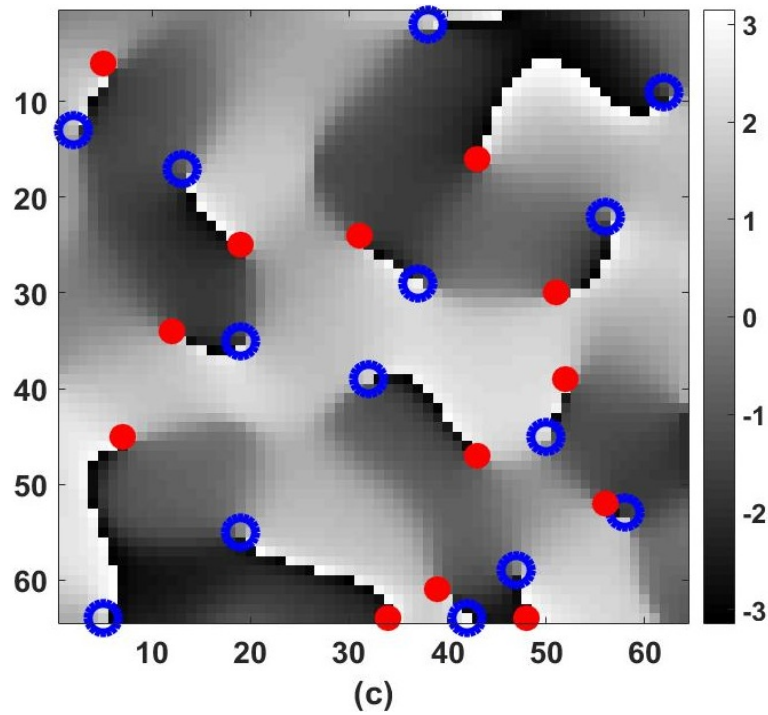
(a)

(b)

65
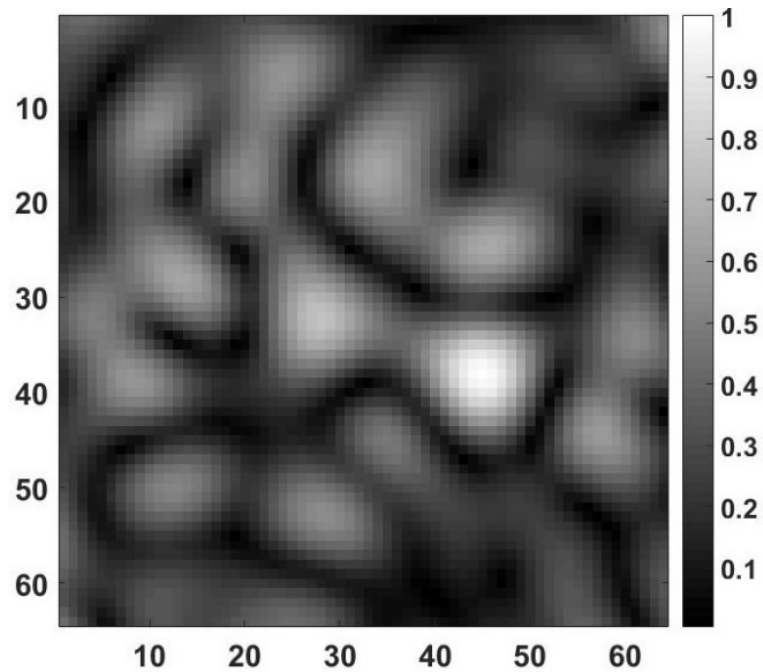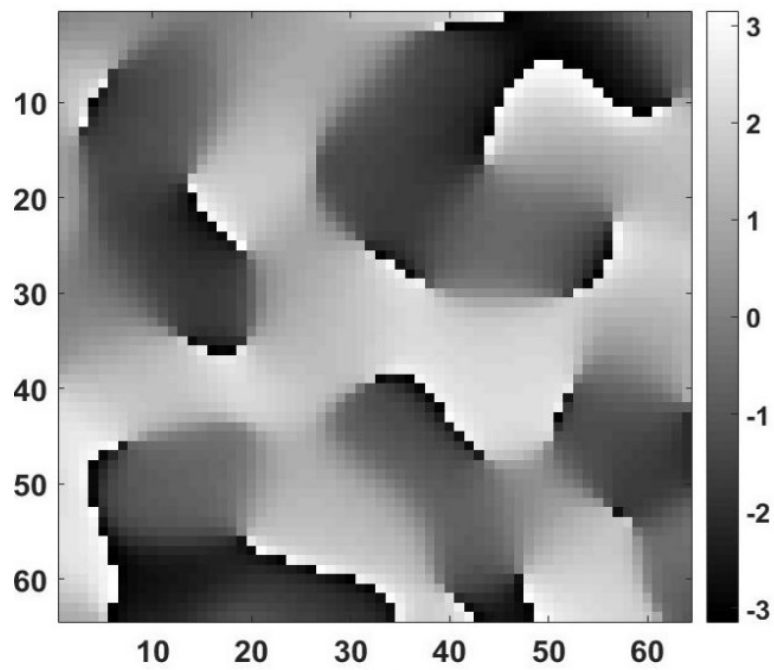
Fig 4.2. (a) An obtained speckle frame (b) Its pseudophase representation (c) Location of vortices obtained from convolution operations on the pseudophase map. The blue circles represent positive vortices and red points indicate negative vortices. Reproduced with permission from [4].

4.2.4 Analyzing time-evolution of the Vortex fields

Once the optical vortices in each of the speckle frames are located, these locations are stored in a three dimensional array, with one temporal and two spatial dimensions indicating the motion of the vortices in the 2-D frame with time. (Fig 4.3) In this way, trails of individual vortices were obtained.
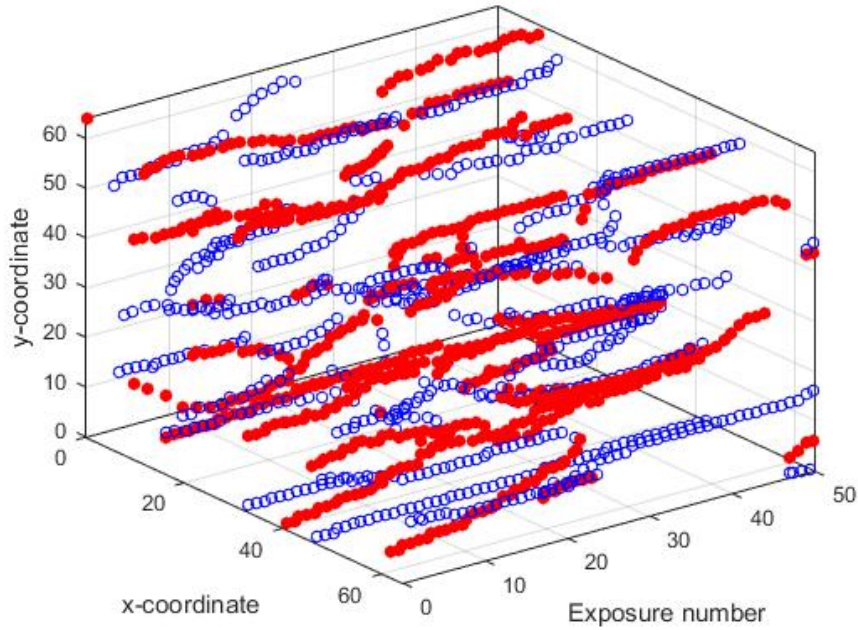
66

Fig 4.3. Vortex trails obtained after stacking together the vortex locations from individual frames. The blue circles represent positive vortices and red points indicate negative vortices. Reproduced with permission from [4].

An algorithm previously used by Majumdar and Kirkpatrick [4] was followed to characterize the vortex trails. The three quantities in focus for each sample in this study were the average trail length (i.e. , the time for which each vortex lasted), the mobility (i.e., the average displacement of the center of the vortex distribution, per frame) and the charge separation (i.e., the average distance between the positive and negative centers, per frame). Mathematically, a positive (negative) center (X_c, Y_c) of a frame is defined as the mean location of all the positive (negative) vortices in the frame

67

$$X_c = \frac{\sum\limits_{i=1}^{n} x_i}{n}; Y_c = \frac{\sum\limits_{i=1}^{n} y_i}{n}, \tag{4.6}$$

where (x_i, y_i) is the coordinates of the ith vortex in the frame and n is the total number of positive (negative) vortices in the frame. The overall center of the frame (X_{co}, Y_{co}) is then the weighed mean location of the positive and negative centers:

$$X_{co} = \frac{n_p X_{cp} + n_n X_{cn}}{n_p + n_n}; Y_{co} = \frac{n_p Y_{cp} + n_n Y_{cn}}{n_p + n_n}, \tag{4.7}$$

where (X_{cp}, Y_{cp}) is the location of the positive center and (X_{cn}, Y_{cn}) is the location of the negative center; n_p and n_n represent the number of positive and negative vortices in the frame, respectively. We defined the mobility, M, for each type of vortex as the mean distance moved by their center, per frame:

$$M = \frac{1}{N-1} \sum_{i=2}^{N} \sqrt{(^{i}X_{co} - {}^{i-1}X_{co})^2 + (^{i}Y_{co} - {}^{i-1}Y_{co})^2}, \tag{4.8}$$

where $\left(^{i}X_{co}, {}^{i}Y_{co} \right)$ is the location of the overall vortex center in the ith frame. N is the total number of frames.

Additionally, the distance between the positive and negative centers of each the frame was studied. We have termed this distance as the charge separation, G. To analyze this,

standardized metrics *SD1* and *SD2* for the data set are measured with the following definitions (refer to chapter 2 for derivation and details)

$$SD1 = Var(\frac{1}{\sqrt{2}}G(i-1) - \frac{1}{\sqrt{2}}G(i))^{\frac{1}{2}};$$

$$SD2 = \left\{2Var(G) - SD1^2\right\}^{\frac{1}{2}},$$

(4.9)

where *Var(G)* stands for the variance of the data series *G* and *G(i)* is the ith element of the series. From this, the ellipticity, ℓ , of the data is measured as

$$e = \frac{SD2}{SD1}.$$

(4.10)

4.3 Results and discussion

4.3.1 Characterization of scattering samples

The manufacturer of the hollow glass spheres used in these experiments indicates that the spheres had a range of dimeters from 9 to 11 microns. A ballistic transmission experiment was set up to identify the diameter of spheres for which the solutions can be assumed to be monodisperse. The ratio of intensity passing through five different concentration of spheres was measured and Beer-Lambert law was applied to estimate the reduced scattering coefficients.
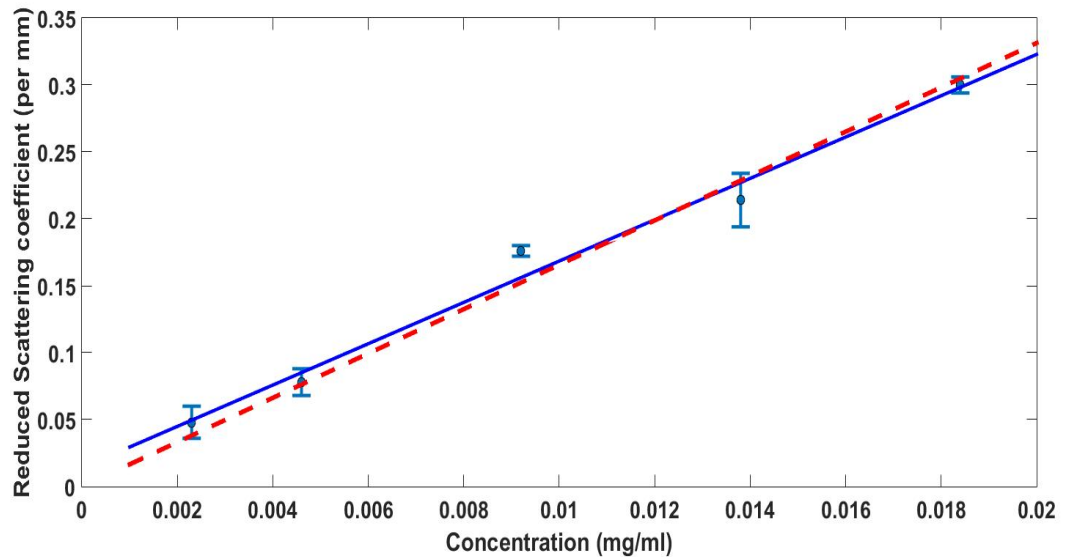
69

Fig 4.4. The reduced scattering coefficients calculated using Beer-Lambert law are indicated by dots (error bars indicate one standard deviation). The solid blue line represents the best linear fit. The broken red line indicates the theoretical fit for spheres of diameter 9.8 microns.

On applying a linear fit to the obtained values, it was found that the best-fitting line has the equation

$$y = 15.4401x + 0.014 \tag{4.11}$$

where y is the reduced scattering coefficient per millimeter and x is the concentration of spheres in mg/ml. This line is represented as the blue solid line in Fig 4.4. On using an online Mie scattering calculator [7], it was found that the scattering behavior was similar to that of a solution of spheres with uniform diameter of about 9.8 microns.

70

4.3.2 Effect of Temperature

Measurements obtained at room temperature (296 K) and those when the samples were heated to a higher temperature (330 K) are compared in the following Table 4.1:

Solution Concentration (mg/ml)	Trail Lengths (No. of frames)		Mobility (No. of pixels)		Ellipticity in charge separation	
	296 K	330 K	296 K	330 K	296 K	330 K
0.060	17.136	5.761	14.498	32.817	2.788	1.554
0.072	9.954	4.798	18.557	32.142	1.898	1.286
0.077	10.550	3.660	20.120	39.407	2.720	1.459
0.080	12.844	5.681	17.399	32.952	2.604	1.608
0.092	24.245	3.637	11.189	54.340	4.734	1.379
0.096	8.364	5.034	20.859	32.536	2.049	1.394
0.120	8.243	4.931	22.866	31.152	1.697	1.469
0.157	15.884	3.480	15.851	48.538	2.697	1.296

Table 4.1. Changes in trail length, vortex mobility and the ellipticity in charge separation due to temperature

As is evident from the table, a higher temperature results in trails lasting a shorter period of time, while the vortices in each frame become more mobile. Additionally, the charge

71

separation ellipticity is reduced at higher temperatures. These trends were observed for each of the concentrations studied.

4.3.3 Effect of type of Motion

In the above section, all samples were inside stationary cuvettes at all times. Thus, the only motion involved were the Brownian (disordered) motion of the scatterer spheres within the solution. We notice that even with the type of motion remaining the same, evolving speckle fields have variations in their properties. Thus, when comparing different types of motion, we need to account for the difference in motion within the same type. To achieve this, we introduce a "decorrelation factor" D such that

$$D = 100 \times (1 - \Gamma) \tag{4.12}$$

where Γ is the mean autocorrelation coefficient between the consecutive speckle frames obtained from the measurements on any particular sample. A higher value of D for the same type of motion indicates a faster decorrelation of the speckle frames in a sample, and thus results in shorter trail lengths and higher mobilities. For instance, from the solutions examined in Table 1, that value of D at 296 K was measured to be 0.462 ± 0.237 while that at 330 K was measured to be 5.694 ± 3.019 .

With this reasoning, to obtain normalized traillengths, the measured traillengths were multiplied by a factor of D to account for the rate of motion. The results are displayed in Fig 4.5(a).

Normalized Traillengths

*p<0.01
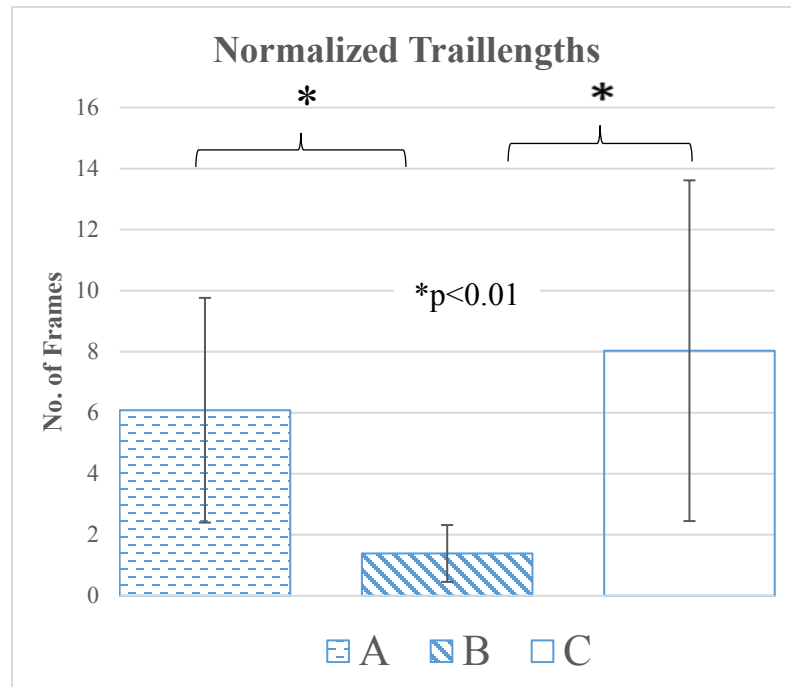
(a)
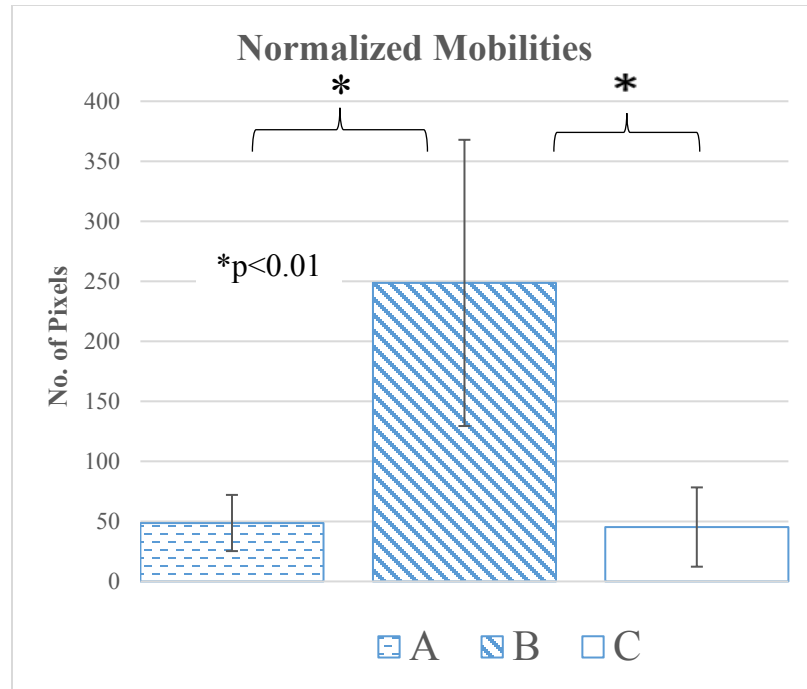
(b)

Fig 4.5. (a) Normalized traillengths and (b) normalized mobilities for three different systems: (A) Solutions of microspheres in stationary cuvettes (14 samples) , (B) Cuvettes translating without any solution (12 samples), (C) Solutions in translating cuvettes (12 samples). Error bars represent 1 standard deviation.

Similarly, to obtain the normalized vortex mobilities, the measured mobilities were divided by a factor of D to account for the rate of motion. The results are displayed in Fig 4.5(b).

Student's t-test for difference of means was followed for checking statistically significant differences. It was found that both the trail lengths and mobilities in the translation of empty cuvettes were significantly different from the other two cases, but these two cases (solutions in stationary cuvettes and solutions in moving cuvettes) were not significantly different from each other.

74

4.3.4 A solid aluminum block- The case of minimal decorrelation

When studying the evolution of vortex field in the speckle patterns obtained as a result of interference from a solid diffusively reflecting block, the idea of using the decorrelation factor as above cannot be applied. This is because we are looking at scattering form a single steady (and thus, non-decorrelating) surface. If measured, the autocorrelation coefficient between consecutive speckle frames is $\Gamma > 0.9999$, thus resulting in extremely small values of D below 0.01. An example of the vortex trails obtained is shown in Fig 4.6.
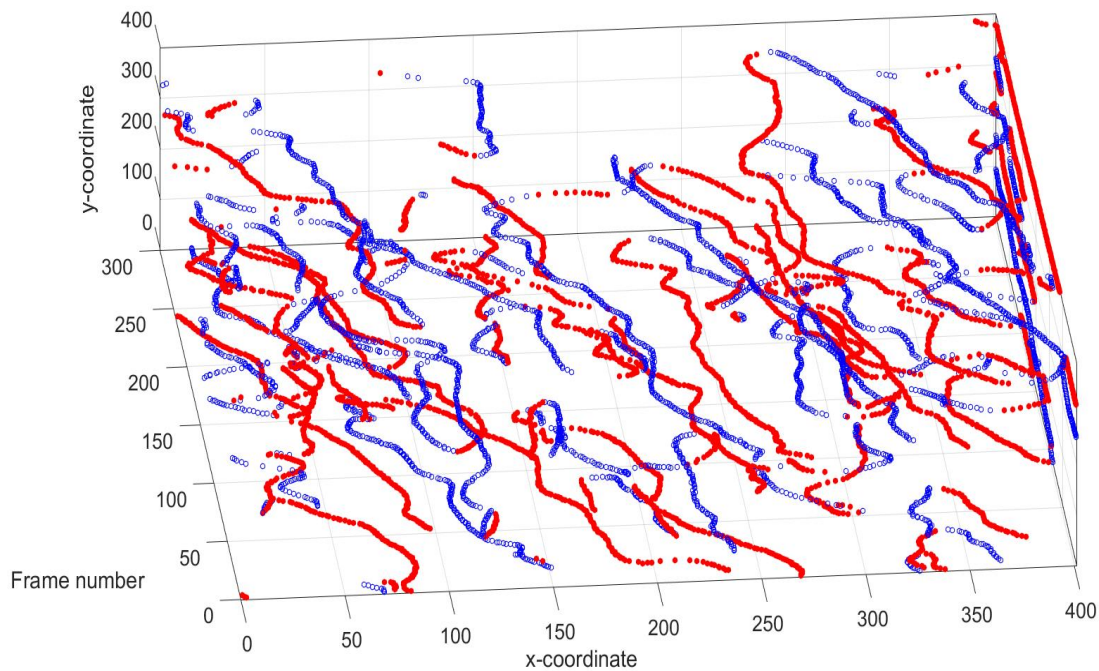

Fig 4.6. Vortex trails obtained from a piece of aluminum block translating at 0.33 mm/min. For simplicity, only 300 frames are shown. The images were captured at 200 fps. The blue circles represent positive vortices and red points indicate negative vortices.

75

As can be noticed, the vortex motion reflect the linear shift of the scattering source quite closely, as the source undergoes negligible decorrelation. On close inspection, the point of pair production and annihilation of the optical vortices can also be noticed, as displayed in Fig 4.7.
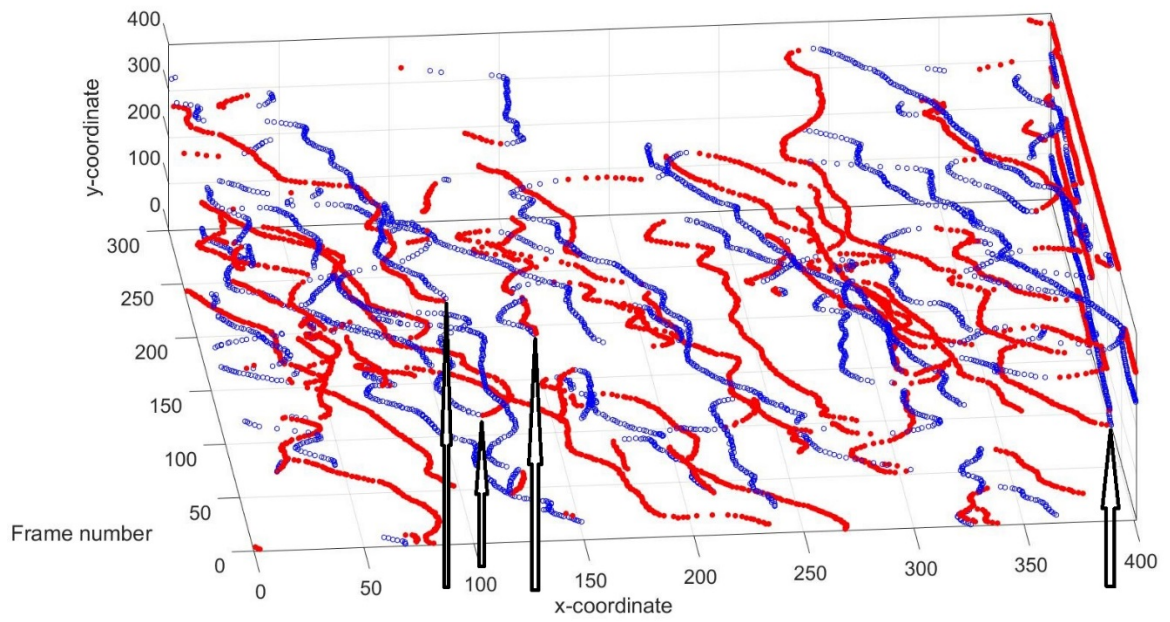
(a)
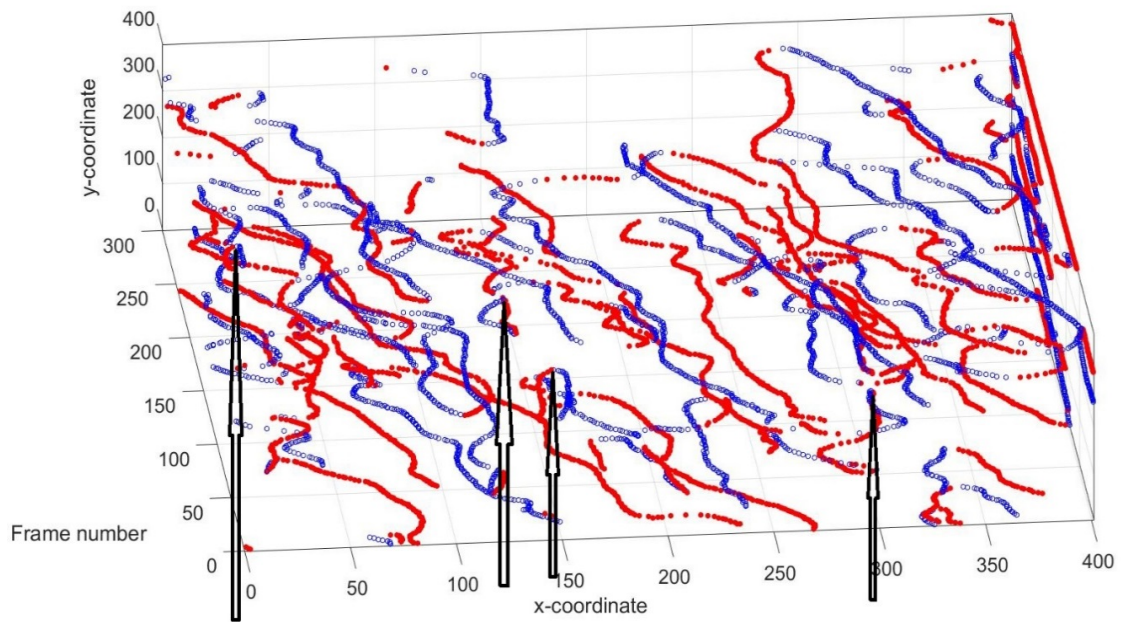
(b)

Fig 4.7. The points of (a) pair production and (b) annihilation of certain vortex pairs indicated by arrows. The blue circles represent positive vortices and red points indicate negative vortices.

4.4 Conclusion

Small particles such as the hollow glass spheres used in these experiments, when suspended in a fluid, exhibit Brownian motion. Einstein's studies using kinetic theory were the first mathematical detailing of these systems [8]. Advancements using Langevin's equation and

78

the equipartition theory estimate the mean-squared speed $<v^2>$ of Brownian particles as

$$\frac{1}{2}m<v^2>=\frac{1}{2}kT \tag{4.13}$$

where m is the mass of the particle with absolute temperature T and k is the Boltzmann's constant [9]. From Eq. (4.13), it can estimated that increasing the temperature from 296 K to 330 K, as we have done, results in an approximate increase of 5.6% in particle speed. This increase in speed of scattering particles is reflected in the higher mobility of the resultant optical vortices, as well. It was also noticed that as the particle motion became more rapid, the ellipticity in the vortex charge separation reduced. The reduction in ellipticity values indicate a lowering of long-range correlations in the variables for which the ellipticity is measured [10]. A quickening of scatterer motion thus makes short-term variations in the vortex distribution more significant.

As has been previously reported [4], a higher mobility of the vortices is typically associated with shorter trail lengths. This trend continues with the normalized variables in section 4.3.3. Fluid samples exhibiting Brownian motion provide a source to study disordered motion. On the other extreme lies completely ordered motion. Such a motion exhibits a Gaussian decorrelation relation between frames [11]. In our experiments, ordered motion is represented by empty cuvettes translating at a uniform speed. Most biophysical phenomena, however, exhibit dynamics in between the completely ordered and completely disordered regimes. To mimic this, we introduced the microsphere solutions to the above case of translating cuvettes. Thus, we have the Brownian motion within the solutions added to the ordered motion of the bulk of the solution and the cuvette. Our vortex analysis clearly

indicates statistically significant differences between the purely disordered and purely ordered regimes, as well as between the purely ordered and partially disordered (mixed) regimes. No significant difference was found between the purely disordered and the mixed regimes.

4.5 Summary

In this paper, ways to analyze optical vortex fields to distinguish the dynamics of the underlying scatterers are presented. Vortices were detected in speckle fields obtained from scattering systems act as proxies to actual biophysical systems. Some of the results reflect similar studies in computer-simulated speckle fields reported previously [4]. An aspect for future investigation could be to study the effects of concentration of scatterers, which we haven't touched as we were more focused to look into Brownian motion as a single entity to be compared with other types of motions. Another addition could be to vary the relative weights of ordered and disordered motions and establish a boundary where vortex analysis can distinguish between such mixed dynamics and purely Brownian dynamics. We conclude by noting that the methods and analysis presented here could potentially improve our current understanding of light-matter interaction, and expand the field of singular optics.

4.6 References

[1] J. W. Goodman, *Speckle Phenomena in Optics Theory and Applications*, Roberts & Co., Englewood, CO, USA [2007].

[2] J. F. Nye, and M. V. Berry," Dislocations in Wave Trains", Proceedings of Royal Society of London A 336, pages 165-190 [1974].

[3] M. S. Soskin, V. N. Gorshkov, M. V. Vasnetsov, J. T. Malos, and N. R. Heckenberg, "Topological charge and angular momentum of light beams carrying optical vortices", Physical Review A 56 (5), pages 4064-4075 [1997].

[4] A. Majumdar, and S. J. Kirkpatrick, "Statistical studies on optical vortices in dynamic speckle fields", Journal of Biomedical Photonics and Engineering 4 (2), 020301 [2018].

[5] R. C. Gonzalez, and R. E. Woods, *Digital Image Processing*, 2nd edition, Prentice Hall [2002].

[6] S. J. Kirkpatrick, K. Khaksari, D. Thomas, and D. D. Duncan, "Optical vortex behavior in dynamic speckle fields", Journal of Biomedical Optics 17 (5) , 050504 [2012].

[7] https://omlc.org/calc/mie_calc.html

[8] A. Einstein, *Investigations on theory of the Brownian Movement*, Dover [1956].

[9] F. Reif, *Fundamentals of Statistical and Thermal Physics*, McGraw-Hill [1965].

[10] A. Majumdar, and S. J. Kirkpatrick, "Spatial Poincaré Plots as Descriptors of Speckle Pattern Second-Order Statistics", Journal of Biomedical Photonics and Engineering 3 (3), 030501 [2017].

[11] D. D. Duncan, S. J. Kirkpatrick, "Can laser speckle flowmetry be made a quantitative tool?" Journal of the Optical Society of America A 25(8), 2088-2094 [2008].

81

5 Other applications of Poincaré Analysis

In chapter 2, we noted that Poincaŕe analysis can be applied to any data series that can be represented in a one-dimensional array. In the chapters 3 and 4, we have seen the application of Poincaré descriptors to analyze the charge separation in dynamic optical vortex distributions. In this chapter, we look into some other aspects of speckle fields where these descriptors, as well as other parameters described in chapter 2, can be applied. Excerpts of this chapter have been taken, with permission, from "Poincaré descriptors of long and short- range correlations in the spatial and temporal fluctuations of coherently scattered light" by A. Majumdar and S.J. Kirkpatrick, published in the Proceedings of SPIE in 2018. Excerpts of this chapter have also been taken, with permission, from "Ellipticity imaging for visualizing and quantifying long and short range correlations in laser speckle data II: phantom and animal studies" by Majumdar *et al.*, published in the Proceedings of SPIE in 2019. The letter of permission for both of them is attached in Appendix A.1. Excerpts of this chapter have also been taken, with permission, from "Spatial Poincaré Plots as Descriptors of Speckle Pattern Second-Order Statistics" by A. Majumdar and S.J. Kirkpatrick, published in the Journal of Biomedical Photonics and Engineering in 2017. The letter of permission is attached in Appendix A.2. The calculations in this chapter have been performed in a MATLAB environment using the code in Appendices B.3, B.4 and B.5.

5.1 Speckle Size measurement

5.1.1 Power Spectral Density (PSD) function

Laser speckle patterns are obtained as a result of the random interference of coherent light scattering from a rough surface or a scattering volume [1]. Speckle patterns have been used extensively for a variety of metrology applications including surface roughness [2], strain

measurement [3] and fluid flow [4-8]. The second-order statistics of a speckle pattern provide a description of the spatial structure of the pattern, with the minimum size of a speckle, or the minimum correlation length of the pattern, being an important parameter in the characterization of this structure [9]. It has been noted in the literature that proper spatial sampling of the intensity of the scattered field is an important experimental consideration in speckle-based metrology applications [10]. In order to meet the spatial Nyquist criteria, the minimum speckle size of the measured intensity pattern should be at least two pixels [8]. Goodman [9] introduced the *M* parameter which can be interpreted as the number of speckles per pixel and may be used as a means of quantifying this. In practice, the minimum speckle size has traditionally been estimated in two dimensions in terms of the width of the power spectral density (PSD) function of the measured intensity distribution [9]

$$PSD = |\Im\{I(x,y)\}|^2 \tag{5.1}$$

where $\Im$ represents the Fourier Transform operator and $I(x,y)$ is the distribution of the measured intensity of the speckle field. The typical plot of the PSD function is shown in Fig 5.1. The minimum speckle size, Λ, is estimated from this distribution as [8,11]

$$\Lambda = \frac{2D}{W_{PSD}} \tag{5.2}$$

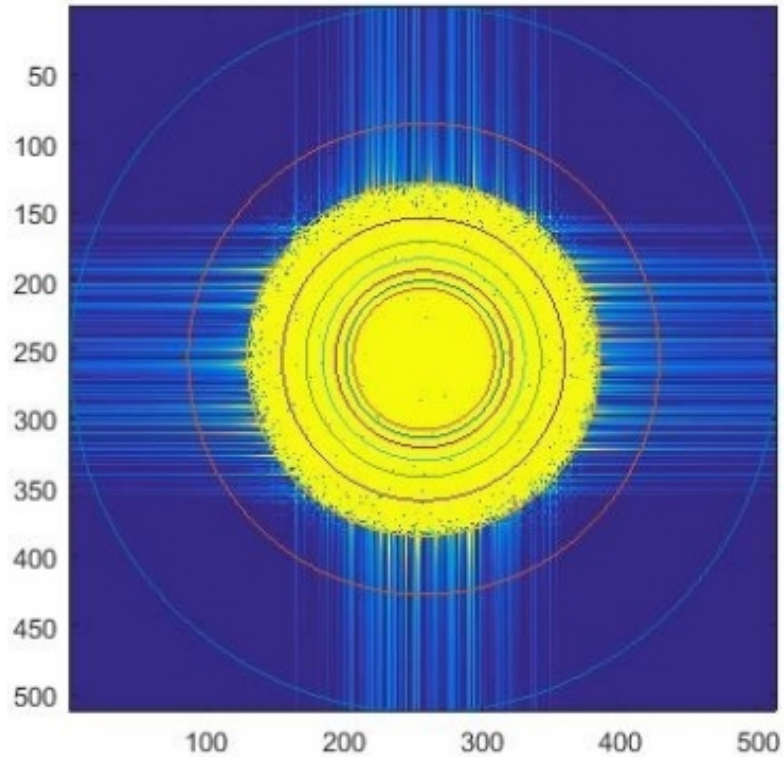where D is the width of the speckle image and W_{PSD} is the width of the PSD function.

83

Fig 5.1. The Power Spectral Density (PSD) of a frame of 512 X 512 pixels in a speckle

field. It can be seen that the PSD width is half the overall width of the field, which in

accordance with (1) gives a speckle size of 4 pixels. For easy visualization, concentric

circles corresponding to each minimum speckle size from 2 to 10 pixels is drawn on the

PSD. The outermost circle corresponds to size=2 pixels and each subsequent circle

inwards corresponds to an increase in size by 1 pixel.

5.1.2 Review of relevant Poincaré analysis details

A Poincaré plot is a statistical tool to study variations, or, alternatively, self-similarity in a

quantity. In these plots, sequential measures of the quantity of interest are plotted against

the previous measures. Thus, the ith measure is plotted against the $(i + 1)^{th}$ measure, for

i=1,2...N-1, where N is the total number of measurements available in the data. In

84

biomedical signal processing, these plots have been used extensively to study heart rate variability over time [12]. Specific standard descriptors $SD1$ and $SD2$ can be obtained from the data. These descriptors are mathematically defined as

$$SD1 = Var(\frac{1}{\sqrt{2}}X_n - \frac{1}{\sqrt{2}}X_{n+1})^{\frac{1}{2}} = \frac{1}{\sqrt{2}}\{Var(X_n - X_{n+1})\}^{\frac{1}{2}}$$

$$SD2 = \{2Var(X) - SD1^2\}^{\frac{1}{2}}$$

(5.3)

where X_{n+1} and X_n represent the current and previous measures, respectively and $Var(X)$ stands for the variance in the measurement of *X*. As can be seen from Eq. (3), *SD1* is a measure of the standard deviation of the differences in successive data points. This is commonly known as the standard deviation in successive differences in statistics [12]. Thus, the descriptor *SD1* represents the short-term variations in the data [12]. Also, from the expression for *SD2*, it is noted that this represents the difference between the overall variations in the data, *Var(X)*, and the variations attributed to short-term differences, *SD1*. Thus, the descriptor *SD2* effectively represents the long-term variations in the data. By these definitions, the ratio SD2/SD1 is an indication of the prominence of long-term variations in the data, compared to short-term variations [12]. We have referred to this ratio as the ellipticity *e* of the data. We have chosen this nomenclature, as the descriptors SD2 and SD1 are the semi-major and semi-minor axes, respectively, of an ellipse commonly used to study information from these plots [12] (Fig 5.2).
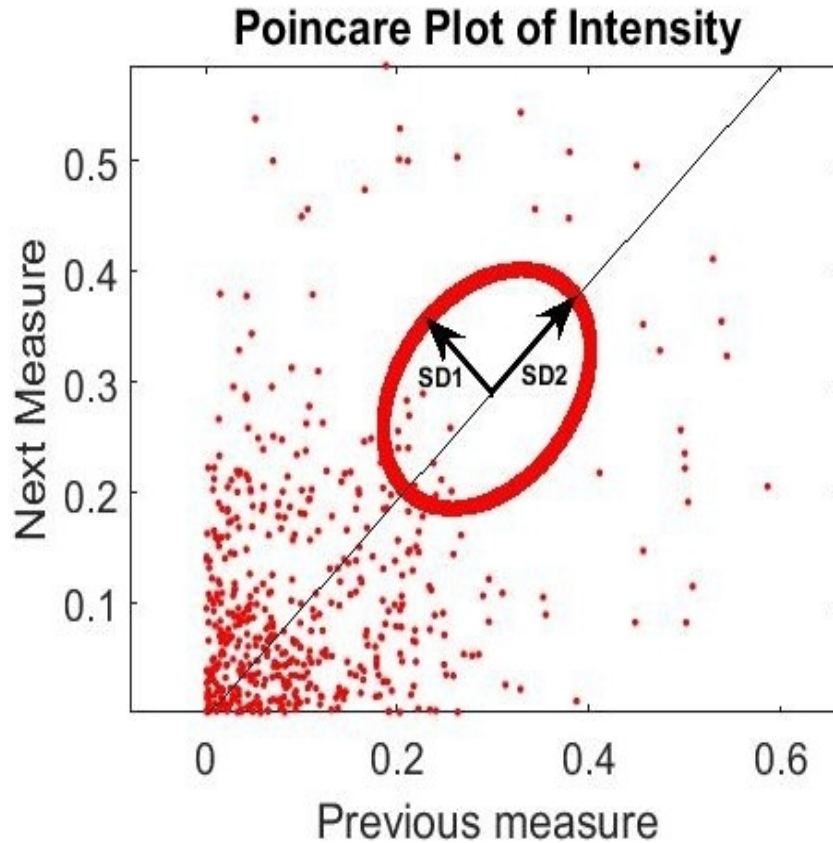
85

Fig. 5.2. A typical Poincaré plot. Any particular point represents a data point plotted in vertical axis against the previous data point in the horizontal axis.

As demonstrated by Brennan, Palaniswami and Kamen [12], and also described in chapter 2, for uncorrelated data, this ratio is unity. The larger the deviation of this ratio is from unity, the greater the correlation among the data points. In our research, we have made use of this ability of the Poincaré plot descriptors to indicate whether or not the intensity in neighboring pixels is correlated, which ultimately reflects the minimum speckle size in the pattern.

86

5.1.3 Necessary modifications for speckle size calculations

In order to take advantage of the ability or Poincaré plots to demonstrate correlation between neighboring data points, the traditional definition of Poincaré plot descriptors were modified for the purposes of this study. Instead of plotting every $(i + 1)^{th}$ measurement against the ith measurement, we introduced a coarsing factor k, such that every $(i + k)^{th}$ measurement was plotted against the ith measurement, for $i = 1,2,…N\text{-}k$, with N being the number of data points (pixels) available. In previous publications [12], similar descriptors have been referred to as *lag m* Poincaré descriptors. Thus, similar to Eq. 5.3, the modified descriptors are defined as

$$SD1_k = Var(\frac{1}{\sqrt{2}} X_n - \frac{1}{\sqrt{2}} X_{n+k})^{\frac{1}{2}}$$

$$SD2_k = \left\{ 2Var\left(X \right) - SD1_k^{\,2} \right\}^{\frac{1}{2}}$$

(5.4)

### 5.1.4 Ellipticity in simulated speckle fields and comparison with PSD

Speckle fields were generated using an algorithm as described by Duncan and Kirkpatrick [13]. Using this algorithm, we generated speckle patterns with known minimum speckle sizes varying between 2 and 9 pixels. The intensity distribution of individual rows in these speckle patterns was examined. It has been shown previously [8] that the intensity PSD width accurately gives the minimum speckle size input by the user while implementing this algorithm. Our objective herein was to use modified spatial Poincaré plot descriptors of the intensity distribution along a slice (single row) in the speckle pattern to indicate correlation

87

lengths along this row of speckles and ultimately describe the spatial structure (*i.e,* the minimum speckle size) of the speckle pattern.

The ellipticity *e* of the obtained intensity distribution was measured for different coarsing factors. In all further discussion, the ellipticity calculated from the data while setting the coarsing factor as k is denoted as e_k. Thus, $e_k = SD2_k / SD1_k$. The ellipticity decreased steadily with an increase in the coarsing factor, until it reached a value approaching unity for all speckle sizes. As mentioned above, this is the point where the distribution is seen to be uncorrelated. By further increasing the coarsing factor beyond this point, the ellipticity exhibited small oscillations, but remained approximately equal to 1.0.
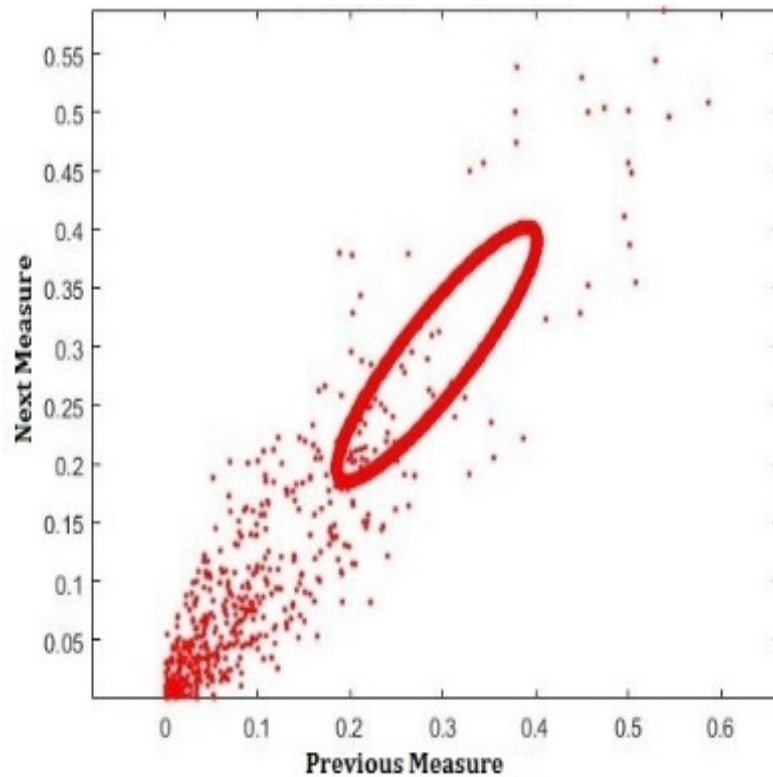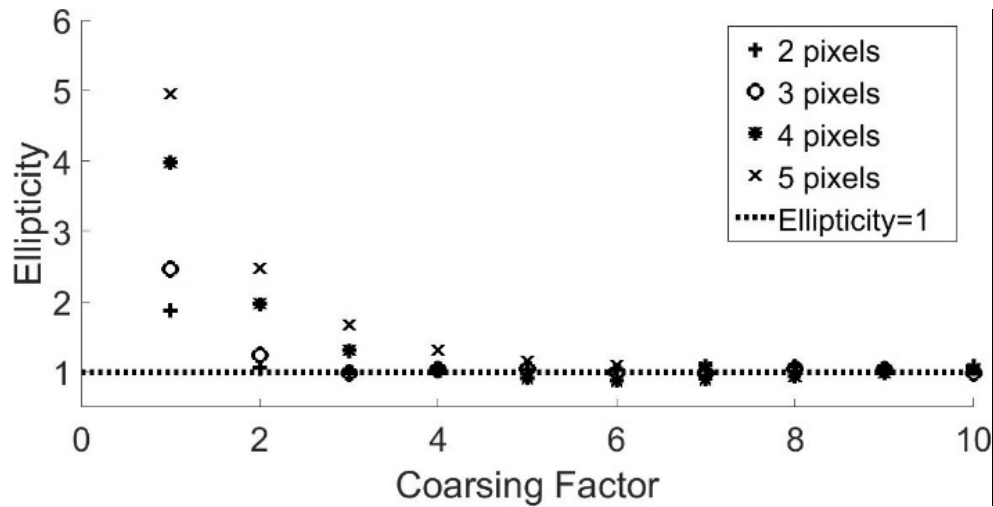

(a)
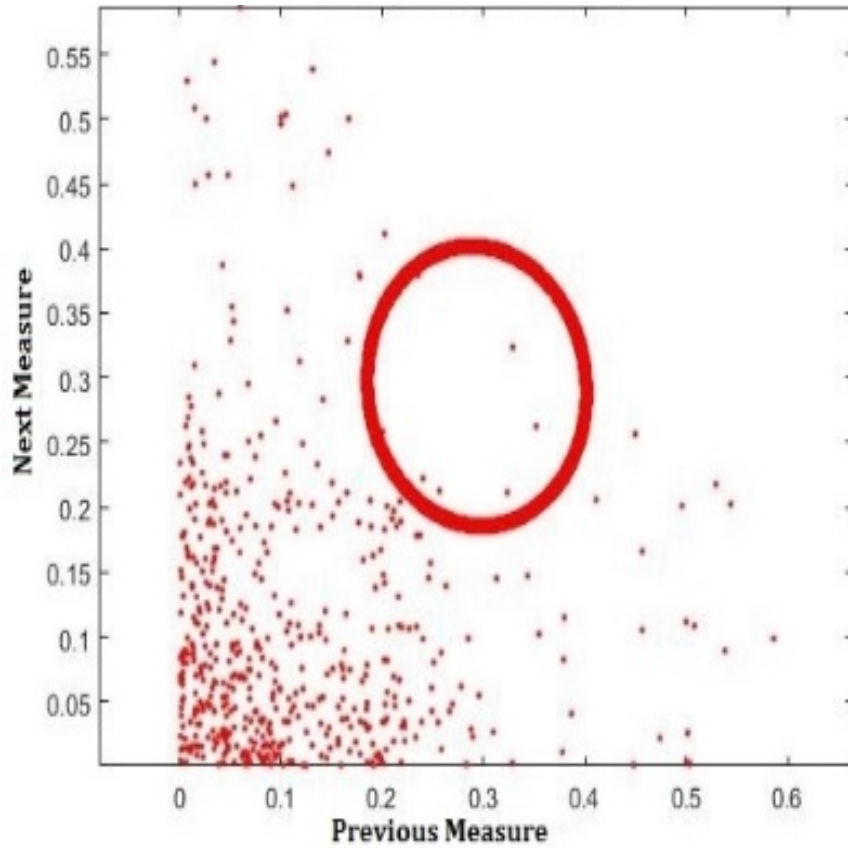
(b)

Fig. 5.3. Poincaré plot of intensity for minimum speckle size of 5 pixels. (a) When the coarsing factor was 1 pixel, we obtained SD1 = 0.0344 and SD2= 0.1490, giving an ellipticity factor of 4.3314 (b) When the coarsing factor was 6 pixels, we obtained SD1 = 0.1105 and SD2= 0.1058, giving an ellipticity factor of 0.9575. Thus, the observed correlation is lost at high coarsing factors.

89

(a)
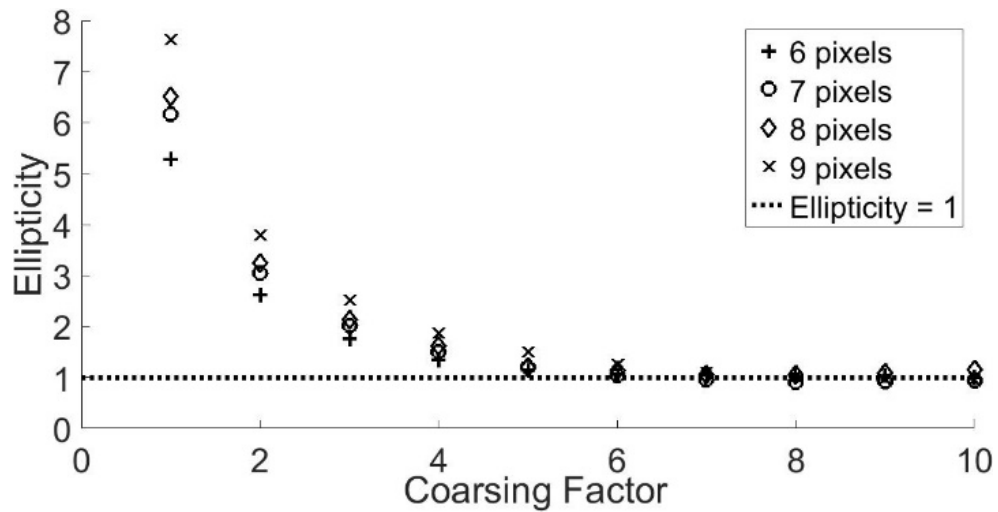
(b)

Fig 5.4. Dependence of measured ellipticity on the coarsing factor for minimum speckle sizes of (a) 2 to 5 pixels (b) 6 to 9 pixels. It can be noted that the ellipticity value tends to asymptotically settle around unity, for each of the speckle sizes. We call this asymptotic regime as the uncorrelated regime.

90

We observed that on steadily increasing the coarsing factor for the calculation of the standard descriptors, the correlation seen in the intensity data was lost when the coarsing factor equals or exceeds the pre-determined minimum speckle size. For example, by examining the case where the minimum speckle size in the simulated speckle pattern was 5 pixels in the data of Fig 5.4, it was observed that as long as the coarsing factor was smaller than 5 pixels, the descriptors were able to detect a correlation between the observed intensity values. This results in ellipticity values greater than unity. We refer to this region as the correlated regime. However, when the coarsing factor became larger than 5 pixels, the intensity values detected were seen to be uncorrelated. This resulted in ellipticity values close to unity. We refer to this region as the uncorrelated regime. The minimum speckle size can thus be viewed as the coarsing factor at the transition point between the correlated and uncorrelated regimes.

It is clear from the above discussion that a correlation exists between the minimum speckle size in the speckle pattern and the value of ellipticity at a particular coarsing factor. We observed that for any particular coarsing factor, larger speckle sizes resulted in higher values of ellipticity, unless we are in the uncorrelated regime, in which case $e_k \approx 1.0$. The characteristic feature of the uncorrelated regime is that the values of ellipticity lie very close to unity and do not continue to decrease monotonically with an increase in coarsing factor. Thus, speckle sizes from multiple speckle fields can be compared by comparing their ellipticity in their intensity distribution, using a low coarsing factor (such as k =1, which lies in correlated regime irrespective of the speckle size).

91

In fact, this ability of the spatial Poincaré plots to distinguish between speckle sizes at low coarsing factors gains significance, especially at larger minimum speckle sizes. Figures 5.5(a) and 5.5(b) demonstrate the challenge with using the PSD for comparing speckle patterns with minimum speckle sizes of 9 and 10 pixels, respectively. According to Eq. (5.2), the width of the PSD function for the patterns on a 512 X 512 window, are roughly 114 and 102 pixels, respectively. This translates to the radius of the functions to be 57 and 51 pixels. This difference of 6 pixels on a field of 512 X 512 pixels is more challenging to ascertain, compared to the approach using spatial Poincaré plots, as demonstrated in Fig 5.5(c).

(a)

(b)

93
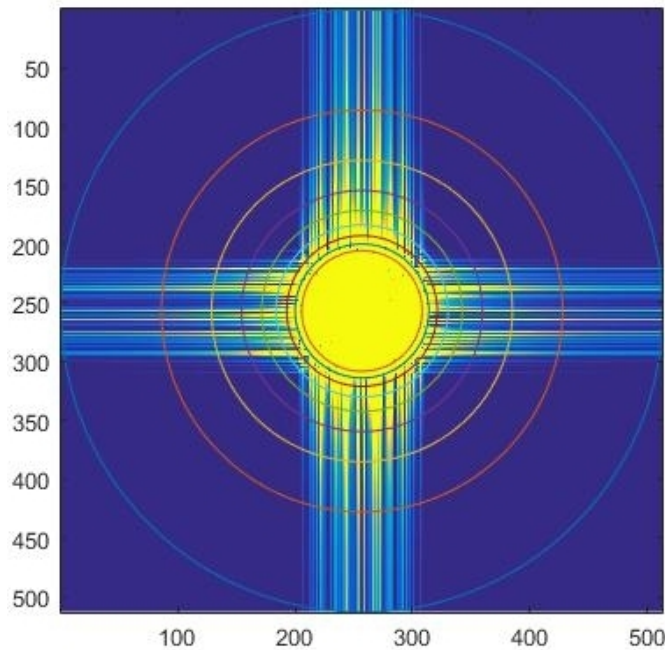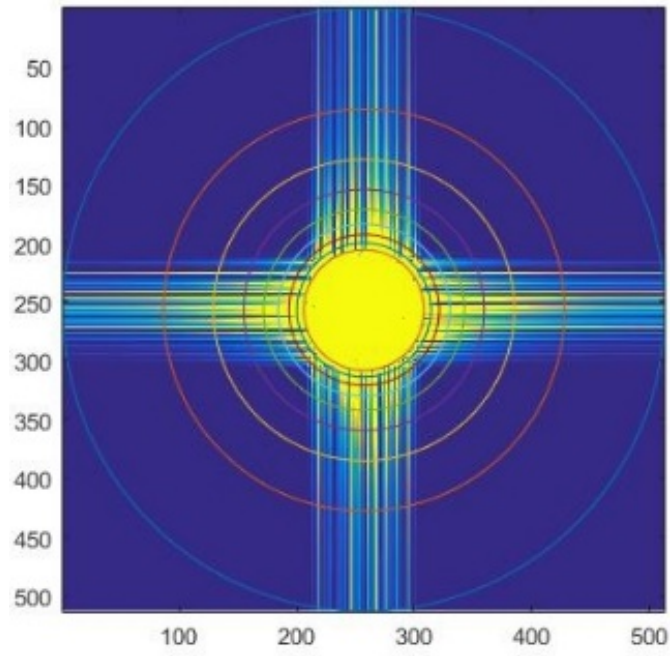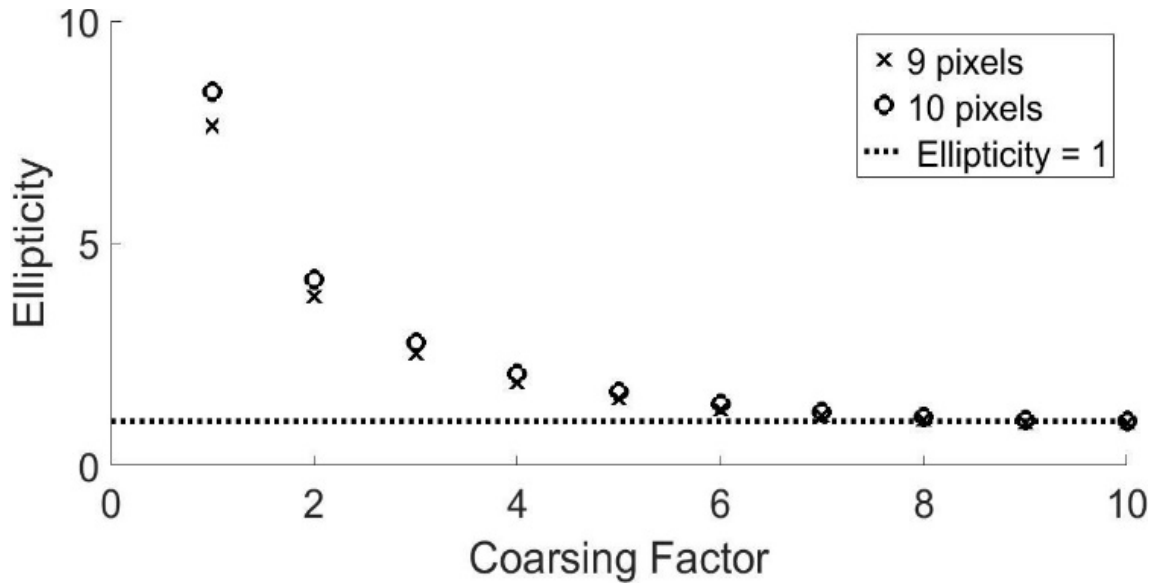
(c)

Fig. 5.5. Comparison of ease of detecting differences in systems with large minimum speckle sizes: PSD for a minimum speckle size of (a) 9 pixels and (b) 10 pixels compared to the (c) ellipticity values for the same minimum speckle sizes.

For a coarsing factor of $k=1$, which lies in the correlated regime for all speckle sizes greater than 1 pixel, the calculated values of e_k for the speckle patterns of 9 and 10 pixels minimum speckle sizes were 7.63 and 8.41 respectively. For k=2, these values dropped to 3.80 and 4.18, respectively. Thus, for large speckles, calculating the ellipticity value using a coursing factor of 1 pixel, for example, provides a method of easier comparison than the visual determination of PSD width.

The decrease in ellipticity values with increasing coarsing factor roughly follows an inverse power relation

94

$$e_m = A + \frac{B}{m^C} \tag{5.5}$$

where e_m is the ellipticity measured at coarsing factor m. A, B and C are constants. This is further demonstrated in the work that follows.

We next calculated the ellipticity values along the time-axis. In other words, the intensity variation of the same pixel was recorded as the speckle fields evolved through 1000 frames. The calculated ellipticity values from this data series is plotted in Fig 5.6 for different decorrelation rates. These decorrelation rates are defined in terms of the number of frames it takes for the overall speckle correlation to reduce to a value of $1/e$. Four different rates were considered, with their correlation factor reducing to the above value in 211, 106, 76 and 55 frames, during the 1000-frame simulation. These decorrelation rates shall be called R1, R2, R3 and R4 respectively, for the rest of this discussion. This is also the order of the rate of decorrelation, from the slowest to the fastest. All the speckle frames were simulated with a minimum speckle size of 3 pixels. Smooth curve-fitting plots of Eq. (5.5) was also done on these values and the obtained values for the constants in the equation are provided.
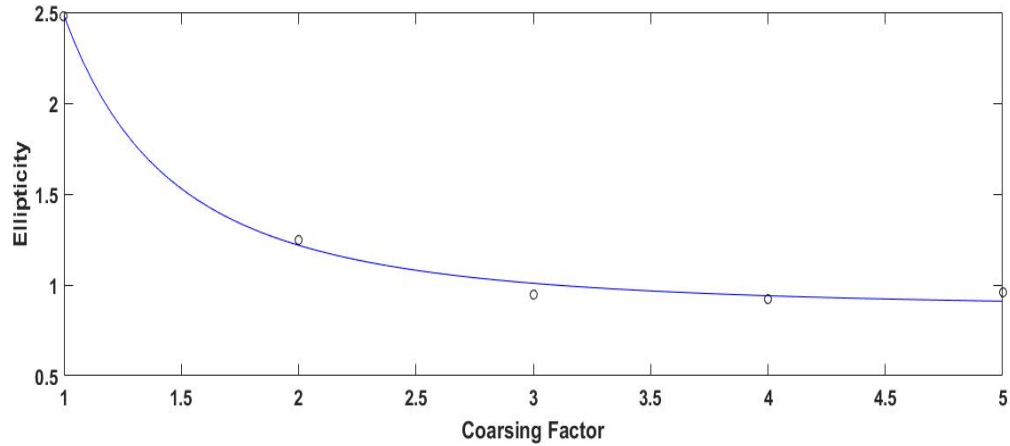

(a)

95

(b)

(c)

96
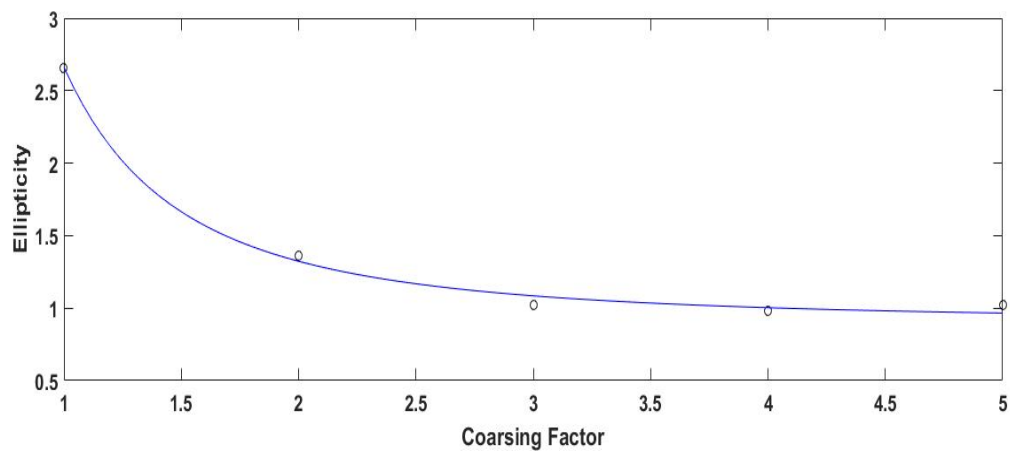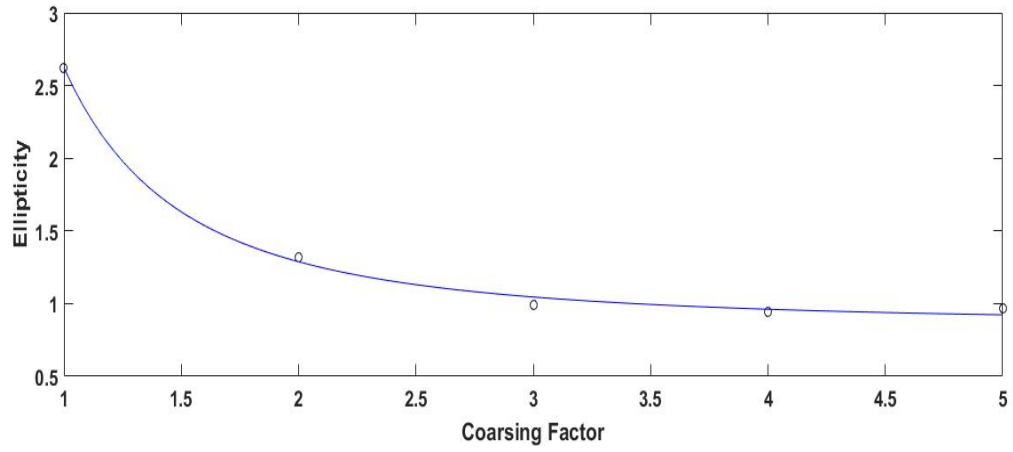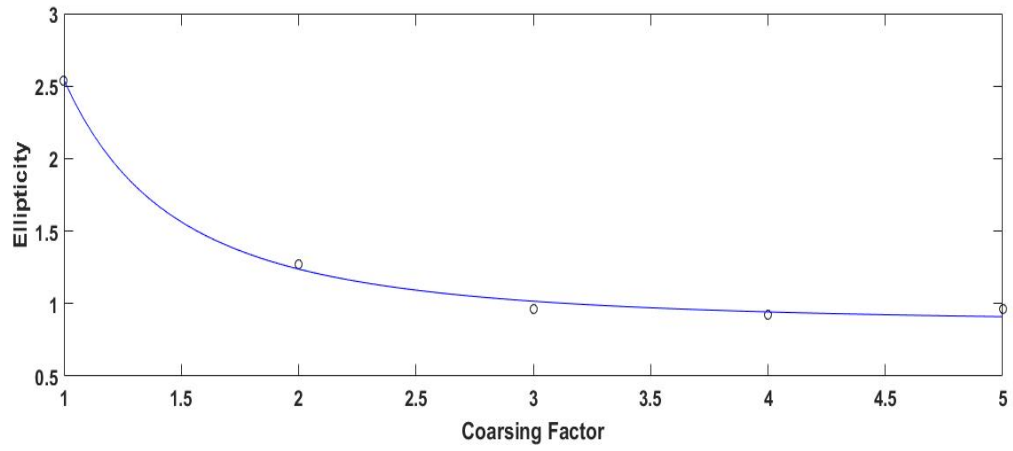
(d)

Fig 5.6. Ellipticity values plotted against the coarsing factor at different rates of decorrelation, along with smooth curve-fitting in accordance with Eq. (5.5). The constants A,B and C, respectively for each of the decorrelation rates are: (a) R1: 0.9026,1.7601,2.0644; (b) R2: 0.8555,1.7669,2.0302; (c) R3: 0.8564,1.6859,2.1421; (d) R4: 0.8629,1.6197,2.1873

It is noted that the variation of ellipticity with coarsing factor follows a similar pattern in the temporal direction above, as it does in the spatial directions. A more interesting observation is obtained on comparing the ellipticity values for different rates of decorrelation (Fig 5.7).
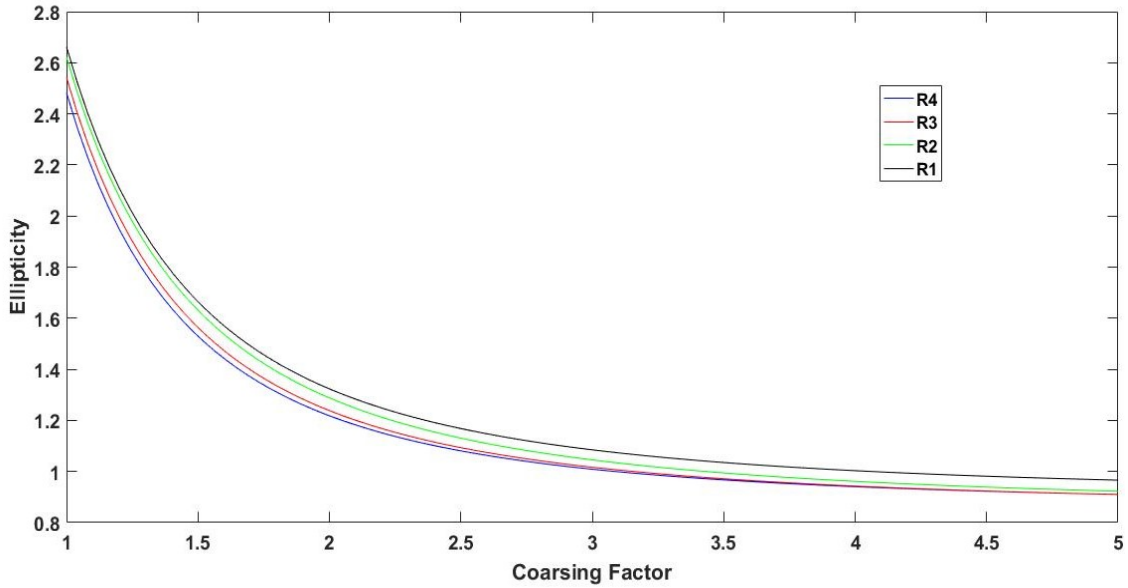
97

Fig 5.7. Obtained ellipticity curve when Eq (5.5) is plotted for each of the decorrelation
rates, with the constants A, B and C as mentioned in Fig 5.6.

Analytically, we can calculate the values of e_m at any coarsing factor from Eq (5.5). For
example, for m=1, i.e. the case of standard Poincaré descriptors, for each rate of
decorrelation. In that case, the values are simply $'A+B'$. We note that these equal 2.6627,
2.6224, 2.5423 and 2.4821 for R1, R2, R3 and R4, respectively. It can be observed that,
for any coarsing factor, the ellipticity decreased as the rate of decorrelation increased. This
result can be used to quantify motion and thus might provide new insights into speckle
imaging techniques.

Next, we considered the tine-averaging of speckle patterns. For this study, we generated a
series of 6000 speckle frames, each frame being 512X512 pixels. This was followed by the
addition of these speckle patterns, frame by frame. We noted that as expected from theory

98

[14], the contrast of the resultant speckle pattern reduced as more frames were added (Fig 5.8).
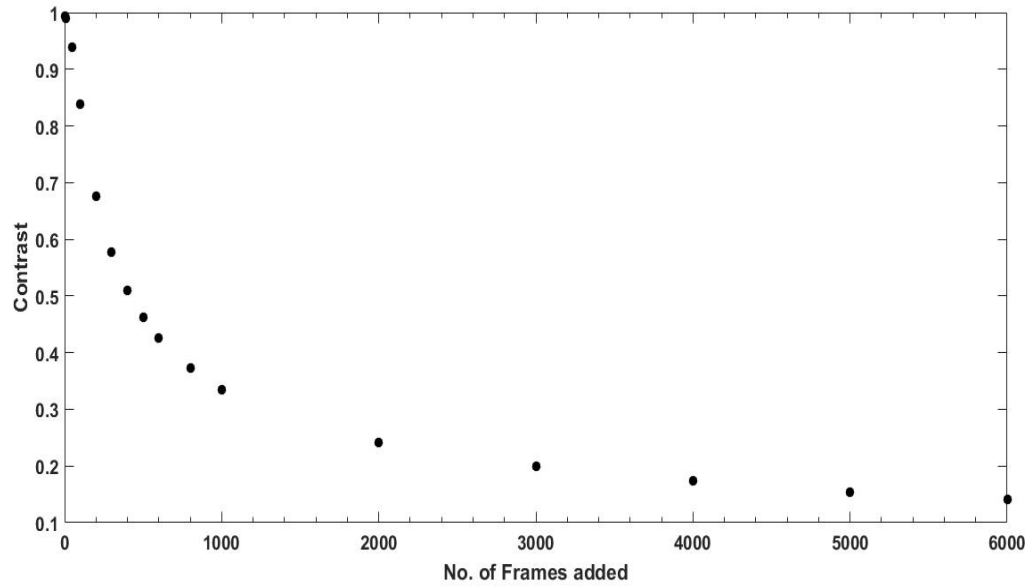
Fig 5.8. Reduction in speckle contrast of the resultant frame, as the number of frames added to obtain it increased

This decrease in contrast can be visually noticed from the resultant pattern. Fig 5.9 demonstrates the resultant patterns obtained, as well as the probability density function of the intensity of the 512X512 pixels, for some of the time-averaged stages of the speckle pattern.

www.manaraa.com

(a)

100

(b)

(c)

(d)

101

(e)

(f)

102

(g)

(h)

Fig 5.9(A)

103

(a)

(b)

(c)

104

(d)

(e)

(f)

105
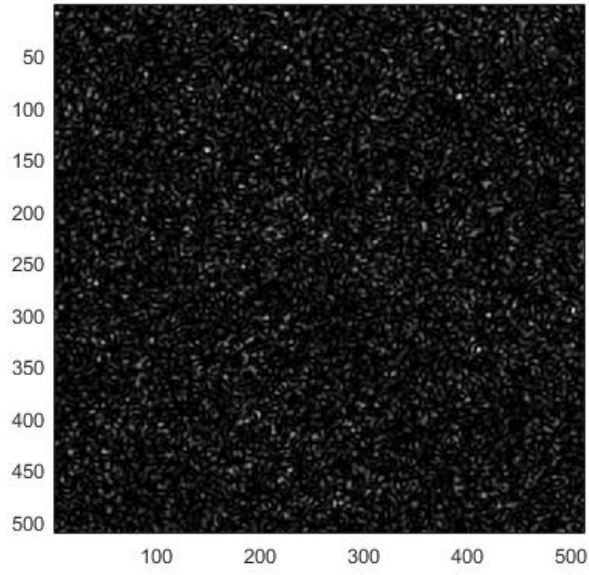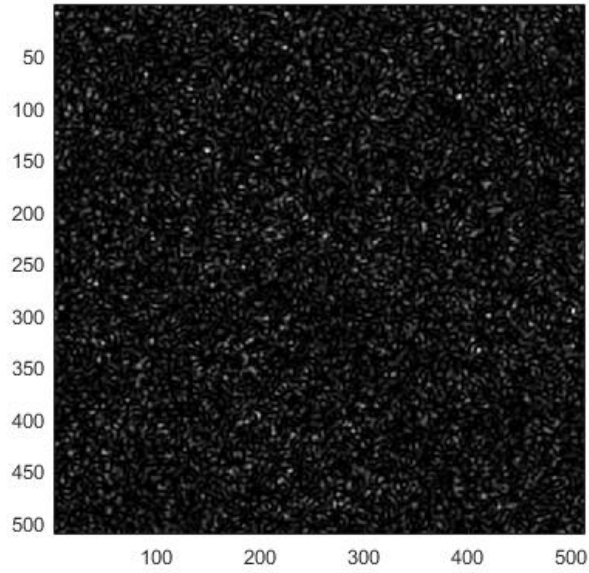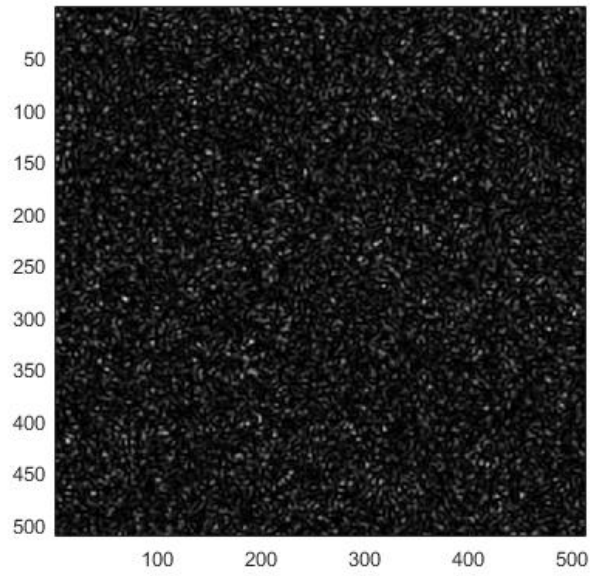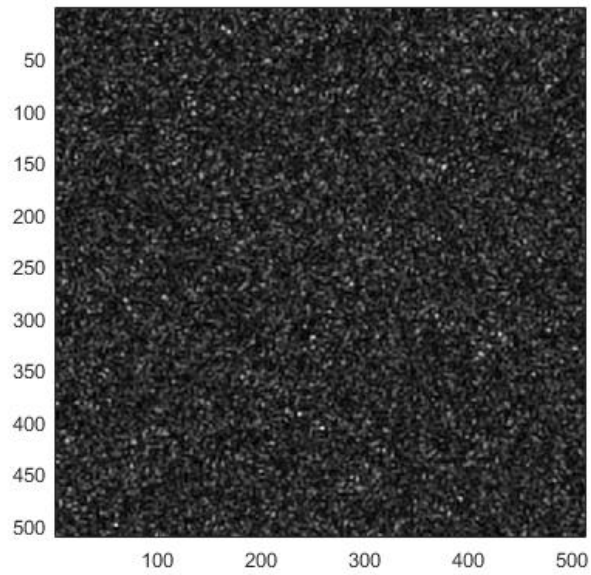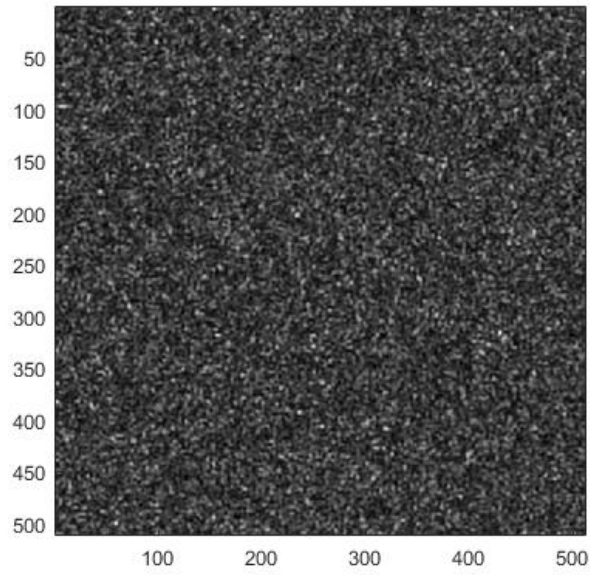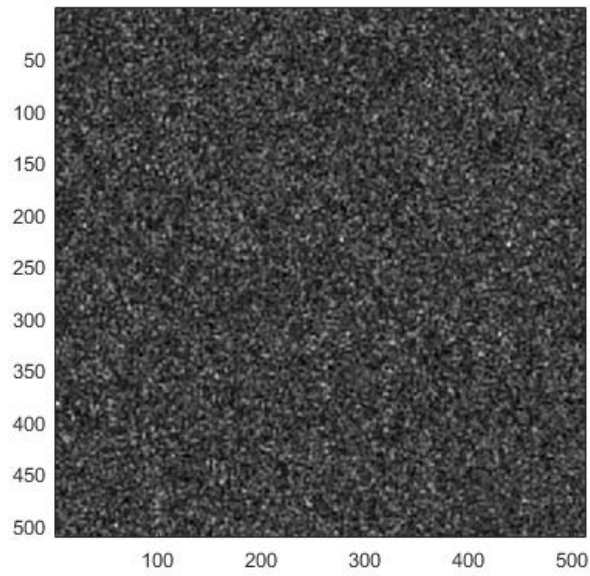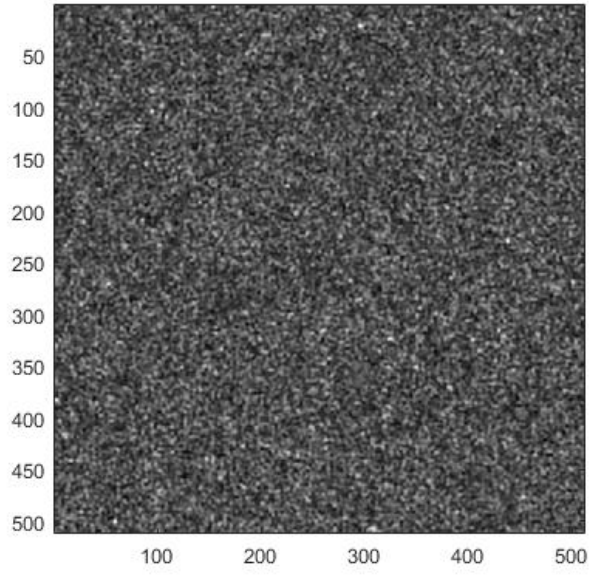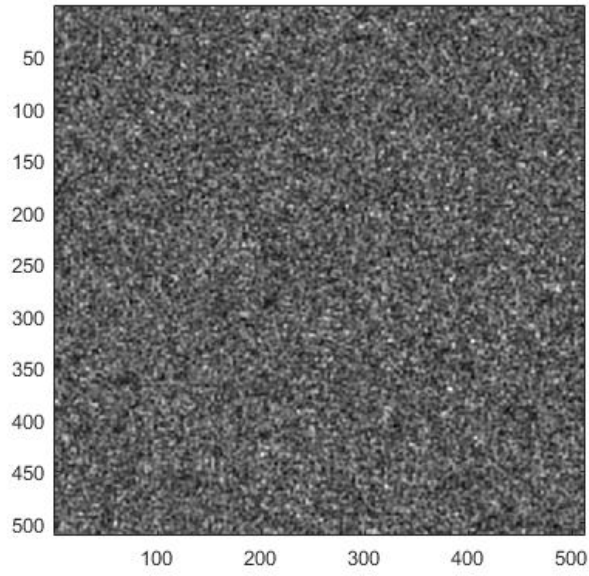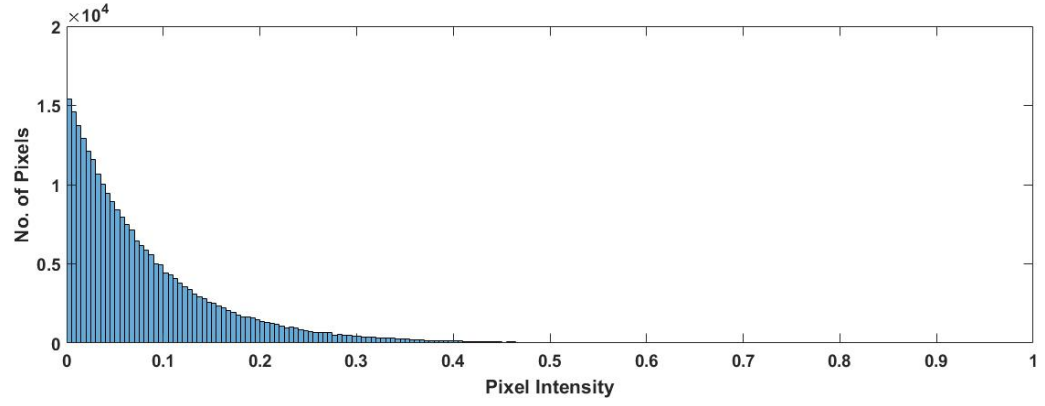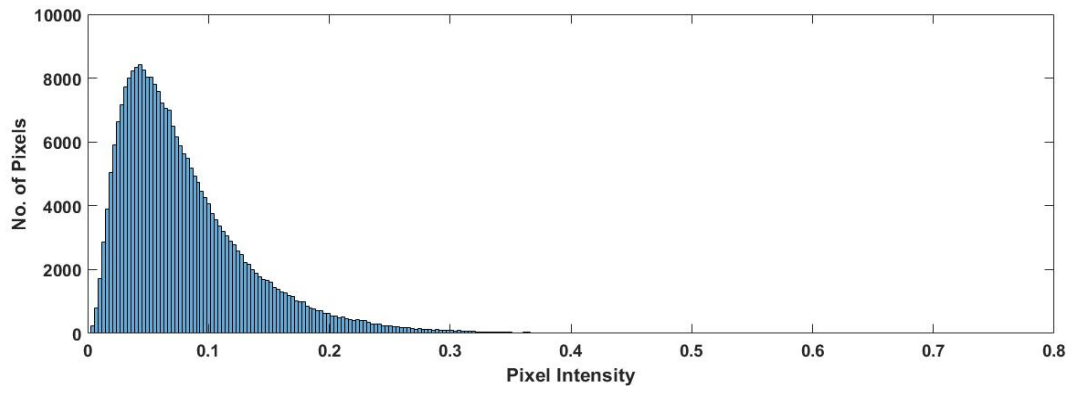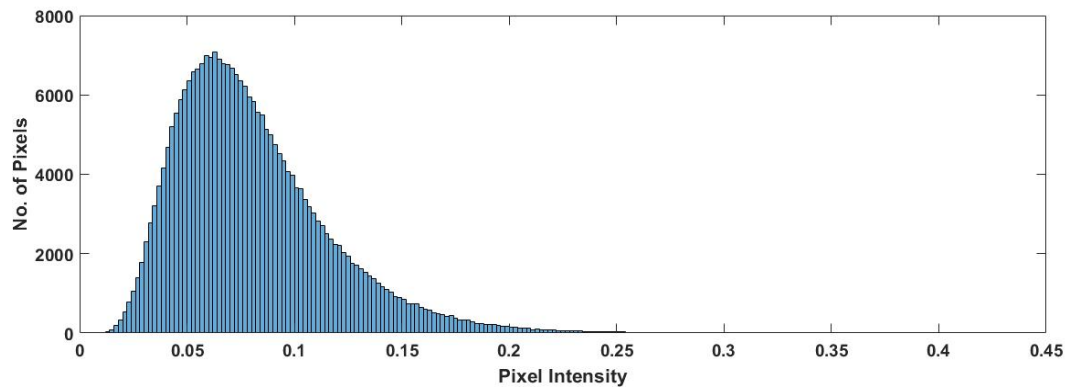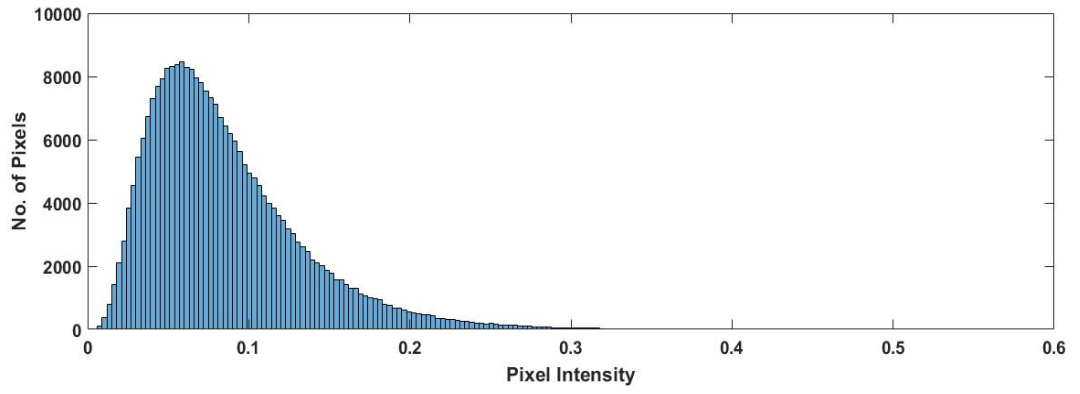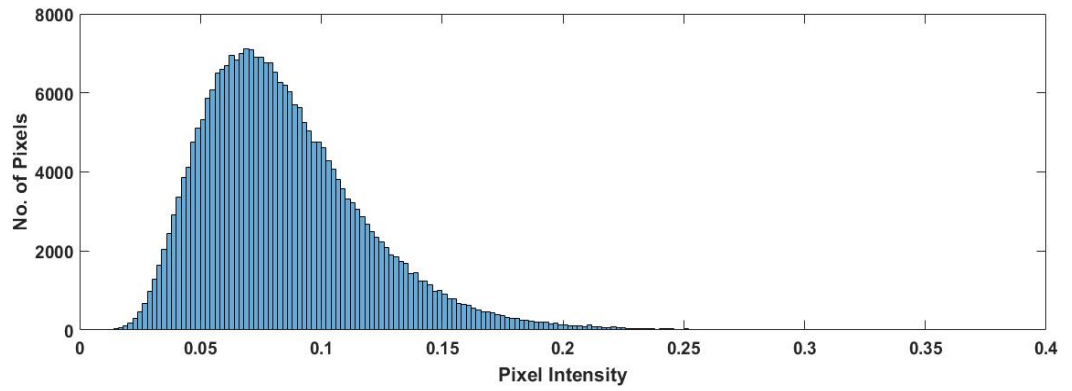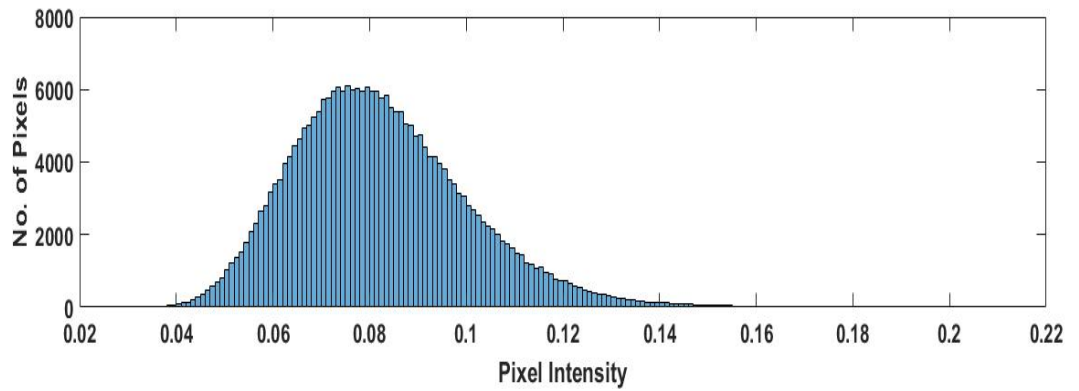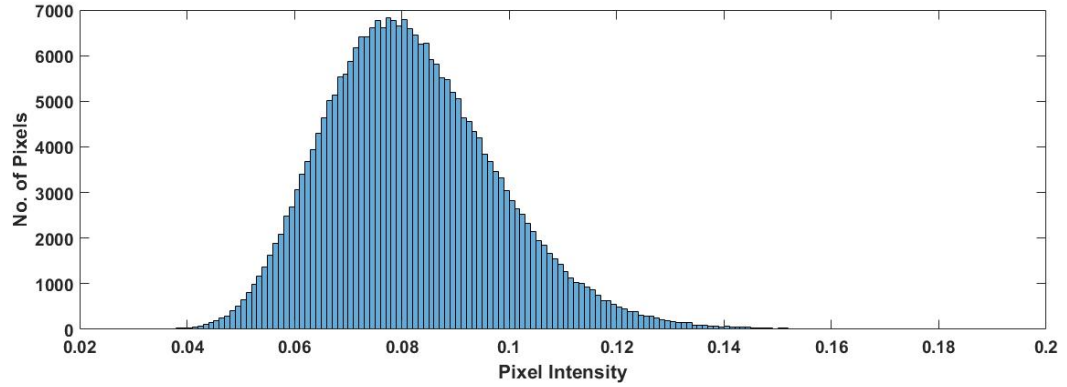
(g)
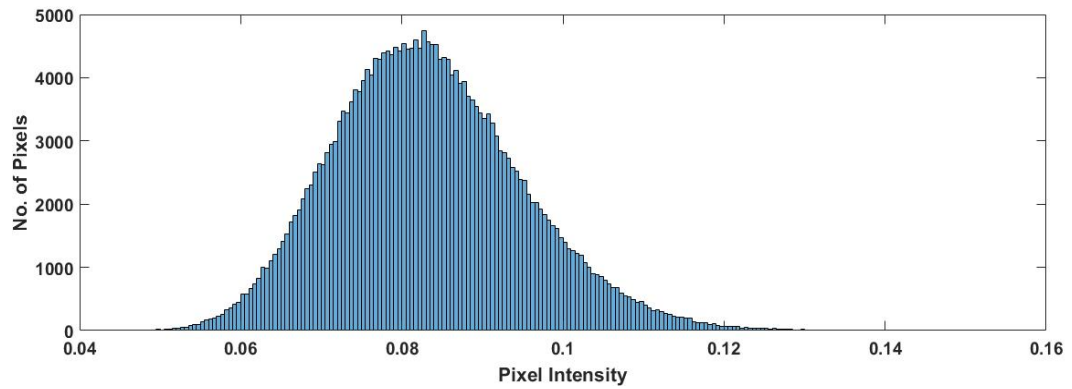
(h)

Fig 5.9(B)

Fig 5.9. The (A) speckle patterns and (B) intensity probabiity distribution functions for the resultant frame, obtained from the addition of (a)1 (b)10 (c)100 (d)500 (e)1000 (f)2000 (g)3000 and (h)6000 individual speckle patterns. The speckle contrast of each of the patterns in Fig 5.9(A) are, in order from (a) to (h): 0.9927, 0.9895, 0.8392, 0.463, 0.3343, 0.2415, 0.2002 and 0.1411

The corresponding power spectral density (PSD) plots are in Fig 5.10.

106

(a)

(b)

107

(c)

(d)

108

(e)

109

(f)

110

(g)

111
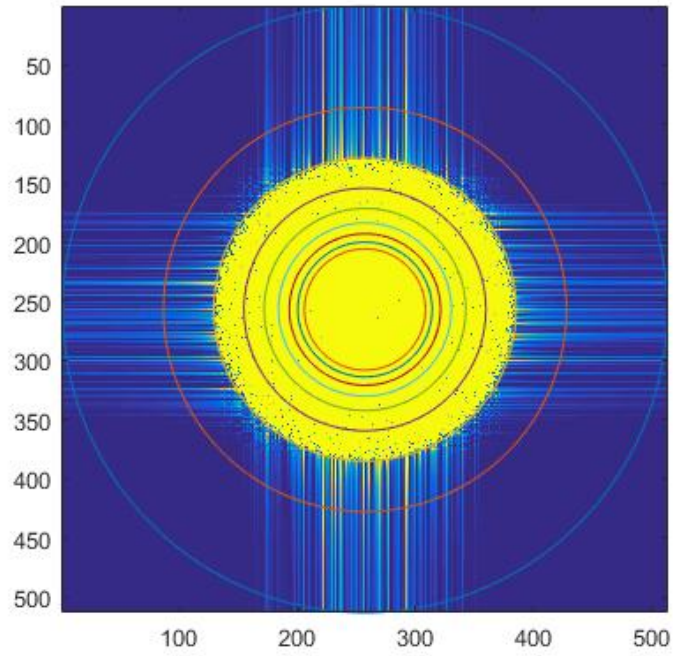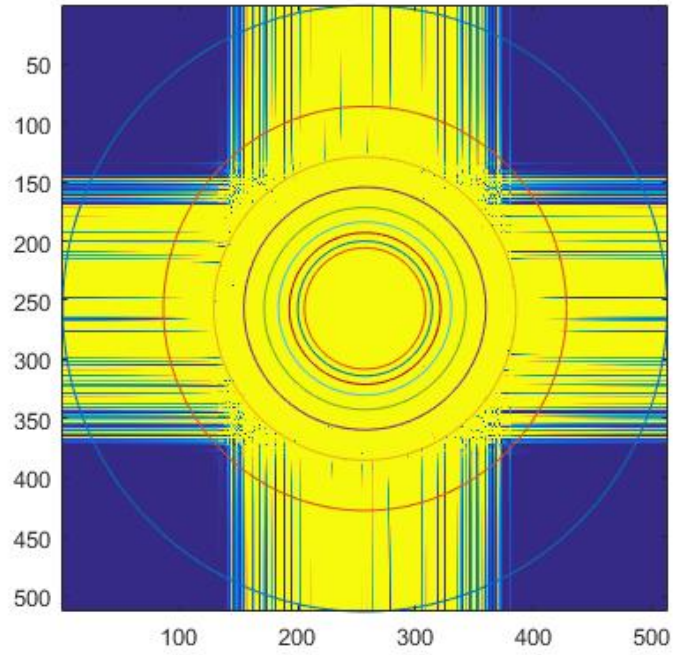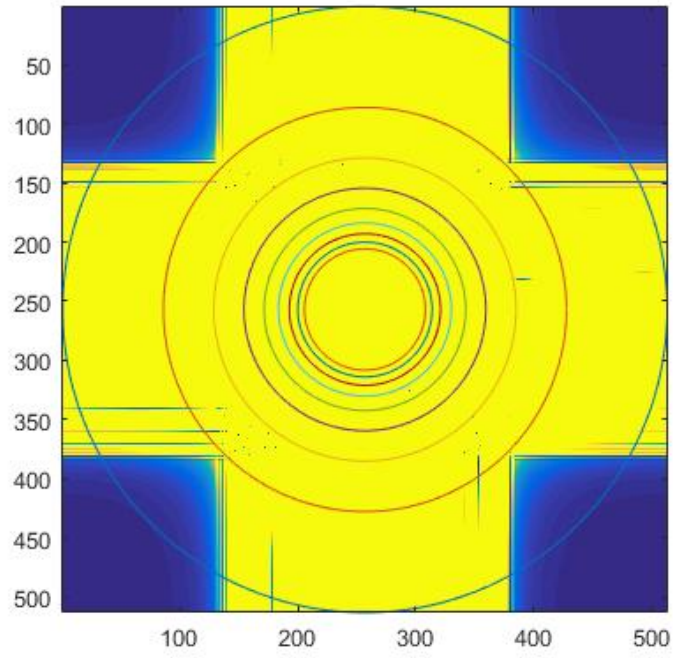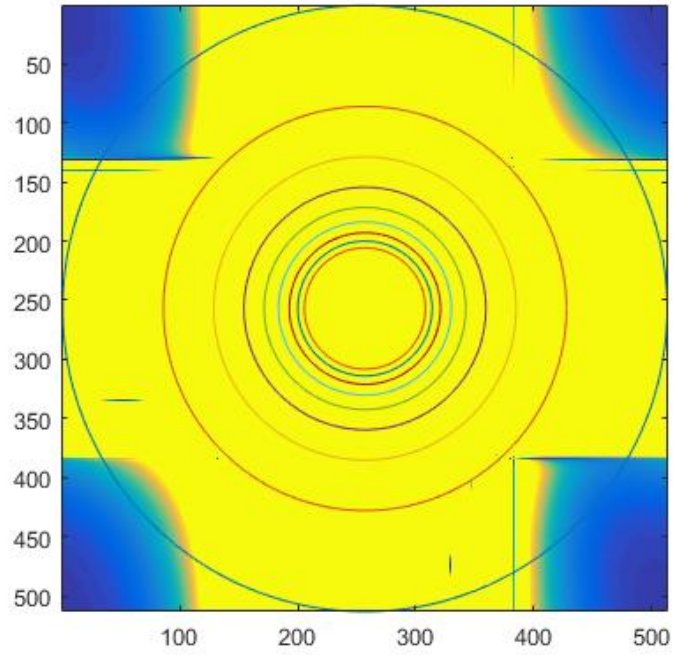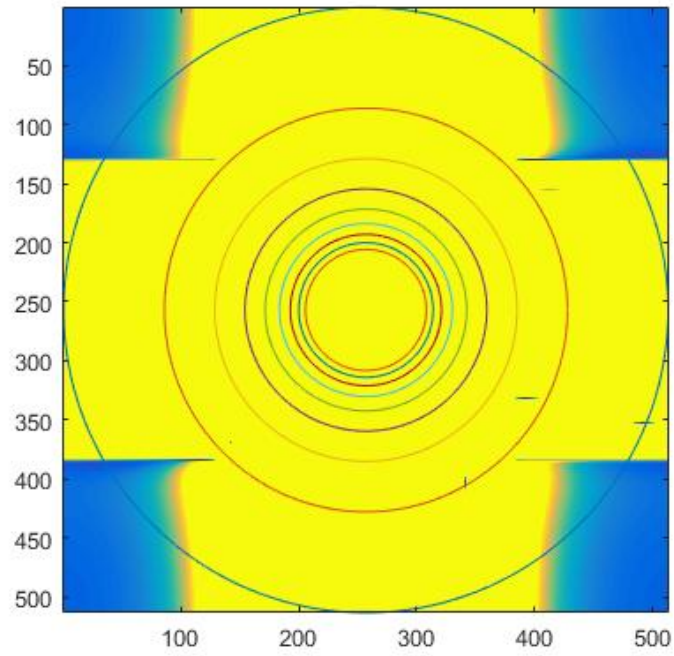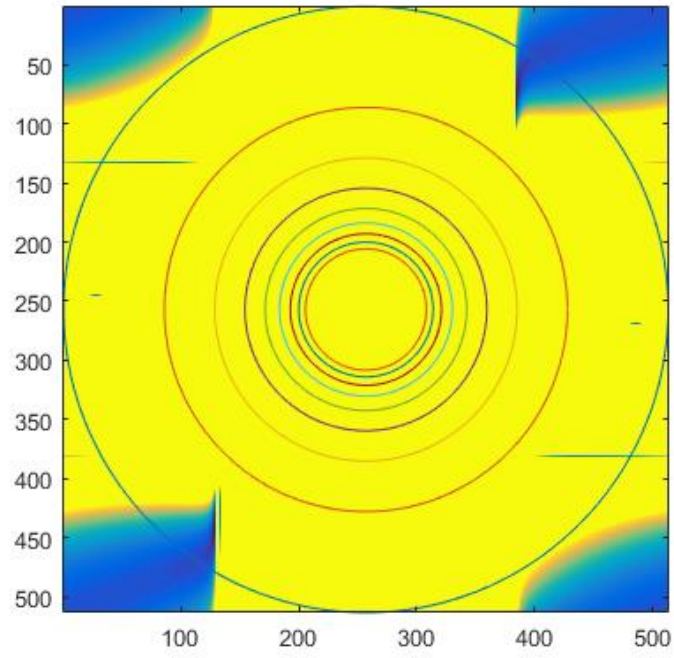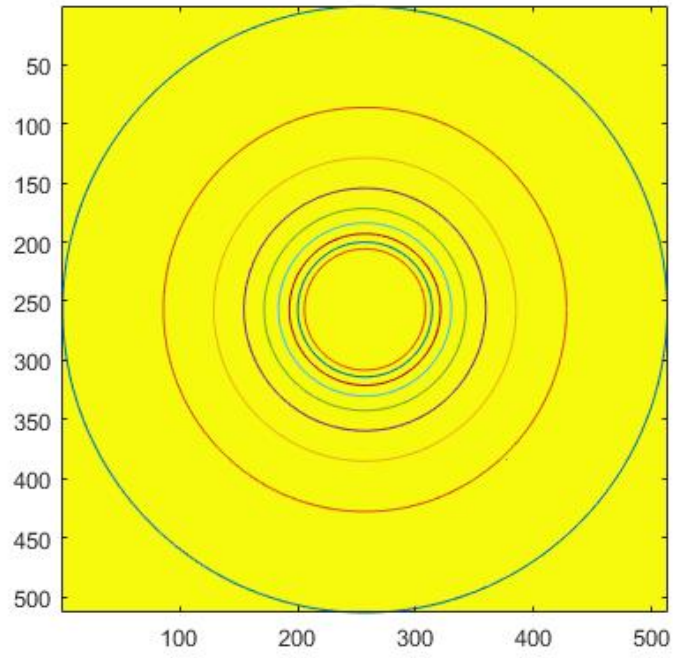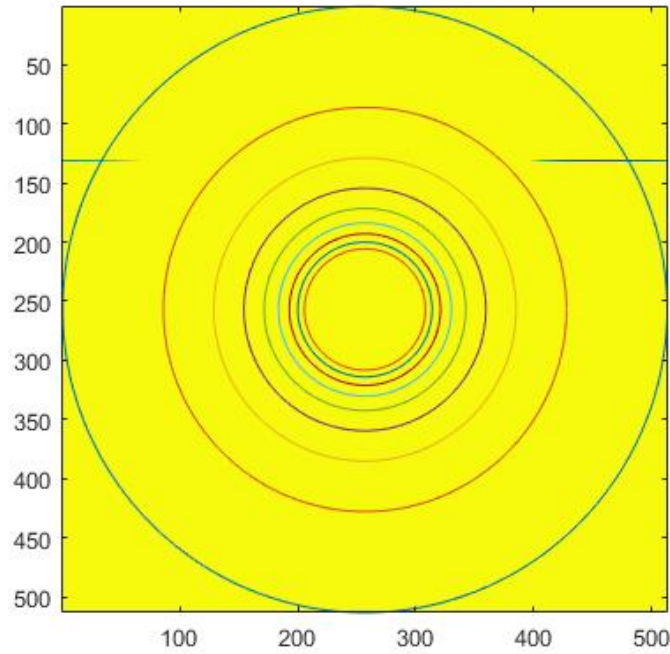
(h)

Fig 5.10. (a)-(h) Power spectral density function plotted for the corresponding speckle

patterns of Fig 5.9(A).

It can be noted that the clarity of the width of the PSD function reduces as the speckle

contrast reduces, with the addition of more speckle frames. While the initial images (up to

1000 frame addition) clearly indicate the minimum speckle size of 4 pixels, the latter mages

do not have a similar well-defined boundary. In such a situation, the application of Poincaré

descriptors to estimate the speckle size was found to be an useful alternative. The ellipticity

factors measured from the above patterns are noted in Table 5.1.

112

Coarsing Factor	No. of Speckle fields added to obtain the resultant frame							
	1	10	100	500	1000	2000	3000	6000
1	3.58	3.56	3.56	3.55	3.57	3.54	3.54	3.56
2	1.78	1.76	1.77	1.76	1.77	1.76	1.75	1.77
3	1.22	1.21	1.21	1.21	1.22	1.21	1.21	1.21
4	1.03	1.02	1.02	1.02	1.02	1.02	1.01	1.02
5	1.00	0.99	0.99	0.99	0.99	0.98	0.99	0.99
6	1.01	1.01	1.01	1.01	1.01	1.01	1.02	1.01

Table 5.1. Ellipticity values for time-averaged speckle patterns do not vary with the number of patterns that are used in the averaging process. These values are typical of a speckle pattern with a minimum speckle size of 4 pixels.

As can be seen, the values do not undergo any major change with a depreciation of the speckle contrast. Thus, these descriptors provide a robust method to estimate the spatial structure, even if the recorded intensity is time-averaged.

Finally, we discuss the ability of Poincaré descriptors to indicate the existence of elongated speckles. These speckle patterns have different correlation lengths in different directions. Fig 5.11 demonstrates a couple of examples of such patterns.

113
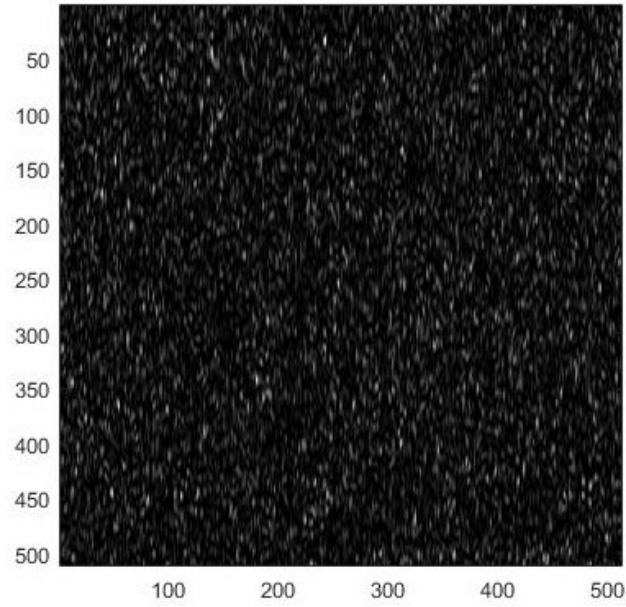
(a)
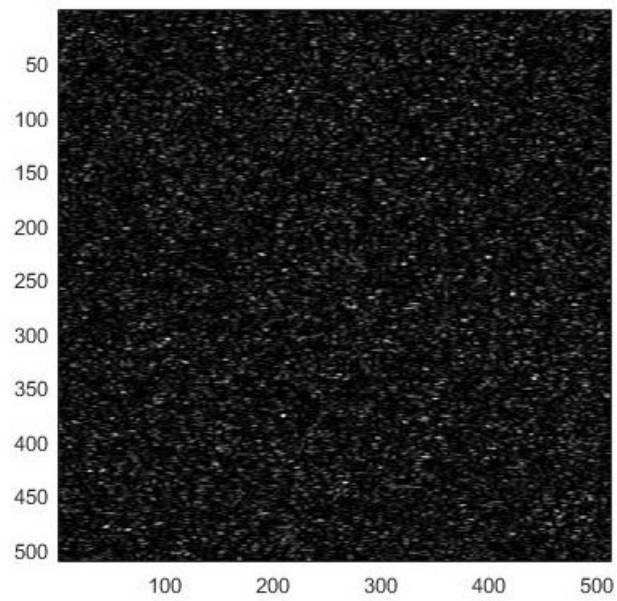
(b)

Fig 5.11. Elongated speckle patterns: (a) correlation lengths in x- and y- directions are 2

and 8 pixels, respectively; (b) correlation lengths in x- and y- directions are 5 and 2

pixels, respectively

114

The power spectral density functions of these patterns are plotted in Fig 5.12 and their ellipticity measurements are displayed in Table 5.2.

(a)

115
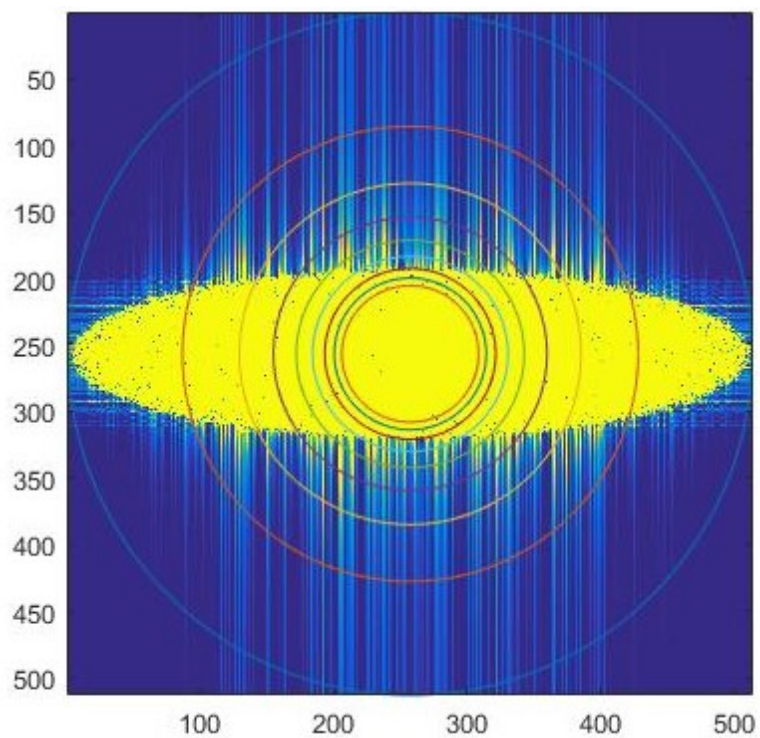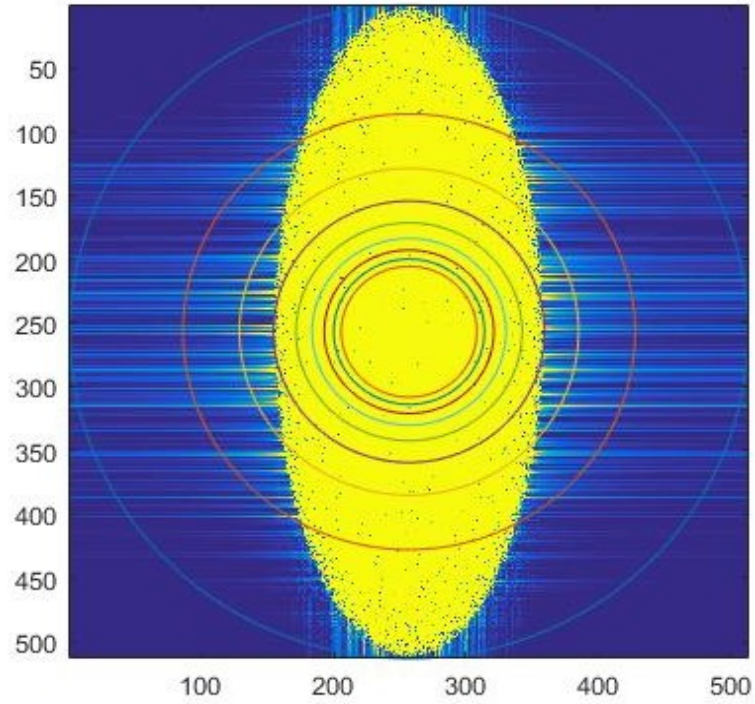
(b)

Fig 5.12. Power spectral density plots of the corresponding speckle patterns shown in Fig

5.11

Coarsing	Figs 5.11(a) and 5.12(a)		Figs 5.11(b) and 5.12(b)	
Factor	Row	Column	Row	Column
1	1.78	7.16	4.47	1.77
2	1.02	3.56	2.22	1.02
3	1.00	2.36	1.48	1.01
4	0.99	1.77	1.16	1.00

116

5	1.00	1.42	1.02	1.00
6	1.00	1.21	0.99	1.00
7	0.99	1.08	1.00	1.00
8	0.99	1.02	1.01	1.00
9	0.99	0.99	1.01	1.00
10	0.99	0.99	1.00	0.99

Table 5.2. Ellipticity values of the speckle patterns represented by Fig 5.11, with the corresponding PSDs in Fig 5.12. As the speckles have different dimensions along the rows and columns, their ellipticities in the two directions are different.

As can be seen, the PSD functions of each of the speckle patterns reflect the varied sizes in different directions. As a result, the plots are elliptical instead of circular, which was the case in all previous studies. Equivalent results are obtained when the ellipticities are measured. So far, the ellipticity values were statistically indistinguishable in the two dimensions, as the speckle patterns had the same correlation length in both. However, elongated speckles necessitate that the measurements be done in each direction separately. The measured values match the expected (from Eq. (5.5) and previous measurements on symmetrical speckles) values for the speckle size in each direction of each of two speckle patterns, i.e., 2,8,5 and 2 pixels for the 2nd, 3rd, 4th and 5th columns of the table, respectively.

117

5.1.5 Analysis

It is worth recalling that the definition of speckle size has a statistical foundation. It is approximated as the square root of the coherence area of the intensity pattern, which is a function of the covariance of the intensity distribution [9]. As the geometry of this coherence is fundamentally two-dimensional, the process of considering its square root only gives an estimate of the one-dimensional *width* of the speckle. For small speckles, the variation of pixel intensity in both dimensions is prominent. As such, defining a boundary between which pixels are to be considered "within the speckle" and which are to be considered "outside" is a relatively straightforward task. However, as the size of the speckles increases, the variation in intensity across the pixels becomes more gradual (*i.e.,* the change extends over more pixels). The stark contrast between neighboring pixels diminishes and thus, the boundaries between individual speckles becomes spatially less distinct.

In conclusion, the use of modified Poincaré plot descriptors has been demonstrated as an approach for describing the spatial structure in the measured intensity of a speckle pattern. The Poincaré analysis as presented above, provides a useful approach for estimating the minimum speckle size in a speckle pattern. A comparison was made with the results using an alternative approach, that is, the use of the power spectral density function for the calculation of minimum speckle size. The concept of employing a cutoff ellipticity of $e = 1.0$ to identify the minimum sampled speckle size was also introduced. Additionally,

it was shown that for large speckles, the present approach has some advantages over the commonly employed approach of assessing the width of the PSD of the speckle pattern.

Additionally, this idea has been expanded to study speckle patterns with varying temporal distances. Also, the earlier process of using data from individual row/column of pixels has been expanded to include the data from the entire speckle field. This eliminates the risk of selecting a row or column of undeveloped speckle pattern, when the methods are applied to imaging speckles. The effects of time-averaging on the measurements were also explored. A key observation here was that despite the lowering of contrast upon frame-by-frame addition of the patterns, the Poincaré descriptors were able to indicate the minimum speckle size in individual patterns. As contrast values are an indication of speed, this above result, along with those obtained from temporal coarsing, might provide a basis to quantify motion and concurrently study the spatial structure of the scatterers involved. This could potentially be an improvement over the current methods of speckle contrast imaging.

Finally, in addition to providing a method for estimating the minimum speckle size in a speckle pattern, spatial Poincaré plots also provide information about the relative contributions of short-term and long-term variations in the spatial structure of the measured intensity distribution of scattered coherent fields.

119

5.2 Phantom and Animal studies

5.2.1 Mean Squared Successive Differences (MSSD)

In addition to the definitions of standard Poincaré parameters as mentioned in section 5.1.2 (and not the modified definitions of section 5.1.3), we use the Mean Squared Successive Differences (MSSD) of a series of variables, introduced in chapter 2, and defined as

$$MSSD = \frac{1}{N-1}\sum_{i=1}^{N-1}(X_i - X_{i+1})^2 \tag{5.6}$$

where N is the total number of elements in the series X. In this section, we first look at the applications of Poincaré descriptors and MSSD to study fluid motion in a phantom sample. This is followed by imaging of an anesthetized mouse using these descriptors.

5.2.2 Fluid Phantoms

5.2.2.1 Preparation

A phantom flow system was constructed using a glass capillary (1 mm outside diameter and 100 μm internal diameter) connected to rubber tubing and a variable height reservoir used to contain the phantom fluid and to alter the velocity of the fluid inside the capillary by changing its height. The phantom fluid was a suspension of polystyrene microspheres with a scattering coefficient of 54 cm^{-1} at 650 nm, the wavelength of the diode laser used for the imaging. The image magnification was approximately 0.1. The glass capillary was illuminated at a slight off-normal angle and the backscattered speckle patterns were

observed normally at 150 frames per second. A total of 256 speckle images were obtained for each different speed of fluid flow.. Each successive speed increment (decrement) was performed by raising (lowering) the pressure head of the fluid source by 0.5 mm. The obtained images were of size 1024 X 1024 pixels. Once the images were obtained, the Poincaré descriptors and MSSD were calculated along the temporal dimension for each pixel. The temporal ellipticity was also obtained from these calculations, for each pixel.

5.2.2.2 Results

A sample speckle image obtained from the phantoms is shown in Fig 5.13.
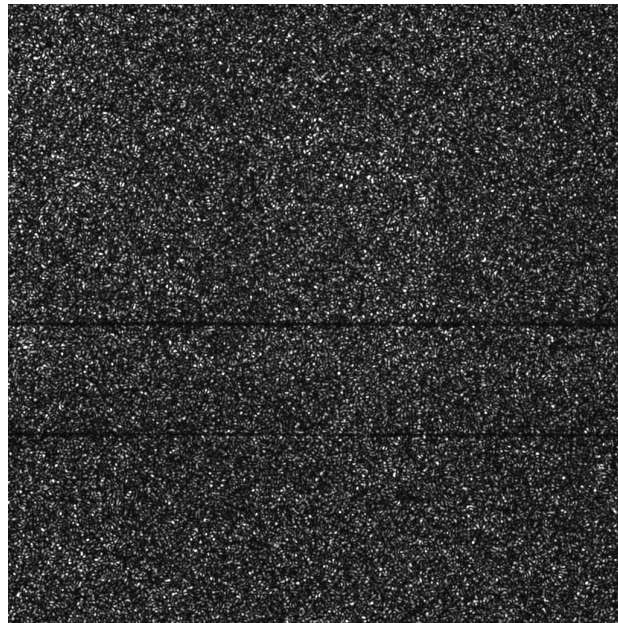
Fig 5.13. Speckle image from the fluid phantoms. The outline of the tube carrying the fluid is clearly visible.

The calculated values of ellipticity and MSSD for each pixel along the temporal axis produced a spatial distribution as shown in Fig 5.14.

121
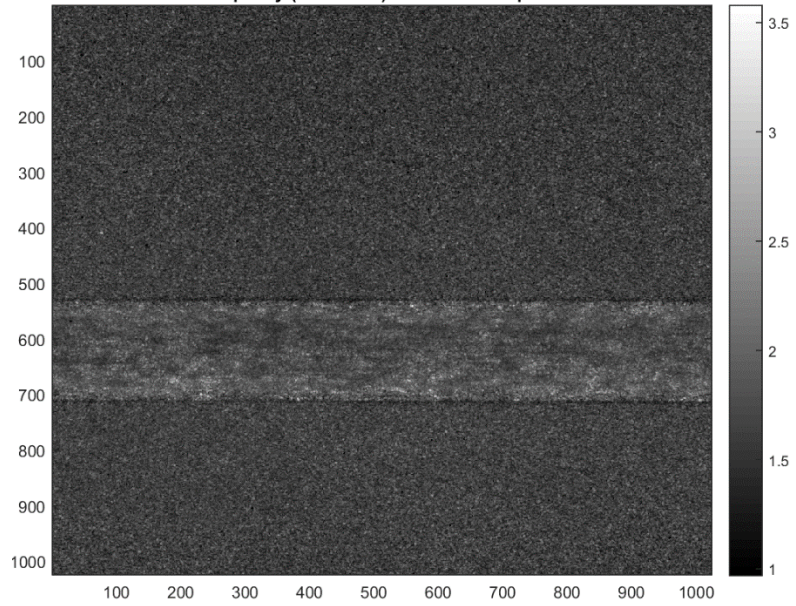
(a)
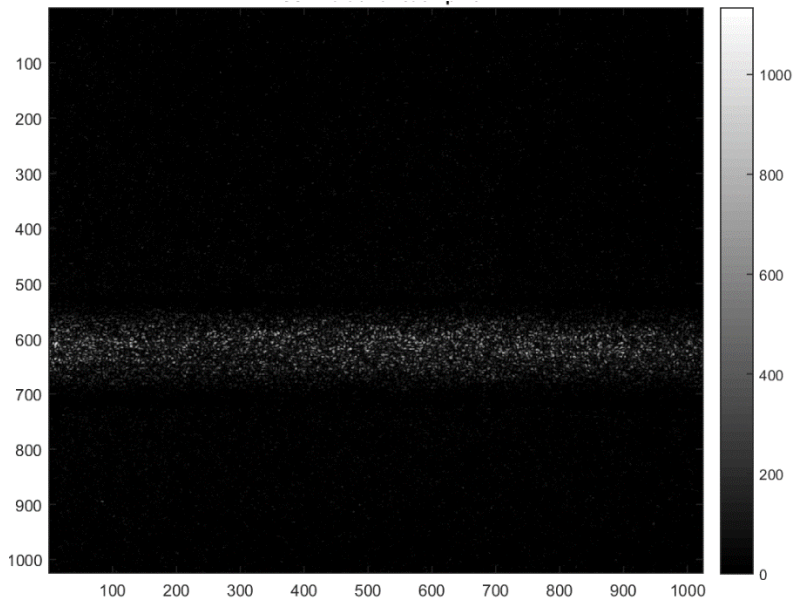
(b)

Fig 5.14 (a) Ellipticity and (b) MSSD values obtained from the speckle intensities

of 256 temporally varying frames of a flowing fluid phantom

122

The mean ellipticity and MSSD along a horizontal row of pixels within the tube carrying the phantom were then calculated for each speed. The obtained values are shown in Fig 5.15 and Fig 5.16.
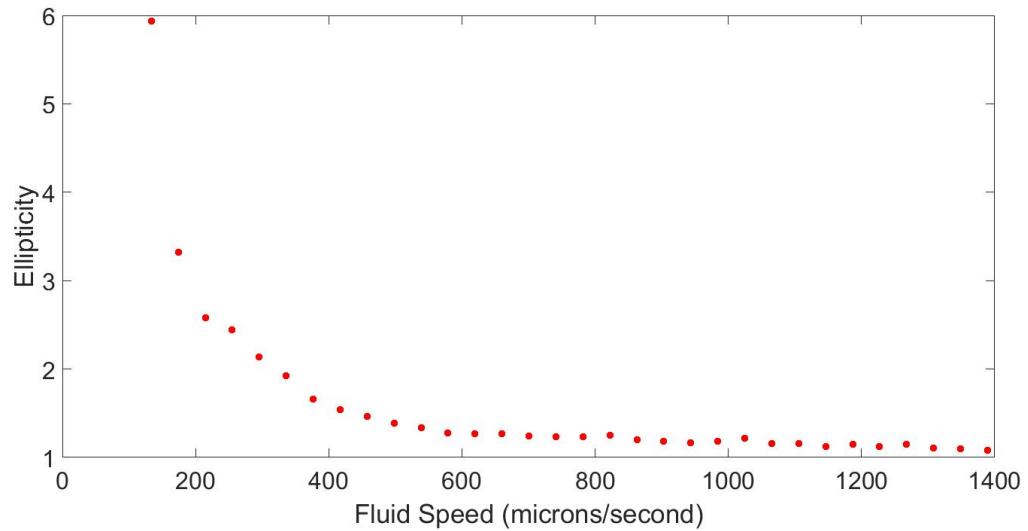
Fig 5.15. Mean ellipticity along a row of pixels inside the phantom samples
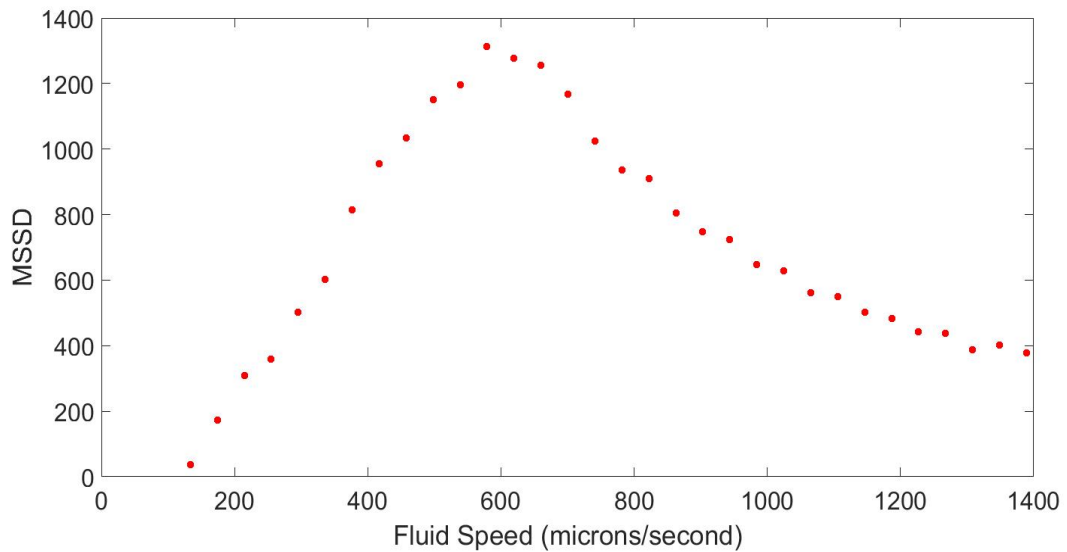
Fig 5.16. Mean MSSD along a row of pixels inside the phantom samples

123

It was observed that the ellipticity variation with fluid speed best fit an inverse power law,

$$e = 1.0471 + \frac{29176}{v^{1.8031}} \tag{5.7}$$

as shown in Fig 5.17. Here, e represents the ellipticity in speckle intensity and v represents the fluid speed.
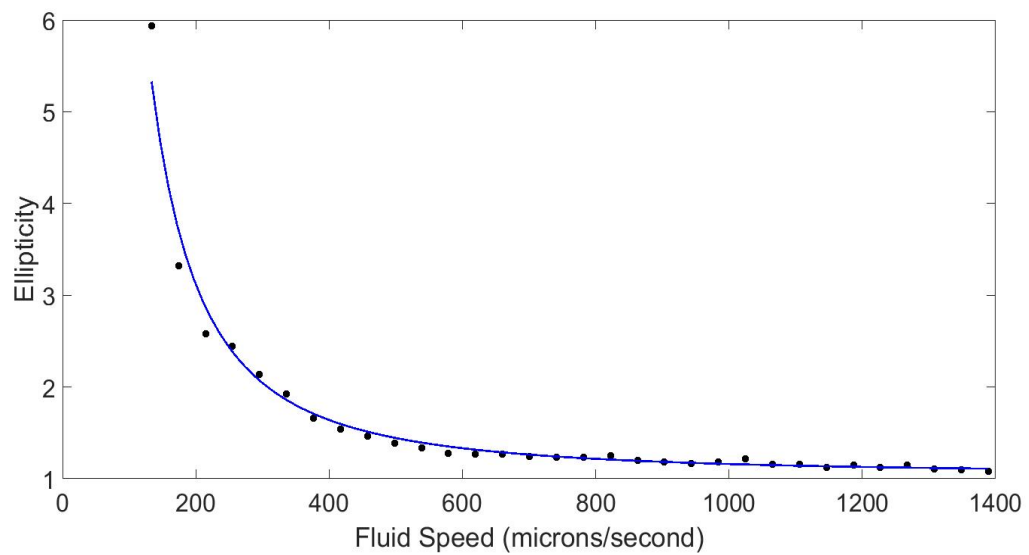
Fig 5.17. Inverse power law of Eq. (5.7) fit onto the ellipticity values of Fig 5.15.

Furthermore, the MSSD values were observed to feature two distinct trends- at lower fluid speeds (up to about 600 microns/second), they increased linearly in close accordance with Eq. (5.8)

$$MSSD = -344.94 + 2.943v \tag{5.8}$$

while at higher speeds the decreased by the power law

124

$$MSSD = 1406 - 20.4549(v - 600)^{0.595} \qquad (5.9)$$

These expressions are fit into the MSSD data as shown in Fig 5.18.
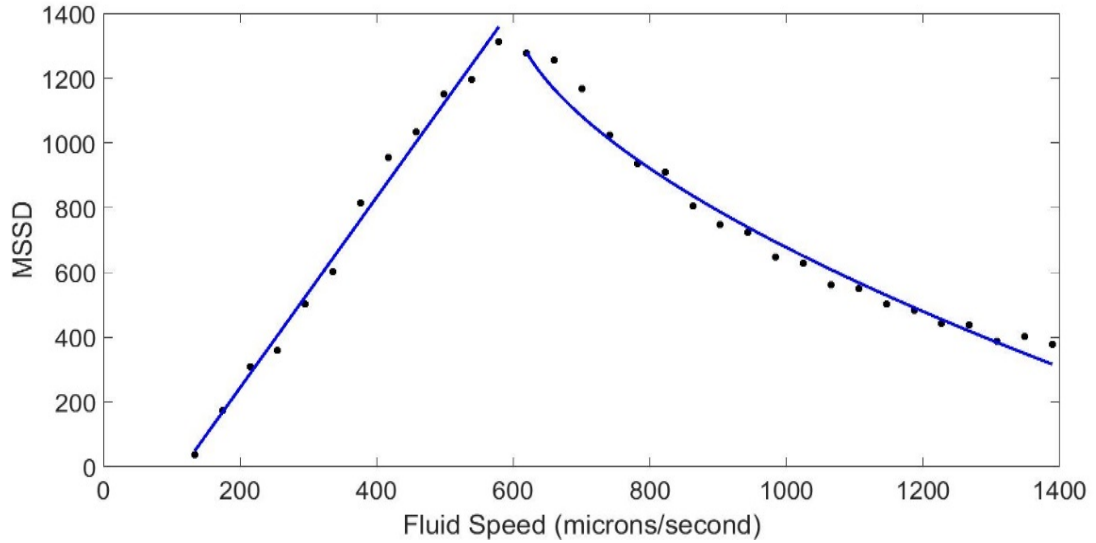
Fig 5.18. At low fluid speeds of up to 600 microns/second, the MSSD values of speckle intensities closely represent Eq. (5.8), while they represent Eq. (5.9) at higher speeds.

We have previously (section 5.1.4) reported the inverse power law variation in ellipticity in speckle field intensity. The falling of ellipticity to values close to 1.0 represents the regime where the long-range (represented by SD2) and short-range (represented by SD1) correlations reach values close each other. We have previously used the term "uncorrelated regime" for this region. The possible factor in this case is that at high speeds, two consecutive speckle frames captured by our imaging system are sufficiently different from each other that their correlation is not distinguishable from randomly picked speckle frames in the series. However, at lower speeds, multiple consecutive frames have a higher correlation among themselves compared to random frames, as demonstrated by a higher

125

ellipticity (which, in turn indicates a significant long-range correlation values given by SD2). This limitation of the imaging system is perhaps better demonstrated by the distinct change in the trend of the MSSD values. MSSD values represent the extent of variation in consecutive speckle frames. At the speeds up to about 600 microns/second, we clearly notice that the observed values indicate an increasing variation in consecutive speckle frame intensities, with increasing speeds. This is an expected observation as the increase in speed results in higher motion of the fluid between frames. However, at speeds above 600 microns/second, the speckles obtained by the imaging system represent random motion more closely as the speeds keep increasing.

This analysis, however, does indicate that when the ellipticity measurements show that the dynamics of a system are in the uncorrelated regime, MSSD measurements might be a better method to distinguish speeds. We now look at applying these measurement techniques to measure flow in animal tissues.

5.2.3 Animal Tissue

5.2.3.1 Preparation

Flow in cranial vessels were imaged using the EI approach. Mice were anesthetized and a cranial window opened in the skull. All animal procedures were reviewed and approved by the Institutional Review Board. Matching sets of image sequences were made on the mice using the same experimental set-up as described above. Coherent image sequences 255 frames long were taken on the mice both prior to and subsequent to the administration of acetylcholine (0.3 µg/kg) via intravascular (tail) infusion. Acetylcholine is known to

have a strong and relatively long-lasting effect of altering blood flow and perfusion. The dose of acetylcholine used lowered the systemic blood pressure and also lowered the heart rate, thus reducing blood flow systemically.

5.2.3.2 Results

Similar to the section 5.2.2.2, the spatial distribution of ellipticity and MSSD values were obtained for the entire image field (Fig 5.19).
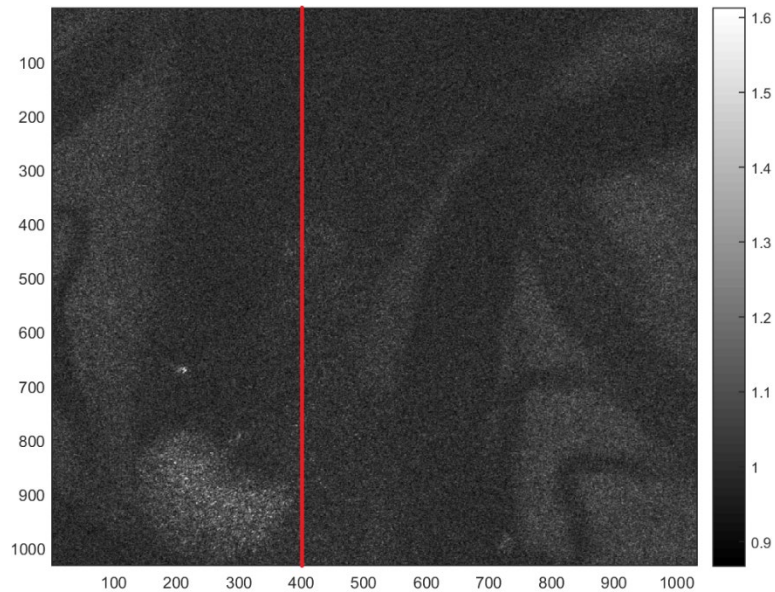
(a)
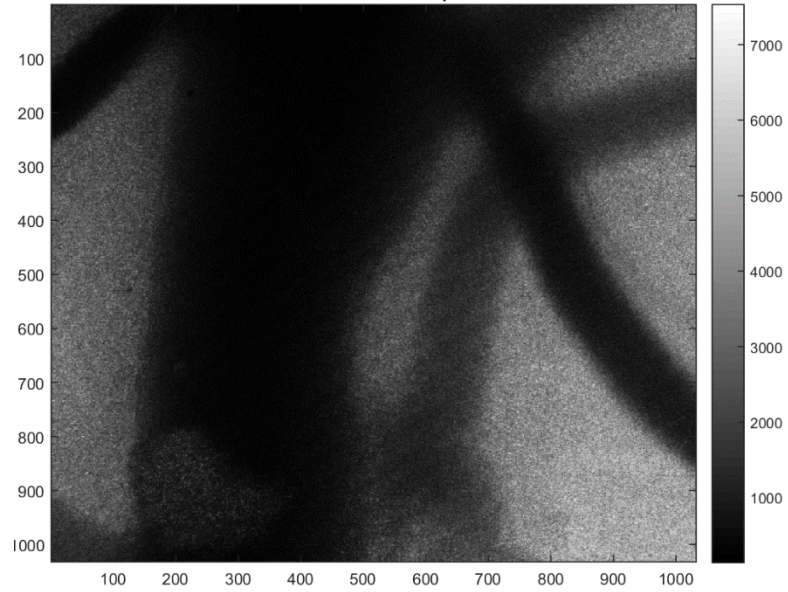
(b)

Fig 5.19. (a) Ellipticity and (b) MSSD values obtained from the speckle intensities of 256 temporally varying frames of an anesthetized mouse tissue. Flow calculations along the red line (column #400) are demonstrated in the text.

Ito calculate the flow speed inside the large blood vessel in Fig 5.19, we picked the speckle distribution along column # 400. Here, we hypothesize that as the imaging settings were kept the same as that for the phantom samples, we can use the mathematical models obtained there for estimating the fluid speeds in the animal samples. The ellipticity value calculated along this column before drug application was 1.02 and after application was 1.04, indicating that the flow is in the uncorrelated regime for our setup. Thus, we use the MSSD model from Eq. (6). MSSD values changed from 120.78 to 365.19 on drug application. This indicates a reduction in speed from 1652.3 μm/s to 1338.2 μm/s.

128

Similar calculations were done on column # 450 of the region in Fig 5.20.
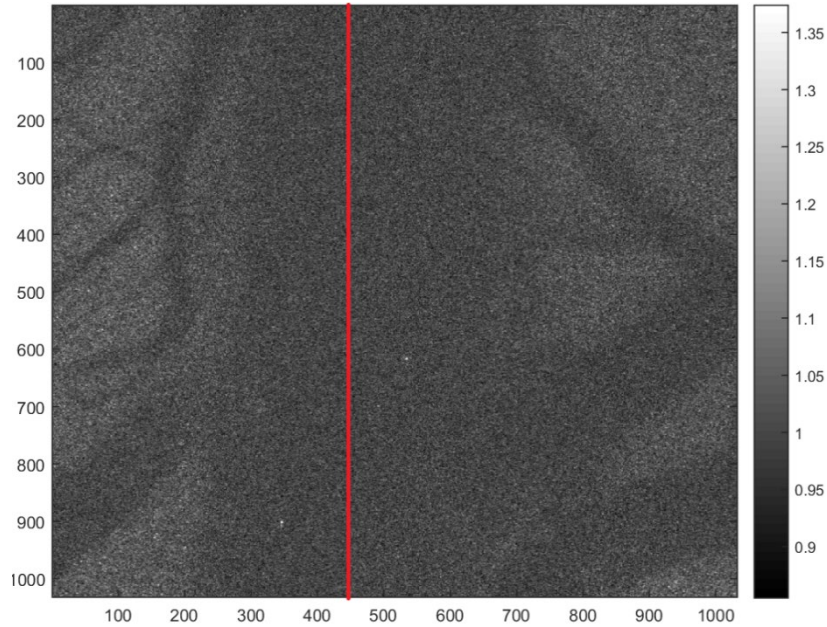
Fig 5.20. Ellipticity values obtained from an animal sample. Flow calculations along the red line (column # 450) are demonstrated in the text.

An increase in MSSD from 177.60 to 274.39 along this column, indicates a reduction in flow speed from 1584.7 μm/s to 1449.7 μm/s on drug application.

We also noticed that some additional features of the region of interest become visible through imaging the ellipticity, compared to imaging the temporal contrast of the dynamic sample. This is demonstrated in Figs 5.21 and5.22.

(a)

(b)

130
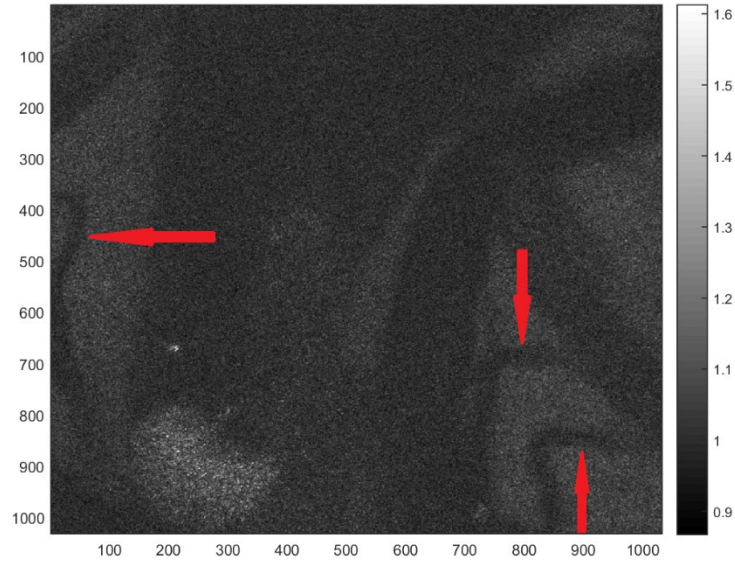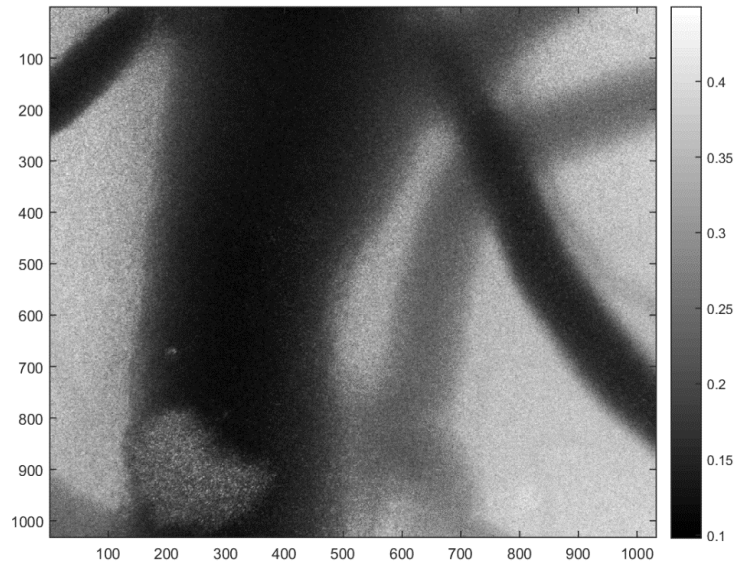
Fig 5.21. (a) Ellipticity image of the region from Fig 5.19 and (b) Temporal contrast image of the same region. The red arrows in (a) indicate features that are not visible in (b).
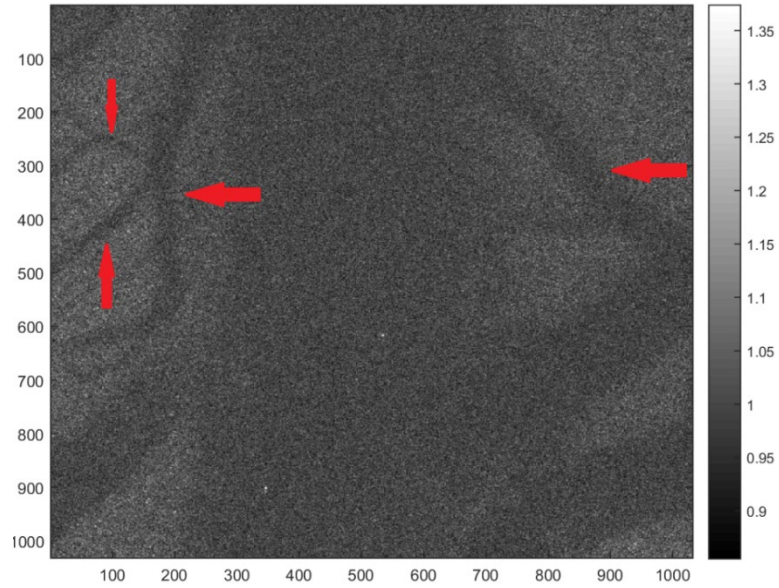
(a)
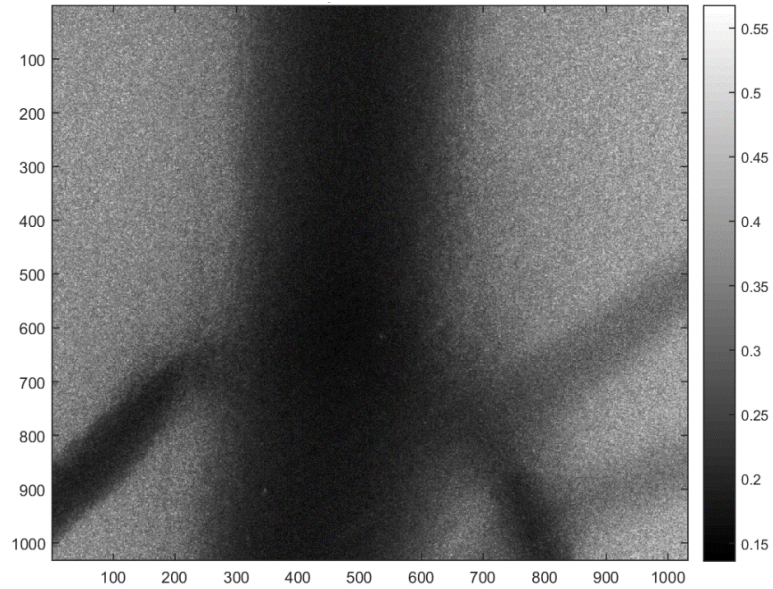
(b)

Fig 5.22. (a) Ellipticity image of the region from Fig 5.20 and (b) Temporal contrast image of the same region. The red arrows in (a) indicate features that are not visible in (b).

We would like to re-emphasize here that the ellipticity calculations treat short- and long-range variations separately. As a result, some of the lesser prominent flow patterns (and in extension, likely blood vessels), which might cause very short-range contributions to the overall variations, might be highlighted.

5.2.4 Analysis

We have presented new approaches to look into dynamic fluid motion in this paper. Fluid phantoms were first analyzed to arrive at mathematical models for measuring speed using two statistical parameters, namely the ellipticity of speckle intensity distributions and the mean squared successive differences (MSSD) in these distributions. These models were

132

then applied to estimate fluid speed in animal tissue samples. A comparison between the observed features from the ellipticity measurement and the more traditionally used contrast measurements was also drawn. It was noted that ellipticity measurements helped in highlighting certain features which aren't visible in contrast measurements.

The limitations of ellipticity measurements were observed at high fluid speeds. To a certain extent, these limitations are compensated by using the MSSD measurements for calculations, instead. It is speculated that capturing the speckle images at a faster frame rate might help in resolving these limitations. Moreover, modifications to our analysis can be done to look at the parameters SD1 and/or SD2 separately, to see if they provide additional information about the dynamics. In conclusion, the methods presented here appear to be extremely and further research in these approaches might open avenues for new imaging technologies.

5.3 Summary

It is demonstrated that the use of spatial Poincaré plots provides an efficient means to describe short and long-range correlations in the spatial structure of the measured intensity distribution of scattered coherent fields. The intensity distribution over a row of pixels in single frames of speckle fields with varying speckle sizes was considered. Statistical descriptors from the spatial Poincaré plots for these intensity data with variable lags were used to estimate the short and long-term variations in the measured intensities, and from these descriptors, the minimum speckle size in the speckle patterns was estimated. This

approach yielded similar results for speckle size estimates as the more standard method of calculating the power spectral density of the intensity pattern and simultaneously provided information on the relative contributions of short-term and long-term variations in the measured intensity to the spatial structure of the scattered fields. The cases of time-averaged speckle patterns and elongated speckles were looked at.

Additionally, alternative approaches to use dynamic laser speckle data to quantify biophysical dynamics including ordered flows and random motions were presented. These yield images that superficially resemble traditional laser speckle contrast images. However, instead of relying on the statistics of the local time integrated intensity values calculated over temporal and sliding spatial windows as is done in LSCI to create images, these approaches directly yield images that quantify the relative dominance of long-range correlations in the temporal dimension of a series of speckle patterns to the short-range correlations in the same dimension. They rely on a Poincaré analysis of the speckle data which yields metrics that statistically describe both the short-terms variations in the temporal speckle intensity (i.e., the standard deviation in successive differences) and also the corresponding long term variations.

5.4 References

[1] J. C. Dainty (Ed.), *Laser Speckle and Related Phenomena*, 2nd edition, Springer Verlag [1984].

134

[2] X. Zhao, and G. Zhao, "Surface roughness measurement using spatial-average analysis of objective speckle pattern in specular direction," Optics and Lasers in Engineering 47 (11), pages 1307-1316 [2009].

[3] R. Jones, and C. Wykes, *Holographic and Speckle Metrology*, 2nd edition, Cambridge University Press [1989].

[4] A. F. Fercher, and J. D. Briers, "Flow visualization by means of single-exposure speckle photography," Optics Communications 37 (5), pages 326-330 [1981].

[5] Y. Aizu, and T. Asakura, "Coherent optical techniques for diagnostics of retinal blood flow," Journal of Biomedical Optics 4 (1), pages 61-75 [1999].

[6] P. Li, S. Ni, L. Zhang, S. Zeng, and Q. Luo, "Imaging cerebral blood flow through the intact rat skull with temporal laser speckle imaging," Optics Letters 31(12), pages 1824-1826 [2006].

[7] J. C. Ramirez-San-Juan, R. Ramos-Garcia, I. Guizar-Iturbide, G. Martinez-Niconoff, and B. Choi, "Impact of velocity distribution assumptions on simplified laser speckle imaging equation," Optics Express 16 (5), pages 3197- 3203 [2008].

[8] K. Khaksari, and S. J. Kirkpatrick, "Laser speckle contrast imaging is sensitive to advective flux," Journal of Biomedical Optics 21 (7), 076001 [2016].

[9] J. W. Goodman (Ed.), *Speckle Phenomena in Optics: Theory and Applications*, Roberts & Company, Englewood, CO [2007].

[10] S. J. Kirkpatrick, D. D. Duncan, and E. M. Wells-Gray, "Detrimental effects of speckle-pixel matching in laser speckle contrast imaging," Optics Letters 33 (24), pages 2886-2888 [2008].

135

[11] K. Khaksari, and S. J. Kirkpatrick, "Combined effects of scattering and absorption on laser speckle contrast imaging," Journal of Biomedical Optics 21 (7), 076002 [2016].

[12] M. Brennan, M. Palaniswami, and P. Kamen, "Distortion properties of the interval spectrum of IPFM generated heartbeats for heart rate variability analysis," IEEE Transactions on Biomedical Engineering 48 (11), pages 1251-1264 [2001].

[13] D. D. Duncan, and S. J. Kirkpatrick, "The copula a tool for simulating speckle dynamics," Journal of the Optical Society of America A 25 (1), pages 231-237 [2008].

[14] J. W. Goodman (Ed.), *Statistical Optics*, 2nd edition, Roberts & Company, Englewood, CO, USA [1985].

Conclusion

In chapter 1, the presence of optical vortices in Gaussian beams is mathematically explained. It is discussed that these vortices do not need special circumstances to be present, and occur by default. Some historical developments in the detection of these vortices were also reviewed. In chapter 2, the mathematical basis for Poincaré analysis was laid down. It was noted that most of this background is applicable to the analysis of any data series that can be represented in a one-dimensional array. Some discussion was provided for the special case when the data series under consideration is the intensity variation in a laser speckle field.

Chapter 3 dealt with simulated speckle fields. These simulations considered three different modes of decorrelation: Gaussian mode generally associated with ordered dynamics; Lorentzian (or exponential) mode generally associated with Brownian dynamics; and a constant sequential autocorrelation for academic curiosity. For all the three modes, three different rates (quantified by the autocorrelation coefficient) of decorrelation were explored. Quantifiable properties, namely the average trail lengths, vortex displacement (or mobility) and the charge separation were introduced and measured. It was noted that the trail lengths and mobility were able to distinguish the different modes of decorrelation, and in extension, scatterer dynamics. It was further noted that a Poincaré analysis of the charge separation gives an estimate of the rate of dynamics. In chapter 4, similar measurements were performed for physical systems with varying dynamics. Purely ordered and purely Brownian motion, as well as superpositions with varying proportions of each of the extremes, were exhibited by microsphere solutions in de-ionized water. The

microspheres act as scattering elements, while water acts as the primary medium, thus creating an environment similar to biological tissues, as far as optics is concerned. Furthermore, vortices from a static scattering system were examined using the rough surface of a metal slab. The obtained results complemented those from simulated speckle fields.

In chapter 5, further applications of Poincaré analysis in speckle fields were provided. It was demonstrated that, as a measure of correlation, Poincaré descriptors can be used to estimate speckle sizes. This holds even in case of heavy degradation of speckle contrast. Methods to use these descriptors to quantify flow in fluid phantoms as well as animal tissues were also discussed. Speckle images obtained form an anesthetized mouse brain were used for this. Comparisons with existing methods were made and the advantages and limitations were discussed.

The first two chapters provide the theoretical background needed to appreciate the remainder of the work presented in this dissertation. In further chapters, optical vortices were detected using a pseudo-phase algorithm in simulated speckle fields (chapter 3) as well as those speckle fields obtained from fluid phantoms (chapter 4). The dynamics of these vortices in evolving speckle fields were then explored. Applications of aspects of Poincaré analysis introduced in chapter 2 were imbibed in the above exploration of vortex dynamics, while further expansion of this analysis method in studying stationary and evolving speckle fields is demonstrated in chapter 5.

Limitations of our research and some avenues for future work have been pointed out towards the end of some of the chapters. Simulations and modeling of additional biological

features (such as tumors) can be done for similar analysis. Additionally, looking at the Poincaré descriptors individually to see if they contain useful information encoded in them, under any specific circumstances, might be another direction to pursue. As such, the work presented here is supposed to serve as a starting point for many possible directions for future research and applications.

As pointed out in chapter 1, the existence of optical vortices (or "phase singularities" or "wavefront dislocations") has been known for a few decades. However, to the best of our knowledge, no work has been done to link these vortices to biophysical dynamics. The research presented in this dissertation demonstrates time evolving phenomena in biological systems, such as tissues, result in optical vortex fields that also display temporal evolution. Furthermore, and perhaps more critically, the characteristics of the evolution of these vortex fields, correlate with the dynamics of the associated biological system. Novel quantitative measures have been introduced and discussed to draw these correlations. Outside of biophysical dynamics, other directions in which the results from this research can be potentially applied are improvements in optical imaging techniques and an improved understanding of light scattering and propagation. This would, in turn, expand our current knowledge about the physics of coherent wave fields.

A Copyright documents

A.1 Permission obtained from SPIE

February 20, 2019

SPIE
PO Box 10
Bellingham WA 98227-0010 USA

Respected SPIE Authority,

I am completing a doctoral dissertation at Michigan Technological University. I would like your permission to reprint in my dissertation excerpts from any publications in the proceedings of SPIE authored or co-authored by me. More specifically, this includes the following papers at the moment:

- Anindya Majumdar and Sean J. Kirkpatrick "Poincaré descriptors of long and short-range correlations in the spatial and temporal fluctuations of coherently scattered light", Proc. SPIE 10493, Dynamics and Fluctuations in Biomedical Photonics XV, 1049305 (13 February 2018); doi: 10.1117/12.2287590;

- Anindya Majumdar and Sean J. Kirkpatrick "Optical vortices as potential indicators of biophysical dynamics", Proc. SPIE 10063, Dynamics and Fluctuations in Biomedical Photonics XIV, 1006306 (3 March 2017); doi: 10.1117/12.2251026;

Additionally, I would like permission to include any excerpts from publications authored or co-authored by me in the proceedings of SPIE to be published in 2019.

I understand that all permissions are effective subject to proper citation of these materials in my dissertation.

The requested permission extends to any future revisions and editions of my dissertation, including non-exclusive world rights in all languages, and to the prospective publication of my dissertation by ProQuest through its ProQuest® Dissertation Publishing business. ProQuest may produce and sell copies of my dissertation on demand and may make my dissertation available for free internet download at my request. These rights will in no way restrict republication of the material in any other form by you or by others authorized by you. Your signing of this letter will also confirm that SPIE owns the copyright to the above-described material.

If these arrangements meet with your approval, please sign this letter where indicated below and return it to me in the enclosed return envelope. Thank you very much.

Sincerely,
Anindya Majumdar
Dept. of Biomedical Engineering
Michigan Technological University
Houghton, MI 49931 USA

PERMISSION GRANTED FOR THE USE REQUESTED ABOVE:

SPIE   Publisher's permission is hereby granted under the following conditions: (1) you obtain permission of the author(s); (2) the material to be used has appeared in our publication without credit or acknowledgment to another source; and (3) you credit the original SPIE publication. Include the authors' names, title of paper, volume title, SPIE volume number, and year of publication in your credit statement.

Director of Publications
SPIE
ox 10, Bellingham, WA 98227-0010 USA
/676-3290 (Pacific Time) eric@spie.org

Eric Pepper, Director of Publications   Date 21 February 2019

A. 2 Permission obtained from J-BPE

February 20, 2019

Journal of Biomedical Photonics & Engineering
34 Moskovskoe Shosse,
Samara, 443086, Russian Federation

Respected J-BPE Authority,

I am completing a doctoral dissertation at Michigan Technological University. I would like your permission to reprint in my dissertation excerpts from any publications of J-BPE authored or co-authored by me. More specifically, this includes the following papers at the moment:

• A. Majumdar and S.J. Kirkpatrick, "Spatial Poincaré plots as descriptors of Speckle Pattern Second-order Statistics", J. of Biomedical Photonics and Eng. 3(3), 030501 (2017)

• A. Majumdar and S.J. Kirkpatrick, "Statistical studies on optical vortices in dynamic speckle fields", J. of Biomedical Photonics and Eng. 4(2), 020301 (2018)

Additionally, I would like permission to include any excerpts from publications authored or co-authored by me in J-BPE to be published in 2019.

I understand that all permissions are effective subject to proper citation of these materials in my dissertation.

The requested permission extends to any future revisions and editions of my dissertation, including non-exclusive world rights in all languages, and to the prospective publication of my dissertation by ProQuest through its ProQuest® Dissertation Publishing business. ProQuest may produce and sell copies of my dissertation on demand and may make my dissertation available for free internet download at my request. These rights will in no way restrict republication of the material in any other form by you or by others authorized by you. Your signing of this letter will also confirm that J-BPE owns the copyright to the above-described material.

If these arrangements meet with your approval, please sign this letter where indicated below and return it to me. Thank you very much.

Sincerely,
Anindya Majumdar
Dept. of Biomedical Engineering
Michigan Technological University
Houghton, MI 49931 USA

PERMISSION GRANTED FOR THE USE REQUESTED ABOVE:

Editor-In-Chief,
PhD., Dr.Sc., SPIE & OSA Fellow, FiDiPro
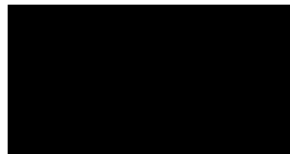
Prof. Valery V. Tuchin

Date: February 25, 2019

B MATLAB Codes

B.1 Generating simulated speckle frames, locating optical vortices in them and calculating their statistical properties

(Primarily used for calculations in Chapter 3)

```matlab
close all


%SECTION 1: Create the Speckle fields and Locate the vortices
stream = RandStream.create('mrg32k3a','seed',sum(100*clock));%mt19937ar or

mrg32k3a

M = 512;

rand(stream,M);

fact = 10;

N = fact*M;% Nyquist is 2*M...

num =50;

mult = 4 ;% increase this parameter to shorten correlation time

resolution=4;%change this to determine how close is close enough to call a continuous

trail; number 'n' means two consecutive vortices have to be within 'n-1'

%resolution has to be at least 2, for anything meaningful

%resolution=4 seems optimal

coarsing_factor=1;%Default=1, Increase this if it is required to skip certain frames (Ex:

coarsing factor=2 would mean all computations done on alternating frames)
```

```matlab
% The following is the specified correlation between the phase realizations

%r_spec = cos(pi*[0:num-1]/(num-1));corr= 'Constant';% Gives constant sequential

speckle

correlation

r_spec = 2*(1-([0:num-1]/(num-1)).^2).^2 - 1;corr= 'Gaussian'; % Yields Gaussian

%r_spec = 2*(1-([0:num-1]/(num-1))).^2 - 1;corr= 'Lorentzian'; % Yields exponential

% create circular frequency plane mask

mask = linspace(-1,1,M);

mask = ones(M,1)*mask;

mask = sqrt(mask.^2 + mask'.^2);

mask = (mask <= mask(M/2,1));

bf = fix((N-M)/2) + 1;

ef = bf + M - 1;

nonz = find(mask);% list of non-zero entries for speckle generation

cube = zeros(M,M,num);

speckle_corr = zeros(1,num);

%upto line 42 to simulate a hole in phase mask

rad=0; %hole radius, max = M/2

mid=M/2;

hole=zeros(M,M);

for a=1:M

for b=1:M

if(sqrt(((a-mid)^2) + ((b-mid)^2))<=rad)
```

143

```matlab
hole(a,b)=1;

end

end

end

mask=mask-hole;

phase_var = zeros(1,num);

phase_corr = zeros(1,num);

amp = sqrt(-2*log(rand(1,M^2)));% amplitude

phi0 = 2*pi*rand(1,M^2);% initial phase

phi = atan2(sqrt(1-r_spec),sqrt(1+r_spec));% phase increments

% First pass through the following loop must be for the case where the

% phase realizations are perfectly correlated, i.e., r == 1.

max_neg=0; %to keep track of maximum positive and negative vortices in an exposure in

the

entire simulation

max_pos=0;

for nnn = 1:num;

% nnn

% realizations

z = amp.*cos(phi0 + phi(nnn));% Gaussian samples

t = normcdf(z);% uniform samples

if nnn == 1;t_ref = t;end% reference phase realization

% compute statistics only over the elements that contribute to the speckle
```

144

```matlab
% pattern
phase_var(nnn) = var(t_ref(nonz) - t(nonz));
t = reshape(t,[M M]);
% keep_t(:,:,nnn) = t;
x = zeros(N,N);
x(bf:ef,bf:ef) = exp(mult*complex(0,1)*2*pi*t).*mask;
x = fft2(fftshift(x));
x = x(1:N/fact,1:N/fact);
rx = real(x);
ix = imag(x);
x = x.*conj(x);
x = x/max(x(:));
if nnn == 1;x_ref = x;end
coef = corrcoef(x_ref,x);% speckle pattern correlation coefficient
speckle_corr(nnn) = coef(1,2);
cube(:,:,nnn) = x;
figure(1);imagesc(sqrt(x));axis image;colormap gray
title('speckle pattern')
figure(2);contour(rx,[0 0],'r-');axis ij;axis square;
title('zero-contours of real and imaginary field components')
hold on
contour(ix,[0 0],'k-');
hold off
```

145

```matlab
phase = atan2(ix,rx);
d1=[0 1;0 -1];
d2=[1 -1;0 0];
d3=[-1 0;1 0];
d4=[0 0;-1 1];
f1=conv2(phase,d1,'same');f1=round(f1/(2*pi));
f2=conv2(phase,d2,'same');f2=round(f2/(2*pi));
f3=conv2(phase,d3,'same');f3=round(f3/(2*pi));
f4=conv2(phase,d4,'same');f4=round(f4/(2*pi));
dmap=f1+f2+f3+f4;
[neg_rr,neg_cc] = find(dmap == -1);%Topological charge of -1
figure(3);imagesc(phase);colormap gray;axis image
title('phase and point-singularities');hold on
n_v_of_minus_one=length(neg_rr);
plot(cc,rr,'bo');hold on
[pos_rr,pos_cc] = find(dmap == 1);%Topological charge of 1
plot(cc,rr,'ro');hold on
n_v_of_one=length(pos_rr);
[rr,cc] = find(dmap ~= 0);
n_v=length(rr);
figure(4);imagesc(phase);colormap gray;colorbar;axis image
title('phase and point-singularities');hold on
plot(cc,rr,'y.','markersize',24);hold off;
```

146

```
B_pos(1:length(pos_rr),1,nnn)=pos_rr;A_pos=B_pos(:,1,nnn);

B_pos(1:length(pos_rr),2,nnn)=pos_cc;C_pos=B_pos(:,2,nnn);

B_pos(1:length(pos_rr),3,nnn)=nnn;D_pos=B_pos(:,3,nnn);

B_neg(1:length(neg_rr),1,nnn)=neg_rr;A_neg=B_neg(:,1,nnn);

B_neg(1:length(neg_rr),2,nnn)=neg_cc;C_neg=B_neg(:,2,nnn);

B_neg(1:length(neg_rr),3,nnn)=nnn;D_neg=B_neg(:,3,nnn);

figure(5);imagesc(sqrt(x));colormap gray;axis image;

title('speckle pattern and point-singularities (Green -1; Red +1)');drawnow;hold on

plot(pos_cc,pos_rr,'ro');hold on;plot(neg_cc,neg_rr,'go');drawnow;hold off

figure(6);plot3(D_pos,C_pos,abs(M-A_pos),'r.','markersize',20);axis ij;axis tight;grid

on;box on;hold on

plot3(D_neg,C_neg,abs(M-A_neg),'g.','markersize',20);

xlabel('Exposure number');ylabel('x-coordinate');zlabel('y-coordinate');

title('Vortex paths (Green -1; Red +1)');

if (n_v_of_one > max_pos)

max_pos=n_v_of_one;

end

if (n_v_of_minus_one > max_neg)

max_neg=n_v_of_minus_one;

end

end

%SECTION 2: Constructing the matrices of vortex paths

pos_vortices=0;
```

147

```matlab
neg_vortices=0;

total_pos_trails=0;

total_neg_trails=0;

pos_location= NaN(max_pos,3,num);

neg_location= NaN(max_neg,3,num);

for k=1:num

for p= 1:max_pos

if(B_pos(p,3,k)>0)

pos_vortices= pos_vortices+1;

pos_location(p,1,k)=B_pos(p,2,k);

pos_location(p,2,k)=abs(M-B_pos(p,1,k));

pos_location(p,3,k)=k;

else

pos_location(p,3,k)=0;

end

end

for m=1:max_neg

if(B_neg(m,3,k)>0)

neg_vortices = neg_vortices+1;

neg_location(m,1,k)=B_neg(m,2,k);

neg_location(m,2,k)=abs(M-B_neg(m,1,k));

neg_location(m,3,k)=k;

else
```

148

```matlab
neg_location(m,3,k)=0;

end

end

end

temp_pos= pos_location; %do not want to change location matrices, so introducing new
matrices

temp_neg= neg_location;

%code to form matrices of trails, with each trail in a single row

pos_trails= NaN(max_pos*num,3,num); %maximum possible size; initial allocation runs
faster than dynamic allocation

neg_trails= NaN(max_neg*num,3,num); %worst case scenario: every trail has a single
vortex

%computing positive trails

framecheck=1;

currenttrail=0;

while (framecheck<num+1)

% checking_frame = framecheck

for(s=1:max_pos)

% total_pos_trails= total_pos_trails + 1;

if(temp_pos(s,3,framecheck)>0) %checking existence

% found_new_vortex_at_position = s

currenttrail=currenttrail+1;

total_pos_trails= total_pos_trails + 1;
```

149

```matlab
currentframe=framecheck;

count=1;

pos_trails(currenttrail,1,count)=temp_pos(s,1,currentframe);

pos_trails(currenttrail,2,count)=temp_pos(s,2,currentframe);

pos_trails(currenttrail,3,count)=1;

compare_x=temp_pos(s,1,currentframe);

compare_y=temp_pos(s,2,currentframe);

temp_pos(s,3,currentframe)=0;

count=count+1;

currentframe=currentframe+coarsing_factor;

proceed=1;

while(currentframe < num+1 && proceed ==1)

% comparing_with_frame = currentframe

need=1;

p=1;

while(p<max_pos+1 && need==1)

if(abs(temp_pos(p,1,currentframe)-compare_x)<resolution && abs(temp_pos(p,2,

currentframe)-compare_y)<resolution && temp_pos(p,3,currentframe)>0)

% above line is for checking continuity and existence;

% continuity condition can be user defined

pos_trails(currenttrail,1,count)=temp_pos(p,1,currentframe);

pos_trails(currenttrail,2,count)=temp_pos(p,2,currentframe);

pos_trails(currenttrail,3,count)=1;
```

150

```matlab
compare_x=temp_pos(p,1,currentframe);

compare_y=temp_pos(p,2,currentframe);

temp_pos(p,3,currentframe)=0;

need=0;

count=count+1;

currentframe=currentframe+coarsing_factor;

% if(currentframe>num)

% currentrail = currenttrail +1 ;

% end

end

p=p+1;

end

if (need==1)

proceed=0;

% currenttrail=currenttrail+1;

% total_pos_trails= total_pos_trails + 1;

end

end

end

end

framecheck=framecheck+1;

%currenttrail = currenttrail+1;

end
```

151

```matlab
%computing negative trails

framecheck=1;

currenttrail=0;

while (framecheck<num+1)

%checking_frame = framecheck

for(s=1:max_neg)

if(temp_neg(s,3,framecheck)>0) %checking existence

currenttrail=currenttrail+1;

total_neg_trails= total_neg_trails + 1;

currentframe=framecheck;

count=1;

neg_trails(currenttrail,1,count)=temp_neg(s,1,currentframe);

neg_trails(currenttrail,2,count)=temp_neg(s,2,currentframe);

neg_trails(currenttrail,3,count)=1;

compare_x=temp_neg(s,1,currentframe);

compare_y=temp_neg(s,2,currentframe);

temp_neg(s,3,currentframe)=0;

count=count+1;

currentframe=currentframe+coarsing_factor;

proceed=1;

while(currentframe < num+1 && proceed ==1)

% comparing_with_frame = currentframe

need=1;
```

```matlab
p=1;

while(p<max_neg+1 && need==1)

if(abs(temp_neg(p,1,currentframe)-compare_x)<resolution && abs(temp_neg(p,2,

currentframe)-compare_y)<resolution && temp_neg(p,3,currentframe)>0)

% above line is for checking continuity and existence;

% continuity condition can be user defined

neg_trails(currenttrail,1,count)=temp_neg(p,1,currentframe);

neg_trails(currenttrail,2,count)=temp_neg(p,2,currentframe);

neg_trails(currenttrail,3,count)=1;

compare_x=temp_neg(p,1,currentframe);

compare_y=temp_neg(p,2,currentframe);

temp_neg(p,3,currentframe)=0;

need=0;

count=count+1;

currentframe=currentframe+coarsing_factor;

end

p=p+1;

end

if (need==1)

proceed=0;

% currenttrail=currenttrail+1;

end

end
```

153

```matlab
end

end

framecheck=framecheck+1;

end

corr

mult

average_pos_trail_length= pos_vortices/total_pos_trails

average_neg_trail_length= neg_vortices/total_neg_trails

average_trail_length= (pos_vortices + neg_vortices)/(total_pos_trails + total_neg_trails)

%SECTION 3: Displaying in a trailwise manner, for better visualisation

%each trail formns a column

trailwise_pos= NaN(num,2,max_pos*num);

trailwise_neg= NaN(num,2,max_neg*num);

for(n=1:num)

for(p=1:max_pos*num)

trailwise_pos(n,1,p)=pos_trails(p,1,n);

trailwise_pos(n,2,p)=pos_trails(p,2,n);

end

end

for(n=1:num)

for(p=1:max_neg*num)

trailwise_neg(n,1,p)=neg_trails(p,1,n);

trailwise_neg(n,2,p)=neg_trails(p,2,n);
```

154

```matlab
end

end

%SECTION 4: Calculating the deviation in vortex paths

%NOT used in dissertation but included for any future applications

memory_neg= NaN(max_neg*num,3,num);

memory_pos= NaN(max_pos*num,3,num);

check_x=M+5;%setting up initial values where there can never be vortices

check_y=M+5;

for p=1:max_neg*num

m=1;

for q=1:num

if((neg_trails(p,1,q)~=check_x |neg_trails(p,2,q) ~= check_y)&& neg_trails(p,3,q)>0)

memory_neg(p,1,m)=neg_trails(p,1,q);

memory_neg(p,2,m)=neg_trails(p,2,q);

memory_neg(p,3,m)=1;

check_x= memory_neg(p,1,m);

check_y= memory_neg(p,2,m);

m=m+1;

end

end

end

check_x=M+5;%same for positive vortices

check_y=M+5;
```

```
for p=1:max_pos*num

m=1;

for q=1:num

if((pos_trails(p,1,q)~=check_x |pos_trails(p,2,q) ~= check_y)&& pos_trails(p,3,q)>0)

memory_pos(p,1,m)=pos_trails(p,1,q);

memory_pos(p,2,m)=pos_trails(p,2,q);

memory_pos(p,3,m)=1;

check_x= memory_pos(p,1,m);

check_y= memory_pos(p,2,m);

m=m+1;

end

end

end

memory_theta_pos= NaN(max_pos*num,num);

memory_theta_neg= NaN(max_neg*num,num);

for k= 2:num-1 % first and last columns are meaningless, as they cannot represent
deviations

for p=1:max_pos*num % calculations for each positive vortex, one by one

if(memory_pos(p,3,k-1)> 0 && memory_pos(p,3,k)> 0 && memory_pos(p,3,k+1) > 0)%
checking that 3 consecutive positive vortices exist

sam1=memory_pos(p,1,k-1);%x-coordinate of previous vortex

sam2=memory_pos(p,2,k-1);%y-coordinate of previous vortex

sam3=memory_pos(p,1,k);%x-coordinate of current vortex
```

156

```matlab
sam4=memory_pos(p,2,k);%y-coordinate of current vortex

sam5=memory_pos(p,1,k+1);%x-coordinate of next vortex

sam6=memory_pos(p,2,k+1);%y-coordinate of next vortex

memory_theta_pos(p,k)=real(acosd(real(dot(((sam3+i*sam4)-(sam1+i*sam2)),

((sam5+i*sam6)-(sam3+i*sam4))))/(norm((sam3+i*sam4)-

(sam1+i*sam2))*norm((sam5+i*sam6)-

(sam3+i*sam4)))));

end

end

for p=1:max_neg*num %similar as above, with the negative vortices

if(memory_neg(p,3,k-1)> 0 && memory_neg(p,3,k)> 0 && memory_neg(p,3,k+1) >

0)%

checking that 3 consecutive positive vortices exist

sam1=memory_neg(p,1,k-1);%x-coordinate of previous vortex

sam2=memory_neg(p,2,k-1);%y-coordinate of previous vortex

sam3=memory_neg(p,1,k);%x-coordinate of current vortex

sam4=memory_neg(p,2,k);%y-coordinate of current vortex

sam5=memory_neg(p,1,k+1);%x-coordinate of next vortex

sam6=memory_neg(p,2,k+1);%y-coordinate of next vortex

memory_theta_neg(p,k)=real(acosd(real(dot(((sam3+i*sam4)-(sam1+i*sam2)),

((sam5+i*sam6)-(sam3+i*sam4))))/(norm((sam3+i*sam4)-

(sam1+i*sam2))*norm((sam5+i*sam6)-

(sam3+i*sam4)))));
```

157

```matlab
end

end

end

% figure(13);hist(memory_theta_pos,36);xlabel('Deviation in degrees');ylabel
('Frequency'); title('Deviation of positive vortices');

% figure(14);hist(memory_theta_neg,36);xlabel('Deviation in degrees');ylabel
('Frequency'); title('Deviation of negative vortices');

%better visualisation trailwise

memory_trailwise_pos= NaN(num,2,max_pos*num);

memory_trailwise_neg= NaN(num,2,max_neg*num);

for(n=1:num)

for(p=1:max_pos*num)

memory_trailwise_pos(n,1,p)=memory_pos(p,1,n);

memory_trailwise_pos(n,2,p)=memory_pos(p,2,n);

end

end

for(n=1:num)

for(p=1:max_neg*num)

memory_trailwise_neg(n,1,p)=memory_neg(p,1,n);

memory_trailwise_neg(n,2,p)=memory_neg(p,2,n);

end

end

%%
```

```matlab
x=memory_theta_pos(:);

[f,xi] = ksdensity(x,'width',10);

%xi=xi./mean(xi);

%figure(2);plot(xi,f./sum(f));%Normalizes

figure(12);plot(xi,f);

grid on

%figure(13);semilogy(xi/mean(xi),f./.0129);

xlabel('Angle');ylabel('P(I)');axis([0 180 0 .025]);

title(['PDF for ' ,corr]);

hold on

y=memory_theta_neg(:);

[f,yi] = ksdensity(y,'width',10);

%xi=xi./mean(xi);

%figure(2);plot(xi,f./sum(f));%Normalizes

figure(12);plot(yi,f,'r');

legend('positive','negative')

grid on

%figure(13);semilogy(xi/mean(xi),f./.0129);

xlabel('Angle');ylabel('P(I)');

title(['PDF for ' ,corr])

%SECTION 5: Plots and statistical calculations

pos_count=zeros(num,1);

neg_count=zeros(num,1);
```

159

```
for(i=1:num)

count=0;

for(j=1:max_pos)

if(pos_location(j,3,i)>0)

count=count+1;

end

end

pos_count(i)=count;

count=0;

for(j=1:max_neg)

if(neg_location(j,3,i)>0)

count=count+1;

end

end

neg_count(i)=count;

end

figure(14);plot(pos_count,'r.','markersize', 18);hold on;

plot(neg_count,'b.','markersize', 18)

xlabel('Frames');ylabel('No.of vortices');

title(['Vortex count for ' ,corr, '(Red:+1, Blue:-1)'])

pos_traillength=zeros(total_pos_trails,1);

neg_traillength=zeros(total_neg_trails,1);

for(i=1:total_pos_trails)
```

160

```matlab
count=0;

for(j=1:num)

if(isfinite(trailwise_pos(j,2,i)))

count=count+1;

end

end

pos_traillength(i)=count;

end

for(i=1:total_neg_trails)

count=0;

for(j=1:num)

if(isfinite(trailwise_neg(j,2,i)))

count=count+1;

end

end

neg_traillength(i)=count;

end

figure(15);plot(pos_traillength,'r.','markersize', 18);hold on;

plot(neg_traillength,'b.','markersize', 18)

xlabel('Trail #');ylabel('Trail Length');

title(['Trail lengths for ' ,corr, '(Red:+1, Blue:-1)'])

standard_deviation_pos_traillength= std(pos_traillength)

standard_deviation_neg_traillength= std(neg_traillength)
```

161

```matlab
total_pos_trails

total_neg_trails

pos_motions=zeros(total_pos_trails,1);

neg_motions=zeros(total_neg_trails,1);

total_pos_motions=0;

total_neg_motions=0;

for(i=1:total_pos_trails)

count=0;

for(j=1:num)

if(isfinite(memory_trailwise_pos(j,2,i)))

count=count+1;

end

end

pos_motions(i)=count-1;

total_pos_motions = total_pos_motions + pos_motions(i);

end

for(i=1:total_neg_trails)

count=0;

for(j=1:num)

if(isfinite(memory_trailwise_neg(j,2,i)))

count=count+1;

end

end
```

162

```matlab
neg_motions(i)=count-1;

total_neg_motions = total_neg_motions + neg_motions(i);

end

figure(16);plot(pos_motions,'r.','markersize', 18);hold on;

plot(neg_motions,'b.','markersize', 18)

xlabel('Trail #');ylabel('# Motions');

title(['Net Motions for ' ,corr, '(Red:+1, Blue:-1)'])

pos_motion_probability = total_pos_motions / pos_vortices

neg_motion_probability = total_neg_motions / neg_vortices

mean_pos = NaN(num,2);

mean_neg = NaN(num,2);

mean_location= NaN(num,2);

for(i=1:num)

mean_pos(i,1)= mean(pos_location(1:pos_count(i),1,i));

mean_pos(i,2)= mean(pos_location(1:pos_count(i),2,i));

mean_neg(i,1)= mean(neg_location(1:neg_count(i),1,i));

mean_neg(i,2)= mean(neg_location(1:neg_count(i),2,i));

mean_location(i,1)= (mean_pos(i,1)*pos_count(i) +

mean_neg(i,1)*neg_count(i))/(pos_count

(i) + neg_count(i));

mean_location(i,2)= (mean_pos(i,2)*pos_count(i) +

mean_neg(i,2)*neg_count(i))/(pos_count

(i) + neg_count(i));
```

163

```matlab
figure(17);%plot3(i,mean_pos(i,1),mean_pos(i,2),'r.','markersize',20);axis ij;axis tight;

grid on;box on;hold on

%plot3(i,mean_neg(i,1),mean_neg(i,2),'b.','markersize',20); hold on;

plot3(i,mean_location(i,1),mean_location(i,2),'g.','markersize',20); hold on;

xlabel('Frame number');ylabel('x-coordinate');zlabel('y-coordinate');

title(['Path of mean vortex (Blue -1; Red +1; Green Central) for ', corr]);

end

pos_displacement= NaN(num,1);

neg_displacement= NaN(num,1);

central_displacement = NaN(num,1);

gap= NaN(num,1);

pos_displacement(1)=0;

neg_displacement(1)=0;

central_displacement(1)=0;

gap(1)= sqrt(power(mean_pos(1,1)-mean_neg(1,1),2)+power(mean_pos(1,2)-

mean_neg(1,2),2));

for(i=2:num)

gap(i)= sqrt(power(mean_pos(i,1)-mean_neg(i,1),2)+power(mean_pos(i,2)-

mean_neg(i,2),

2));

pos_displacement(i)= sqrt(power(mean_pos(i,1)-mean_pos(i-

1,1),2)+power(mean_pos(i,2)-

mean_pos(i-1,2),2));
```

164

```matlab
neg_displacement(i)= sqrt(power(mean_neg(i,1)-mean_neg(i-
1,1),2)+power(mean_neg(i,2)-
mean_neg(i-1,2),2));
central_displacement(i)= sqrt(power(mean_location(i,1)-mean_location(i-1,1),2)+power
(mean_location(i,2)-mean_location(i-1,2),2));
end
average_positive_displacement = mean(pos_displacement)
average_negative_displacement = mean(neg_displacement)
average_central_displacement = mean(central_displacement)
average_gap = mean(gap)
stdev_positive_displacement = std(pos_displacement)
stdev_negative_displacement = std(neg_displacement)
stdev_central_displacement = std(central_displacement)
stdev_gap = std(gap)
figure(18);plot(pos_displacement,'r.','markersize', 18);hold on;
plot(neg_displacement,'b.','markersize', 18); hold on;
plot(central_displacement,'y.','markersize', 18); hold on;
plot(gap,'g.','markersize', 18);
xlabel('Frame #');ylabel('Distance in coordinate points');
title(['Net Displacements for ' ,corr, '(Red:+1, Blue:-1, Green: Gap, Yellow:Central)'])
% SECTION 6: Code for Poincare plots
array1= gap;param='Gap';
%options:"pos_count","neg_count","pos_count+neg_count","pos_displacement","
```

neg_displacement","central_displacement","gap"

```matlab
len=length(array1);

ele=len+1;

array2=zeros(ele,1);

array2(1)=0;

array2(2:ele)=array1;

center=(min(array1)+ max(array1))/2;

%%Alternative Definitions
%sd1c=sqrt((1/(len-1))*sum(((array1(2:len)-array2(2:len))-mean(array1(2:len)-array2(2:
len)))).^2)/2)

%sd2c= sqrt((1/len)*sum(((array1(2:len)+array2(2:len))-mean(array1(2:len)+array2(2:
len)))).^2)/2)

sdsd=sqrt(mean((array1(2:len)-array2(2:len)).^2)-(mean(array1(2:len)-array2(2:len)))^2);

sdrr=sqrt(mean((array1).^2)-(mean(array1))^2);

sd1=sqrt((sdsd^2)/2)

sd2= sqrt(2*sdrr^2-sd1^2)

figure(19);

plot(array2(2:ele-1),array1(2:ele-1),'r.');

title(['Poincare Plot for ', corr, ' of ', param]);

xlabel('Previous measure');ylabel('Next Measure');hold on;

plot(1:1.5*max(array1),1:1.5*max(array1),'--');hold on;

t=0:0.01:2*pi;

sds1=1.01;
```

166

```matlab
sds2=1.85

xx = sds2*sin(t) ;

yy = sds1*cos(t) ;

xx2 = xx*cos(pi/4) - yy*sin(pi/4) + center;

yy2 = xx*sin(pi/4) + yy*cos(pi/4) + center;

plot(xx2,yy2);axis equal;

% str={'sd1=',sd1,'sd2=',sd2,'mult=',mult,'corr=',corr};

% text(max(array1)/10,1.2*max(array1),str);

%for intensity on each column

mid= round(M/2);

for nnn=1:num

hor_intensity (nnn,:)= cube(mid,:,nnn);

vert_intensity (nnn,:)= cube(:,mid,nnn);

end

corr

fact
```

B.2 Calculating vortex statistics from speckle images obtained from dynamic scattering systems

(Primarily used for calculations in Chapter 4)

```matlab
clear all;

close all;

warning off;
```

167

```matlab
images=300;%No. of images to input

skip_factor=1 %'1' means read every image;ensure that (images*skip_factor)< images in

folder

path= 'F:\Anindya\Alum\dyn3_far5_10fps\';

prefix='open_2018-06-20-150109-';

format='.bmp';      %input

sc= [120 40];%starting coordinates for cropped square

sz= 400; %size of cropped square

rect=[sc(1) sc(2) sz sz];

% cube= zeros(size(640,480,images));

max_pos=0;

max_neg=0;

coarsing_factor=1;


speckle_cor=zeros(images);%Speckle Correlation, compared to Frame#1

rel_cor1=zeros(images);

vortex_cor=zeros(images);%Vortex Correlation, compared to Frame#1

rel_cor2=zeros(images);

resolution=80;

spread=zeros(sz,sz,images);

for num=1:images

 %num

 im= imread(strcat(path,prefix,num2str(skip_factor*(num-1),'%04g'),format));
```

```matlab
im=im(:,:,1);

im=double(im);

raw_cube(:,:,num)=im;
```
```matlab
%figure(1);imagesc(im); colormap gray;hold on;title('Original Image');
% rectangle('Position',rect,'EdgeColor','r');
```

```matlab
im=im/max(im(:));

im=im-mean(im(:));

imfilt=imgaussfilt(im,12);

cube(:,:,num)=imfilt;

im=imcrop(im,rect);
```
```matlab
%figure(2);imagesc(im); colormap gray;hold on;title('Cropped Section');axis square;
```

```matlab
spect=hilbert(imfilt);

spect=hilbert(spect');

spect=spect';

hrx1=real(spect(rect(2):(rect(2)+rect(4)),rect(1):(rect(1)+rect(3))));

hix1=imag(spect(rect(2):(rect(2)+rect(4)),rect(1):(rect(1)+rect(3))));
```

```matlab
imfilt=imcrop(imfilt,rect);
```
```matlab
%figure(3);imagesc(imfilt); colormap gray;hold on;title('Filtered Section');axis square;
```

```matlab
phase = atan2(hix1,hrx1);
```

169

```matlab
d1=[0 1;0 -1];

d2=[1 -1;0 0];

d3=[-1 0;1 0];

d4=[0 0;-1 1];

f1=conv2(phase,d1,'same');f1=round(f1/(2*pi));

f2=conv2(phase,d2,'same');f2=round(f2/(2*pi));

f3=conv2(phase,d3,'same');f3=round(f3/(2*pi));

f4=conv2(phase,d4,'same');f4=round(f4/(2*pi));

dmap=f1+f2+f3+f4;

[neg_rr,neg_cc] = find(dmap == -1);%Topological charge of -1

 %figure(4);imagesc(phase);colormap gray;axis image

% title('phase and point-singularities');hold on

n_v_of_minus_one=length(neg_rr);


% plot(cc,rr,'bo');hold on

[pos_rr,pos_cc] = find(dmap == 1);%topological charge of 1

% plot(cc,rr,'ro');hold on

n_v_of_one=length(pos_rr);

[rr,cc] = find(dmap ~= 0);

n_v=length(rr);

%figure(5);imagesc(phase);colormap gray;colorbar;axis image; hold on;

%title('phase and point-singularities');hold on

%plot(cc,rr,'y.','markersize',24);hold off;
```

170

```matlab
B_pos(1:length(pos_rr),1,num)=pos_rr;A_pos=B_pos(:,1,num);

B_pos(1:length(pos_rr),2,num)=pos_cc;C_pos=B_pos(:,2,num);

B_pos(1:length(pos_rr),3,num)=num;D_pos=B_pos(:,3,num);


B_neg(1:length(neg_rr),1,num)=neg_rr;A_neg=B_neg(:,1,num);

B_neg(1:length(neg_rr),2,num)=neg_cc;C_neg=B_neg(:,2,num);

B_neg(1:length(neg_rr),3,num)=num;D_neg=B_neg(:,3,num);


%figure(6);imagesc(phase);colormap gray;axis image;hold on;

%title('speckle pattern and point-singularities (Green -1; Red +1)');drawnow;hold on

%plot(pos_cc,pos_rr,'rx','markersize',10,'LineWidth',4);hold

on;plot(neg_cc,neg_rr,'go','markersize',10,'LineWidth',4);drawnow;hold off

figure(7);plot3(D_pos,C_pos,abs(sz-A_pos),'bo','markersize',5);axis ij;axis tight;grid

on;box on;hold on

plot3(D_neg,C_neg,abs(sz-A_neg),'r.','markersize',20);

xlabel('Exposure number');ylabel('x-coordinate');zlabel('y-coordinate');

if (n_v_of_one > max_pos)

    max_pos=n_v_of_one;

end

if (n_v_of_minus_one > max_neg)

    max_neg=n_v_of_minus_one;
```

171

```
end

end


num=images;

M=sz;

spread=zeros(M+1,M+1,num);

SECTION 2: Constructing the matrices of vortex paths


  pos_vortices=0;

  neg_vortices=0;

  total_pos_trails=0;

  total_neg_trails=0;


pos_location= NaN(max_pos,3,num);

neg_location= NaN(max_neg,3,num);

for k=1:num

  for p= 1:max_pos

    if(B_pos(p,3,k)>0)

      pos_vortices= pos_vortices+1;

      pos_location(p,1,k)=B_pos(p,2,k);

      pos_location(p,2,k)=abs(M-B_pos(p,1,k));

      pos_location(p,3,k)=k;

    else
```

172

```
                pos_location(p,3,k)=0;

            end

        end

        for m=1:max_neg

            if(B_neg(m,3,k)>0)

                neg_vortices = neg_vortices+1;

                neg_location(m,1,k)=B_neg(m,2,k);

                neg_location(m,2,k)=abs(M-B_neg(m,1,k));

                neg_location(m,3,k)=k;

            else

                neg_location(m,3,k)=0;

            end

        end

    end

'spread' contains vortex coordinates

for k=1:size(pos_location,3)

    for i=1:max_pos

        if(pos_location(i,3,k)==k)

            spread(pos_location(i,1,k)+1,pos_location(i,2,k)+1,k)=1;%1 shifted to avoid 0

        end

    end

end

for k=1:size(neg_location,3)
```

173

```matlab
    for i=1:max_neg

        if(neg_location(i,3,k)==k)

            spread(neg_location(i,1,k)+1,neg_location(i,2,k)+1,k)=-1;

        end

    end

end

for i=2:num

speckle_cor(i)=corr2(cube(:,:,1),cube(:,:,i));%Speckle Correlation, compared to Frame#1

rel_cor1(i)=corr2(cube(:,:,i-1),cube(:,:,i));

vortex_cor(i)=corr2(spread(:,:,1),spread(:,:,i));%Vortex Correlation, compared to Frame#1

rel_cor2(i)=corr2(spread(:,:,i-1),spread(:,:,i));

end


temp_pos= pos_location; %do not want to change location matrices, so introducing new matrices

temp_neg= neg_location;

%code to form matrices of trails, with each trail in a single row

pos_trails= NaN(max_pos*num,3,num); %maximum possible size; initial allocation runs faster than dynamic allocation

neg_trails= NaN(max_neg*num,3,num); %worst case scenario: every trail has a single vortex

%computing positive trails
```

```matlab
framecheck=1;

currenttrail=0;

while (framecheck<num+1)

  % checking_frame = framecheck

   for(s=1:max_pos)

     % total_pos_trails= total_pos_trails + 1;

     if(temp_pos(s,3,framecheck)>0) %checking existence

     %    found_new_vortex_at_position = s

        currenttrail=currenttrail+1;

        total_pos_trails= total_pos_trails + 1;

        currentframe=framecheck;

        count=1;

        pos_trails(currenttrail,1,count)=temp_pos(s,1,currentframe);

        pos_trails(currenttrail,2,count)=temp_pos(s,2,currentframe);

        pos_trails(currenttrail,3,count)=1;

        compare_x=temp_pos(s,1,currentframe);

        compare_y=temp_pos(s,2,currentframe);

        temp_pos(s,3,currentframe)=0;

        count=count+1;

        currentframe=currentframe+coarsing_factor;

        proceed=1;

        while(currentframe < num+1 && proceed ==1)

         %   comparing_with_frame = currentframe
```

```matlab
        need=1;

        p=1;

        while(p<max_pos+1 && need==1)
    if(abs(temp_pos(p,1,currentframe)-compare_x)<resolution &&

abs(temp_pos(p,2,currentframe)-compare_y)<resolution &&

temp_pos(p,3,currentframe)>0)

        % above line is for checking continuity and existence;

        % continuity condition can be user defined

        pos_trails(currenttrail,1,count)=temp_pos(p,1,currentframe);

        pos_trails(currenttrail,2,count)=temp_pos(p,2,currentframe);

        pos_trails(currenttrail,3,count)=1;

        compare_x=temp_pos(p,1,currentframe);

        compare_y=temp_pos(p,2,currentframe);

        temp_pos(p,3,currentframe)=0;

        need=0;

        count=count+1;

        currentframe=currentframe+coarsing_factor;
%       if(currentframe>num)
%           currentrail = currenttrail +1 ;
%       end
    end

    p=p+1;
    end
```

176

```matlab
        if (need==1)

            proceed=0;

         %   currenttrail=currenttrail+1;

         %   total_pos_trails= total_pos_trails + 1;

            end


            end

        end

    end
framecheck=framecheck+1;

%currenttrail = currenttrail+1;

end


%computing negative trails

framecheck=1;

currenttrail=0;

while (framecheck<num+1)

  %checking_frame = framecheck

  for(s=1:max_neg)


    if(temp_neg(s,3,framecheck)>0) %checking existence
```

```matlab
currenttrail=currenttrail+1;

total_neg_trails= total_neg_trails + 1;

currentframe=framecheck;

count=1;

neg_trails(currenttrail,1,count)=temp_neg(s,1,currentframe);

neg_trails(currenttrail,2,count)=temp_neg(s,2,currentframe);

neg_trails(currenttrail,3,count)=1;

compare_x=temp_neg(s,1,currentframe);

compare_y=temp_neg(s,2,currentframe);

temp_neg(s,3,currentframe)=0;

count=count+1;

currentframe=currentframe+coarsing_factor;

proceed=1;

while(currentframe < num+1 && proceed ==1)

  %  comparing_with_frame = currentframe

    need=1;

    p=1;

    while(p<max_neg+1 && need==1)

if(abs(temp_neg(p,1,currentframe)-compare_x)<resolution &&

abs(temp_neg(p,2,currentframe)-compare_y)<resolution &&

temp_neg(p,3,currentframe)>0)

    % above line is for checking continuity and existence;

    % continuity condition can be user defined
```

178

```matlab
            neg_trails(currenttrail,1,count)=temp_neg(p,1,currentframe);

            neg_trails(currenttrail,2,count)=temp_neg(p,2,currentframe);

            neg_trails(currenttrail,3,count)=1;

            compare_x=temp_neg(p,1,currentframe);

            compare_y=temp_neg(p,2,currentframe);

            temp_neg(p,3,currentframe)=0;

            need=0;

            count=count+1;

            currentframe=currentframe+coarsing_factor;

            end

            p=p+1;

              end


              if (need==1)

                proceed=0;

              %  currenttrail=currenttrail+1;


              end



              end

            end

        end

    framecheck=framecheck+1;
```

```
end


average_pos_trail_length= pos_vortices/total_pos_trails

average_neg_trail_length= neg_vortices/total_neg_trails

average_trail_length= (pos_vortices + neg_vortices)/(total_pos_trails + total_neg_trails)


%SECTION 3: Displaying in a trailwise manner, for better visualisation

    %each trail formns a column

    trailwise_pos= NaN(num,2,max_pos*num);

    trailwise_neg= NaN(num,2,max_neg*num);

    for(n=1:num)

        for(p=1:max_pos*num)

        trailwise_pos(n,1,p)=pos_trails(p,1,n);

        trailwise_pos(n,2,p)=pos_trails(p,2,n);


    end

    end

    for(n=1:num)

        for(p=1:max_neg*num)


        trailwise_neg(n,1,p)=neg_trails(p,1,n);

        trailwise_neg(n,2,p)=neg_trails(p,2,n);

    end
```

180

```matlab
    end


%SECTION 4: Calculating the deviation in vortex paths

memory_neg= NaN(max_neg*num,3,num);

memory_pos= NaN(max_pos*num,3,num);

check_x=M+5;%setting up initial values where there can never be vortices

check_y=M+5;

for p=1:max_neg*num

    m=1;

for q=1:num

    if((neg_trails(p,1,q)~=check_x |neg_trails(p,2,q) ~= check_y)&& neg_trails(p,3,q)>0)

        memory_neg(p,1,m)=neg_trails(p,1,q);

        memory_neg(p,2,m)=neg_trails(p,2,q);

        memory_neg(p,3,m)=1;

        check_x= memory_neg(p,1,m);

        check_y= memory_neg(p,2,m);

        m=m+1;

    end

end

end

check_x=M+5;%same for positive vortices

check_y=M+5;

for p=1:max_pos*num
```

181

```matlab
      m=1;

  for q=1:num

    if((pos_trails(p,1,q)~=check_x |pos_trails(p,2,q) ~= check_y)&& pos_trails(p,3,q)>0)

      memory_pos(p,1,m)=pos_trails(p,1,q);

      memory_pos(p,2,m)=pos_trails(p,2,q);

      memory_pos(p,3,m)=1;

      check_x= memory_pos(p,1,m);

      check_y= memory_pos(p,2,m);

      m=m+1;

    end

  end

end


memory_theta_pos= NaN(max_pos*num,num);

memory_theta_neg= NaN(max_neg*num,num);

for k= 2:num-1  % first and last columns are meaningless, as they cannot represent

deviations


for p=1:max_pos*num % calculations for each positive vortex, one by one

  if(memory_pos(p,3,k-1)> 0 && memory_pos(p,3,k)> 0 && memory_pos(p,3,k+1) >

0)% checking that 3 consecutive positive vortices exist

  sam1=memory_pos(p,1,k-1);%x-coordinate of previous vortex

  sam2=memory_pos(p,2,k-1);%y-coordinate of previous vortex
```

182

```matlab
sam3=memory_pos(p,1,k);%x-coordinate of current vortex

sam4=memory_pos(p,2,k);%y-coordinate of current vortex

sam5=memory_pos(p,1,k+1);%x-coordinate of next vortex

sam6=memory_pos(p,2,k+1);%y-coordinate of next vortex

memory_theta_pos(p,k)=real(acosd(real(dot(((sam3+i*sam4)-
(sam1+i*sam2)),((sam5+i*sam6)-(sam3+i*sam4))))/(norm((sam3+i*sam4)-
(sam1+i*sam2))*norm((sam5+i*sam6)-(sam3+i*sam4)))));

    end

    end

for p=1:max_neg*num %similar as above, with the negative vortices

    if(memory_neg(p,3,k-1)> 0 && memory_neg(p,3,k)> 0 && memory_neg(p,3,k+1) >
0)% checking that 3 consecutive positive vortices exist

    sam1=memory_neg(p,1,k-1);%x-coordinate of previous vortex

    sam2=memory_neg(p,2,k-1);%y-coordinate of previous vortex

    sam3=memory_neg(p,1,k);%x-coordinate of current vortex

    sam4=memory_neg(p,2,k);%y-coordinate of current vortex

    sam5=memory_neg(p,1,k+1);%x-coordinate of next vortex

    sam6=memory_neg(p,2,k+1);%y-coordinate of next vortex

    memory_theta_neg(p,k)=real(acosd(real(dot(((sam3+i*sam4)-
(sam1+i*sam2)),((sam5+i*sam6)-(sam3+i*sam4))))/(norm((sam3+i*sam4)-
(sam1+i*sam2))*norm((sam5+i*sam6)-(sam3+i*sam4)))));

    end

    end
```

183

```matlab
end
% figure(13);hist(memory_theta_pos,36);xlabel('Deviation in
degrees');ylabel('Frequency'); title('Deviation of positive vortices');
% figure(14);hist(memory_theta_neg,36);xlabel('Deviation in
degrees');ylabel('Frequency'); title('Deviation of negative vortices');


%better visualisation trailwise
    memory_trailwise_pos= NaN(num,2,max_pos*num);
    memory_trailwise_neg= NaN(num,2,max_neg*num);
    for(n=1:num)

      for(p=1:max_pos*num)

      memory_trailwise_pos(n,1,p)=memory_pos(p,1,n);

      memory_trailwise_pos(n,2,p)=memory_pos(p,2,n);


      end

      end
    for(n=1:num)

      for(p=1:max_neg*num)

      memory_trailwise_neg(n,1,p)=memory_neg(p,1,n);

      memory_trailwise_neg(n,2,p)=memory_neg(p,2,n);

       end

      end
```

184

```matlab
%%
% x=memory_theta_pos(:);
% [f,xi] = ksdensity(x,'width',10);
% %xi=xi./mean(xi);
% %figure(2);plot(xi,f./sum(f));%Normalizes
% figure(12);plot(xi,f);
% grid on
% %figure(13);semilogy(xi/mean(xi),f./.0129);
% xlabel('Angle');ylabel('P(I)');axis([0 180 0 .025]);
% %title(['PDF for ' ,corr]);
% hold on
% y=memory_theta_neg(:);
% [f,yi] = ksdensity(y,'width',10);
% %xi=xi./mean(xi);
% %figure(2);plot(xi,f./sum(f));%Normalizes
% figure(12);plot(yi,f,'r');
% legend('positive','negative')
% grid on
% %figure(13);semilogy(xi/mean(xi),f./.0129);
% xlabel('Angle');ylabel('P(I)');
% %title(['PDF for ' ,corr])
```

185

```matlab
%SECTION 5: Plots and statistical calculations


pos_count=zeros(num,1);

neg_count=zeros(num,1);

for(i=1:num)

 count=0;

for(j=1:max_pos)


if(pos_location(j,3,i)>0)

   count=count+1;

end

end

pos_count(i)=count;

 count=0;

for(j=1:max_neg)


if(neg_location(j,3,i)>0)

   count=count+1;

end

end

neg_count(i)=count;

end
```

186

```matlab
% figure(14);plot(pos_count,'r.','markersize', 18);hold on;

% plot(neg_count,'b.','markersize', 18)

% xlabel('Frames');ylabel('No.of vortices');

% %title(['Vortex count for ' ,corr, '(Red:+1, Blue:-1)'])


pos_traillength=zeros(total_pos_trails,1);

neg_traillength=zeros(total_neg_trails,1);

for(i=1:total_pos_trails)

 count=0;

for(j=1:num)


if(isfinite(trailwise_pos(j,2,i)))

   count=count+1;

end

end

pos_traillength(i)=count;

end

for(i=1:total_neg_trails)

 count=0;

for(j=1:num)


if(isfinite(trailwise_neg(j,2,i)))
```

187

```matlab
        count=count+1;
    end
end
neg_traillength(i)=count;
end
% figure(15);plot(pos_traillength,'r.','markersize', 18);hold on;
% plot(neg_traillength,'b.','markersize', 18)
% xlabel('Trail #');ylabel('Trail Length');
% %title(['Trail lengths for ' ,corr, '(Red:+1, Blue:-1)'])
%
% standard_deviation_pos_traillength= std(pos_traillength)
% standard_deviation_neg_traillength= std(neg_traillength)


total_pos_trails
total_neg_trails



pos_motions=zeros(total_pos_trails,1);
neg_motions=zeros(total_neg_trails,1);
total_pos_motions=0;
total_neg_motions=0;
for(i=1:total_pos_trails)
 count=0;
```

188

```matlab
for(j=1:num)

    if(isfinite(memory_trailwise_pos(j,2,i)))
        count=count+1;
    end
end
pos_motions(i)=count-1;

total_pos_motions = total_pos_motions + pos_motions(i);

end
for(i=1:total_neg_trails)
 count=0;
for(j=1:num)

    if(isfinite(memory_trailwise_neg(j,2,i)))
        count=count+1;
    end
end
neg_motions(i)=count-1;

total_neg_motions = total_neg_motions + neg_motions(i);
end
% figure(16);plot(pos_motions,'r.','markersize', 18);hold on;
% plot(neg_motions,'b.','markersize', 18)
% xlabel('Trail #');ylabel('# Motions');
```

189

```matlab
% %title(['Net Motions for ' ,corr, '(Red:+1, Blue:-1)'])


pos_motion_probability = total_pos_motions / pos_vortices

neg_motion_probability = total_neg_motions / neg_vortices


mean_pos = NaN(num,2);

mean_neg = NaN(num,2);

mean_location= NaN(num,2);


for(i=1:num)

 mean_pos(i,1)= mean(pos_location(1:pos_count(i),1,i));

 mean_pos(i,2)= mean(pos_location(1:pos_count(i),2,i));

 mean_neg(i,1)= mean(neg_location(1:neg_count(i),1,i));

 mean_neg(i,2)= mean(neg_location(1:neg_count(i),2,i));


 mean_location(i,1)= (mean_pos(i,1)*pos_count(i) +

mean_neg(i,1)*neg_count(i))/(pos_count(i) + neg_count(i));

 mean_location(i,2)= (mean_pos(i,2)*pos_count(i) +

mean_neg(i,2)*neg_count(i))/(pos_count(i) + neg_count(i));


% figure(17);%plot3(i,mean_pos(i,1),mean_pos(i,2),'r.','markersize',20);axis ij;axis

tight;grid on;box on;hold on
```

190

```matlab
% %plot3(i,mean_neg(i,1),mean_neg(i,2),'b.','markersize',20); hold on;
% plot3(i,mean_location(i,1),mean_location(i,2),'g.','markersize',20); hold on;
% xlabel('Frame number');ylabel('x-coordinate');zlabel('y-coordinate');
% title(['Path of mean vortex (Blue -1; Red +1; Green Central)']);
end


pos_displacement= NaN(num,1);

neg_displacement= NaN(num,1);

central_displacement = NaN(num,1);

gap= NaN(num,1);


pos_displacement(1)=0;

neg_displacement(1)=0;

central_displacement(1)=0;

gap(1)= sqrt(power(mean_pos(1,1)-mean_neg(1,1),2)+power(mean_pos(1,2)-
mean_neg(1,2),2));


for(i=2:num)
   gap(i)= sqrt(power(mean_pos(i,1)-mean_neg(i,1),2)+power(mean_pos(i,2)-
mean_neg(i,2),2));
   pos_displacement(i)= sqrt(power(mean_pos(i,1)-mean_pos(i-
1,1),2)+power(mean_pos(i,2)-mean_pos(i-1,2),2));
```

191

```matlab
    neg_displacement(i)= sqrt(power(mean_neg(i,1)-mean_neg(i-
1,1),2)+power(mean_neg(i,2)-mean_neg(i-1,2),2));

    central_displacement(i)= sqrt(power(mean_location(i,1)-mean_location(i-
1,1),2)+power(mean_location(i,2)-mean_location(i-1,2),2));
end


average_positive_displacement = mean(pos_displacement)

average_negative_displacement = mean(neg_displacement)

average_central_displacement = mean(central_displacement)

average_gap = mean(gap)

stdev_positive_displacement = std(pos_displacement)

stdev_negative_displacement = std(neg_displacement)

stdev_central_displacement = std(central_displacement)

stdev_gap = std(gap)


% figure(18);plot(pos_displacement,'r.','markersize', 18);hold on;

% plot(neg_displacement,'b.','markersize', 18); hold on;

% plot(central_displacement,'y.','markersize', 18); hold on;

% plot(gap,'g.','markersize', 18);

% xlabel('Frame #');ylabel('Distance in coordinate points');

% title(['Net Displacements (Red:+1, Blue:-1, Green: Gap, Yellow:Central)'])


% SECTION 6: Code for Poincare plots
```

192

```matlab
array1= gap;param='Gap';
%options:"pos_count","neg_count","pos_count+neg_count","pos_displacement","neg_di
splacement","central_displacement","gap"
len=length(array1);
ele=len+1;
array2=zeros(ele,1);
array2(1)=0;
array2(2:ele)=array1;


% center=(min(array1)+ max(array1))/2;


sdsd=sqrt(mean((array1(2:len)-array2(2:len)).^2)-(mean(array1(2:len)-array2(2:len)))^2);
sdrr=sqrt(mean((array1).^2)-(mean(array1))^2);
sd1=sqrt((sdsd^2)/2);
sd2= sqrt(2*sdrr^2-sd1^2);
e=sd2/sd1
% figure(19);
% plot(array2(2:ele-1),array1(2:ele-1),'r.');
% title(['Poincare Plot  of ', param]);
% xlabel('Previous measure');ylabel('Next Measure');hold on;
% plot(1:1.5*max(array1),1:1.5*max(array1),'--');hold on;
% t=0:0.01:2*pi;
% sds1=1.01;
```

193

```matlab
% sds2=1.85
% xx = sds2*sin(t) ;
% yy = sds1*cos(t) ;
% xx2 = xx*cos(pi/4) - yy*sin(pi/4) + center;
% yy2 = xx*sin(pi/4) + yy*cos(pi/4) + center;
% plot(xx2,yy2);axis equal;


% str={'sd1=',sd1,'sd2=',sd2,'mult=',mult,'corr=',corr};
% text(max(array1)/10,1.2*max(array1),str);


%for intensity on each column
% mid= round(M/2);
% for nnn=1:num
% hor_intensity (nnn,:)= cube(mid,:,nnn);
% vert_intensity (nnn,:)= cube(:,mid,nnn);
% end
%%
figure(20);plot(speckle_cor);title('Speckle Correlation');
figure(21);plot(vortex_cor);title('Vortex Correlation');
figure(22);plot(rel_cor1);title('Speckle Rel Correlation');
figure(23);plot(rel_cor2);title('Vortex Rel Correlation');
```

194

B.3 Power spectral density (PSD) function creation and plotting

(Primarily used for calculations in Chapter 5)

(Variable "cube" needs to be created using either B.1 or B.2 above and input through the first 2 lines here)

```
clearvars -except cube M;

arr = cube(:,:,31);

psd=(fft2(arr)).*conj(fft2(arr));

image(fftshift(psd));axis equal;axis tight;

k=size(arr);

len=k(1);

hold on;

ang=0:0.01:2*pi;

for r=2:10

    rad= len/r;

    plot(len/2+rad*cos(ang), len/2+rad*sin(ang));hold on;

end
```

B.4 Measuring Poincaré analysis parameters with changing coarsing factors

(Primarily used for calculations in chapter 5; Can be used for calculating all parameters described in chapter 2)

(The variable "cube" needs to be imported from B.1 or B.2)

```
array1= cube(10,:,35);
```

195

```matlab
len=length(array1);

%fact

phi0= var(array1);% same value for cf=0

gamma0=mean(array1.^2);% same value for cf=0

%array1=correl(:,200);

for cf=1:20 %coarsing factor  %for standard poincare, cf =1

sum=0;

sum_gamma=0;

sum_phi=0;

for i=1:len-cf

sum=sum+array1(i)-array1(i+cf);

sum_gamma=sum_gamma+array1(i)*array1(i+cf);

sum_phi=sum_phi+(array1(i)-mean(array1))*(array1(i+cf)-mean(array1));

end


sd1= sqrt(var(array1(1:len-cf)-array1(cf+1:len))/2);

sd2= sqrt(2*var(array1)-sd1^2);



% plot(array2(cf+1:len),array1(cf+1:len),'r.');

% title('Poincare Plot of Intensity');

% xlabel('Previous measure');ylabel('Next Measure');hold on;

% plot(0.01:1.5*max(array1),0.01:1.5*max(array1),'--');hold on;
```

196

```matlab
%
% for t=0:0.01:2*pi
%     x_new = sd2*cos(t);
%     y_new = sd1*sin(t);
%     x= x_new*cos(pi/4)- y_new*sin(pi/4)+center;
%     y= x_new*sin(pi/4) + y_new*cos(pi/4)+center;
%     plot(x,y,'r.','markersize',20);hold on;axis equal;
% end
cf


error1 = 2*(mean(array1)^2)-(mean(array1))*(mean(array1(1:len-
cf))+mean(array1(1+cf:len)));
error2=(sum/(len-cf))^2;
error3 = mean(array1(1:len-cf).^2) + mean(array1(1+cf:len).^2)- 2*mean(array1.^2);
error= error1+ (error3-error2)/2;
gamma_k=sum_gamma/(len-cf);
phi_k=sum_phi/(len-cf);
%check_eqn7_phi_k= gamma_k + error1 - (mean(array1)^2)
%sd1
%sd2
e=sd2/sd1
% str={'sd1=',sd1,'sd2=',sd2,'esf=',esf};
% text(max(array1)/10,0.5*max(array1),str)
```

197

end

B.5 Measurements on speckle images obtained from fluid phantom and animal tissues

(Primarily used for calculations in Chapter 5)

```matlab
close all

clear all

%

warning off

% Import

images=256;%No. of images to input

skip_factor=1;%'1' for no skipping

coarsing_factor=1;

datapath= 'F:\Photonics_west_data\Sar_data\';

imagepath= 'F:\phantom\BW\';

%prefix='fc2_save_2018-06-22-163243-';

format='.bmp';      %input

for folder = 51:67 % folders are numbered

% for count = 0:5

% if(count<=6)

%   datapath= 'F:\Photonics_west2109_data\Saratov_data\Mouse_data\Mouse capillaries

speckle\';
```

```matlab
% imagepath= 'F:\mouse\BW\';

% folder=count+1;

% M=1032;

% N=1032;

% else

% datapath= 'F:\Photonics_west_data\Sar_data\';

% imagepath= 'F:\phantom\BW\';

% if(count<=2)

%     folder=count;

% else folder=count+30;

% end


M=1024;

N=1024;


% count
folder
for num=1:images

 %num

 im= imread(strcat(datapath,num2str(folder,'%02g'),'\',num2str((num-

1),'%04g'),format));

 im=im(:,:,1);

 im=double(im);
```

199

```matlab
%raw_cube(:,:,num)=im;

%imfilt=imgaussfilt(im,12);

cube(:,:,num)=im;
end
% M= 1032; %no. of rows

% N=1032; % no. of columns

%end AM import

coarsing_factor=1;      % set '1' for standard poincare default

sd1_matrix = NaN(N,M);

sd2_matrix = NaN(N,M);

esf_matrix = NaN(N,M);

mssd_matrix= NaN(N,M);

for i = 1:M

    for j = 1:N

        for k = 1:num   % This loop is needed to make array1 2-D

array1(k)= cube(i,j,k); % obtained from the cube of intensity

        end

len=length(array1);

ele=len+coarsing_factor;

array2=zeros(1,ele);

array2(1:coarsing_factor)=0;

array2(coarsing_factor+1:ele)=array1;
```

```matlab
sdsd=sqrt(mean((array1(coarsing_factor+1:len)-array2(coarsing_factor+1:len)).^2)-
(mean(array1(coarsing_factor+1:len)-array2(coarsing_factor+1:len)))^2);

sdrr=sqrt(mean((array1).^2)-(mean(array1))^2);

sd1=sqrt((sdsd^2)/2);

sd2= sqrt(2*sdrr^2-sd1^2);

sum=0;

for t=1:len-coarsing_factor

sum=sum+((array1(t)-array1(t+coarsing_factor))^2);

end

mssd_matrix(i,j)=sum/(len-coarsing_factor);%mean squared successive differences

sd1_matrix(i,j)=sd1;

sd2_matrix(i,j)=sd2;

esf_matrix(i,j)=sd2/sd1;

% test_matrix(i,j)=(sd2-sd1)./(sd2+sd1);

    end

end

%The following is to display the images

figure(1);imagesc(sd1_matrix);title('SD1 value for each pixel');colorbar;%colormap gray

figure(2);imagesc(sd2_matrix);title('SD2 value for each pixel');colorbar;%colormap gray

figure(3);imagesc((test_matrix));title('(Long Term - Short Term Variation) / Total

Variation');colorbar;%colormap gray

figure(4);imagesc(sqrt(esf_matrix));title('Ellipticity (=SD2/SD1) value for each

pixel');colorbar;%colormap gray
```

201

```matlab
figure(5);imagesc(mssd_matrix);title('MSSD value for each pixel');colorbar; %colormap
gray

sd1_max=max(max(sd1_matrix))

sd2_max=max(max(sd2_matrix))

e_min=min(min(esf_matrix))

mssd_max= max(max(mssd_matrix))


sd1_mean_in=nanmean(sd1_matrix(620,:))

sd2_mean_in=nanmean(sd2_matrix(620,:))

e_mean_in=nanmean(esf_matrix(620,:))

mssd_mean_in= nanmean(mssd_matrix(620,:))


sd1_mean_out=nanmean(sd1_matrix(320,:))

sd2_mean_out=nanmean(sd2_matrix(320,:))

e_mean_out=nanmean(esf_matrix(320,:))

 mssd_mean_out= nanmean(mssd_matrix(320,:))


%The following is to save the images

fig1=imagesc(sd1_matrix);title('SD1 value for each pixel');colorbar;colormap gray;

saveas(fig1, [strcat(imagepath,num2str(folder,'%02g'),'\fig1')], 'bmp');

fig2=imagesc(sd2_matrix);title('SD2 value for each pixel');colorbar;colormap gray;

saveas(fig2, [strcat(imagepath,num2str(folder,'%02g'),'\fig2')], 'bmp');
```

202

```matlab
fig3=imagesc((test_matrix));title('(Long Term - Short Term Variation) / Total
Variation');colorbar;colormap gray;
saveas(fig3, [strcat(imagepath,num2str(folder,'%02g'),'\fig3')], 'bmp');
fig4=imagesc(sqrt(esf_matrix));title('Ellipticity (=SD2/SD1) value for each
pixel');colorbar;colormap gray;
saveas(fig4, [strcat(imagepath,num2str(folder,'%02g'),'\fig4')], 'bmp');
fig5=imagesc(mssd_matrix);title('MSSD value for each pixel');colorbar; colormap gray;
saveas(fig5, [strcat(imagepath,num2str(folder,'%02g'),'\fig5')], 'bmp');


% %The following is to draw the Poincare plots
choose_x= 258; %Between 0 and M, choose the x-coordinate of point for which Poincare
plot needs to be drawn
choose_y= 242; %Between 0 and M, y-coordinate
for k = 1:num   % This loop is needed to make array1 2-D
array1(k)= cube(choose_x,choose_y,k); % obtained from the cube of intensity
    end
len=length(array1);
ele=len+coarsing_factor;
array2=zeros(1,ele);
array2(1:coarsing_factor)=0;
array2(coarsing_factor+1:ele)=array1;
sdsd=sqrt(mean((array1(coarsing_factor+1:len)-array2(coarsing_factor+1:len)).^2)-
(mean(array1(coarsing_factor+1:len)-array2(coarsing_factor+1:len)))^2);
```

```matlab
sdrr=sqrt(mean((array1).^2)-(mean(array1))^2);

sd1=sqrt((sdsd^2)/2);

sd2= sqrt(2*sdrr^2-sd1^2);

center=(min(array1)+ max(array1))/2;

figure(10);plot(array2(coarsing_factor+1:len),array1(coarsing_factor+1:len),'r.');hold on

title('Poincare Plot of Intensity ');

xlabel('Previous measure');ylabel('Next Measure');hold on;

plot(0.01:1.5*max(array1),0.01:1.5*max(array1),'b-','linewidth',2);hold on;

for t=0:0.01:2*pi

    x_new = sd2*cos(t);

    y_new = sd1*sin(t);

    x= x_new*cos(pi/4)- y_new*sin(pi/4)+center;

    y= x_new*sin(pi/4) + y_new*cos(pi/4)+center;

    plot(x,y,'r.','markersize',20);hold on;axis equal;

end


str={'sd1=',sd1,'sd2=',sd2,'e=',sd2/sd1};

text(max(array1)/10,0.8*max(array1),str)



% Examine pdf of ellipticity

% x=esf_matrix(:);

% [f,xi] = ksdensity(x);
```

204

```matlab
% %xi=xi./mean(xi);

% figure(5);plot(xi,f./sum(f),'k','linewidth',2);%Normalizes

% grid on

% %xlabel('I/<I>');ylabel('P(I)');%axis([0 2.5 0 1]);

% title('PDF ');

% hold on


%plot a single pixel intensity through stack

% I=squeeze(cube(500,193,[1:num]));

% figure(6);plot([1:num],I);

% xlabel('Frame number');ylabel('Pixel intensity');

% title('Intensity of single pixel through speckle cube');

%%

%start SJK contrast measurements

% % cube=imread('C:\Russia_Saratov2017\Saratov_data\Mouse_data\Mouse capillaries

speckle\18\0100.bmp');

% % cube=double(cube);

% % cube=cube(:,:,1);

% cube=cube(:,:,100);

% %num=100;

% %

%

%
```

205

```matlab
% % Spatial contrast

% %for k=no_frames:num-no_frames

% kernel=ones(3,3);

% sc=size(kernel);

% Nk=sc(1)*sc(2);

% fac=Nk/(Nk-1);

% mu_x=imfilter(cube,kernel,'conv')/Nk;

% x_sq=imfilter(cube.*cube,kernel,'conv')/(Nk-1);

% var_x=sqrt(x_sq - fac*mu_x.^2);

% Cs=var_x./mu_x;

% s = fix(3/2);

% Cs(1:s,:,:) = [];

% Cs(end-s+1:end,:,:) = [];

% Cs(:,1:s,:) = [];

% Cs(:,end-s+1:end,:) = [];

% %figure(13);imagesc(Cs,[0 1.5]);colorbar

% figure(6);imagesc(Cs);colorbar;%colormap gray

% title(['Spatial contrast frame']);%pause

% % for j=1:num

% %h=fspecial('gaussian',[15 15],.1);

% h=fspecial('disk',[5]);

% filteredCs = imfilter(Cs, h);

% figure(7); imagesc(filteredCs);colorbar;title(['Spatial Contrast']);
```

206

```matlab
%
% %Temporal Contrast
% depth=num;
% kernel=ones(1,1,depth);
% Nk=depth;
% fac=Nk/(Nk-1);
% mu_x=imfilter(cube,kernel,'conv')/Nk;
% x_sq=imfilter(cube.*cube,kernel,'conv')/(Nk-1);
% var_x=real(sqrt(x_sq - fac*mu_x.^2));
% Ct=var_x./mu_x;
% s = fix(3/2);
% Ct(1:s,:,:) = [];
% Ct(end-s+1:end,:,:) = [];
% Ct(:,1:s,:) = [];
% Ct(:,end-s+1:end,:) = [];
% %Ct=squeeze(Ct(:,:,depth));
% %Ct=flipud(Ct);
% figure(8);imagesc(Ct);colorbar;title(['Temporal Contrast']);
%end SJK contrast measurements


%start AM contrast measurements
%Temporal contrast %calculate this before spatial contrast to avoid issues
%with the variable 'std'
```

```matlab
for i=1:M
    for j=1:N
mu(i,j)=mean(cube(i,j,:));
sigma(i,j)=std(cube(i,j,:));
temp_K(i,j)=sigma(i,j)/mu(i,j);
    end
end
figure(6);imagesc(temp_K);colorbar;title('Temporal Contrast ');
fig6=imagesc(temp_K);colorbar;title('Temporal Contrast ');colormap gray;
saveas(fig6, [strcat(imagepath,num2str(folder,'%02g'),'\fig6')], 'bmp');
%Integrated image Spatial contrast
frame=images;%No. of images to add for spatial contrast %measure of camera
integration
cw=7;%contrast window
K=zeros(size(cube,1),size(cube,2));
matres(:,:,1)=cube(:,:,1);
for i=2:frame
    matres(:,:,i)=matres(:,:,i-1)+cube(:,:,i);
%    i %to know where the loop is
end
for i=1+floor(cw/2):M-floor(cw/2)
    for j=1+floor(cw/2):N-floor(cw/2)
```

208

```matlab
        spat_K(i,j)=std2(matres(i-floor(cw/2):i+floor(cw/2),j-
floor(cw/2):j+floor(cw/2),frame))/mean(mean(matres(i-floor(cw/2):i+floor(cw/2),j-
floor(cw/2):j+floor(cw/2),frame)));
    end
end
figure(7);imagesc(spat_K);colorbar;title('Spatial Contrast (for integrated image)');
fig7=imagesc(spat_K);colorbar;title('Spatial Contrast (for integrated image)');
saveas(fig7, [strcat(imagepath,num2str(folder,'%02g'),'\fig7')], 'bmp');
% %Single Frame Spatial Contrast
%For future purposes; NOT in dissertation
sing_frame=cube(:,:,25); %input frame
for i=1+floor(cw/2):M-floor(cw/2)
    for j=1+floor(cw/2):N-floor(cw/2)
        spat_K2(i,j)=std2(sing_frame(i-floor(cw/2):i+floor(cw/2),j-
floor(cw/2):j+floor(cw/2)))/mean(mean(sing_frame(i-floor(cw/2):i+floor(cw/2),j-
floor(cw/2):j+floor(cw/2))));
    end
end
%figure(8);imagesc(spat_K2);colorbar;title('Spatial Contrast (for single frame)');
fig8=imagesc(spat_K2);colorbar;title('Spatial Contrast (for single frame)');
saveas(fig8, [strcat(imagepath,num2str(folder,'%02g'),'\fig8')], 'bmp');
tempK_min=min(min(temp_K))
tempK_max=max(max(temp_K))
```

209

```matlab
tempK_mean=mean(temp_K(620,:))

%end AM contrast measurements

end
```