Data-driven methods in fluid dynamics: Sparse classification from experimental data

Zhe Bai[1], Steven L. Brunton[1,*], Bingni W. Brunton[2], J. Nathan Kutz[3], Eurika Kaiser[4], Andreas Spohn[4], and Bernd R. Noack[4,5]
[1] Department of Mechanical Engineering, University of Washington, Seattle, WA 98195, USA
[2] Department of Biology, University of Washington, Seattle, WA 98195, USA
[3] Department of Applied Mathematics, University of Washington, Seattle, WA 98195, USA
[4] Institut PPRIME, CNRS – Université de Poitiers – ENSMA, UPR 3346, Départment Fluides, Thermique, Combustion, F-86036 Poiters CEDEX, France
[5] ISM, Technische Universität Braunschweig, Germany

Abstract This work explores the use of data-driven methods, including machine learning and sparse sampling, for systems in fluid dynamics. In particular, camera images of a transitional separation bubble are used with dimensionality reduction and supervised classification techniques to discriminate between an actuated and an unactuated flow. After classification is demonstrated on full-resolution image data, similar classification performance is obtained using heavily sub-sampled pixels from the images. Finally, a sparse sensor optimization is used to determine optimal pixel locations for accurate classification. With 5-10 specially selected sensors, the median cross-validated classification accuracy is $\geq 97\%$, as opposed to a random set of 5-10 pixels, which result in classification accuracy of 70-80%. The methods developed here apply broadly to high-dimensional data from fluid dynamics experiments. Relevant connections between sparse sampling and the representation of high-dimensional data in a low-rank feature space are discussed.

Keywords– Flow visualization, reduced-order models, proper orthogonal decomposition, machine learning, classification, sparse sampling, compressed sensing.

1 The importance of data science in fluid dynamics

Fluid dynamics plays a central role in numerous scientific, industrial, and technological applications, including transportation (planes, trains, automobiles), energy (wind, tidal, combustion), and mixing (medicine, chocolate), to name only a few. Understanding and controlling fluid flows provides an opportunity to dramatically improve performance in these systems, resulting in lift increase, drag reduction, and mixing enhancement, all of which further important engineering goals of the modern world. Rapidly developing methods in data science, largely borne out of the computer science, statistics, and applied mathematics communities, offer a paradigm shift in our ability to measure, model, and manipulate fluid flows.

* Corresponding author. *E-mail address:* sbrunton@uw.edu (S.L. Brunton).

Fluid flows are often characterized by high-dimensional, multi-scale, and nonlinear phenomena that evolve on an attractor. Although the Navier-Stokes equations provide a detailed partial differential equation model, it is often difficult to use this representation for engineering design and control. An insightful quote of Richard Feynman, in his lecture on fluid mechanics, summarizes the central dilemma [1]:

> "The test of science is its ability to predict. Had you never visited the earth, could you predict the thunderstorms, the volcanos, the ocean waves, the auroras, and the colorful sunset?"

Instead of analyzing equations in isolation, we collect measurements of flows in relevant configurations and develop a hierarchy of models to describe critical features of the flow, rather than every subtle detail. In particular, extracting large coherent structures in fluids has provided valuable insights. The proper orthogonal decomposition (POD) [2, 3], which is often formulated using the singular value decomposition (SVD) [4, 5, 6], is a form of dimensionality reduction, which takes high-dimensional data from simulations or experiments and extracts relevant low-dimensional features. In many ways, these fundamental techniques in dimensionality reduction for fluids are among the first use of data-science in complex systems.

Reduced-order modeling has been especially important in obtaining computationally efficient models suitable for closed-loop feedback control. Many competing design constraints factor into effective control design, although one of the most important considerations is the latency in making a control decision, with larger latency imposing fundamental limitations on robust performance [7]. Thus, as flow speeds increase and flow structures become more complex, it becomes increasingly important to make fast control decisions based on efficient low-order models. A major open problem in control theory, with particular relevance for flow control, is the optimal placement of sensors and actuators for a control objective.

Powerful new techniques from data-science are poised to transform the analysis of complex data from dynamical systems, such as fluids. In particular, machine learning [8, 9] provides advanced capabilities to extract features and correlations. Sparse sampling techniques, including compressed sensing [10, 11, 12, 13, 14, 15], sparse regression [16, 17, 18], and sparse classification [19, 20, 21], allow for the recovery of relevant large-scale information from surprisingly few measurements.

Here, we combine machine learning and sparse sampling for efficient measurement and characterization of a fluid system. An overarching goal is to reduce the burden of data acquisition and processing. Specifically, we apply sparse classification to fluid imaging. Flow visualization, such as particle image velocimetry (PIV) [22, 23, 24], is a cornerstone in fluid mechanics, providing an understanding of flow structures and mechanisms that may be manipulated by closed-loop feedback flow control. Real-time feedback control based on PIV is becoming increasingly feasible, although it remains expensive, both in hardware cost and computational power. The methods here are designed to extract valuable data from inexpensive camera images of bubble visualizations, and they may also be used with PIV to reduce the data required for reconstruction, resulting in higher sampling rates and more inexpensive time-resolved systems. Finally, we design optimal sensor locations for categorical decisions [21], which may be eventually used for control.

1.1 Recent advances in sparsity and machine learning for fluids

Advanced methods from machine learning and compressed sensing have already begun to enter fluid dynamics. Unsupervised clustering has proven effective in determining probabilistic reduced-order models based on data from the mixing layer [25], and it has also been used to determine when to switch between various POD subspaces [26]. Graph theory has recently been applied to understand the network structure underlying unsteady fluids [27]. Finally, machine learning control, based on genetic programming [28], has been applied to numerous closed-loop flow control experiments with impressive performance increases that exceed many alternative control strategies [29, 30, 31].

Sparse sensing has rapidly been embraced by fluid dynamics researchers, most likely due to the ability to sample considerably less often than suggested by the Shannon-Nyquist sampling theorem [32, 33]. Although fluids data is typically quite large, is expensive to collect and analyze, and has a large separation of spatial and temporal scales, it generally has dominant low-dimensional structures that are useful for analysis and control. Compressed sensing has been applied in a variety of contexts in fluid dynamics [34, 35, 36, 37]. Sparsity techniques have also been applied to the computation of the dynamic mode decomposition (DMD) [38, 39, 40], including promoting sparsity for mode selection [41], spatial compressed DMD [42], and non-time resolved sampling strategies designed for particle image velocimetry (PIV) [43, 44]. The DMD is rapidly developing, with data science, machine learning, and control extensions [45, 46, 47, 48, 49, 50]. Sparsity methods have also been applied more broadly in dynamical systems [51, 52, 53, 54].

2 Experimental description

Experiments are conducted in a low-speed water tunnel at the Institute PPRIME, Poitiers. The closed-circuit, free surface water tunnel has a test section of $2.1\,m$ length, $0.5\,m$ width and $0.34\,m$ height. The ramp model consists of a flat plate of length $L = 100\,mm$ followed by a smooth ramp of height $60\,mm$ and length $600\,mm$. The model is $498\,mm$ wide and spans the width of the test section, except for $1\,mm$ gaps between the walls and the ramp. The ramp leading edge divides the oncoming flow into an upper stream following the ramp contour and a stream below the model. Downstream of the ramp, a horizontal plate prolongates the separated flow to reduce the impact of temporal changes in the flow structure during forcing. The stagnation point on the leading edge is controlled by adjustable pressure losses at the outlet of the upper stream. The Reynolds number is given as $Re = UL/\nu$ with respect to the free-stream velocity U, and the kinematic viscosity ν of water. The Reynolds number is fixed to $Re = 7900 \pm 100$. A schematic is shown in Fig. 1 (left).

Beginning from the leading edge, a laminar zero pressure gradient boundary layer develops along the flat plate. Above the smooth ramp this boundary layer separates under the influence of an adverse pressure gradient which is fixed by the shape of the ramp. Downstream of the flow detachment, the newly-formed separated shear layer becomes unstable and undergoes laminar-to-turbulent transition, allowing the

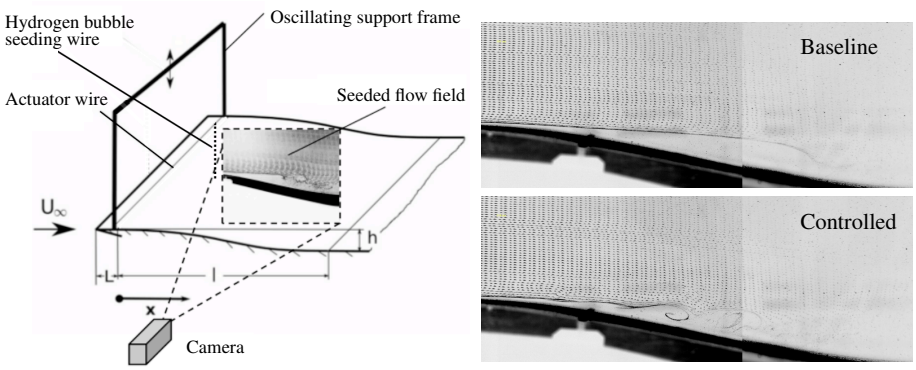

Fig. 1 (left) Schematic illustrating the experimental set-up, including bubble visualizations of the separated flow past a backward facing ramp. (right) Bubble visualizations for flow past a ramp are shown for the baseline case (top) and the case with control (bottom).

flow to reattach. Between the wall and the separated main flow, recirculating fluid marks the extensions of the laminar separation bubble (LSB). The ramp contour follows a polynomial shape of order 7 for which Sommer [55] numerically determined the position of the laminar separation bubble.

Locally-controlled forcing is enabled by a stainless steel wire of $0.13 \pm 0.01\,mm$ in diameter and supported by an oscillating holder. The wire crosses the span of the model and is located at $90 \pm 2.5\,mm$ downstream of the leading edge. A vertical sinusoidal motion of the wire is imposed using a line servo (RS-2 modelcraft) piloted by an Arduino-Due microprocessor. The frequency is varied between 0.1 and $3\,Hz$ and the oscillation amplitude is set at $3 \pm 1\,mm$. In all experiments, the mean vertical position of the oscillating wire was fixed at $3.5 \pm 0.5\,mm$ above the ramp model.

Flow visualizations are obtained using the hydrogen bubble technique [56]. For that purpose, a $0.050 \pm 0.005\,mm$ thick stainless steel wire deformed into a zigzag pattern is fixed in the middle of the ramp at $300 \pm 5mm$ downstream of the leading edge. When applying a negative potential, between 30 and 90 Volts, hydrogen bubbles are produced at the wire and convected downstream. A computer controlled function generator is employed to trigger the release of bubbles to obtain periodic timelines. These timelines mark the position of the separated shear layer and patches related to the rolling up of tracer particles by vortical structures during reattachment, as shown in Fig. 1 (right) for the baseline and controlled cases.

The images have a resolution of 2116×812 pixels, and they are acquired at 10 Hz. During the process of recording the image sequence, the bubble diameter increase and the timely precision of bubble release diminishes due to electrochemical processes close to the electrodes. Furthermore, during their progression in the downstream direction, the bubbles shrink. Therefore, the intensity of light reflections and contrast change in time and space during an image sequence. In the following analysis, we classify baseline and control cases using the full image data, with lighting changes, etc., and we also use an isolated data set that consists of a short sequence of images with constant lighting and bubble density. Throughout, these will be referred to as “Full Data" and “Isolated Data", with the modifiers “Baseline" or “Controlled".

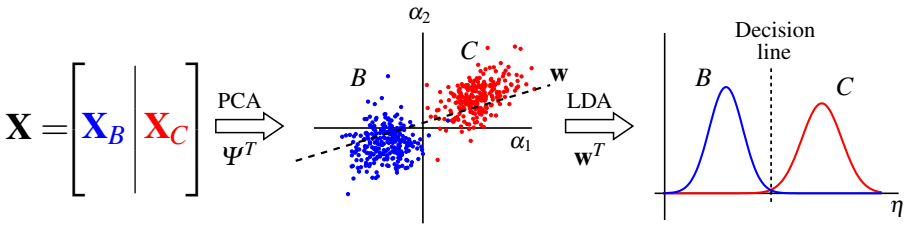

Fig. 2 Schematic illustrating the use of PCA (feature extraction) and LDA (supervised classifier) for the automatic classification of data into two categories B and C.

3 Classification of fluids based on image data

Here, we demonstrate supervised learning techniques to distinguish between the baseline and controlled fluid flow fields from camera images. Supervised learning requires labeled training data, where the desired distinction (i.e., baseline vs controlled) is recorded in a vector of labels (i.e., 'B' corresponds to baseline images and 'C' corresponds to controlled images). In contrast, unsupervised learning, such as K-means, seeks to find natural clustering of the data in some feature space.

3.1 Methods – machine learning and dimensionality reduction

The methods presented here are general, and may be used to estimate other relevant flow quantities, as long as there is a labeled set of training data. Figure 2 shows a schematic of the supervised classification algorithm used in this work. A data matrix $\mathbf{X} = \left[\mathbf{X}_B \,\middle|\, \mathbf{X}_C\right]$ is constructed by concatenating image vectors from the baseline ('B') and controlled ('C') cases. Each image is reshaped into a large column vector with as many rows as pixels in the original image, similar to how velocity fields are stacked in the method of snapshots [6]. The mean image is subtracted from $\mathbf{X}$.

Next, a low-rank feature space, Ψ, is obtained by applying the principal components analysis (PCA), which is closely related to POD/SVD:

$$\mathbf{X} = \Psi \Sigma \mathbf{V}^*. \tag{1}$$

In this low-dimensional coordinate system, the data is assumed to separate into clusters according to the labels. Often the basis Ψ is truncated to only contain energetic modes. A state $\mathbf{x}$ may be approximated in this truncated coordinate system as $\mathbf{x} \approx \Psi\alpha$, where α are the PCA/POD coordinates of $\mathbf{x}$ in Ψ.

Finally, it is possible to identify the direction $\mathbf{w}$ in feature space that optimally separates the data clusters using the linear discriminant analysis (LDA) [8, 9]. Once the discriminant vector $\mathbf{w}$ is determined, it is possible to project images into a decision space by taking the inner product of the image PCA coordinates α with $\mathbf{w}$.

$$\eta = \mathbf{w}^T \alpha = \mathbf{w}^T \Psi^T \mathbf{x}. \tag{2}$$

The value of η determines whether the image $\mathbf{x}$ is classified as category 'B' or 'C'.

The performance of a classifier is determined using cross-validation, whereby the data is randomly partitioned into a training set (80%) and a test set (20%). The classifier is built using only training data and it is then used to predict labels in the test set; the percentage of correctly identified test labels determines the accuracy of the classifier. Many rounds of cross-validation are performed on different $80\%/20\%$ random shuffling of the data.

There are many alternatives to the choices above. First, if the data does not cluster in a PCA feature space, then *feature engineering* will be critical to determine the transformations that isolate features to distinguish the data. Next, there is a host of advanced supervised learning techniques including quadratic discriminant analysis (QDA), support vector machines (SVM), and decision trees, to name a few [8, 9]. However, we prefer to use PCA/LDA because of the ease of implementation and their usefulness with optimization algorithms in later sections. And most importantly, the data is well-separated with LDA in a PCA feature space.

PCA is often computed using an SVD, which is a spatial-temporal decomposition of data $\mathbf{X}$ into a hierarchy of spatial coherent structures, given by the columns of Ψ, and temporal coherent patterns, given by the columns of $\mathbf{V}$. The importance of each mode is quantified by the entries of the diagonal matrix Σ. For high-dimensional data the SVD may be computed using the method of snapshots [6]:

$$\mathbf{X}^*\mathbf{X} = \mathbf{V}\Sigma^2\mathbf{V}^* \quad \Longrightarrow \quad \mathbf{X}^*\mathbf{X}\mathbf{V} = \mathbf{V}\Sigma^2. \tag{3}$$

Thus Σ and $\mathbf{V}$ may be obtained by an eigendecomposition of the symmetric matrix $\mathbf{X}^*\mathbf{X}$. Afterwards, the modes Ψ may be constructed as: $\Psi = \mathbf{X}\mathbf{V}\Sigma^{-1}$. Note that Ψ and $\mathbf{V}$ are both unitary matrices.

3.2 Classification results on high resolution image data

Figure 3 shows the results of principal components analysis on the high-resolution full image sequence data. The modal variance decays somewhat slowly, and the modes and coefficients are shown below. Mode 2 corresponds to a lighting change observed in the full image sequence, which can also be seen in the spikes in the temporal coefficients in both the baseline and controlled data. When performing PCA on the isolated image sequence, there is no longer a mode corresponding to a change in lighting, and the modal energy decays more rapidly.

Figure 4 shows the baseline and controlled data projected into the first three PCA coordinates, for both the full image sequence data and the isolated image sequence data. In both cases, the baseline and control sequences are well separated, although the separation is better for the isolated images, which have more uniform conditions. Figure 5 shows the separating plane determined by LDA. Table 1 quantifies the performance of LDA classification in a PCA space with 5 modes and with 10 modes. With 10 PCA modes, the LDA classifier is perfect in both the isolated and full image sequences. Using only 5 modes, the full image sequence has around 4% error.

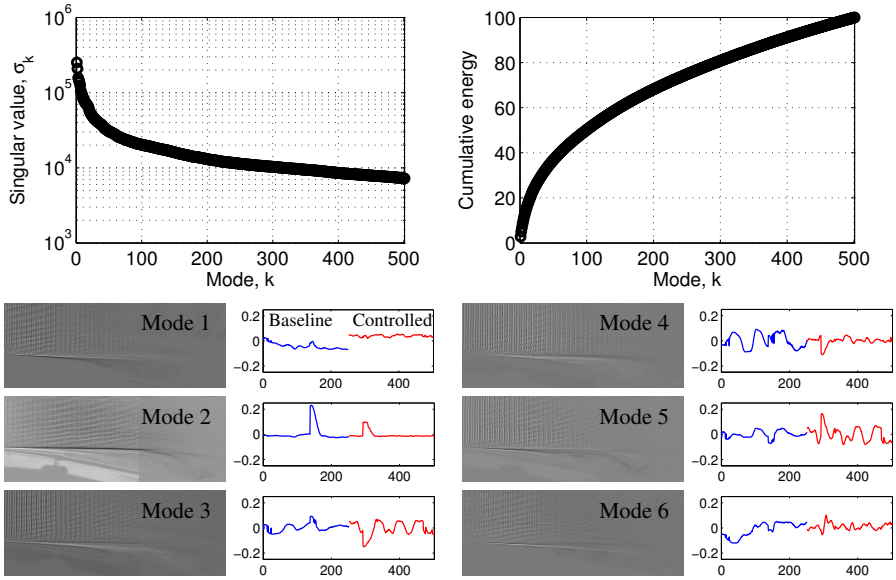

Fig. 3 PCA results on full image sequence data. The singular values (top) indicate the energy of each mode. The PCA modes (left) and coefficients (right) show dominant spatial/temporal features.

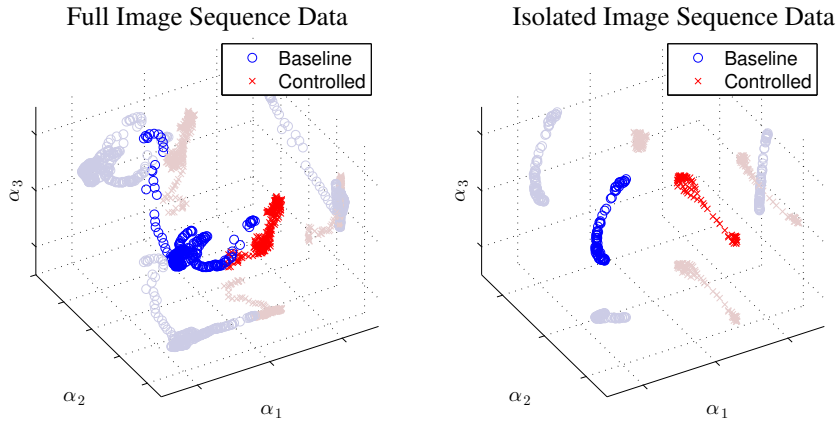

Fig. 4 Data plotted in the first three PCA coordinates $\alpha = (\alpha_1, \alpha_2, \alpha_3)^T$. The full data (left) is reasonably well-separated. The isolated data (right) is very well separated.

Table 1 Performance of LDA classification in a PCA feature space with 5 and 10 modes on the full image sequence data and the isolated image sequence data.

		Full Image Sequence	Isolated Image Sequence
Error	5 Modes	$3.82 \pm 1.79\%$	0.00%
	10 Modes	0.00%	0.00%

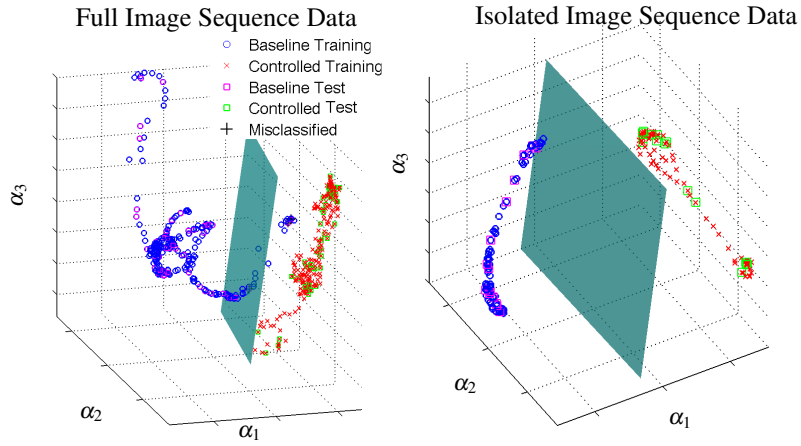

Fig. 5 The LDA separating plane is shown for one instance of cross-validation. Although all controlled data are correctly classified, any purple squares to the right of the plane are misclassified, and are also labeled with black crosses.

4 Sparse classification on compressed/subsampled data

After demonstrating in the previous section that flows may be classified accurately using full-resolution images, here we show that similar classification may be achieved using heavily subsampled or compressed image data. This is important to reduce the data acquisition and processing required for high-level decisions. Reducing processing is important for mobile applications, where on-board computations are power constrained, and for control, where the fastest decision is desirable.

4.1 Methods – sparsity and low rank structures

In this section, we assume that we take subsampled or compressed measurements $\mathbf{Y}$, which are related to the full resolution data $\mathbf{X}$ by:

$$\mathbf{Y} = \mathbf{C}\mathbf{X}. \tag{4}$$

The matrix $\mathbf{C} \in \mathbb{R}^{p \times n}$ is a measurement matrix. It may consist of p random rows of the identity matrix, which would correspond to p single-pixel measurements at those locations. Alternatively, $\mathbf{C}$ may be a matrix of independent, identically distributed Gaussian or Bernoulli random variables. Random Gaussian measurements are generically powerful for signal reconstruction [12], but single pixel measurements are particularly useful for engineering purposes. Beyond their use in classifying images, we may consider point sensor placement on a wing or in the ocean or atmosphere to accumulate information about complex time-varying flows.

Even with a significant reduction in the data, accurate classification is possible, since the relevant information exists in a low-dimensional subspace. Nearly all natural images are sparse in a discrete Fourier transform (DFT) basis, meaning that most of the Fourier coefficients are small and may be neglected; this is the foundation of image compression. Fluid velocity fields are also sparse in the Fourier domain [35].

If the data $\mathbf{X}$ is sparse in a basis Ψ (either DFT or PCA), then we may write:

$$\mathbf{Y} = \mathbf{C}\mathbf{X} = \mathbf{C}\Psi\mathbf{S}, \tag{5}$$

where the columns of $\mathbf{S}$ are sparse vectors (i.e., mostly zero), and the basis Ψ is a unitary matrix. Compressed sensing is based on the observation that under certain conditions on the measurement matrix $\mathbf{C}$, the projection $\mathbf{C}\Psi$ will act as a near isometry on sparse vectors [11, 12, 13]. This means that inner products of the columns of $\mathbf{Y}$ will be similar to the inner products of corresponding columns of $\mathbf{S}$. Further, since Ψ is unitary in the case of a DFT or PCA basis, these inner products of columns of $\mathbf{Y}$ will also resemble inner products of columns of $\mathbf{X}$. Thus, using the method of snapshots, we recover the dominant correlations in the data $\mathbf{X}$ from the SVD of $\mathbf{Y}$:

$$\mathbf{Y}^*\mathbf{Y} \approx \mathbf{X}^*\mathbf{X} = \mathbf{S}^*\mathbf{S}. \tag{6}$$

4.2 Classification results on subsampled data

Figure 6 shows the PCA projection of the baseline and controlled data for random single-pixel subsampling of the data. In the top row, $p = 1718$ random pixels are used, which account for 0.1% of the total pixels in the image. Decreasing the number of random pixels causes the clusters to merge, making classification more difficult.

Figure 7 shows the cross-validated classification error versus the number of random sensors chosen. In both the top and the bottom plots, LDA classification is applied in a PCA feature space with 10 modes, and 1000 instances were used for cross-validation. For the isolated image sequence data, the median error is 0% for as few as 34 random sensors, and for the full image sequence data, the median error is 0% for 344 random sensors. As might be expected, it is easier to classify baseline and control images in the isolated image sequence, because it is more uniform and coherent. However, depending on the $80\%/20\%$ partition used for cross-validation, the classification error may be nearly 50%.

The ability to perform accurate classification with $p \sim \mathscr{O}(10)$–$\mathscr{O}(100)$ randomly selected single-pixel sensors has significant implications in the data-driven processing and control of fluid systems from optical measurements. First, less spatial data must be collected, reducing data transfer and making improved temporal sampling rates possible. Second, all computations are done in a low-dimensional subspace, making it possible to make control decisions with low latency.

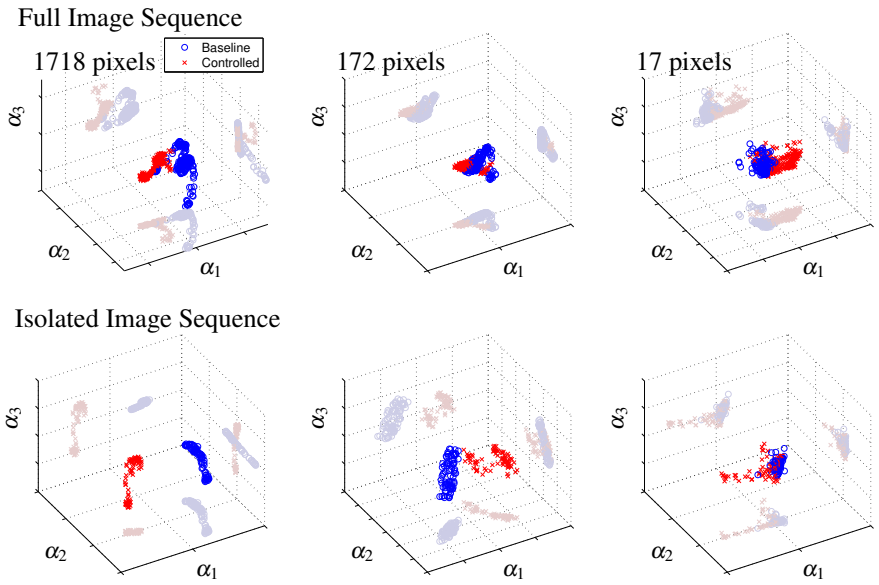

Fig. 6 Subsampled data plotted in the first three PCA coordinates for the full image sequence (top) and isolated image sequence (bottom). The number of random single-pixel sensors range from 1718 (left), to 172 (middle), to 17 (right). With more compression, the clusters begin to merge.

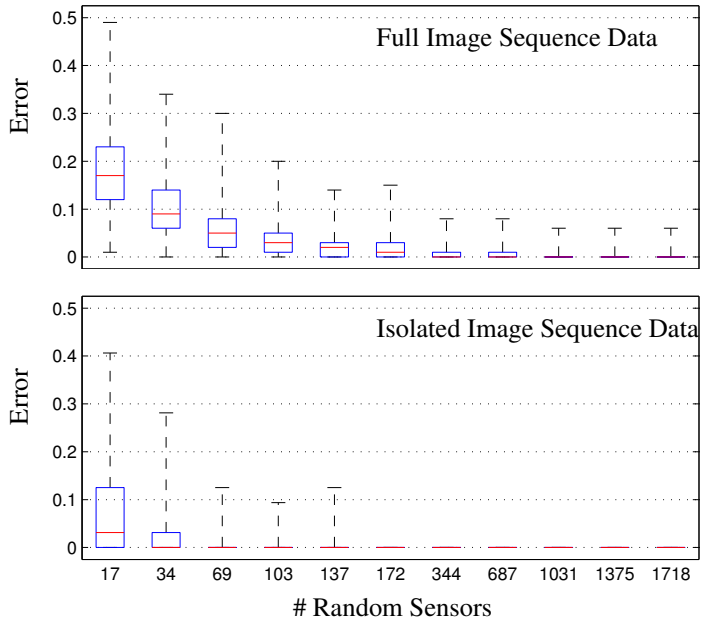

Fig. 7 Error vs. number of random single-pixel sensors on full image sequence (top) and isolated image sequence (bottom) for 10 PCA modes. 1000 instances are used for cross-validation. The red line is the median, and the dashed lines and blue box boundaries denote quartiles of the distribution.

5 Optimal sensor placement and enhanced sparsity

In the previous section, we demonstrated that machine learning may be applied to heavily subsampled data, although performance was degraded at large compression ratios. Here, we demonstrate an algorithm that optimizes sensor locations for categorical decisions, resulting in accurate classification with an order of magnitude less sensors than achieved with random placement [21].

5.1 Methods – optimal sensor placement

One of the cornerstone advances in compressed sensing is that it is now possible to solve for the sparsest solution vector to an underdetermined system of equations

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{7}$$

using convex optimization. Previously, solving for the sparsest vector $\mathbf{x}$ would involve a combinatorial brute-force search to find the $\mathbf{x}$ with smallest ℓ_0 norm, where $\|\mathbf{x}\|_0$ is equal to the number of nonzero elements in $\mathbf{x}$. However, it is now known that we may approximate the sparsest solution with *high probability* by minimizing the ℓ_1 norm, $\|\mathbf{x}\|_1 = \sum_{k=1}^{N} |x_k|$, which is a convex minimization. Therefore, it is now possible to solve increasingly large systems in a way that scales favorably with Moore's law of exponentially increasing computer power. There are a number of technical restrictions on the sizes of $\mathbf{x}$ and $\mathbf{b}$ as well as the spectral properties of the matrix $\mathbf{A}$ [11, 12, 13].

Recently, the ℓ_1 convex-minimization architecture has been leveraged to solve for optimal sensor placement for categorical decision making [21]. This optimization seeks to find a small number of pixels that are able to capture as much information as possible about the position of an image in the decision space. Specifically, we seek to find the sparsest vector $\mathbf{s} \in \mathbb{R}^n$ that satisfies the following relationship:

$$\mathbf{s} = \underset{\mathbf{s}'}{\operatorname{argmin}} \|\mathbf{s}'\|_1 \text{ such that } \Psi^T \mathbf{s} = \mathbf{w}. \tag{8}$$

The vector $\mathbf{s}$ is the size of a full image, but it contains mostly zeros. Since $\mathbf{w}$ is in an r-dimensional feature space, Eq. (8) may be solved with a vector $\mathbf{s}$ with at most r nonzero components. Thus, it is possible to sample the image data at these r critical pixel locations, and perform classification in an r dimensional subspace. This is called the sparse sensor placement optimization for classification (SSPOC) algorithm. We will demonstrate that accurate classification may be achieved using an order of magnitude fewer sensors, as compared with randomly placed sensors.

5.2 Classification on optimized sensors

Figure 8 shows the PCA clustering of data using 6 optimal sensor locations (top) and 6 randomly chosen pixels (bottom). The cluster separation with optimal sensors is striking, when compared with the clusters from random sensors. The cross-validated classification performance is shown in Fig. 9. The optimal 6 sensor locations provide a significant improvement over random.

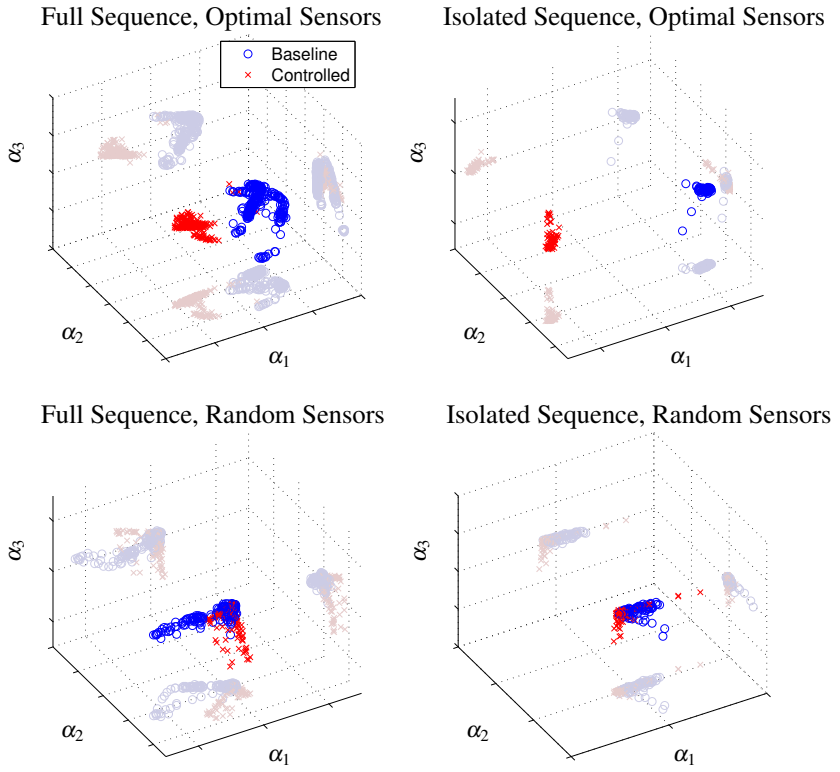

Fig. 8 PCA clustering of data using optimal sensors (top) and using random sensors (bottom).

Figure 10 shows the ensemble of sensor locations determined by the SSPOC algorithm over 100 instances of cross-validation. A number of interesting features are found in this data, including sampling of the boundary layer profile and the shear layer. The boundary layer sampling is more pronounced in the isolated image sequence data. In the baseline case, the shear layer remains steady and is nearly horizontal, as opposed to the controlled case, where the Kelvin-Helmholtz instability causes vortex roll-up to occur much sooner (see Fig. 11).

In the image sequence of the controlled case, the disturbance propagation can be observed close to the ramp wall before the flow actually separates. This may explain why so few sensors are along the separation line in the isolated image sequence.

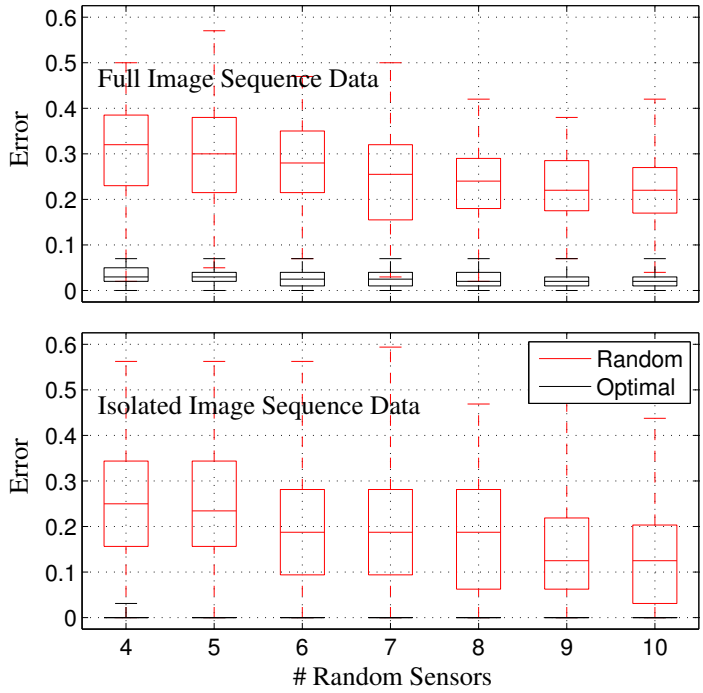

Fig. 9 Comparison of cross-validated error using optimal sensor locations (black) and random sensors (red) on the full image sequence (top) and the isolate image sequence (bottom). Here, the LDA classification is done directly in the pixel space.

6 Conclusions and discussion

In this analysis, we demonstrate that methods from machine learning and sparse sampling may be applied to classify fluid flows from inexpensive camera images. In particular, we use linear discriminant analysis (LDA) clustering techniques in a POD/PCA reduced subspace to classify images of a transitional separation bubble with and without forcing. Sparsity techniques are used to demonstrate that similar classification performance can be obtained with many fewer pixel measurements. Finally, a sparse sensor optimization algorithm is used to determine the fewest pixel sensors required for classification. We find that a small handful of sensors (between 5 and 10) result in a median cross-validated classification performance of $\geq 97\%$.

There are numerous avenues to extend this work in fluid dynamics. First, it would be natural to apply these methods to multi-way classification in flows with more distinct states. It may also be possible to estimate the phase of a periodic or quasi-periodic flow for use in a closed-loop feedback control strategy. The sparse estimation of bifurcation regimes may also be useful for parameterized reduced-order modeling techniques [25, 26].

More generally, we envision an increasing role for innovations in data-science for complex fluid flows. With increasingly large data sets, methods to distill mean-

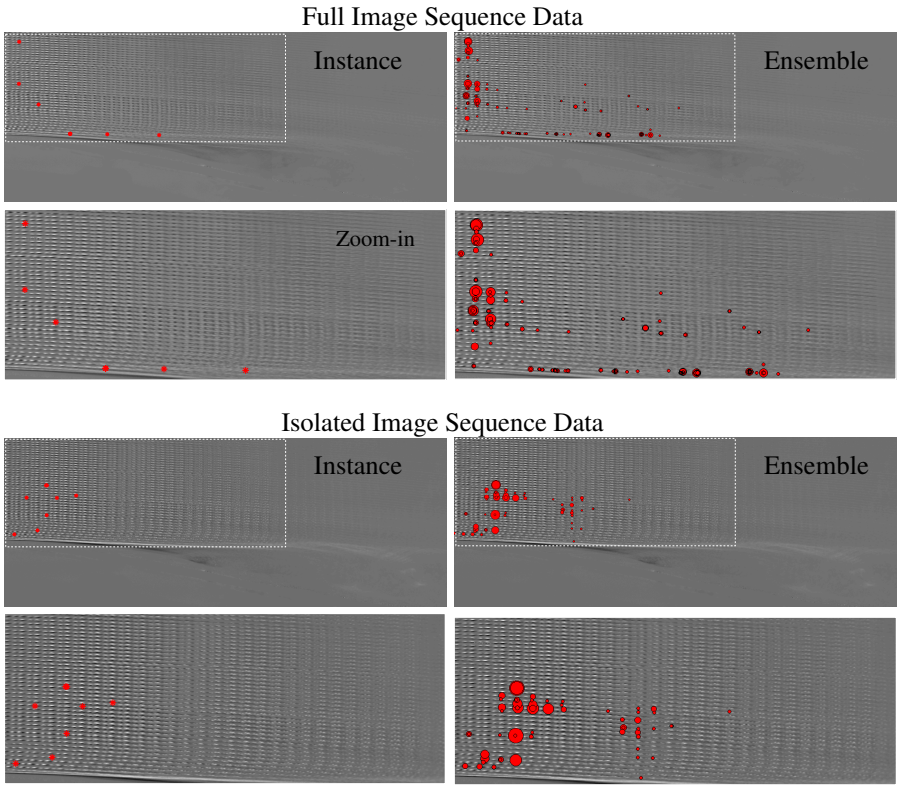

Fig. 10 Optimal sensor locations (red) for full image sequence data (top) and isolated image sequence data (bottom). A single cross-validation instance is shown on the left, and the ensemble of sensor locations are shown on the right. In each case, the second row provides a zoom-in near the ramp. The size of the circle denotes how frequently this location was chosen in the ensemble.

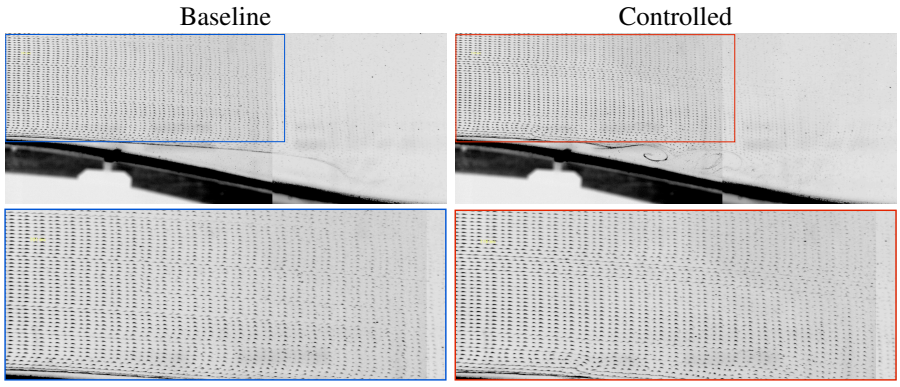

Fig. 11 Bubble visualizations for flow past a ramp with zoom-in around inlet.

ingful features from subsampled data will become more important. Furthermore, bio-inspired engineering and control design will likely favor low-dimensional computations that evolve on subspaces or manifolds that capture relevant information for control and decision tasks. The simultaneous explosion of data, the miniaturization of sensing and actuation hardware, and the renaissance of techniques in applied mathematics make this an exciting time for data-driven control in fluid dynamics.

Acknowledgements We gratefully acknowledge discussions with Josh L. Proctor about sparsity methods in machine learning. SLB acknowledges generous support from the Department of Energy (DOE DE-EE0006785) and from the University of Washington department of Mechanical Engineering. SLB and BWB acknowledge sponsorship by the UW eScience Institute as Data Science Fellows. EK, AS, and BRN acknowledge additional support by the ANR SepaCoDe (ANR-11-BS09-018) and ANR TUCOROM (ANR-10-CEXC-0015).

References

1. Richard P Feynman, Robert B Leighton, and Matthew Sands. *The Feynman Lectures on Physics*, volume 2. Basic Books, 2013.
2. G. Berkooz, P. Holmes, and J. L. Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 23:539–575, 1993.
3. P. J. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge Monographs in Mechanics. Cambridge University Press, Cambridge, England, 2nd edition, 2012.
4. G. Golub and W. Kahan. Calculating the singular values and pseudo-inverse of a matrix. *Journal of the Society for Industrial & Applied Mathematics, Series B: Numerical Analysis*, 2(2):205–224, 1965.
5. G. H. Golub and C. Reinsch. Singular value decomposition and least squares solutions. *Numerical Mathematics*, 14:403–420, 1970.
6. L. Sirovich. Turbulence and the dynamics of coherent structures, parts I-III. *Q. Appl. Math.*, XLV(3):561–590, 1987.
7. S. Skogestad and I. Postlethwaite. *Multivariable feedback control: analysis and design*. John Wiley & Sons, Inc., Hoboken, New Jersey, 2 edition, 2005.
8. Christopher M Bishop et al. *Pattern recognition and machine learning*, volume 1. springer New York, 2006.
9. J. N. Kutz. *Data-Driven Modeling & Scientific Computation: Methods for Complex Systems & Big Data*. Oxford University Press, 2013.
10. D. L. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.
11. E. J. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, 2006.
12. E. J. Candès and T. Tao. Near optimal signal recovery from random projections: Universal encoding strategies? *IEEE Transactions on Information Theory*, 52(12):5406–5425, 2006.
13. E. J. Candès. Compressive sensing. *Proceedings of the International Congress of Mathematics*, 2006.
14. R. G. Baraniuk. Compressive sensing. *IEEE Signal Processing Magazine*, 24(4):118–120, 2007.
15. J. A. Tropp and A. C. Gilbert. Signal recovery from random measurements via orthogonal matching pursuit. *IEEE Transactions on Information Theory*, 53(12):4655–4666, 2007.
16. Trevor Hastie, Robert Tibshirani, Jerome Friedman, T Hastie, J Friedman, and R Tibshirani. *The elements of statistical learning*, volume 2. Springer, 2009.

17. Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*. Springer, 2013.
18. Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.
19. J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma. Robust face recognition via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 31(2):210–227, 2009.
20. L. Clemmensen, T. Hastie, D. Witten, and B. Ersbøll. Sparse discriminant analysis. *Technometrics*, 53(4), 2011.
21. B. W. Brunton, S. L. Brunton, J. L. Proctor, and J. N. Kutz. Optimal sensor placement and enhanced sparsity for classification. *arXiv preprint arXiv:1310.4217*, 2013.
22. D. P. Hart. High-speed PIV analysis using compressed image correlation. *Journal of Fluids Engineering*, 120:463–470, 1998.
23. S. Petra and C. Schn orr. Tomopiv meets compressed sensing. *Pure Mathematics and Applications*, 20(1-2):49–76, 2009.
24. C. E. Willert and M. Gharib. Digital particle image velocimetry. *Experiments in Fluids*, 10(4):181–193, 1991.
25. E. Kaiser, B. R. Noack, L. Cordier, A. Spohn, M. Segond, M. Abel, G. Daviller, J. Osth, S. Krajnovic, and R. K. Niven. Cluster-based reduced-order modelling of a mixing layer. *J. Fluid Mech.*, 754:365–414, 2014.
26. David Amsallem, Matthew J Zahr, and Charbel Farhat. Nonlinear model order reduction based on local reduced-order bases. *International Journal for Numerical Methods in Engineering*, 92(10):891–916, 2012.
27. Aditya G Nair and Kunihiko Taira. Network-theoretic approach to sparsified discrete vortex dynamics. *Journal of Fluid Mechanics*, 768:549–571, 2015.
28. John R Koza. *Genetic programming: on the programming of computers by means of natural selection*, volume 1. MIT press, 1992.
29. N. Gautier, J-L Aider, T. Duriez, BR Noack, M. Segond, and M. Abel. Closed-loop separation control using machine learning. *Journal of Fluid Mechanics*, 770:442–457, 2015.
30. T. Duriez, V. Parezanovic, J.-C. Laurentie, C. Fourment, J. Delville, J.-P. Bonnet, L. Cordier, B. R. Noack, M. Segond, M. Abel, N. Gautier, J.-L. Aider, C. Raibaudo, C. Cuvier, M. Stanislas, and S. L. Brunton. Closed-loop control of experimental shear flows using machine learning. AIAA Paper 2014-2219, 7th Flow Control Conference, 2014.
31. V. Parezanovic, J.-C. Laurentie, T. Duriez, C. Fourment, J. Delville, J.-P. Bonnet, L. Cordier, B. R. Noack, M. Segond, M. Abel, T. Shaqarin, and S. L. Brunton. Mixing layer manipulation experiment – from periodic forcing to machine learning closed-loop control. *Journal Flow Turbulence and Combustion*, 94(1):155–173, 2015.
32. H. Nyquist. Certain topics in telegraph transmission theory. *Transactions of the A. I. E. E.*, pages 617–644, FEB 1928.
33. C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27(3):379–423, 1948.
34. I. Bright, G. Lin, and J. N. Kutz. Compressive sensing and machine learning strategies for characterizing the flow around a cylinder with limited pressure measurements. *Physics of Fluids*, 25:127102–1–127102–15, 2013.
35. Zhe Bai, Thakshila Wimalajeewa, Zachary Berger, Guannan Wang, Mark Glauser, and Pramod K Varshney. Low-dimensional approach for reconstruction of airfoil data via compressive sensing. *AIAA Journal*, 53(4):920–933, 2014.
36. J-L Bourguignon, JA Tropp, AS Sharma, and BJ McKeon. Compact representation of wall-bounded turbulence using compressive sampling. *Physics of Fluids (1994-present)*, 26(1):015109, 2014.
37. Ido Bright, Guang Lin, and J Nathan Kutz. Classification of spatio-temporal data via asynchronous sparse sampling: Application to flow around a cylinder. *arXiv:1506.00661*, 2015.
38. C. W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, and D. S. Henningson. Spectral analysis of nonlinear flows. *Journal of Fluid Mechanics*, 641:115–127, 2009.

39. P. J. Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, August 2010.
40. J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz. On dynamic mode decomposition: theory and applications. *J. Computational Dynamics*, 1(2):391–421, 2014.
41. M. R. Jovanović, P. J. Schmid, and J. W. Nichols. Low-rank and sparse dynamic mode decomposition. *Center for Turbulence Research*, 2012.
42. S. L. Brunton, J. L. Proctor, and J. N. Kutz. Compressive sampling and dynamic mode decomposition. *arXiv preprint arXiv:1312.5186*, 2014.
43. Jonathan H Tu, Clarence W Rowley, J Nathan Kutz, and Jessica K Shang. Spectral analysis of fluid flows using sub-nyquist-rate piv data. *Experiments in Fluids*, 55(9):1–13, 2014.
44. F. Gueniat, L. Mathelin, and L. Pastur. A dynamic mode decomposition approach for large and arbitrarily sampled systems. *Physics of Fluids*, 27(2):025113, 2015.
45. J. Gosek and J. N. Kutz. Dynamic mode decomposition for real-time background/foreground separation in video. *submitted for publication*, 2013.
46. M. O. Williams, C. W. Rowley, and I. G. Kevrekidis. A kernel approach to data-driven koopman spectral analysis. *arXiv preprint arXiv:1411.2260*, 2014.
47. M. O. Williams, I. G. Kevrekidis, and C. W. Rowley. A data-driven approximation of the koopman operator: extending dynamic mode decomposition. *arXiv:1408.4408*, 2014.
48. M. S. Hemati, M. O. Williams, and C. W. Rowley. Dynamic mode decomposition for large and streaming datasets. *Physics of Fluids*, 26(11):111701, 2014.
49. M. O. Williams, C. W. Rowley, I. Mezić, and I. G. Kevrekidis. Data fusion via intrinsic dynamic variables: An application of data-driven koopman spectral analysis. *EPL (Europhysics Letters)*, 109(4):40007, 2015.
50. J. L. Proctor, S. L. Brunton, and J. N. Kutz. Dynamic mode decomposition with control: Using state and input snapshots to discover dynamics. *arxiv*, 2014.
51. H. Schaeffer, R. Caflisch, C. D. Hauck, and S. Osher. Sparse dynamics for partial differential equations. *Proceedings of the National Academy of Sciences USA*, 110(17):6634–6639, 2013.
52. Alan Mackey, Hayden Schaeffer, and Stanley Osher. On the compressive spectral method. *Multiscale Modeling & Simulation*, 12(4):1800–1827, 2014.
53. S. L. Brunton, J. H. Tu, I. Bright, and J. N. Kutz. Compressive sensing and low-rank libraries for classification of bifurcation regimes in nonlinear dynamical systems. *SIAM Journal on Applied Dynamical Systems*, 13(4):1716–1732, 2014.
54. J. L. Proctor, S. L. Brunton, B. W. Brunton, and J. N. Kutz. Exploiting sparsity and equation-free architectures in complex systems (invited review). *The European Physical Journal Special Topics*, 223(13):2665–2684, 2014.
55. F. Sommer. Mehrfachlösungen bei laminaren Strömungen mit Druckinduzierter Ablösung: eine Kuspen-Katastrophe. *VDI Fortschrittsbericht, Reihe 7, Nr. 206, VDI Verlag Düsseldorf (Dissertation Bochum)*, pages 429–443, 2992.
56. Frederick Anthony Schraub, SJ Kline, J Henry, PW Runstadler, and A Littell. Use of hydrogen bubbles for quantitative determination of time-dependent velocity fields in low-speed water flows. *Journal of Fluids Engineering*, 87(2):429–444, 1965.