Ultrafast Imaging in Combustion

M. Tech Thesis

By

Kushagra Parihar

Department of Mechanical Engineering

Indian Institute of Technology Indore

May,2025

Ultrafast Imaging in Combustion

A THESIS

Submitted in the partial fulfillment of the

requirements for the award of the degree

of

Master of Technology

by

Kushagra Parihar

Department of Mechanical Engineering

IIT INDORE

Indian Institute of Technology Indore

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **Ultrafast Imaging in Combustion** in the partial fulfillment of the requirements for the award of the degree of **Master of Technology** and submitted to **Department of Mechanical Engineering, IIT Indore,** is an authentic record of my own work carried out during the period from June 2024 to May 2025 under the supervision of **Dr. Devendra Deshmukh, Professor, Department of Mechanical Engineering, IIT Indore.**

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other institution.

(Kushagra Parihar)

This is to certify that the above statement made by the candidate is correct to the best of my/our knowledge.

(Dr. Devendra Deshmukh)

Mr. Kushagra Parihar has successfully given his M. Tech Oral examination held on <26/05/2025>

Signature of Supervisor of M. Tech. thesis 26/5/25

Date:

Convener, DPGC

Date: 02-06-2025

ACKNOWLEDGEMENTS

First and foremost, I would like to express my heartfelt gratitude to my supervisor, **Dr. Devendra Deshmukh**, for his expert guidance, unwavering support, and insightful feedback throughout my two years of **Master of Technology**. His encouragement has been instrumental in shaping this work.

A special note of appreciation goes to Devashish Chorey and Mr. Arpit Joglekar for their constant moral support and for creating a collaborative and motivating environment that helped me stay focused and complete this work successfully. I would also like to acknowledge Mr. Anshul Kashyap, Kartik Kumar Gupta, Kartik Kumar, Singh Prashant and my fellow batchmates for their moral support and camaraderie throughout this journey. I am thankful to Mr. Ashwin Wagh, Department Manager, Spray and Combustion Lab, IIT Indore, for his guidance and support during this project.

Above all, I am deeply grateful to my parents for their unconditional love, blessings, and encouragement, which have been the foundation of my strength and perseverance.

Finally, I extend my thanks to everyone who, in any way, has supported and contributed to the completion of this project.

With Regards,

Kushagra Parihar

Dedicated

To

My beloved

Parents

Abstract

The evolution of imaging technologies—from Galileo's telescope to modern high-speed cameras—has significantly advanced scientific exploration. While spatial resolution has traditionally been the focus, the ability to capture events at extremely short time scales, or temporal resolution, is equally crucial for observing ultrafast phenomena such as shockwaves, droplet dynamics, and plasma discharges. Traditional high-speed cameras, however, face limitations in frame rate, cost, and data volume.

This thesis explores **Compressed Imaging**, a transformative approach that overcomes these limitations by capturing temporally encoded information in a single shot and reconstructing it computationally into high-speed video. We focus on **Compressed Optical Shearing Ultrafast Photography (COSUP)**—a cost-effective and simple setup that leverages compressed sensing principles to achieve ultrafast imaging without requiring expensive or complex hardware.

By developing and optimizing a COSUP-based imaging system, this work demonstrates the ability to record high-temporal-resolution videos of fast physical phenomena using a single frame. The methodology reduces experimental repetition and data overhead, offering a practical solution for high-speed imaging in resource-constrained environments. The results underline COSUP's potential as a powerful tool in scientific diagnostics, fluid dynamics, and other domains where capturing transient events is critical

Table of Contents

List of Figures

Chapter 1: Introduction

1.1 Historical Background and Motivation

Imaging has been a cornerstone of scientific discovery for centuries. The journey began with Galileo's telescope in the early 1600s, which revolutionized astronomy and provided humankind with its first glimpse of celestial bodies in detail. As technology evolved, the development of photographic techniques, X-rays, electron microscopy, and digital imaging expanded our capacity to visualize the microscopic and the vast.

Historically, the focus in imaging systems has been on enhancing spatial resolution—the ability to discern fine structural details in a scene. However, as scientific inquiry moved into domains involving rapid physical, chemical, and biological processes, temporal resolution—the ability to capture events occurring over extremely short time scales—became equally crucial. For example, processes such as laser-matter interactions, cavitation in fluids, or neural activity in the brain happen so rapidly that traditional imaging systems fail to resolve them meaningfully.

Although high-speed cameras have bridged some of this gap by providing frame rates in the order of hundreds of thousands or even millions of frames per second (fps), they come with several drawbacks. These systems are often expensive, bulky, and generate large amounts of data that are difficult to store and process. More critically, they frequently require multiple experimental runs to capture different stages of a phenomenon—an approach that is not viable for non-repeatable or destructive events.

This need for a more efficient and scalable imaging method laid the foundation for the adoption of Compressed Sensing (CS) in high-speed imaging. Emerging from advances in signal processing and optimization theory in the early 2000s, CS has enabled a new generation of imaging techniques that capture detailed dynamic information using a single frame, thereby overcoming many of the limitations of conventional systems.

1.2 Concept of Compressed Sensing

Compressed sensing is based on the insight that most natural signals contain redundancies and are not entirely random or complex. For instance, a video of a falling object or a propagating wave can often be predicted using only a few parameters. CS takes advantage of this sparsity to reduce the number of measurements required to reconstruct a signal or image.

In the context of high-speed imaging, CS enables the capture of ultrafast scenes in a single exposure, using an optical setup that encodes both spatial and temporal information into a single compressed frame. This frame is later decoded using powerful computational algorithms, which reconstruct the full sequence of frames that would have otherwise required high-speed continuous capture.

This approach fundamentally shifts the burden from hardware (fast sensors, large memory buffers) to software (efficient algorithms, smart encoding), resulting in cheaper, faster, and more flexible imaging systems.

1.3 Compressed Imaging: A Paradigm Shift

Compressed imaging represents a breakthrough in the way dynamic scenes are recorded. Instead of acquiring individual frames sequentially like a traditional camera, it captures a coded projection of the entire temporal event in one go. This allows scientists to document a transient event—such as a spark, explosion, or biological impulse—in a single snapshot.

This approach relies heavily on the synergy between:

- Optical encoding (using masks or modulators to encode spatial and temporal information),
- Photon detection (via standard or specialized sensors), and
- Computational decoding (via optimization or learning-based reconstruction techniques).

What makes compressed imaging so powerful is its flexibility. By changing the encoding scheme or reconstruction algorithm, one can tune the system to suit various applications and constraints—whether it's maximizing frame rate, enhancing resolution, or minimizing noise.

Two landmark systems in this domain are:

- Single-Shot Compressed Ultrafast Photography (CUP) – known for ultra-high-speed capabilities.
- Compressed Optical Shearing Ultrafast Photography (COSUP) – known for its simpler and more affordable setup.

1.4 Single-Shot Compressed Ultrafast Photography (CUP)

CUP is one of the most advanced forms of compressed imaging available today. It captures events at frame rates exceeding 1 billion fps, allowing researchers to observe phenomena that were once thought to be too fast to image. CUP systems integrate a streak camera, which transforms temporal variations into spatial displacements, along with a spatial light modulator (SLM) or digital micromirror device (DMD) that adds a unique encoding to the incoming light.

As the encoded light enters the streak camera, it is deflected over time across a fluorescent screen, resulting in a 2D pattern that contains embedded information about the entire temporal sequence of the event. This pattern is then captured by a standard camera, and software is used to reconstruct the complete high-speed video.

Despite its complexity, CUP has proven to be invaluable in cutting-edge applications such as:

- Visualizing the propagation of light through various media,
- Investigating plasma arcs and combustion events,
- Observing fast biological responses like neuron firing or muscle contraction.

The system is ideal for single-occurrence events that cannot be repeated, such as detonations, rare natural phenomena, or sensitive biological responses.

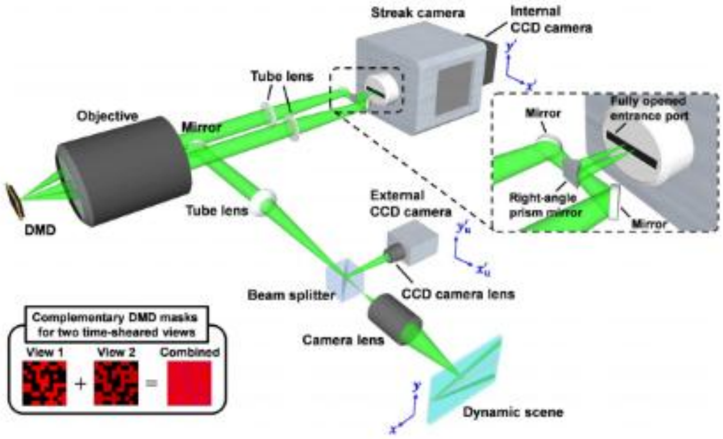


Fig1.1: Single Shot CUP

1.5 Compressed Optical Shearing Ultrafast Photography (COSUP)

COSUP offers a more accessible approach to compressed imaging without sacrificing much in performance. It uses a Galvano scanner—a device with a rapidly oscillating mirror—to shear the incoming light across the sensor during the exposure. As a result, each pixel on the sensor captures light from slightly different moments in time, effectively compressing a video sequence into a single frame.

To aid in the decoding process, a binary mask generated by a DMD is used to encode additional spatial information into the image. This mask acts like a "barcode" that helps algorithms to unravel the temporal sequence during post-processing.

COSUP can achieve frame rates up to 1.5 million fps, which is ideal for capturing:

- Fluid dynamics, such as droplet impacts and turbulence,
- Micro-explosions in combustion engines,
- Mechanical deformation under stress.

Unlike CUP, COSUP is more compact, cost-effective, and easier to integrate with existing microscopy or imaging setups, making it popular in research labs with limited resources.

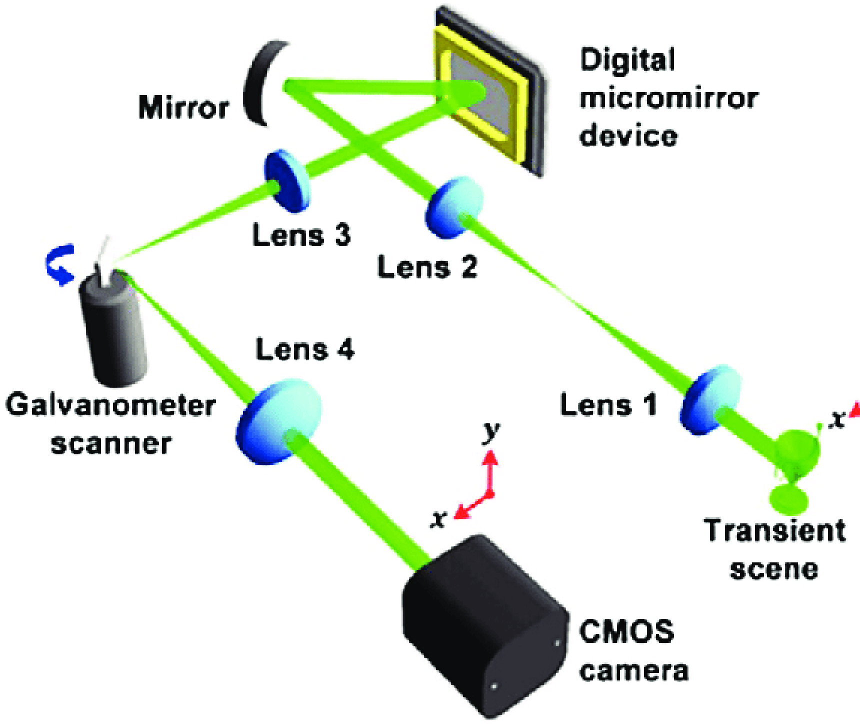


Fig 1.2: COSUP set-up

1.6 Comparison with Traditional High-Speed Cameras

While traditional high-speed cameras have played a crucial role in scientific imaging, they have several limitations:

- They are expensive and require specialized cooling and synchronization systems.
- They often miss key moments due to limitations in buffer memory or trigger delays.
- They demand repetitive testing to build a full picture of an event, which is impractical for non-repeatable experiments.

In contrast, compressed imaging techniques like CUP and COSUP:

- Capture all necessary data in a single frame, eliminating timing and buffering issues.

- Reduce the cost and complexity of the setup by using computational power to replace expensive hardware.
- Enable high-throughput experiments, where time is critical and setup needs to be minimal.

1.7 Applications in Science and Industry

Compressed sensing-based imaging is already transforming a wide range of scientific and industrial applications:

In Aerospace and Defence:

- Visualizing shockwave interactions around supersonic projectiles.
- Monitoring fuel combustion inside rocket and jet engines.
- Analysing laser targeting systems and impact damage.

In Biology and Medicine:

- Capturing cellular dynamics such as mitosis or signal transduction.
- Recording neural activity with millisecond resolution.
- Enabling real-time diagnostics for critical care scenarios.

In Material Science:

- Studying fracture propagation and structural failures in materials.
- Investigating phase changes under thermal or mechanical stress.

In Thermal and Spectral Imaging:

- Detecting rapid temperature changes in hot surfaces.
- Conducting chemical identification via hyperspectral analysis.

1.8 Emerging Trends and Future Directions

The field of compressed sensing in high-speed imaging is rapidly evolving, with several exciting frontiers:

Deep Learning Integration:

New deep neural networks can learn to reconstruct high-speed sequences more accurately and efficiently than traditional optimization methods. These systems can even operate in real time, enabling live feedback and control in industrial applications.

Meta surface-Based Optics:

Advanced flat optics, known as Meta surfaces, are being developed to reduce the size and complexity of optical systems. These could replace bulky lenses and mirrors with thin, tuneable surfaces, making compressed imaging more portable and scalable.

3D and Holographic Imaging:

Efforts are underway to extend compressed sensing into three-dimensional and volumetric domains. This would allow scientists to not only record rapid events but also understand their spatial evolution over time.

Real-Time Monitoring:

The combination of fast reconstruction algorithms and low-latency hardware may soon enable real-time compressed sensing for applications in robotics, surveillance, and live diagnostics, where every millisecond counts.

Chapter 2: Literature Review

CoSUP is a very new technology, comes into picture since 2016 not much work has been done in this, but some of the literature review is as follows:

1. A thesis by 'Constanza Cendon Contreras' gives the idea about Compressed Optical Streaking Ultra High-Speed Photography, a technique that reduces both high cost and high memory requirements in situ and delivers frame rates up to 400,000 fps after postprocessing is used for a variety of experiments, testing its applicability for dynamic, fast-moving targets as well as fluorescent samples. All the basic concepts and physics behind are given in this thesis. The fundamentals like Image properties and transformations (Fourier Transform, Wavelet Transform), Compressed Sensing, Optimization Algorithms, Image reconstruction parameters are explained in this thesis. Idea about other parameters like Optical setup (Optical shearing, Optical masking, Camera Integration, Resolution, Mask optimization) is also there in the thesis.

High-speed imaging has emerged as a crucial tool across disciplines such as physics, chemistry, biology, and engineering. Historically rooted in the need to capture transient phenomena beyond human perceptibility, the field has evolved from mechanical streak cameras to advanced ultrafast digital imaging platforms. While early high-speed photography focused on increasing spatial resolution, modern applications increasingly prioritize temporal resolution to capture dynamic events unfolding on microsecond, nanosecond, or even femtosecond timescales.

Recent advancements have introduced imaging systems capable of recording tens of millions of frames per second (fps), essential for studying turbulence, rapid biochemical reactions, and ultrafast light-matter interactions. However, the cost, complexity, and memory requirements of such systems pose significant limitations. The work under review proposes a cost-effective alternative using compressed sensing principles—Compressed Optical Shearing Ultrafast

Photography (COSUP)—offering a middle ground between speed, resolution, and affordability.

Foundations of Compressed Sensing and Optical Streaking

The theoretical backbone of COSUP lies in compressed sensing (CS), a mathematical framework that allows signal reconstruction from a seemingly insufficient number of samples. CS leverages the inherent sparsity of signals in certain transform domains—commonly the Fourier or wavelet domains—to reconstruct the original signal using optimization techniques. This principle contradicts the classical Nyquist-Shannon sampling theorem, which states that accurate signal reconstruction requires sampling at twice the highest frequency present.

In the context of imaging, compressed sensing allows for the reconstruction of high-resolution images or videos from under-sampled data, thereby reducing the burden on data acquisition hardware and memory. A critical component of this process is the application of mathematical transforms that concentrate signal energy into a few significant coefficients. Fourier and wavelet transforms have thus become standard tools in compressed imaging systems, each suited to different types of signals. Wavelets offer the advantage of simultaneous spatial and frequency localization, which is especially useful for transient and non-periodic phenomena.

COSUP also incorporates optical shearing, a method of encoding temporal information into spatial dimensions by redirecting the incident light beam via a time-dependent optical element such as a rotating mirror or galvanometric scanner. This enables the recording of fast transient events as spatial displacements on a sensor, which can later be deconvoluted into individual frames using CS algorithms.

Historical Developments in Ultrafast Imaging Technologies

The evolution of ultrafast imaging technologies has been marked by a series of pivotal innovations. One of the most influential methods is Compressed Ultrafast Photography (CUP), which integrates streak camera functionality with compressed sensing algorithms to achieve frame rates exceeding 100 billion fps.

CUP represents a leap forward in capturing dynamic 2D scenes, although its reliance on costly and delicate streak cameras limits accessibility.

Alternative techniques have emerged to address CUP's limitations. Single-pixel cameras, for instance, utilize digital micromirror devices (DMDs) and single detectors to reconstruct images through CS, offering lower resolution but significantly reducing hardware costs. Similarly, Frequency Recognition Algorithm for Multiple Exposures (FRAME) and Light in Flight Holography represent active detection techniques, requiring tailored light sources but achieving exceptional temporal resolution.

COSUP distinguishes itself by adopting a passive detection strategy using a galvanometer scanner in place of the streak camera. Although this substitution results in lower temporal resolution—typically around 1.5 million fps—it dramatically lowers system cost and complexity. Moreover, COSUP can be integrated with other imaging modalities such as multispectral or fluorescence imaging, further broadening its application range.

Computational Techniques and Reconstruction Algorithms

The reconstruction phase in COSUP employs the Two-Step Iterative Shrinkage/Thresholding Algorithm (TwIST), which is particularly well-suited for solving underdetermined linear systems with sparse constraints. TwIST provides improved convergence and stability over basic iterative shrinkage methods, especially in the presence of ill-conditioned system matrices or noisy data. The algorithm minimizes a composite cost function balancing data fidelity and sparsity-promoting regularization, typically using the $\ell 1$-norm as a proxy for sparsity.

In practical implementation, TwIST enables the recovery of frame sequences from a single sheared image and a corresponding sampling mask. Despite limitations in spatial fidelity for complex or fine-detailed targets, the algorithm is robust enough to preserve the main structural components of the scene, as confirmed by both qualitative inspection and quantitative SSIM analysis.

Challenges and Future Directions

While COSUP represents a promising step toward democratizing ultrafast imaging, several challenges remain. Chief among them is the trade-off between spatial resolution and sequence depth—enhancing one often compromises the other. Additionally, current reconstruction methods struggle with multiple temporally separated pulses or objects with fine peripheral features, indicating a need for more adaptive or object-aware reconstruction algorithms.

Another challenge is the generalization of mask design. While some empirical findings are provided, the optimal mask configuration likely depends on the characteristics of the target scene. A potential solution lies in integrating machine learning techniques to adaptively generate or select masks based on scene context.

Future work may also explore the integration of COSUP with hyperspectral imaging, digital holography, or 3D reconstruction frameworks. Given its affordability and flexibility, COSUP could play a pivotal role in applications where ultrafast imaging is desired but cost constraints preclude traditional methods, including resource-limited biomedical research, educational laboratories, and field diagnostics.

2.Based on the tutorial titled "Tutorial on Compressed Ultrafast Photography" by Lai, Marquez, and Liang, the literature review can be synthesized and summarized as follows:

1. Emergence of Ultrafast Imaging Techniques

The exploration of ultrafast phenomena, which unfold over femtosecond to microsecond timescales, is crucial in understanding biological, chemical, and physical processes. Traditional imaging methods, like pump-probe techniques, require repeated measurements and extensive scanning. However, many transient phenomena—such as spontaneous neural activity or light scattering in biological tissues—are nonrepeatable or difficult to reproduce, making these conventional approaches impractical.

To address this challenge, single-shot ultrafast optical imaging has emerged. These methods capture entire dynamic scenes within a single exposure, avoiding the need for repetitive experiments. Ultrafast imaging techniques fall into two main categories:

- Active-illumination techniques: Use short optical probe pulses and encode temporal information into spectral or spatial features.
- Passive-detection techniques: Rely solely on capturing photons emitted or scattered from dynamic scenes without requiring external light modulation.

While active methods offer femtosecond resolution and high sensitivity, they fail to image self-luminescent scenes (e.g., photoluminescence or plasma emission). Conversely, passive methods can overcome this but are often limited by the slower response of electronic components compared to optical ones.

2. Compressed Ultrafast Photography (CUP)

Introduced in 2014 by Dr. Lihong V. Wang's lab, CUP is a game-changing single-shot ultrafast imaging method that synergizes compressed sensing (CS) and streak imaging. Unlike conventional streak cameras that suffer from limited spatial information, CUP leverages the sparsity of the target scene to enable spatiotemporal data acquisition and computational image reconstruction.

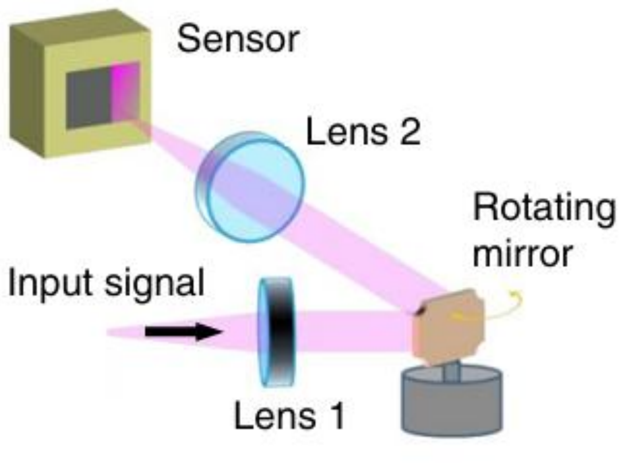


Fig 2.1: Ultrafast imaging set-up

The CUP system comprises:

- Spatial encoding (via binary masks),
- Temporal shearing (via deflection of photoelectrons),
- Spatiotemporal integration (using a 2D sensor),
- Followed by image reconstruction using advanced algorithms.

This approach enables CUP to surpass limitations of 1D high-speed sensors and traditional CCDs. It maintains temporal continuity in recordings, supports both active and passive modalities, and allows for large frame depths. Its compatibility with scientific-grade CCD/CMOS sensors further enhances its practicality.

Fig 2.2: Pseudo patterns

3. Technical Advancements and Variants

CUP has evolved rapidly, with enhancements in both hardware and reconstruction algorithms:

- Multi-view CUP systems improve reconstruction fidelity by capturing both time-sheared and time-unsheared views.
- Hardware flexibility: Spatial encoding can be achieved using DMDs, LC-SLMs, printed masks, or photolithographic techniques.
- High light throughput: CUP collects spatiotemporal information in a single exposure, unlike point or line scanning techniques.

The forward model of CUP mathematically describes how a 3D scene (x, y, t) is projected into 2D snapshots. Image reconstruction solves an inverse problem using optimization frameworks like:

- Two-step iterative shrinkage/thresholding (TwIST),
- Total Variation (TV) regularization,
- Alternating Direction Method of Multipliers (ADMM),
- Plug-and-Play (PnP) ADMM, which allows integrating off-the-shelf denoisers.

Moreover, deep learning has significantly advanced CUP reconstruction:

- Models such as D-HAN (Deep High-dimensional Adaptive Network) integrate physical modelling with convolutional neural networks (CNNs),
- These networks can be trained end-to-end and achieve high-quality video reconstruction from compressed measurements,
- They also allow for data-driven encoding mask optimization.

3. Literature review based on the thesis titled "Single-shot real-time compressed ultrahigh-speed imaging enabled by a snapshot-to-video autoencoder (SMART-COSUP)."

High-speed imaging plays a critical role in capturing transient physical phenomena that occur at micro- to nanosecond timescales. Applications span from ultrafast biological dynamics to explosive chemical reactions and laser-material interactions. Traditional high-speed imaging systems rely on hardware-based solutions such as streak cameras or ultra-high frame rate CMOS sensors. However, these systems face trade-offs in resolution, cost, and data throughput. To address these limitations, recent research has shifted toward computational imaging methods, particularly those based on compressed sensing (CS) and deep learning.

Background: Compressed Ultrafast Photography

Compressed Ultrafast Photography (CUP), first proposed by Gao et al. in 2014, combines a coded aperture, temporal shearing, and compressed sensing to capture dynamic scenes in a single 2D snapshot. CUP uses a streak camera to encode the temporal dimension spatially, achieving frame rates exceeding 100 billion fps. However, the high cost and limited accessibility of streak cameras limit CUP's widespread adoption.

To overcome this, Compressed Optical-Streaking Ultrafast Photography (COSUP) was developed. COSUP adapts the CUP concept to more accessible hardware, such as CMOS cameras, by employing spatial encoding and temporal shearing via a galvanometer scanner. Despite its simplicity and affordability, COSUP faces two significant challenges:

1. Long reconstruction times due to iterative algorithms (e.g., TwIST, ADMM).
2. Variable reconstruction quality highly dependent on sparsity assumptions and system calibration.

3. Traditional Reconstruction Algorithms

3.1 Analytical-Modelling-Based Techniques

These techniques rely on mathematical models of the imaging system and prior knowledge of the signal structure. Commonly used algorithms include:

- TwIST (Two-step Iterative Shrinkage/Thresholding) – Solves the inverse problem with sparsity priors.
- ADMM (Alternating Direction Method of Multipliers) – Decomposes the problem for parallel computation.
- PnP-ADMM with BM3D Denoising – Enhances ADMM with plug-and-play denoising.

Despite their mathematical elegance, these methods:

- Require many iterations (tens to hundreds).
- Are not suitable for real-time applications ($\geq$16 Hz).

- Need careful tuning of hyperparameters and prior selection.

3.2 Machine-Learning-Based Techniques

Recent efforts utilize data-driven models to learn mappings from snapshots to videos:

- Multilayer Perceptrons (MLPs) – High parameter count, poor scalability.
- U-Net – Better spatial feature extraction but suffers from:
    - Temporal incoherence.
    - Dimensional mismatches requiring pseudo-inverse operations.
    - Spatial patching, which breaks scene continuity.

These methods demonstrate faster inference but often lack generalization and reconstruction fidelity in complex dynamic scenes.

4. Advancements: Snapshot-to-Video Autoencoder (S2V-AE)

To address the limitations of both analytic and early deep learning models, the authors propose a Snapshot-to-Video Autoencoder (S2V-AE). This deep neural network learns to directly reconstruct a temporal data cube (x,y,t)(x, y, t)(x,y,t) from a single 2D snapshot (x,y)(x, y)(x,y).

4.1 Architecture

Figure 1: Schematic of S2V-AE

- Encoder: 5 convolutional layers → Bi-directional LSTM → Fully connected layers → Temporal latent vectors.
- Generator: 7 transposed convolutional layers → Each latent vector generates one frame.

This architecture separates spatial and temporal learning tasks:

- The encoder captures temporal dynamics from spatial patterns.
- The generator ensures spatial consistency via GAN-based training.

4.2 Training Strategy

The S2V-AE training is twofold:

1. Train the generator with GANs using multiple discriminators for diverse frame synthesis.
2. Train the encoder to match real data using MSE loss against frames generated by the fixed generator.

This staged training:

- Avoids mode collapse (via multiple discriminators with random projections).
- Ensures temporal coherence (via Bi-LSTM)

Chapter 3: Methodology

1. Image Compression:

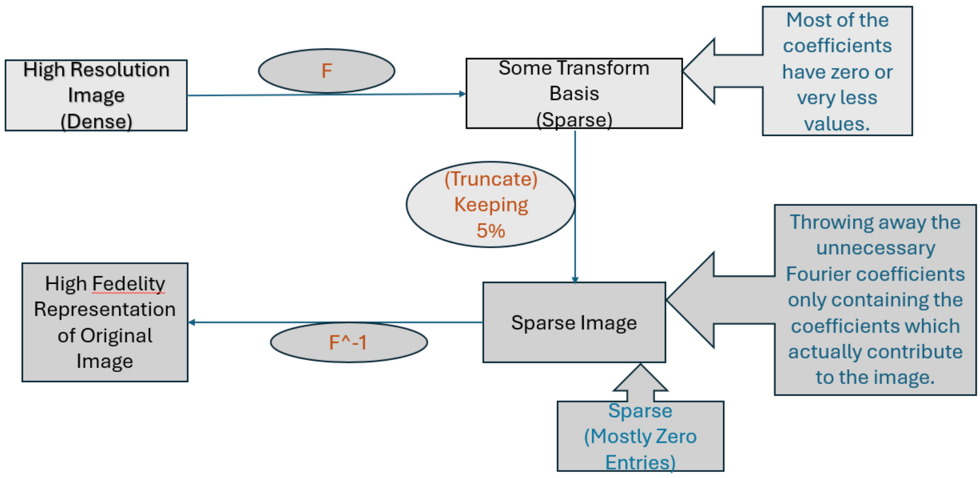


Fig 3.1: Flowchart of Image Compression

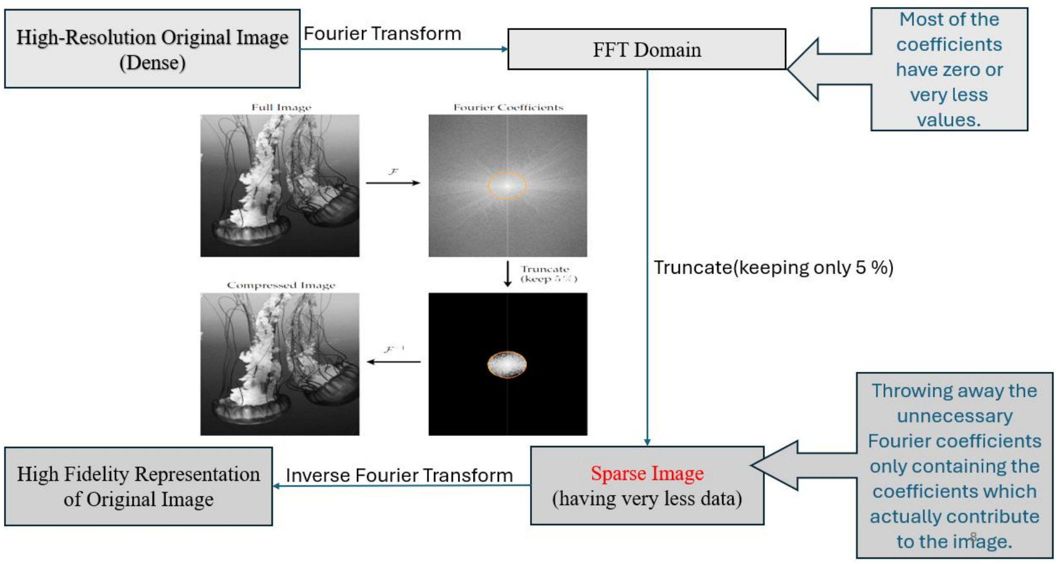


Fig 3.2: Physics of Image Compression

In many real-world applications, especially in signal and image processing, efficient representation and compression of data is critical. One powerful approach involves leveraging the sparsity of signals in the frequency domain, particularly through the Fourier Transform. The illustration above captures a simplified pipeline for compressing and reconstructing an image using the principles of Fourier domain sparsity.

Step 1: ***Transformation to Frequency Domain***

The process begins with a high-resolution, densely sampled original image. This image is typically rich in visual detail, making it data-heavy. To reduce its storage and computational footprint, the image is transformed from the spatial domain into the frequency domain using the Fourier Transform. This transformation represents the image in terms of its sinusoidal components—essentially decomposing it into various frequency contributions.

Step 2: ***Observation of Sparsity***

Upon transformation, the image is now represented as a matrix of Fourier coefficients, which quantify the amplitude of specific frequency components present in the image. Interestingly, in most natural images, the energy of the signal is concentrated in a small number of low-frequency components, while the majority of the higher-frequency coefficients are either very small or negligible. This phenomenon is known as sparsity—the signal has very few significant components in the transformed domain.

Step 3: ***Truncation and Data Reduction***

Recognizing this sparsity allows for a significant reduction in data. The image can be compressed by truncating the Fourier coefficient matrix—retaining only the central region which contains the most significant 5% of the coefficients. This region corresponds to the low-frequency components which hold most of the visual information in the image. The remaining 95%, which mostly contains noise or insignificant detail, is discarded.

This reduced dataset is referred to as the sparse representation of the image. Despite keeping only a small portion of the coefficients, this sparse image still retains the core information needed to reconstruct a recognizable version of the original.

Step 4: ***Reconstruction with Inverse Fourier Transform***

To visualize or use the image again in the spatial domain, an Inverse Fourier Transform is applied to the sparse frequency data. This results in a compressed version of the original image. Though it may be slightly degraded due to the data loss from truncation, it still provides a high-fidelity representation that is visually very close to the original.

This process effectively filters out redundant or unnecessary data, preserving only the most informative components. The result is a much lighter version of the original image, suitable for storage or transmission, with minimal impact on quality.

3.2 Why Images are compressible?

- Let we have a 20*20 B&W Image.
- Total combination possible= 2^ (20*20) =2^400
- Total possible images=2^400
- Total number of nucleons in the known universe=10^80
- Here, (2^400)> (10^80), means bigger than the universe.
- Now think of a Megapixel Image of millions of coloured pixels and having the choice of brightness and saturation of every pixel.

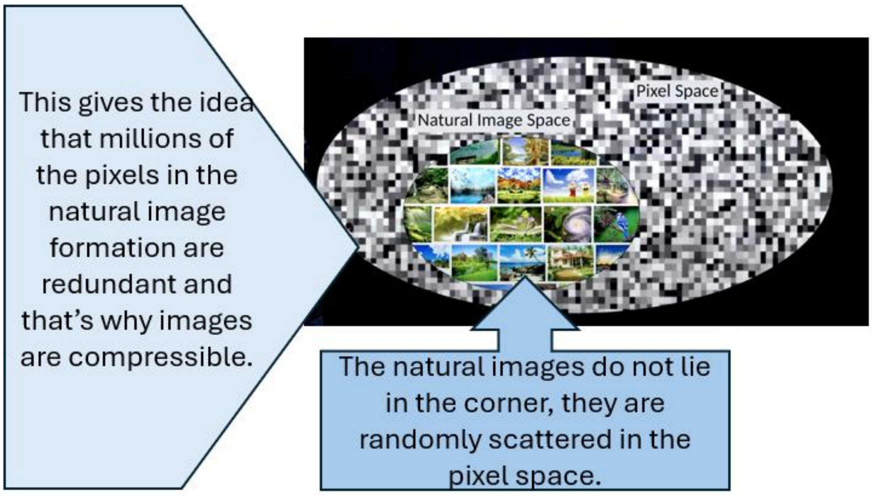


Fig 3.3: Pixel Space

Natural image space consists of all types of images that we can experience in our whole life from each camera angle. It can be anything, a pen, a coffee cup, a human face, anything. It is a very huge space.

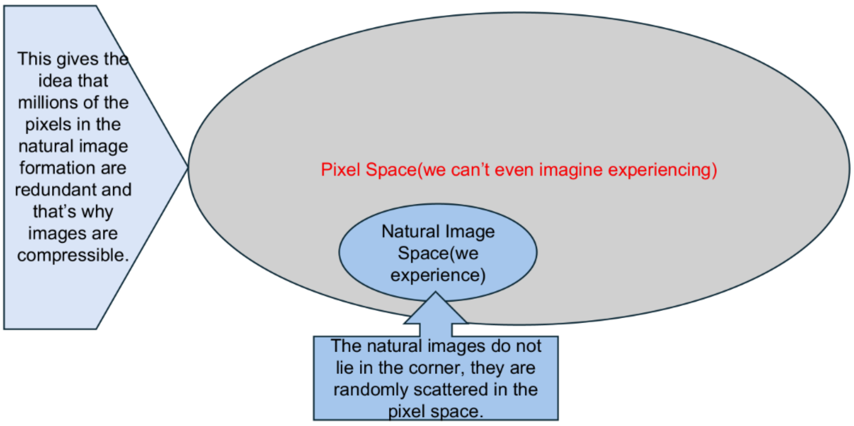


Fig 3.4: Natural Image Space

Natura image space live inside of pixel space. Natural images occupy a miniscule, tiny fraction of a tiny corner of this possible pixel space.

This gives the idea that millions of pixels in the natural image formation are redundant and that's why images are compressible.

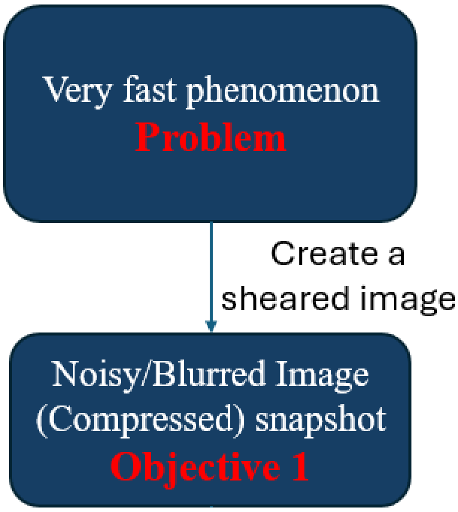


Fig 3.5: Flowchart showing image compression.

1. Introduction to Image Complexity

Images, at a fundamental level, are arrays of pixel values. Even a small black-and-white image with limited resolution holds an astronomical number of possible combinations. Consider a simple 20×20 black-and-white (binary) image. Each pixel has two possible states (black or white), resulting in:

$$\text{Total combinations} = 2^{(20\times 20)} = 2^{400}$$

This number 2 raised to the 400th power is staggeringly large. For perspective, the estimated number of nucleons (protons and neutrons) in the entire known universe is about 10^ {80}, a figure vastly smaller than 2^{400}. This comparison illustrates how large the theoretical space of all possible images is, even for small resolutions.

However, despite this theoretical immensity, only a tiny fraction of these possible images actually occurs in the natural world. This observation leads to one of the most profound insights in image processing: real-world images are not randomly distributed throughout this enormous pixel space—they are highly structured and sparse in nature.

2. The Concept of Natural Image Space

The total set of all possible images forms what we can call the Pixel Space. This space encompasses every potential arrangement of pixel values for a given resolution and color depth. But in reality, natural images—those we capture with cameras or observe with the naked eye—occupy only a very small subset of this massive pixel space.

The graphic representation in the image above captures this idea effectively. The pixel space is depicted as a large area containing every possible image configuration. Within this space lies a concentrated and irregular region: the Natural Image Space. This region is not in the corners or aligned neatly; rather, it is randomly and sparsely distributed across the pixel space. This signifies that natural images do not follow uniform randomness but instead emerge from complex, structured generative processes (such as physical constraints, lighting, perspective, and object coherence).

---

3. Redundancy in Natural Images

Most natural images contain large areas with similar or predictable pixel values. This spatial correlation between neighbouring pixels implies that not all pixel values carry unique information. For instance, in a photograph of the sky, large portions of the image may have nearly the same shade of blue. Likewise, edges, gradients, and textures follow regular patterns that can be approximated by mathematical models or learned features.

This redundancy is a core reason why images are compressible. The apparent complexity (e.g., megapixels of data) is deceptive, because much of this data is repetitive or predictable. It is not the pixel values themselves that matter, but the patterns and relationships between them.

In terms of information theory, the entropy of natural images is much lower than that of random images. Random images would explore a wider variety of pixel configurations with no correlation, resulting in high entropy. But natural images follow specific distributions, reducing the actual information content per pixel.

---

4. Compressibility and its Implications

Because natural images reside on a low-dimensional manifold within the high-dimensional pixel space, they can be encoded with fewer bits without significant loss of visual quality. This is the theoretical basis for image compression algorithms like JPEG, PNG, and modern deep learning-based codecs.

These algorithms exploit redundancy by:

- Transforming the image into a more compact domain (e.g., Discrete Cosine Transform in JPEG)
- Quantizing or removing less significant components
- Using statistical models to encode the remaining data efficiently

The insight from the figure is crucial: we do not need to encode all pixel configurations—only those that lie within the natural image manifold. The rest of the pixel space is effectively irrelevant for practical applications.

This has profound implications not only for compression but also for image generation, super-resolution, denoising, and inpainting. By learning the manifold of natural images using data-driven approaches (like autoencoders or GANs), we can recover or generate plausible images even with limited information.

---

5. From Pixel Space to Learning-Based Representations

Modern machine learning techniques, especially deep neural networks, attempt to learn the structure of natural images in latent space. Autoencoders, for example, map high-dimensional image data into a lower-dimensional latent representation and reconstruct the original image from this compressed code.

This approach assumes that the set of natural images is manifold-structured—meaning it can be embedded in a lower-dimensional space without significant loss of information. As such, tasks like image classification or reconstruction are not conducted across the entire pixel space, but over this reduced and meaningful subspace.

The understanding that natural images are not uniformly distributed—but clustered and sparse within pixel space—has been foundational in the development of:

- Compressed sensing: acquiring fewer measurements than traditional sampling methods, then reconstructing signals/images using prior knowledge about their sparsity.
- Image priors: constraints or learned patterns about what valid images should look like.
- Generative modelling: producing realistic images from compact latent codes (as in GANs or VAEs).

3.3 What is Sparsity? (Help in storing less data):

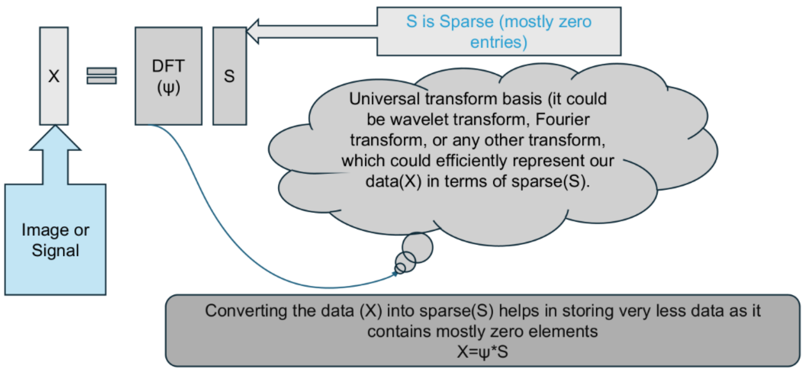


Fig 3.6: Sparsity Vector

In the domain of signal and image processing, efficient data representation is a fundamental goal. Natural signals and images, though seemingly complex, often contain patterns and redundancies that can be exploited for more compact representation. One of the most powerful concepts enabling this compact representation is sparsity, especially when leveraged through appropriate transform domains such as Fourier or wavelet transforms.

Transform-Domain Sparsity

Any signal or image, denoted by X, can often be represented in another domain using a mathematical transform. This operation aims to uncover a latent structure within the data—specifically, a domain in which most of the information can be captured by only a few non-zero coefficients. Mathematically, this is expressed as:

$$X=\Psi\cdot S$$

Here:

- X is the original signal or image (usually in spatial or time domain),
- Ψ is the transform basis (e.g., Discrete Fourier Transform, Wavelet Transform),
- S is the sparse representation of the signal in the transform domain.

In this representation, the majority of the elements in S are zero or negligibly small. This sparsity is what allows us to store or transmit the signal more efficiently. The transformation reorganizes the signal's energy into fewer significant coefficients, while the rest can be discarded or highly compressed without significant loss of information.

Role of Universal Transform Bases

A universal transform basis is a pre-defined mathematical framework that enables signals to be decomposed into simpler building blocks. Common choices include:

- Fourier Transform: Captures global frequency components, ideal for periodic or smooth signals.

- Wavelet Transform: Captures both frequency and spatial information, suitable for analysing transient or localized features like edges in images.
- Cosine Transform (used in JPEG): Efficient for image compression by capturing low-frequency components that dominate visual perception.

The essence of choosing an appropriate transform lies in how efficiently it compresses the information from X into a sparse S. For many natural signals, a small subset of basic functions in these transforms is sufficient to approximate the original signal to a high degree of accuracy.

Advantages of Sparse Representations

The sparse model provides multiple benefits in practice:

1. Data Compression: Storing only the significant non-zero coefficients in S drastically reduces storage requirements.
2. Efficient Processing: Algorithms operating on sparse data are typically faster, as fewer elements are involved in computation.
3. Noise Robustness: Sparse representations help in denoising, as random noise tends not to align well with the sparse structure.
4. Compressed Sensing: In applications where data acquisition is expensive or slow (e.g., medical imaging, ultrafast photography), sparsity enables the recovery of full signals from a few measurements.

The illustration emphasizes that sparsity is a property not of the raw signal X but of its representation in the transform domain Ψ. Once a sparse representation S is obtained, the original data can be reconstructed by applying the inverse transform.

3.4 Compressed Sensing:

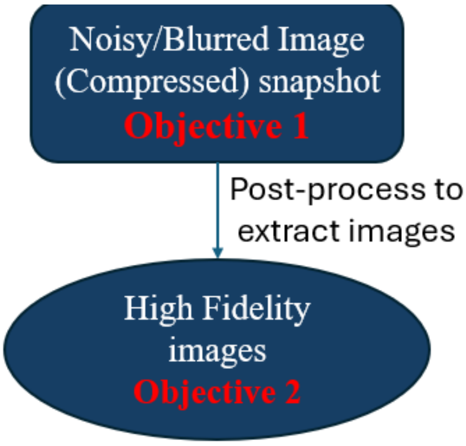


Fig 3.7**:** Flow chart showing compressed sensing.

- Earlier what we do is:

We take a high-dimensional image□ Fourier transforms it so that most of the coefficients become zero□ we truncate it, so that we get a sparse vector that mostly contains the zero coefficients□ Now when we take inverse Fourier transform we get the exact image as before with essential features restored.

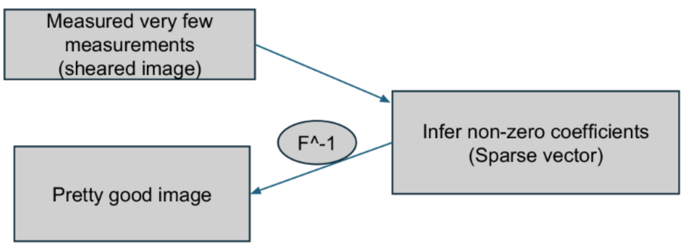
- Now the question arises:

1. Why should I take a high-dimensional image as after doing the above procedure I only left with only 5 % of the information and with the help of that I get the actual image itself?

2. Can't I infer the non-zero coefficients of the image which after being inverse Fourier transform gives me the actual image?

Compressed Sensing:

The real idea behind compressed sensing is that, we are going to take very less measurements instead of high-resolution measurements, using those measurements we are trying to infer non-zero coefficients(i.e, sparse vector) using that sparse vector after taking inverse Fourier transform we will get the actual image which is pretty good.



In traditional signal acquisition systems, capturing high-resolution images or signals typically requires many measurements. These measurements must satisfy the Nyquist-Shannon sampling theorem, which dictates that the sampling rate must be at least twice the highest frequency present in the signal to accurately reconstruct it. However, this results in substantial data volumes, leading to increased costs in terms of storage, processing time, and transmission bandwidth. Compressed sensing (CS) challenges this conventional paradigm by suggesting that we can recover signals and images from significantly fewer samples than previously believed necessary—provided the signal has certain properties like sparsity.

The diagram provided encapsulates the core idea of compressed sensing: acquiring fewer measurements and reconstructing the original signal through inference of sparse coefficients. Instead of recording a high-dimensional signal in full, CS captures a few linear projections and uses mathematical techniques to reconstruct the original data.

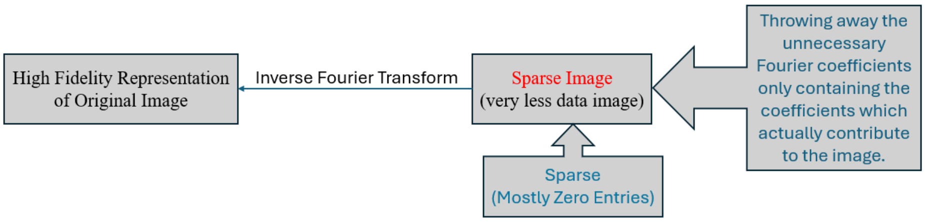


Fig 3.8: Compressed Sensing

Principle of Sparsity

At the heart of compressed sensing lies the assumption of sparsity. Sparsity refers to the idea that many signals or images, though they may appear complex in the spatial or time domain, are simple in some transform domain. For instance, a natural image might appear intricate in pixel space but could have only a few non-zero coefficients in the Fourier or wavelet domain. In mathematical terms, if a signal X of length N can be represented in a transform basis Ψ as:

$X=\Psi.S$

where S is a sparse vector (i.e., most of its entries are zero or nearly zero), then X is said to be sparse in the basis Ψ. The compressed sensing framework exploits this sparsity to recover X from far fewer measurements than N.

Measurement and Inference

The traditional approach involves measuring all entries of the signal and then applying compression techniques to remove redundant information. Compressed sensing flips this model by measuring the signal in a compressed form to begin with. In the diagram, the "sheared image" refers to this compressed measurement. Rather than acquiring the full-resolution image, a few measurements are taken directly in a transformed or encoded format.

These limited measurements are represented as:

$Y = C.X = C.\Psi.S$

Here, C is a sensing matrix that projects the original signal onto a lower-dimensional space. Because we do not directly observe s, our task becomes one of inferring it. This is an underdetermined system—fewer equations than unknowns—but thanks to the sparsity of S, we can use optimization techniques such as L1-norm minimization (Basis Pursuit) to accurately recover S.

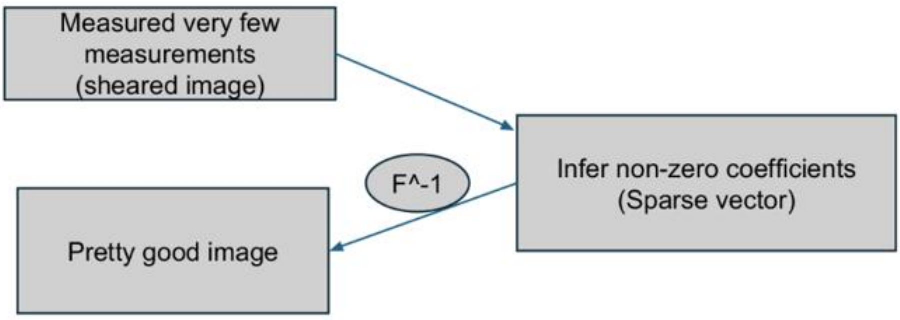


Fig 3.9: Principle of Sparsity

Signal Reconstruction

Once we estimate the sparse vector S, we apply an inverse transformation (e.g., inverse Fourier transform as denoted by F^ {-1} in the diagram) to recover the original signal:

$X = \Psi . S$

Surprisingly, this reconstruction is often nearly identical to what we would have obtained using full-resolution data. This capability opens the door to efficient data acquisition systems that are faster, cheaper, and more scalable.

The final step shown in the diagram "Pretty good image" emphasizes that while we may not always achieve perfect fidelity, the reconstructed image retains most of the essential information. For many applications, such as medical imaging, astronomy, or high-speed photography, this level of reconstruction is more than adequate.

Compressed Sensing Mathematical formula:

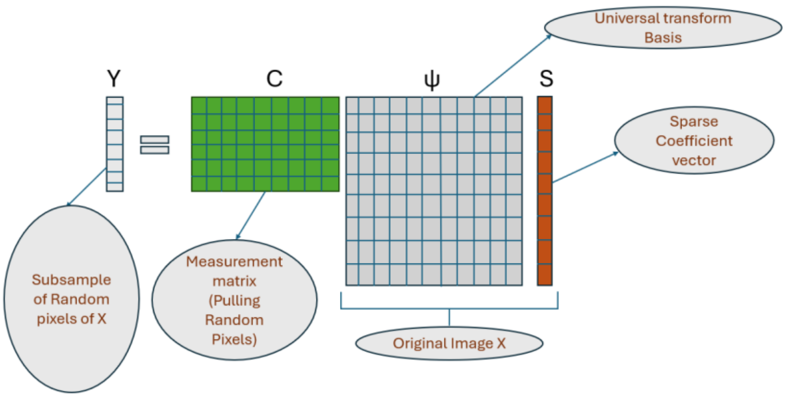


Fig 3.10: CS Mathematical Formula

Compressed Sensing (CS) is a powerful framework that leverages sparsity in signals to reconstruct them from a reduced number of measurements, significantly below the Nyquist rate. This concept is particularly useful in applications like image acquisition, where capturing every pixel may be expensive or time-consuming. The diagram provided illustrates the architecture of compressed sensing applied to imaging, specifically how sparse signal reconstruction is achieved through a combination of sampling, transformation, and optimization.

1. Subsampling and Measurement Matrix

The original image, denoted as X, is assumed to be high-dimensional, such as a full-resolution image. Rather than capturing all the pixel values, compressed sensing techniques operate on a subsampled version of the image. This subsampling is depicted as vector Y, which represents a collection of randomly selected pixels from the image X.

To achieve this, a measurement matrix C is employed. This matrix essentially acts as a selector that "pulls" or extracts random pixel values from the original image. These selected values form the measurement vector Y. This process allows for a significant reduction in the number of samples required, without sacrificing the ability to reconstruct the original image, provided that certain conditions are met.

2. Universal Transform Basis

While the image X may not be sparse in its raw pixel form, it is often sparse in some transform domain. For instance, many natural images are sparse when transformed into the wavelet or discrete cosine domain. This transformation is achieved through a universal transform basis, denoted by the matrix Ψ. When the image is represented in this domain, it can be expressed as a linear combination of a few significant components, resulting in a sparse coefficient vector S.

The relationship between the original image and its sparse representation is given by:

$$X=\Psi.S$$

Here, Ψ is the transformation matrix, and S contains the sparse coefficients.

3. Measurement Model

Combining the measurement process and the transform domain, the overall model of compressed sensing becomes:

$$Y = C.\Psi.S$$

This equation indicates that the subsampled measurements Y are obtained by first transforming the sparse coefficients S into the image domain using Ψ and then applying the measurement matrix C to extract a subset of pixel values. Since both C and Ψ are known, the objective becomes to recover S from Y, even though the number of observations is significantly fewer than the dimensionality of the image.

Reconstruction of Sparse Signals

The core challenge in compressed sensing is the accurate reconstruction of the original image X, or equivalently, the sparse coefficient vector S, from the limited measurements Y. Since this system is underdetermined (fewer equations than unknowns), traditional linear algebraic methods are insufficient. Instead, are employed, leveraging the assumption that S contains very few non-zero entries.

3.5 Concept Of TwIST Algorithm:

Understanding the Underdetermined Inverse Problem in Compressed Sensing

Compressed sensing introduces a paradigm shift in data acquisition by enabling accurate signal reconstruction from far fewer measurements than traditional methods. A key challenge in this framework is solving what is known as an underdetermined inverse problem. This problem arises because we are trying to recover a high-dimensional signal or image from a limited number of measurements. The mathematical formulation is generally written as:

$$Y = C.\Psi.S$$

Where:

- Y is the measurement vector, containing a limited set of observations (randomly selected pixels in the context of imaging).

- C is the measurement matrix, which defines how and which pixels from the original image are sampled.
- Ψ is the transform basis, such as Fourier or wavelet basis, in which the signal or image is sparse.
- S is the sparse coefficient vector, which we aim to recover.

In this scenario, the goal is to find the sparse vector S, which when transformed back using Ψ, and sampled via matrix C, matches the observed vector Y. The issue is that since Y contains only a small number of entries compared to the size of S, the system is underdetermined meaning there are more unknowns than equations. This naturally leads to an infinite number of potential solutions.

Sparsity and the Quest for the Best Solution

The underdetermined nature of the problem necessitates an additional constraint to isolate a unique solution. This is where the concept of sparsity comes into play. Sparsity assumes that the true signal S contains very few non-zero entries when represented in a proper basis. In other words, most elements in S are zero or negligible, and only a few carry the actual information.

Given this assumption, among the infinite possible vectors SSS that could satisfy the equation Y=C.Ψ.S, we are interested in finding the sparsest one—the vector with the fewest non-zero elements. This idea can be expressed mathematically as an optimization problem:

$$\min \|S\|_0 \quad \text{subject to} \quad Y = C\Psi S$$

Where ‖S‖0 counts the number of non-zero elements in S. However, solving this exact formulation is computationally infeasible as it is a combinatorial problem. As a practical alternative, the problem is relaxed into the convex optimization form:

$$\min \|S\|_1 \quad \text{subject to} \quad Y = C\Psi S$$

Here, $\|S\|_1$ denotes the sum of the absolute values of the elements in S, which can be efficiently solved and often leads to the same result as the original sparse solution under certain conditions.

This approach leads to the identification of the sparsest solution that explains the measurement a solution that likely corresponds to the actual image features or signal characteristics in the transform domain.

Solving the Inverse Problem with TwIST Algorithm

To address the above optimization problem, several algorithms have been developed, among which the TwIST algorithm (Two-step Iterative Shrinkage/Thresholding) is particularly effective. TwIST is designed to handle large-scale, sparse inverse problems efficiently and is well-suited for image processing tasks.

The key idea behind TwIST is to iteratively refine the estimate of the sparse vector S by combining the current and previous estimates in a specific way, improving convergence speed and robustness over traditional methods like ISTA (Iterative Shrinkage-Thresholding Algorithm). TwIST uses the following iterative update rule:

$$S^{(k+1)} = (1-\alpha)S^{(k-1)} + \alpha \cdot \text{Shrink}\left(S^{(k)} + \beta \cdot \nabla f(S^{(k)})\right)$$

Where:

- Shrink is a soft-thresholding function that promotes sparsity.
- ∇f represents the gradient of the data fidelity term.
- α and $\beta$are parameters controlling convergence and stability.

TwIST is advantageous because it not only accelerates the recovery process but also stabilizes the reconstruction in the presence of noise or partial observations. It effectively balances the trade-off between fidelity to the observed measurements Yand the sparsity of the solution S.

In summary, the underdetermined inverse problem in compressed sensing is addressed through a combination of mathematical modeling, sparsity assumptions, and optimization techniques. The TwIST algorithm serves as a practical and powerful tool to recover high-quality reconstructions from minimal and incomplete data, making it a cornerstone in modern signal and image processing.

3.6 Compressed sensing: conditions when it works:

1. 'C' to be incoherent w.r.t 'ψ', rows of 'C' should be orthogonal to columns of 'ψ'.
2. There should be randomness associated with matrix 'C'. Incoherency is generally achieved by randomness.
3. The sparse vector S is said to be K-Sparse if it contains exactly k non-zero entries in it. To find out a Sparse vector S which is K-Sparse we need to take measurements more than K.

No. of measurements(P):

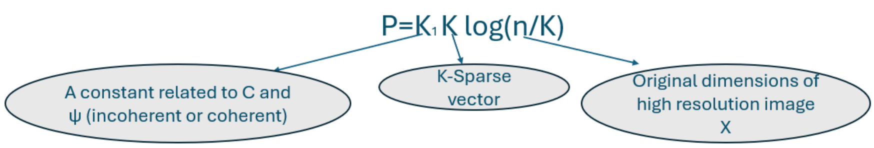


Fig 3.11: Condition of working

In compressed sensing, a key question is: *how many measurements are needed to accurately reconstruct a signal or image from its compressed form?* The formula:

$$P=K1\ K\ \log(n/K)$$

provides a theoretical guideline to answer this. Here, P represents the minimum number of measurements required for accurate recovery. This equation is derived from foundational results in information theory and sparsity-driven signal reconstruction. Let's break down each component and their significance.

- **K**: This denotes the sparsity level of the signal or image in a given basis Ψ. A signal is said to be K-sparse if only Kof its coefficients are significantly non-zero when represented in a proper transform domain (e.g., Fourier, wavelet). The smaller the K, the fewer measurements are needed.
- **n**: This represents the total number of elements or pixels in the high-resolution signal or image X. It reflects the ambient dimension, i.e., the size of the original uncompressed data.
- **log(n/K)**: This term arises from probabilistic bounds on recovering sparse signals. It adjusts the number of required measurements depending on how large the original data is compared to the number of non-zero components. The logarithmic dependence ensures that the number of required samples grows slowly as the resolution increases.
- **K1**: This is a constant that depends on the relationship between the measurement matrix C and the sparsifying basis Ψ. Specifically, it is influenced by how coherent or incoherent these two matrices are. Incoherence is a desirable property—it ensures that the measurements are spread out in a way that captures different aspects of the signal, which helps in accurate reconstruction. A lower coherence results in a smaller K1, thus reducing the number of required samples.

3.7 Restricted Isometric Property in Image Processing:

In the field of image processing, the ability to recover high-quality images from limited, corrupted, or noisy data is of paramount importance. Traditional approaches often require large amounts of data, but modern techniques like compressed sensing have revolutionized this by enabling the recovery of images from significantly fewer measurements. A core theoretical tool that underpins these advances is the Restricted Isometry Property (RIP). Originally introduced within the framework of compressed sensing, RIP provides the mathematical foundation to ensure accurate reconstruction of sparse signals, including natural images, under certain conditions. This document explores the concept of RIP, its mathematical formulation, significance in image processing, and its applications in various image reconstruction tasks.

Compressed sensing (CS) is a technique that enables the acquisition and recovery of sparse signals from far fewer samples than traditionally required. It hinges on two main principles: sparsity and incoherence. RIP is critical because it guarantees that all subsets of columns taken from the sensing matrix behave almost like an orthonormal system. This orthonormal-like behavior is necessary to ensure that different sparse signals map to sufficiently different measurement vectors, thus avoiding ambiguity in reconstruction.

In image processing, RIP has significant implications for tasks such as:

- Image reconstruction: When reconstructing images from under-sampled data (e.g., in MRI), RIP ensures that the transformation matrix retains the critical features of sparse image representations.
- Denoising: In scenarios where images are corrupted by noise, algorithms relying on sparsity and RIP can effectively isolate and remove noise.
- Inpainting: Missing parts of an image can be recovered accurately using RIP-compliant methods, assuming the known parts of the image form a sparse representation.

Several reconstruction algorithms leverage RIP to ensure robust and accurate image recovery. One notable example is the Two-step Iterative Shrinkage/Thresholding (TwIST) algorithm, which is particularly effective for large-scale sparse reconstruction problems. These algorithms generally operate under the assumption that the measurement matrix satisfies the RIP condition, thereby ensuring that the optimization problem they solve has a unique and stable solution.

In practical applications, although checking RIP for a given matrix is computationally infeasible, it is well-established that certain types of random matrices (e.g., Gaussian or Bernoulli) satisfy the RIP with high probability when the number of measurements is sufficient. This insight guides the design of sensing systems in hardware implementations.

3.8 MATLAB code:

CUP Reconstruction:

```
close all;
clear;
clc;

taus = (5:0.05:5);
num = size(taus,2);

for i = 1:num
    tau = taus(i);
    frames = 200;

    PTRN = double(imresize(imread('pattern1.tiff'),0.25));
    pattern = 1;
    if size(PTRN, 3) > 2
        PTRN = sum(PTRN, 3);
    end
    bkg = mean(mean(mean(PTRN(1:10,1:10))));
    PTRN = max(PTRN - bkg, 0);
    PTRN = circshift(PTRN, [+200 +0]);
```

```
    KP = zeros(size(PTRN, 1), size(PTRN, 2), frames);
    for ff = 1:frames
        KP(:,:,ff) = circshift(PTRN, [(ff-1) 0]);
    end
    clear PTRN;
    KP = norm1(KP);

    sim = double(imresize(imread('shearedraccon.tif'), [256, 256]));
    if size(sim, 3) > 2
        sim = sum(sim, 3);
    end
    bg = mean(mean(sim(1:50, 1:50)));
    sim = sim - bg;
    sim(sim < 0) = 0;
    sim = norm1(sim);

    maxiterations = 50;
    piter = 8;
    tolA = 1e-8;
```

```
    A = @(x) Rfuntwist(x, KP);
    AT = @(x) RTfuntwist(x, KP);
    Psi = @(x, th) denoising(x, th, piter);
    Phi = @(x) TVnorm3(x);

    [cube, base, x_debias, obj, times, debias_start, mse_twist] = ...
        TwISTmod(sim, A, tau, 'AT', AT, 'Psi', Psi, 'Phi', Phi, ...
        'Initialization', 2, 'Monotone', 1, ...
        'StopCriterion', 1, 'MaxIterA', maxiterations, ...
        'ToleranceA', tolA, 'Debias', 0, 'Verbose', 1);

    for ii = 1:frames
        im = cube(:,:,ii) - base;
        im(im < 0.01) = 0;
        imagesc(im, [1e-3 0.1]);
        axis('image');
        colormap(hot);
        axis off;
        pause(0.1);
        F(ii) = getframe;
    end
    movie(F, 3, 5);
```

```
    x = zeros(size(cube));
    for ii = 1:frames
        x(:,:,ii) = max(0, cube(:,:,ii) - base);
        x(:,:,ii) = circshift(x(:,:,ii), [-(ii-1) 0]);
    end
    x = max(0, x - 0.0);
    x = x ./ max(x(:));

    str = strcat('processed_file_kushagra_M', num2str(pattern));
    save([str, '.mat'], 'x');
end
```

This MATLAB script performs image reconstruction using the TwIST (Two-step Iterative Shrinkage/Thresholding) algorithm over a range of regularization parameters. It is designed to process a sheared image using a predefined calibration pattern and output both visual results and processed data.

Initialization and Loop Structure

The program initializes a list of regularization values (taus) and loops over each value to test its impact on reconstruction performance. Within each iteration, the algorithm is executed independently using the current tau.

Pattern and Sheared Image Preprocessing

A calibration pattern image (pattern1.tiff) is read, resized, and converted to grayscale if necessary. Background subtraction is performed using the mean of the top-left region, and a vertical shift is applied to align the pattern with the sheared image. A 3D matrix (KP) is generated by vertically shifting the pattern across a sequence of frames to simulate movement.

The sheared image (shearedraccon.tif) is similarly resized, converted to grayscale if needed, and normalized after background subtraction. These preprocessing steps enhance contrast and ensure numerical consistency for the TwIST algorithm.

TwIST Algorithm Configuration

The forward model (A) and its transpose (AT) are defined as function handles using external helper functions. Total variation denoising and regularization functions (Psi and Phi) are also configured. The algorithm is then executed via TwISTmod, which returns a reconstructed 3D image cube, baseline image, and performance metrics.

Visualization and Post-processing

The reconstructed frames are visualized sequentially using MATLAB's image display functions, with colormap adjustments and timing control for animation. Each frame is stored to form a video preview. In the post-processing stage, the baseline is subtracted from each frame, and non-negative normalization is applied. The result is then realigned using reverse vertical shifts.

Finally, the processed 3D dataset is saved to a .mat file with a unique identifier. This format allows future analysis or re-visualization of the denoised and reconstructed image sequence.

CUP Post Processing:

```
clear; clc; close all;
taus = (5:0.05:5);
num = size(taus, 2);
for i = 1:num
    tau = taus(i);
    filename = 'processed_file_kushagra_M';
    pattern = 1;
    str = strcat(filename, num2str(pattern));
    frames = 200;

    [data, map] = importdata(strcat(str, '.mat'));
    img0 = data;
    clear data;
    extI = double(imresize(imread("referenceM.tif"), [256,256]));
    [nR, nC, nF] = size(img0);
    pix = 10; bin = 40; Mag = 0.1; scanRange = 100;
    dt = scanRange / 256;
    dt = (1e-9 / 1e-12) * dt;
    cropBoxX = [1 256];
    cropBoxY = [1 256];
    cropBoxT = [1 frames];
```

```
    videoFrameRate = 30;
    cm_hot = colormap(hot);
    close all;
    cm_hot(1, :) = [0 0 0];

    bkg = mean(mean(mean(img0(1:10, 1:10, :))));
    img0 = max(img0 - bkg, 0);
    img0 = img0 ./ max(img0(:));

    img1 = zeros((cropBoxY(2) - cropBoxY(1) + 1), (cropBoxX(2) - cropBoxX(1) + 1), nF);
    for ff = 1:nF
        img1(:,:,ff) = img0(cropBoxY(1):cropBoxY(2), cropBoxX(1):cropBoxX(2), ff);
    end
    for ff = 1:nF
        img1(:,:,ff) = max(0, img1(:,:,ff) - img1(:,:,nF));
    end
    img1 = img1(:,:,max(1, round(cropBoxT(1))):min(nF, round(cropBoxT(2))));
    nT = round(cropBoxT(2)) - round(cropBoxT(1));
    for ff = 1:nF
        img1(:,:,ff) = flipud(img1(:,:,ff));
    end
```

```
    shiftPerFrame = 1;
    for ff = 1:nF
        img1(:,:,ff) = circshift(img1(:,:,ff), [round((ff-1)*shiftPerFrame) 0]);
    end

    extI = max(extI - mean(mean(extI(1:10, 1:10))), 0);
    extI = circshift(extI, [10 -10]);
    extI = extI(cropBoxY(1):cropBoxY(2), cropBoxX(1):cropBoxX(2));
    extI = extI ./ max(extI(:));
    extI = rot90(extI, 2);

    x = 1:256; y = 1:256;
    [xx, yy] = meshgrid(x, y);
    h = exp(-((xx-128).^2)/(2^2)) .* exp(-((yy-128).^2)/(2^2));
    extI = conv2(extI, h, 'same');

    img2 = img1;
    for ff = 1:(cropBoxT(2) - cropBoxT(1) + 1)
        img2(:,:,ff) = img2(:,:,ff) .* (extI.^1);
    end
```

```
    img2 = img2 ./ max(img2(:));
    extI = max(0, extI - 0.05);
    extI = extI ./ max(extI(:));
    img2 = max(0, img2 - 0.05);
    img2 = img2 ./ max(img2(:));

    sizePerPix = pix * bin / Mag;
    sizePerPix = (1e-6 / 1e-3) * sizePerPix;
    dT = dt * bin;
    [nRR, nCC, nFF] = size(img1);
    X = (-round(nCC/2):nCC-round(nCC/2)-1) .* sizePerPix;
    Y = (-round(nRR/2):nRR-round(nRR/2)-1) .* sizePerPix;
    [XX, YY] = meshgrid(X, Y);
    T = (0:nFF-1) .* dT;

    save('postProcessedData.mat', 'img2', 'extI', 'X', 'Y', 'T');

    f = figure(1);
    imagesc(X, Y, max(0, extI));
    colormap(gray); axis off; axis image; clim([0 1]);
    hold on;
```

```
    quiver(4, -10, 5, 0, 'ShowArrowHead', 'off', 'LineWidth', 2, 'Color', 'white');
    text(4.5, -9, '5 mm1', 'Color', 'white', 'FontSize', 14);
    hold on;
end
```

This MATLAB script performs post-processing on 3D image data loaded from a .mat file, applies spatial and temporal adjustments, and generates visual outputs and a video. The process starts by loading volumetric image data and a reference image, followed by setting essential parameters such as pixel size, binning factor, magnification, cropping ranges, and time increments.

First, the script removes background noise by subtracting the average intensity of a small corner region and normalizes the image intensity. It crops the spatial region of interest based on predefined coordinates and normalizes each frame by subtracting the last frame to reduce baseline effects. The script also flips each frame vertically and applies a frame-dependent vertical shift correction to compensate for misalignments.

Next, an external reference image undergoes normalization, cropping, rotation, and blurring using a Gaussian filter, enhancing its spatial features. This processed reference image is multiplied element-wise with every frame in the main image stack to improve signal contrast, followed by final normalization and background subtraction.

The code recalculates spatial coordinates considering binning and magnification and generates corresponding temporal coordinates based on the adjusted time per frame. The processed image stack, reference image, and coordinate arrays are saved into a .mat file.

For visualization, the script creates a grayscale plot of the reference image with scale annotations and saves it. It then iterates through the first 150 frames of the processed 3D data, displaying each frame with a heat colormap and saving each as a TIFF image. Finally, it compiles these frames into an AVI video file with a specified frame rate, allowing easy review of the temporal sequence.

3.9 DATA COLLECTION:

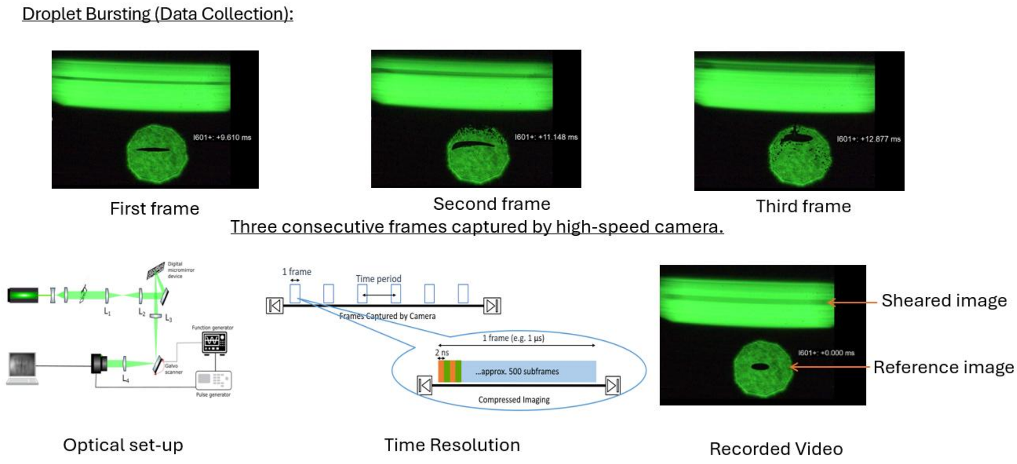


Fig 3.12: Data Collection

The phenomenon of **droplet bursting** represents a rapid physical process with dynamic fluid motion and surface tension interactions that occur over microsecond to nanosecond timescales. Capturing such events requires an advanced imaging setup capable of extremely high temporal resolution. Traditional high-speed cameras are often inadequate due to their frame rate limitations or the sheer volume of data generated in short time intervals.

The provided visual outlines a process where high-speed imaging, aided by compressed ultrafast photography (CUP), is employed to observe and analyze the bursting of a droplet. In this specific case, a high-speed imaging system captures three critical frames in sequence, revealing the evolution of the droplet over a few milliseconds. This process allows for an in-depth study of fluid dynamics, shockwave propagation, and material response.

Sheared and Reference Images in CUP

Two core elements are highlighted in the recorded video section of the image: the sheared image and the reference image.

- Sheared Image: This represents a temporally encoded version of the dynamic scene. The shearing process, typically induced by a fast galvanometric scanner or an equivalent optical system, shifts the image along one spatial axis in direct proportion to time. As a result, each line in the image corresponds to a different temporal moment, effectively compressing a sequence of subframes into a single image. This technique enables visualization of rapid motion in a single snapshot.

- Reference Image: Captured without any temporal modulation, this image provides a static view of the droplet just before or after the event. It serves as a spatial anchor during the reconstruction process, ensuring that the recovered frames align correctly with the physical geometry of the object or scene being studied.

The sheared and reference images work together to enable a full reconstruction of the dynamic process, overcoming the limitations of conventional frame-by-frame capture.

Time Resolution and Compressed Imaging

A key advantage of this system is its ultra-high temporal resolution, achieved through compressed imaging. As visualized in the schematic labeled "Time Resolution," the sheared image captures approximately 500 subframes compressed into one single frame. The actual time resolution can reach the scale of 2 nanoseconds per subframe, which is orders of magnitude faster than standard high-speed cameras.

In the process:

- The DMD modulates the incoming light using a binary pattern.
- The galvo scanner moves the image on the camera sensor during the exposure, mapping temporal changes onto spatial shifts.
- The camera collects this sheared and modulated signal in one shot.

During reconstruction, algorithms like TwIST (Two-Step Iterative Shrinkage Thresholding) are used to recover the temporal sequence from the encoded image. This approach provides an efficient way to study fast processes like droplet bursting with minimal loss of temporal detail.

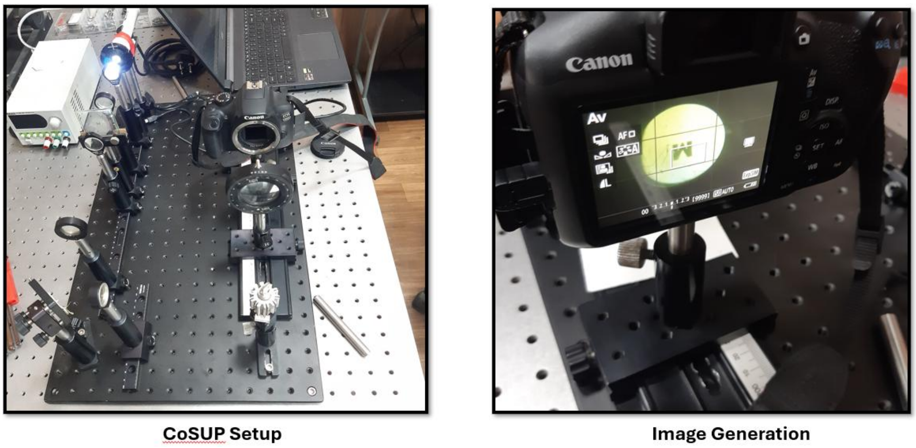


Fig 3.13: Optical Set-up

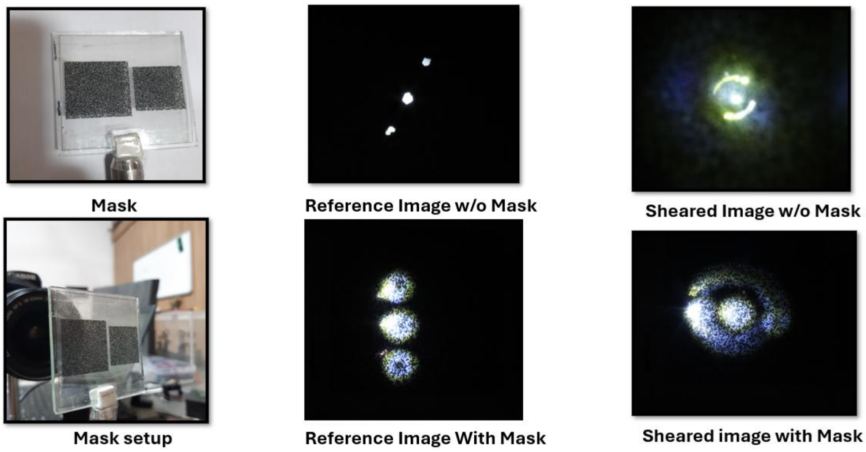


Fig 3.14: Mask & without Mask data collection

This setup is designed to capture ultrafast dynamic events by encoding both spatial and temporal information into a single image. It relies on carefully synchronized optical and electronic components to achieve extremely high time resolution. Here's a step-by-step explanation of how each part works:

1. Laser Source (Illumination)

At the far left of the setup, a laser or light source provides the necessary illumination. This is typically a pulsed laser, offering brief but intense flashes of light, which are ideal for freezing motion during fast dynamic events such as droplet bursting. The wavelength and pulse duration of the laser are chosen based on the experimental needs.

2. Collimating and Focusing Optics (Lenses L1–L4)

- L1 and L2: These lenses form a collimating and focusing system that directs the laser beam through the optical path with minimal divergence. They ensure the beam is appropriately aligned and shaped for interaction with the modulation devices.
- L3: This lens focuses the modulated light onto the camera sensor, but before reaching the sensor, the light passes through other key components.
- L4: Positioned after the galvo scanner, this lens helps collect the sheared image and focuses it precisely onto the camera's sensor.

Each lens is positioned to maintain beam quality and to match the system's focal requirements at different stages of the path.

3. Digital Micromirror Device (DMD)

The DMD plays a crucial role in spatial encoding. It contains an array of microscopic mirrors that can rapidly tilt to modulate incoming light based on a pre-programmed binary pattern. This pattern encodes spatial information onto the light, which is essential for later decoding during image reconstruction.

The DMD helps introduce structured light modulation into the system, which acts like a fingerprint on the image, allowing the post-processing algorithm to distinguish and reconstruct temporal frames from a single sheared image.

4. Galvanometric (Galvo) Scanner

The Galvo scanner consists of a fast-rotating mirror controlled by a voltage signal. As the mirror moves, it introduces a continuous deflection to the light beam. This creates a temporal shear in the captured image—each row (or column) of the image corresponds to a slightly different moment in time.

The galvo scanner is synchronized with the rest of the system to translate the time-domain event into a spatial distortion. This shear encodes temporal progression into a single camera exposure.

5. Synchronization Electronics (Function and Pulse Generators)

These components control the timing of all key devices in the setup:

- The function generator sends signals to the galvo scanner, DMD, and potentially to the laser, ensuring everything operates in precise timing.
- The pulse generator triggers the camera and the laser pulses at exact intervals, enabling perfect synchronization between illumination and capture.

Together, these generators act like the brain of the system, coordinating all activities down to the nanosecond.

6. Camera and Data Acquisition System

The camera, placed at the output end, captures the sheared and modulated light as a single image. This image contains embedded spatial and temporal information. Although the camera itself captures just one frame, the encoded data inside this frame allows for reconstruction of hundreds of subframes using computational algorithms.

A connected computer system collects the image data and processes it using algorithms such as TwIST (Two-Step Iterative Shrinkage/Thresholding). This post-processing step reconstructs the time-resolved video from the compressed data captured by the system.

CHAPTER 4: Results

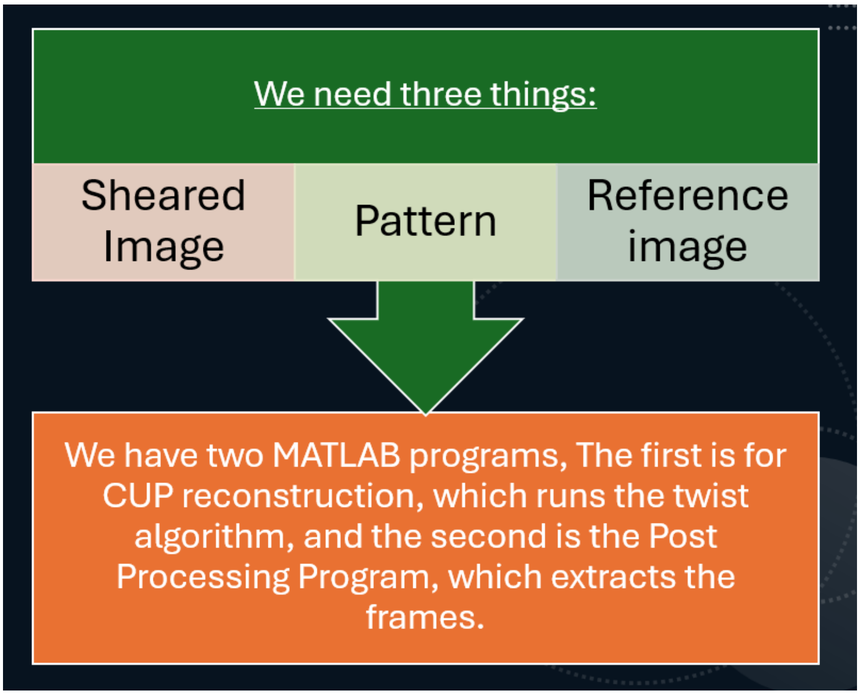


Fig 4.1 Things Needed

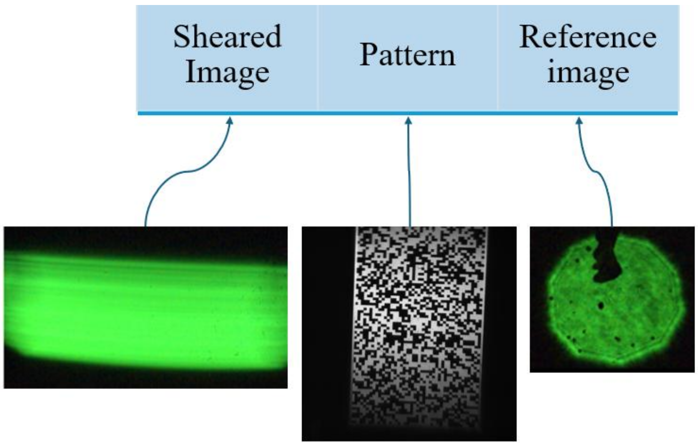


Fig 4.2 Pictorial View

The image outlines three key inputs necessary for CUP reconstruction:

1. Sheared Image – This is the primary data obtained from the CUP camera. It contains a temporally encoded version of the scene, where each row of the image corresponds to a different time slice due to the shearing process applied during capture.

2. Pattern – A known, fixed modulation pattern is imposed on the incoming light signal, typically using a digital micromirror device (DMD) or spatial light modulator (SLM). This pattern is essential for unmixing the temporally encoded data.
3. Reference Image – This static image of the scene, captured without any temporal encoding, provides spatial context. It helps in aligning and calibrating the reconstruction output to the actual physical layout of the scene.

Together, these three inputs form the core dataset required to reconstruct a high-fidelity video from a single CUP frame.

1. CUP Reconstruction Program

This program focuses on recovering the video frames from the sheared image using an iterative computational algorithm. The core method employed is the Two-Step Iterative Shrinkage/Thresholding (TwIST) algorithm, which is known for its efficiency in solving inverse problems such as image reconstruction. TwIST works by iteratively refining an estimate of the true image sequence based on the input sheared image and the known modulation pattern.

The main tasks performed by this program include:

- Preprocessing the sheared image and pattern to prepare them for reconstruction.
- Setting up the measurement matrix based on the pattern and shearing geometry.
- Iteratively applying the TwIST algorithm to solve the inverse problem.
- Outputting a reconstructed video volume where each slice represents a different time step.

2. Post-Processing Program

Once the frames are reconstructed, this second MATLAB program handles their visualization, enhancement, and export. This includes:

- Extracting individual frames from the video volume.
- Aligning the reconstructed video with the reference image for spatial correction.
- Normalizing and enhancing the frames for better visual quality.
- Optionally saving the output as a video or image sequence for analysis and presentation.

4.1 <u>RESULTS FOR DROPLET IMAGE:</u>

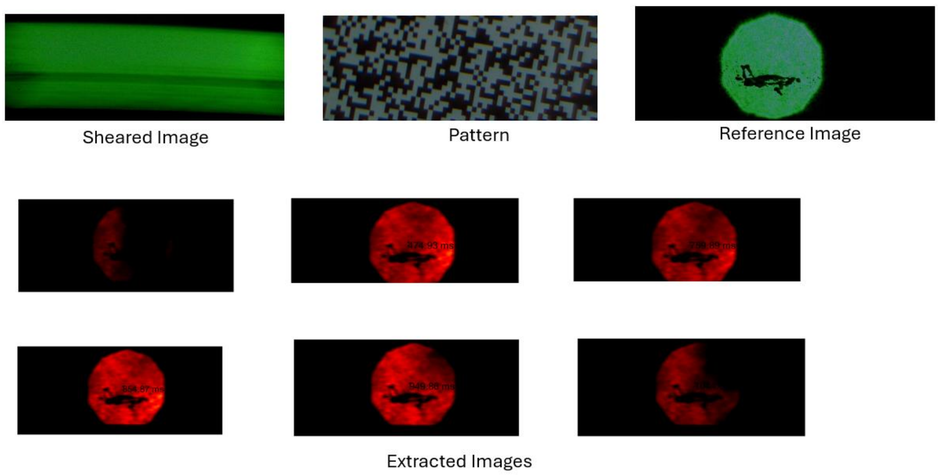


Fig 4.3: Results for Droplet Image

The image illustrates a computational imaging process where a sheared image, combined with a known coded pattern, is used to reconstruct temporal frames from a high-speed event. The top row shows the original sheared image, the binary-coded pattern used for encoding, and a green-tinted reference image containing a clear object (a droplet). The bottom two rows present the extracted frames over time, rendered in red with visible timestamps, showing the progressive movement or appearance of the object. This setup demonstrates how temporal information can be recovered from a spatially sheared input using structured illumination or coding techniques.

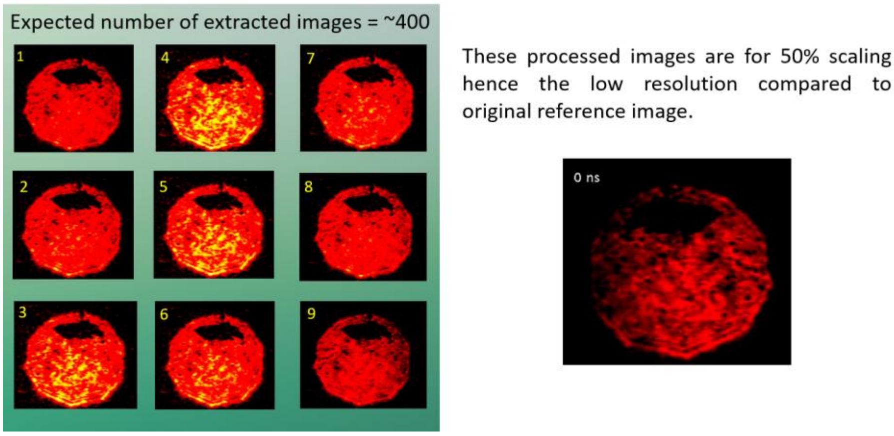


Fig 4.4: Extracted Images of Droplet

4.2 RESULTS FOR RACCON IMAGE:

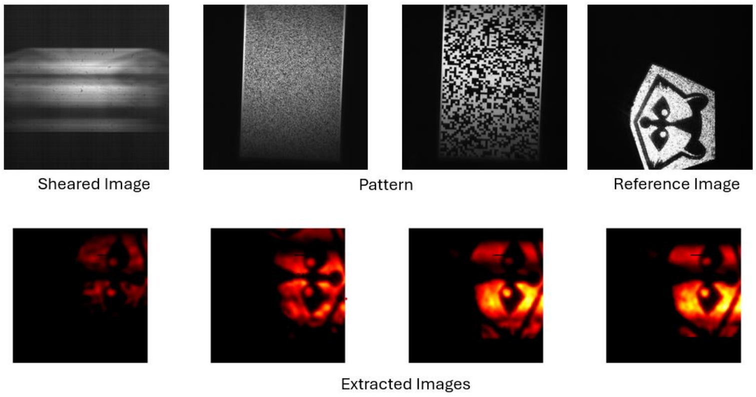


Fig 4.5: Results for Raccoon Image

This image set demonstrates a compressed ultrafast imaging technique where a temporally sheared scene is encoded using a known binary pattern to enable frame-by-frame reconstruction. The top row shows the essential components: a sheared image capturing overlapping temporal slices, a random or pseudo-random pattern used for encoding temporal information, and a reference image for comparison. The bottom row presents the reconstructed frames, where motion and structural changes over time are visualized in red, showing dynamic content unfolding within the encoded exposure. This technique enables high-speed event capture beyond traditional camera frame rates by exploiting spatial-temporal encoding and computational reconstruction.

The bottom row shows a sequence of reconstructed frames that capture the motion of a raccoon image moving downward over time. In each consecutive frame, the raccoon’s face appears slightly lower than in the previous one, illustrating its vertical movement. The gradual shift in position across the frames represents different time instances, effectively showing a time-lapse of the raccoon’s descent. This progression demonstrates how the temporal information encoded in the sheared image has been successfully decoded to reveal the dynamic motion of the subject.

4.3 <u>RESULTS FOR SHAPE M:</u>

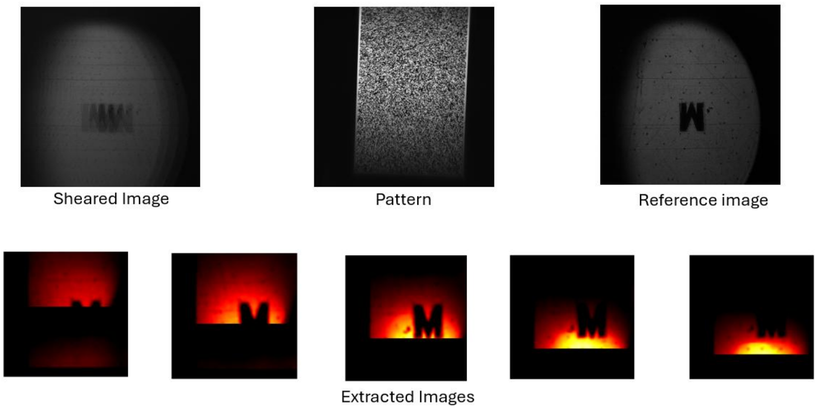


Fig 4.6: Results for Shape M image

The displayed figure presents a sequence from a computational imaging setup used to recover fast temporal events. The top row includes a sheared image, a random encoding pattern, and a reference image showing the letter "M". The sheared image captures multiple temporal instances compressed into a single frame, while the binary pattern is used for temporal encoding. The bottom row reveals the extracted images reconstructed from the sheared input. These frames show the "M" character moving vertically upward over time, becoming clearer and more centered with each step. This illustrates how the encoded temporal information is successfully decoded, enabling visualization of high-speed vertical motion across different time slices.

The sequence of extracted images in the bottom row clearly captures the vertical motion of the letter "M" over time. At the beginning of the sequence, the "M" appears at the lower part of the frame, partially visible and dim, indicating the starting position of the motion. As time progresses through the frames, the "M" moves upward steadily, becoming more centered, distinct, and fully visible. This upward displacement reflects a smooth vertical translation, suggesting the object is moving continuously in the upward direction during the exposure period. The bright red background shifting along with the letter enhances the sense of motion, while the pattern-based decoding ensures the temporal order of this motion is preserved. This set effectively demonstrates how dynamic scenes can be captured and reconstructed from a single sheared image using temporal coding techniques.

CONCLUSION:

This thesis presents a comprehensive study and implementation of Compressed Optical Shearing Ultrafast Photography (COSUP), offering an efficient and cost-effective solution for capturing ultrafast dynamic events. Through theoretical analysis, optical setup design, algorithmic development, and experimental validation, the work successfully demonstrates how a single temporally sheared image, modulated with a known spatial pattern, can be computationally decoded to reconstruct high-speed video sequences. The system effectively captures phenomena like droplet bursting and object motion—traditionally requiring expensive high-speed cameras—using affordable hardware and compressed sensing principles.

The integration of the TwIST algorithm played a crucial role in enabling accurate frame recovery from underdetermined measurements, validating the robustness of sparse reconstruction methods in practical applications. The vertical motion observed in experiments—such as the downward movement of a raccoon image or the upward displacement of the letter “M”—underscores the technique’s temporal fidelity and ability to resolve dynamic motion sequences with clarity.

Moreover, the system offers significant advantages over traditional high-speed imaging: reduced hardware complexity, minimal data storage, and one-shot capture capability for non-repeatable events. These strengths position COSUP as a promising alternative for ultrafast imaging needs in fluid dynamics, combustion research, biomedical diagnostics, and other time-critical domains. The findings lay the groundwork for further research in integrating deep learning for faster reconstruction and exploring applications in 3D or hyperspectral ultrafast imaging, thereby expanding the impact of compressed sensing in scientific visualization

FUTURE SCOPE

The work presented in this thesis lays a strong foundation for further exploration and development in the field of compressed ultrafast imaging, particularly with the COSUP (Compressed Optical Shearing Ultrafast Photography) technique. Several promising directions can enhance both the capabilities and practical deployment of this technology. One major avenue is the integration of deep learning algorithms for real-time reconstruction. Traditional iterative methods like TwIST, while effective, are computationally intensive. Deep neural networks trained on representative datasets can drastically reduce reconstruction time, enabling real-time feedback for live experiments or diagnostics.

Another key area is the optimization of mask patterns used during encoding. Currently, random or pseudo-random binary patterns are used, but adaptive or learning-based mask generation could significantly improve reconstruction quality, especially for scenes with complex dynamics. Research into scene-aware or object-aware encoding could make the system more intelligent and robust under varied experimental conditions.

Hardware-wise, the COSUP system could be extended to support multi-modal imaging, such as combining ultrafast temporal resolution with spectral or depth information, thereby making it suitable for advanced applications like chemical plume analysis, tissue diagnostics, or turbulence visualization in 3D. Furthermore, miniaturization of the setup using meta-surfaces or integrated optics could lead to portable ultrafast cameras suitable for use in field conditions or compact laboratory environments.

Finally, COSUP holds potential in non-traditional domains such as biomedical imaging, materials testing, aerospace diagnostics, and autonomous navigation, where high-speed events must be captured accurately without bulky or expensive hardware. By refining the system for robustness, speed, and integration, this research can significantly contribute to democratizing ultrafast imaging across academia, industry, and even consumer technologies.

REFERENCE:

 Gao, Liang, et al. *Compressed Ultrafast Photography: Capturing the Light Transport in a Scene at Pico-Second Resolution.* Nature, vol. 516, no. 7529, 2014, pp. 74–77.

 Contreras, Constanza Cendon. *Compressed Optical Streaking Ultra High-Speed Photography.* Master's Thesis, University of Rochester, 2017.

 Lai, P., Marquez, A., and Wang, Lihong V. *Tutorial on Compressed Ultrafast Photography.* Journal of Biomedical Optics, vol. 25, no. 7, 2020.

 Liang, Jinyang, et al. *Single-shot Real-time Compressed Ultrafast Photography.* Light: Science & Applications, vol. 6, 2017, e17005.

 Donoho, D. L. *Compressed Sensing.* IEEE Transactions on Information Theory, vol. 52, no. 4, 2006, pp. 1289–1306.

 Beck, A., and Teboulle, M. *A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems.* SIAM Journal on Imaging Sciences, vol. 2, no. 1, 2009, pp. 183–202.

 Bioucas-Dias, J., and Figueiredo, M. A. T. *A New TwIST: Two-Step Iterative Shrinkage/Thresholding Algorithms for Image Restoration.* IEEE Transactions on Image Processing, vol. 16, no. 12, 2007, pp. 2992–3004.

 Baraniuk, Richard G. *Compressive Sensing [Lecture Notes].* IEEE Signal Processing Magazine, vol. 24, no. 4, 2007, pp. 118–121.

 Candes, Emmanuel J., Romberg, Justin, and Tao, Terence. *Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information.* IEEE Transactions on Information Theory, vol. 52, no. 2, 2006, pp. 489–509.

 Eldar, Yonina C., and Kutyniok, Gitta, eds. *Compressed Sensing: Theory and Applications.* Cambridge University Press, 2012.