Dynamic Mode Decomposition for Computer Vision and Signal Processing

Santosh Tirunagari

A thesis submitted for the degree of Doctor of Philosophy in Computer Science from the University of Surrey

University of Surrey

Department of Computer Science
Faculty of Engineering and Physical Sciences
University of Surrey
Guildford, Surrey GU2 7XH
United Kingdom

Supervisors:
Principal Supervisor: Dr. Norman Poh
Co-Supervisor: Prof. Miroslaw Bober
Co-Supervisor: Dr. David Windridge

December 2016
Santosh Tirunagari, 2016
Declaration of Originality

This thesis and the work to which it refers are the results of my own efforts. Any ideas, data, images or text resulting from the work of others (whether published or unpublished) are fully identified as such within the work and attributed to their originator in the text, bibliography or in footnotes. This thesis has not been submitted in whole or in part for any other academic degree or professional qualification. I agree that the University has the right to submit my work to the plagiarism detection service TurnitinUK for originality checks. Whether or not drafts have been so-assessed, the University reserves the right to require an electronic version of the final document (as submitted) for assessment as above.

November, 2016
Santosh Tirunagari
Summary

The method of Dynamic Mode Decomposition (DMD) was introduced originally in the area of Computational Fluid Dynamics (CFD) for extracting coherent structures from spatio-temporal complex fluid flow data. DMD takes in time series data and computes a set of modes, each of which is associated with a complex eigenvalue. DMD analysis is closely associated with spectral analysis of the Koopman operator, which provides linear but infinite-dimensional representation of nonlinear dynamical systems. Therefore, by using DMD a nonlinear system could be described by a superposition of modes whose dynamics are governed by the eigenvalues. The key advantage of DMD is its data-driven nature which does not rely on any prior assumptions except the inherent dynamics which are observed over time. Its capability for extracting relevant modes from complex fluid flows has seen significant application across multiple fields, including computer vision, robotics and neuroscience. This thesis, in order to expand DMD to other applications, advances the original formulation so that it can be used to solve novel problems in the fields of signal processing and computer vision.

In signal processing this thesis introduces the method of using DMD for decomposing a univariate time series into a number of interpretable elements with different subspaces, such as noise, trends and harmonics. In addition, univariate time series forecasting is shown using DMD. The computer vision part of this thesis focuses on innovative applications pertaining to the areas of medical imaging, biometrics and background modelling. In the area of medical imaging a novel DMD framework is proposed that introduces windowed and reconstruction variants of DMD for quantifying kidney function in Dynamic Contrast Enhanced Magnetic Resonance imaging (DCE-MRI) sequences, through movement correction and functional segmentation of the kidneys. The biometrics portion of this thesis introduces a DMD based classification pipeline for counter spoofing 2D facial videos and static finger vein images. The finger vein counter spoofing makes use of a novel atemporal variant of DMD that captures micro-level artefacts that can differentiate the quality and light reflection properties between a live and a spoofed finger vein image, while the DMD on 2D facial image sequences distinguishes attack specific cues from a live face by capturing complex dynamics of head movements, eye-blinking and lip-movements.
in a data driven manner. Finally, this thesis proposes a new technique using DMD to obtain a background model of a visual scene in the colour domain. These aspects form the major contributions of this thesis.

The results from this thesis present DMD as a promising approach for applications requiring feature extraction including: (i) trends and noise from signals, (ii) micro-level texture descriptor from images, and (iii) coherent structures from image sequences/videos, as well as applications that require suppression of movements from dynamical spatio-temporal image sequences.
Acknowledgements

I would like to express my gratitude to all the people who have helped me to pursue my PhD degree at University of Surrey, Guildford, UK. Firstly I would like to thank Dr. Norman Poh, my principal supervisor, who has been more than a supervisor, a mentor and an advisor. I extend my sincere thanks to Dr. David Windridge and Prof. Miroslaw Bober, who have been my thesis co-supervisors. Without any doubt I am able to finish this research work because of my supervisors support. I also wish to thank Dr. Kevin Wells, who has been very cooperative during our collaboration. I would like to thank the Department of Computer Science and Center for Vision Speech and Signal Processing (CVSSP) for supporting me with a three-year bursary. I partly got benefited from Medical Research Council (MRC) [grant number MR/M023281/1] project “Modelling the progression of Chronic Kidney Disease (CKD)”. I sincerely thank CVSSP IT team for their help and invaluable support through out this program. I would like to extend my appreciation towards my family members for always encouraging me to strive for the best. Finally, I thank my friends for their support in the successful completion of this work.
List of Publications

Publications towards PhD

Journal Papers


Conference Proceedings


# Contents

<table>
<thead>
<tr>
<th>Contents</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Declaration</td>
<td>iii</td>
</tr>
<tr>
<td>Summary</td>
<td>v</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>vii</td>
</tr>
<tr>
<td>List of Publications</td>
<td>ix</td>
</tr>
<tr>
<td>Contents</td>
<td>x</td>
</tr>
<tr>
<td>List of Figures</td>
<td>xiv</td>
</tr>
<tr>
<td>List of Tables</td>
<td>xix</td>
</tr>
<tr>
<td>Acronyms</td>
<td>xxi</td>
</tr>
<tr>
<td>Symbols</td>
<td>xxiii</td>
</tr>
</tbody>
</table>

1 Introduction  
  1.1 Research Objectives ........................................... 2  
  1.2 Research Questions ........................................ 3  
  1.3 Contributions ............................................ 3  
  1.4 Structure of the Thesis .................................. 5  

2 Dynamic Mode Decomposition  
  2.1 Methodological Overview .................................... 7  
    2.1.1 Arnoldi Approach ...................................... 8  
    2.1.2 SVD Approach .......................................... 10  
  2.2 Related Work ............................................. 12  
    2.2.1 Recent Advancements ................................ 13  
  2.3 Advancements in this thesis .............................. 17  

3 Univariate Time Series Analysis  
  3.1 Introduction .............................................. 21
3.2 Methodology .......................................................... 22
  3.2.1 Embedding ....................................................... 22
  3.2.2 Dynamic Mode Decomposition .............................. 23
  3.2.3 Reconstruction and Forecasting with Selected DMD Triples . 23
  3.2.4 Hankelisation and Diagonal Averaging ..................... 24
3.3 Singular Spectrum Analysis ........................................ 24
  3.3.1 SSA Recurrent Forecasting ................................. 25
3.4 Dataset ................................................................. 25
  3.4.1 Real Data ....................................................... 25
  3.4.2 Simulated Data ............................................... 26
3.5 Experiments and Results .......................................... 26
  3.5.1 Effect of Window Length $L$ .............................. 26
  3.5.2 DMD for Extracting Trends ............................... 27
  3.5.3 DMD for Time Series Forecasting ......................... 29
  3.5.4 Comparison to SSA .......................................... 29
3.6 Conclusion and Discussions ..................................... 32

4 Quantifying Kidney Function: Movement Correction and Seg-
  mentation of DCE-MRI .............................................. 39
  4.1 Introduction ....................................................... 40
  4.1.1 Our approach ................................................ 43
  4.1.2 Contributions ................................................. 45
  4.1.3 Organisation .................................................. 46
  4.2 Methodology ....................................................... 46
    4.2.1 Dynamic Mode Decomposition (DMD) .................... 48
    4.2.2 Ordering Dynamic modes ................................. 48
    4.2.3 Reconstruction from DMD modes (R-DMD) ............. 48
    4.2.4 Windowed-DMD (W-DMD) ................................. 49
    4.2.5 Evaluation criteria ...................................... 50
  4.3 Dataset ............................................................... 51
  4.4 Experiments and Results ....................................... 52
    4.4.1 Selection of optimal $W$ (window length) ............ 53
    4.4.2 Performance of W-DMD ................................... 54
    4.4.3 Importance of W-DMD ..................................... 55
    4.4.4 Qualitative assessment of DMD ......................... 55
    4.4.5 Reconstruction using most significant modes .......... 58
    4.4.6 Effect of mode selection in the reconstruction: ...... 58
    4.4.7 Comparison with other registration methods .......... 59
    4.4.8 Segmentation ............................................... 60
    4.4.9 Evaluation of the Segmentation ....................... 63
  4.5 Summary and Conclusions ...................................... 66

5 2D Facial Spoof Detection ............................................. 69
  5.1 Introduction ....................................................... 69
5.1.1 Background on attacks directed at biometric systems . . . . . . . . 70
5.1.2 Motivation . . . . . . . . . . . . . . . . . . . . . . . . . . . . 71
5.1.3 Contributions . . . . . . . . . . . . . . . . . . . . . . . . . . . 72
5.1.4 Organisation . . . . . . . . . . . . . . . . . . . . . . . . . . . 73
5.2 Existing solutions . . . . . . . . . . . . . . . . . . . . . . . . . . . . 73
5.2.1 Key considerations in 2D facial counter-spoofing . . . . . . . 80
5.2.2 Characteristics of DMD . . . . . . . . . . . . . . . . . . . . . . 81
5.2.3 Competing methods . . . . . . . . . . . . . . . . . . . . . . . . 81
5.3 Methodology . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 83
5.3.1 Dynamic Mode Decomposition (DMD) . . . . . . . . . . . . . . 83
5.3.2 Principal Component Analysis (PCA) via snapshot approach 84
5.3.3 Local Binary Patterns (LBP) . . . . . . . . . . . . . . . . . . . 85
5.3.4 SVM classifier - Histogram Intersection Kernel . . . . . . . . . 85
5.3.5 Performance and Threshold Criteria . . . . . . . . . . . . . . . 86
5.4 Dataset . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 87
5.4.1 Print-attack Dataset . . . . . . . . . . . . . . . . . . . . . . . 87
5.4.2 Replay-attack Dataset . . . . . . . . . . . . . . . . . . . . . . 87
5.4.3 CASIA-FASD dataset . . . . . . . . . . . . . . . . . . . . . . . 88
5.5 Experiments & Results . . . . . . . . . . . . . . . . . . . . . . . . . . 88
5.5.1 Potential of DMD . . . . . . . . . . . . . . . . . . . . . . . . . 90
5.5.2 Selection of DMD modes . . . . . . . . . . . . . . . . . . . . . 91
5.5.3 Impact of LBP parameters on the classification performance . . . 93
5.5.4 Performance at the video level . . . . . . . . . . . . . . . . . . 94
5.5.5 Performance of window based approach . . . . . . . . . . . . . . . 95
5.5.6 Experiments on additional research questions . . . . . . . . . . 96
5.5.7 Proposed pipeline on CASIA-FASD Dataset . . . . . . . . . . . 98
5.5.8 Overall Summary . . . . . . . . . . . . . . . . . . . . . . . . . 98
5.6 Conclusions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 102

6 Finger Vein Spoof Detection: Micro Texture Descriptor 103
6.1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 103
6.1.1 Existing countermeasures in Finger vein spoofing . . . . . . . 104
6.1.2 Preliminary analysis: Window-DMD . . . . . . . . . . . . . . 105
6.2 Texture Methods . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 106
6.2.1 Local Binary Patterns (LBP) . . . . . . . . . . . . . . . . . . . 107
6.2.2 Discrete Wavelet Transform (DWT) . . . . . . . . . . . . . . 107
6.2.3 Discrete Cosine Transform (DCT) . . . . . . . . . . . . . . . 108
6.2.4 Histogram of Gradients (HoG) . . . . . . . . . . . . . . . . . . 108
6.2.5 Filter based methods . . . . . . . . . . . . . . . . . . . . . . . . 108
6.3 Methodology . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 109
6.3.1 Window Dynamic Mode Decomposition (W-DMD) . . . . . . 109
6.3.2 SVM-Minimum Intersection Kernel . . . . . . . . . . . . . . 112
6.4 Performance criteria . . . . . . . . . . . . . . . . . . . . . . . . . . . 112
6.4.1 EER based on F-ratio ........................................ 113
6.5 Dataset ....................................................... 114
6.6 Experiments and Results ..................................... 114
  6.6.1 Selection of the W-DMD component. ................. 114
  6.6.2 Comparison with state-of-the-art methods. .......... 115
  6.6.3 Conclusions ............................................. 117

7 Colour Background Initialisation ............................. 119
  7.1 Scene Background Initialisation ........................ 119
  7.2 Data, Methodology & Implementation ................... 120
  7.3 Results & Discussions .................................... 125
  7.4 Conclusions ............................................... 126

8 Conclusion and Future Work ................................ 131
  8.1 Main Achievements ........................................ 133
  8.2 Future Work ............................................... 134

References ....................................................... 137
# List of Figures

2.1 LES of 2D jet with pressure data showing a close passive scalar difference across the interface between two flows. A total of 133 snapshots were considered with a time interval of 0.006 seconds.  

2.2 Comparison of (a) PCA mode and (b) DMD mode for close passive scalar field.  

2.3 Visual comparison between a noisy and redundant DMD spatial mode.  

2.4 (a) 5 randomly selected images from a Dynamic Contrast Enhanced Magnetic Resonance Imaging (DCE-MRI) sequence of a healthy volunteer. The middle image shows the peak stage of renal perfusion inside the kidneys. Quantification of kidney function is performed by calculating the mean of pixel intensity values inside the kidneys. (b) image showing their corresponding dynamic mode-2. (c) image showing the binarized version of that dynamic mode. (d) Kidney template with the largest area of connected pixels.  

2.5 Four frames from a valid access, controlled scenario video clips showing the eye-blinking and the lip movements. The bottom row shows their corresponding first dynamic mode image. The images in the top two rows show the corresponding extracted frames from each of the two videos containing a valid-access request in a controlled scenario. Since the person in these videos was speaking and blinking his eyes inevitably from time to time, one can observe that the corresponding first DMD mode highlights dynamic changes in the lip movements and eyes blinks. However, eye blinking is less obvious for the second video.  

2.6 (a) Exemplar frame from ‘HighwayI’ sequence of the SBI dataset where the background is occluded with a number of foreground objects. (b) Background models generated by DMD_CT.  

2.7 (a) Reconstruction using DMD modes whose eigenvalues are closer to origin, revealing trends in the data. (b) Forecasting trends in the data using DMD.  

3.1 Effect of window length $L$ over the reconstruction performance.
3.2 (a) DMD eigenvalue plot, showing eigenvalues \{18, 19, 20, 21, 22\} that are vertically closer to the X-axis. (b) Corresponding absolute frequencies. 27

3.3 Trend analysis of the time series, revealing two trends (i) trend-1: linear trend and (ii) trend-2: periodic trend. 28

3.4 (a) DMD triples vs reconstruction error. (b) A complete reconstruction. 28

3.5 (a) Trend forecast. (b) A complete forecast of the timeseries. 29

3.6 MSE while reconstructing the original time series in the presence of noise at \(-2dB\), using first (a) DMD triples, (b) SVD triples. 30

3.7 Effect of noise in reconstruction at various SNRs across 500 iterations. 30

3.8 Reconstruction and forecasting at \(-2dB\). 32

3.9 Time series analysis on Airline passengers data: (a) trend analysis, (b) periodicity analysis, (c) complete reconstruction, (d) trend forecast, (e) periodicity forecast and (f) a forecast of the timeseries. 34

3.10 Time series analysis on BJ Sales data: (a) trend analysis, (b) periodicity analysis, (c) complete reconstruction, (d) trend forecast, (e) periodicity forecast and (f) a forecast of the timeseries. DMD gives better performance compared to SSA in (f) for forecasting. 35

3.11 Time series analysis on UK gas data: (a) trend analysis, (b) periodicity analysis, (c) complete reconstruction, (d) trend forecast, (e) periodicity forecast and (f) a forecast of the timeseries. DMD gives better performance compared to SSA in (f) for forecasting. 36

4.1 DCE-MRI image with anatomical parts. 41

4.2 (Top) 5 frames at time = \{30, 50, 74, 82, 100\}s selected from a DCE-MRI sequence of a healthy volunteer. The yellow and red reference lines show the alignment of the kidney and liver regions. The first image shows the peak stage of contrast agent inside the kidney region. The regions of kidney and liver clearly depict the translation of movements in the vertical direction, arising due to the patient movements. (Bottom) The yellow and red reference lines clearly showing the proper alignment of the kidney and liver regions after processing with WR-DMD. The video based results can be viewed at https://youtu.be/TWq34TFGncU and https://youtu.be/UT7f4ch4H-I. 42
4.3 Flow chart showing the steps involved in the methodological framework. First, a DCE-MRI sequence consisting of \( N \) images is processed using the W-DMD algorithm in order to output each \( N - 2 \) W-DMD components C1 and C2. At this stage the W-DMD(C1) produces the low rank images and W-DMD(C2) produces sparse images. W-DMD(C1) is given as an input to DMD which produces \( N - 3 \) DMD modes. On one hand, the first 3 DMD modes are then selected for reconstructing the motion stabilised image sequence. On the other hand, we select a dynamic image corresponding to mode 2, where, thresholding is performed and converted to binary image. After binarisation, the largest area with connected pixels is selected as a ROI template. Finally, the produced template is projected onto motion corrected DCE-MRI data to compute the kidney function.

4.4 Methodological pipeline showing the working mechanism of W-DMD. DMD runs over the window containing first three images in the sequence, obtaining two dynamic modes. The first dynamic mode 'c1' capturing the low rank image across the window and second dynamic mode c2 capturing the sparse representation. The next step exclude the first image and consider images \( \{2, 3, 4\} \), followed by \( \{3, 4, 5\} \) and \( \{4, 5, 6\} \) producing c1 and c2 components. Finally, all of the c1s and c2s across all the windows are concatenated to obtain W-DMD component-1 (W-DMD (C1)) and W-DMD component-2 (W-DMD (C2)).

4.5 Exemplars of Dynamic MR images from 10 healthy volunteers’ kidney slice produced by DCE-MRI sequence considered as 10 different datasets in this study. The images here show the central kidney slice at time 120s aortic peak enhancement after the contrast agent is injected. The yellow boundary on the kidneys are a results of manual delineation from a human expert. The mean intensity values are calculated in this region across the time producing time-intensity plots.

4.6 Motion magnitude with respect to first image in the sequence across the dataset-1.

4.7 (Top) 5 images from dataset-1 W-DMD(C1) at time = \( \{30, 50, 74, 82, 100\} \)s showing the low-rank images. The first image shows the peak stage of contrast agent inside the kidney region. (Bottom) Corresponding images from the W-DMD(C2) showing their sparse representation.

4.8 Time intensity curves across 10 datasets (sequentially from left to right).

4.9 (Top) The top 10 most significant DMD modes on W-DMD(C1) from dataset-1. (Bottom) bottom 10 least significant DMD modes on W-DMD(C1).

4.10 A smaller mean motion magnitude indicates a more stable sequence. Here M is the number of modes that were used in the reconstruction process. As the number of modes increase, the mean motion magnitude also increases.
LIST OF FIGURES

4.11 Mean motion magnitude across 10 datasets calculated using block matching algorithm utilising WR-DMD. 59

4.12 Comparison with intensity based image registration methods. Each box-plot has 10 values corresponding to 10 datasets. A smaller value in the X-axis indicates greater smoothness; so smaller is better. 60

4.13 DMD modes 1, 2, 4, 5 and 83 from a health volunteer’s data (dataset-1). (a) DMD mode-1 showing the background image captured from the dataset. (b) DMD mode-2 showing predominantly the kidney region. (c) DMD mode-4 showing predominantly the spleen region. (d) DMD mode-5 showing predominantly the liver region. (e) DMD mode-83 showing the noise component from the dataset. 61

4.14 (Top) Image showing the kidney regions in Dynamic mode-2 across the 10 datasets used in this study. (Middle) Images showing the binarisation effect on dynamic mode-2 across the datasets. (Bottom) Images showing the kidney template selected by the largest area of connected components. 62

4.15 Kidney function produced by blind bounding-box region, DMD framework, three domain experts, and ground-truths for datasets 1-10. 65

5.1 Categorisation of specific methodologies existing in the literature. 74

5.2 Flow chart showing the steps involved in the methodological pipeline. 83

5.3 Examples of the genuine and spoof attacks in controlled and adverse scenarios. The top row shows original images of valid access, photo and video attacks (left to right respectively). The middle row shows their corresponding first DMD mode and the bottom row shows their corresponding first PCA mode. 91

5.4 Eigenvalue plot of DMD for one of the video clips in the replay-attack dataset. Each dot in the circle is a complex eigenvalue. If there are N images in the video, there are N − 1 complex eigenvalues. 92

5.5 DMD modes from 1 to 6 (with HTER in % for test set on replay-attack dataset shown in brackets) showing the dynamics within the video frame sequence for valid access (top row), photo (middle row) and video attacks (bottom row). 92

5.6 Impact of block-size and radius of LBP on the HTER across the print-attack dataset (left) and replay-attack dataset (right). 94

5.7 HTER of the top X×10-frames evaluated on the development and test sets from print-attack dataset (left) and replay-attack dataset (right), here X ranges from 1, 2, . . . , 240. 96

5.8 Examples of the valid access and spoof attacks in controlled and adverse scenario for cropped face regions. The top row shows original images of valid access, photo and video attacks (left to right respectively). The middle row shows their corresponding first DMD mode and bottom row shows their corresponding first PCA mode. 97
5.9 Examples of the valid and spoof attacks in CASIA-FASD (High Resolution HR). The top row shows original images of valid access, photo, cut photo and video attacks (left to right respectively). The middle row shows their corresponding first DMD mode and bottom row shows their corresponding first PCA mode. .......................... 99

5.10 Overall summary of the results in HTER (%) for the proposed pipelines. Here, RA denotes Replay-Attack, PA: Print-Attack and CF: CASIA-FASD datasets. (Note: we have not considered the entire video for analysis in the case of CASIA-FASD dataset). .......................... 101

6.1 Top row shows the raw images for valid access and spoof from the same person. Middle row shows the W-DMD component-1 of the raw image and bottom row shows the W-DMD component-2 of the raw image. .......................... 106

6.2 Top row shows the raw images for valid access and spoof from the same person. Middle row shows W-DMD component-1 of the raw image and bottom row shows W-DMD component-2 of the raw image. .......................... 106

6.3 Texture visualisation of full and cropped finger vein images using filter based methods and DWT. .......................... 109

6.4 Flow chart showing the steps involved in the methodological pipeline. .......................... 110

6.5 Methodological pipeline showing the working mechanism of W-DMD. .......................... 111

6.6 Dataset divided into full and cropped finger vein images for valid and spoof samples. .......................... 114

6.7 Comparison of (a) F-ratio and (b) EER (F) on the state-of-the-art methods for full and cropped finger vein images. .......................... 116

7.1 Example frames from the seven sequences of the SBI dataset (first row) and corresponding ground-truth (GT) (second row). .......................... 121

7.2 Flow chart showing the steps involved in the methodological framework. A colour image sequence consisting of $N$ images in R, G and B channels are vertically concatenated forming a data matrix $X_{RGB}$ as shown in the Equation 7.1. DMD is applied to this matrix to obtain $N - 1$ dynamic mode images. Frequencies are calculated on the dynamic eigenvalues. The mode with the frequency $\approx 0$ is then selected as the background initialised image. The three channels are then normalised and combined to obtain colour background initialised image. Finally, the original colour is transferred from the original images to the obtained background image. .......................... 122

7.3 Qualitative comparison of DMD and DMD_CT with the benchmarked methods. .......................... 128
List of Tables

1.1 Theoretical advancements and novel applications of DMD in the fields of signal processing and computer vision that are addressed in this thesis. 5

2.1 Algorithmic advancements of DMD. 20

3.1 Mean and standard deviation of MSE scores across 500 iterations at various SNRs. 31

3.2 The parameters that have been used for analysing the real data using DMD and SSA. Here parameters, T and L represent the total length of the time series and length of the window considered for the analysis respectively. ‘Rec’ denotes reconstruction and ‘For’ denotes forecasting. MSE denotes the mean square error between the reconstructed/forecast time series and the original timeseries. 37

4.1 Comparison of proposed methods with other movement correction approaches that are based on matrix decomposition methods. 44

4.2 Evaluation results for the DMD framework with respect to the three ground-truths \( G_i \), \( \forall i \in \{1, 2, 3\} \). 63

4.3 Jaccard similarity coefficients for kidney segmentation achieved by the DMD framework, blind bounding-box region and three domain experts with respect to the ground-truth. 64

4.4 Mean Square Error (MSE) based evaluation of the kidney functions for DMD, human experts and the bounding box. 64

5.1 Categorisation of existing facial counter-spoofing methods: Part-1. 75

5.2 Categorisation of existing facial counter-spoofing methods: Part-2. 76

5.3 Categorisation of existing facial counter-spoofing methods: Competing methods: Part-3. 77

5.4 Comparison of the DMD with Pinto et al.’s visual rhythm and Anjos et al.’s method in terms of a number of attributes. 82

5.5 Number of videos in each set according to the amount of hand-based and fixed-based attacks. 88
5.6 Comparison of the proposed pipeline with the competition methods. . . 94
5.7 HTER on print-attack and replay-attack datasets . . . . . . . . . . . 95
5.8 Results in HTER % on replay and print-attack datasets . . . . . . . 97
5.9 Classification performance in terms of HTER (%) on the on CASIA-FASD 99
5.10 Comparison of HTER (%) on test sets for the proposed pipeline with respect to the current state of the art. . . . . . . . . . . . . . . . . . . . 100

6.1 categorisation of the texture analysis methods based on spatial or frequency and local or global approach. . . . . . . . . . . . . . . . . . . . 107
6.2 Table showing the size of the resultant feature vector for each of the methods used in this study. . . . . . . . . . . . . . . . . . . . . . . . . . . . 112
6.3 Number of samples divided into protocols in the dataset. . . . . . . . 114
6.4 Performance of the W-DMD components on cropped and full image data.115
6.5 Performance of the state of the art methods in comparison with the W-DMD method for cropped and full image data. . . . . . . . 116

7.1 Resolution of images in the sequences adopted for evaluation. . . . . 121
7.2 Quantitative comparison of CQM values for DMD & DMD_CT with the benchmark methods. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 129
**Acronyms**

- BSIF  Binarized Statistical Image Features
- CB  Computed Background
- CFD  Computational Fluid Dynamics
- CQM  Colour Quality Measure
- dB  Decibels
- DCE-MRI  Dynamic Contrast Enhanced Magnetic Resonance Imaging
- DCT  Discrete Cosine Transform
- DMD  Dynamic Mode Decomposition
- DMD\_CT  Dynamic Mode Decomposition with Colour Transfer
- DoG  Difference of Gaussian
- DWT  Discrete Wavelet Transform
- EER  Equal Error Rate
- EER (F)  Equal Error Rate based on F-ratio
- FAR  False Acceptance Rate
- FRR  False Rejection Rate
- FV  Finger Vein
- GT  Ground-Truth
- HoG  Histogram of Gradients
- HR  High Resolution
- HTER  Half Total Error Rate
ICA Independent Component Analysis
KH KelvinHelmholtz
KLT Karhunen-Loeve Transform
LBP Local Binary Patterns
LPQ Local Phase Quantization
MSE Mean Square Error
MSS Monogenic Scale Space
PCA Principal Component Analysis
PPCR Progressive Principal Component Registration
PSNR Peak Signal to Noise Ratio
RDDR Robust Data Decomposition Registration
RMSE Root Mean Square Error
ROI Region of Interest
RPCA Robust Principal Component Analysis
RSVD Randomised Singular Value Decomposition
R-DMD Reconstruction Dynamic Mode Decomposition
SBI Scene Background Initialisation
SC-SOBS Spatially Coherent Self-Organizing Background Subtraction
SNR Signal to Noise Ratio
STD Standard Deviation Filter
STICA Spatio-temporal Independent Component Analysis
SSA Singular Spectrum Analysis
SVD Singular Value Decomposition
SVM Support Vector Machines
WLD Weber Local Descriptor
YUV Luminance and Chrominance
Symbols

\( \bar{x}_r \) The \( r^{th} \) image frame in a sequence \( X \), whose size is \( mn \times 1 \).

\( m \) The total number of rows in an image frame.

\( n \) The total number of columns in an image frame.

\( X \) The data matrix of size \( mn \times N \) consisting of all the images in the sequence.

\( X^{RGB} \) The colour image sequence consisting of \( N \) images in R, G and B channels are vertically concatenated forming a data matrix.

\( N \) The total number of image frames.

\( A \) The linear map, from one video frame to another in the sequence.

\( r \) the vector of residuals that accounts for behaviours that cannot be described completely by \( A \).

\( P_2 \) The data matrix consisting of image frames from \([ \bar{x}_2, \bar{x}_3, \cdots, \bar{x}_N ] \).

\( P_1 \) The data matrix consisting of image frames from \([ \bar{x}_1, \bar{x}_2, \cdots, \bar{x}_{N-1} ] \).

\( K_N(A, \bar{x}_1) \) The span of krylov subspace, where \( K_N(A, \bar{x}_1) = \text{span}\{ \bar{x}_1, A\bar{x}_1, A^2\bar{x}_1, \cdots, A^{N-1}\bar{x}_1 \} \)

\( H \) The companion matrix shifts from image frames 1 through \( N-1 \) and approximates the last frame \( N \) by linearly combining the previous \( N-1 \) frames, i.e.,

\( P_2 = c_0\bar{x}_1 + \cdots + c_N\bar{x}_{N-1} = \{ \bar{x}_1, \bar{x}_2, \bar{x}_3, \cdots, \bar{x}_{N-1} \}c \).

\( U \) The left singular vectors obtained when SVD is performed on \( P_1 \).

\( \Sigma \) The singular values obtained when SVD is performed on \( P_1 \).

\( V^* \) The right singular vectors obtained when SVD is performed on \( P_1 \).

\( L \) The lower triangular matrix from LU decomposition.

\( U \) The upper triangular matrix from LU decomposition.

\( L^+ \) The pseudo inverse of \( L \) from the LU-decomposition of \( P_1 \).
\( d \) The total number of singular values.
\( \Sigma \) The \( i^{th} \) singular value.
\( \tilde{H} \) The reduced order model of the full system \( A \).
\( \omega \) The eigenvectors of \( \tilde{H} \).
\( \sigma \) The eigenvalues of \( \tilde{H} \).
\( U^* \) The conjugate transpose of \( U \).
\( V^* \) The conjugate transpose of \( V \).
\( \Sigma^{-1} \) The inverse of the singular values \( \Sigma \).
\( \Psi \) The dynamic modes or the DMD modes.
\( \sigma \) The complex DMD eigenvalues containing growth/decay rates and frequencies of the corresponding DMD modes \( \Psi \).
\( \sigma_j \) The diagonal elements of \( \sigma \).
\( \mathcal{V} \) Vandermonde matrix representing the temporal evolution for \( \sigma_j \).
\( \mathcal{V}(L) \) The standard Vandermonde matrix for reconstruction but if \( f > L \), is used for forecasting.
\( \mu_j \) The DMD modes with frequencies \( \mu_j = \frac{\ln(\sigma_j)}{\delta t} \).
\( \delta t \) The lag between the vectors.
\( \mathcal{V}(f) \) The vandermonde matrix with \( \sigma^{f-1} \).
\( F \) The number of future vectors that are to be forecast.
\( b \) The initial amplitudes for the dynamic modes.
\( \circ \) The point-wise (element-wise) multiplication of \( b \) with every column of \( \mathcal{V}(f) \).
\( T \) The length of the univariate timeseries \( x \).
\( X \) The multidimensional representation of \( x \) into \([\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_L]\), of size \( K \times L \).
\( \tilde{x}_n = [x_n, x_{n+1}, \ldots, x_{n+K-1}]^T \in \mathbb{R}^K \) is called a \( K \)-lagged vector.
\( L \) The length of multidimensional window.
\( K \) The number of segments i.e., \( K = T - L + 1 \).
\( \tilde{x} \) The reconstructed and Hankelized univariate timeseries.
\( k \) The index of the time series.
\( \Re(\cdot) \) The real part of \( \tilde{x}_k \).
num(.) The number of combinations of \((m, n)\), \(m + n = k + 1\).

\(G_i\) The ground-truth functional segmentation results obtained from \(e_i\).

\(e_i\) The results obtained from the human experts \(\forall i \in \{1, 2, 3\}\).

\(D\) The functional segmentation results of the DMD framework.

\(D_i\) The evaluation \(D\) for the DMD framework is given by the average of \(D_i \forall i \in \{1, 2, 3\}\).

\(E_i\) The evaluation criteria for the human experts given by \(\frac{|e_i \cap G_i|}{|e_i \cup G_i|} \forall i \in \{1, 2, 3\}\).

\(\mathcal{T}\) Domain of an SVM output.

\(\mathcal{Y}\) Output scores of SVM classifier i.e., probability density of \(\mathcal{T}\).

\(\mathcal{Y}_1\) The comparison with \(\mathcal{Y}\) is due to a positive class (valid access).

\(\mathcal{Y}_0\) The comparison with \(\mathcal{Y}\) is due to a negative class (spoof/attack).
Chapter 1

Introduction

Spatio-temporal data are collected across time as well as space i.e., data are collected from different locations at different intervals of time. Common examples include air pollution data collected from different parts of the UK at various intervals of time and video data, where pixel intensity values change over time as the video progresses. Analysis of such data involves modelling the time evolution of the response variables over the spatial domain, especially in order to detect meaningful patterns as well as to make useful predictions. However, extracting relevant meaningful sub-components or modes from spatio-temporal data poses considerable challenges. In particular, in the field of computational fluid dynamics (CFD) gaining a deep and accurate understanding of complex fluid flow behaviour over time requires a form of reduced order models [1] or mode decomposition techniques [2]. The mode decomposition of the form that is dealt with in this thesis is Dynamic Mode Decomposition (DMD) [3], a mathematical method specifically developed to extract the relevant modes from empirical data generated by spatio-temporal non-linear complex fluid flows. These modes provide a reduced-order spatial representation of the complex flow behaviour over the time, called coherent flow structures. The extraction of these coherent flow structures is of great importance in CFD, especially in applications involving internal combustion engines (ICEs), where extreme fluid velocities causes chaotic and stochastic fluid property changes over space and time\(^1\). Modelling these stochastic fluid changes sheds light on the fundamentals of the combustion process, which might enable researchers to design better vehicles and reduce exhaust gas emissions, which is today one of the major environmental hazards caused by the oil based energy and transport sectors [4].

Since its introduction in CFD in 2008 [5], DMD has gained significant applications in various fields due to its modelling of spatio-temporal data in a data driven fashion. For example, very recently in computer vision DMD has been used ro-

\(^1\)A video showing the chaotic and stochastic fluid changes in an ICE can be found at https://youtu.be/1o3PpYvZVa8
bustly to separate grey scale video frames into a background model and multiple foreground objects [6]. In robotics and neuroscience, involving signal processing, DMD has been used to estimate perturbation in human robot interactions [7] and to extract coherent patterns in large-scale neural recordings [8]. In order to expand DMD to other applications, this thesis will advance the original formulation so that it can be used to solve novel problems in the fields of computer vision and signal processing.

1.1 Research Objectives

DMD is a powerful data driven method that can potentially be applied to a plethora of research problems in the fields of computer vision and signal processing. The overall aim of this thesis is to develop new variants and novel applications utilising DMD, with the specific objectives summarised as follows:

1. **To design a method of DMD that allows processing of univariate time series data.** DMD is fundamentally a multidimensional method, and has never been applied to a univariate time series. The objective here is to design DMD through a four stage Hankelisation approach, thereby extending DMD for modelling trends, noise and forecasting univariate time-series.

2. **To introduce a new variant of DMD, based on a sliding window protocol (W-DMD), which could be used for analysing both images and image sequences (videos).** Since DMD is used for analysing image sequences, this new variant (W-DMD) based on a sliding window protocol can be used to analyse an individual image, thus enabling DMD to serve as a micro-texture feature descriptor of that image. While the same variant used in the video sequences can be used for removing periodic motion components.

3. **To introduce a new method for motion stabilisation utilising DMD.** The noise components from the image sequences are captured in the least significant DMD modes. These modes, essentially, contain motion components. Therefore reconstruction (R-DMD) with the most significant modes robustly can stabilise complex movements in a dynamic medical image sequence.

4. **To showcase the potential of DMD in extracting coherent structures.** While DMD is well established for extracting coherent structures in the field of CFD, this thesis also shows that DMD can indeed extract these coherent structures (features) in the fields of biometrics and medical imaging. For example, in biometrics these coherent structures include facial dynamics such as lip movements, eye blinks and other attack specific cues such as light reflection properties, planar and moire effects, that can be used to discriminate between a valid and a spoofed facial video. In medical imaging, the coherent structures can be used for automatic segmentation of a kidney region.
from Dynamic Contrast Enhanced Magnetic Resonance Imaging (DCE-MRI) sequences.

5. To modify the existing DMD techniques to allow processing of coloured video sequences. The DMD method introduced in computer vision essentially converts coloured images into grey scale for processing. Therefore, the objective here is to modify the standard DMD by vertically concatenating the RGB channels of a coloured image sequence instead of converting the image into grey scale.

1.2 Research Questions

The following research questions would help achieve the aforementioned research objectives:

1. Can DMD process univariate time series data being a multi dimensional method?

2. Can DMD capture background models in the colour domain?

3. Can DMD pick up coherent structures from medical images and biometric videos as it does from CFD images?

4. Can DMD be applied to image domain?

1.3 Contributions

The aforementioned research objectives have led to several novel contributions as presented in Table 1.1. Each contribution is briefly discussed below:

1. **Introducing a Hankelisation approach for DMD, thus extending DMD to modelling and forecasting univariate time-series.** DMD applications consider a data matrix as an input. For instance, to analyse an image sequence (a video) using DMD the input data is converted into column vectors, each of which representing a single frame of the video. However, being a fundamentally multidimensional method, DMD has never been applied to a univariate time series. Therefore, the novel contribution here is to introduce the method of DMD through a Hankelisation approach for decomposing a univariate time series into a number of interpretable elements in terms of noise, trend and harmonics. Moreover, time series forecasting is also shown using the proposed methodology.

2. **Introduction of DMD variants (windowed (W-DMD)) with reconstructive properties (R-DMD) that enable robust stabilisation of an image sequence, with application to DCE-MRI involving kidneys.**
Furthermore, enabling functional segmentation utilising DMD. Quantifying kidney function in DCE-MRI is often difficult due to complex organ movements and segmentation of the kidney region of interest (ROI). This problem is tackled in this thesis using a three fold approach: (i) by performing movement correction using novel windowed and reconstruction variants of DMD (WR-DMD) specifically proposed to stabilise the complex movements in DCE-MRI, and (ii) by obtaining segmentation of the kidney ROI using a novel pipeline consisting of DMD+thresholding+blob analysis. Kidney function is thus quantified as a mean of pixel intensity values inside the segmented kidney ROI.

3. Embedding DMD as a feature extraction algorithm on a video sequence for classification, with application to detecting spoofed or replayed video attacks in 2D face recognition. The first contribution here is to introduce the use of DMD for countering spoof attacks in face recognition. Since DMD has not been used with classification of video sequences, this thesis proposes a DMD based classification pipeline. The second contribution is related to solving the DMD. While the original DMD uses QR-decomposition and singular value decomposition (SVD) methods, this thesis uses LU decomposition [9]. Second, in the modified DMD, the absolute value of the complex modes are considered when rendering the “DMD image”. Third, a novel way to select the DMD modes is presented through a phase angle criterion after discarding the background mode. In short, this part of the thesis makes DMD practical for classification and demonstrates its effectiveness by separating valid access videos from spoofed ones.

4. Extending DMD as a basis for micro-texture descriptor for classification, with application to detecting live from spoofed finger vein images. In general, DMD is used to analyse a set of image sequences. However, in this thesis, an approach, W-DMD as a micro texture descriptor, is proposed exclusively for analysing an image in order to detect spoofed finger vein images.

5. Extending DMD to process colour images directly for the purpose of background modelling. The DMD method that is introduced in [6] for obtaining the background model essentially requires the conversion of colour images to grey scale ones. However, it is sometimes necessary to have the scene background in the colour domain, particularly for applications that include computational photography and video inpaintings. Therefore, the key contribution in this part of the thesis is in modifying the DMD algorithm to obtain a coloured background model, thereby avoiding the conversion of colour images into grey scale ones.

The novel problems and theoretical advancements that have been addressed in this thesis are presented in Table 1.1.
<table>
<thead>
<tr>
<th>Novel Application</th>
<th>Field</th>
<th>Theoretical Advancement</th>
<th>Description</th>
<th>Machine learning Task</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Univariate Time Series Analysis</strong></td>
<td>Signal Processing</td>
<td>DMD with an Hankel matrix approach. Four stage pipeline: Embedding + DMD + Reconstruction/Forecasting+ Anti Diagonal Averaging.</td>
<td>Trend extraction, noise cancellation and forecasting</td>
<td>Regression</td>
</tr>
<tr>
<td><strong>Quantifying Kidney Function in DCE-MRI (Medical Imaging)</strong></td>
<td>Computer Vision</td>
<td>Windowed DMD (W-DMD), Reconstruction DMD (R-DMD), WR-DMD, ordering modes by phase angles, absolute DMD modes+ thresholding+ blob analysis.</td>
<td>Movement correction and functional segmentation of kidneys in DCE-MRI</td>
<td>Clustering</td>
</tr>
<tr>
<td><strong>2D facial Spoof Detection (Biometrics)</strong></td>
<td>Computer Vision</td>
<td>DMD+ Local Binary Patterns (LBP) + Support Vector Machines (SVM), LU decomposition based Arnoldi DMD</td>
<td>DMD based classification pipeline for differentiating valid vs spoof 2D faces</td>
<td>Classification</td>
</tr>
<tr>
<td><strong>Finger Vein Spoof Detection (Biometrics)</strong></td>
<td>Computer Vision</td>
<td>W-DMD for images and DMD+SVM pipeline</td>
<td>Image micro-texture descriptor</td>
<td>Classification</td>
</tr>
<tr>
<td><strong>Scene background model initialisation (Background Modelling)</strong></td>
<td>Computer Vision</td>
<td>Vertical concatenation of RGB channels of a coloured image sequence as an input to DMD</td>
<td>Allow processing for coloured images</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 1.1: Theoretical advancements and novel applications of DMD in the fields of signal processing and computer vision that are addressed in this thesis.

### 1.4 Structure of the Thesis

The remainder of the thesis is organised as follows:

- Chapter 2 presents the methodological overview of DMD and showcases two approaches, namely: i) Arnoldi approach and ii) singular value decomposition (SVD) approach for obtaining the dynamic modes. This chapter also presents a survey on recent theoretical advancements in the DMD literature with a focus on various applications.

- Chapter 3 for the first time introduces the method of DMD for analysing univariate time series by forecasting as well as extracting trends and frequencies. This chapter both theoretically as well as empirically presents similarities and differences with singular spectrum analysis (SSA), which is another popular technique in the field of signal processing, especially for analysing univariate time series data.
Chapter 4 deals with the problem of quantifying kidney function in dynamic medical image sequences. This chapter offers a two fold solution: (i) the movement correction problem is solved through the introduction of windowed (W-DMD) and reconstruction (R-DMD) variants of DMD, and ii) the intermediate mode-2 obtained in the WR-DMD process is coupled with thresholding and connected component analysis to obtain the functional segmentation of the kidney region. Using this template the kidney function is thus calculated. This chapter presents a complete automated solution for quantifying kidney function in dynamic medical image sequences.

Chapter 5 introduces the problem of detecting spoofed 2D faces in the field of biometrics. This chapter demonstrates the potential of DMD as a pre-processing technique in highlighting lip movements, facial textures and eye blinks from facial authentication videos. These highlighted features, when coupled with a texture-based descriptor and a discriminative classifier can easily differentiate between a valid and a spoofed facial video. Furthermore, the existing state of the art solutions that tackle 2D facial spoofing problem are showcased and compared with the characteristics of the DMD method.

Chapter 6 introduces the problem of detecting spoofed finger vein images in the field of biometrics. The W-DMD method, which has been introduced for the movement correction problem in chapter 5 is reintroduced here, specifically for the image domain. In this chapter, W-DMD is shown to extract local variations as low rank representations inside a finger vein image, capturing light reflections, illuminations and planar effects that are typically different for valid access and spoofed images.

Chapter 7 presents a technique that enables DMD to obtain a background model in the colour domain. This chapter demonstrates the effectiveness of the proposed technique on a publicly available Scene Background Initialisation (SBI) dataset. The results in this chapter both qualitatively and quantitatively show that DMD can successfully obtain a coloured background model.

Chapter 8 provides the concluding remarks of this thesis, and discusses promising directions for future research.
Chapter 2

Dynamic Mode Decomposition

This chapter is divided into three major parts. The first part of this chapter introduces the method of DMD and presents two approaches to solving it. On the other hand, the second part presents a survey of related work in the field of CFD as well as recent advancements in the fields of computer vision and signal processing, viz., (i) sparsity promoting DMD for (a) subtle emotion detection and (b) estimating perturbation in a robot; (ii) DMD and (iii) randomised DMD for background/foreground extraction, and (iv) multi resolution DMD for background subtraction and object tracking. Finally, a brief motivation on the advancements that are addressed in this thesis are presented.

2.1 Methodological Overview

Dynamic mode decomposition (DMD) is a new matrix decomposition algorithm developed by Prof. Peter Schmid in the year 2008 [5]. DMD takes in a time series of data and computes a set of dynamic modes, each of which are associated with a complex eigenvalue. The real part of the complex eigenvalue represents a growth/decay frequency of the dynamic mode. Similarly, oscillation frequencies are given by the imaginary part of the complex eigenvalues. DMD is closely related to spectral analysis of Koopman operator which makes DMD model a nonlinear system as a linear combination of the modes whose dynamics are governed by the eigenvalues. The Koopman operator is a linear but infinite-dimensional operator whose modes and eigenvalues capture the evolution of nonlinear dynamical systems [10]. Therefore a nonlinear system could be described by superposition of dynamic modes whose dynamics are governed by eigenvalues. Therefore DMD differs from dimensionality reduction methods such as Principal Component Analysis (PCA), which although computes orthogonal modes, lacks temporal representation.

Let \( \mathbf{x}_r \) be the \( r^{th} \) video frame or an image in a sequence, whose size is \( m \times n \). This
video frame $\bar{x}$ is converted to $mn \times 1$ column vector, resulting in the construction of a data matrix $X$ of size $mn \times N$ for $N$ video frames.

$$X = [\bar{x}_1, \bar{x}_2, \bar{x}_3, \cdots, \bar{x}_N] = \left( \begin{array}{cccc} x_1^1 & x_2^1 & \cdots & x_N^1 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{mn} & x_2^{mn} & \cdots & x_N^{mn} \end{array} \right). \quad (2.1)$$

A linear mapping $A$ (Koopman operator) from one video frame to another in the sequence is assumed as these frames are sampled at equal intervals.

$$X = [\bar{x}_1, A\bar{x}_1, A^2\bar{x}_1, \cdots, A^{N-1}\bar{x}_1].$$

$$[\bar{x}_2, \bar{x}_3, \cdots, \bar{x}_N] = A[\bar{x}_1, \bar{x}_2, \cdots, \bar{x}_{N-1}]. \quad (2.2)$$

Here, $r$ is the vector of residuals that accounts for behaviours that cannot be described completely by $A$, $e_{N-1} = \{0, 0, 1\} \in \mathbb{R}^{N-1}$, $P_2 = [\bar{x}_2, \bar{x}_3, \cdots, \bar{x}_N]$ and $P_1 = [\bar{x}_1, \bar{x}_2, \cdots, \bar{x}_{N-1}]$. The system $A$ is unknown and it captures the overall dynamics within the video or an image sequence in terms of the eigenvalues and eigenvectors of $A$, which are referred to as the DMD eigenvalues and DMD modes respectively.

Computing an eigen solution for $A$ is computationally expensive since, $A$ is large ($mn \times mn$). From the literature there are two approaches for obtaining these eigenvalues and modes. The first is Arnoldi based approach, which is useful for theoretical analysis due to its connection with Krylov subspace methods (Eq.2.2) [11–13]. The second is a singular value decomposition (SVD) based approach that is more robust to noise in the data and to numerical errors.

2.1.1 Arnoldi Approach

Methods based on Krylov subspaces [12,13] could be efficiently used for solving $A$, which is computationally expensive when solved normally due to its size ($mn \times mn$). The columns in the Krylov subspace are non-orthogonal but orthogonality could be achieved using Arnoldi iteration.

Krylov Subspaces

Krylov subspace was first introduced by the Russian applied mathematician and naval engineer Alexei Krylov in 1931 [11]. The methods based on this approach are often used for obtaining eigenvalues of large sparse matrices or for solving large systems of linear and of non-linear equations [12,13]. These methods are favoured because of their simplicity, generality and computational efficiency [14].

Given a square matrix $A$ and a non-zero vector $\bar{x}_1$, the subspace defined by

$$K_N(A, \bar{x}_1) = span\{\bar{x}_1, A\bar{x}_1, A^2\bar{x}_1, A^3\bar{x}_1, \cdots, A^{N-1}\bar{x}_1\}.$$
is referred to as the $N^{th}$ Krylov subspace of $(A, \bar{x}_1)$ and is denoted by $K_r(A, \bar{x}_1)$. Here, $A$ is an $mn \times mn$ matrix and $\bar{x}_1$ a $mn \times 1$ vector.

**Arnoldi Iteration**

Solving eigenvalues of a large (generally sparse) matrix $A$ of size $mn \times mn$ can be achieved using power iteration methods such as Arnoldi iteration. Starting with an initial random vector $\bar{x}_1$, the method calculates $A\bar{x}_1, A^2\bar{x}_1, A^3\bar{x}_1$, iteratively storing and normalising the result into $\bar{x}_1$ on every iteration. This process continues until the sequence converges to the eigenvector corresponding to the largest eigenvalue. However, using power iterations to solve eigenvalue problems requires expensive computation. Therefore, methods such as Krylov subspace $K_r(A, \bar{x}_1)$ can be used for achieving similar results with fewer computations [14].

Using the Arnoldi iteration, eigenvalues can be computed as an orthogonal projection of $A$ onto $K_N(A, \bar{x}_1)$ subspace represented by $H$.

$$ H = P_1^* A P_1. \quad (2.4) $$

The dimensionality of $H$ is $N - 1 \times N - 1$ and some of its eigenvalues converge to the eigenvalues of the full system $A$. From Equation 2.4, we then have:

$$ AP_1 = P_1 H + e_N, \quad (2.5) $$

where, $e_N$ is the $N^{th}$ canonical basis vector in $N^{mn}$.

Solving the eigenvalues for $H$ is computationally efficient and can be obtained using QR decomposition [15] or the SVD decomposition [16]. The introduced $H$ is a companion matrix, that simply shifts from frames 1 through $N - 1$ and approximates the last frame $N$ by linearly combining the previous $N - 1$ frames, i.e., $P_2 = c_0\bar{x}_1 + ... + c_N\bar{x}_{N-1} = \{\bar{x}_1, \bar{x}_2, \bar{x}_3, \cdots, \bar{x}_{N-1}\}c$.

$$ H = \begin{pmatrix} 0 & 0 & \ldots & 0 & -c_0 \\ 1 & 0 & \ldots & 0 & -c_1 \\ 0 & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 1 & 0 & -c_{N-2} \\ 0 & 0 & 0 & 1 & -c_{N-1} \end{pmatrix}. \quad (2.6) $$

In matrix form, we then have:

$$ P_2 \approx P_1 H. \quad (2.7) $$

From Equations 2.5 and 2.7, we have:

$$ AP_1 \approx P_2 \approx P_1 H. \quad (2.8) $$

Hence this procedure will result in the low dimensional system matrix $H$, which can be calculated through QR decomposition,
Here, $Q^+$ is the pseudo inverse of $Q$ from the QR-decomposition of $P_1$ subspace. We solve the $H$ matrix problem using eigenvalue analysis and obtain eigenvalues and eigenvectors. It is known that eigenvalues of $H$ approximate some of the eigenvalues of the full system $A$. The associated eigenvectors of $H$ provide the coefficients for the linear combination that is necessary to express the dynamics within the video frame basis [6].

$$H \omega = \sigma \omega,$$  \hspace{1cm} (2.10)

DMD modes are thus calculated as follows:

$$\Psi = P_2 \omega$$  \hspace{1cm} (2.11)

### 2.1.2 SVD Approach

A more robust solution can be achieved using SVD [17] on $P_1$ in Eq. (2.2), to obtain $U$, $\Sigma$ and $V^*$ matrices.

$$P_2 = AU\Sigma V^*.$$  \hspace{1cm} (2.12)

$$\therefore P_1 = \sum_{i=1}^{d} \Sigma_i u_i v_i^*, \text{ here, } \Sigma_i \text{ is the } i^{th} \text{ singular value of } P_1, \text{ } u_i \text{ and } v_i^* \text{ are the corresponding left and right singular vectors respectively; and } d \text{ is the total number of singular values. Re-arranging Eq. (2.12), we obtain the the full-rank matrix } A,$$

$$A = P_2 V \Sigma^{-1} U^*.$$  \hspace{1cm} (2.13)

Since the eigenvalue analysis is agnostic to any linear projection, solving the eigen problem of $\tilde{H}$ is easier than that of solving for $A$ directly. Moreover, the associated eigenvectors of $\tilde{H}$ provide the coefficients for the linear combination that is necessary to express the dynamics within the time series basis.

$$\tilde{H} \omega = \sigma \omega,$$  \hspace{1cm} (2.14)

where, $\omega$ are the eigenvectors and $\sigma$ a diagonal matrix containing the corresponding eigenvalues of $\tilde{H}$ matrix. The eigenvalues of $\tilde{H}$ approximate some of the eigenvalues of the full system $A$ [6]. We then have:

$$AU \approx U \tilde{H},$$  
$$AU \approx U \omega \sigma \omega^{-1},$$  \hspace{1cm} (2.15)

$$A(U \omega) \approx (U \omega) \sigma.$$

Therefore, $\tilde{H}$ is determined on the subspace spanned by the orthogonal singular basis vectors $U$ obtained via $P_1$.

$$\tilde{H} = U^*(A)U,$$

$$\tilde{H} = U^*(P_2 V \Sigma^{-1} U^*)U,$$  \hspace{1cm} (2.16)
which can be rewritten as:
\[
\tilde{H} = U^*P_2V\Sigma^{-1}.
\]  (2.17)

Here \(U^* \in \mathbb{C}^{(N-1) \times mn}\) and \(V \in \mathbb{C}^{(N-1) \times (N-1)}\) are the conjugate transpose of \(U\) and \(V^*\), respectively; and \(\Sigma^{-1} \in \mathbb{C}^{(N-1) \times (N-1)}\) denotes the inverse of the singular values \(\Sigma\).

By replacing \(\Psi = U\omega\) in Eq. (2.15) i.e., \(A(\Psi) \approx (\Psi)\sigma\), we obtain the dynamic modes \(\Psi\). Therefore, we have:
\[
\Psi = P_2V\Sigma^{-1}\omega
\]  (2.18)

The complex eigenvalues \(\sigma\) contain growth/decay rates and frequencies of the corresponding DMD modes [3, 18]. If \(\sigma_j\) are the diagonal elements of \(\sigma\) from Eq. (2.14), the temporal behaviour of the DMD modes is then formed via Vandermonde matrix \(V\), which raises its column vector to the appropriate power. \(V(f)\) with \((L-1) \times (f+1)\) elements will be defined by the following:
\[
V(f) = \begin{pmatrix}
1 & \sigma_1^1 & \sigma_2^1 & \cdots & \sigma_f^1 \\
1 & \sigma_1^2 & \sigma_2^2 & \cdots & \sigma_f^2 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & \sigma_1^{L-1} & \sigma_2^{L-1} & \cdots & \sigma_f^{L-1}
\end{pmatrix},
\]  (2.19)

\(V(L)\) is a standard Vandermonde matrix for reconstruction but if \(f > L\), this is used for forecasting, which will be discussed in section 2.1.2.

DMD modes with frequencies \(\mu_j\) are defined by:
\[
\mu_j = \frac{\ln(\sigma_j)}{\delta t},
\]  (2.20)

where \(\delta t\) is the lag between the vectors and may be considered to consist of \(K\) time steps. The real part of \(\mu_j\) regulates the growth or decay of the DMD modes, while the imaginary part of \(\mu_j\) drives oscillations in the DMD modes.

**Reconstruction and Forecasting**

The DMD reconstruction of the data \(\hat{X}\) at any instance \(f\) after the initial vector \(\bar{x}_1\) is given by:
\[
\hat{X} = \Psi(V(f) \circ b), \quad \forall f \in \{1, 2, \cdots, F\}.
\]  (2.21)

Where \(V(f) = \sigma^f\) and \(F\) is the number of future vectors that are to be forecast. The vector \(b \approx \Psi^{-1}\bar{x}_1\) contains the initial amplitudes for the dynamic modes. Operator \((\circ)\) is the point-wise (element-wise) multiplication of \(b\) with every column of \(V(f)\). The first vector \(\bar{x}_1\) of the data \(X\) reduces to
\[
\bar{x}_1 = \Psi b,
\]  (2.22)

and \(b\) can be obtained through the Moore-Penrose pseudo inverse,
\[
b = (\Psi^*\Psi)^{-1}\Psi^*\bar{x}_1.
\]  (2.23)
CHAPTER 2. DYNAMIC MODE DECOMPOSITION

Here \((\cdot)^*\) is the conjugate transpose. The significance of the DMD modes \(\Psi\) is determined by their scaling factors \(b\) when it comes to approximating the entire data sequence \(P_2\) during reconstruction and forecasting.

2.2 Related Work

Tirunagari et al. in [4] compared PCA and DMD in extracting Kelvin-Helmoltz instabilities from complex flows. They showed that DMD is superior to PCA in analysing complex fluid flows. The Kelvin-Helmholtz instability can occur when there is passive scalar shear in a single continuous fluid, or when there is a passive scalar difference across the interface between two fluids as shown in Figure 2.1. The study of KH-instability (i.e., formed by coherent flow structures) has a major impact on the mixing of the fuel and the oxidiser in the internal combustion engine. The analysis of these structures in a turbulent jet is essential in understanding the fundamentals of fluid dynamics.

![LES of 2D jet with pressure data showing a close passive scalar difference across the interface between two flows. A total of 133 snapshots were considered with a time interval of 0.006 seconds.](image)

The PCA modes are the optimal decomposition for the flow and capture large scale structures thereby providing information on the large scale behaviour. The total energy is contained within the first few PCA modes. Generally 95% of the total energy can used to describe an average flow field with in the very first few PCA modes. The Dynamic modes represent the perturbation dynamics and capture the characteristic patterns located near the shear layer. DMD spectrum quantitatively describes the jet behaviour where as PCA lacks this property.

From the PCA mode (Figure 2.2 (a)), it can be observed that there is a formation of symmetric structures of Kelvin-Helmholtz instability. But from the Dynamic mode (Figure 2.2 (b)), the Kelvin-Helmholtz instability is clearly visible. The dynamic mode shows a characteristic pattern located near the shear layer, which represents the roll-up of the symmetric vortex sheet. The Dynamic mode also depict small scale structures near the nozzle exit region.
2.2. RELATED WORK

This is consistent with the literature in both experimental and numerical flow field data problems [18]. Indeed, not only can DMD extract dynamic information effectively by capturing the spatially-coherent structures, but also can represent the temporal dynamic characteristics [18]. Schmid et al. [17] used DMD on a sequence of fluid flow images and illustrated how it can detect dynamically relevant coherent structures for characterising fluid behaviour over a given time interval. Schmid [18] introduced a DMD-variant by taking into consideration an approximation of the linear mapping for temporal ‘snapshots’ and afterwards detects the relevant frequencies. This method is, however, dependent on the snapshots acquired. Problems addressed by DMD include: determining the flow over a square cavity, calculating the wake structure behind a flexible membrane, and determining the instabilities observed in experiments of a jet passing between two cylinders [18].

2.2.1 Recent Advancements

DMD has witnessed significant theoretical and algorithmic advancements which enhanced the robustness of the method and its applicability of the approach as shown in Table 2.1.

This section presents the recent advancements of the DMD with applications in computer vision and signal processing, viz., (i) sparsity promoting DMD for subtle emotion detection, (ii) DMD for background/foreground extraction and (iii) randomised DMD for background modelling.

Sparsity Promoting DMD for Subtle Emotion Detection

DMD with sparsity promotion (DMDSP), learns the sparse structures of the underlying dynamics and their scaling factors. DMD was originally developed to extract reduced order models of a complex system by projecting onto a low-complex sub-

Figure 2.2: Comparison of (a) PCA mode and (b) DMD mode for close passive scalar field.
space spanned by dynamic modes with a few degrees of freedom. Therefore, DMD is only suitable for analysing temporal dynamics but not learning their sparse structures. Therefore, a sparsity-constrained variant of DMD, introduced by Jovanovic et al. [23] will make it able to identify a smaller set of important DMD modes which are important for reconstruction of the original data sequence. Sparsity promoting DMD can be regarded as a post processing technique for identifying DMD modes and DMD eigenvalues selection using convex optimisation techniques.

We know that the complex DMD eigenvalues $\sigma$ contains growth/decay rates and frequencies of the corresponding DMD modes [3,18]. While, the temporal behaviour of the DMD modes are preserved via Vandermonde matrix $\mathbf{V}$. The significance of the DMD modes $\Psi$ are based on their scaling factors called amplitudes denoted by $b$.

$$b = (\Psi^*\Psi)^{-1}\Psi^*\bar{x}_1. \quad (2.24)$$

In the case of DMDSP the amplitudes are given by $\alpha = [\alpha_1, \cdots, \alpha_N]$. Similar to DMD, these amplitudes are calculated based on the growth/decay rates of the DMD eigenvalues in order to approximate the entire data sequence optimally.

$$\min_\alpha J(\alpha) = \|\Sigma \mathbf{V}^* - \sigma D_\alpha \mathbf{V}\|_F^2, \quad (2.25)$$

which is a convex optimisation problem and its solution leads to

$$\alpha = ((\sigma^*\sigma) \circ (\mathbf{V}_{\text{and}} \mathbf{V}_{\text{and}}^*))^{-1}\text{diag}(\mathbf{V}_{\text{and}} \mathbf{V} \Sigma^* \sigma). \quad (2.26)$$

Here the over line denotes the complex-conjugate of a matrix and $\circ$ is the element wise multiplication of two matrices. The DMD modes are then sorted in decreasing order of their amplitudes. The most significant DMD modes captures the most important dynamic structures in order to achieve a good quality approximation of the original image sequence.

DMDSP has been recently used for detecting subtle emotions from the facial videos [27]. In crime investigations, it is important to identify criminals who conceal their emotions as part of their deception. Therefore is often crucial to detect subtle emotions are presented in diverse real-life situations. The spontaneous subtle emotions are typically expressed through micro-expressions, which are tiny, sudden and short-lived dynamics of facial muscles [27]. The authors, decomposed the video sequences into spatial and temporal dynamic modes at several specific spectral frequencies; then, enforced sparsity constraints to learn the temporal and spectral structures of sparsity to eliminate irrelevant facial dynamics and identify dynamics of micro-expressions which ultimately would ease the challenge in the visual recognition of spontaneous subtle emotions as shown in Figure 2.3.

**DMDSP for Perturbation Estimation in Human Robot Interaction**

A robot’s behaviour must be stable to external forces, especially when interacting with humans. Traditionally, this behaviour is captured through several sensors fit-
2.2. RELATED WORK

(a) Subtle Emotion (b) DMD Mode (c) DMDSP Mode

Figure 2.3: Visual comparison between a noisy and redundant DMD spatial mode shown in (b) and a clear and significant DMDSP spatial mode in (c) from a subtle expression in (a). Image taken from [27].

ted within the robot. These sensors are often noisy in nature. DMDSP in [7] was shown to be able to extract the temporal dynamics by separating the noise from regular oscillations in sensor readings during cyclic robot movements under different behaviour configurations. The results in [7] showed that the DMD approach outperforms the other interpolation schemes particularly in the presence of sensor noise. This shows the potential of DMD as a highly promising method for robotics.

DMD for Extracting Spatial-Temporal Coherent Patterns in Large-Scale Neural Recordings

Typically in the neuroscience community large-scale neural activities are recorded using several electrodes simultaneously over several hours. Understanding these activities sheds light into the human brain activities. DMD in [8] was used to extract coherent patterns across both space and time, from such neural recordings containing sub-dural electrode array recordings from human subjects performing a known motor activation task. DMD was also coupled with a clustering algorithm for extracting sleep spindle networks from the same subjects.

DMD for Background Subtraction

A recent study in [6] has introduced the method of Dynamic Mode Decomposition (DMD) for robustly separating video frames into a background model and foreground components. The DMD method is applied on the data matrix containing an image sequence from a surveillance video. The DMD eigenvalues with frequencies \( \mu_j \) are calculated as \( \frac{\ln(\sigma_j)}{\delta t} \). Here, \( \delta t \) is the time difference between the images and considered to be 1 [6]. The real part of \( \mu_j \) regulates the growth or decay of the DMD modes, while the imaginary part of \( \mu_j \) drives oscillations in the DMD modes. The frequencies near the origin (zero-modes) are interpreted as background (low-rank)
portions of the given image sequence, and the frequencies bounded away from the origin are their sparse counterparts. Specifically, the parts in image sequence that do not change in time, have an associated frequency \( \| \mu_j \| \approx 0 \), which corresponds to background.

Randomised DMD for Background Subtraction

The DMD method when solving using the SVD can be time consuming when dealing with large scale problems such as video data. Therefore, Randomised SVD (RSVD) is used as an accurate and robust solution to estimate a number of dominant eigen components quicker as shown in [25].

SVD decomposition of a matrix \( \mathbf{P}_1 \) is defined as

\[
\mathbf{P}_1 = \mathbf{U} \Sigma \mathbf{V}^*,
\]

where \( \mathbf{U} \) and \( \mathbf{V} \) are the left and right singular vectors, \( \Sigma \) contains singular values, and \( (\cdot)^* \) denotes the transpose operator.

While RSVD calculates the first \( i \)th eigen components of the data by using QR decomposition and mapping \( \mathbf{P}_1 \) to a smaller matrix as

\[
\Omega = \text{randn}(N, i), \quad Y = X \Omega, \quad Y = QR \quad B = Q^T \mathbf{P}_1,
\]

where \text{randn} generates a random matrix of size of its inputs and \( N \) is the number of fragments. After decomposing \( \mathbf{B} \) using SVD, the final factors are obtained using \( \mathbf{Q} \) and the eigen factors of \( \mathbf{B} \).

Multi-Resolution DMD (MR-DMD) for Background Subtraction and Object Tracking

Recently in [24] DMD with a multi-resolution approach (MR-DMD) decomposed video streams into multi-time scale features and objects. The MR-DMD approach is similar to that of applying standard DMD technique at several resolutions after discarding the slow varying modes (background modes or the most significant modes). In other words, the MR-DMD approach runs the standard DMD algorithm over a sequence of images to produce several dynamic modes. Later, the most significant modes (background modes) are discarded; therefore, DMD is run using the least significant modes. This process is continued at several time resolutions allowing an image sequence to be separated for objects moving at different rates against the slowly varying background. This approach therefore, allows for multiple target tracking and detection. MR-DMD method has efficiently demonstrated, shifting \textit{El Niño} mode from ocean temperature data. In another application, MR-DMD was able to detect objects moving at different scales from a video [24].
2.3 Advancements in this thesis

One of the key advantages of DMD is its data-driven nature which does not rely on any prior assumption about the form of objects in the image sequence except its inherent image dynamics which are captured in terms of ‘DMD’ modes. This makes DMD a complete data-driven approach. The other advantage of this method is its ability to identify dominant motions in an image sequence. Just as Principal Component Analysis (PCA) is a tool that can identify dominant principal components of an image sequence based on co-variance, DMD can identify dominant modes of motion in an image sequence. Therefore, it is thus potentially well-suited to applications where there is motion information.

For example in this thesis, particularly in medical imaging, DMD modes with dominant motion information could isolate key relevant organs that contain wide variation of blood flow and filtration patterns (Figure 2.4). Discarding DMD modes with redundant motion components can provide stabilised motion (breathing) compensated medical imaging videos. This can potentially provide clinicians with a better way to manage their clinical time and cost.

![Figure 2.4](image)

Figure 2.4: (a) 5 randomly selected images from a Dynamic Contrast Enhanced Magnetic Resonance Imaging (DCE-MRI) sequence of a healthy volunteer. The middle image shows the peak stage of renal perfusion inside the kidneys. Quantification of kidney function is performed by calculating the mean of pixel intensity values inside the kidneys. (b) image showing their corresponding dynamic mode-2. (c) image showing the binarized version of that dynamic mode. (d) Kidney template with the largest area of connected pixels.

Similarly in the context of 2D facial counter-spoofing, extracting relevant modes
from videos in face recognition can provide a good separability between valid videos and the spoofed ones. Particularly, for biometric applications such as replay and print attacks, in which observations are carried out in a continuous pattern i.e. in a video sequence. DMD has the potential of discovering the facial dynamics that captures the vitality signs of valid face videos (Figure 2.5) whilst extracting artefacts of spoof videos such as moire\(^1\) and planar effects\(^2\), at the same time.

\[\text{Figure 2.5: Four frames from a valid access, controlled scenario video clips showing the eye-blinking and the lip movements. The bottom row shows their corresponding first dynamic mode image. The images in the top two rows show the corresponding extracted frames from each of the two videos containing a valid-access request in a controlled scenario. Since the person in these videos was speaking and blinking his eyes inevitably from time to time, one can observe that the corresponding first DMD mode highlights dynamic changes in the lip movements and eyes blinks. However, eye blinking is less obvious for the second video.}\]

\(^1\)The moire patterns are often regarded as undesired artefacts of images, the effect of which is similar to fibres in moire silk.

\(^2\)The planar effect is the appearance of an picture that is flat rather than 3D, as opposed to a real face.
DMD not only captures the dynamic frequency information about the data but also preserves the details of their temporal evolution under consideration. Therefore, its frequency based representation can be used to identify stationary modes, i.e., modes with frequency $\approx 0$, which reveals background of a video sequence as shown in Figure 2.6.

![Figure 2.6: (a) Exemplar frame from ‘HighwayI’ sequence of the SBI dataset where the background is occluded with a number of foreground objects. (b) Background models generated by DMD_CT.](image)

Similarly in the case of signals, reconstruction of the original time series with DMD modes whose eigenvalue frequency $\approx 0$ captures the trend (trend-1) information as shown in Fig. 2.7 (a). Reconstruction with other DMD modes that are closer to the origin, provides a smoother version (trend-2) of the time series (Figure 2.7 (a)). The assumptions of DMD formulation makes it easier to forecast the trends in a time series as shown in Figure 2.7 (b).
<table>
<thead>
<tr>
<th>DMD variant</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exact DMD [10]</td>
<td>Exact DMD accepts the input data in snapshot pairs unlike DMD that accepts input data as a sequence of snapshots. This allows data from multiple experiments to be aggregated into a single dataset, thus allowing multi modal data analysis. On the other hand, while DMD projects data onto PCA modes. The Exact DMD algorithm produces modes, without utilising the PCA modes.</td>
</tr>
<tr>
<td>Total Least Squares DMD (TLS-DMD) [19]</td>
<td>Introduced to address the bias in measurement noise when dealing with the experimental data. The bias is due to the fact that Exact DMD is interpreted as a regression problem which is solved using ordinary least squares assuming the regressor is noise free.</td>
</tr>
<tr>
<td>Optimised DMD [20]</td>
<td>Introduced to solve two limitation of standard DMD using optimisation i.e., mode selection and noise sensitivity of DMD in producing the last snapshot.</td>
</tr>
<tr>
<td>Optimal Mode Decomposition [21]</td>
<td>Solves DMD as an optimisation problem with rank constraint that produce linear models with smaller residual errors.</td>
</tr>
<tr>
<td>DMD with Control [22]</td>
<td>Introduced to analyse data obtained from input output systems. Especially for actuated systems, where DMD analysis is incapable of producing an input-output model. DMD with Control has the ability to disambiguate the effects of system actuation from the open loop dynamics.</td>
</tr>
<tr>
<td>Sparsity Promoting DMD (DMDSP) [23]</td>
<td>Introduced for identifying a smaller set of important DMD modes and eigenvalues. DMDSP learns the sparse structures of the underlying dynamics and their scaling factors. Sparsity promoting DMD can be regarded as a post processing technique for identifying DMD modes and DMD eigenvalues selection using convex optimisation techniques.</td>
</tr>
<tr>
<td>Multi-Resolution DMD (MR-DMD) [24]</td>
<td>Designed for extracting DMD modes and eigenvalues from dataset containing multiple timescales. MR-DMD decomposes video streams into multi-time scale features and objects by applying standard DMD technique at several resolutions after discarding the slow varying modes background modes. This process is continued at several time resolutions allowing an image sequence to be separated for objects moving at different rates against the slowly varying background. This approach therefore, allows for multiple target tracking and detection.</td>
</tr>
<tr>
<td>Randomised DMD [25]</td>
<td>Introduced for improving the time complexity of DMD by solving SVD through random projections.</td>
</tr>
<tr>
<td>Tensor DMD [26]</td>
<td>Introduced for reducing the computational and memory complexity of DMD via using tensor train networks.</td>
</tr>
</tbody>
</table>

Table 2.1: Algorithmic advancements of DMD.
Chapter 3

Univariate Time Series Analysis

This chapter of the thesis, introduces the method of Dynamic Mode Decomposition (DMD) for analysing univariate time series by forecasting as well as extracting trends and frequencies. The key advantage of DMD is its data-driven nature which does not rely on any prior assumptions (like Singular Spectrum Analysis (SSA)) except the inherent dynamics which are captured in terms of ‘DMD’ modes. Indeed, this chapter will show that the DMD eigenvalues with frequencies that are closer to the origin in the complex plane capture the trends in the time series. Moreover, the temporal evolution of the DMD modes, which is preserved via the Vandermonde matrix, can be used to reconstruct the desired components and perform forecasting at the same time. The results at various noise levels on simulated data suggest that DMD is a promising approach to modelling a time series with a noisy structure. Although these properties are not new in the DMD literature, the novel contributions of this thesis are in making the method of DMD work for univariate time series through Hankelisation. Thus, this is the first work that shows DMD can be used for modelling, predicting, and forecasting a univariate time series.

3.1 Introduction

Time series analysis includes extracting meaningful information like filtering trends, and forecasting future values [28, 29]. It has applications in many areas, such as quantitative finance and biomedical engineering [30–32]. This signal processing problem is often tackled by decomposing the data into several components in the frequency domain, thus decoupling signals or trends from noise. To this end, matrix decomposition techniques such as singular spectrum analysis (SSA) [30, 33], are often used. Analogous to SSA, this thesis presents a data-driven approach, Dynamic Mode Decomposition (DMD).

From Chapter 2, DMD could be able to extract the relevant modes from em-
pirical data generated by non-linear complex fluid flows [3, 4, 17, 18] as well as for robustly separating video frames into a background model and multiple foreground objects [6, 25, 34]. All the aforementioned DMD applications consider a data matrix as an input. For instance, to analyse an image sequence (a video) using DMD, the input data are converted into column vectors, each of which representing a single frame of image. However, being a fundamentally multidimensional method, DMD has never been applied to a univariate time series. Therefore, our contribution is to introduce the method of DMD through a Hankelisation approach for decomposing a univariate time series into a number of interpretable elements in terms of noise, trend and harmonics. Moreover, time series forecasting using DMD is also presented in this chapter.

The methodological pipeline consists of four stages similar to that of SSA. However, it is different from SSA in the formulation. The beauty of DMD is that it captures frequency components that describe the growth/decay and oscillations in the data whilst at the same time it can forecast time series during the reconstruction stage.

The following research questions are answered in this chapter:

1. How to extend the method of DMD to process a univariate time series data?
2. What are the potential advantages of DMD when compared to the current state-of-the-art methods in the field of signal processing?
3. How robust is the DMD method when compared to those methods, particularly in the presence of a added white Gaussian noise?

In the remainder of this chapter, methodology is described in Section 3.2. Experiments and results are presented in Section 3.5. Finally, conclusions and discussions are presented in Section 3.6.

### 3.2 Methodology

This section formulates DMD for univariate time series analysis. Our algorithm consists of four stages: (1) embedding, (2) dynamic mode decomposition (DMD), (3) reconstruction and forecasting, and (4) Hankelisation and diagonal averaging. Each step is explained in more detail in the following subsections:

#### 3.2.1 Embedding

Embedding can be considered as a mapping that converts a univariate time series data \( x = [x_1, x_2, \ldots, x_T] \) of length \( T \) into a multidimensional series \( X = [\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_L] \), of size \( K \times L \). The vector \( \bar{x}_n = [x_n, x_{n+1}, \ldots, x_{n+K-1}]^T \in \mathbb{R}^K \) is called a \( K \)-lagged vector, and \( L \in [2 \leq L \leq \frac{T}{2}] \) is the length of window whereas \( K = T - L + 1 \), is the number of segments. Thus, the time series is decomposed into
3.2. METHODOLOGY

$K$ overlapping segments of length $L$. This produces a trajectory matrix $X \in \mathbb{R}^{K \times L}$. Unfolding the matrix $X$ with elements $m \times n$, we have:

$$
X = \begin{pmatrix}
  x_1 & x_2 & \ldots & x_L \\
  x_2 & x_3 & \ldots & x_{L+1} \\
  \vdots & \vdots & \ddots & \vdots \\
  x_K & x_{K+1} & \ldots & x_{K+L-1}
\end{pmatrix}.
$$

(3.1)

Note that $X$ is a Hankel matrix, because its anti-diagonal elements $(m+n) = \text{const}$, are equal. Where, $m, n$ are the row and column indices of the Hankel matrix.

3.2.2 Dynamic Mode Decomposition

The embedded data matrix $X$ containing $K$-lagged vectors $\bar{x}_n \in \mathbb{R}^K$ which are evenly spaced by $\delta t = K$ is then solved through DMD via SVD approach to obtain the dynamic modes $\Psi = P_2 V \Sigma^{-1} \omega$ and their corresponding complex eigenvalues $\sigma$ (refer Chapter 2, Section 2.1.2 for the methodology). The complex eigenvalues $\sigma$ contain growth/decay rates and frequencies of the corresponding DMD modes [3,18]. If $\sigma_j$ are the diagonal elements of $\sigma$ from Eq. (2.14), the temporal behaviour of the DMD modes is then formed via Vandermonde matrix $V$, which raises its column vector to the appropriate power. $V(f)$ with $(L-1) \times (f+1)$ elements will be defined by the following:

$$
V(f) = \begin{pmatrix}
  1 & \sigma_1^1 & \sigma_2^1 & \ldots & \sigma_f^1 \\
  1 & \sigma_1^2 & \sigma_2^2 & \ldots & \sigma_f^2 \\
  \vdots & \vdots & \vdots & \ddots & \vdots \\
  1 & \sigma_{L-1}^1 & \sigma_{L-1}^2 & \ldots & \sigma_{L-1}^f
\end{pmatrix},
$$

(3.2)

$V(L)$ is a standard Vandermonde matrix for reconstruction but if $f > L$, this is used for forecasting, which will be discussed in section 2.1.2.

3.2.3 Reconstruction and Forecasting with Selected DMD Triples

Since the slow varying frequencies $\mu_j$ capture trends in the time series, it is desirable to reconstruct the trajectory matrix with selected DMD triples. A DMD triple consists of a DMD eigenvalue, its corresponding DMD mode and an associated scaling factor, i.e., amplitude.

$$
\hat{X} = \Psi_{\{S\}}(V_{\{S\}}(f) \circ b_{\{S\}}), \quad S \subset \{1, ..., L-1\}.
$$

(3.3)

$V_{\{S\}}(f)$ is a subset of $V(f)$ formed by selecting only the appropriate rows which correspond to the selected DMD eigenvalues. $V_{\{S\}}(f)$ will have the dimension of $S \times (f+1)$ and $b_{\{S\}}$ has the dimension of $(S \times 1)$ for the selected $S$ eigenvalues.
3.2.4 Hankelisation and Diagonal Averaging

Finally, the reconstructed trajectory matrix $\hat{X}$ is Hankelised, in order to ensure that the anti-diagonal elements are equalised. The Hankelisation operator $\mathbb{H}$ for an $K \times F$ matrix $\tilde{X}$ is defined as

$$
\mathbb{H}\tilde{X} = \tilde{X} = \begin{pmatrix}
\tilde{x}_1 & \tilde{x}_2 & \ldots & \tilde{x}_F \\
\tilde{x}_2 & \tilde{x}_3 & \ldots & \tilde{x}_{F+1} \\
\vdots & \vdots & \ddots & \vdots \\
\tilde{x}_K & \tilde{x}_{K+1} & \ldots & \tilde{x}_{K+F-1}
\end{pmatrix}.
$$

(3.4)

The desired or reconstructed time series is given by:

$$
\Re(\tilde{x}_k) = \frac{1}{\text{num}(D_k)} \sum_{m,n \in D_k} \hat{X}_{m,n}.
$$

Here $k$ is the index of the time series, $\Re(.)$ is the real part of $\tilde{x}_k$, $\text{num}(.)$ is the number of combinations of $(m,n)$, $m + n = k + 1$ and $D_k$ is given by $\{(m, n) : 1 \leq m \leq K, 1 \leq n \leq F, m + n = k + 1\}$.

The methodology, as summarised via utilising Eq. (3.3), can be used to decompose data into a number of desired subcomponents including trend, noise and future components, but at the expense of introducing a time stamp delay. This is because, $P_2$ has been used for projection from Eq. (2.18), which does not contain information about the first data point. Therefore, this thesis recommends that the first column vector of the trajectory should be repeated.

3.3 Singular Spectrum Analysis

SSA is another powerful method for analysing real-valued time series [35]. The initial idea of SSA is associated with the work of Broomhead [36, 37]. It combines the multivariate statistics, classical time series analysis, dynamical systems, and signal processing.

SSA is becoming an effective method in various areas such as economics [35, 38] and biomedical engineering [39–41]. Basically, SSA decomposes a data into a number of interpretable elements with different subspaces, such as noise and trend [42], and can be used for any time series with complex structure [38]. For instance, for decades, SSA has been used for both trend detection and prediction in financial data [35, 38]. SSA can be used in several applications including trends fitting, extraction of cycles with various periods and amplitudes, smoothing, and finding some structures in short time series.

Ordinary SSA is a subspace decomposition algorithm with four stages similar to that of the proposed method but with different formulation in the second and third stages. SSA decomposes the resulting matrix of the embedding step using SVD:

$$
X = \sum_{i=1}^{d} X_i = \sum_{i=1}^{d} \sqrt{\sigma_i} u_i v_i^T,
$$

(3.5)
where \( \sigma_i \) is the \( i \)th eigenvalue of covariance matrix \( \mathbf{XX}^T \), \( \mathbf{u}_i \) is the corresponding eigenvector, \( d \) is the total number of eigenvalues, and \( \mathbf{v}_i = \mathbf{X}^T \mathbf{u}_i / \sqrt{\sigma_i} \).

In the reconstruction step, first the elementary matrices from the previous stage are grouped into several sub-matrices
\[
\mathbf{X} = \sum_{q=1}^{Q} \hat{\mathbf{X}}_q,
\]
(3.6)
where \( Q \) determines the total number of groups, index \( q \) refers to \( q \)th subgroup of eigenvalues, and \( \hat{\mathbf{X}}_q \) indicates the sum of \( \mathbf{X}_i \) within group \( q \). Finally, the desired subgroup is selected and transformed into a Hankel matrix.

SSA has recurrent and vector based Forecasting techniques:

### 3.3.1 SSA Recurrent Forecasting

If the desired group of eigentriples is considered as \( J \) and the corresponding \( i \)th eigenvector as \( \mathbf{u}_{I_i} \), then
\[
r = \frac{1}{1 - \upsilon^2} \sum_{i \in I} \pi_i \mathbf{u}_{I_i},
\]
(3.7)
where \( r = (a_{I_1}, \ldots, a_1)^T \), \( \mathbf{u}_{I_i} \) and \( \pi_i \) respectively show all except for the last and the last column of \( \mathbf{u}_{I_i} \), and \( \upsilon^2 = \sum_{i \in I} \pi_i^2 \). Consequently, the recurrent forecasting can be formulated as
\[
x_i = \begin{cases} 
\tilde{x}_i & i = 1, \ldots, T \\
\sum_{j=1}^{i-1} a_j x_{i-j} & i = T + 1, \ldots, T + F
\end{cases}
\]
(3.8)
where \( F \) is the number of forecasting points. The reconstructed terms and Eq. (3.7) are used to forecast the value of new points. The linear operator \( \varphi \) can be defined as
\[
\varphi \mathbf{x} = \begin{pmatrix} 
\mathbf{x} \\
\mathbf{r}^T \mathbf{x}
\end{pmatrix},
\]
(3.9)
where \( \mathbf{x} \) shows \( K - 1 \) components of vector \( \mathbf{x} \). Therefore, the trajectory matrix of the time series \( \mathbf{x} \) of length \( T + F \) can be written as the vector representation of Eq. (3.8)
\[
\mathbf{z}_i = \begin{cases} 
\mathbf{x}_i & i = 1, 1, \ldots, L \\
\varphi \mathbf{z}_{i-1} & i = L + 1, \ldots, T + F
\end{cases}
\]
(3.10)

### 3.4 Dataset

#### 3.4.1 Real Data

To demonstrate and validate the methodology, publicly available Box & Jenkins airline data [29] have been used. This consists of the monthly total of international
airline passengers from the year 1949 to 1960. For comparing with SSA, BJ sales and UK gas data are also considered. All of the aforementioned datasets have been obtained from Github repository for R time series datasets\(^1\). These datasets then have been transformed into the logscale (y-axis) and the time (x-axis) is considered in increments of 1.

### 3.4.2 Simulated Data

The simulated data are a mixture of two signals

\[
x(t) = as(t) + \beta e(t),
\]

(3.11)

where \(s(t)\) shows the desired signal and \(a\) is the mixing vector. \(x(t)\) is the input or mixture being examined, \(e(t)\) indicates the unwanted signal or noise, and \(\beta\) is the noise level.

Signal-to-noise ratio (SNR) in terms of mean square (MS) is used as a measure for noise level which is adjusted by changing \(\beta\)

\[
\text{SNR} = \frac{\text{MS}(s(t))}{\beta^2}.
\]

(3.12)

To evaluate the simulation performance, the evaluation criteria in terms of mean square error (MSE) is considered as

\[
\text{MSE} = \frac{\text{MS}(s(t) - \hat{s}(t))}{\text{MS}(s(t))},
\]

(3.13)

where \(\hat{s}(t)\) is the estimated source.

### 3.5 Experiments and Results

The real time series containing airline data is transformed to a trajectory matrix with \(K\)-lagged vectors. The parameter that needs to be predetermined is the window length \(L\) because our conjecture is that an appropriate choice of this can produce a better result.

#### 3.5.1 Effect of Window Length \(L\)

Figure 3.1 shows our experimental results with window lengths \(L\) for reconstructing the original time series. Our result shows that the reconstruction error for \(L \in \{12, ..., 66\}\) gave a stable performance with relatively low reconstruction error. If \(L\) is relatively small, it can cause the trajectory matrix \(X\) to miss the trends in the data especially when dealing with seasonal components. Furthermore, if \(L\) is closer to \(\frac{T}{2}\), where \(T\) is the length of the time series, the decomposition of the data

\(^1\)https://github.com/vincentarelbundock/Rdatasets
becomes more detailed. Therefore, the selection of $L$ can neither be close to 2 nor $\frac{T}{2}$. Therefore, this study recommends choosing $L$ between 8% to 45% of the length of the time series. In our experiments we arbitrarily considered $L$ to be 25% of the length of the time series, i.e., $L = 35$.

![Figure 3.1: Effect of window length $L$ over the reconstruction performance.](image)

### 3.5.2 DMD for Extracting Trends

The logarithmic values of DMD eigenvalues give us the frequencies, the real part captures the growth/decay and imaginary part, the oscillations within the time series. The frequencies that are closer to origin are the slow varying components (for example, eigenvalues $\{18, 19, 20, 21, 22\}$ in Figure 3.2 (a) which are vertically closer to the X-axis) that essentially capture the trends.

![Figure 3.2: (a) DMD eigenvalue plot, showing eigenvalues $\{18, 19, 20, 21, 22\}$ that are vertically closer to the X-axis. (b) Corresponding absolute frequencies.](image)

The ones which are further away from the origin capture various frequency
growth/decay rates. In order to establish the order in the DMD, the absolute frequencies of the DMD eigenvalues are sorted in the ascending order. The absolute value of the frequencies associated with each of the eigenvalues are shown in Figure 3.2 (b).

Reconstruction of the original time series with the aforementioned slow varying DMD triples, i.e., with \{18, 19, 20\} in Figure 3.1 (or the first three DMD triples) gave us the trend (trend-1) as shown in Figure 3.3.

Reconstruction with other DMD triples that are closer to the origin, i.e., \{18, 19, 20, \ldots, 26\} gave a smoother version (trend-2) of the time series (Figure 3.3).

As we reconstruct the data with increasing DMD triples, the reconstruction mean square error (MSE) decreases (Figure 3.4 (a)). A complete reconstruction of the original time series with all of the DMD triples produces the original time series as shown in Figure 3.4 (b) with a reconstruction MSE of 0.0006.
3.5.3 DMD for Time Series Forecasting

To demonstrate the capability of DMD for forecasting, 124 out of the total 144 datapoints are considered as the training set, leaving the remaining 20 points as the test set. The window length $L$ considered comprises of 25% of the data i.e., $L = 31$. The forecast trends in the data are shown in Figure 3.5 (a). The MSE recorded for the forecast trend-1 was 0.0007 and for the smoother trend-2 was 0.0080 when evaluated against the reconstructed trend. The result of the forecast is shown in Figure 3.5 (b) with a MSE of 0.0090 when compared to the original data.

3.5.4 Comparison to SSA

On Simulated Data with Effect of Noise

In order to compare with SSA, the performance of both the methods should be investigated in the presence of noise. For this purpose, a simulated signal consisting of two sinusoids with different frequencies and powers is considered, which were then mixed with a Gaussian noise at various signal-to-noise ratio (SNR) levels \{-10, -9, \cdots, 19, 20\} dB.

Selection of the triples

The proposed method using the first 9 DMD triples and SSA with first 4 singular triples were applied as they those parameters gave the least reconstruction MSE across all the noise levels on these simulated data as shown in Figure 3.6.

Robustness to Noise

The mean and standard deviation of the MSE is reported in Table 3.1 across 500 iterations. From Figure 3.7, it can be noticed that, as the SNR increases, the MSE
in reconstruction (with respect to the clean signal) decreases in both of the methods. The average time taken for DMD across all the noise levels was 3.76, while for SSA was 46.94 seconds\textsuperscript{2}. This result shows that DMD is computationally more efficient than SSA and is consequently a promising approach for analysing a univariate time series in the presence of noise.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3_6.png}
\caption{MSE while reconstructing the original time series in the presence of noise at $-2dB$, using first $t$ (a) DMD triples, (b) SVD triples.}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3_7.png}
\caption{Effect of noise in reconstruction at various SNRs across 500 iterations.}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure3_8.png}
\caption{Original simulated signal and the noisy signal at SNR = $-2dB$. The reconstruction along with forecasting of 750 points, using the first 5 DMD triples and 4 singular triples is shown. The MSE recorded after reconstruction}

\textsuperscript{2}On a Dell PowerEdge R715 AMD - 2-CPU/24-Core - 128GB RAM Running Precise (Ubuntu 12.04.5).
3.5. EXPERIMENTS AND RESULTS

Table 3.1: Mean and standard deviation of MSE scores across 500 iterations at various SNRs.

<table>
<thead>
<tr>
<th>SNR(dB)</th>
<th>DMD</th>
<th>SSA</th>
</tr>
</thead>
<tbody>
<tr>
<td>-10</td>
<td>6.931 ± 0.2755</td>
<td>9.7204 ± 0.2753</td>
</tr>
<tr>
<td>-9</td>
<td>5.5023 ± 0.2031</td>
<td>7.7185 ± 0.2214</td>
</tr>
<tr>
<td>-8</td>
<td>4.3147 ± 0.1278</td>
<td>6.0885 ± 0.1565</td>
</tr>
<tr>
<td>-7</td>
<td>3.4595 ± 0.1321</td>
<td>4.8467 ± 0.1366</td>
</tr>
<tr>
<td>-6</td>
<td>2.7449 ± 0.1166</td>
<td>3.8485 ± 0.1155</td>
</tr>
<tr>
<td>-5</td>
<td>2.1912 ± 0.0957</td>
<td>3.0755 ± 0.0904</td>
</tr>
<tr>
<td>-4</td>
<td>1.728 ± 0.0697</td>
<td>2.4346 ± 0.0695</td>
</tr>
<tr>
<td>-3</td>
<td>1.3793 ± 0.0621</td>
<td>1.9267 ± 0.0544</td>
</tr>
<tr>
<td>-2</td>
<td>1.0953 ± 0.0372</td>
<td>1.5304 ± 0.0367</td>
</tr>
<tr>
<td>-1</td>
<td>0.8635 ± 0.0326</td>
<td>1.2191 ± 0.033</td>
</tr>
<tr>
<td>0</td>
<td>0.6884 ± 0.0291</td>
<td>0.9691 ± 0.0315</td>
</tr>
<tr>
<td>1</td>
<td>0.5475 ± 0.0211</td>
<td>0.7705 ± 0.0223</td>
</tr>
<tr>
<td>2</td>
<td>0.4344 ± 0.016</td>
<td>0.6093 ± 0.0188</td>
</tr>
<tr>
<td>3</td>
<td>0.3448 ± 0.014</td>
<td>0.4841 ± 0.0144</td>
</tr>
<tr>
<td>4</td>
<td>0.2753 ± 0.011</td>
<td>0.3858 ± 0.011</td>
</tr>
<tr>
<td>5</td>
<td>0.2178 ± 0.0067</td>
<td>0.3045 ± 0.0092</td>
</tr>
<tr>
<td>6</td>
<td>0.1723 ± 0.0063</td>
<td>0.2427 ± 0.0066</td>
</tr>
<tr>
<td>7</td>
<td>0.1384 ± 0.0047</td>
<td>0.1935 ± 0.0053</td>
</tr>
<tr>
<td>8</td>
<td>0.1084 ± 0.0043</td>
<td>0.1528 ± 0.004</td>
</tr>
<tr>
<td>9</td>
<td>0.0867 ± 0.0031</td>
<td>0.1221 ± 0.0033</td>
</tr>
<tr>
<td>10</td>
<td>0.069 ± 0.003</td>
<td>0.0968 ± 0.0028</td>
</tr>
<tr>
<td>11</td>
<td>0.0543 ± 0.0022</td>
<td>0.0763 ± 0.0024</td>
</tr>
<tr>
<td>12</td>
<td>0.0431 ± 0.0015</td>
<td>0.0609 ± 0.0016</td>
</tr>
<tr>
<td>13</td>
<td>0.0344 ± 0.0012</td>
<td>0.0484 ± 0.0013</td>
</tr>
<tr>
<td>14</td>
<td>0.0273 ± 0.0012</td>
<td>0.0384 ± 0.0011</td>
</tr>
<tr>
<td>15</td>
<td>0.0217 ± 0.0008</td>
<td>0.0306 ± 0.0009</td>
</tr>
<tr>
<td>16</td>
<td>0.0172 ± 0.0007</td>
<td>0.0243 ± 0.0007</td>
</tr>
<tr>
<td>17</td>
<td>0.0138 ± 0.0005</td>
<td>0.0193 ± 0.0005</td>
</tr>
<tr>
<td>18</td>
<td>0.0109 ± 0.0004</td>
<td>0.0153 ± 0.0005</td>
</tr>
<tr>
<td>19</td>
<td>0.0086 ± 0.0003</td>
<td>0.0122 ± 0.0003</td>
</tr>
<tr>
<td>20</td>
<td>0.0069 ± 0.0003</td>
<td>0.0097 ± 0.0003</td>
</tr>
</tbody>
</table>

and forecast was 0.0205 and for SSA was 0.0241. The time taken for DMD and SSA was 0.17 and 23.23 seconds respectively.

On Real Data

The results on the real world datasets are shown in Figures 3.9 (Airline passengers data), 3.10 (BJ sales data) and 3.11 (UK gas data). The MSE error in reconstructing and forecasting these data, including the parameters chosen are given in Table 3.2. These results show that DMD based univariate time series analysis is superior to SSA particularly when forecasting the time series.
CHAPTER 3. UNIVARIATE TIME SERIES ANALYSIS

Figure 3.8: Reconstruction and forecasting at $-2dB$

3.6 Conclusion and Discussions

This chapter of the thesis proposed the method of DMD in a four-stage pipeline to decompose a univariate time series into a number of interpretable elements with different components, such as noise and underlying data trend. In addition, this four-stage DMD pipeline is also shown to be used for forecasting a time series.

DMD eigenvalues with frequencies that are closer to the origin in the complex plane capture the trends in the time series. Moreover, the temporal evolution of the DMD modes, which is preserved via the Vandermonde matrix, can be used to reconstruct the desired components and perform forecasting at the same time. Window length $L$ needs to be predetermined and this thesis recommends choosing $L$ between 8% to 45% of the length of the time series. In reconstruction, to compensate one time stamp delay, we recommend that the first column vector of the trajectory matrix be repeated once. The experimental results at various noise levels on simulated data suggests that DMD is a promising approach to modelling a time series with a noisy structure. DMD was compared to SSA, a popular method employed extensively in biomedical signal processing applications for separating artefacts and extracting features from electromyography (EMG), electrocardiogram (ECG) and electroencephalogram (EEG) signals [39].
Thus DMD can potentially be applied to analyse biomedical signals, especially EEG, where brain rhythms manifest themselves as narrow frequency band components. As a good example, sleep is a dynamic process which consists of different stages with different neural activity levels. Each stage is characterised by a distinct set of physiological and neurological features and dominant frequency band. SSA, however, does not exploit the narrowband property of these kind of cyclic data since it requires manual grouping or optimisation [41] to detect the desired features. DMD on the other hand due to its intrinsic nature of complex eigenvalues that capture frequency bands, can help find cycling components automatically. Therefore, in such studies, DMD might capture clues for chaotic subcomponents/trends and noise. These subcomponents later can be used as feature vectors for e.g. in sleep stage classification, epilepsy detection, and seizure predictions (for e.g., using SSA in [30–32, 39]. Moreover, our experiments showed that DMD is computationally more efficient than SSA as well as robust in the presence of added white Gaussian noise.
CHAPTER 3. UNIVARIATE TIME SERIES ANALYSIS

Figure 3.9: Time series analysis on Airline passengers data: (a) trend analysis, (b) periodicity analysis, (c) complete reconstruction, (d) trend forecast, (e) periodicity forecast and (f) a forecast of the timeseries.
3.6. CONCLUSION AND DISCUSSIONS

Figure 3.10: Time series analysis on BJ Sales data: (a) trend analysis, (b) periodicity analysis, (c) complete reconstruction, (d) trend forecast, (e) periodicity forecast and (f) a forecast of the timeseries. DMD gives better performance compared to SSA in (f) for forecasting.
Figure 3.11: Time series analysis on UK gas data: (a) trend analysis, (b) periodicity analysis, (c) complete reconstruction, (d) trend forecast, (e) periodicity forecast and (f) a forecast of the timeseries. DMD gives better performance compared to SSA in (f) for forecasting.
Table 3.2: The parameters that have been used for analysing the real data using DMD and SSA. Here parameters, T and L represent the total length of the time series and length of the window considered for the analysis respectively. ‘Rec’ denotes reconstruction and ‘For’ denotes forecasting. MSE denotes the mean square error between the reconstructed/forecast time series and the original timeseries.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Parameters</th>
<th>DMD Triples</th>
<th>SSA Triples</th>
<th>MSE in Rec,For</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>T</td>
<td>L</td>
<td>Trend</td>
<td>Period</td>
</tr>
<tr>
<td></td>
<td>Rec For</td>
<td>Rec For</td>
<td>Rec For</td>
<td>Rec For</td>
</tr>
<tr>
<td>Airline Passengers</td>
<td>144 130</td>
<td>36 33</td>
<td>2 2</td>
<td>7 9</td>
</tr>
<tr>
<td>BJSales</td>
<td>150 135</td>
<td>38 34</td>
<td>2 2</td>
<td>7 7</td>
</tr>
<tr>
<td>UKgas</td>
<td>97 87</td>
<td>24 22</td>
<td>2 2</td>
<td>12 12</td>
</tr>
</tbody>
</table>
Chapter 4

Quantifying Kidney Function: Movement Correction and Segmentation of DCE-MRI

Quantification of kidney function in Dynamic Contrast Enhanced Magnetic Resonance imaging (DCE-MRI) requires proper segmentation of the kidney region of interest (ROI). However, images of the kidneys in DCE-MRI contain unwanted complex organ movements due to respiration. This gives rise to motion artefacts that hinder the clinical assessment of kidney function. Therefore, prior to segmentation, kidneys in DCE-MRI should undergo a proper movement correction. However, due to the rapid change in contrast agent within the DCE-MR image sequence, commonly used intensity-based image registration techniques are likely to fail. While semi-automated approaches involving human experts are a possible alternative, they pose significant drawbacks including inter-observer variability, and the bottleneck introduced through manual inspection of the multiplicity of images produced during a DCE-MRI study. Therefore, prior to quantifying kidney function, the DCE-MRI sequence should undergo movement correction and then functional segmentation of the kidney ROI. To address the issue of movement correction, this chapter presents a novel automated, registration-free movement correction approach based on windowed and reconstruction variants of Dynamic Mode Decomposition (WR-DMD). The proposed method is validated on ten different healthy volunteer kidney DCE-MRI datasets. The results, using block matching evaluation on the image sequence produced by WR-DMD, show the elimination of 99% of mean motion magnitude when compared to the original datasets, thereby demonstrating the viability of automatic movement correction using WR-DMD method. Furthermore, this chapter also shows that the obtained DMD modes when coupled with simple thresholding
and connected component analysis, can produce automatic segmentation of the kidney ROI. The segmentation result obtained from this technique is comparable to that of the human experts (and very significantly better than of a priori region segmentation). The result of the proposed technique gives a Jaccard coefficient of 0.87, while the experts’ annotation has an accuracy of 0.85, 0.88 and 0.87 when scored among themselves.

4.1 Introduction

Diagnosis of renal dysfunction based on blood and urine tests often produces inaccurate results as the creatinine levels in blood are detectable only after 60% of the renal dysfunction has taken place [43]. Therefore, to address this limitation Dynamic Contrast Enhanced Magnetic Resonance Imaging (DCE-MRI) has been proposed [44, 45]. DCE-MRI is a non-ionising alternative to conventional radioisotope renography. It has particular attraction in cases of Chronic Kidney Disease (CKD), which requires repeated functional assessment, as well as in paediatric cases where exposure to repeated radiation doses is of greater concern than in adults [46–48]. A further benefit of DCE-MRI is that anatomical images can also be obtained during the same imaging session, providing a direct comparisons to the observed physiological abnormalities (see Figure 4.1).

In order to obtain useful diagnostic/prognostic information, the dynamic change of pixel intensities as contrast agent flows through the metabolic region within the kidney must be quantified in terms of function such as blood flow, filtration rate or functional volume. For this purpose, a kidney template is required via segmentation.

Quantification of kidney function is often obfuscated by manual segmentation [50, 51] of the kidney ROI, (i.e., a region of kidney is selected as a template by human experts by manually delineating the kidney ROI). Semi-automatic methods, however, work by specifying the target ROI a-priori to an automated segmentation algorithm [52]. Although these approaches are potentially correct, the major issue is the need for human intervention in the segmentation process of the target region. In addition, this can be labour intensive, time-consuming and inefficient as the human expert has to examine the whole sequence of images to find the most suitable frame. This is typically also inconvenient since human experts require proprietary software for delineating the ROI, and also error-prone as the selection of the template (ROI) is subjected to observer variations [53]. Automated methods [54, 55] have the potential to overcome these limitations and moreover offer a more reproducible approach.

Previously, automated segmentation methods utilising clustering and classification methods such as k-means clustering [50] and k-nearest neighbour classification [56] have been suggested. These methods work by considering the signal intensity values across the images in time, thus obtaining a high-dimensional feature vector in each voxel based on the actual tissue response before and after injecting the contrast agent. Methods based on active contours [57] and related methods
4.1. INTRODUCTION

Figure 4.1: DCE-MRI image with anatomical parts: kidney, liver and spleen. Renal perfusion takes place inside the kidney regions after the injection of the contrast agent. The kidney function is then quantified by calculating the mean pixel intensity values inside the kidney region (in practice using a tracer-kinetic method [49]). For this purpose, a proper segmentation of kidney region is required.

have also offered solutions that consider the region boundary coupled with shape constraints [58]. A recent work by Hodneland et al. in 2014 applied the temporal tissue response and minimal boundary length as shape information for obtaining the kidney segmentation [59] in 4D DCE-MRI videos. Zollner et al. [43] introduced the Independent Component Analysis (ICA) technique for functional segmentation of human kidney ROI in DCE-MRI recordings. An approach based on spatio-temporal ICA (STICA) [60] was also developed recently that offers a fully data-driven approach exploiting the distribution of the properties of the spatial data incrementally in the direction of the time axis. A major limitation for this ICA based approach is finding an optimal filter that maximises the statistical independence of the observed signals. The ICA method is typically also approached with a substantial number of assumptions/heuristics and is computationally expensive.

However, absolute quantification inside the kidney via segmentation is also often obfuscated by complex patient movements, arising due to respiration, pulsation and involuntary movements as shown in Figure 4.2 (top)) (the corresponding dynamic sequence can be viewed at https://youtu.be/TWq34TFGNcU). These movements induce pixel displacements in and around the kidney region, leading to motion artefacts (intensity fluctuations) in the time-intensity curves produced from a fixed region of interest (ROI) placed within the kidney. Such motion artefacts can affect
the assessment of the kidney function. Therefore, prior to segmentation, kidneys in DCE-MRI should undergo a proper movement correction.

![Figure 4.2: (Top) 5 frames at time = \{30, 50, 74, 82, 100\}s selected from a DCE-MRI sequence of a healthy volunteer. The yellow and red reference lines show the alignment of the kidney and liver regions. The first image shows the peak stage of contrast agent inside the kidney region. The regions of kidney and liver clearly depict the translation of movements in the vertical direction, arising due to the patient movements. (Bottom) The yellow and red reference lines clearly showing the proper alignment of the kidney and liver regions after processing with WR-DMD. The video based results can be viewed at https://youtu.be/TWq34TFGNcU and https://youtu.be/UT7f4ch4H-I.](image)

Senneville et al.’s [61] work assumes that the kidney is a rigid body and its shape does not change during the MRI data acquisition. For registration purposes in Senneville et al.’s work, a human expert selects a reference image from the image sequence. The expert then manually delineates the kidney region of interest (ROI), the ‘template’. Using this template the registration of the kidneys is conducted across the DCE-MRI sequence using contrast invariant similarity matching. Other methods that require human input are reported in [62–65]. Although these approaches are potentially effective, a major issue is the need for human intervention for delineating process of the kidney region. Other issues are reproducibility and the intrinsic bottle-neck associated with the speed of processing that automation could address.

On the other hand, approaches based on matrix decomposition such as Principle Component Analysis (PCA), Independent Component analysis (ICA) and Robust PCA (RPCA) have also been proposed as a preprocessing step prior to image registration. Progressive principal component registration (PPCR) introduced by Melbourne et al. [66,67] is a PCA based approach that iteratively removes misalignment from the DCE-MRI sequence whilst using a standard registration algorithm such as fluid registration [68]. The authors’ main assumption in this work is that
PCA captures contrast changes or intensity fluctuations in the first few principal components and motion in the last principal components (when sorted according to their proportion of variance explained through the cumulative sum of eigenvalues). However, using PCA for motion compensation depends purely upon the nature of the motion i.e., for example, periodic motion of free breathing can appear in the first few principal components along with contrast changes. In order to deal robustly with various breathing protocols, Robust Data Decomposition Registration (RDDR) [69] was introduced. RDDR uses robust principal component analysis (RPCA) [70] coupled with a registration algorithm based on residual complexity minimisation [71]. RPCA decomposes the DCE-MRI data into a series of low rank and a sparse components separating motion components from the contrast enhanced ones. The intensity fluctuations which remain unchanged are then registered. The explicit separation of sparse components provide RPCA a greater degree of robustness when compared to a regular PCA based approach. The ICA based approach has also been used to decompose DCE-MRI data prior to registration in free breathing cardiac MRI [72]. In all of the aforementioned approaches the main objective is to remove motion elements from the DCE-MRI time-series whilst utilising image registration methods.

Spatio-temporal ICA (STICA) [60,73] is one method that does not use any kind of registration procedure in its approach. According to STICA’s assumptions, free breathing, which induces the movement artefacts is regarded as one of the independent processes; i.e., different regions in DCE-MRI that respond differently with respect to the contrast agent are assumed to be spatially independent, and are assumed to be temporally independent of each other [60]. Quantitative assessment in [60] using ROI analysis shows virtually no movement in either the first independent component or the second. The third independent component shows the movement artefacts. Limitations of this approach can include finding an optimal filter that can maximise the statistical independence of the these DCE signals over space and time simultaneously.

The assumptions of our approach are borrowed from PCCR and RDDR methods. Similar to STICA our proposed WR-DMD technique is also an image registration free approach. Comparisons with the aforementioned decomposition based methods are made in Table 4.1.

4.1.1 Our approach

Our approach is a three fold approach: (i) first, we perform movement correction of DCE-MRI sequence. (ii) Second, we obtain segmentation of the kidney ROI. (iii) Using the obtained template ROI, kidney function is thus quantified as a mean of pixel intensity values.
### Table 4.1: Comparison of proposed methods with other movement correction approaches that are based on matrix decomposition methods.

<table>
<thead>
<tr>
<th>Authors</th>
<th>Method</th>
<th>Matrix Factorisation</th>
<th>Registration</th>
<th>Assumptions</th>
<th>Demerits/merits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Melbourne et al. [66,67]</td>
<td>Progressive Principal Component Registration (PPCR)</td>
<td>PCA</td>
<td>Multi-resolution FFD [74]</td>
<td>Contrast changes captured in the first few principal components and motion in the last principal components.</td>
<td>Limitations in handling periodic motion of free breathing. Computationally expensive for using registration algorithm.</td>
</tr>
<tr>
<td>Hamy et al. [69]</td>
<td>Robust Data Decomposition Registration (RDDR)</td>
<td>RPCA</td>
<td>Residual complexity minimisation [71]</td>
<td>Sparse components separate motion components from the contrast enhanced images.</td>
<td>Computationally expensive. Nevertheless robust to various breathing protocols.</td>
</tr>
<tr>
<td>Kiani et al. [60]</td>
<td>Online - STICA</td>
<td>ICA</td>
<td>-</td>
<td>Different regions in DCE-MRI respond differently, over the time are assumed to be independent.</td>
<td>Computationally expensive for optimising filters that can maximise the statistical independence.</td>
</tr>
<tr>
<td>Proposed method</td>
<td>WR-DMD</td>
<td>DMD</td>
<td>-</td>
<td>Sparse components separate motion components from the contrast enhanced images. Contrast changes captured in the most significant dynamic modes and motion in the least significant.</td>
<td>W-DMD disentangles periodic free breathing. DMD separates motion components in least significant modes. R-DMD reconstructs perfectly aligned sequence with the most significant modes. Computationally inexpensive.</td>
</tr>
</tbody>
</table>
4.1 INTRODUCTION

Movement Correction

Our approach for movement correction is a two step process (Figure 4.3), where at first the DCE-MRI sequence is processed through window-DMD method [75] to compensate for the pseudo-periodic breathing motion (the importance of running W-DMD as a first step process is shown in Section 4.4.3). The windowed version of DMD method (here, W=3) runs over three consecutive images (since motion is periodic for every three images as evaluated in Section 4.4.1) in an overlapping fashion as shown in Figure 6.5.

The output of DMD at each window produces two images namely W-DMD component-1 (C1) revealing the low rank image and W-DMD component-2 (C2) revealing the sparse component. At this stage of our approach we discard the sparse components i.e., W-DMD(C2)s to compensate the pseudo-periodic free breathing motion from the DCE-MRI sequence. Second, we proceed with giving W-DMD component-1 as an input to standard DMD algorithm, which decomposes the W-DMD(C1) sequence into several images called dynamic modes (see, Figure 4.9). The dynamic contrast changes are captured in the most significant modes, and motion components are captured in the least significant modes. Using the first three significant DMD modes, the original sequence is then reconstructed via R-DMD method. Our result in Figure 4.2(bottom) shows that the motion artefacts are compensated in an exemplar 4D dynamic medical imaging application.

Functional Segmentation

The W-DMD component-1 (W-DMD(C1)) when given as an input to standard DMD algorithm, decomposes the sequence into several images called dynamic modes. The dynamic contrast changes are captured in the most significant modes revealing the key relevant organs in the DCE-MRI sequence. The second, most significant dynamic mode that highlights the kidney region, is coupled with thresholding and connected component analysis to obtain the functional segmentation of the kidney ROI. This ROI, as a template, is projected onto every DCE-MRI image in the sequence and mean pixel intensity of the kidney is then quantified.

4.1.2 Contributions

Our implementation is novel in the sense that it uses windowed-DMD on dynamic image sequences for the first time to compensate for motion artefacts via producing low rank images as W-DMD component-1. Even-though the low-rank and sparse representations of an image sequence have been reported for DMD [76], the method that we propose here is essentially different. In [76], the authors exploit the low-rank and sparse representation within each frame. Specifically, low-rank revealing the background and sparse presenting the foreground of that particular frame. The original DMD method extracts modes from a sequence of images and interprets modes in the image space, whereas our reconstruction variant of the method re-
projects the DMD modes back to original image sequence, thereby, stabilising the complex movements. The novel DMD functional segmentation framework is thus our another contribution, with application to exemplar 4D dynamic medical image sequences containing kidneys. We validate the implemented framework on real medical images as well as synthetically generated image sequences in order to assess the segmentation of key structures. We also show that extracting relevant key organs from the DCE-MRI images is computationally efficient.

The contributions of this chapter can be listed as follows: (i) Introducing WR-DMD framework for the first time to reconstruct aligned dynamical medical image sequences. (ii) Coupling DMD modes with thresholding and blob analysis for segmenting kidney ROI. (iii) Validating our techniques using medical data with applications to DCE-MRI; and (iv) Improving the understanding of the application through WR-DMD framework.

It is important to note that modes in the DMD are generally complex (i.e. containing real and imaginary parts); we consider only the absolute value of the complex modes in this chapter as a modified approach.

The following research questions are answered in this chapter:

1. How can DMD be used for quantifying a kidney function in Dynamic Contrast Enhanced Magnetic Resonance imaging (DCE-MRI) sequences?

2. How well can DMD tackle the movement correction problem present in those dynamical sequences?

3. How well can DMD highlight the contrast agent in one of its dominant modes? Would the highlighted contrast agent in the dominant mode provide a kidney template for segmentation?

4. How well DMD performs when compared to human experts as well as computational algorithms?

4.1.3 Organisation

The remainder of this chapter is organised as follows: in Section 4.2, we consider the theory for WR-DMD. Section 4.3 presents ten different datasets used in this study. In Section 4.4 we present our experiments and results and finally, conclusions are drawn in Section 4.5.

4.2 Methodology

In this section, we present our methodological pipeline which consists of windowed DMD (W-DMD) and reconstruction based DMD (R-DMD). In addition, this section also presents thresholding, selection of kidney regions using connected component analysis and finally modelling of the kidney function. The overall process pipeline
Figure 4.3: Flow chart showing the steps involved in the methodological framework. First, a DCE-MRI sequence consisting of $N$ images is processed using the W-DMD algorithm in order to output each $N - 2$ W-DMD components $C_1$ and $C_2$. At this stage the W-DMD($C_1$) produces the low rank images and W-DMD($C_2$) produces sparse images. W-DMD($C_1$) is given as an input to DMD which produces $N - 3$ DMD modes. On one hand, the first 3 DMD modes are then selected for reconstructing the motion stabilised image sequence. On the other hand, we select a dynamic image corresponding to mode 2, where, thresholding is performed and converted to binary image. After binarisation, the largest area with connected pixels is selected as a ROI template. Finally, the produced template is projected onto motion corrected DCE-MRI data to compute the kidney function.

is shown in Figure 4.3. At the end of this section, we also present our evaluation criteria for both: (i) movement correction via mean motion magnitude computed through block matching algorithm, and (ii) functional segmentation via the Jaccard similarity coefficient.
4.2.1 Dynamic Mode Decomposition (DMD)

In a dynamic sequence of \( N \) images \( X \), let \( \bar{x}_r \) be the \( r^{th} \) column vectorised image (size of \( \bar{x}_r \) is \( mn \times 1 \) whose initial size was \( m \times n \) in the image domain). For \( N \) images in the sequence this results in the construction of a data matrix \( X \) of size \( mn \times N \) for \( N \) images. This data matrix containing DCE-MRI image sequence is then solved through DMD via SVD approach to obtain the dynamic modes. However, the modes obtained using the procedure discussed in Chapter 2 are not ordered or ranked. Therefore, to obtain the most stable (significant) DMD modes, the complex DMD eigenvalues are to be ranked, which is discussed in the following section 4.2.2.

4.2.2 Ordering Dynamic modes

In order to select the most significant dynamic modes we calculate the phase-angles for the complex eigenvalues. The eigenvalues are the diagonal elements of \( \sigma \) i.e., \( \text{diag}(\sigma) \). The absolute value for the phase-angles are calculated and modes with unique phase-angles are selected. Doing this will remove one of the conjugate pairs in the dynamic modes. These conjugate modes have same phase-angles but with different signs and look and capture similar information [77]. After discarding one of the conjugate pairs, the dynamic modes are then sorted in ascending order of their phase-angles [78]. The resultant dynamic modes are thus sorted according to their significance. In this study we have considered the first three significant dynamic modes when reconstructing the original sequence.

4.2.3 Reconstruction from DMD modes (R-DMD)

The novel reconstruction DMD aims at reconstructing the image sequence from the dynamic modes. This can be achieved in a least squares solution.

\[
\left\| \Psi - \hat{P}_2 \Sigma V^{-1} \omega \right\| (4.1)
\]

Therefore, the reconstruction of the original image sequence can be formulated as follows:

\[
\hat{P}_2 = \Psi \omega^{-1} \Sigma V^{-1} (4.2)
\]

If the contrast changes are captured in the most significant modes and motion components in the least significant ones, it is desirable to discard the least significant modes. The crux of making this work is to select the \( K \) modes that are contributing to the contrast changes; and not the motion changes.

The original sequence is thus constructed using first \( k \) significant significant modes along with their corresponding eigenvectors from \( \tilde{H} \).

\[
\hat{P}_2 = \Psi_{\{1..k\}} \omega_{\{1..k\}}^{-1} \Sigma V^{-1} (4.3)
\]
4.2. METHODOLOGY

Figure 4.4: Methodological pipeline showing the working mechanism of W-DMD. DMD runs over the window containing first three images in the sequence, obtaining two dynamic modes. The first dynamic mode ’c1’ capturing the low rank image across the window and second dynamic mode c2 capturing the sparse representation. The next step exclude the first image and consider images {2, 3, 4}, followed by {3, 4, 5} and {4, 5, 6} producing c1 and c2 components. Finally, all of the c1s and c2s across all the windows are concatenated to obtain W-DMD component-1 (W-DMD (C1)) and W-DMD component-2 (W-DMD (C2)).

4.2.4 Windowed-DMD (W-DMD)

The Windowed version of DMD method runs DMD over a window of consecutive images in a sequence in an overlapping fashion. The output of DMD at each window produces $W - 1$ dynamic modes where $W$ is the length of the window. To compensate the periodic free breathing from the DCE-MRI sequence in this study we consider $W = 3$. For instance, running DMD on the window containing the first three images in the sequence, we obtain two dynamic modes. The first dynamic mode ‘c1’ captures the low rank image across the window and the second dynamic mode ‘c2’ captures the sparse representation, which essentially contain motion artefacts pertaining to periodic free breathing. In the next step we exclude the first image and consider images {2, 3, 4}, followed by {3, 4, 5} and {4, 5, 6} and so on as shown in Figure 4.4. Finally, we concatenate all of the c1s across all the windows to obtain W-DMD component-1 (W-DMD (C1)) and similarly concatenation of the c2s produce W-DMD component-2 (W-DMD (C2)) [75].
4.2.5 Evaluation criteria

In this section, we present the evaluation criteria for our approaches in movement correction as well as in functional segmentation.

**Movement Correction**

The evaluation is based on two perspectives.

1. From the clinical perspective, we would like to examine the feasibility of using our approach as a means for removing respiratory motion artefacts from the dynamic image sequence containing dramatic regional changes in intensity due to contrast agent flow affecting the quality of the resulting time intensity curves used for analysis. The curves are produced by calculating the mean intensity of the target ROI in an image, which in this case is based on the kidney. ROI analysis using time-intensity curves containing respiratory motion artefacts may affect subsequent compartmental model fitting, as the motion may also obscure subtle time-intensity features. Therefore, we have considered the smoothness of the time intensity curve as a surrogate metric for quality of motion compensation. The smoother this curve is, the better the performance.

2. From the signal processing and computer vision perspective, we would like to examine the strength of our approach by calculating the mean motion magnitude across the DCE-MRI sequence. Since, the respiratory motion represents an obfuscating issue within the dynamic image sequence, it distracts attention away from, and potentially masks, areas that may exhibit subtle pathology within the image. For this purpose we evaluate the motion between two consecutive images over the entirely reconstructed dynamic sequence in order to characterise the amount of motion as an indication of the stability of the dynamic sequence. This criteria is evaluated using a method called block matching motion estimation [79]. A well reconstructed dynamic sequence should have a smaller overall global mean.

Block-matching algorithm estimates the motion between two images using “blocks” of pixels i.e., by matching the block of pixels in image $K$ to a block of pixels in image $K'$ by moving the block of pixels over a search region. The block subdivides the image $K$ in block sizes [height width] and Overlap [r c] parameters. For each subdivision or block in image $K'$, the algorithm establishes a search region based on the maximum displacement [r c] parameter. The block searches for the new block location using an exhaustive search method [80]. We have considered [height, width] = [15, 15] and [r,c] = [5, 5].
Functional Segmentation

To evaluate the performance of our segmentation approach, we use Jaccard similarity coefficient, a standard measure of similarity between finite sample sets [81] (here, in particular, the similarity between the two segmented sets). It is defined as the size of the intersection (of pixels values) between the segmented sets divided by the size of the union of the segmented sets:

\[ J(A, B) = \frac{|A \cap B|}{|A \cup B|} \]  \hspace{1cm} (4.4)

Here A and B are two segmented images. Values of the Jaccard index range from 0 to 1. (0 if the intersection of the two sets is empty; 1 if the two sets are equal; the more similar the sets are, the closer to 1 is the metric).

Let \( e_i \) and \( d \) \( \forall i \in \{1, 2, 3\} \) be the segmented results from the three human experts and DMD framework respectively. For an unbiased evaluation criterion, let us calculate three different ground-truths \( G_i \) based on the segmentation results \( e_i \) obtained from the human experts.

\[
G_1 = (e_2 \cap e_3), \\
G_2 = (e_1 \cap e_3) \& \\
G_3 = (e_2 \cap e_1).
\] \hspace{1cm} (4.5)

The ground-truth for the human expert-1 is the mutual agreement of segmentation results from expert-2 and expert-3. Similarly the ground-truth for expert-2 can be calculated as the mutual agreement of expert-1 and expert-3 and vice versa.

The evaluation criteria \( E_i \) for the human experts can be calculated as follows:

\[
E_i = J(e_i, G_i) = \frac{|e_i \cap G_i|}{|e_i \cup G_i|} \ \forall \ i \in \{1, 2, 3\}. \hspace{1cm} (4.6)
\]

The evaluation \( D \) for the DMD framework is given by the average of \( D_i \) \( \forall i \in \{1, 2, 3\} \):

\[
D = \frac{1}{3} \sum_{i=1}^{3} D_i. \hspace{1cm} (4.7)
\]

where,

\[
D_i = J(d, G_i) = \frac{|d \cap G_i|}{|d \cup G_i|} \ \forall \ i \in \{1, 2, 3\}. \hspace{1cm} (4.8)
\]

4.3 Dataset

The datasets obtained were from ten healthy volunteers’ as shown in Figure 4.5 acquired after injection of 0.05 mmol/kg of Gd-DTPA (Magnevist) contrast agent, on a 1.5T Siemens Avanto MRI scanner, using a 32 channel body phased array coil. The MRI acquisition sequence consisted of a 3D spoilt gradient echo sequence utilising an Echo time, TE of 0.6ms, Repetition time, TR of 1.6ms, and a flip angle,
FA = 17 degrees with a temporal resolution of 1.5s collected for 180s. The acquired DCE-MRI datasets cover the abdominal region, enclosing left and right kidneys and abdominal aorta.

Kidney Research UK, funded the DCE-MRI data acquisition as part of a reproducibility study. The data were collected at the Institute of Child Health, University College London in collaboration with Prof. Isky Gorden.

4.4 Experiments and Results

In this section we discuss our experimental procedure along with the results. The objectives of our experiments are as follows:

1. Selection of optimal window length ($W$): The W-DMD algorithm needs parameter $W$ i.e the window length to run standard DMD. The objective of this experiment is to determine the optimal $W$ which depends on the breathing cycle of the volunteer.

2. Importance of W-DMD method: To show the importance of W-DMD which constitutes a simplified variant of the proposal in our two step approach, we evaluate our results with and without using the W-DMD method. Since, the W-DMD method can compensate for the pseudo-periodic free breathing
from the contrast enhanced images by discarding sparse components, we conjecture that without using it, the reconstruction results may contain some flickering effects. Therefore, we wish to examine whether using windowed DMD on a sliding sequence of 3 consecutive images will really stabilise these breathing motion artefacts.

3. **Qualitative assessment of dynamic information captured by DMD:** From the outset, we conjecture that DMD can capture both large scale dynamics related to contrast changes and small scale dynamics related to motion artefacts and noise. Therefore, we wish to examine whether these features are indeed picked up by DMD or not.

4. **Reconstruction using most significant modes:** Since DMD can capture the contrast enhanced images in the most significant dynamic modes and motion in the least significant, we conjecture that reconstruction with the top three most significant DMD modes could provide a perfectly aligned image sequence.

5. **Effect of mode selection in the reconstruction:** It is of interest to investigate the impact of the most significant dynamic modes on the generalisation performance of the stability within the image sequence. For this reason, we consider modes in $M \times 4$ as a set where $M$ ranges from 1, $\cdots$, 16. We conjecture that stability within the image sequence should decrease with an increasing number of modes selected for reconstruction.

6. **Comparison with other registration methods:** We wish to examine the performance strength of our approach by comparing with two registration methods from the literature.

7. **Segmentation:** Due to the injection of contrast agent, large scale intensity fluctuations can be observed inside the kidney region which also has minor impact on the liver region. We therefore wish to examine whether these features are indeed picked up by DMD.

### 4.4.1 Selection of optimal $W$ (window length)

In order to determine optimal $W$ we would like to see if there exists any periodicity in the motion. For this purpose we calculate the motion amongst the images in the sequence with respect to the first frame using the block matching algorithm \cite{79}. The motion magnitude across the dataset-1 with respect to first frame of the sequence are shown in Figure 4.6. It can be clearly seen that the motion is periodic for every three frames. Therefore, we have chosen a window length of 3 to conduct our experiments.
4.4.2 Performance of W-DMD

Each of the ten healthy volunteers’ DCE-MRI dynamic sequences consisting of 120 images are given as an input to W-DMD algorithm sequentially. The DMD algorithm at each sliding window consisting of 3 images produces two dynamic modes namely W-DMD component-1 revealing the low rank image and W-DMD component-2 revealing the sparse image as shown in Figure 4.7. All the low rank images are then concatenated to form a sequence of 118 images called W-DMD component-1 (W-DMD(C1)) and similarly the concatenation of sparse images form W-DMD(C2).

For the purpose of clinically relevant evaluation, an expert (radiologist) manually delineated the kidney ROIs across the 10 datasets as shown in Figure 4.5. Using these ROIs the time-intensity curves are obtained. The curves for W-DMD(C1) for the right kidney shows a little reduction in intensity variations as seen in Figure 4.8(W-DMD (red)). The evaluation results after processing with block matching algorithm depicts a decrease of 80.54% average motion magnitude against the original sequences across the 10 datasets as shown in Figure 4.10.
4.4. EXPERIMENTS AND RESULTS

Figure 4.7: (Top) 5 images from dataset-1 W-DMD(C1) at time = \{30, 50, 74, 82, 100\}s showing the low-rank images. The first image shows the peak stage of contrast agent inside the kidney region. (Bottom) Corresponding images from the W-DMD(C2) showing their sparse representation.

4.4.3 Importance of W-DMD

In order to examine the strength of W-DMD, we directly input each of the ten healthy volunteers’ DCE-MRI dynamic sequences to the standard DMD algorithm. Using the top 3 most significant modes, the original sequence is reconstructed using our R-DMD method. The results in Figure 4.10 show greater motion magnitude when compared to results obtained through W-DMD on the datasets \{1, 4, 9, 10\}. The time intensity plots in Figure 4.8 from datasets \{2, 3, 5, 6, 7, 9\} reveal a greater amount of fluctuations from the graphs produced by the R-DMD method even though the results of their mean motion magnitude are lower when compared to W-DMD method. This proves our conjecture that although excluding the W-DMD step stabilises the motion globally, the periodic free breathing would still remain locally. Therefore, discarding the sparse components i.e., W-DMD(C2) eliminates periodic free breathing from the contrast enhanced images.

4.4.4 Qualitative assessment of DMD

In the next step, the W-DMD(C1) containing the low-rank images are then given as an input to DMD algorithm producing 117 dynamic modes. The first mode reveals the low-rank model across all the images and the remaining 116 modes capture the sparse representations. The contrast changes are captured in the most significant modes, in particular, mode-2 capturing kidney region and mode-3 and 4, spleen and the liver regions respectively as shown in Figure 4.9(top) for dataset-1.

Noise and residuals including the motion components are captured in the least significant modes (Figure 4.9(bottom)).
Figure 4.8: Time intensity curves across 10 datasets (sequentially from left to right.)
Figure 4.9: (Top) The top 10 most significant DMD modes on W-DMD(C1) from dataset-1. (Bottom) bottom 10 least significant DMD modes on W-DMD(C1).
4.4.5 Reconstruction using most significant modes

The first 3 modes are selected for the reconstruction of the W-DMD(C1) sequence of images, discarding the rest of the 113 modes. The reconstructed W-DMD(C1) using the WR-DMD algorithm produces stable image sequence compensating for the complex movements. The qualitative results obtained from the WR-DMD (black) in Figure 4.8 show smoother curves in the time-intensity plots when compared to the original sequence and W-DMD(C1) sequence. Complete complex movement artefacts arising due to respiration, pulsation and involuntary movements are all compensated through the WR-DMD reconstructed images as shown in Figure 4.10. A decrease of 99% average motion magnitude can be seen against the original sequences across the 10 datasets.

![Figure 4.10: A smaller mean motion magnitude indicates a more stable sequence. Here M is the number of modes that were used in the reconstruction process. As the number of modes increase, the mean motion magnitude also increases.](image)

4.4.6 Effect of mode selection in the reconstruction:

Since reconstruction needs to operate with significant modes, it is of interest to find out the minimum number of modes required. We hypothesise that a greater number of modes should result in less stable performance; however, at the same time, we would like to know the minimum number of modes that are required to reconstruct the original sequence. Consequently, we select the windows of the first \{4, 8, \cdots 64\} images. We can consider at most 64 modes because one member of each conjugate pairs of the DMD mode is redundant. Consequently, we are left with 61 \sim 62 modes. The results in Figure 4.11 reveals that as the number of modes increases, the mean motion magnitude also increases, consistent with our conjecture that a greater number of modes results in increased motion variance (methodologically, we
would expect the optimal number of DMD modes to be data dependent).

![Figure 4.11: Mean motion magnitude across 10 datasets calculated using block matching algorithm utilising WR-DMD.](image)

### 4.4.7 Comparison with other registration methods

In order to compare our approach with registration based methods, we should follow a gold standard approach that is manually delineating the target region and performing the rigid registration, which requires a human expert and is inconvenient. Nevertheless, to obtain a fair comparison, a human expert manually delineated the area around the right kidney region. Therefore, we have selected only this part of the delineated region for the registration and not the whole image. We have opted to use intensity based methods such as affine and translation registration methods for our comparison, since these methods are less sensitive to contrast (intensity) change, and their implementation is freely available.

Evaluating the performance of our approach on DCE-MRI data is difficult due to the lack of the prior knowledge on contrast change and the motion. We evaluate the total variation of the time intensity curves using a “degree of smoothness” measure. This measures local variability, that depends on the motion induced fluctuations. The degree of smoothness can be calculated as the ratio between the standard

---

deviation and absolute mean of the derivative of the time intensity curve. The results in Figure 4.12 clearly show WR-DMD outperforms the standard registration methods.

### 4.4.8 Segmentation

The W-DMD(C1), DCE-MRI image sequences from the healthy volunteers’ data are given as input to DMD algorithm. The output of DMD is thus modally separated large and small scale fluctuations in voxel intensity. These modes are obtained by suppressing the stable information present within the DCE-MRI image sequences (with the exception of mode 1); DMD is thus able to suppress the background information from the image sequence insofar as this is stationary. Renal perfusion (due to the injection of contrast agent) inside the kidney region exhibits large scale voxel intensity fluctuations, followed, in order of magnitude, by liver and spleen region fluctuations respectively. DMD thus captures key kidney, liver and spleen regions in distinct modes as shown in Figure 4.13. Dynamic mode-1 captures the low-rank image of the DCE-MRI dataset-1 as shown in Figure 4.13(a) revealing the background function of the dataset. The kidney function is empirically observed
4.4. EXPERIMENTS AND RESULTS

Figure 4.13: DMD modes 1,2,4,5 and 83 from a health volunteer’s data (dataset-1). (a) DMD mode-1 showing the background image captured from the dataset. (b) DMD mode-2 showing predominantly the kidney region. (c) DMD mode-4 showing predominantly the spleen region. (d) DMD mode-5 showing predominantly the liver region. (e) DMD mode-83 showing the noise component from the dataset.

to be captured in dynamic mode-2 (b) and the spleen and liver functions captured in dynamic mode-4 (c) and 5 (d) respectively. The lower order modes reveal the noise components inside the dataset-1 as seen in Figure 4.13(e). These results hence further provide visual evidence for our conjecture that DMD is capable of isolating key functional regions such as kidney and liver. These rankings however are not given intrinsically by DMD (as discussed previously); we now look at how this can be accomplished.

The main aim in this study is to segment the kidney region so as to enable quantification of the kidney function. We thus select the second dynamic mode for this functional characterisation as this configuration is observed to produce kidney region segmentation, consistent with the above argument, across all of the datasets as shown in Figure 4.14 (top). These dynamic mode-2s are first normalised in the range of [0, 1]. Binarised images of dynamic mode-2s are then obtained by replacing all pixels in the image with intensity values greater than an adaptive threshold with the value 1 (white), and replacing all other pixels with the value 0 (black). These result in the binarised versions of dynamic mode 2 shown in Figure 4.14 (middle). Kidney templates are then automatically selected as the largest area of connected components of binarised dynamic mode-2 images across all the datasets as shown in Figure 4.14 (bottom). The connected components are selected by scanning the binarised image from top-to-bottom. All the pixels in a connected component are given a greedy label. Thus, all the pixels in the first connected component are labelled as 1 and those in the second as 2 and so on. These components are then ordered by decreasing area (i.e. the sum of all the pixels present in that particular component) and the label with the largest area is then selected as the kidney template. The produced kidney templates are then projected onto DCE-MRI images to obtain the kidney function (as shown in Figure 6.4 from Section 7.2) by calculating the mean intensity pixel values inside region specified by the template. The kidney functions across all the datasets obtained through the DMD framework are shown in Figure 4.15.
CHAPTER 4. QUANTIFYING KIDNEY FUNCTION: MOVEMENT CORRECTION AND SEGMENTATION OF DCE-MRI

Dynamic mode-2 across 10 datasets of DCE-MRI data

Binarisation across 10 datasets of DCE-MRI data

Kidney template across 10 datasets of DCE-MRI data

Figure 4.14: (Top) Image showing the kidney regions in Dynamic mode-2 across the 10 datasets used in this study. (Middle) Images showing the binarisation effect on Dynamic mode-2 across the datasets. (Bottom) Images showing the kidney template selected by the largest area of connected components.
4.4.9 Evaluation of the Segmentation

Evaluation of our framework is two-fold:

- Evaluation of the performance of the segmented kidney template.
- Evaluation of the performance of the quantified kidney function.

After obtaining the kidney ROIs (i.e., the templates), the Jaccard Similarity Measure is used to evaluate the performance of the proposed DMD framework with respect to the ground-truth data. The aim of this experiment is to benchmark the generalisation performance of our framework against expert annotations.

For an unbiased evaluation, three different ground-truths \( G_i \ \forall \ i \in \{1, 2, 3\} \) are calculated based on the three different segmentation results obtained from the human experts. The ground-truth \( G_1 \) for the human expert-1 is calculated as the mutual agreement of the segmentation between expert-2 and expert-3. Similarly, the ground-truth \( G_2 \) for expert-2 is calculated as the mutual agreement of segmentation between expert-1 and expert-3 and vice versa. We also have considered an additional baseline given by the minimum bounding box region around the human annotated regions (to simulate ‘blind’ a priori segmentation). The DMD framework is evaluated as the mean of JSC values from all three ground-truths. Table 4.2 shows the JSC values for the DMD framework against the three ground-truths \( G_i \ \forall \ i \in \{1, 2, 3\} \). The last column of the table presents the mean of the JSC values obtained from the three ground-truths across all the 10 datasets.

Table 4.2: Evaluation results for the DMD framework with respect to the three ground-truths \( G_i \ \forall i \in \{1, 2, 3\} \).

<table>
<thead>
<tr>
<th>Dataset</th>
<th>( G_1 )</th>
<th>( G_2 )</th>
<th>( G_3 )</th>
<th>avg(DMD)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dataset-1</td>
<td>0.8789</td>
<td>0.8672</td>
<td>0.8727</td>
<td>0.8729</td>
</tr>
<tr>
<td>Dataset-2</td>
<td>0.8291</td>
<td>0.8525</td>
<td>0.8758</td>
<td>0.8525</td>
</tr>
<tr>
<td>Dataset-3</td>
<td>0.8426</td>
<td>0.8628</td>
<td>0.8411</td>
<td>0.8488</td>
</tr>
<tr>
<td>Dataset-4</td>
<td>0.9038</td>
<td>0.9133</td>
<td>0.9279</td>
<td>0.9150</td>
</tr>
<tr>
<td>Dataset-5</td>
<td>0.8583</td>
<td>0.9027</td>
<td>0.8854</td>
<td>0.8821</td>
</tr>
<tr>
<td>Dataset-6</td>
<td>0.8651</td>
<td>0.8418</td>
<td>0.8386</td>
<td>0.8485</td>
</tr>
<tr>
<td>Dataset-7</td>
<td>0.8578</td>
<td>0.8239</td>
<td>0.8065</td>
<td>0.8294</td>
</tr>
<tr>
<td>Dataset-8</td>
<td>0.8927</td>
<td>0.9075</td>
<td>0.8861</td>
<td>0.8954</td>
</tr>
<tr>
<td>Dataset-9</td>
<td>0.8949</td>
<td>0.8702</td>
<td>0.8828</td>
<td>0.8826</td>
</tr>
<tr>
<td>Dataset-10</td>
<td>0.8356</td>
<td>0.8169</td>
<td>0.8799</td>
<td>0.8441</td>
</tr>
</tbody>
</table>

The performance of the kidney segmentation achieved by the DMD framework, the blind bounding-box region and three domain experts with respect to their ground-truths are reported in Table 4.3. The JSC values achieved by expert-2 are generally high in comparison to the other experts. The DMD framework has achieved higher JSC values 0.91, 0.89 and 0.88 for the datasets 4 and 8 and 9. The Overall average JSC values for the experts are 0.85, 0.87 and 0.87 and for the DMD.
framework is around 0.86 while for minimum bounding-box region the JSC is below 0.04.

Table 4.3: Jaccard similarity coefficients for kidney segmentation achieved by the DMD framework, blind bounding-box region and three domain experts with respect to the ground-truth.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>DMD</th>
<th>Expert-1</th>
<th>Expert-2</th>
<th>Expert-3</th>
<th>Bbox</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dataset-1</td>
<td>0.8729</td>
<td>0.8642</td>
<td>0.8994</td>
<td>0.8974</td>
<td>0.5891</td>
</tr>
<tr>
<td>Dataset-2</td>
<td>0.8525</td>
<td>0.8718</td>
<td>0.8925</td>
<td>0.8906</td>
<td>0.4563</td>
</tr>
<tr>
<td>Dataset-3</td>
<td>0.8488</td>
<td>0.8642</td>
<td>0.8781</td>
<td>0.8719</td>
<td>0.4298</td>
</tr>
<tr>
<td>Dataset-4</td>
<td>0.9150</td>
<td>0.8880</td>
<td>0.8894</td>
<td>0.8894</td>
<td>0.4073</td>
</tr>
<tr>
<td>Dataset-5</td>
<td>0.8821</td>
<td>0.8668</td>
<td>0.9064</td>
<td>0.8934</td>
<td>0.3162</td>
</tr>
<tr>
<td>Dataset-6</td>
<td>0.8485</td>
<td>0.8589</td>
<td>0.8679</td>
<td>0.8536</td>
<td>0.3016</td>
</tr>
<tr>
<td>Dataset-7</td>
<td>0.8294</td>
<td>0.8548</td>
<td>0.8833</td>
<td>0.8529</td>
<td>0.2692</td>
</tr>
<tr>
<td>Dataset-8</td>
<td>0.8954</td>
<td>0.8155</td>
<td>0.8726</td>
<td>0.8708</td>
<td>0.2984</td>
</tr>
<tr>
<td>Dataset-9</td>
<td>0.8826</td>
<td>0.8221</td>
<td>0.8711</td>
<td>0.8391</td>
<td>0.2624</td>
</tr>
<tr>
<td>Dataset-10</td>
<td>0.8441</td>
<td>0.8426</td>
<td>0.8292</td>
<td>0.8792</td>
<td>0.2455</td>
</tr>
<tr>
<td>Average</td>
<td>0.8671</td>
<td>0.8548</td>
<td>0.8789</td>
<td>0.8738</td>
<td>0.3575</td>
</tr>
</tbody>
</table>

Figure 4.15 shows the kidney functions produced by the DMD framework, ground-truth ($G_i \forall i \in \{1, 2, 3\}$), three domain experts as well as kidney function produced by blind bounding-box over the kidney region. The results in these figures show that kidney function quantified by DMD is closely aligned with the experts annotation.

Evaluation for the kidney functions is calculated based on the Mean Square Error (MSE) criteria as shown in Table 4.4.

Table 4.4: Mean Square Error (MSE) based evaluation of the kidney functions for DMD, human experts and the bounding box.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>avg(DMD)</th>
<th>Expert-1</th>
<th>Expert-2</th>
<th>Expert-3</th>
<th>avg(Bbox)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dataset-1</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0000</td>
<td>0.0029</td>
</tr>
<tr>
<td>Dataset-2</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0000</td>
<td>0.0126</td>
</tr>
<tr>
<td>Dataset-3</td>
<td>0.0002</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0000</td>
<td>0.0082</td>
</tr>
<tr>
<td>Dataset-4</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0085</td>
</tr>
<tr>
<td>Dataset-5</td>
<td>0.0001</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0166</td>
</tr>
<tr>
<td>Dataset-6</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0188</td>
</tr>
<tr>
<td>Dataset-7</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0000</td>
<td>0.0162</td>
</tr>
<tr>
<td>Dataset-8</td>
<td>0.0001</td>
<td>0.0002</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0431</td>
</tr>
<tr>
<td>Dataset-9</td>
<td>0.0000</td>
<td>0.0001</td>
<td>0.0001</td>
<td>0.0000</td>
<td>0.0176</td>
</tr>
<tr>
<td>Dataset-10</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0002</td>
<td>0.0001</td>
<td>0.0213</td>
</tr>
</tbody>
</table>
Figure 4.15: Kidney function produced by blind bounding-box region, DMD framework, three domain experts, and ground-truths for datasets 1-10.
4.5 Summary and Conclusions

This study shows the significance of WR-DMD approach as a viable movement correction algorithm for processing DCE-MRI data. In addition, we also have demonstrated that the functional segmentation algorithm, with thresholding-binarisation and selection of the largest area of connected pixels effectively to quantify kidney function in DCE-MRI data. We applied the DMD framework to ten sets of DCE-MRI data collected from healthy volunteers. The proposed framework has very low time complexity (i.e., image sequence containing 120 frames each of size 120 × 104 takes about 3 to 5s to compute on an 8GB desktop computer). In addition, compared to existing methods, eg. RDDR and STICA, it has the advantage of requiring no parameter tuning. W-DMD can extract low-rank and sparse representations within an image sequence. The motion artefacts for periodic free breathing are captured in the sparse components. The low-rank W-DMD component-1 is then given as an input to the standard DMD algorithm producing dynamic modes. We found that the contrast changes are captured in the most significant dynamic modes and motion in the least significant ones. The original sequence is then reconstructed utilising the top three most significant dynamic modes using R-DMD. The results demonstrate that the proposed WR-DMD method is a promising approach for correcting respiratory or similar motions in complex dynamic medical image sequences containing significant temporal intensity changes due to contrast agent uptake or other comparable mechanisms.

The major point of discussion would be in answering whether the WR-DMD framework be sufficiently robust when it comes to scanning patients who tend to be more susceptible to motion artefacts? from the clinical point of view, this can only be answered with a conjecture as we have no patient data but we would argue ‘yes’ [61]. This question provides a good opportunity to recognise that both volunteers and patients will be susceptible to motion. The major area where this would have application is in repeat measurements of children with chronic renal issues. This requires repeat monitoring/scanning, and such children have, sadly, been subjected to many interventions and procedure. As a result, these might be expected to be highly compliant shallow-breather subjects and where the amount of irregularity in breathing might be less than the general healthy population. Another aspect is that in paediatrics, the amount of actual displacement may be expected to be about half that of a fully grown adult, as the entire body is smaller. How much this affects the relatively smaller kidney is a matter for conjecture and/or further investigation. From computer vision and signal processing point of view, W-DMD, in this study has been demonstrated for removing pseudo periodic breathing motion. It is always necessary to check if there exists any pseudo periodic breathing in the dataset. For this purpose motion amongst the images in the sequence with respect to the first image should be calculated using block matching algorithm as discussed in Section 4.4.1. Later the number of images for which the motion is periodic should be determined. For example, in Section 4.4.1 we show that the motion was periodic for every three images in our datasets and hence we set $W = 3$ as a
parameter in W-DMD to conduct our experiments. Similarly if the periodicity in
the motion is observed for ‘n’ images for a patient/ healthy volunteer, especially
children, who tend to be more susceptible to motion artefacts, we conjecture that
making the window length adaptive to ‘n’ might be sufficient to tackle the pseudo
periodic motion.

DMD also has demonstrated to extract local variations as a low rank repre-
sentation within an image sequence, as well as capturing dominating regions of
causally-connected intensity fluctuations. In our context, perfusion inside the kid-
ney region is the most dominating region of intensity fluctuations due to the injection
of contrast agent. DMD was thus able to naturally capture the kidney region as
mode-2. Using a simple thresholding technique and selecting the largest area with
connected pixels then automatically generates the proposed kidney region. The
proposed framework is extremely promising in obtaining functional segmentation in
general and thus to quantifying segmented functions, in particular the kidney func-
tionality. The results when compared with expert annotations and ground-truth
clearly shows the strength of the framework suggesting that manual selection of
kidney region is no longer needed and that the entire process can be automated.
Chapter 5

2D Facial Spoof Detection

Rendering a face recognition system robust is vital in order to safeguard it against spoof attacks carried out by using printed pictures of a victim (also known as print attack) or a replayed video of the person (replay attack). A key property in distinguishing a live, valid access from printed media or replayed videos is by exploiting the information dynamics of the video content, such as blinking eyes, moving lips, and facial dynamics. We advance the state of the art in facial anti-spoofing by applying Dynamic Mode Decomposition (DMD) as a general-purpose, entirely data-driven approach to capture the above liveness cues. We propose a classification pipeline consisting of DMD, Local Binary Patterns (LBP), and Support Vector Machines (SVM) with a histogram intersection kernel. A unique property of DMD is its ability to conveniently represent the temporal information of the entire video as a single image with the same dimensions as those images contained in the video. The pipeline of DMD+LBP+SVM proves to be efficient, convenient to use, and effective. In fact only the spatial configuration for LBP needs to be tuned. The effectiveness of the methodology was demonstrated using three publicly available databases: print-attack, replay-attack, and CASIA-FASD, attaining comparable results with the state of the art, following the respective published experimental protocols.

5.1 Introduction

The usage of biometrics in the area of security and for authentication purposes is of well-established importance. However, the widespread use of biometrics is hampered by security concerns such as database tampering [82] and sensor tampering (or presentation attack) [83], and many other attacks described by Ratha et al. [84]. While database tampering can lead to authorisation of an illegal individual, or denial of service to a legitimate user, sensor tampering can cause the granting of access right to an unauthorised user; thus, severely compromising the security and
usability of biometric systems. In this chapter, we are concerned with rendering a face recognition system robust against presentation attacks; and specifically, print attacks based on printed facial images, and replay attacks that are carried out by replaying a video sequence using a tablet or a mobile phone.

The key innovation introduced here is the exploitation of the facial dynamic information in a completely data-driven fashion, rather than using prior knowledge regarding live face images such as eye-blinking and lip-movements (e.g. [85]). Since attack types are often unknown and very different from each other. Knowledge-driven approaches are unlikely to be able to scale well to a plethora of attacks. Each new attack potentially requires the algorithm designer to develop a specific cue that is peculiar to the attack being employed. Data-driven approaches, on the hand, can automatically learn to discriminate valid (i.e. live) videos\(^1\) from attack videos given a sufficiently wide range of training data and attack types.

In order to extract facial dynamic information reliably, we propose a modified version of Dynamic Mode Decomposition (DMD). Since DMD has been used extensively in modelling fluid dynamics, we postulate that DMD can also capture the complex dynamics of head movements, eye-blinking, and lip-movements found in a typical video sequence containing face images. However, DMD has not been used for classification tasks before. For this reason, we propose a DMD-based classification pipeline involving a texture-based descriptor, coupled with a discriminative classifier. Our experiments show that DMD can indeed capture unique features that clearly distinguish videos produced by print and replay attacks from live (valid) videos containing an authentic face.

5.1.1 **Background on attacks directed at biometric systems**

Attacks directed at biometric systems [87] can be broadly classified as database tampering and sensor attacks:

Sensor tampering which is also referred to as a presentation attack [88] is targeted towards fooling a biometric sensor. A common type of presentation attack is spoofing, which refers to the use of fake biometric data in order to deceive a biometric system [83]. Spoofing attempts dealing with fake fingerprints dates back to 1998, when Wills and Lees [89] indicated that four out of six devices they tested were vulnerable to fake prints attacks. Research has also been carried out on the use of gum-based fingerprints to fool biometric systems by Matsumoto et al. [90]. Thalheim et al. [91] further demonstrated that fingerprints, iris scans and face recognition systems are vulnerable to replay attacks.

For face recognition, there are two types of presentation attacks, namely print attacks and replay attacks. A print attack refers to facial spoofing carried out by presenting a printed photo to a camera [92]. Indeed, this attack is very easy to carry out because getting hold of a target’s photo is extremely simple. With the

\(^1\)We refer to live, authentic videos as *valid* videos following closely Pinto et al.’s terminology [86]; as opposed to *attack* (or *spoofed*) videos produced by various spoofing methods.
increasing use of social media, few users consider the repercussions associated with an act as simple as uploading their pictures [92]. Replay attacks, on the other hand, are carried out by replaying a previously recorded face image (video) of a target user in order to spoof a biometric system. The video can be replayed easily using a hand-held mobile tablet. Again, with the widespread use of high-quality consumer-grade camcorders and social media networks, this type of attack can be carried out easily in both remote and logical access control systems protected by a face recognition system.

We exclude from this study replay attacks that require an attacker to know about the architecture of a biometric system. These attacks assume that the attacker can intercept the communication channel between the biometric sensor and its feature extractor, thus acquiring a raw biometric image; or intercept the communication between the feature extractor and the biometric matcher, thus acquiring an extracted biometric feature set. It is further assumed that the attacker can replay or reintroduce the copied data (raw image or biometric feature) into the respective communication channel in order to gain access to such a secured system [93]. These attacks are much more sophisticated to carry out and are best counteracted by securing the communication channels; therefore, they are out of the scope of this study.

5.1.2 Motivation

In the context of face counter-spoofing, we hypothesise that DMD has the potential of discovering the facial dynamics that captures the vitality signs of valid face videos whilst extracting artefacts of spoof videos such as moire and planar effects, at the same time. The moire patterns are often regarded as undesired artefacts of images, the effect of which is similar to fibres in moire silk. The planar effect is the appearance of an picture that is flat rather than 3D, as opposed to a real face.

DMD works analogously to Principal Component Analysis (PCA). Whereas DMD contains dynamic information about the data under consideration, PCA lacks this property [18]. Tirunagari et al. [4] compared PCA and DMD in extracting Kelvin-Helmoltz instabilities from complex flows. They showed that DMD is superior to PCA in analysing complex fluid flows. This is consistent with the literature in both experimental and numerical flow field data problems [18]. Indeed, not only can DMD extract dynamic information effectively, thus, temporal dynamic characteristics, it can also capture spatially-coherent structures [18]. Schmidt et al. [17] used DMD on a sequence of fluid flow images and illustrated how it can detect dynamically relevant coherent structures for characterising fluid behaviour over a given time interval. Similarly, we hypothesise that extracting relevant modes from videos in face recognition can provide a good separability between valid videos and the spoofed ones. Particularly, for biometric applications such as replay and print attacks, in which observations are carried out in a continuous pattern i.e. several consecutive images in a video sequence rather than a single snapshot of image
extracted from a video, we propose a modified DMD method that is computationally efficient for handling large size videos. This constitutes the originality of our proposal in tackling both photo and video based spoof attacks.

5.1.3 Contributions

Although much work has been directed towards tackling issues related to print attacks and replay attacks, as well as the development of publicly available databases, e.g., the print-attack database [94], the replay-attack database [95] and the CASIA-FASD database [96], there is still significant room for improvement for anti-spoofing methods in face recognition. In particular, there is no literature on the use of DMD on print and replay attacks. Our first contribution is, therefore, to introduce the use of DMD for countering spoof attacks in face recognition. Since DMD has not been used with classification of video sequences, we propose a system pipeline consisting of DMD, LBP and SVM. This combination is ideal because DMD captures the visual dynamics in the form of a fixed-size image, LBP can effectively capture the dynamic patterns, and SVM is known to be an ideal general-purpose classifier that minimises the empirical risk of classification error. Extensive experiments were conducted on: (i) a print-attack dataset [94] containing 200 videos of real-access and 200 videos of spoof attempts using printed photographs of 50 different identities. (ii) Replay-attack dataset containing 200 videos of valid-access and 1000 videos of spoof attempts using printed photographs, tablets replays and mobile phones replays of 50 different identities. (iii) The CASIA-FASD containing 50 real clients with 150 valid-access videos and 450 attack videos.

Our second contribution is related to making the DMD more practical. While the original DMD uses QR-decomposition [15] (decomposition of a matrix in to an orthogonal matrix Q and an upper triangular matrix R) and singular value decomposition (SVD) methods [16], we propose to use LU decomposition [9] (factors a matrix as the product of a lower triangular matrix L and an upper triangular matrix U). Second, in our modified DMD, we use the absolute value of the complex modes when rendering the “DMD image”. Third, we recommend that the optimal mode to select the DMD modes using the zero phase angle criteria. In short, we have made DMD practical for classification and have demonstrated its effectiveness on separating valid access videos from spoofed ones.

Our final contribution is to simplify the taxonomy of existing facial anti-spoofing methods by categorising them as being data-driven or cue-based. For each category, a number of methods are further enumerated (see Figure 5.1) and supported by the relevant literature as shown in Tables 5.1, 5.2 and 5.3. In this way, the strengths of different solutions proposed in the literature can be taken full advantage in order to improve the liveness detection task of the 2D face recognition technology.

The following research question are answered in this chapter:

1. How can DMD be used to highlight facial dynamics such as lip movements, eye blinks and other attack specific cues that can be used to discriminate
5.2. EXISTING SOLUTIONS

between a valid and a spoofed facial video?

2. What are the current state of the art computer vision methods for detecting spoofing attacks in 2D facial videos?

3. What is the performance of DMD when compared to these methods?

5.1.4 Organisation

Section 5.2, presents state-of-the-art methods using our proposed categorisation. Section 5.3, presents the modified version of Dynamic Mode Decomposition (DMD) method and its application on spoof detection. A brief overview of the selected face spoofing datasets is presented in Section 5.4. Section 5.5 presents the experimental procedure including the results, followed by conclusions and future directions in Section 5.6.

5.2 Existing solutions

According to Chakka et al. [97], techniques to counter spoofing in 2-D face recognition systems are generally divided into motion, liveness, and texture analyses. Motion generally refers to motion features such as optical flow, e.g., [85] and [98]. Liveness analysis refers to the detection of the vitality signs of a biometric sample, such as head and lips movement, and eye-blinking. Finally, texture analysis refers to the use of texture descriptors for discriminating valid videos from the spoofed ones.

We argue that such a categorisation is ambiguous and does not help one to identify the strength of each technique. For example, a liveness analysis method that relies on movements is often implemented using motion descriptors such as optical flow. As a result, it is futile to argue whether or not a method is classified as belonging to liveness analysis or motion analysis.

Instead, we advocate to “decompose” existing methods into primary components in such a way that the strengths of different solutions proposed in the literature can be taken full advantage for improving the liveness detection task of the 2D face recognition technology. Our proposed broad categorisation of methods classifies a method as either data-driven or cue-based. This categorisation and some of the potential subcategories and specific methods are shown in Figure 5.1.

While cue-based methods rely on intuitions and/or observations, data-driven methods tend to use generic image-processing algorithms that are applicable to a wide range of computer vision tasks. It is worth noting that the cue-based and data-driven attributes are not mutually exclusive. Indeed, a cue-based method may rely on generic texture descriptors such as local binary pattern (LBP) or motion descriptors such as optical flow. As a result, categorising a method as being texture-only or motion-only would not help a deeper understanding of which component of the method works, why it works, what assumption is made, and when it may fail. By
decomposing a method based on cue-based attributes and data-driven attributes, the success or failure of a technique can be clearly identified. The result of this analysis is shown in Tables 5.1, 5.2 and 5.3.

**Cue-based methods**

We further distinguish two types of cues, namely liveness cues and attack-specific properties/cues.
Table 5.1: Categorisation of existing facial counter-spoofing methods: Part-1.

<table>
<thead>
<tr>
<th>Broad category</th>
<th>Data-driven</th>
<th>Cue-based (motivated by)</th>
<th>Attack-specific properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Narrow category</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Motion analysis</td>
<td>Consecutive-frame analysis</td>
<td>Frequency or spatial analysis</td>
<td>E.g. blinking eyes, lips movement</td>
</tr>
<tr>
<td>Texture analysis</td>
<td>Video analysis</td>
<td></td>
<td>E.g. planar effect, moire effect</td>
</tr>
<tr>
<td>Li’s method [99]</td>
<td>2D Fourier spectrum</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tan’s method [100]</td>
<td>Variational retinex-based and difference-of-Gaussian (DoG)</td>
<td></td>
<td>The surface roughness in terms of Lambertian reflectance between valid and spoof videos are different</td>
</tr>
<tr>
<td>Peixoto’s method [101]</td>
<td>Applying an adaptive histogram equalisation to the images before extracting latent reflectance features</td>
<td></td>
<td>The brightness of the LCD screen affects the recaptured image, which makes the image edges more susceptible to blurring</td>
</tr>
<tr>
<td>Maatta’s method [102]</td>
<td>Micro-texture analysis via LBP with SVM as a classifier</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tronci’s static method [103]</td>
<td>Texture histogram capturing colour in RGB and HSV space; edge directivity; MPEG descriptors</td>
<td></td>
<td>The loss of information through the image recapturing process and the peculiar noise</td>
</tr>
<tr>
<td>Tronci’s video-based method [103]</td>
<td></td>
<td></td>
<td>eye blinks, mouth alterations and changes in facial expression</td>
</tr>
</tbody>
</table>
Table 5.2: Categorisation of existing facial counter-spoofing methods: Part-2.

<table>
<thead>
<tr>
<th>Broad category</th>
<th>Data-driven</th>
<th>Cue-based (motivated by)</th>
<th>Attack-specific properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Narrow category</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Motion analysis</td>
<td>Consecutive-frame analysis</td>
<td>Frequency or spatial analysis</td>
<td>E.g. blinking eyes, lips movement</td>
</tr>
<tr>
<td>Texture analysis</td>
<td>Video analysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Schwartz's method [104]</td>
<td>shape, colour and texture of the face</td>
<td>Planar object movement cues aiming to detect four basic rigid object motion types, namely, translation, in-plane rotation, panning or out-of-plane rotation (swing)</td>
<td></td>
</tr>
<tr>
<td>Bao’s method [105]</td>
<td>Optical flow</td>
<td>Comparing a half of face with another half that is divided in two ways: horizontally and vertically.</td>
<td></td>
</tr>
<tr>
<td>Kollreider’s method [85] and [98]</td>
<td>Optical flow</td>
<td>Capturing the subtle movements of different facial parts (i.e., face and ear proportion), assuming that facial parts on real faces move differently than on photos.</td>
<td></td>
</tr>
<tr>
<td>Pan’s method [106] and [107]</td>
<td>Optical flow</td>
<td>Inside face changes (e.g., eye blinking) versus outside face (the scene context)</td>
<td></td>
</tr>
</tbody>
</table>
Table 5.3: Categorisation of existing facial counter-spoofing methods: Competing methods: Part-3.

<table>
<thead>
<tr>
<th>Broad category</th>
<th>Data-driven</th>
<th>Cue-based (motivated by)</th>
<th>Narrow category</th>
<th>Attack-specific properties</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Motion analysis</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Consecutive-frame analysis</td>
<td>E.g. planar effect, moire effect</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Video analysis</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Frequency or spatial analysis</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Liveness cues</td>
<td>E.g. blinking eyes, lips movement</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Attack-specific properties</td>
<td></td>
</tr>
<tr>
<td>Pinto et al.’s visual rhythm [86]</td>
<td>Multiple video spectra represented by grey-level co-occurrence matrices represented by 12 statistical descriptors</td>
<td>Frequency analysis of noise residuals</td>
<td>Noise residuals capture moire effect</td>
<td></td>
</tr>
<tr>
<td>Anjos et al.’s method [108]</td>
<td>Optical flow-derived statistics based on Chi-square distance of a pair of Von-mises distributions</td>
<td>Inside face versus outside face context</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Our proposed method</td>
<td>Matrix decomposition of consecutive pairs of detected face images</td>
<td>Dynamic mode decomposition</td>
<td>LBP features classified by SVM</td>
<td>Although not based on cues, DMD can detect eye movements, lips movement, and the facial dynamics.</td>
</tr>
</tbody>
</table>
Liveness cues. We define liveness cues as ones that depend on the vitality signs of a biometric trait. For 2D face recognition, these cues are eye blinking, lips movement, and changes in facial expression. These cues were either made implicit or explicit by authors such as Tronci et al. [103], Li et al. [82], Tan et al. [100], Peixoto et al. [101], Pan et al. [106], and other authors as listed in Tables 5.1, 5.2 and 5.3. While this approach may be effective for countering attacks based on printed face images, it may fail completely with a video-based replay attack because a video can capture all aspects of facial liveness stated above. Given the ease of acquiring facial video and the high accessibility of facial videos in the public domain such as Facebook, often posted by the user himself, the possibility of an attacker employing a video-based replay attack is extremely high. In order to address this weakness, one will need to look for additional attack-specific cues that are described below.

Attack-specific cues are those that do not rely on the facial properties and are often idiosyncratic to attack method. An example of this is the moire effect which is a secondary and visually undesirable artefacts that are caused by two superimposed patterns. In the context of 2D face attack, this effect is caused by the recapture of an image from another display device. This effect was pointed out in Pinto et al.’s work [86]. Interestingly, the non-face information such as the moire effect can be obtained from the facial region but this is still considered an attack-specific cue. Another cue that is peculiar to a planar media such as a printed face image or a face image shown on a digital tablet is the flatness of the replicated image. The first use of this cue is attributed to Bao et al. [105]. Similarly, in the case of valid versus spoofed finger vein image samples, one can observe that both types of images differ in terms of image quality and light reflection. For example, a valid finger vein image and a printed one reflect light in different ways because a vein is a complex non rigid 3D object whereas a photograph can be seen as a planar rigid 2D object. This may cause different reflections and shading. The surface properties of valid finger vein compared to the printed ones are also different.

Since, cue-based methods exploit the peculiarity of a specific attack setting or method, the solution may not always generalized to other attack methods. As a result, one may obtain poorer results than those reported in the initial studies. Indeed, some literature [86] suggests that cue-based methods that aim to exploit printed medium attacks may not work on replay video-based attacks or 3D mask attacks. For instance, Li’s method [82] which is based on facial size difference and invariant facial pose and expression may not work when the assumptions do not hold. One way to invalidate the assumptions thus defeating the countermeasure is by rotating or bending a printed face photo with high very resolution, as suggested by Pinto et al. [86]. Another example is Pan’s method [106] which is based on the cue that the face region and the region outside the face (that is, the scene context), are different. Since the background of an image may be different from one live video to another, the method may fail. Furthermore, with a 3D mask, an attacker can also simulate eye blinking by looking through a perforated 3D mask. Indeed, in an attempt to replicate Bao’s method [105] and Kollreider’s method [98],
Anjos et al. [108] found that these methods did not work well for video-based replay methods even after tuning the parameters associated with the algorithms as optimally as possible on their database following a strictly well-defined and unbiased experimental protocol.

**Data-driven methods**

Data-driven methods can be further divided into motion and texture analyses. By motion analyses, we understand that the method exploits consecutive images in a sequence such as optical flow, or considering a video as a multidimensional sequence or time-series. In contrast, a method is considered texture analysis if it can derive information from a single image frame in the video.

For motion analysis, one can further divide them into two subcategories, using one of the following strategies:

1) **Consecutive-frame analysis**: analysing motion on two consecutive video frames, such as methods based on optical flow.

2) **Video-as-an-image-sequence analysis**: analysing motion by treating a video as a multidimensional sequence. Under this categorisation, Bao’s method [105], Kollreider’s method [98] and Anjos et al.’s method [108] that are based on optical flow are considered consecutive-frames analysis methods. Similarly, our method based on DMD also belongs to this category. However, DMD does so by taking into account of all possible consecutive pairs of images in a video sequence in its formulation (to be discussed in Section 7.2). In this sense, video-as-an-image-sequence analysis is a generalisation of consecutive-frame analysis.

3) **Video-as-an-image-set analysis.** Another possible case that was not covered above is analysing a video by treating it as a collection or set of images. Such an approach typically does not consider the motion information. An example of this approach is to project a set of images into a texture subspace using principal component analysis (PCA). Since the ordering of the image does not matter, the motion information is not exploited. Unlike PCA, DMD treats a video as a sequence of images and projects them in the principal motion subspaces. As a result, DMD is superior to PCA in classifying motion. Indeed, a large body of literature in fluid dynamics have demonstrated the superiority of DMD over PCA. For this reason, we hypothesise that DMD is a viable candidate for classifying facial video liveness.

4) **Video-as-a-set texture analysis.** An extension of texture analysis is to apply a method to the entire set of video images. This leads to a variant of texture analysis that we refer to as video-as-a-set texture analysis. This type of analysis can be done at the feature-level or at the score level. In the first case, new features are derived by combining the features extracted from each image in a video sequence or a batch or chunk of video sequence. The derived features can be combined until an entire video is processed. LBP-TOP is one such example [109]. At the score level, typically the output of a liveness classifier which reflects a hypothesis score is used in the combination process. The classifier first classifies each image in a
video sequence and the resultant scores, one from each image, are combined via a
fixed operator such as mean or a second-level classifier that combines the statistics
(e.g. [110]).

5.2.1 Key considerations in 2D facial counter-spoofing

The growing body of literature in 2D facial counter-spoofing have implicit addressed
two key issues, namely: (i) The importance of motion information. (ii) The need
for a classifier to discriminate between valid and spoof facial videos.

- **Motion-based methods versus non-motion-based methods:** Through
  a number of comparative studies, such as those reported by Anjos et al. [108],
  Chakka et al. [97], Chingovska et al.’s works [95], there is sufficient evidence
to suggest that motion information is vital and methods based on this in-
formation can cover a wide range of attacks targeting the 2D face recogni-
tion technology. Indeed, motion can capture not only liveness cues but also
attack-specific cues as discussed above. Later we will compare both types of
approaches in our experiments, i.e., PCA, which represents a non-motion-based
method versus DMD, which represents a motion-based method.

- **Trainable versus non-trainable methods:** Another important consider-
  ation among all the methods surveyed here is whether or not a method is
  trainable. All data-driven methods are considered trainable by definition be-
  cause each of these methods requires a classifier to be able to discriminate
  valid videos from the spoofed ones. Although these methods can potentially
  be more general and more robust, they can only work well with spoofing
  methods that are found in the training set. If an attack type is not present
  in the training set, it is possible that the classifier may not generalise well. In
  comparison, cue-based methods may not require a classifier; therefore, they
  are considered non-trainable. An example of this is Anjos et al.’s optical flow
correlation algorithm (in [108]) which does not need a trainable classifier but
nevertheless requires a few parameters to be tuned via cross-validation on a
given dataset. Whether or not the algorithm will still work on other unseen
attack method is still an open question. In both cases, the generalisation abil-
ity of a method to an unseen attack type would constitute a future research
study. In summary, although the generalisation ability of a method to an
unseen attack type is not guaranteed, trainable methods have the additional
advantage that given a sufficiently representative training data that contains
a plethora of attack methods, they are a preferred solution because they can
exploit the additional data without significantly modifying the underlying al-
gorithm unlike the cue-based methods. For this reason we consider a trainable
methodology based on DMD.
5.2. EXISTING SOLUTIONS

5.2.2 Characteristics of DMD

Among the methods surveyed, few methods are completely data-driven and do not rely on any cues. For example, Pinto et al.’s method [86] is considered a cue-based method and data-driven because it exploits the moire effect, the derived features of which are then classified using SVM. DMD is a method that qualifies both criteria. Indeed, DMD is an algorithm that exploits the motion information in a video sequence. Although optical flow can also capture this information for a pair of consecutive frames, DMD does so at the sequence level.

Although completely data-driven, we find that DMD can capture liveness cues and attack-specific artefacts at the same time. For instance, DMD can capture eye-blinking and lips movements, as well as the peculiarity of attacks. Furthermore, DMD can also capture the motion due to attack-specific artefacts such as planar movement. This is shown in Figure 5.5.

5.2.3 Competing methods

As explained, DMD is somewhat unique in its category of algorithms because of its ability to process multidimensional temporal sequence, which is a generalisation of consecutive-frames analysis such as algorithms based optical flow. Among them, we have chosen Anjos et al.’s method [108] which is considered superior among all the optical flow methods. We have also chosen Pinto et al.’s method [86] because of its use of a classifier. Table 5.4 compares the DMD with these two algorithms in terms of a number of attributes.

Pinto et al.’s visual rhythm works well only if the assumption about the high frequency image component, that is, the noise artefacts introduced by the recaptured printed/planar medium behave in the same way. Unfortunately, the video replay attack carried out by Pinto et al. was based on LCD screen for replaying the video with a mounted (Sony Cybershot) camera. Therefore, no random movement that is typically produced by hand-held attack media is observed in the test set. It is, therefore, not clear how this movement might affect Pinto et al.’s algorithm.

Similarly, Anjos et al.’s optical flow correlation works well under the assumption that face and non-face regions have different distribution properties in terms of optical flow. Presumably a talking face video sequence would induce large chi-square distances. Anjos et al.’s method also implicitly relies on involuntary eye-blinking motions which happen every 3-4 seconds. However, a valid, face video without movement that happens between two eye-blinking motions would induce much smaller chi-square distances just as a printed facial photo attack does, leading to potential false rejection. A strength of their method is that they can use a standard face detector without explicitly needing to localise the two eyes in an image.

In comparison, the proposed DMD is completely data-driven; yet it is able to detect liveness cues and attack-specific artefacts, as described above. However, DMD needs a classifier. For this reason, we propose to use DMD in conjunction with
Table 5.4: Comparison of the DMD with Pinto et al.’s visual rhythm and Anjos et al.’s method in terms of a number of attributes.

<table>
<thead>
<tr>
<th>Attributes</th>
<th>DMD pipeline (our proposal)</th>
<th>Pinto et al.’s visual rhythm [86]</th>
<th>Anjos et al.’s method [108]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consider multiple frames?</td>
<td>Video-as-an-image-sequence</td>
<td>Video-as-a-set</td>
<td>Video-as-an-image-sequence</td>
</tr>
<tr>
<td>Observational feature</td>
<td>Spatial domain (pixel)</td>
<td>Frequency domain of noise residual</td>
<td>Optical flow</td>
</tr>
<tr>
<td>Does observation capture motion?</td>
<td>Yes, for the entire sequence</td>
<td>Yes, via optical flow and their subsequent concatenated images (“visual rhythm”)</td>
<td>Yes, via optical flow</td>
</tr>
<tr>
<td>Features used for classification</td>
<td>LBP applied to DMD principal modes</td>
<td>12 statistical features derived from the grey-level co-occurrence matrices of the concatenated image-set spectra</td>
<td>A pair of Von-Mises distributions derived from the facial region and non-face region of the optical flow output</td>
</tr>
<tr>
<td>Classifier</td>
<td>SVM</td>
<td>SVM, Partial Least Square</td>
<td>Chi-square distance</td>
</tr>
<tr>
<td>Cues relied upon or extracted</td>
<td>Automatically detect liveness cues and attack-specific artefacts</td>
<td>Authors’ intuition on the moire effect of the replay video</td>
<td>Authors’ intuition that face versus nonface optical flow are different</td>
</tr>
</tbody>
</table>

LBP as a texture descriptor of the visual dynamics, and SVM as a back-end classification engine. This proposed DMD-pipeline was tested on both printed-medium attacks and replayed video attacks across various settings, including hand-held and mounted replay media, using the latest print-attack and replay-attack face datasets. Our experimental results, following strictly the published experimental protocols, show that the proposed DMD pipeline achieved the best performance reported so far (with the exception of Komulainen et al.’s method [111] on CASIA-FASD dataset, to be further discussed in Section 5.5.8) in comparison to the literature, i.e., Chingovska et al.’s work [95]. Indeed, the classification performance was perfect for the printed media and for the video-replay attacks, we achieved an HTER of 0.55% on the development set and 0% on the test set.
5.3 Methodology

In this section, we present the pipeline of our method which consists of DMD, Local Binary Pattern histograms and a kernel based Support Vector Machine (SVM) classifier. The overall methodology or process pipeline is shown in Figure 5.2. First, a video is processed using the DMD algorithm in order to output dynamic mode images. From which, we select a single dynamic mode image corresponding to eigenvalue whose phase angle is $\theta = 0$ or closest to it, after discarding the background mode. Second, LBP histogram features are computed for this dynamic mode image. Finally, the produced LBP code is fed into a trained SVM classifier in order to classify whether the processed video is a valid access or spoof. Half Total Error Rate (HTER) is used to evaluate the performance measure. To validate our DMD pipeline we have used principal component analysis (PCA) based on snapshot approach as a baseline method.

5.3.1 Dynamic Mode Decomposition (DMD)

In a facial authentication video sequence consisting of $N$ images, $X$, let $\bar{x}_r$ be the $r^{th}$ column vectorised frame (size of $\bar{x}_r$ is $mn \times 1$ whose initial size was $m \times n$ in the image domain). For $N$ frames in the sequence this results in the construction of a data matrix $X$ of size $mn \times N$ for $N$ frames. This data matrix containing of facial authentication video sequence is solved through DMD via Arnoldi (LU) approach to obtain the dynamic modes:

$$H = U^{-1}L^+P_2.$$
Here, $L^+$ is the pseudo inverse of $L$ from the LU-decomposition of $P_1$ subspace.

The associated eigenvectors of $H$ provide the coefficients for the linear combination that is necessary to express the dynamics within the video frame basis.

$$H \omega = \sigma \omega,$$

DMD modes are thus calculated as follows:

$$\Psi = P_2 \omega$$

Dynamic modes not only contain the information about dynamic structures, but also about the temporal evolution of patterns within a video sequence. Since at no stage of the algorithm the system matrix $A$ is needed, various extensions of the algorithm are possible. No specific spatial arrangement of the image data is assumed and there are no parameters to be tuned.

5.3.2 Principal Component Analysis (PCA) via snapshot approach

In order to compare DMD, this section explains how PCA can be calculated, and the output of which can be directly used to replace the DMD output. Therefore, we in here use a variant of PCA that is based on the ‘snapshots’ approach which is introduced by Sirovich [112]. Here each video frame at a particular interval in time is considered to be one snapshot. This method was introduced to reduce the computations of covariance matrices. Normally, computing PCA requires solving a $mn \times mn$ eigenvalue problem, where $mn$ is the total number of pixels in a video frame. The snapshots method proposes that the covariance matrix can be approximated by a linear combination of the ‘snapshots’. Instead of computing $mn \times mn$ covariance matrix, this method computes a covariance matrix of size $N \times N$. This method has become extremely popular in the field of CFD [113–115].

To implement the snapshots method, a sequence of images is converted to form a matrix as shown in Equation 7.1

The fluctuating pixel intensity matrix $E$ is calculated by subtracting the mean matrix (average pixel intensities) $\hat{X}$ from $X$,

$$E = X - \hat{X}.$$  

(5.4)

The covariance matrix is computed as

$$C = E^T E.$$  

(5.5)

An eigenvalue problem for the covariance matrix is then solved

$$CZ^i = \lambda^i Z^i.$$  

(5.6)

The eigenvectors are arranged according to the decreasing order of eigenvalues. This ordering is physically meaningful because, it reflects the fluctuating intensity
energies in the PCA modes. Using the ordered eigenvectors the PCA modes are constructed.

\[ \phi_i = \frac{\sum_{n=1}^{N} Z_n^i E_n^i}{\| \sum_{n=1}^{N} Z_n^i E_n^i \|} , \quad i = 1, 2 \ldots, N. \] (5.7)

In our experiments, we consider only the first PCA mode as this mode captures the greatest percentage of the intensity fluctuation from the video frames [4, 113, 114, 116].

5.3.3 Local Binary Patterns (LBP)

Local Binary Patterns (LBP), a powerful image representation, extracts texture information that is invariant to local grey scale variations. During the LBP operation, every image pixel acts as a threshold to its neighbours to obtain a binary bit string which then forms a round number. In real applications, an image is usually divided into several small non-overlapping blocks of the same size to keep the spatial relation of objects. Conventionally, LBP is denoted as B-LBP\textsuperscript{u2}_{P,R}, where \( P \) and \( R \) denote the neighbourhood pixels and radius respectively and the superscript \( u2 \) denotes uniform LBP. \( B \) denotes the number of non-overlapping blocks.

In this study we consider \( P = 8 \) for block-size = \{1 \times 1, 3 \times 3, 5 \times 5\} and radii = \{1, 2, 3\}. The optimal LBP settings are determined by cross validation.

5.3.4 SVM classifier - Histogram Intersection Kernel

Kernel based Support Vector Machine (SVM) is a widely used pattern classification method and is well known for its high classification accuracy. Kernel methods transform data from a low dimensional space to a high dimensional space using non-linear maps. By non-linearly mapping the data onto the high dimensional space, it is theoretically shown that any linearly non-separable data can become linearly separable [117, 118]. Since only the inner product between a pair of observations is required in the SVM formulation, the non-linear mapping does not need to be explicitly defined for an individual observation in the training set. Instead, the non-linear kernel function is also defined for a pair of observations. This maintains the efficiency of the SVM. Therefore, a kernel based SVM in general has a better performance over an original SVM for linearly non-separable classification tasks [119].

In this study we opted to use the Histogram Intersection Kernel [120], which is also known as the Min Kernel because it has been shown to be efficient and effective in image classification tasks [121]. The kernel is defined as follows:

\[ K(x, y) = \sum_{i=1}^{n} min(x_i, y_i), \] (5.8)
where \( n \) is the number of dimensions in the LBP histogram and \( x \) and \( y \) are the LBP histograms. Note that this kernel has no additional parameter to tune, which is convenient to use.

### 5.3.5 Performance and Threshold Criteria

Although the output of SVM can be used to make a hard decision, we shall use its soft output, which is defined as the distance of a test sample from the SVM decision hyperplane. For all experiments, we label samples derived from valid videos as positive; and those derived from attack videos to be negative.

For the purpose of performance evaluation, FAR is defined as the empirical probability estimate of falsely declaring an attack sample as being a valid one; whereas FRR is defined as the empirical probability of falsely rejecting a valid access.

Let \( T \) be the domain of SVM output. The decision is made by comparing the SVM output \( t \in T \) with a decision threshold, \( \Delta \in T \), as follows:

\[
\text{decision}(t) = \begin{cases} 
\text{valid access} & \text{if } t > \Delta \\
\text{spoof/attack} & \text{otherwise,}
\end{cases}
\]

(5.9)

In order to calculate errors, we introduce match and nonmatch score sets, \( Y_1 \subset Y \) and \( Y_0 \subset Y \), respectively. This can result in two errors, namely, false rejection and false acceptance, the rates of which are calculated as follows:

\[
\text{FRR}(\Delta) \equiv P(t < \Delta | Y_1)
\]

\[
\approx \frac{|Y| Y \in Y_1, Y < \Delta|}{|Y_1|}
\]

(5.10)

(5.11)

\[
\text{FAR}(\Delta) \equiv 1 - P(t < \Delta | Y_0),
\]

\[
\approx \frac{|Y| Y \in Y_0, Y > \Delta|}{|Y_0|}
\]

(5.12)

(5.13)

respectively, where the conditioning variable \( Y_1 \) indicates that the comparison is due to a positive class (valid access) whereas \( Y_0 \) indicates that the comparison is due to a negative class (spoof/attack). The average of these both errors is defined by HTER.

\[
\text{HTER}(\Delta) = \frac{1}{2} (\text{FAR}(\Delta) + \text{FRR}(\Delta)).
\]

By taking the geometric average of FAR and FRR, HTER has the advantage that it is not affected by the overwhelmingly large sample size of one class versus another because both types of errors are weighted equally, thus coercing equal contribution of both errors, i.e., assuming equal prior probability for both classes. In our case, this is particularly desirable because the prior probability of an attack is difficult to estimate in practice. In the absence of any additional information, using equal prior probabilities is a reasonable option.
5.4 Dataset

In this section we discuss the datasets used in our study. We have chosen to use datasets containing print attack and replay attack spoof samples. These datasets consist of short video recordings for both valid-access and attack attempts.

5.4.1 Print-attack Dataset

Print-attack Dataset [108] is the first publicly available dataset that provides a precise protocol with training, development, and test sets. It contains a short video of valid access and spoof attacks to 50 different identities. The spoof attack that is emphasised in this dataset is print attack only, whereby an imposter presents a printed photograph of the targeted identity in order to falsify the access to a face biometric authentication system. This dataset, includes two different scenarios: (i) controlled background (the background is uniform) and (ii) adverse background (a non-uniform background). These scenarios provide a valid simulation of the attack environment. Both of the attack attempts were carried out in two ways namely: (i) hand-based, where the imposter holds the printed image using their own hands, and (ii) fixed-support, where the imposter sets the printed image on a fixed support so they do not move during the spoof attempt.

5.4.2 Replay-attack Dataset

To extend the print-attack dataset, Chingovska et al. [95] introduced a replay-attack dataset with more sophisticated attacks. To validate the attacks, the author categorised the attacks in three different scenarios:

- **Print attack**: printed photograph (same protocols as proposed in print-attack).
- **Mobile attack**: the imposter presents photos and videos taken with a mobile phone using the mobile screen.
- **Tablet attack**: the imposter presents a high resolution digital photos and videos using an iPad screen.

The replay-attack dataset consists of 200 videos of valid access (with 375 frames each), and 1000 videos of attack attempts (with 240 frames each). The dataset is divided into three partitions, namely training, development, and testing set. The development set is used for estimating the threshold value and training set is used for estimating any model parameters. Each of these sets are generated by a video gallery of 15 clients for development and training, and 20 clients for testing. This means that the training and testing sets are disjoint and completely independent of each other. There are 120 videos of valid access and 600 spoof videos under six types of attack attempts, namely, tablet-fixed, mobile-fixed, print-fixed, tablet-hand, mobile-hand and print-hand. Table 5.5 shows the details of the replay-attack dataset distribution, including the distribution for print-attack dataset.
Table 5.5: Number of videos in each set according to the amount of hand-based and fixed-based attacks.

<table>
<thead>
<tr>
<th>Type</th>
<th>Train</th>
<th>Devel.</th>
<th>Test</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid-access</td>
<td>60</td>
<td>60</td>
<td>80</td>
<td>200</td>
</tr>
<tr>
<td>Print-attack</td>
<td>30+30</td>
<td>30+30</td>
<td>40+40</td>
<td>100+100</td>
</tr>
<tr>
<td>Phone-attack</td>
<td>60+60</td>
<td>60+60</td>
<td>80+80</td>
<td>200+200</td>
</tr>
<tr>
<td>Tablet-attack</td>
<td>60+60</td>
<td>60+60</td>
<td>80+80</td>
<td>200+200</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>360</td>
<td>360</td>
<td>480</td>
<td>1200</td>
</tr>
</tbody>
</table>

### 5.4.3 CASIA-FASD dataset

Additionally, to validate our proposed pipeline, we consider the CASIA-FASD spoofing dataset [96], which is more challenging than the first two datasets. For instance, it contains printed 2D attack with perforated eyes in order to simulate eye blinking, known as a ‘cut-photo’ attack. This dataset contains print attacks (or warped photo), cut-photo attacks and reply attacks captured in three settings, namely low, normal, and high resolutions. In total, the dataset consists of 600 videos including 150 valid-access and 450 attacks. The training and testing sets include 240 and 360 videos respectively.

### 5.5 Experiments & Results

In this section we discuss the experimental procedures as well as the results. The objectives of our experiments are as follows:

1. **Qualitative assessment of dynamic information captured by DMD.** From the outset, we hypothesise that DMD can capture liveness cues such as eye blinks, lips movements and facial dynamics. Therefore, for valid videos, we wish to examine whether these features are indeed picked up by DMD. In the case of photo attacks, DMD modes should, in theory, be characterised by a lack of motion, resulting in a somewhat noisy image. Finally, we expect that the facial texture properties of valid accesses and video attacks would differ due to the replayed videos being affected by the moire or planar effects. Since DMD captures spatial-temporal features, it is reasonable to expect that the DMD modes would produce somewhat different textures.

2. **Optimisation of the DMD+LBP+SVM pipeline.** In our proposed DMD classification pipeline, there are two parameters that need to be predetermined, namely, (1) the choice of DMD mode, and (2) parameters associated with the LBP.

**Optimal DMD mode:** In theory, for a video with $N$ frames, we obtain $N-1$ DMD modes. Recall that each dynamic mode captures a “principal dynamics” axes of the video sequence. In order to select the most optimal mode,
we choose the eigenvector whose corresponding eigenvalue has a phase angle that is equal to or closest to 0. Although it is possible to use all the dynamic modes, we limit our choice to a single mode because in our preliminary experiments, this configuration produced already very satisfactory classification performance.

**Optimal LBP parameters**: For the LBP, we have to determine the optimal block size from the configurations \( \{1 \times 1, 3 \times 3, 5 \times 5\} \) and the radius from \( \{1, 2, 3\} \) pixels (in the DMD domain). The optimal block size and radius are determined experimentally.

For the SVM, the histogram intersection kernel does not have any parameter to tune; and we shall use the standard SVM parameters without modification.

3. **Comparison of DMD with the state-of-the-art methods**. Once the optimal parameters have been determined on the development set, following the published Print Attack and Replay Attack protocols, the DMD-LBP-SVM pipeline is then applied to the evaluation (test) set. The interest in this experiment is to benchmark the generalisation performance of DMD against the state-of-the-art methods.

4. **Effect of window size on the system performance**. It is of interest to find out the impact of video window size on the generalisation performance. For this reason, we consider the window sets of top \( X \times 10 \)-frames from the video sequence, where \( X \) ranges from 1, 2, \ldots, 240 (240 is the maximum number of available frames on the Print and Replay Attack databases). We hypothesise that performance of DMD should improve with an increasing size of images available within a video.

5. **Additional research questions**. In the subsequent experiments, we shall also address three issues related to the peculiarities of the database and the DMD classification pipeline.

a) **The impact of the face cropping on original images**. We note that in the replay-attack dataset, the faces in the attack videos are larger in comparison to the faces found in the valid videos. In order to investigate if the face size might influence the classification results, we consider only the face regions – as defined by the output of a face detector – with a background window of proportional size.

b) **Pipeline with and without LBP features**. This experiment is designed to assess how much contribution is brought by LBP in the DMD +LBP+SVM pipeline. In theory, if LBP can successfully capture the video dynamics, one could simply do without DMD. In order to create this pipeline (without LBP), we directly use the histogram intensity of the chosen DMD mode – referred to as “DMD features” – and use the SVM with histogram intersection kernel as before. This results in a simplified pipeline labelled as DMD+SVM.
c) **Comparison with PCA as a baseline method.** From the outset, supported by the literature in computation fluid dynamics, we claim that DMD gives better results than PCA for dynamic data [3, 4]. This experiment is, thus, designed to compare both methods. We note that this comparison is interesting in its own right because both methods are data-driven; and in addition, they share similar matrix decomposition algorithms, but differ only in the information being exploited. In particular, DMD is a motion-based method (thus capturing the dynamics in a video) whereas PCA is a nonmotion-based method. By comparing these two methods, we can gauge how much improvement is brought by the visual dynamics as captured by DMD; and as a result, how much degradation one might incur when this information is not exploited.

6. **Repeatability of experimental results on another dataset.** As a final experiment, we shall also investigate the performance of the proposed pipeline on a different dataset i.e., the CASIA-FASD database. Not only is this database more challenging, it also contains another variant of the spoof attack produced by a 2D printed face photo with perforated eyes (cut photo), thus simulating eye-blinking. Since DMD is completely data-driven, we hypothesise that DMD will capture the eye-blinking as well as the artefacts due to the printed face. This would present a challenging task for any facial liveness detection algorithm.

Finally, we conclude this experimental section by summarising the results of the overall experiments and by comparing with the existing state-of-the-art methods.

### 5.5.1 Potential of DMD

Figure 5.3 shows a valid video, a print attack video and a reply attack video, each recorded under either controlled or adverse conditions (in the top row). The middle and bottom rows show their corresponding first dynamic mode and PCA mode images. DMD captures the changes, fluctuations in intensities, and small variations obtained by suppressing the stable information present within the videos. Hence, in this case, DMD is able to suppress the background information from the video sequence (which is stationary). DMD not only captured the movements in the facial expressions like eye blinks and lip movements but also the facial texture of the person inside the video frames. For the valid access, the texture of the person is weaker. On the other hand, we notice a stronger texture in the dynamic mode when the attack was carried out by a video.

It should be noted that the dynamic mode for the print attack displays a corrupted facial image. One reason is that a spoofing video with a printed photograph may have captured uniform illumination from the laptop screen with no dynamics inside the video. Another reason may be that there is no significant fluctuations
within the intensities of this spoof video. As a result, the first DMD mode does not contain any distinctive movement.

In the case of controlled scenario the background was plain, with some shadings and shadows, as shown in the original videos. However, in the DMD mode, the background is completely suppressed. In addition, we observe that the dynamic mode for printed photograph attack contains a corrupted version of the individual’s image with the background suppressed. These images show the strength of DMD as a preprocessing technique to distinguish between a valid and printed photograph attack.

5.5.2 Selection of DMD modes

For \( N \) number of frames, we obtain \( N - 1 \) DMD modes. We calculate the phase angles based on the complex eigenvalues and select the eigenvector which has the eigenvalue phase angle = 0 (or the closest value = 0) and compute the dynamic mode.

Figure 5.4 shows the eigenvalues of upper Hessenberg matrix \( H \), which represents the mapping between video frames. Unstable eigenvalues are located outside the circle \([3, 18]\) and stable eigenvalues can be found on the circle \([4, 17]\). For the selection of modes, we calculate the phase angle for each of the eigenvalues, i.e, the eigenvalue at \((1,0)\) has phase angle = 0 and captures the overall dynamics from the video sequence. The phase angles above the axis \([(-1,0),(1,0)]\) correspond to positive phase and negative phase angles respectively. Each of these dynamic modes captures various dynamic information pertaining to the video sequence, from large scale to small scale, as shown in Figure 5.5.

Figure 5.5 shows the first 6 DMD modes of the three video samples of a valid
Figure 5.4: Eigenvalue plot of DMD for one of the video clips in the replay-attack dataset. Each dot in the circle is a complex eigenvalue. If there are $N$ images in the video, there are $N - 1$ complex eigenvalues.

Figure 5.5: DMD modes from 1 to 6 (with HTER in % for test set on replay-attack dataset shown in brackets) showing the dynamics within the video frame sequence for valid access (top row), photo (middle row) and video attacks (bottom row).

video, photo attack and replay attack. These DMD modes are ordered in increasing phase angles with respect to the axis $[(-1, 0), (1, 0)]$. We observe that modes 4 and 6 of the valid video capture the complex dynamics of eye-blinking whereas modes 1 and 3 captures the lips movement of the video attack. For the print attack, DMD simply captures a non-uniform motion. The captured pattern does not have vitality signs such as eye-blinking, lips movement or head movement. The DMD modes for valid ones are observed with a different texture compared to video attacks. These properties may be sufficient to distinguish between a valid access and a spoof attack. We opted for dynamic mode-1 in our experiments as it gave us the best results by cross validation. This is supported by the literature [1–4,17,18] which indicates that the first dynamic mode captures the largest dynamic scale or principal motion that is present in the sequence of images. In the next subsection, we assess the choice of
5.5. EXPERIMENTS & RESULTS

LBP on the classification performance.

5.5.3 Impact of LBP parameters on the classification performance

On the replay-attack dataset

Recall that our implementation pipeline includes DMD, LBP and SVM (see Figure 5.2). In the training phase, the dynamic mode images corresponding to all the videos within the training set are given as an input to the LBP algorithm to obtain the LBP histogram features. The effectiveness of LBP depends on the different parameters for the block-size $B = \{1 \times 1, 3 \times 3, 5 \times 5\}$ and radius $R = \{1, 2, 3\}$. These parameters are optimised by minimising HTER on the development set. We report our experiments on optimisation of these parameters for different block-size and radius of LBP codes on print and replay-attack datasets.

On the training set, a HTER of 0% is attained across $B = \{1 \times 1, 3 \times 3, 5 \times 5\}$ and radius $R = \{1, 2, 3\}$ as shown in Figure 5.6 (right). For this reason, we opt to concatenate $3 - \text{LBP}_{8,1}^{3}$ and $5 - \text{LBP}_{8,1}^{5}$ features to form a larger feature vector. This hybrid feature vector can better capture the spatial arrangement of the texture properties within the video dynamics. The resultant LBP feature is of a dimensionality $1 - by - 2006$ ($9 \times 59 + 25 \times 59$; here 9 represents the $3 \times 3$ blocks, 25 represents the $5 \times 5$ blocks and 59 is the number of bins in the histogram). These concatenated LBP features for the train set are then fed to an SVM classifier with intersection kernel to train the classifier. We have not performed any parameter tuning on the SVM classifier but have selected 1 as the near boundary coefficient. We then proceed in a similar way to obtain the LBP features for videos in the development set and test set. The concatenated LBP features from the development set are fed into the trained SVM classifier to detect whether the input video is a spoof or not. At this stage, we use the class density distributions to obtain the threshold value $= -0.1370$. Later, this threshold value is used on the test set to discriminate between valid and spoofed videos.

On print-attack dataset

We proceed with a similar implementation process as in the case of the replay-attack in Section 5.5.3. Figure 5.6 (left), however, shows better results when compared to Figure 5.6 (right); in this, we observe an HTER of zero for $3 - \text{LBP}_{8,1}^{3}$, $3 - \text{LBP}_{8,2}^{3}$, $3 - \text{LBP}_{8,3}^{5}$ on the training, development, and test sets respectively. Hence, it was sufficient to consider $3 - \text{LBP}_{8,1}^{3}$ for the print-attack dataset and no further concatenation of LBP codes was required.

Based on the development set, it is clear that $3 - \text{LBP}_{8,1}^{3}$ is a better choice for both print and replay-attack datasets. This turns out to be consistent with Chingovska et al.’s [95] work, although they worked in the spatial, static-image domain, whereas our method operates in the spatial, dynamic domain.
Figure 5.6: Impact of block-size and radius of LBP on the HTER across the print-attack dataset (left) and replay-attack dataset (right). An HTER of zero is observed for $3-LBP_{8,1}^u$, $3-LBP_{8,2}^u$, $3-LBP_{8,3}^u$ on the development set on the print-attack dataset. $3-LBP_{8,1}^u$ and $5-LBP_{8,1}^u$ gave better performance on the replay-attack dataset.

Table 5.6: Comparison of the proposed pipeline with the competition methods.

<table>
<thead>
<tr>
<th>Team</th>
<th>Development</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASIA</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>IGD</td>
<td>6.67</td>
<td>9.13</td>
</tr>
<tr>
<td>MaskDown</td>
<td>0.5</td>
<td>2.50</td>
</tr>
<tr>
<td>LNMIIT</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>MUVIS</td>
<td>0.00</td>
<td>1.25</td>
</tr>
<tr>
<td>PRA Lab</td>
<td>0.00</td>
<td>1.25</td>
</tr>
<tr>
<td>ATVS</td>
<td>0.83</td>
<td>12.00</td>
</tr>
<tr>
<td>Unicamp</td>
<td>9.83</td>
<td>15.62</td>
</tr>
<tr>
<td>Our method</td>
<td>0.5</td>
<td>0.00</td>
</tr>
</tbody>
</table>

5.5.4 Performance at the video level

On replay-attack dataset

Using the threshold value $= -0.1370$ obtained on the development set, we compute the HTER score on the test set. The HTER scores of the propositions submitted to the 2nd competition on "Counter Measures to 2D Face Spoofing Attacks" [122] competition were evaluated on the replay-attack dataset. The CASIA and LNMIIT teams achieved 0% HTER on both the development and the test set. Our method showed 0.5% HTER on development set and 0% on test set.

On print-attack dataset

The results for the print-attack at video level reveal a HTER values of 0%, 0% and 0% on training, development, and test sets respectively. These results were also
Table 5.7: HTER on print-attack and replay-attack datasets

<table>
<thead>
<tr>
<th></th>
<th>Train</th>
<th>Development</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Print</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

achieved by the *UOULU* and *IDIAP* teams in the 1st competition on "Counter Measures to 2D Face Spoofing Attacks" [97] by using texture analysis methods.

5.5.5 Performance of window based approach

Since DMD needs to operate with a sequence of video frames, it is of interest to find out the minimum number of frames required. We hypothesise that more frames should result in more stable performance; however, at the same time, we would like to know the minimum number of frames that is required to distinguish a valid access from attack videos. Consequently, we select the windows of the first 10, 20, \cdots 240 frames from the training videos and evaluate it on development and test sets. We can consider at most 240 frames because all attack videos are truncated to 240 frames.

**On replay-attack dataset**

The first 10, 20, \cdots 240 frames in the video sequence are considered and given as an input to the DMD algorithm. Therefore, for every training video in print and replay-attack datasets, we obtain 24 sets of windows, each of which differ by an increment of 10 frames. We then use the same LBP parameters as reported in Section 5.5.3, to train each set using the SVM classifier and record the threshold value on the development set. The trained SVM is then applied to the test set to record the classification.

Figure 5.7 (right) shows HTER values for the first 10, 20, \cdots 240 frames in the video sequence of the replay-attack respectively. We notice that the HTER on the train set was 0% in all the cases and the HTER on the development and the test set follows a decreasing trend as the number of frames in the train set are increased. Hence, this validates our hypothesis that more frames result in a more stable performance. Although this is consistent with our expectation, the result also reveals that significantly more frames are required to detect replayed video attacks using tablets and mobile devices than the print attack (to be reported next)

The same experiment is repeated with the print attack and DMD attains 0% HTER, i.e., perfect classification. with just the first 10 frames across the training, development, and tests as shown in Figure 5.7 (left). This result compares favourably with Anjos et al.’s (2011) [94], in which the authors showed that 140 images are required to attain their best recognition performance. The method is based on optical flow. An enhanced version of their methodology in [108] requires more amount of images as an input for achieving improved results. This suggests
that DMD pipeline can capture dynamics information more efficiently than methods based on optical flow features.

### 5.5.6 Experiments on additional research questions

#### Performance on cropped frames

Due to the fact that spoof attacks in the replay-attack dataset have larger faces in comparison to the faces of the valid access, it is of interest to investigate if this might influence the classification results. Therefore we evaluate our experimental protocol only when the face regions are considered.

For this reason, we implement Viola-Jones face detection algorithm [123] to detect the face regions. All of the cropped images were of size 64 × 64. Examples of the valid access and spoof attacks in controlled and adverse scenarios for cropped
5.5. EXPERIMENTS & RESULTS

Table 5.8: Results in HTER % on replay and print-attack datasets for DMD+SVM, DMD+LBP+SVM (face regions and entire video), PCA+SVM and PCA+LBP+SVM (face regions and entire video) features

<table>
<thead>
<tr>
<th>Pipelines</th>
<th>Replay Attack Dev</th>
<th>Replay Attack Test</th>
<th>Print Attack Dev</th>
<th>Print Attack Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>DMD+SVM (face region)</td>
<td>8.50</td>
<td>7.50</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>DMD+LBP+SVM (face region)</td>
<td>5.33</td>
<td>3.75</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PCA+SVM (face region)</td>
<td>20.00</td>
<td>21.50</td>
<td>16.25</td>
<td>15.11</td>
</tr>
<tr>
<td>PCA+LBP (face region)</td>
<td>11.67</td>
<td>17.50</td>
<td>9.50</td>
<td>5.11</td>
</tr>
<tr>
<td>DMD+LBP+SVM (entire video)</td>
<td>0.50</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
</tr>
<tr>
<td>PCA+LBP+SVM (entire video)</td>
<td>21.75</td>
<td>20.50</td>
<td>11.50</td>
<td>9.50</td>
</tr>
</tbody>
</table>

Figure 5.8: Examples of the valid access and spoof attacks in controlled and adverse scenario for cropped face regions. The top row shows original images of valid access, photo and video attacks (left to right respectively). The middle row shows their corresponding first DMD mode and bottom row shows their corresponding first PCA mode.

face regions is shown in Figure 5.8. The top row shows original images of valid access, photo and video attacks (left to right respectively). The middle row shows their corresponding first DMD mode and bottom row shows their corresponding first PCA mode. Similar experimental protocol was followed and the results are shown in Table 5.8. We see that an HTER of 5.3% was recorded on the development set of the replay-attack dataset and an HTER of 3.7% was recorded on the test set. The print attack recorded 0% on both the development and test sets.

DMD features vs LBP features

It is of interest to see whether LBP could bring any improvements to our proposed pipeline. Therefore, we investigate this setting by feeding DMD features directly to the SVM classifier. We do not investigate this scenario on the full images because the resultant image dimension will be prohibitively large (The full frame size is 240 × 320 and the resultant feature vector would be a size of 76800 × V, where V is
the total number of videos). In the case of cropped images the feature vector is of size $4096 \times V$ since the frame size is $64 \times 64$. The results for DMD features recorded an HTER of 8.5% and 7.5% on development and test set respectively as shown in Table 5.8. In the case of the print attack both development and test set have an HTER = 0. When compared to DMD+LBP features we see an HTER of 5.3% and 3.7% being recorded on the development and test sets of the replay-attack dataset. Therefore, we clearly see that the classification performance is improved using the LBP feature of the DMD+LBP+SVM pipeline.

**DMD vs PCA**

In order to further indicate the potential significance of the DMD method we compare our results with the PCA algorithm as a baseline method. The results for PCA features give an HTER of 20% and 21.5% on the development and test sets of the replay-attack dataset respectively as shown in Table 5.8. PCA+LBP features recorded an HTER of 11.6% and 17.5% on development and test set respectively. In the case of the print attack, both development and test sets have an HTER of 16.2% and 15.1% for PCA features while PCA+LBP recorded an HTER of 9.5% and 5.1% on development and test sets. Comparing to the results of DMD+SVM and DMD+LBP+SVM (entire video & face regions), PCA+SVM and PCA+LBP+SVM (entire video & face regions) show lower classification performance.

**5.5.7 Proposed pipeline on CASIA-FASD Dataset**

We have only considered the face regions for this dataset as the background of videos for different types of attacks are different, and this could possibly influence the classification performance. Examples of the valid and spoof attacks in CASIA-FASD (High Resolution HR) are shown in Figure 5.9. The top row shows original images of valid access, photo, cut photo and video attacks (left to right respectively). The middle row shows their corresponding first DMD mode and the bottom row shows their corresponding first PCA mode. The first DMD mode for the cut photo attack clearly shows high intensity values in the eye region as the eyes were continuously blinking for this type of attack.

Similar experimental protocols have been followed as above and the results are listed in Table 5.9. Once again, we observe that the DMD+LBP+SVM pipeline attains the best performance, although the error rates are considerably higher than those reported on the print and replay attack datasets.

**5.5.8 Overall Summary**

In this study, we have demonstrated the potential of DMD as a preprocessing technique in highlighting the lip movements, facial textures and eye blinks across an entire video sequence. We have studied the effects of LBP parameters on the classification performance. Concatenation of the LBP codes $3 - \text{LBP}_{8,1}^3$, $5 - \text{LBP}_{8,1}^3$ to
5.5. EXPERIMENTS & RESULTS

Figure 5.9: Examples of the valid and spoof attacks in CASIA-FASD (High Resolution HR). The top row shows original images of valid access, photo, cut photo and video attacks (left to right respectively). The middle row shows their corresponding first DMD mode and bottom row shows their corresponding first PCA mode.

Table 5.9: Classification performance in terms of HTER (%) on the on CASIA-FASD dataset for DMD+SVM, DMD+LBP+SVM, PCA+SVM and PCA+LBP+SVM

<table>
<thead>
<tr>
<th></th>
<th>On face region</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>DMD+SVM</td>
<td>29.50</td>
<td></td>
</tr>
<tr>
<td>DMD+LBP+SVM</td>
<td>21.75</td>
<td></td>
</tr>
<tr>
<td>PCA+SVM</td>
<td>33.50</td>
<td></td>
</tr>
<tr>
<td>PCA+LBP+SVM</td>
<td>24.50</td>
<td></td>
</tr>
</tbody>
</table>

obtain an LBP code of dimensionality $1 - by - 2006$ performed well on the replay-attack dataset. For the print-attack dataset there is no need to concatenate LBP codes, since the codes corresponding to $3 - \text{LBP}_{8,1}^{2}$, $3 - \text{LBP}_{8,2}^{2}$ and $3 - \text{LBP}_{8,3}^{2}$ demonstrated an HTER of 0% across training, development, and test sets respectively. We have also conducted experiments at both the video level and frame level. At the video level, the proposed methodology achieves perfect recognition performance at HTER of 0% on the print attack dataset; whereas for the replay attack, 0.5% and 0% is recorded on the development and test sets, respectively. These results show a good separability between the spoof and non-spoof samples for the considered datasets. Since DMD needs to operate with a sequence of frames in a video, it is desirable to find out the minimum number of frames required. For this purpose, we have repeated the experiments with progressively increasing frame sizes.
Table 5.10: Comparison of HTER (%) on test sets for the proposed pipeline with respect to the current state of the art. Here, E: on entire video and F: on face regions.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Print attack</th>
<th>Replay attack</th>
<th>CASIA-FASD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anjos et al. (2013) [108]</td>
<td>1.52</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Schwartz et al. (2011) [104]</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Pereira et al. (2013) [124]</td>
<td>-</td>
<td>11.79</td>
<td>30.33</td>
</tr>
<tr>
<td>Proposed method</td>
<td>0.00</td>
<td>0.00</td>
<td>-</td>
</tr>
<tr>
<td>DMD+LBP+SVM</td>
<td>0.00</td>
<td>3.75</td>
<td>21.75</td>
</tr>
<tr>
<td>DMD+LBP+SVM\textsuperscript{E}</td>
<td>0.00</td>
<td>7.50</td>
<td>29.50</td>
</tr>
<tr>
<td>PCA+SVM\textsuperscript{F}</td>
<td>15.11</td>
<td>21.50</td>
<td>33.50</td>
</tr>
<tr>
<td>PCA+LBP+SVM\textsuperscript{F}</td>
<td>5.11</td>
<td>17.11</td>
<td>24.50</td>
</tr>
<tr>
<td>PCA+LBP+SVM\textsuperscript{E}</td>
<td>9.50</td>
<td>20.50</td>
<td>-</td>
</tr>
</tbody>
</table>

of 10, 20, · · · 240. Consistent with our hypothesis, more frames result in more stable performance. For the print-attack dataset, our algorithm produces 0% HTER across all the datasets. This implies that the artefacts produced by the print attack videos do not contain any long-term dynamics; and that the attack can be detected by DMD effectively. In addition, we have addressed related research issues such as: i) evaluating the experimental protocol on face regions instead of evaluating on the entire video ii) comparing DMD with a baseline method i.e PCA, iii) addressing the importance of LBP in the methodological pipeline and finally evaluating the performance on a different dataset with diversified attacks. The experimental results show that: i) DMD+LBP+SVM pipeline on the face regions degrades slightly in performance when compared to the analysis on entire videos by HTER 3% – 6%. ii) DMD based pipelines such as DMD+SVM and DMD+LBP+SVM work better than the PCA based pipelines, thus demonstrating the significance of motion based methods in distinguishing between valid accesses and spoof attacks compared to non-motion based methods such as PCA. Our experiments on CASIA-FASD dataset also show the strength of the proposed DMD+LBP+SVM pipeline. The overall summary of the results with the considered pipelines is shown in Figure 5.10. Our experimental results, following strictly the published experimental protocols, show that the proposed DMD pipeline achieved the best performance reported so far with an exception of Komulainen et al.’s method [111]. The authors proposed
to use context information such as facial, upper body and background information. In our case, we just used the facial features because our primary objective is to evaluate the effectiveness of DMD for distinguishing valid videos from the spoofed ones given only facial information. Komulainen et al.’s [111] experiment informs us that using contextual information is extremely important. Indeed, when DMD is given the entire video frame instead of the face region only, the DMD+LBP+SVM pipeline also attains zero HTER (see Table 5.10, proposed methods). This experimental setting is particularly interesting because the faces in the replayed videos have slightly larger size than those of the valid videos. This was designed to be so in order to avoid the frames of the replay device (tablet or mobile phone) from being captured and subsequently exploited by the algorithm designer. Nevertheless, DMD inadvertently captures the subtle cue in facial size difference, leading to perfect recognition on the test set. It is, therefore, reasonable to expect improved performance with DMD given more contextual information such as the upper body part as in komulainen et al.’s work. Our advantage, in our case is that the same DMD+LBP+SVM pipeline can be used straightforwardly without modification.

Finally, to conclude this summary section, we compare the experimental results obtained so far using variants of DMD with various methods reported in the literature. The results as indicated in Table 5.10 demonstrate that DMD compares favourably, and even exceeds, current state-of-the-art methods.
5.6 Conclusions

This study shows the significance of the DMD method as a preprocessing technique when coupled with LBP and SVM to effectively detect spoof samples. We applied the DMD method to 1200 video clips of photo and video attacks on 50 clients, under different lighting conditions acquired from the replay-attack dataset; 400 video clips of photo attacks from the print-attack dataset; and 600 video clips from the CASIA-FASD dataset. The results are exceedingly promising in tackling the photo, cut photo and video attack challenges.

DMD can extract temporal dynamics efficiently in a data-driven manner and the resultant “texture dynamics” can be efficiently represented by LBP features with appropriate parametrisation. In the tested scenarios, \(3 - \text{LBP}_{8,1}^2\) appears to be suitable across the various print and replay attacks that we have tested on. This parameter is likely to be dependent on image resolution and camera quality. The SVM histogram intersection kernel works particularly well with the histogram features derived from LBP, with the additional convenience of no additional parameter tuning being required. The pipeline as a whole is thus efficient and easy to use, DMD and SVM in combination with the histogram intersection kernel hence requiring no tuning. Compared to other approaches reported to date, DMD appears to give superior performance. We attribute this to (1) the capacity of DMD to extract the dynamics of video sequence, as well as (2) the choice of pipeline within which DMD is deployed. Since DMD can capture video dynamics such as blinking eyes or mouth movements, or other non-obvious and subtle cues, this spares us from computing features that explicitly identify these cues. This is the main difference between our approach and cue-based methods. Although our work also makes use of LBP, it is used to represent the texture dynamics as opposed to static textures such as in the approach of Chingovska et al.’s [95]. We also note that a number of other methods have attempted to directly exploit dynamics such as those reported in Chakka et al.’s [97] work. However, the performance reported in our work arises mainly from the capability of DMD to extract the dynamics of video automatically, and from the unique combination of DMD+LBP+SVM in a classification pipeline.
Chapter 6

Finger Vein Spoof Detection: Micro Texture Descriptor

Recent studies have shown that it is possible to attack a finger vein (FV) based biometric system using printed materials. In this study, we propose a novel method to detect spoofing of static finger vein images using Windowed Dynamic mode decomposition (W-DMD). This is an atemporal variant of the recently proposed Dynamic Mode Decomposition for image sequences. The proposed method achieves better results when compared to established methods such as local binary patterns (LBP), discrete wavelet transforms (DWT), histogram of gradients (HoG), and filter methods such as range-filters, standard deviation filters (STD) and entropy filters, when using SVM with a minimum intersection kernel. The overall pipeline which consists of W-DMD and SVM, proves to be efficient, and convenient to use, given the absence of additional parameter tuning requirements. The effectiveness of our methodology is demonstrated using FV-Spoofing-Attack database which is publicly available. Our test results show that W-DMD can successfully detect printed finger vein images because they contain micro-level artefacts that not only differ in quality but also in light reflection properties compared to valid/live finger vein images.

6.1 Introduction

Biometric techniques for person authentication, have significant advantages over keys, passwords, and PIN numbers because these conventional authentication mechanisms can be stolen, lost, or forgotten. Hence we have seen massive progress in research and deployment of different biometrics for person authentication such as face, fingerprint, iris, finger vein, signature, and other modalities [125–139].

Biometric spoofing, which is also known as presentation attack [88], refers to the
using of fake biometric data in order to deceive a biometric system [83]. Fingerprints and palm prints can be frayed but also replicated using everyday materials such as silicon gel, gums, and other materials, as reported in [140] and elsewhere. Indeed, there have been many successful spoofing attempts reported in the literature using commercial systems, e.g., [125, 141]. Fortunately, countermeasures are also being developed to neutralise the attacks [142].

Finger vein biometric trait as an authentication method has gained popularity in recent years, particularly within the financial sector. Unlike finger prints, finger vein patterns exist inside the human body, thus eliminating tampering and providing an ideal identification method without being intrusive. It is widely used in Japan, China, Poland, and Turkey [143] and recently banks in the UK have also planned to introduce this technology ¹. However, a recent successful spoofing attempt on finger vein biometric was reported by IDIAP [144]. Although the attack was carried out under the user’s cooperation, the uncooperative scenario is certainly more realistic. This shows the vulnerability of the system and suggests that finger vein, like many other biometrics, is not free from spoofing, as commonly believed.

6.1.1 Existing countermeasures in Finger vein spoofing

IDIAP initiated an open competition for FV-spoofing counter-measures. Their article [143] reported 4 different approaches from the competition, which are briefly described here.

IDIAP proposed a texture-based algorithm exploiting the frequency domain of the images. The results from their algorithm were the baseline for the competition. GUC team used a texture-based approach extracting binarized statistical image features (BSIF) [145], which represent each pixel as a binary code obtained by computing its response to a filter [146] learnt using the statistical properties of natural images [147]. B-lab team utilised monogenic scale space (MSS) [148] based global descriptors. This method extracts the local energy and local orientation at a coarse level, allowing location of the difference between spoofed and non-spoofed vein blocks. The winning team, GRIPPRIAMUS used local descriptors such as $3\text{-LBP}_{8,1}^{u2}$ [149] for cropped images and Local Phase Quantization (LPQ) [150] and Weber Local Descriptor (WLD) [151] on full images. The results from the competition show that texture based methods are one of the best counter-measure strategies for detecting the finger vein spoofing attacks especially on printed media.

Despite the difference in the nature between finger vein counter spoofing and finger vein recognition, both tasks typically rely on texture or feature extraction with respect to a single finger vein image acquired using an infra red or near infra red camera. Our hypothesis is that our proposed W-DMD method will thus also differentiate spoofed and non-spoofed finger vein biometrics access using a single image.

¹http://www.bbc.co.uk/news/business-29062901
Our proposed method, W-DMD is a variant of the DMD method that has been used to distinguish a live access from printed media or replayed videos by exploiting the information dynamics of the video content, such as blinking eyes, moving lips, and facial dynamics in an entirely data-driven approach. In general, DMD is used to analyse a set of image sequences; in this chapter, however, we present an approach, W-DMD, exclusively for analysing an image in order to detect spoofed finger vein images.

We also aim to find answers for the following questions:

1. If DMD can capture temporal information then would W-DMD capture texture information?
2. If DMD can capture principal movements then would W-DMD capture texture gradients?
3. What would be the effect of texture gradients on classification performance?
4. How effective is the W-DMD compared to the plethora of existing descriptors?

6.1.2 Preliminary analysis: Window-DMD

Having examined the images of valid versus spoofed finger vein image samples, we observe that both types of images differ in terms of image quality and light reflection. For example, a valid finger vein image and a printed one reflect light in different ways because a vein is a complex non rigid 3D object whereas a photograph can be seen as a planar rigid 2D object. This may cause different reflections and shading. The surface properties of valid finger vein compared to the printed ones are also different. We therefore propose to solve the finger vein spoof detection problem by analysing the finger vein for capturing the aforementioned information at micro-texture level.

Experimental analysis: Visual examination

The effectiveness of the proposed W-DMD is demonstrated for cropped finger vein images in Figure 6.1 and for full images in Figure 6.2. The top row in these figures show the raw images for the valid access and spoof, whereas the middle and bottom rows show the first two W-DMD components of the raw images respectively. As can be observed, although the raw images are not visibly different from one another, the decomposed components by W-DMD clearly show that the texture properties of both types of images are distinctive. In particular, the first W-DMD component can capture the maximum variances at column levels whereas its second component captures the residuals or noise. Therefore, our preliminary visual examination shows that W-DMD is potentially a useful approach not only to solve FV spoof detection problem, but also for addressing the biometric person authentication task; however, the latter topic falls outside the scope of this study.
Figure 6.1: Top row shows the raw images for valid access and spoof from the same person. Middle row shows the W-DMD component-1 of the raw image and bottom row shows the W-DMD component-2 of the raw image.

Figure 6.2: Top row shows the raw images for valid access and spoof from the same person. Middle row shows W-DMD component-1 of the raw image and bottom row shows W-DMD component-2 of the raw image.

6.2 Texture Methods

In reference to Table 6.1, we divide the features by two characteristics, namely, spatial versus frequency domain; and global versus local representation. By spatial representation, we understand that a particular feature extracts information at the pixel level, whereas by frequency domain, we imply that the feature representation
captures image information using Fourier transforms or other basis functions. A feature representation is considered local if it captures information from neighbouring pixels (with respect to a reference pixel). A global feature is one that summarises the information from a given image patch. The proposed W-DMD is considered a spatial approach that represents data globally. As a result, it is not necessary to divide an image into sub-blocks; although this depends on the size of the image and the information content therein.

Table 6.1 shows the different texture features commonly reported in the literature; they include Discrete Cosine Transform (DCT), Local Binary Pattern (LBP), Histogram of Gradient (HoG), Discrete Wavelet Transform (DWT) and a number of filters. We propose W-DMD as a novel addition to the above list; in doing so, we also develop the existing DMD method by applying it on a single image. Our pipeline involves: (i) decomposing candidate images into components and (ii) classification via Support Vector Machines with a minimum intersection kernel. The pipeline is shown in Figure 6.4.

Table 6.1: categorisation of the texture analysis methods based on spatial or frequency and local or global approach.

<table>
<thead>
<tr>
<th>Method</th>
<th>Spatial or frequency</th>
<th>Local or global</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBP</td>
<td>Spatial</td>
<td>Local</td>
</tr>
<tr>
<td>W-DMD</td>
<td>Spatial domain</td>
<td>Global</td>
</tr>
<tr>
<td>DWT</td>
<td>Frequency domain</td>
<td>Global and local</td>
</tr>
<tr>
<td>DCT</td>
<td>Frequency domain</td>
<td>Global and local</td>
</tr>
<tr>
<td>HoG</td>
<td>Spatial</td>
<td>Global</td>
</tr>
<tr>
<td>Filters</td>
<td>Spatial</td>
<td>Local</td>
</tr>
</tbody>
</table>

6.2.1 Local Binary Patterns (LBP)

The local binary pattern, introduced by Ojala et al. [149] is a powerful operator which extracts micro-texture information that is invariant to local grey scale variations. During the LBP operation, every image pixel acts as a threshold to its neighbours to obtain a binary bit string which then forms a round number. In real applications, an image is usually divided into several small non-overlapping blocks of the same size to keep the spatial relation of objects. Conventionally, LBP is denoted as $B_{P,R}^{u2}$, where $P$ and $R$ denote the neighbourhood pixels and radius respectively and the superscript $u2$ denotes uniform LBP. $B$ denotes the number of non-overlapping blocks. In this study, we have considered $P = 8$ for block-size = $3 \times 3$ and radii = 1, i.e $3-LBP_{8,1}^{u2}$.

6.2.2 Discrete Wavelet Transform (DWT)

The wavelet transforms are used to analyse a signal (image) at different frequencies with different resolutions. It represents the same signal, but corresponding to
different frequency bands. Many wavelet families have been developed with different properties [152]. For 2-D images, applying DWT corresponds to processing the image by 2-D filters in each dimension. The filters divide the input image into (Higher(L), Lower (L)) four non-overlapping multi-resolution sub-bands LL, LH, HL and HH. The sub-band LL represents the coarse-scale DWT coefficients while the sub-bands LH, HL and HH represent the fine-scale of DWT coefficients. We have used 'Haar' with LH sub band in this study.

6.2.3 Discrete Cosine Transform (DCT)

The DCT transforms input signal in the spatial domain to a frequency domain. DCT has become a popular technique in image processing for its compression ability, capturing the intrinsic properties of an image with only few coefficients. The DCT coefficients, which are located at the upper left corner, holds most of the image energy [153]. For instance, the top $20 \times 20 = 400$ DCT coefficients are enough to represent the image and hence considered as a feature vector in our analysis. We have used the well known form of DCT in literature i.e., DCT-II introduced by Wang [154].

6.2.4 Histogram of Gradients (HoG)

The Histogram of oriented gradients (HoG) is a feature descriptor proposed by Dalal and Triggs for human detection [155]. HoG soon gained significant attention for face recognition and texture analysis. The HoG descriptor counts the number of occurrences of gradient orientation in a localised portions of an image. The feature is represented by a histogram of the quantised angular bins according to the gradient orientation.

6.2.5 Filter based methods

The design of digital filters are central to any image processing applications. They have applications in noise elimination, image smoothing and as well useful in extracting textural properties [156].

**Entropy:** The entropy is a statistical measure of randomness that can be used to characterise the texture of the input image [157,158]. Entropy is defined as:

$$ E = - \sum p \log_2(p) $$

where $p$ contains the histogram counts of an image.

**Standard Deviation:** The standard deviation is used to estimate the variation in an image block. In our work, we have considered block of size $3 \times 3$ and estimate the standard deviation in each block which are used as feature values.
6.3. METHODOLOGY

Range: The range filter is used to estimate the difference of maximum and minimum values within a block of size $3 \times 3$ and hence stored as feature values for each block.

![Texture visualisation of full and cropped finger vein images using filter based methods and DWT.](image)

**Figure 6.3:** Texture visualisation of full and cropped finger vein images using filter based methods and DWT.

6.3 Methodology

As we have indicated in the introduction a number of pre-existing methods suitable for texture characterisation within a spoofing-detection context, we in this section briefly present how the alternative texture based methods have been used for our evaluation along with proposed Windowed Dynamic Mode Decomposition (W-DMD). We will also present the classification method employed.

The overall process pipeline used for evaluation is shown in Figure 6.4. First, an image is processed using one of the texture-based methods; output features are then fed into a trained SVM classifier in order to classify whether the processed image is a valid or spoof.

6.3.1 Window Dynamic Mode Decomposition (W-DMD)

Let $X$ be an image whose columns are given by $X = [\bar{x}_1, \bar{x}_2, \bar{x}_3, \cdots, \bar{x}_N]$. Since the consecutive columns or rows of an image are highly correlated to one another, we will assume that there exists a linear mapping $A$ from one column to the consecutive in the sequence. This enables us to represent the image $X$ within the span of Krylov subspace $[18, 78]$. 
\[ X = [\bar{x}_1, A\bar{x}_1, A^2\bar{x}_1, A^3\bar{x}_1, \ldots, A^{N-1}\bar{x}_1] \]  \hspace{1cm} (6.2) \\

In this chapter, we limit the Krylov span for 3 columns and proceed with a sliding window approach to mimic the sequential aspect of standard DMD \cite{78}. 

\[ S = [\bar{x}_1, A\bar{x}_1, A^2\bar{x}_1]. \]  \hspace{1cm} (6.3) \\

\[ [\bar{x}_2, \bar{x}_3] = A[\bar{x}_1, \bar{x}_2]. \]  \hspace{1cm} (6.4) \\

\[ S_2 = AS_1. \]

Solving eigenvectors for \( A \) is computationally more expensive. Hence, we represent Equation 6.4 as a standard Arnoldi iteration problem which introduces \( H \). The eigenvalues from \( H \) approximate some of the eigenvalues of the full system \( A \). 

\[ AS_1 \approx S_1 H. \] \\
\[ S_2 \approx S_1 H. \]  \hspace{1cm} (6.5) \\

\[ AS_1 \approx S_2 \approx S_1 H. \]

The associated eigenvectors of \( H \) thus provide the coefficients for the linear combination that is necessary to express the spatial variations from one column of image to another.

Let us define \( L \) and \( U \) to be the matrices as a result of applying \( LU \) decomposition on \( S_1 \) and \( L^+ \) be the pseudo inverse of \( L \). The \( H \) matrix can then be calculated by the equation:

\[ H = U^{-1}L^+ S_2 \]  \hspace{1cm} (6.6) \\

For the first three columns we obtain two dynamic modes (in general for \( N \) columns, we get \( N-1 \) DMD modes \cite{78}). In the next step we exclude the first column and consider columns \( \{2, 3, 4\} \), followed by \( \{3, 4, 5\} \) and \( \{4, 5, 6\} \) and so on as shown in Figure 6.5, for capturing these two dynamic modes. The first dynamic mode \( DM \) captures the intensity variations across the columns and second dynamic mode \( NM \) captures the noise residuals. Finally, we concatenate all of the \( DMs \) to obtain
6.3. METHODOLOGY

W-DMD component-1 (W-DMD (C1)) and similarly concatenation of the NMs produce W-DMD component-2 (W-DMD (C2)). This procedure is summarised in Algorithm 3).

![Diagram showing the working mechanism of W-DMD]

Figure 6.5: Methodological pipeline showing the working mechanism of W-DMD.

**Algorithm 1 W-DMD**

**Input:** Sequences of columns in an image $X = [\bar{x}_1, \bar{x}_2, \bar{x}_3, \cdots, \bar{x}_n]$.

**Output:** Dynamic image DM, residual image NM.

1. for $\{ i = 1 \text{ to } n - w \}$ %w = 2
2. $S = [\bar{x}_i, \bar{x}_{i+1}, \bar{x}_{i+2}]$
3. $[DM(i), NM(i)] = \text{get-WDMD}(S)$
4. endfor

**Algorithm 2 get-WDMD**

**Input:** Sequence set of columns in a window $S = [\bar{x}_1, \bar{x}_2, \bar{x}_3]$

**Output:** Dynamic column DM, residual column NM.

1. $S_1 \leftarrow [\bar{x}_1, \bar{x}_2]$
2. $S_2 \leftarrow [\bar{x}_2, \bar{x}_3]$
3. $[L \ U] \leftarrow \text{lu}(S_1)$
4. $H \leftarrow (U^{-1}L^+ S_2)$
5. $[\text{Evec} \ \text{Eval}] \leftarrow \text{eig}(H)$
6. $DM \leftarrow \text{abs}(S_1 \text{Evec}(:, 1))$
7. $NM \leftarrow \text{abs}(S_1 \text{Evec}(:, 2))$

The finger vein images in the dataset were reduced to a size of 20% of their original size prior to processing with the aforementioned methods. Table 6.2 shows the size of the resultant feature vector obtained after processing a finger vein image.
Table 6.2: Table showing the size of the resultant feature vector for each of the methods used in this study.

<table>
<thead>
<tr>
<th>Method</th>
<th>Cropped</th>
<th>Full</th>
</tr>
</thead>
<tbody>
<tr>
<td>LBP</td>
<td>1x531</td>
<td>1x531</td>
</tr>
<tr>
<td>DWT</td>
<td>1x36</td>
<td>1x70</td>
</tr>
<tr>
<td>DCT</td>
<td>1x400</td>
<td>1x400</td>
</tr>
<tr>
<td>HoG</td>
<td>1x81</td>
<td>1x81</td>
</tr>
<tr>
<td>Entropy</td>
<td>1x138</td>
<td>1x270</td>
</tr>
<tr>
<td>STD</td>
<td>1x138</td>
<td>1x270</td>
</tr>
<tr>
<td>Range</td>
<td>1x138</td>
<td>1x270</td>
</tr>
<tr>
<td>W-DMD</td>
<td>1x3330</td>
<td>1x6550</td>
</tr>
<tr>
<td>W-DMD+LBP</td>
<td>1x531</td>
<td>1x531</td>
</tr>
<tr>
<td>Raw</td>
<td>1x3390</td>
<td>1x6650</td>
</tr>
</tbody>
</table>

6.3.2 SVM-Minimum Intersection Kernel

In the following evaluation, the Minimum Intersection Kernel mapping is applied within an SVM context to transform the data into an appropriate high dimensional space in order to increase the likelihood of linear separability [117] [118]. This kernel having been shown to be efficient in image classifications [121] is defined:

\[ K(x, y) = \sum_{i=1}^{n} \min(x_i, y_i) \]  

(6.7)

Here, \( n \) is the number of dimensions in the feature vector and \( x \) and \( y \) are the features. We note that this kernel has no training parameter. For the SVM optimisation procedure, we set the slack parameter, to be 0.01 for all the experiments.

6.4 Performance criteria

Let \( \mathcal{T} \) be the domain of SVM output. The decision is made by comparing the SVM output \( t \in \mathcal{T} \) with a decision threshold, \( \Delta \in \mathcal{T} \), as follows:

\[
\text{decision}(t) = \begin{cases} 
\text{valid access} & \text{if } t > \Delta \\
\text{spoof/attack} & \text{otherwise},
\end{cases}
\]  

(6.8)

In order to calculate errors, we introduce match and nonmatch score sets, \( \mathcal{Y}_1 \subset \mathcal{Y} \) and \( \mathcal{Y}_0 \subset \mathcal{Y} \), respectively. This can result in two errors, namely, false rejection and false acceptance, the rates of which are calculated as follows:

\[
\text{FRR}(\Delta) \equiv P(t < \Delta|\mathcal{Y}_1) \\
\approx \frac{\mid \mathcal{Y} \cap \mathcal{Y}_1, \mathcal{Y} < \Delta \mid}{|\mathcal{Y}_1|} \\
\]  

(6.9)  

\[
\text{FAR}(\Delta) \equiv 1 - P(t < \Delta|\mathcal{Y}_0), \\
\approx \frac{\mid \mathcal{Y} \cap \mathcal{Y}_0, \mathcal{Y} > \Delta \mid}{|\mathcal{Y}_0|} \\
\]  

(6.11)
respectively, where the conditioning variable \( Y_1 \) indicates that the comparison is due to a positive class (valid access) whereas \( Y_0 \) indicates that the comparison is due to a negative class (spoof/attack).

Note that FRR is an increasing function of the decision threshold; whereas FAR is a decreasing function of the decision threshold. The two error rates cross over each other on Equal Error Rate, i.e.,

\[
\text{EER}(\Delta_{EER}) = \text{FAR}(\Delta_{EER}) = \text{FRR}(\Delta_{EER})
\]

where \( \Delta^* \in Y \) is the unique decision threshold of the EER. In practice, EER is found by searching for a threshold that minimises the absolute difference between FAR and FRR.

\[
\Delta_{EER} = \arg \min_{\Delta} |\text{FAR}(\Delta) - \text{FRR}(\Delta)|
\]

Since most of our experiments resulted in no observable classification errors, we resolve to estimating the system performance using parametric criteria such as F-ratio, and d-prime statistics as reported in [159]. These criteria assume that the classifier output conditioned upon the class label (valid or spoof finger veins) follows normal distribution. Since we use SVM in all cases, the output scores are not skewed and can roughly be described by the first (mean \( \mu \)) and second order (variance \( \sigma \)) moments, i.e., \( \mu^1 \) and \( \sigma^1 \) for the valid samples and \( \mu^0 \) and \( \sigma^0 \) for the spoof samples, respectively.

### 6.4.1 EER based on F-ratio

A large value of F-ratio implies higher separability; and this can be measured even when no error is observed. The F-ratio is defined as:

\[
\text{F-ratio} = \frac{\mu^1 - \mu^0}{\sigma^1 + \sigma^0}.
\]  

(6.13)

The corresponding Equal Error Rate (EER), in which one assumes that the probabilities of false acceptance and false rejection are equal, can be obtained from the F-ratio in a closed form, as follows:

\[
\text{EER} = \frac{1}{2} - \frac{1}{2} \text{erf} \left( \frac{\text{F-ratio}}{\sqrt{2}} \right) \equiv \text{EER F-ratio},
\]  

(6.14)

where “erf” is the error function.

\[
\text{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z \exp [-t^2] \, dt,
\]  

(6.15)

In [160], it is shown that the theoretical EER found using F-ratio correlates well with the empirical EER obtained by error counting.
Figure 6.6: Dataset divided into full and cropped finger vein images for valid and spoof samples.

6.5 Dataset

IDIAP’s FV-Spoofing-Attack\(^2\) dataset [143, 144] is considered for this study. The dataset consists of cropped (Figure 6.6(right)) and full versions (Figure 6.6(left)) of valid accesses and printed attacks from 110 clients (440 samples). Table 6.3 shows the number of samples divided into protocols.

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Training set</th>
<th>Development set</th>
<th>Test set</th>
</tr>
</thead>
<tbody>
<tr>
<td>full</td>
<td>120</td>
<td>120</td>
<td>200</td>
</tr>
<tr>
<td>cropped</td>
<td>120</td>
<td>120</td>
<td>200</td>
</tr>
</tbody>
</table>

6.6 Experiments and Results

Both valid and spoof images are collected for training, development and testing sets respectively (recall our methodological procedure from Figure 6.4). For each valid and spoof image the texture features are calculated and introduced to the SVM classifier with min kernel and near-boundary coefficient value = 0.01. EER based on F-ratio EER (F) and F-ratio are computed on the output of the SVM scores.

6.6.1 Selection of the W-DMD component.

W-DMD decomposes an image into two components (when \(w = 3\)), one capturing intensity variations in the image and the other the noise residuals. We therefore wish to establish which of these two components is important in detecting spoof images. We thus extract the W-DMD components from the train, development and

\(^2\)https://www.idiap.ch/dataset/fvspoofingattack
test data sets, with texture features from the development set fed into the trained SVM classifier to detect whether the input image is a spoof or not in order to use the class density distributions to obtain a threshold value (point of intersection of FAR and FRR scores) from the development set. Later, this threshold value is used on the test set to discriminate between valid and spoofed videos.

Table 6.4: Performance of the W-DMD components on cropped and full image data.

<table>
<thead>
<tr>
<th></th>
<th>Cropped EER (F)%</th>
<th>Full EER (F)%</th>
<th>F-ratio</th>
<th>Cropped F-ratio</th>
<th>Full F-ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>W-DMD (C1)</td>
<td>1.59</td>
<td>0.08</td>
<td>2.14</td>
<td>2.15</td>
<td>3.15</td>
</tr>
<tr>
<td>W-DMD (C2)</td>
<td>7.81</td>
<td>1.80</td>
<td>1.41</td>
<td>2.09</td>
<td></td>
</tr>
</tbody>
</table>

Our findings on application of this classification procedure are that W-DMD component-1, which appears to capture the variation of light reflections within images performed well when compared to W-DMD component-2. The results on cropped images recorded an EER(F) \(^3\) of 1.5% for W-DMD component-1 when compared to component-2 which has given 7.8%. For the full images, the classification achieved better results compared to cropped images, presumably since these images possess finger boundaries and light reflections. Results on full images again show the strength of W-DMD component-1, recording EER(F) of 0.8% compared to 1.8% obtained by W-DMD component-2. These results are shown in Table 6.4.

6.6.2 Comparison with state-of-the-art methods.

To evaluate the performance of the proposed method on the FV-spoofing dataset, we compare it to the indicated state-of-the-art texture analysis methods: 1) Block-LBP, 2) DWT, 3) DCT, 4) HoG and 5) filter based methods including: a) Entropy, b) Standard deviation and c) Range.

On full images

In the case of full images, we see that design filters and DCT methods give a lower performance when compared to raw input images (which is considered as a baseline in our experiments). We see that the methods W-DMD, W-DMD+LBP, DWT, LBP and HoG have all shown better performance than the baseline. W-DMD achieves the greatest F-ratio (3.15) in comparison to the rest of the methods. These results are shown in Table 6.5. The results in performance order are shown Figures 6.7 (a) & (b).

On cropped images

Similar to the results on the full image set, the design filters and DCT method give a lower performance compared to the raw image. The W-DMD method has again

\(^3\)EER calculated from F-ratio
Table 6.5: Performance of the state of the art methods in comparison with the W-DMD method for cropped and full image data.

<table>
<thead>
<tr>
<th></th>
<th>Cropped EER (F)(%)</th>
<th>Cropped F-ratio</th>
<th>Full EER (F)(%)</th>
<th>Full F-ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>W-DMD</td>
<td>1.59</td>
<td>2.14</td>
<td>0.08</td>
<td>3.15</td>
</tr>
<tr>
<td>W-DMD+LBP</td>
<td>2.98</td>
<td>1.88</td>
<td>0.10</td>
<td>3.09</td>
</tr>
<tr>
<td>DWT</td>
<td>1.89</td>
<td>2.07</td>
<td>0.16</td>
<td>2.95</td>
</tr>
<tr>
<td>LBP</td>
<td>3.04</td>
<td>1.87</td>
<td>0.22</td>
<td>2.84</td>
</tr>
<tr>
<td>HoG</td>
<td>1.74</td>
<td>2.11</td>
<td>0.46</td>
<td>2.60</td>
</tr>
<tr>
<td>RAW</td>
<td>5.34</td>
<td>1.61</td>
<td>0.61</td>
<td>2.50</td>
</tr>
<tr>
<td>STD</td>
<td>5.77</td>
<td>1.57</td>
<td>1.04</td>
<td>2.31</td>
</tr>
<tr>
<td>Range</td>
<td>5.72</td>
<td>1.57</td>
<td>1.27</td>
<td>2.23</td>
</tr>
<tr>
<td>DCT</td>
<td>15.11</td>
<td>1.03</td>
<td>3.70</td>
<td>1.78</td>
</tr>
<tr>
<td>Entropy</td>
<td>15.26</td>
<td>1.02</td>
<td>8.70</td>
<td>1.35</td>
</tr>
</tbody>
</table>

achieved remarkable performance with EER (F-ratio) of 1.5% and F-ratio of 2.14 when compared to rest of the methods as shown in Figures 6.7 (a) & (b).

The overall performance of W-DMD with respect to the rest of the methods considered in this experiment is given in Table 6.5.

Comparing both experimental configurations, it can be observed that the cropped images are harder to classify. This is expected since the border information is not available. As a result, the performance of the classifier is entirely dependent on the capability of the feature to extract micro-texture information without using additional cues.
6.6.3 Conclusions

This study shows the significance of the proposed novel method, W-DMD, as a viable feature extraction method when coupled with an SVM to effectively detect spoof samples. We applied W-DMD on finger vein images for valid and print attacks from 110 clients (240 (training) + 240 (development)). The results demonstrate that the proposed W-DMD is very promising in tackling the print attack challenge.

W-DMD can extract local variations as low rank representation inside an image capturing light reflections, illuminations and planar effects which are typically different for valid access and spoof images. This is the main intrinsic difference between W-DMD features and other existing features. The SVM with Minimum Intersection Kernel works particularly well with the W-DMD features, with the convenience of no additional parameter required for training, other than setting the near boundary coefficient value to 0.01. The entire pipeline proves to be efficient in comparison to the performance of other state of the art methods such as LBP, DWT, HoG and filter methods including entropy, std and range. Local-spatial texture based methods (LBP), Global-spatial (HoG) and frequency based (DWT) methods have shown better results when compared to the baseline raw input image. In future work, we would like to explore the effect of window size within the W-DMD method.
Chapter 7

Colour Background
Initialisation

The objective of Scene Background Initialisation (SBI) is to obtain a background model from a sequence of images where the background is occluded with a number of foreground objects. It has a number of applications, including video surveillance, video segmentation, video compression, video inpainting, privacy protection for videos, and computational photography [161]. Recent studies [6] have introduced the method of Dynamic Mode Decomposition (DMD) for robustly separating video frames into a background model and foreground components. While, the method introduced operates by converting colour images to grey scale, we in this study propose a technique to obtain the background model in the colour domain. The effectiveness of our technique is demonstrated using publicly available Scene Background Initialisation (SBI) dataset. Our results both qualitatively and quantitatively show that DMD can successfully obtain a coloured background model.

7.1 Scene Background Initialisation

The objective of Scene Background Initialisation (SBI) is to obtain a background model from a sequence of images where the background is occluded with a number of foreground objects. It has a number of applications, including video surveillance, video segmentation, video compression, video inpainting, privacy protection for videos, and computational photography [161]. The DMD method that is introduced in [6] for obtaining the background model essentially requires the conversion of colour images into grey scale images. However, it is sometimes necessary to have the scene background in the colour domain, particularly for applications that include computational photography and video inpaintings. Therefore, our novel
contribution is in modifying the DMD algorithm to obtain a colour background model, thereby avoiding the conversion of colour images to grey scale.

This section of the thesis, answers the following research questions:

1. Can DMD capture background models in the colour domain?
2. How effective is the proposed DMD technique compared to existing benchmarked methods for scene background initialisation?

**Benchmarked Methods**

A recent study by Maddalena et al. [162,163] reviews and benchmarks five different methods that are suitable for obtaining a background model from a given image sequence, namely, temporal median (baseline) method, Spatially Coherent Self-Organizing Background Subtraction (SC-SOBS) [164], WS2006 [165], RSL2011 [166] and Photomontage [167]. In the temporal median method the background model is obtained as the median of pixel values at the same location throughout the image sequence. SC-SOBS estimates the background model by detecting moving foreground objects using a self-organising neural background model. WS2006 initialises the background model in a two step process. First, for each pixel, the longest stable sequence of values that have similar intensities in the image sequence is considered as a candidate background. Second, using a RANSAC method the stable sequences which are likely to arise from the background are selected. The temporal mean of the selected subsequence provides the estimated background model. For RSL2011 the background estimation is carried out at the image block level using Markov random fields. A combined frequency is calculated on an image block and its corresponding neighbourhood. The background model is then estimated from the blocks that provide the smoothest frequency from the image sequence. Finally, the Photomontage method initialises the background model through a framework that selects the image blocks that require minimum cost in editing images in the sequence.

This chapter will benchmark the DMD method against the aforementioned list of methods.

### 7.2 Data, Methodology & Implementation

This section presents the SBI dataset, methodological framework as well as the implementation details.

**SBI dataset**

The publicly available SBI\(^1\) [161] dataset includes seven image/bootstrap sequences along with their corresponding ground-truths (GT) as shown in Figure 7.1. These

\(^1\)http://sbmi2015.na.icar.cnr.it
sequences are challenging in their own way and are discussed in section 7.3.

Table 7.1 provides the resolution information of the images in the sequences.

<table>
<thead>
<tr>
<th>Name</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hall&amp;Monitor</td>
<td>352x240</td>
</tr>
<tr>
<td>HighwayI</td>
<td>320x240</td>
</tr>
<tr>
<td>HighwayII</td>
<td>320x240</td>
</tr>
<tr>
<td>CaVignal</td>
<td>200x136</td>
</tr>
<tr>
<td>Foliage</td>
<td>200x144</td>
</tr>
<tr>
<td>People&amp;Foliage</td>
<td>320x240</td>
</tr>
<tr>
<td>Snellen</td>
<td>144x144</td>
</tr>
</tbody>
</table>

DMD for Colour Image Sequences

The methodological framework (Figure 7.2) consists of:

1. DMD applied to R, G and B channels of the colour image sequence.
2. Selection of the DMD mode representing the background model.
3. Normalising and combining three channels of the selected DMD mode to obtain the colour background image.
4. Transferring colour from the original images to the obtained background image.

In a dynamic sequence of \( N \) colour images \( P^{RGB} \), let \( \bar{x}^{RGB}_r \) be the \( r^{th} \) image whose size is \( m \times n \times 3 \) i.e., with a height of \( m \) pixels and width of \( n \) pixels across the three colour channels. This image \( \bar{x}_r \) is then separated for R, G and B channels. Later, these channels are vectorised to obtain a column vector of size \( mn \times 1 \) for each channel. These column vectors are then vertically concatenated to produce a
CHAPTER 7. COLOUR BACKGROUND INITIALISATION

Separate R, G & B

DMD

Vectorise & Vertically concatenate R, G, B channels

Apply DMD, Select background mode & Combine R, G & B

Estimate colour distribution

Calibrate DMD+CT

Figure 7.2: Flow chart showing the steps involved in the methodological framework. A colour image sequence consisting of \( N \) images in R, G, and B channels are vertically concatenated forming a data matrix \( X_{RGB} \) as shown in the Equation 7.1. DMD is applied to this matrix to obtain \( N-1 \) dynamic mode images. Frequencies are calculated on the dynamic eigenvalues. The mode with the frequency \( \approx 0 \) in R and B channels are vectorially concatenated forming a data matrix \( X_{RGB} \) as shown in the Equation 7.2. The selected mode is then separated as the background initialised image. The three channels are then normalised and combined to obtain colour background initialised image. Finally, the original color is transferred from the original images to the obtained background image.
vector of size \((3 \times mn) \times 1\) resulting in the construction of a data matrix \(P^{RGB}\) of size \((3 \times mn) \times N\) for \(N\) images in the sequence.

\[
P^{RGB} = [\bar{x}_1, \bar{x}_2, \bar{x}_3, \ldots, \bar{x}_N] = \begin{pmatrix}
R_1^1 & R_1^2 & \ldots & R_1^N \\
\vdots & \vdots & \ddots & \vdots \\
R_{mn}^1 & R_{mn}^2 & \ldots & R_{mn}^N \\
G_1^1 & G_1^2 & \ldots & G_1^N \\
\vdots & \vdots & \ddots & \vdots \\
G_{mn}^1 & G_{mn}^2 & \ldots & G_{mn}^N \\
B_1^1 & B_1^2 & \ldots & B_1^N \\
\vdots & \vdots & \ddots & \vdots \\
B_{mn}^1 & B_{mn}^2 & \ldots & B_{mn}^N
\end{pmatrix}
\] (7.1)

DMD is then applied on the data matrix \(P^{RGB}\) to obtain \(N - 1\) dynamic modes for \(N\) images in the sequence. DMD modes with frequencies \(\mu_j\) is given by \(\mu_j = \frac{\ln(\sigma_j)}{\delta t}\). Here, \(\delta t\) is the time difference between the images and considered to be 1 in this study. The real part of \(\mu_j\) regulates the growth or decay of the DMD modes, while the imaginary part of \(\mu_j\) drives oscillations in the DMD modes. The frequencies near the origin (zero-modes) are interpreted as background (low-rank) portions of the given image sequence, and the frequencies bounded away from the origin are their sparse counterparts. Specifically, the parts in image sequence that do not change in time, have an associated frequency at the origin of the complex plane with \(\|\mu_j\| \approx 0\), which corresponds to background model. Therefore, frequencies calculated on the dynamic eigenvalues and the mode with the frequency \(\approx 0\) is then selected as the background mode.

The methodology, as summarised from steps 1–9, in Algorithm 3, allows us to obtain the DMD mode that reveals the background. Although this DMD mode has a dimension of \((3 \times mn) \times 1\) which can be translated back to the original image space of size \(m \times n \times 3\), it is nonetheless not a properly normalised image. As a result, the DMD mode needs to be calibrated in order to produce a proper image. To this end, we normalise the DMD mode to the range of \([0,1]\) and then apply a colour transfer from the statistically calculated mode image using the method proposed by Reinhard et al. [168]. In our preliminary experiments, we notice that the background models obtained from the statistically calculated mode and WS2006 [165] have equal CQM values and qualitatively they look similar.

**Implementation**

The overall implementation is shown in Algorithm 3. Recall our methodological pipeline from Figure 7.2. For each of the colour image sequences in the dataset consisting of \(N\) images, R, G and B channels are separated and converted to column vectors. These vectors are then vertically concatenated to form a data matrix as shown in the Section 7.2 (Equation 7.1). DMD is then applied on the data matrix
CHAPTER 7. COLOUR BACKGROUND INITIALISATION

Algorithm 3 Dynamic Mode Decomposition

Input: Sequence of images in a video where R, G and B channels are vertically concatenated $P^{RGB} = [\bar{x}_1, \bar{x}_2, \bar{x}_3, \ldots, \bar{x}_N]$.

Output: Dynamic mode $D^{RGB}M$ corresponding to the frequency $||\mu_j|| \approx 0$.

1: $P_1 \leftarrow [\bar{x}_1, \bar{x}_2, \bar{x}_3, \ldots, \bar{x}_{N-1}]$
2: $P_2 \leftarrow [\bar{x}_2, \bar{x}_3, \bar{x}_4, \ldots, \bar{x}_N]$
3: $[U \Sigma V] \leftarrow \text{svd}(P_1, 'econ')$
4: $\tilde{H} \leftarrow U^*P_2V\Sigma^{-1}$
5: $[\omega \sigma] \leftarrow \text{eig}(\tilde{H})$
6: $||\mu_j|| \leftarrow \text{abs}(\log(\sigma_{jj}))$
7: $D_{Ms} \leftarrow \text{abs}(P_2V\Sigma^{-1}\omega)$
8: $\text{index} \leftarrow \text{find}(||\mu_j|| \approx 0)$
9: $D^{RGB}M \leftarrow (D_{Ms}(:, \text{index}))$
10: $D^{RGB}M_{DMD} \leftarrow \text{combineRGBChannels}(DM)$
11: $D^{RGB}M_{DMD\ CT} \leftarrow \text{colourtransfer}(D^{RGB}M_{DMD}, \text{mode}(P^{RGB}), 4)$

to obtain $N - 1$ dynamic modes for $N$ images in the sequence. Frequencies are calculated on the dynamic eigenvalues and the mode with the frequency $\approx 0$ is selected as the background mode. The three channels of that particular mode are first normalised and then combined to produce a colour background initialised image for a given input image sequence. In order to retain the colourmap from the original sequence, Reinhard et al.’s [168] method of colour transfer is used. We refer the background produced from DMD as ‘DMD’ and background produced after colour transfer as ‘DMD\_CT’.

Reinhard et al.’s Colour Transform

The source image contains the colour space that needs to be transferred onto the target image i.e., the ‘DMD’ image to obtain the colour transferred ‘DMD\_CT’ image. Both the source and the target images are then converted to $L^*a^*b^*$ space. Each of the three channels of the target image are normalised (zscore) by subtracting the mean and dividing it by their standard deviation. In the next step means are added and standard deviation is multiplied from the $L^*a^*b^*$ channels of the source image. Finally the target image is converted back into the RGB from the $L^*a^*b^*$ space.

Evaluation metric

There are several approaches to evaluate the performance of the initialised background model, as reported in [161]: (i) Average Grey-level Error (AGE), (ii) Error Pixels (EP), (iii) Clustered Error Pixels (CEP), (iv) Peak-Signal-to-Noise-Ratio (PSNR), (v) Multi-Scale Structural Similarity Index (MS-SSIM), and Colour Quality Measure (CQM). We have adopted CQM for evaluating the performance of our technique since we were more interested in measuring the quality of the ‘colour’
produced through DMD. This makes CQM a good choice for evaluating the performance of our technique. CQM is a recently proposed metric which is based on reversible luminance and chrominance (YUV) colour transformation and PSNR measure [169]. The units of CQM are denoted in decibels $dB$. The higher the CQM value, the better is the background model.

Let $GT$ be the ground-truth image and $CB$ be the DMD computed background. CQM is obtained in three stages. At first, the $GT$ and $CB$ images are transformed from RGB domain into YUV. PSNR is then calculated separately between the YUV components of both $GT$ and $CB$ images.

$$PSNR = 10 \cdot \log_{10} \left( \frac{(L - 1)^2}{MSE} \right),$$

where $L$ is the maximum number of grey levels and MSE is the Mean Squared Error between $GT$ and $CB$ images. Finally, CQM value is obtained using the equation below.

$$CQM = (PSNR_Y \times R_W) + \left( \frac{PSNR_U + PSNR_V}{2} \right) \times C_W. \quad (7.3)$$

Here $R_W$ and $C_W$ are the weights on human perception, which are given by 0.9449 and 0.0551 respectively.

The evaluation is performed through the Matlab codes provided via SBI dataset website².

### 7.3 Results & Discussions

DMD background mode considers only those parts in an image sequence that do not change in time as a background model. When this condition holds, DMD performs better than Median, WS2006 and Photomontage methods. For example, in the case of Snellen sequence, the foreground leaves continuously move through out the sequence and occupy most of the scene. Therefore, DMD is able to consider only those parts in the sequence that have not changed in time as its background model. The aforementioned methods here failed to remove the leaves from their background models as shown in Figure 7.3 (c,e & g). This is also confirmed by the quantitative results as DMD (36.17) and DMD_CT (36.85) out perform Median (36.07), WS2006 (24.99) and Photomontage (26.92) methods. Another example where the condition for DMD holds are the sequences of Highways, which generally reveal more than 50% of the background and most importantly the vehicles do not remain stationary anywhere in the scene throughout the sequence. Therefore, DMD on these sequences could obtain a perfect background model by eliminating the foreground moving vehicles. Although, the aforementioned methods have been successful in obtaining a perfect background model, quantitatively DMD outperforms these methods as shown in Table 7.2. Similarly, on the People&Foliage sequence DMD produced better results when compared to median, SC-SOBS and WS2006. On the Foliage,

²http://sbmi2015.na.icar.cnr.it/SBIdataset.html
CHAPTER 7. COLOUR BACKGROUND INITIALISATION

DMD both qualitatively and quantitatively outperformed Median by eliminating the greenish halos produced due to the moving leaves in the foreground.

On the contrary, when the condition is violated, block-based algorithms such as RS2011 may work better. For example, on the CaVignal sequence, DMD_CT and DMD qualitatively could not outperform the benchmark methods. This is due to the fact that, the man standing in the left of the sequence covers the scene for the first 60% of sequence before he starts walking and again stands on the right for the last 10% of the sequence. Therefore, holding to the condition, DMD has successfully eliminated the images when the person walks but not when he remains standing. The only effective solution to this problem, is by tracking the object across the video sequence. However, since DMD does not rely on such a prior knowledge, its failure in this case is not unexpected. Nevertheless, quantitatively, DMD_CT (44.87) outperforms rest of the methods but RS2011 (52.59). Similarly for Hall&Monitor sequence, a walking man, in the corridor occupied the same scene region for more than 65% of the image sequence. Therefore, DMD included the ghost artefact of the man in its background model. Although methods such as WS2006, Photomontage and RS2011 are able to obtain a perfect background model, interestingly they have not performed well quantitatively comparing with DMD.

Prior to comparing the overall average CQM value across the 7 sequences with the methods presented in [161], we first compare the results produced from DMD (Figure 7.3 (h)) and DMD_CT (Figure 7.3 (i)). After transferring the colourmap from the original image sequence to ‘DMD’ background initialised image, we see that the intensity values in ‘DMD_CT’ are adjusted to the original sequence’s colour format. For instance, on the HighwayI and HighwayII sequence, the background model produced from DMD has a different colour intensity compared to the ground-truth, when the colourmap is transferred, the background model produced by DMD_CT acquires the original sequence’s colour format. Quantitative results show the background models produced from DMD_CT have better CQM values compared to the background models produced from DMD (except on the Hall&Monitor sequence, but the difference is marginal), as shown in Table 7.2. DMD_CT (42.82) on an average across all the seven sequence performed better than Median (39.00) WS2006 (39.87) and Photomontage (42.82) methods and stands rank 3 as shown in Table 7.2.

7.4 Conclusions

In this chapter, we have introduced a technique using DMD to obtain the background model in the colour domain. The significance of our technique is demonstrated on a publicly available scene background initialisation (SBI) dataset. When DMD condition holds, i.e., “DMD background mode considers only those parts in an image sequence that do not change in time as a background model”, it performs better than Median, WS2006 and Photomontage. Contrarily, when the condition is violated, block-based algorithm such as RS2011 may work better. Since DMD_CT (with colour transfer) produces visibly higher quality image than DMD, we recom-
mend that DMD\textsubscript{CT} be used when condition is true. Our results in this chapter shows that DMD can successfully obtain a coloured background model.
CHAPTER 7. COLOUR BACKGROUND INITIATION

Figure 7.3: (a) Exemplar frames corresponding to \{295, 0, 0, 0, 261, 10, 0\} from seven different sequences of the SBI dataset. (b) Corresponding GT’s for the sequences. (c-g) Background models generated by benchmarked methods: (c) Median, (d) SC-SOBS, (e) WS2006, (f) RS2011, (g) Photomontage. (h) DMD and (i) DMD CT. Both DMD and DMD CT are the proposed methods in this thesis. The images \{a-g\} are taken from [161].
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CaVignal</td>
<td>42.27</td>
<td>52.59</td>
<td>44.87</td>
<td>32.06</td>
<td>40.17</td>
<td>33.14</td>
</tr>
<tr>
<td>Foliage</td>
<td>39.11</td>
<td>43.1</td>
<td>34.65</td>
<td>45.61</td>
<td>34.39</td>
<td>28.73</td>
</tr>
<tr>
<td>Hall&amp;Monitor</td>
<td>43.19</td>
<td>35.08</td>
<td>42.39</td>
<td>41.73</td>
<td>42.53</td>
<td>62.57</td>
</tr>
<tr>
<td>HighwayI</td>
<td>65.58</td>
<td>38</td>
<td>55.77</td>
<td>59.03</td>
<td>52.36</td>
<td>42.67</td>
</tr>
<tr>
<td>HighwayII</td>
<td>54.38</td>
<td>51.98</td>
<td>46.95</td>
<td>35.08</td>
<td>45.07</td>
<td>42.32</td>
</tr>
<tr>
<td>People&amp;Foliage</td>
<td>35.37</td>
<td>37.06</td>
<td>31.97</td>
<td>47.15</td>
<td>28.43</td>
<td>27.5</td>
</tr>
<tr>
<td>Snellen</td>
<td>44.75</td>
<td>50.26</td>
<td>36.85</td>
<td>26.92</td>
<td>36.17</td>
<td>36.07</td>
</tr>
<tr>
<td>Avg(CQM)</td>
<td>46.38</td>
<td>44.01</td>
<td>41.92</td>
<td>41.08</td>
<td>39.87</td>
<td>39</td>
</tr>
</tbody>
</table>

Table 7.2: Quantitative comparison of CQM values for DMD & DMD_CT with the benchmark methods.
Chapter 8

Conclusion and Future Work

This thesis focused on advancing the original formulation of the DMD methodology for solving novel problems in the fields of computer vision and signal processing. In the order of decreasing importance of technical contributions: a) extracting trends and forecasting (univariate) time series; b) movement correction and functional segmentation in medical imaging, with application to DCE-MRI of kidney sequences; c) detecting spoofed face image sequences in biometrics; (d) using DMD as texture descriptor for distinguishing genuine versus replicated finger vein images, and (e) obtaining a background model from an image video in the colour domain. These five aspects form the major contributions of this thesis.

DMD applications typically consider a data matrix as an input. For instance, DMD with application to image sequence analysis requires converting the input data into column vectors. However, DMD for univariate time series analysis has never been studied. Therefore, the first contribution of this thesis was to introduce the method of DMD through a Hankelisation approach, for decomposing a univariate time series into a number of interpretable elements with different subspaces, such as noise, trends and harmonics. In addition, time series forecasting was also shown using DMD. Furthermore, DMD when compared to singular spectrum analysis (SSA) was computationally efficient. The results at various noise levels on simulated data suggested that DMD is certainly a promising approach for any time series with a noise structure. Although these properties were not new in the DMD literature, our contributions were in making DMD work for univariate time-series through Hankelisation. Thus, this is the first work that showed DMD could be used for modelling, predicting and forecasting a univariate time-series.

The second contribution of this thesis introduced DMD for movement correction and segmentation of kidneys in DCE-MRI. A proper segmentation of kidney region in DCE-MRI is required to assess the normal functioning of the kidneys via quantification of the kidney function. However this is difficult due to unwanted complex organ motion caused by respiration. Therefore, preprocessing for motion correc-
tion is required prior to segmentation of the kidney region. However, for motion correction, due to the rapid change in contrast agent within the DCE-MR image sequence, commonly used intensity based image registration techniques are likely to fail. To address this issue, this thesis proposed a novel automated, registration-free movement correction approach based on windowed (W-DMD) and reconstruction (R-DMD) variants of DMD (WR-DMD). Moreover, the DMD modes that obtained during the intermediate stages of this process, when coupled with simple thresholding and connected component analysis provided an automatic segmentation of the kidney ROI. The proposed method was validated on ten different healthy volunteers’ kidney DCE-MRI datasets.

The third contribution of this thesis introduced DMD for detecting spoofed face image sequences in the field of biometrics. Since DMD has been used extensively in modelling fluid dynamics for extracting coherent structures, this thesis showed that DMD could also capture the complex dynamics of head movements, eye-blinking and lip-movements found in a typical video sequence containing face images. These unique features clearly distinguished videos produced by print and replay attacks from live (valid) videos containing an authentic face. However, DMD has not been used for classification tasks before. For this reason, this thesis proposed a DMD-based classification pipeline involving a texture-based descriptor called local binary patterns (LBPs), coupled with a discriminative classifier, support vector machines (SVMs) with a histogram intersection kernel. The effectiveness of the methodology was demonstrated using three publicly available databases: 1) print-attack, 2) replay-attack and 3)CASIA-FASD, attaining comparable results with the state of the art, following the respective published experimental protocols.

The fourth contribution of this thesis proposed a novel method to detect spoofing of static finger vein images using W-DMD. This is an atemporal variant of DMD for image sequences. The overall pipeline which consisted of W-DMD and SVM, proved to be efficient and convenient to use, given the absence of additional parameter tuning requirements. The effectiveness of our methodology was demonstrated using the publicly available FV-Spoofing-Attack database. Our test results showed that W-DMD could successfully detect printed finger vein images because they contained micro-level artefacts that not only differed in quality but also in light reflection properties when compared to valid/live finger vein images. The proposed method achieved better results when compared to established methods, when using SVM with a minimum intersection kernel.

Finally, as a fifth contribution, this thesis proposed a technique using the method of DMD to obtain a scene background model in the colour domain. The effectiveness of the proposed technique was demonstrated using a publicly available Scene Background Initialisation (SBI) dataset. The results both qualitatively and quantitatively showed that DMD can successfully obtain a coloured background model without the need for converting the colour images into grey scale ones.

The remainder of this chapter presents the main achievements in section 8.1 and discusses future directions in Section 8.2
8.1 Main Achievements

The following paragraphs summarise the main achievements of the work that was proposed in this thesis.

1. **Hankelised DMD to allow processing of univariate time series data.**
   DMD is fundamentally a multidimensional method, and has never been applied to a univariate time series. Therefore, Chapter 3 of this thesis for the first time proposed the method of DMD in a four-stage pipeline to decompose a univariate time series into a number of interpretable elements with different components, such as noise and underlying data trend. In addition, this four-stage pipeline utilising DMD also showed the forecasting of time series data.

2. **DMD based on a sliding window protocol (W-DMD) for analysing both images and image sequences (videos).**
   DMD is generally used for analysing image sequences. Therefore, in Chapters 4 and 6, a new variant (W-DMD) is proposed based on a sliding window protocol which can be used to analyse individual images. This enabled DMD to be used as a micro-texture feature descriptor in Chapter 6. In this chapter W-DMD was shown to extract local variations as low rank representation inside a finger vein image, capturing light reflections, illuminations and planar effects that are typically different for valid access and spoofed images, while the same variant for the video sequences in Chapter 4 has been used for removing periodic motion components.

3. **DMD for motion stabilisation.**
   The noise components from the image sequences are captured in the least significant DMD modes, which also contained motion components. Therefore, reconstruction (R-DMD variant) with the most significant modes robustly stabilised complex movements in a dynamic medical image sequence in Chapter 4.

4. **DMD for extracting coherent structures in biometrics and medical imaging.**
   DMD is well established for extracting coherent structures in the field of CFD. Chapters 4, 5 and 6 have also shown that DMD can indeed extract these coherent structures in the fields of biometrics and medical imaging. For example, in Chapters 5 and 6, i.e. in biometrics, these coherent structures include facial dynamics such as lip movements, eye blinks and other attack specific cues such as light reflection properties, planar and moire effects, that can be used to discriminate between a valid and a spoofed facial video or finger vein image. In Chapter 4, i.e. in medical imaging, the coherent structures have been used for highlighting the kidney, liver and spleen regions in DCE-MRI sequences.

5. **DMD to allow processing of coloured video sequences.**
   The DMD method introduced in computer vision essentially converts the coloured images
into grey scale for processing. Therefore, Chapter 7 presents a technique that enables DMD to obtain a background model in the colour domain. This chapter also demonstrated the effectiveness of the proposed technique on a publicly available Scene Background Initialisation (SBI) dataset. The results in this chapter both qualitatively and quantitatively showed that DMD can successfully obtain a coloured background model.

8.2 Future Work

In this section, we present areas covered in this thesis for various future research work.

DMD for EEG data analysis

EEG data are important in the diagnosis of many neurobiological events and often contaminated with noise artefacts. The contamination is mainly due to electrode movements, 50 or 60 Hz powerline interference and white noise. These artefacts influence the quality of EEG signals and pose significant challenges in analysing them. SSA is well established in the area of EEG signal processing for denoising as well as for extracting meaningful subcomponents. The subcomponents later have been used as feature vectors for e.g. in sleep stage classification, epilepsy detection and seizure predictions [30–32,39]. Since in this thesis we have shown that DMD is a faster algorithm than SSA and works analogously to SSA, our immediate future research will aim to apply the four staged DMD pipeline to EEG signal processing.

DMD for classification

The locus of our future research will also aim to improve DMD for classification in the following ways:

- **Concatenation of all the dynamic modes.** As the DMD algorithm generates \( N - 1 \) dynamic modes (where \( N \) is the number of frames in the video), it is of interest to concatenate some or all of these dynamic modes to represent a single image and produce corresponding LBP codes in order to further improve the classification performance, especially for very difficult 2D facial video spoofing attacks.

- **Application of LBP on each dynamic mode.** LBP codes could be produced on each of the \( N - 1 \) dynamic modes individually. It is possible to compute a mean LBP code from these modes in order to further improve the overall classifier performance.

- **Multiple kernel learning approach.** A linear combination of Mercer kernels is also a Mercer kernel. Consequently, a Histogram Intersection kernel
could be computed for each of the dynamic modes individually and a linear combination of these kernels could then be considered for classification.
References


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


REFERENCES


