

TECHNICAL UNIVERSITY OF CRETE, GREECE
SCHOOL OF ELECTRONIC AND COMPUTER ENGINEERING

Fluid Flow Motion Estimation using Video Data



Konstantinos Bacharidis

Thesis Committee

Professor Michalis Zervakis(ECE)

Professor Euripides G.M Petrakis (ECE)

Associate Professor Michail G. Lagoudakis (ECE)

Chania, March 2014



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Εκτίμηση Κίνησης Ροής Υγρού με χρήση δεδομένων Βίντεο



Κωνσταντίνος Μπαχαρίδης

Εξεταστική Επιτροπή

Καθηγητής Μιχάλης Ζερβάκης (ΗΜΜΥ)

Καθηγητής Ευριπίδης Γ.Μ Πετράκης (ΗΜΜΥ)

Αναπληρωτής Καθηγητής Μιχαήλ Γ. Λαγουδάκης (ΗΜΜΥ)

Χανιά, Μάρτιος 2014

Abstract

Computer Science nowadays is being associated with an increasing number of Scientific fields, with one of them being Hydrology. One of the subjects of Hydrology widely assisted by computers is fluid flow motion estimation. The fluid flow field contains information about the velocity of the flow as well as the flow characteristics, such as its vorticity. The estimation of the flow field until now required the use of on-field measurements with conventional equipment, such as accelerometers and allowed a narrow time window for measuring certain hydrological natural phenomena, such as flash flood streams. Computer Vision allows non-intrusive, precise and constant measurements of the flow by using video data. Motion estimation approaches, however, must take into account the dynamic nature of the fluid flow in order to provide accurate estimates of the motion field. This thesis addresses the problem of motion estimation along with the problem of motion pattern recognition for fluid flows. The proposed approach, which was introduced by Chang et.al. [28] for the topic of fluid flow motion estimation, uses a statistical estimation method combined with a differential framework to evaluate each possible displacement. The outcome is a smooth global motion field representing satisfyingly the major motions present in the flow.

The motion pattern recognition problem has long troubled Hydrologists since they can not have an actual visualization of all the flow patterns present in the fluid flow using existing conventional equipment. This leads to the loss of information related to the flow characteristics. For this topic, this thesis presents two approaches which allow the identification and characterization of flow patterns in the fluid motion field. The first approach is based on the work of Prof. Einar Heiberg [30] and Andreas Andersson [44], and allows the identification and classification of homogeneous plane wave and vortical flow patterns using vector pattern matching and was implemented for the blood flow characterizations in the human heart. The second is a novel correlation based approach which uses the Mean Squared Error as means of vector pattern similarity measure. Both approaches use the 2-Dimensional motion field of the fluid to identify and categorize homogeneous and vortical flow patterns present in the flow.

In both Motion estimation and Characterization the thesis develops novel improvements addressing either the efficiency or the computational complexity of estimation. The proposed methods are applied and evaluated on actual video scenes of river flow under different conditions.

Περίληψη

Στις μέρες μας η Επιστήμη των Υπολογιστών σχετίζεται ολοένα και με περισσότερα άλλα επιστημονικά πεδία με ένα από αυτά να είναι εκείνο της Υδρολογίας. Ένα από τα θέματα στα οποία η Υδρολογία έχει συσχετιστεί ευρέως με την επιστήμη των υπολογιστών, και συγκεκριμένα με τον κλάδο της Μηχανικής Όρασης, είναι η εκτίμησή της κίνησης της ροής υγρών. Το πεδίο ροής ενός υγρού περιέχει πληροφορίες σχετικά με την ταχύτητα της ροής, καθώς και τα χαρακτηριστικά της, όπως ο στροβιλισμός. Η εκτίμηση του πεδίου ροής μέχρι τώρα απαιτούσε επιτόπιες μετρήσεις με την χρήση συμβατικού εξοπλισμού, όπως επιτανχυσιόμετρα, γεγονός που επέτρεπε μόνο ένα στενό χρονικό παράθυρο για τις επιτόπου μετρήσεις συγκεκριμένων υδρολογικών φυσικών φαινομένων, όπως πλημμυρικά ρεύματα(χειμαρροί). Η Μηχανική Όραση επιτρέπει μη παρεμβατικές, ακριβείς και συνεχείς μετρήσεις της ροής με τη χρήση βίντεο δεδομένων. Ωστόσο, οι διάφορες προσεγγίσεις για την εκτίμηση της κίνησης πρέπει να λαμβάνουν υπόψη τη δυναμική φύση της ροής του υγρού, προκειμένου να παρέχουν ακριβείς εκτιμήσεις για το πεδίο ροής. Η παρούσα διπλωματική εργασία ασχολείται με το πρόβλημα της εκτίμησης της κίνησης καθώς και με το πρόβλημα της αναγνώρισης προτύπων κίνησης που συμπληρώνουν την κύρια ροή του υγρού. Η προτεινόμενη προσέγγιση, η οποία αρχικά αναπτύχθηκε από τον Chang et.al. [28] για το θέμα της εκτίμησης της κίνησης για την ροή ενός υγρού, χρησιμοποιεί μια μέθοδο στατιστικής εκτίμησης σε συνδυασμό με ένα διαφορικό πλαίσιο για την αξιολόγηση κάθε δυνατής μετατόπισης. Το αποτέλεσμα είναι ένα ενιαίο ομοιογενές πεδίο κίνησης το οποίο αναπαριστά ικανοποιητικά τις κυρίαρχες κινήσεις που υπάρχουν στη ροή.

Το πρόβλημα αναγνώρισης των προτύπων κίνησης έχει απασχολήσει σε μεγάλο βαθμό τους Υδρολόγους, δεδομένου ότι δεν μπορούν να έχουν μια πραγματική απεικόνιση όλων των μοντέλων κίνησης που υπάρχουν στη ροή του υγρού χρησιμοποιώντας τον υπάρχοντα συμβατικό εξοπλισμό και, ως εκ τούτου, χάνοντας πληροφορίες σχετικά με τα χαρακτηριστικά της ροής. Για το θέμα αυτό η παρούσα διπλωματική εργασία παρουσιάζει δύο προσεγγίσεις που επιτρέπουν τον προσδιορισμό και τον χαρακτηρισμό των προτύπων ροής που υπάρχουν στο πεδίο ροής του υγρού. Η πρώτη προσέγγιση βασίζεται στην δουλειά που παρουσίασαν οι Καθ. Einar Heiberg [30] και Andreas Andersson [44] και η οποία επιτρέπει την αναγνώριση και την ταξινόμηση ομοιογενών επίπεδων κυματοειδών και στροβιλοειδών κινήσεων, χρησιμοποιώντας μια μέθοδο ταύτισης διανυσματικών μοτίβων και η οποία εφαρμόστηκε σε περιπτώσεις ροής αίματος της ανθρώπινης καρδιάς. Η δεύτερη είναι μια νέα προσέγγιση συσχετισμού που χρησιμοποιεί το Μέσο Τετραγωνικό Σφάλμα(M.S.E.) ως μέτρο ομοιότητας μεταξύ των διάφορων μοτίβων. Και οι δύο προσεγγίσεις χρησιμοποιούν το δυσδιάστατο πεδίο κινήσεως του υγρού για τον εντοπισμό και την κατηγοριοποίηση των ομοιογενών και

στροβιλοειδών πρότυπων κίνησης που είναι παρόντα στη ροή.

Η παρούσα διπλωματική εργασία αναπτύσσει νέες βελτιώσεις στοχεύοντας είτε στην βελτίωση της απόδοσης είτε στην μείωση της υπολογιστικής πολυπλοκότητας τόσο στην εκτίμηση όσο και στον χαρακτηρισμό της ροής. Οι προτεινόμενες μέθοδοι εφαρμόζονται και αξιολογούνται πάνω σε πραγματικές σκληρές βίντεο ροών ποταμών κάτω από διαφορετικές συνθήκες.

Acknowledgements

First, I would like to express my sincere appreciation and thanks to my advisor Michalis Zervakis for trusting me with this amazing topic and for his guidance throughout the course of this thesis. Also, the Display Lab Supervisor Ntina Moirogiwrgou for all the help she provided me.

Next, I would like to thank professors Einar Heiberg and Ryota Tsubaki for their help and guidance throughout this thesis. Their contribution is of high importance. Also, many thanks to Andreas Andersson whose master thesis provided me with all the theoretical background needed for the implementation of one of the approaches presented.

Furthermore, I would like to thank my friends from Chania and Alexandria, Eleni(Buklou), Eleni(Paoki), Katerina(Chan...), Nikos(Kargio...), Marina(Xazoulita), Nikolas ,Christos (Fazer),Vasilis(Pou...io), Katerina(Xanthia) and of course my girlfriend Dimitra. The memories of all the amazing adventures we have encountered will accompany me for ever. Thank you guys!

Last, but not least, I would like to dedicate this thesis to my parents and my sister for their understanding and support throughout my life.

Contents

1	Introduction	1
1.1	Thesis Contribution	2
1.2	Thesis Outline	3
2	Background	5
2.1	Motion Field and Motion Estimation	5
2.2	Optical Flow	5
2.2.1	Problem Restrictions	6
2.2.2	Brightness Constraint	7
2.2.3	The Aperture Problem	7
2.3	Optical Flow Computation Techniques	9
2.3.1	Differential Methods	9
2.3.2	Correlation based Methods	12
2.3.3	Feature based Methods	15
2.3.4	Frequency based Methods	15
2.3.5	Hierarchical Based Methods	16
2.3.6	Method Comparison	17
2.4	Pattern Recognition	19
2.4.1	Pattern Recognition as a Classification Task	19
2.4.2	Supervised Learning	20
2.4.3	Unsupervised Learning	20
2.4.4	Motion Pattern Recognition	21
3	Problem Statement and Related Work	23
3.1	Motion Estimation in Fluid Flows	23
3.2	Related Work on Motion Estimation in Fluid Flows	24

CONTENTS

3.2.1	Methods based on Properties of Fluid Mechanics	25
3.2.2	Methods based on the Physical Properties of Waves	27
3.2.3	Particle based Methods	30
3.2.4	Methods Based on Statistical Estimation	35
3.3	Pattern Recognition in Fluid Flows	36
3.4	Related Work on Pattern Recognition in Fluid Flows	37
3.4.1	Methods based on Image Processing	37
3.4.2	Methods based on Vector Field Topology	40
3.4.3	Methods based on Physical Characteristics	44
4	Proposed Approach	49
4.1	2-D Statistical Estimation of Fluid Flows	49
4.1.1	The Statistical Estimation Basis	52
4.1.2	Estimation of the Distribution function	52
4.1.3	Coefficient Estimation Process	54
4.1.4	Coefficient Pruning	56
4.1.5	Individual Motion Vector	58
4.1.6	Global Optimization	58
4.1.7	Conjugate Gradient Method	60
4.1.8	Tracer Identification and Motion Visualization	63
4.2	2-D Flow Characterization Using Vector Pattern Matching	64
4.2.1	Normalization of the vector field	65
4.2.2	Filter Creation	66
4.2.3	Filter Convolution	69
4.2.4	Outer Product Tensor	72
4.2.5	Eigenvector and Eigenvalue Computation	73
4.3	2-D Flow Characterization based on Mean Squared Error	75
4.3.1	Reference Pattern Creation	76
4.3.2	Normalization of the Reference Vector field	77
4.3.3	Mean Squared Error Calculation	78
4.3.4	Pattern Classification	78
5	Results	81
5.1	Algorithm Accuracy	82
5.1.1	2-D Statistical Estimation of Fluid Flow Fields	82

5.1.2	2-D Fluid Flow Characterization Using Vector Pattern Matching .	92
5.1.3	2-D Fluid Flow Characterization Using M.S.E.	98
5.2	Algorithm Performance	101
5.2.1	2-D Statistical Estimation of Fluid Flows	101
5.2.2	2-D Fluid Flow Characterization Using Vector Pattern Matching .	105
5.2.3	2-D Fluid Flow Characterization Using M.S.E.	106
5.3	Vector Pattern Matching Vs Mean Squared Error	106
5.4	Fluid Flow Viewer	108
6	Conclusion	115
6.1	Conclusion	115
6.2	Future Work	116
6.2.1	2-D Statistical Estimation on Fluid Flows	116
6.2.2	2-D Flow Characterization using Vector Pattern Matching	119
6.2.3	2-D Flow Characterization using Mean Squared Error	121
6.2.4	Fluid Flow Viewer(F.F.V.)	122
	References	127

CONTENTS

List of Figures

2.1	Uniform Sphere under a light source. Figure taken from B. Jahne [1].	6
2.2	The optical flow constraint line and the normal velocity component. Figure taken from Beauchemin [2].	8
2.3	In the aperture position 2 the motion can be fully measured as there is sufficient local structure whereas in aperture positions 1,3 only normal motions can be estimated. Figure taken from Beauchemin [2]	8
2.4	Search Pattern of the Three Step Search algorithm. The figure was taken from http://www.ece.cmu.edu/~ee899/project/deepak_mid.htm	14
2.5	Hierarchical Model for optical flow computation. This pyramid scheme was taken from Beauchemin [2].	17
2.6	Supervised learning work-flow diagram. At the first stage a model is defined and trained using labelled data with known responses. After the model is trained it can be used to predict response values using new unlabelled data.	20
2.7	Unsupervised Learning model where the calibration of the classifier depends on the resulted classification data.	21
3.1	Example of a simulated irregular wave profile: Two-dimensional waves using eq.(3.15) with M=150. Figure was taken from Saikano [18].	28
3.2	The two-step optimization algorithm used to minimize eq.(3.19) and compute the optical flow components and the wave parameters. Figure was taken from Saikano [18].	29
3.3	The PIV technique implementation procedure in small scale flows such as the measurement of fluid flows using laser beams as tracer pointers. Figure taken from http://www.dantecdynamics.com/measurement-principles-of-piv	31

LIST OF FIGURES

3.4	Bradley’s System Layout and Study site at Clear Creek near Oxford, Iowa. The selected area is video taped by a camera laid on top of a bridge. The video data are used in the motion estimation step. The actual river flow velocity is computed using the derived velocity field and some know ground points. Figure was taken from Bradley et. al. [20].	32
3.5	A comparison of the spatial resolution between PIV and PTV. Figure was taken from Stitou and Riethmuller [25].	33
3.6	Laboratory flume flow visualized by tracer; (a) oblique-angled image and the location of searching line (b) an example of STI for a thick searching line indicated in (a). Figure taken from Fujita et al. [27].	35
3.7	Results of flow characterization of blood flow in the human heart and the aorta as presented by Heiberg et.al. [30]. A complete vortex ring can be seen below the mitral valve (the valve between the left atrium and the left ventricle) in a healthy normal heart. The vortex core is shown as a white isosurface and streamlines are released from the isosurface. The time in the heart cycle is the beginning of the atrial contraction, when blood flows from the left atrium to the left ventricle. Streamlines were also released in the valve orifice.	39
3.8	Patterns used by Heiberg et. al. [30]	40
3.9	Parts of a gas furnace chamber as present by the method of . Pattern matching of the vector field of the chamber and $3 \times 3 \times 3$ (red), a $5 \times 5 \times 5$ (yellow) and a $8 \times 8 \times 8$ (green) rotation mask was computed. The similarities are visualized using a marching tetrahedra with an isovalue of 0.5. On the right, the main inflow of the gas is shown. At the bottom, some streamlines are drawn additionally. Figure was taken from Ebling and Scheuermann [31].	41
3.10	Vector field topology: critical points classified by the eigenvalues of the Jacobian. R1 and R2 correspond to the real parts of the eigenvalues, whereas I1 and I2 correspond to the imaginary parts. Figure taken from Post, Vrolijk et. al. [35].	42
3.11	Vector field topology: a topological skeleton of a flow around a cylinder, where at: attachment, ce: center, de: detachment and sp: saddle point. Figure taken from Post, Vrolijk et. al. [35]	42

3.12	Invariant moment values for proto-typical flow features. Figure taken from Schlemmer et. al. [33].	44
3.13	Streamlines with centers of curvature. This figure was taken from Sadarjoen et al. [38].	45
3.14	Result of this method for an image taken in the Pacific Ocean with selective streamlines, (blue)indicates the detected vortices, (yellow)other motions. This figure was taken from Sadarjoen et al. [38]	46
4.1	Flow diagram presenting the the two basic steps/stages of the presented approach. The first stage results in a motion field with discrete displacements whereas the second stage in a smooth global motion field, which if implemented for every pixel will result in a dense optical flow field. . . .	50
4.2	The reference filter patterns used for identifying vortical and turbulent flow, (left) filter with orientation vector n_1 , (center) with orientation vector n_2 and (right) with orientation vector n_3	68
4.3	Example of a filter pattern describing a homogenous flow pattern with orientation vector n_{homog}	69
4.4	Field movement which shows no vortex form. The result of the eigenvalues correctly classifies the given flow pattern as non-vortical motion.	74
4.5	Field movement which shows vortex form. The result of the eigenvalues correctly classifies the given flow pattern as vortical motion pattern. . . .	74
4.6	The eight sub-directions of movement for homogeneous flow.	76
4.7	The first six of the eight possible direction patterns of homogeneous flow with normalization. The unit orientation vectors used(from up left to right order) are: (1) n_1 , (2) n_2 , (3) n_3 , (4) n_4 , (5) n_5 and (6) n_6	78
5.1	Small Stream Flow, (left)Input frame and (right)Resulted motion field, N_s size: 9×9 and D_s size: 5×5	83
5.2	River Flowing covered with Ice, (left)Input frame and (right)Resulted motion field, N_s size: 9×9 and D_s size: 5×5	83
5.3	River Flowing covered with Ice, we can observe that as the size of the destination neighbourhood rises so does the estimation error resulting to more erroneous motion vectors.	84

LIST OF FIGURES

5.4	Small Stream flow, (left and up)Three Step Search algorithm,(right and up)Presented algorithm, (left and down) New Three Step Search algorithm and (right and down) Four Step Search algorithm, N_s size: 9×9 and D_s size: 5×5 , Similarity rates: 63.37% with TSS, 64.15% with NTSS and 63.14% with FSS.	85
5.5	River Flowing covered with Ice, (left and up)Three Step Search algorithm,(right and up)Presented algorithm, (left and down) New Three Step Search algorithm and (right and down) Four Step Search algorithm, N_s size: 12×12 and D_s size: 7×7 , Similarity rates: 85.74% with TSS, 85.01% with NTSS and 82.59% with FSS.	86
5.6	River Flowing covered with Ice, (circle)starting point and (x-mark)destination point (left)User selects the correct direction of motion, downward motion and (right)User selects the wrong direction of motion, upward motion.	88
5.7	Acheloos River Flowing, Tracers used: surface foam, N_s size: 12×12 , D_s size: 7×7	89
5.8	Artificial River Flowing, Tracers used: White balls, N_s size: 12×12 , D_s size: 7×7	89
5.9	Small Stream flow, (left)Input frame, (center) Resulted motion field from the First Stage, (right)Resulted motion field from the Second Stage, N_s size: 12×12 , D_s size: 7×7	90
5.10	Small Stream flow, (left and up) Horn- Schunck's Method, (right and up) Lucas- Kanade's Method, (left and down)Our approach with basis function a 2-D Gaussian function and N_s size: 12×12 , D_s size: 7×7	91
5.11	River Flowing covered in ice, (left and up) Horn- Schunck's Method, (right and up) Lucas- Kanade's Method, (left and down)Our approach with basis function a 2-D Gaussian function and N_s size: 12×12 , D_s size: 7×7	92
5.12	Small Stream Flow, (left) initial image, (right) Resulted motion field from Lucas- Kanade's algorithm, with red marked vectors corresponding to detected vortical flow patterns.	93
5.13	Small Stream Flow, (red)the detected vortical flow pattern, (blue) Other flow patterns.	93

5.14	Small Stream Flow, (red mark) clockwise vortexes, (green mark) counter-clockwise vortexes, (pink mark) uncategorised vortical flow patterns and (blue) other flow types, Search window size: 10×10 and Threshold values: $T_{left} = 6, T_{right} = 8$	94
5.15	Small Stream Flow, (red mark) the clockwise vortex detected.	95
5.16	Small Stream Flow, (green mark) the counter-clockwise vortex detected.	95
5.17	An example of the importance of correct thresholding, we can observe that despite the fact that a vortex has been identified as a vortex the threshold value did not allow the vortex (black mark) to be further classified (clockwise or counter clockwise).	96
5.18	The homogeneous flow structures, red mark, identified in the Small Stream motion field, using the condition that the eigenvalues λ_1, λ_2 being zero.	96
5.19	Enhanced display of the previous figure. The structures are correctly identified as homogeneous flow patterns. Search window size and filter pattern size: 10×10	97
5.20	Classification output of our approach for the case of a River flowing covered in Ice, displaying the homogeneous flow structures. Initial image displayed in figure 5.2. Search window size and filter pattern size: 10×10	98
5.21	Enhanced display of figure 5.20. The structures are correctly identified as homogeneous flow patterns. Search window size and filter pattern size: 10×10	99
5.22	Classification output of our approach for the case of a River flowing covered in Ice, displaying the vortical flow structures detected. Initial image displayed in figure 5.2. Search window size and filter pattern size: 10×10	100
5.23	Enhanced display of figure 5.22. The structures are correctly identified as vortical flow patterns. Search window size and filter pattern size: 10×10	101
5.24	River Flowing covered in Ice, initial image is shown in figure 5.2. Search window size: 8×8 . All the possible motions are displayed.	102
5.25	Enhanced region of figure 5.24. Search window size: 8×8 . All the possible motions are displayed. The majority of the structures are classified correctly.	102
5.26	Small Stream Flow, initial image is shown in figure 5.12. Search window size: 8×8 . All the possible motions are displayed.	103
5.27	Enhanced region of figure 5.26. Search window size: 8×8 . All the possible motions are displayed. The majority of the structures are classified correctly.	103

LIST OF FIGURES

5.28	Enhanced region of figure 5.26. Search window size: 8×8 . All the possible motions are displayed. The majority of the structures are classified correctly.	104
5.29	Resulted classification only for vortical flows(clockwise vortex(red) and counter-clockwise vortex(green)), for the case of the small Stream flow.	104
5.30	Enhanced region of figure 5.29. Search window size: 8×8 . Clockwise vortex(red) and counter-clockwise vortex(green).	105
5.31	Initialization window of Fluid Flow Viewer.	107
5.32	The main process panel of F.F.V., in this figure the selected method is Lucas- Kanade.	107
5.33	Lucas- Kanade result with all the image pixels present for the case of the small stream flow image. The whole output figure can be displayed when the user pushes the figure display button. This allows a more detailed examination of motion field.	108
5.34	Lucas- Kanade result with all the image pixels present for the case of the small stream flow image. Only the user defined region's motion field is displayed. The green mark denotes the desired region and the blue mark denotes the region's background(undesired regions).	109
5.35	Lucas- Kanade result with a Macro-block pixel display for the case of the small stream flow image. The whole output figure can be displayed when the user pushes the figure display button. This allows a more detailed examination of motion field.	110
5.36	2-D Flow Characterization using Vector Patter Matching result. The whole output figure can be displayed when the user pushes the figure display button. The user defined filter size is 10×10 and also all structures are to be classified to all the motion types available(homogeneous(green mark) and vortical(red mark)) are selected by the user.	111
5.37	2-D Flow Characterization using Vector Patter Matching result for the user defined region of examination. The same display parameters are present as in the previous figure.	111
5.38	2-D Flow Characterization using M.S.E. result. The whole output figure can be displayed when the user pushes the figure display button. The user defined filter size is 10×10 and also all structures are to be classified to all the motion types available(homogeneous with its subclasses and vortical with its subclasses(clockwise, counter-clockwise) are selected by the user.	112

5.39 2-D Flow Characterization using M.S.E. result for the user defined region of examination. The same display parameters are present as in the previous figure. 112

5.40 Histogram displaying the motion distribution analogy in the motion field. The current histogram displays the motion types analogy for the case of the small Stream flow. The motion classification was made with the 2-D Flow Characterization using M.S.E method. Similar motion distribution histogram can also be constructed with the 2-D Flow Characterization using Vector Patter Matching. 113

5.41 Average velocity components values. The current average velocity magnitude was calculated using the motion field information provide by Lucas-Kanade’s result. Average velocity magnitudes can be computed also using the two other available methods(2-D Statistical Estimation and Horn-Schunck’s methods). 113

LIST OF FIGURES

List of Algorithms

1	Pruning Coefficient Algorithm	57
2	The standard conjugate gradient algorithm for nonconvex problems	62
3	The Flow Characterization algorithm using Vector Pattern Matching . . .	75
4	The Flow Characterization algorithm using the Mean Squared Error. . .	79

LIST OF ALGORITHMS

Chapter 1

Introduction

Motion estimation is one of the key features in Computer Science in the fields of Image Processing and Computer Vision. The extracted motion field is a source of information used in numerous applications in computer science and robotics, such as video compression, motion compensation, visual odometry e.t.c., providing information about the 3-D velocity field or even the scene formation allowing object or surface detection within the viewed scene.

The motion information can be used by other scientific fields to export further information. One of its applications is in Hydrology. The derived motion field from fluid flow scenes such as rivers, oceans, contains information about the velocity field of the flow as well as the flow types being present.

Motion field extraction in fluid flows can be especially difficult due to the dynamic nature of fluid motion especially in natural phenomena such as rivers, streams and ocean waves. Dynamic motion differs from the motion of rigid bodies as it is subjected to forces, such as wind, which alter the motion. In order to detect this irregular and multi-directional motion a number of optical flow models based on fluid dynamics have been developed. Other techniques providing valuable results are based on statical estimation of the intensity distributions along the image plane. Even block matching techniques yield satisfying results of the perceived motion and are widely used in river flow extraction.

An essential aspect in Hydrology, despite the flow field motion, is the detection and characterization of flow types. Flow types, such as vortices are of particular importance. Their velocity fields provide additional information concerning the structural fatigue of the surface with which the liquid interacts. Concerning this topic basic image processing

1. INTRODUCTION

pattern recognition techniques, such as moment invariants, show remarkable results and are being widely used for this purpose.

Another difficulty, despite the dynamic nature of the fluid flows mentioned earlier, is fluid flow data acquisition. Unfortunately, there are not many available image data to be used, especially for natural phenomena, and even the ones used for the research purposes are not widely accessible. Due to this fact, we had to use as much data as we could harvest from field experiments as well as data found on the web. These factors made the approach comparison essentially impossible, and thus, we had to rely on result comparison between widely used and available methods we could lay our hands on.

1.1 Thesis Contribution

This thesis describes new approaches developed for the purpose of extraction and characterization of the motion field for fluid flows, such as rivers, streams, waves e.t.c. We present a new approach of a pre-existing motion estimation algorithm for fluid flows based on the statistical estimation methodology, introduced by Chang et.al. [28] for 3-Dimensional fluid flows, whose theoretical background is enhanced and altered in order to be used for 2-Dimensional image data, which is presented in Chapter 4, section 4.1. The presented algorithm shows satisfying results in identifying the main motions in the fluid flow. We have also, added another aspect to this approach, the tracer identification and visualization step, which allows the identification of tracers¹ and their motion field computation which can be associated with the motion field of the fluid, reducing the computational cost.

Furthermore, we incorporate the extracted motion field into two motion pattern characterization algorithms in order to identify and characterize the existing flow patterns. The first algorithm is a variation of a pre-existing algorithm initially developed for 3-Dimensional blood flow patterns in human heart, based on vector pattern matching. We alter the theoretical background so that it can be used on 2-Dimensional fluid flow fields, Chapter 4, section 4.2. The second algorithm presented is a simple vector matching algorithm which uses the Mean Squared Error(M.S.E.), as the benchmark value for the vector comparison, Chapter 4, section 4.3.

¹The definition of tracers is presented in Chapter 3, in the Particle Tracing method class.

Finally, we combine all the aforementioned methods as well as some pre-existing known motion extraction methods, such as Lucas Kanade, Horn-Schunck, into an automated graphical tool, Fluid Flow Viewer (F.F.V.), suitable for extracting, identifying and visualizing fluid flows.

1.2 Thesis Outline

In Chapter 2 we provide the background information needed for this thesis. We present the motion extraction problem as well as the major motion estimation and optical flow techniques developed for addressing this problem. Moreover, we present the pattern recognition methods used in Computer Science and Computer Vision. In Chapter 3 we state the motion extraction and characterization problem in fluid flows and we refer to different approaches that have been developed so far. In Chapter 4 we describe our approach in the topics of motion field extraction and flow pattern characterization. In Chapter 5 we evaluate the accuracy of the presented algorithms compared with pre-existing methods and we also present a Graphical User Interface(GUI), developed for extracting, identifying and visualizing fluid flow motion, named Fluid Flow Viewer(F.F.V.). Finally, Chapter 6 acts as an epilogue for this thesis, presenting our conclusions along with future improvements. Reference is made to applications for which the presented GUI is intended to. Also, some ideas for upgrading this graphical tool in the future are being mentioned.

1. INTRODUCTION

Chapter 2

Background

2.1 Motion Field and Motion Estimation

The Motion Field is the perspective projection of the velocity field of a viewer in a 3-Dimensional scene onto the 2-Dimensional image plane. Motion Estimation is the process of determining the motion field of a scene using adjacent frames in a video sequence. This is an ill-posed problem as the data available are only spatial and temporal variations in the image brightness pattern, from which we can only recover an estimate of the motion field, known as optical flow.

2.2 Optical Flow

By the term of Optical flow, we refer to the apparent velocities estimated by the variations in the image brightness. The optical flow in the image plane and the motion field of a moving object in a 3-Dimensional scene does not share an obvious relation. This can be seen, for example, in the case of a uniform sphere, shown in the following figure(Figure 2.1). Under stationary light source, when the sphere is rotated the optical flow is zero whereas the motion field follows the rotational motion. The reason the optical flow is zero is due to the fact that the shading of the sphere, caused by the light source illumination, is also stationary which combined with the uniformity of the sphere's surface, leads to no variations in the image brightness. Furthermore, in the case of a moving light source and a stationary sphere, the optical flow seems to be zero whereas the motion

2. BACKGROUND

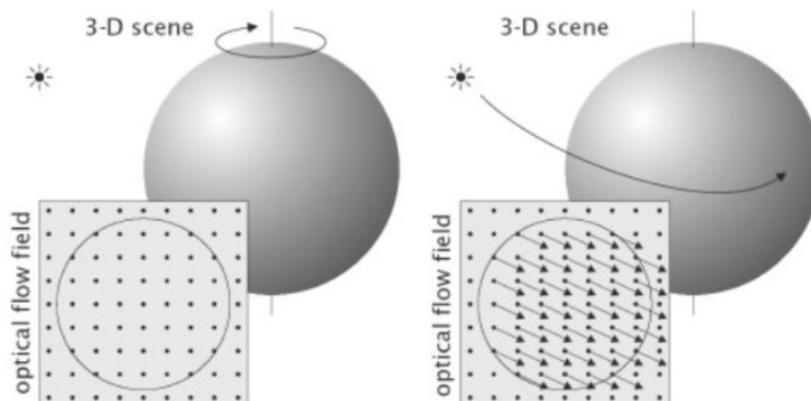


Figure 2.1: Uniform Sphere under a light source. Figure taken from B. Jahne [1].

field is zero.

As we can see, from the previous example, estimating the actual motion field in the optical flow is difficult when we are facing cases where shading is present. To avoid the brightness variations caused by shading effects we need to introduce some restrictions to address the motion estimation problem.

2.2.1 Problem Restrictions

The first assumptions being made to address the shading effects are, that the surface is flat and that the incident illumination is uniform across the surface. Using these assumptions we can now express the brightness at a point in the image as proportional to the reflectance of the surface at the corresponding point. Another assumption we make, is that the reflectance varies smoothly and has no spatial discontinuities which assures us that the image brightness is differentiable and also events such as occlusion or transparencies which cause inconsistencies in the optical flow are not present in the image sequences used during the implementation of the proposed algorithms. Given these assumptions made, the only thing left to do is to derive an equation relating the variation in the image brightness to the motion of the pattern.

2.2.2 Brightness Constraint

The most basic assumption being made, in order to measure image motion, is that the intensity structures of local time-varying image regions are approximately constant under motion for at least a short duration as presented by Horn and Schunck [3]. Let $I(x,y,t)$ denote the image brightness at a point (x,y) in the image plane at time t . According to the previous assumption the brightness at this point can be expressed as:

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \quad (2.1)$$

where Δx and Δy is the displacement of the local image region at (x,y,t) after time Δt . Expanding the left side of this equation using Taylor series yields:

$$I(x, y, t) = I(x, y, t) + \frac{\delta I}{\delta x} \Delta x + \frac{\delta I}{\delta y} \Delta y + \frac{\delta I}{\delta t} \Delta t + O^2 \quad (2.2)$$

where $\frac{\delta I}{\delta x}, \frac{\delta I}{\delta y}, \frac{\delta I}{\delta t}$ are the partial derivatives of the image function in the x , y , and t dimensions, and O^2 the second and higher order terms which can be ignored. If we combine the equations (2.1) and (2.2) the optical flow constraint equation is formed:

$$\nabla I \cdot \mathbf{v} + I_t = 0 \quad (2.3)$$

where $\nabla I = (I_x, I_y)$ is the spatial intensity gradient and $\mathbf{v} = (u, v)$ is the optical flow vector. Due to the fact that the time displacement is between two frames, $\Delta t = 1$ and thus, it can be removed.

The optical flow constraint equation defines a single local constraint on image motion (Figure 2.2). In the figure the normal velocity \mathbf{v}_\perp is defined as the vector perpendicular to the constraint line and is the velocity with the smallest magnitude which satisfies the optical flow constraint equation.

2.2.3 The Aperture Problem

The constraint provided by the optical flow constraint line is not sufficient to compute both components of \mathbf{v} as the optical flow constraint equation allows as to estimate only \mathbf{v}_\perp , which is the motion component in the direction of the local gradient of the image intensity function. This is known as the aperture problem. At positions where there is

2. BACKGROUND

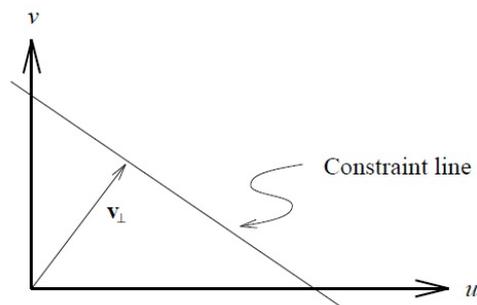


Figure 2.2: The optical flow constraint line and the normal velocity component. Figure taken from Beauchemin [2].

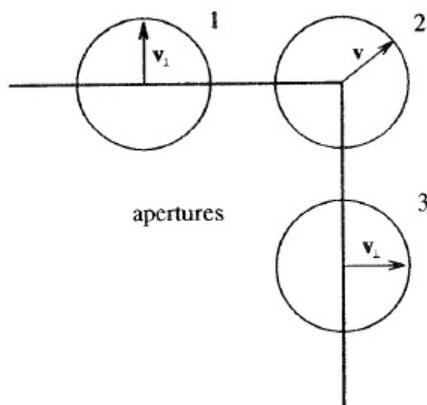


Figure 2.3: In the aperture position 2 the motion can be fully measured as there is sufficient local structure whereas in aperture positions 1,3 only normal motions can be estimated. Figure taken from Beauchemin [2]

a gradient in both directions, like a corner, there is sufficient intensity structure, which allows the computation of the gradient in both directions. This can be seen in the following figure(Figure 2.3) which illustrates how the component perpendicular to the gradient is missing while in the case of corner-like structures there is sufficient information to compute the optical flow vector, due to the high curvature the display.

2.3 Optical Flow Computation Techniques

Various methods have been proposed for the computation of optical flow. Each of them using a different approach to address the problem of calculating optical flow. In the following sections we will present the major classes in which these approaches can be classified, presenting their basic idea and some of their most known methods. Classes such as differential methods, feature based methods and others will be presented.

2.3.1 Differential Methods

The Differential methods class uses the spatio-temporal derivatives of image intensities to compute the image velocity field by assuming the image domain to be continuous in space and time or in other words differentiable. Differential methods can be further categorized into Global and Local methods using 1_{st} and 2_{nd}-order derivatives to compute the optical flow.

- **Global Methods**

Global methods use (2.3) combined with a regularization term such as a smoothness constraint to form a functional to be minimized over the image domain. Horn and Schunck [3] was first to use a regularization by requiring a slowly varying optical flow field. Ever since there have been many approaches made to this topic, such as Bergen [4] and others.

Horn and Schunck: Horn and Schunck's [3] method is one of the classic method used for optical flow computation. The method uses the optical flow constraint equation in conjunction with a smoothness constraint allowing the computation of the optical flow to be done on the whole image, yielding a global flow field, which would vary smoothly. To succeed this, a constraint is added in order to minimize the square of the magnitude of the optical flow vector which combined with optical flow constraint equation results in the estimation error to be minimized:

$$E_{total} = E_{const} + w^2 E_{smooth} \quad (2.4)$$

where E_{const} is the optical flow constraint equation deviation and E_{smooth} is the smoothness constraint multiplied with a weighting factor w .

2. BACKGROUND

The constraint equation and the smoothness constraint errors are expressed as follows:

$$E_{const} = (\nabla I \cdot \mathbf{v} + I_t) \quad (2.5)$$

$$E_{smooth} = \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2 \quad (2.6)$$

where $\frac{\partial}{\partial x} \frac{\partial}{\partial y}$ are spatial derivatives of the flow components along the x and y axis.

This estimation error can denote two characteristics for the optical flow field. The first is the amount of deviation of the flow vector from the spatial and temporal gradient and the second is the absence of smoothness in the flow vector field.

Horn and Schunk [3] by using a Laplacian estimation for the optical flow gradients, end up with a system of two equations for every pixel point:

$$(w^2 + I_x^2 + I_y^2)(u - \bar{u}) = -I_x [I_x \bar{u} + I_y \bar{v} + I_t] \quad (2.7)$$

$$(w^2 + I_x^2 + I_y^2)(v - \bar{v}) = -I_y [I_x \bar{u} + I_y \bar{v} + I_t] \quad (2.8)$$

in which \bar{u}, \bar{v} : averages of the components of the flow vector which are calculated from a spatial neighbourhood around the pixel of interest, I_x, I_y are the spatial gradients and I_t is the temporal gradient, w is the weighting parameter.

This system of equations can be computed using an iterative method in which each flow vector component is calculated using the previous iteration's information:

$$u^{n+1} = \bar{u}^n - \lambda \cdot I_x \quad (2.9)$$

$$v^{n+1} = \bar{v}^n - \lambda \cdot I_y \quad (2.10)$$

where (u^{n+1}, v^{n+1}) is the new optical flow velocity components and (\bar{u}^n, \bar{v}^n) are the average velocity components computed from the spatial neighbourhood at the previous iteration and λ being expressed as follows:

$$\lambda = \frac{[I_x \bar{u}^n + I_y \bar{v}^n + I_t]}{w^2 + I_x^2 + I_y^2}$$

2.3 Optical Flow Computation Techniques

The use of Horn- Schunk's method, results in the motion vectors being computed in accordance with the neighbouring vectors which will result to homogeneous regions in the image, and thus, producing a uniform flow field.

Ever since, various approaches have been introduced based on Horn and Schunk's method which aim to further improve the estimation of the motion field.

- **Local Methods**

Local methods use normal velocity information in local neighbourhoods to perform a least squares minimization to find the best fit for \mathbf{v} . One of the most popular local method for optical flow computation is the one of Lucas and Kanade's [5].

Lucas and Kanade: In Lucas and Kanade's method the optical flow vector is found by assuming that a spatial neighbourhood surrounding the pixel have similar vectors. The flow at pixel(x,y) is approximated by using a weighted least squares method:

$$E_{\mathbf{v}} = \sum_{\vec{x} \in \Omega} W^2(\vec{x}, t) [\nabla I(\vec{x}, t) \cdot \mathbf{v} + I_t(\vec{x}, t)] \quad (2.11)$$

where $W(\vec{x}, t)$ is the weight associated to each pixel $\vec{x} = (x, y)$ of the neighbourhood Ω , ∇I are the spatial gradient and I_t is the temporal gradient.

The use of the weights is to decrease the importance of distant neighbours which will produce higher error. Smaller weights will be associated to farther pixels reducing the influence of the spatial and temporal gradients of the neighbouring pixels to the computation of optical flow vector. Thus, we end up with a least squares problem which solution will produce the optical flow vector \mathbf{v} :

$$A^T W^2 A \mathbf{v} = A^T W^2 \mathbf{b} \quad (2.12)$$

where A is a vector of the spatial gradients of the neighbourhood pixels, W is the weight array for each neighbouring pixel and \mathbf{b} is a vector of the temporal gradients of these neighbouring pixels. The matrix $A^T W^2 A$ is defined as follows:

$$A^T W^2 A = \begin{bmatrix} \sum W^2 I_x^2 & \sum W^2 I_x I_y \\ \sum W^2 I_x I_y & \sum W^2 I_y^2 \end{bmatrix}$$

If we solve for the optical flow \mathbf{v} , we have:

$$\mathbf{v} = (A^T W^2 A)^{-1} A^T W^2 \mathbf{b} \quad (2.13)$$

2. BACKGROUND

and can be found by summing the spatial derivatives of the neighbouring pixels combined with the associated weights, as we have presented previously.

Both aforementioned methods rely on 1st-order intensity derivative to extract the motion vectors. As we have seen these methods are susceptible to the aperture problem, limiting the field of implementation as well the estimation results. This problem may be resolved by differentiating the optical flow constraint equation to obtain 2nd-order intensity derivatives, such as the one introduced by Nagel [12] which shows that image points with high Gaussian curvature (which can be expressed as $\det(\nabla\nabla I)$), such as grayscale corners, allow the estimate of full velocity in closed form.

- **Global versus Local Differential Methods**

Local-based methods have numerous advantages compared to the Global ones. First of all, the locality of the estimation region means that we do not have to rely on the entire image, in order to extract good estimates for the motion vectors. Furthermore, the local character of this method allows different regions to maintain their vector information while global methods enforce a global uniform vector field. This advantage makes this method ideal for estimating the optical flow in cases such as the ones of occluding objects. In such cases, the objects have similar spatial gradients but different orthogonal components. This aspect helps the discretion between the flow field of the occluding objects, whereas the global based methods would result in the mitigation of the flow field which will lead to the weakness of these desired flow discontinuities.

On the other hand, local based methods are more sensitive to noise due to the local nature of the estimation. This is something that does not happen in the global methods due to the smoothness constraint and the global nature of the estimation, which allows the motion information to spread over the image domain, filling homogeneous regions and thus resulting in a dense optical flow field.

2.3.2 Correlation based Methods

Correlation-based approaches use the intensity conservation assumption of the optical flow constraint. This allows the application of a similarity measure between the neigh-

2.3 Optical Flow Computation Techniques

bourhood of the pixel in the initial image and a destination neighbourhood in the second image. Maximization of the similarity measure means that we have found the best fitting optical flow vector computed by the displacement from the initial to the final position of the pixel between the frames. One disadvantage is that the optical flow vectors computed are discrete, which may lead to information loss concerning the real motion. The similarity measure can be expressed as:

$$\sum_{x,y \in \text{image domain}} \sum f(I(x,y,t), I(x+\delta x, y+\delta y, t+\delta t)) \quad (2.14)$$

which the summation of a function f applied to the spatial neighbourhood of the pixel (x,y) given a displacement value between two adjacent frames, $\delta t = 1$. Most common similarity measures used in correlation based methods are the product of the pixel values or the sum of squared difference(SSD).

Correlation-based methods have a number of advantages. First of all, they are less sensitive to noise than differential methods. The estimation will always be accurate as long as the correct neighbourhood yields the highest similarity value compared to all the other candidate neighbourhoods. Furthermore, they can be implemented to all image structures, even to occluding structure cases and yield satisfying results.

However, despite the advantages, correlation methods display significant disadvantages. One of the disadvantages is the computational cost due to the fact that we have to calculate the similarity measure for a number of different displacements for each pixel which in the case of large displacements they will lead to great computational cost. Furthermore, if the displacement exceeds the area surrounding the pixel, then the optical flow vectors computed are not accurate at all. Also, the size of the neighbourhood as well as the search window used, affect the final result. The window has to be large enough to contain enough information in order to find the correct result, and also has to be proportional to the size of the search area for the displacement. Finally, in cases of homogeneous areas correlation methods will result in ambiguous results due to the similar intensity values present.

Block Matching:

Despite their disadvantages, correlation based block matching techniques are being widely used. Most known methods presented in the block matching field are the Three

2. BACKGROUND

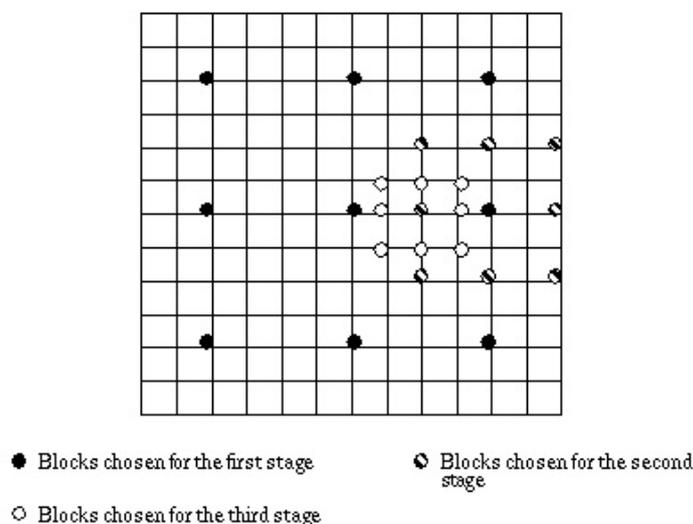


Figure 2.4: Search Pattern of the Three Step Search algorithm. The figure was taken from http://www.ece.cmu.edu/~ee899/project/deepak_mid.htm

Step Search Algorithm (**TSS**) introduced by Koga et al. [7], the Fourth Step Search Algorithm (**FSS**) by L. M. Po and W. C. Ma [8], the Diamond Search Algorithm (**DS**) by S. Zhu and K. K. Ma [9] and many others.

As expected, these algorithms use correlation- based methods with similarity measures such as the mean squared error (MSE), minimum absolute difference (MAD) and sum absolute difference (SAD) for extracting the motion field. For eliminating the issue of large and small displacements affecting the accuracy of the results the block matching approaches use either varying window sizes instead of fixed window sizes for their search(e.g. TSS algorithm) or different window shapes(e.g. Diamond Search algorithm, Cross Search algorithm) or even both.

For example, the TSS, in the first step the search is based on 9×9 search window at nine locations, in the second step the search is reduced in a 5×5 search window surrounding the location determined by the first step and finally in the third step the search window used has size 3×3 (Figure 2.4).

Hybrid Correlation Based Methods:

Another approach used in correlation based methods is using correlation in conjunction with feature based methods. One good example, is the work of Barnard and Thompson [6], which is a hybrid method using feature-based methodology in accordance with correlation-based methods to determine similarity between points.

2.3.3 Feature based Methods

Feature based methods are based on the idea of focusing the estimation on the vector which contain the most information which will give the best estimation for the motion. The basic steps in this method are the detection of features and the determination of correspondences between images. Furthermore, we need to find a way to match feature points between images. The most often, methods used for this purpose are correlation-based methods, such as squared correlation, absolute value difference, sum of squares difference(SSD), yielding valuable results.

The only drawback of this method, is that features of interest must be selected carefully so that they are easy to identify and that they contain as much information as possible in order to have accurate estimations. For examples, feature elements can be considered corners whose curvature can provide accurate estimations for the optical flow field.

2.3.4 Frequency based Methods

Frequency based methods use orientation sensitive filters, such as Gabor filters, in the Fourier domain. Fourier- based methods allow the motion estimation in image structures, in which feature-based or correlation-based methods have a difficulty in estimating their motion. This is due to the fact that the extraction of spatio-temporal energy, in Fourier space, results in oriented energy patterns which may contain more useful/obvious information for the motion estimation. This concept was first introduced by Adelson and Bergen [10].

2. BACKGROUND

Fourier space in signal processing is widely used since the functions describing a signal can be expressed as a summation of cosine and sine functions allowing the analysis of the function. Also, the computations in the frequency domain are done much easier which in the case of image processing will reduce the computational cost for the motion estimation process.

The Fourier transform, as presented in Beauchemin [2], of the brightness conservation assumption, equation(2.1), can be expressed as follows:

$$\bar{I}(\mathbf{k}, \omega) = \bar{I}_0(\mathbf{k}) \delta(\mathbf{v}^T \mathbf{k} + \omega) \quad (2.15)$$

where $\bar{I}_0(\mathbf{k})$ is the Fourier transform of the intensity $\bar{I}(x, t)$, δ is the Dirac delta function and \mathbf{k}, ω denote spatio-temporal frequency. In this way, the optical flow constraint equation can be expressed as:

$$\mathbf{v}^T \mathbf{k} + \omega = 0 \quad (2.16)$$

This equation, according to Beauchemin [2]:” shows that the velocity of a translating 2-D pattern is a function of its spatio-temporal frequency and forms a plane through the origin of the Fourier space”.

2.3.5 Hierarchical Based Methods

Hierarchical Methods address the problem of displacements occurring over large areas of the screen in cases where the frame rate can not capture accurately the rate of the motion, due to its low speed. As mentioned before, in such cases correlation based methods do not produce valid results, due to the fact that they are space limited with their search window’s size.

The main idea behind the hierarchical approach is that images created from the initial image with a decreasing resolution will allow us to compute initial motion estimations which will be used as initial guesses of the motion vectors as we move to higher resolution images of the initial image. This approach can be represented in a form of a pyramid data structure such as a Gaussian or Laplacian pyramid(Figure 2.5).

Lower resolution levels of the pyramid will correspond to greater displacements. As we proceed to higher resolution levels of the pyramid these motion vectors are further constrained until the bottom of the pyramid or the input image is reached.

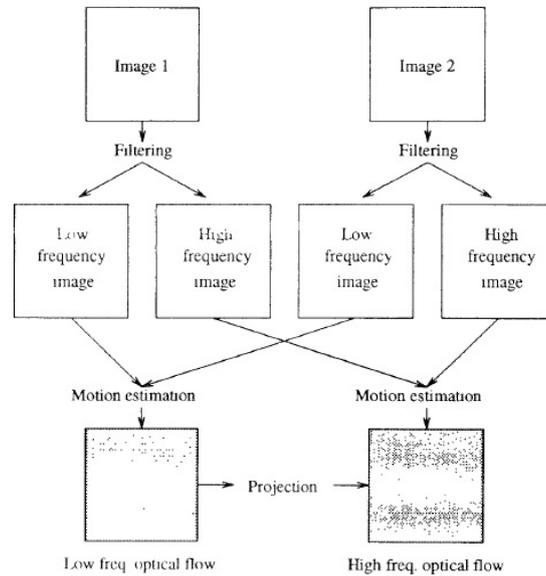


Figure 2.5: Hierarchical Model for optical flow computation. This pyramid scheme was taken from Beauchemin [2].

The hierarchical approach can be combined with almost all optical flow techniques, for example as Beauchemin [2] mentions in his review, there have been approaches that combine the hierarchical approach with correlation based methods, for motion vector computation (Ananda [11]), or even with differential approaches, who use known differential techniques such as Horn- Schunck’s method, to compute the motion vectors at each stage.

2.3.6 Method Comparison

All the aforementioned methods have their advantages and disadvantages, making them ideal for specific flow cases but at the same time less effective on other.

Differential methods are computationally less effective compared to correlation and hierarchical based methods but the local class tends to be more pruned to noise. Even more the global differential class does not allow flow field discontinuities which in the case of occluding objects can be essential but it is ideal when we are dealing with homogeneous flow fields. The motion field propagation results in dense optical flow fields which are

2. BACKGROUND

smoothed and spread uniformly in the image plane.

Although correlation based methods can be more computationally expensive they are less sensitive to noise compared to the differential class. Furthermore, they will result in the best estimation for the motion vectors as long as the correct destination neighbourhood shows the highest similarity value. On the other hand, their accuracy is directly related to the size of the search window. The size of the search window must continuously be calibrated according to the magnitude of the expected motion. Cases in which large displacements occur and the size of the search window is small will result in false estimations.

Frequency based methods, are fast due to the fact that the computation process is performed in the Fourier domain. Also, image structures such as dot patterns which may yield ambiguous results with correlation or feature based techniques, whereas Frequency based methods will result in unique energy patterns and thus, resulting in more accurate estimations. Even more the fact that every signal function can be expressed as a combination of cosine and sine functions in the Fourier domain allows the analysis to be easier.

Hierarchical methods can be combined with all the previous methods improving their results with a cost in computational speed. The resolution discrimination allows the estimated motion to be more accurate and methods whose result is directly associated with the volume of displacement such as correlation based or even differential, e.g. Lucas - Kanade, to be more accurate. This is due to the pyramid structure of hierarchical methods which leads to the refinement of the estimated motion vectors as we move to higher resolution stages. The fact that the estimation of the motion at each level is done based on the previous estimated motion combined with the information of the current pyramid level results in accurate motion vector field.

In conclusion, there is no best approach. Everyone has its weaknesses and everyone thrives in specific cases. The choice must be done based on the expected motion volume (i.e. large or small displacements), on the noise present in the data, on the data structure and nature and finally based on the importance of the computational cost.

2.4 Pattern Recognition

Pattern recognition is a classification task. In image analysis, pattern recognition aims at the detection and extraction of patterns from image data, as well as recognizing specified classes of objects by features extracted from the image data. By the term pattern, we refer to a subset of data that may be described by some well-defined set of rules. With the process of detection, the aim is to find an unknown number of instances of a known kind of patterns in the image, for example cars, whereas in the recognition process the task is to recognise a detected object as one of a specific kind, e.g. sports cars. The pattern detection class consists of methods concerning image segmentation, object matching and object detection while the recognition class, consists of methodologies in the fields of feature computation/reduction, classification and clustering. Pattern recognition in image analysis has numerous applications spanning to various fields of science, from automatic analysis of satellite pictures (e.g., weather condition, water reserves, mineral prospects,...) and analysis of medical images, to even identification of people from fingerprints, retinal scans or handwriting.

2.4.1 Pattern Recognition as a Classification Task

Pattern Recognition as a Classification Task involves the task of classifying objects according to feature values. Features are evaluated to separate objects into different classes. The basic step of the classification process is feature detection. Prerequisite for feature detection is the extraction of structures with common features using image segmentation. The step of segmentation is essential in the pattern recognition process because often, features are not computed from single pixels but from pixel sets. Their computation is erroneous if feature values change over the set. From all the above, pattern recognition as a process pipeline can be expressed as $Image \rightarrow Segmentation \rightarrow FeatureComputation \rightarrow Classification$.

The classification process involves the grouping of patterns (samples) according to their features into different classes. First of all the features to be used as a classification measure must be decided so that they will be relevant to the problem. The next step is to select the appropriate classification technique based on the type of the features. There have been many techniques developed for pattern classification, which are being categorized into supervised or unsupervised learning algorithms.

2. BACKGROUND

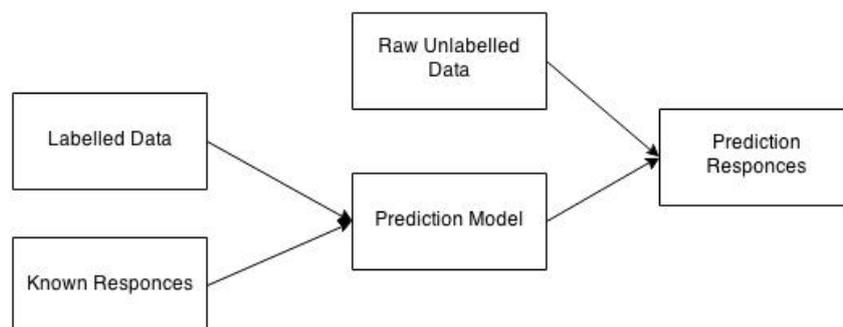


Figure 2.6: Supervised learning work-flow diagram. At the first stage a model is defined and trained using labelled data with known responses. After the model is trained it can be used to predict response values using new unlabelled data.

2.4.2 Supervised Learning

Supervised learning algorithms use labelled data to train a function used for the classification of unlabelled data. The variables under investigation can be split into two groups: independent variables and dependent variables. The target of the analysis is to specify a relationship between the independent variables and the dependent variable.

Supervised learning algorithms work as follows: Given a set of N training examples of the form $\{(x_1, y_1), \dots, (x_N, y_N)\}$ such that x_i is the feature vector of the i -th example and y_i is its label, a learning algorithm searches for an expression function $h : X \rightarrow Y$ with X being the input and Y the output spaces accordingly, as displayed in (Figure 2.6).

The most widely used learning algorithms are Support Vector Machines, linear regression, logistic regression, naive Bayes, linear discriminant analysis, decision trees, k-nearest neighbour algorithm, and Neural Networks (Multilayer perceptron).

2.4.3 Unsupervised Learning

In Unsupervised learning algorithms there are no labelled data to train a classification function. The targeted variable is unknown or we have insufficient recorded cases of it.

In unsupervised learning the classification model inference and application both rely on test data meaning that we cannot test our data and the only source of validation comes from the resulted classification which may be used to further calibrate the classification model (Figure 2.7). Approaches to unsupervised learning include methods such as clus-

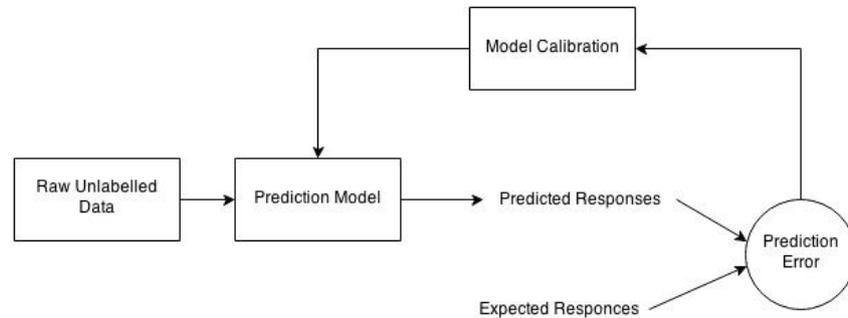


Figure 2.7: Unsupervised Learning model where the calibration of the classifier depends on the resulted classification data.

tering (e.g., k-means, mixture models, hierarchical clustering), hidden Markov models, blind signal separation using feature extraction techniques for dimensionality reduction.

2.4.4 Motion Pattern Recognition

One of the fields of computer vision in which pattern recognition has been widely used is motion pattern recognition. In recent years there has been an increased interest in the modelling and recognition of motion patterns. This motion information can be used in numerous applications such as human activity recognition, modelling human activity in robotics, blood flow types in the human heart, face recognition, fingerprint recognition and many others.

Like all pattern recognition methods, the first stage of motion pattern recognition methodologies, includes the most important motions to be segmented. In the second stage, the optical flow is estimated on the basis of the motions detected in the previous stage. To extract relevant motions, there have been many approaches spanning from the most simple ones such as the use of simple correlation methods e.g. sum of squares difference (SSD) along with clustering methods, e.g. k-means or EM, to more complex methodologies based, for example, on a spatio-temporal filtering, or even probabilistic and statistical methodologies.

2. BACKGROUND

Chapter 3

Problem Statement and Related Work

3.1 Motion Estimation in Fluid Flows

Methods of detecting the apparent motion of fluid-like motion in a video of, for example, clouds, ocean/river waves, and smoke have been one of the intensive research topics in computer vision. Motion field extraction in fluid flows can be especially difficult due to the dynamic nature of fluid motion. Dynamic motion differs from the motion of rigid bodies as it is subjected to forces, such as wind, which alter the motion. Fluid motion is the opposite of rigid body motion meaning that the positions among points in a fluid change constantly and with an unknown scale. The motion analysis of fluids, requires the consideration of the features present in fluid dynamics. According to Chang [28], a fluid with constant density and temperature is described by a velocity field u and a pressure field p , which depend on the space boundaries and change over time. The variation of these features can be described using the Navier-Stokes equations [13]:

$$\frac{du}{dt} = \frac{\partial u}{\partial t} + (u \cdot \nabla) u = -\frac{1}{\rho} \nabla p + \nu \nabla^2 u + f \quad (3.1)$$

$$\nabla \cdot u = 0 \quad (3.2)$$

where ν is the kinematic viscosity of the fluid, ρ is its density and f is an external force. The first equation resembles equation(2.3) where in this case the unknown variable is

3. PROBLEM STATEMENT AND RELATED WORK

the velocity vector. The presence of external forces such as gravity, the gradient field of pressure and diffusion affect the way in which the velocity field varies over time. Equation(3.2), as Chang mentions, denotes the incompressible nature of the fluid, whose volume does not change over time. Based on the motion of fluids described by the Navier-Stokes equations, we understand that fluids have non-rigid motion. Non rigid motion means that methods based on affine translational models cannot describe accurately the fluid motion.

There have been many approaches presented which try to estimated the fluid motion. Some of them are based on fluid dynamics using a continuity equation describing a specific feature of the fluid, such as the conservation of brightness or mass, or even differential frameworks combined with a smoothness velocity constraint. Other methods, are based on classical optical flow methods, such as differential methods e.g. weighted Lucas and Kanade's method, or even feature based methods along with correlation based methods. The latter use natural or artificial particles, which are deposited into the fluid flow to provide with more distinctive features which will improve the motion estimates. The basic categories of these techniques, along with the most representative ones, will be presented in the following section.

3.2 Related Work on Motion Estimation in Fluid Flows

Optical flow algorithms are mainly based on the brightness constancy assumption, which allows a mapping between adjacent frames based on intensity similarities. However, the non-rigid nature of fluid flow requires the use of constraints with the form of either clearly mathematical models or physically based feature constraint functions(i.e. constraint functions based on the physical properties and quantities of fluids)in order to model the multi-directional nature of the fluid motion.

To address the problem of estimating the fluid flow motion there have been many approaches which can be separated into the following categories:

- **Methods based on Properties of Fluid Mechanics**
- **Methods based on Physical Properties of Waves**

- Particle based Methods
- Methods based on Statistical Estimation

3.2.1 Methods based on Properties of Fluid Mechanics

One of the approaches used in Fluid Flow motion estimation is incorporating known optical flow techniques with the physical properties of fluid mechanics, such as the use of the conservation of brightness or mass.

Conservation of Mass Constraint: Conservation of mass can be a useful physical constraint for the fluid motion, according to Wildes et al. [14]. If $p(x, y, z, t)$ is considered as the density of a fluid then it is associated to a velocity field $\mathbf{V}(x, y, z, t) = (U(x, y, z, t), V(x, y, z, t), W(x, y, z, t))$. If now we assume that this fluid respects the conservation of mass law, then these two features can be connected as follows:

$$\nabla \cdot (p\mathbf{V}) + \frac{\partial p}{\partial t} = 0 \quad (3.3)$$

with ∇ the three-dimensional spatial gradient operator. For the case, where the transaction along the image plane is modelled as yielding intensities proportional to an object's density we can express the image intensity values as:

$$I(x, y, t) = \int_{z_1(x,y)}^{z_2(x,y)} p(x, y, z, t) dz \quad (3.4)$$

where $z_1(x, y)$ and $z_2(x, y)$ are the bounding surfaces of the specimen that is being imaged. If we impose equation(3.3) to equation(3.4) we get:

$$\int_{z_1(x,y)}^{z_2(x,y)} \nabla \cdot (p\mathbf{V}) dz + \int_{z_1(x,y)}^{z_2(x,y)} \frac{\partial p}{\partial t} dz = 0 \quad (3.5)$$

These equations, as it is presented by Wildes et al. [14], along with the work previously done by Fitzpatrick [15], ends up in the following equation:

$$\nabla_{x,y} \cdot I\mathbf{v} + \frac{\partial}{\partial t} I = - [p\mathbf{n} \cdot \mathbf{V}_{z_1}^{z_2}] \quad (3.6)$$

3. PROBLEM STATEMENT AND RELATED WORK

where \mathbf{v} is a new two-dimensional velocity field as the density-weighted average of the original three-dimensional velocity field as:

$$\mathbf{v} \equiv \frac{\int_{z_1}^{z_2} p \mathbf{V}_{x,y} dz}{\int_{z_1}^{z_2} p dz} \quad (3.7)$$

For the case of null normal flow at the boundaries, the right-hand side of equation(3.6) vanishes. Equation(3.6), allows us according to Wildes to derive that "the two-dimensional transmittance image of a three-dimensional fluid flow respecting the conservation of mass in three dimensions is a two-dimensional flow that respects the conservation of mass in two dimensions and it is known as the continuity equation". The continuity equation, based on the conservation of mass law, in fluid dynamics links the density and velocity of a fluid.

Wildes et. al. [14] describes that the application of the conservation of mass flow continuity equation to a temporally varying image yields:

$$I_x u + I_y v + I u_x + I v_y + I_t = 0 \quad (3.8)$$

which can be also written as:

$$\nabla \cdot (I \mathbf{v}) + I_t = 0 \quad (3.9)$$

where $\mathbf{v} = (u(x, y, t), v(x, y, t))$ is the imaged flow, $\nabla \equiv \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)$ is the spatial gradient operator, and subscripts denote partial differentiation. The previous equation, as we have mentioned earlier, is the continuity equation and can be contrasted with the brightness constancy constraint, equation(2.3).

The continuity equation constraint has been used by various researchers in fluid flow motion estimation which incorporate it with additional constraints to further restrict the velocity motion field and to ameliorate the effects of noise, such as Wildes et. al. [14] and Nakajima et. al. [16], in which the continuity equation is used to follow a certain property of the fluid, along with a smoothness velocity constraint based on the first derivative of a velocity.

For example, Wildes et. al. [14] imposes a smoothness constraint c_s and then follows the methodology of Horn and Schunck. The measures of continuity and smoothness can be combined and thus leading to a problem of the form:

$$\min \int \int (\lambda c_c + c_s) dx dy \quad (3.10)$$

3.2 Related Work on Motion Estimation in Fluid Flows

with λ being a weighting parameter that trades off adherence to the continuity constraint and smoothness of flow and the smoothness constraint is expressed as:

$$c_s = u_x^2 + u_y^2 + v_x^2 + v_y^2 \quad (3.11)$$

Minimization of the constraint equation with respect to flow parameters (u, v) variation calculus problem and can be expressed using partial equations as follows:

$$\nabla^2 u = -\lambda (I_{tx} + I_{xx}u + I_{yx}v + 2I_x u_x + I_y v_x + I_x v_y + I v_{yx} + I u_{xx}) I \quad (3.12)$$

$$\nabla^2 v = -\lambda (I_{ty} + I_{xy}u + I_{yy}v + I_y u_x + I_x u_y + 2I_y v_y + I u_{xy} + I v_{yy}) I \quad (3.13)$$

The final optical flow field \mathbf{v} is just the result of the numerical system of the matrices.

3.2.2 Methods based on the Physical Properties of Waves

Another approach, in the topic of fluid flow motion estimation, is based on the assumption that many fluid-like motion changes are due to wave phenomena that lead to a brightness change. This approach has been adopted by many researchers, such as Jahne et. al. [17] which estimated the orientation of wave motion and Saikano [18] who applied a wave generation equation along with a two-step optimization, and others.

From the aforementioned researchers, one of the most popular methods representing this category is the one presented by Saikano [18]. He introduced a wave phenomenon-based optical flow framework for fluid-like images in which a wave generation equation from ocean engineering is used to model an image brightness change.

According to Saikano, if $H(x, y, t)$ is the image intensity at pixel coordinates (x, y) , at time which is described according to the wave generation theory, the fluid motion is described by a multi-directionality irregularity (MI) model as follows:

$$H(x, y, t) = \sum_{m=1}^M \alpha_m^* \cos(k_m^{x*} x \cos \vartheta_m^* + k_m^{y*} y \sin \vartheta_m^* - 2\pi f_m^* t + \varepsilon_m) \quad (3.14)$$

where α_m^* : amplitude, $(k_m^{x*}, k_m^{y*})^*$: wave-number components, f_m^* : frequency, ϑ_m^* : orientation, ε_m^* : noise and M: the number of cosine functions used to describe the wave.

From this equation we can derive the following observations, first of all a wave motion is multi-directional and also, that its motion is associated with features such as frequency,

3. PROBLEM STATEMENT AND RELATED WORK

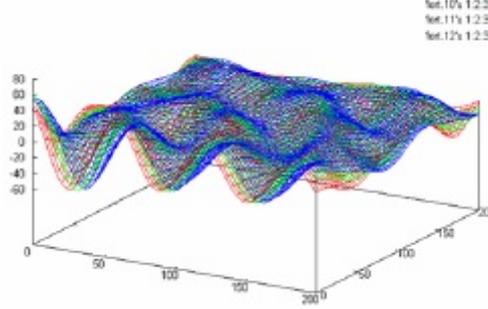


Figure 3.1: Example of a simulated irregular wave profile: Two-dimensional waves using eq.(3.15) with $M=150$. Figure was taken from Saikano [18].

amplitude, wave-number, wavelength, height, and orientation. An irregular motion like the wave motion can be represented by a number of cosine functions M (Figure 3.1).

The multi-directionality irregularity (MI) model used to express the image intensity is associated to the optical flow model using Haussecker's [19] model:

$$\nabla I \mathbf{v} + I_t = \frac{d(\text{physics})}{dt} \quad (3.15)$$

where

$$\frac{d(\text{physics})}{dt} = \frac{d(H(x, y, t))}{dt} \quad (3.16)$$

where $\mathbf{v}(u, v) = \left(\frac{dx}{dt}, \frac{dy}{dt} \right)$: optical flow, I : image intensity, $\nabla = \frac{\partial}{\partial x} + \frac{\partial}{\partial y}$, $I_x^n = \frac{(I_{i+1,j}^n - I_{i,j}^n)}{\Delta x}$, $I_y^n = \frac{(I_{i+1,j}^n - I_{i,j}^n)}{\Delta y}$, $I_t^n = \frac{(I_{i+1,j}^n - I_{i,j}^n)}{\Delta t}$, H : image intensity as express by the MI model.

The previous equation means that the image brightness changes according to the physical wave phenomena. Furthermore, Saikano imposes a constraint is in order to ensure a more stable and estimation. If $c = |\mathbf{v}| = (u^2 + v^2)$ is the optical flow, then the constraint introduced is:

$$c^2 \propto |\gamma - f|^{1/2} \quad (3.17)$$

where $\gamma = \frac{3g}{16\pi h}$, g : gravity acceleration and h : water depth.

3.2 Related Work on Motion Estimation in Fluid Flows

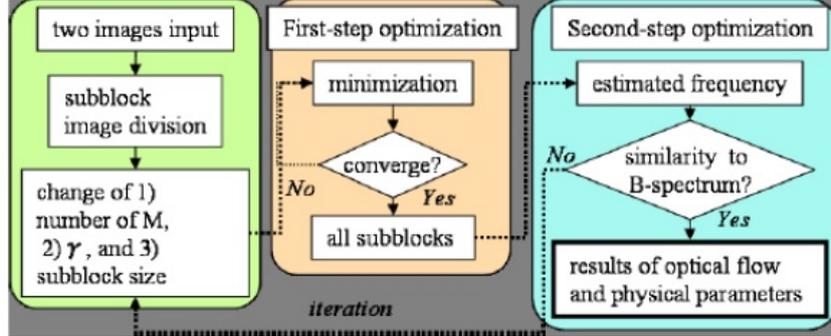


Figure 3.2: The two-step optimization algorithm used to minimize eq.(3.19) and compute the optical flow components and the wave parameters. Figure was taken from Saikano [18].

Saikano also, introduces an *objective function* in order to estimate two optical flow components along with five wave-related parameters which are the two wave-number components, frequency, amplitude, and orientation:

$$E(u, v, k_m^x, k_m^y, f_m, \alpha_m, \vartheta_m) = \sum_{\Omega \in \mathbf{R}^2} \rho_{imgvar}(e_0, \sigma) + \lambda_1 \sum_{\Omega \in \mathbf{R}^2} \rho_{waveconstraint}(e_1, \sigma) + \lambda_2 \sum_{\Omega \in \mathbf{R}^2} \rho_{smoothconstraint}(e_2, \sigma) \quad (3.18)$$

where $\rho(z, \sigma) = \log(1 + 0.5(z/\sigma)^2)$, $\partial\rho/\partial e = 2e/(2\sigma^{2+e^2})$, $e_0 = |I_t + I_x u + I_y v - \frac{\partial H}{\partial t}|$, $e_1 = |u^2 + v^2 - \alpha^2(\gamma - f_m^2)^{1/2}|$, $e_2 = |u_x^2 + u_y^2 + v_x^2 + v_y^2|$

Lastly, using a two-step optimization algorithm (Figure 3.2) they minimize the previous equation(3.19) in order to compute the optical flow components as well as the wave related parameters. This is a recursive algorithm which minimizes the aforementioned equation using as a means of comparison the deviation between the estimated frequency and the one computed using the Brettschneider energy spectrum:

$$B(f) = 0.25H_{1/3}^2 T_{1/3} (T_{1/3} f)^{-5} \exp-1.03 (T_{1/3} f)^{-4} \quad (3.19)$$

where $H_{1/3}$: significant wave height, $T_{1/3}$: significant frequency.

The advantage of this method is that the smooth and discontinuous motion in the inhomogeneous image brightness of clouds, for example, as estimated by this method is

3. PROBLEM STATEMENT AND RELATED WORK

estimated visually plausible, whereas the previous methods estimate only smooth and uniform motion.

3.2.3 Particle based Methods

Another approach, on the topic of fluid flow motion estimation, is using Correlation based methods along with Particle based methods. There have many methods developed with the most representative ones being, the Particle Image Velocimetry method(PIV), the Particle Tracking Velocimetry method(PTV) and the Space-Time Image Velocimetry method(STIV). The aforementioned techniques are Particle based techniques using optical flow methods such as Correlation based, feature based,e.t.c to estimate the fluid flow motion.

Particle: By the particle, we define an element or a group of elements(pattern), which can be used as a tracer, allowing its tracking along the frames. The displacement of the tracer is later used for extracting the motion field.

Tracers can be grouped in two different classes:(1)natural tracers and (2)artificial tracers. In the class of natural tracers we include elements such as leaves, water foam, debris floating along the fluid flow. On the other hand, artificial tracers, are element induced by humans in order to be used as tracers, such as balls and buoys.

Particle Image Velocimetry method(PIV):

In the Particle Image Velocimetry method the displacement of an image pattern in the template window at a certain time point is estimated by searching the location of the most similar image pattern in next time snap. Finally, the velocity is calculated from the displacement of the image pattern from the first to second images divided by the time interval between the first and second images.

There have been many approaches on the subject of the image pattern comparison in the PIV method, such as simple correlation methods like the one presented by Bradley et. al. [20], who used a linear correlation coefficient in a non-shifting window, or more complex ones, such as the one introduced by Lecordier et.al. [21], who developed an iterative evaluation program which allows the shifting and rotation of the interrogation windows according to a previously estimated displacement field.

3.2 Related Work on Motion Estimation in Fluid Flows

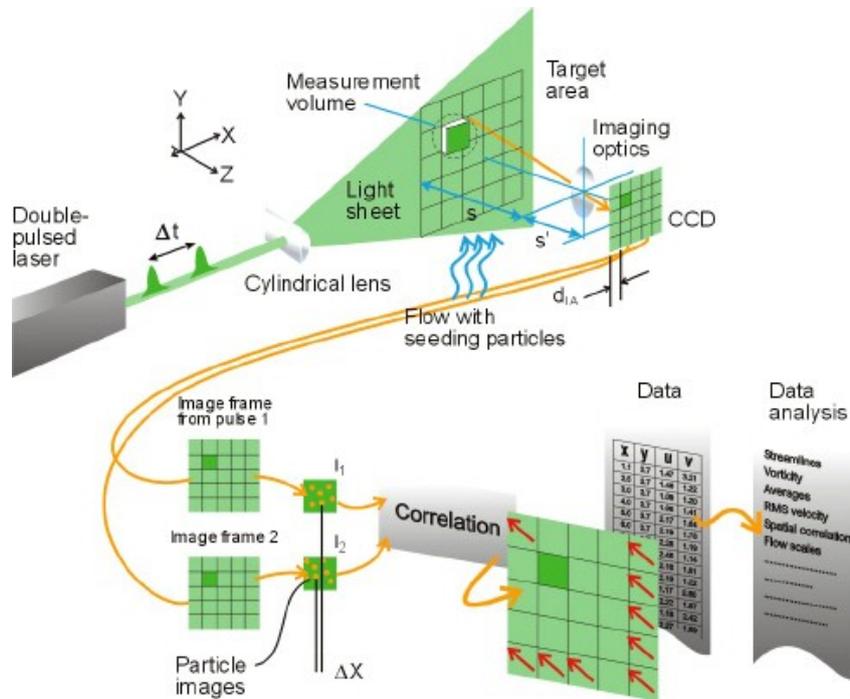


Figure 3.3: The PIV technique implementation procedure in small scale flows such as the measurement of fluid flows using laser beams as tracer pointers. Figure taken from <http://www.dantecdynamics.com/measurement-principles-of-piv>.

PIV techniques have been successfully used various fields of Science such as in fluid mechanics, in the measurement of fluid density or fluid flow , in aerodynamics measuring the turbulence, or even in medicine for the measurement of blood flow. PIV techniques are being used from the measurement of small scale flows (with the use of laser beams as tracer pointers(Figure 3.3)), to the measurement of large scale flows such as rivers and streams.

One of the most impressive works in the area of large scale flows are the those of Fujita [23], Tsubaki et. al. [22] and Bradley et. al. [20]. All of them have developed automated systems for the measurement of River and Stream flows using PIV techniques and are now being used as surveillance systems for river flow monitoring(Figure 3.4). For example, Bradley uses, as mentioned previously, a linear correlation coefficient R for

3. PROBLEM STATEMENT AND RELATED WORK

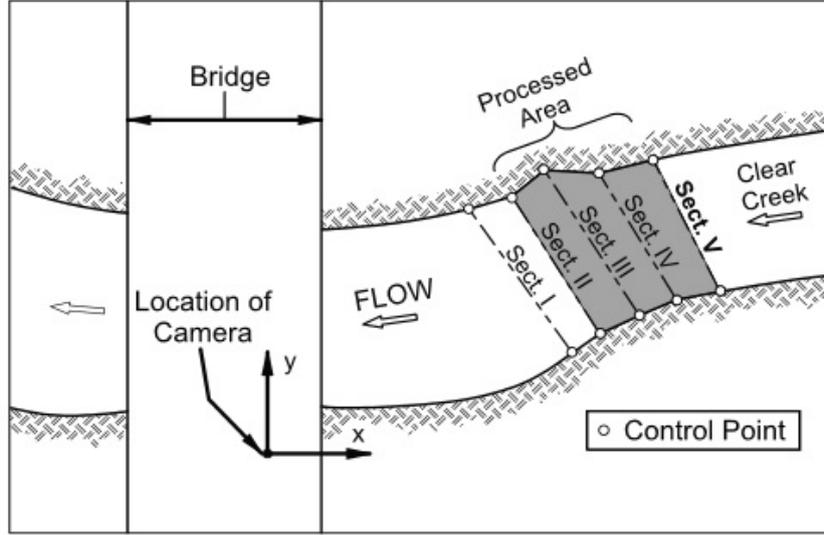


Figure 3.4: Bradley's System Layout and Study site at Clear Creek near Oxford, Iowa. The selected area is video taped by a camera laid on top of a bridge. The video data are used in the motion estimation step. The actual river flow velocity is computed using the derived velocity field and some know ground points. Figure was taken from Bradley et. al. [20].

similarity benchmark between the two sets of pixels:

$$R = \frac{\sum_l \sum_k (a_{kl} - \bar{a}_{kl}) (b_{kl} - \bar{b}_{kl})}{\sqrt{\sum_l \sum_k (a_{kl} - \bar{a}_{kl})^2 \sum_l \sum_k (b_{kl} - \bar{b}_{kl})^2}} \quad (3.20)$$

where a_{kl} are the grey-scale values for pixels in the interrogation spot, and b_{kl} are the grey-scale values for the corresponding pixels in the search area and $\bar{a}_{kl}, \bar{b}_{kl}$ the average intensity values in the interrogation spot and the search area.

Particle Tracking Velocimetry method(PTV):

On the other hand, the technique of Particle Tracking Velocimetry (PTV), single particle images taken from a sequence of frames are combined together in order to form an image of the particle present in the sequence of frames(Figure 3.5) and it is known as particle matching. PTV has been employed in many ways but essentially there are two distinct types:(a) multiply-exposed single images and (b)singly-exposed multiple images.

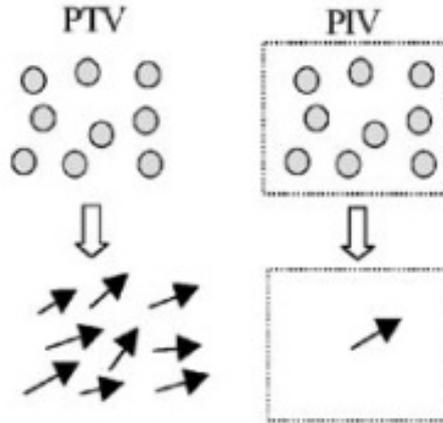


Figure 3.5: A comparison of the spatial resolution between PIV and PTV. Figure was taken from Stitou and Riethmuller [25].

Multiply-exposed single image techniques, such as the one presented by Agui and Jimenez [24], rely on relatively sparse seeding of the flow to avoid overlapping images. A singly-exposed multiple image PTV technique is tracking a given particle through sequential images which contain a relatively high density of particles.

Most of PTV implementations have relied on large search windows in the second image of an image pair in order to track particles in the pair. This necessitates low seeding densities to avoid pairing ambiguity. On the other hand, they tend to be more accurate than correlation based PIV techniques since they are relatively unaffected by the presence of displacement gradients.

There have been many approaches in the PTV class using various correlation based techniques. Stitou and Riethmuller [25], for example, presented a PTV class algorithm in which the particle extraction is based on an intensity level threshold. The threshold value is determined using the local intensity distributions. The particle tracking is the hybrid of a cross-correlation and a tracking method. Initially velocity distributions are computed using the cross-correlation method at grid points and later, the particle matching is carried out based on the velocity at the grid. To reduce the mismatching, the vector is compared with the surroundings.

After comparison of PTV and PIV methods, in cases of high seeding density and small particle flows, PIV methods appear to be the most accurate ones whereas, in cases

3. PROBLEM STATEMENT AND RELATED WORK

of low seeding density and large particle flows, PTV methods exceed the PIV in accuracy.

Space-Time Image Velocimetry(STIV):

In the STIV method, the one-dimensional luminance distribution on the interrogation line is combined with a two-dimensional space-time image (Fujita and Tsubaki [26], Fujita et al. [27]). Then, the space-time image generated is analysed to obtain the velocity component directed to the interrogation line direction. The gradient of the image pattern indicates the speed in which the luminance distribution is propagated in the interrogation line along the space- time image.

The calculation of the orientation angle of the interrogation pattern is useful because it allows the rejection of non-coherent, to the appropriate motion vector, motion vectors. From each frame, as shown by Fujita et al. [27], the orientation angle of the interrogation line can be calculated and used to compute the coherence measure C :

$$C = \frac{\sqrt{(J_{xx} - J_{tt})^2 + 4J_{xt}^2}}{J_{xx} + J_{tt}} \quad (3.21)$$

where C is the Coherence measure, J_{xx} , J_{tt} , J_{xt} are structure tensors calculated as follows:

$$J_{xx} = \int_A \frac{\partial g}{\partial x} \frac{\partial g}{\partial x} dx dx \quad (3.22)$$

$$J_{tt} = \int_A \frac{\partial g}{\partial t} \frac{\partial g}{\partial t} dt dt \quad (3.23)$$

$$J_{xt} = \int_A \frac{\partial g}{\partial x} \frac{\partial g}{\partial t} dx dt \quad (3.24)$$

where

$$\frac{\partial g}{\partial x} = \frac{g_{i+2} - 8g_{i+1} + 8g_{i-1} - g_{i-2}}{12\Delta x} \quad (3.25)$$

where $\frac{\partial g}{\partial x}$ is the 4th order central difference scheme and $g(x, t)$ is the gray intensity level.

For example, as shown in Figure 3.6, the motion vectors belonging to the brick must be excluded from the calculation of the mean velocity of the flow pattern. The exclusion of the wrong motion vectors allows the computation of more accurate motion vectors for the fluid flow.

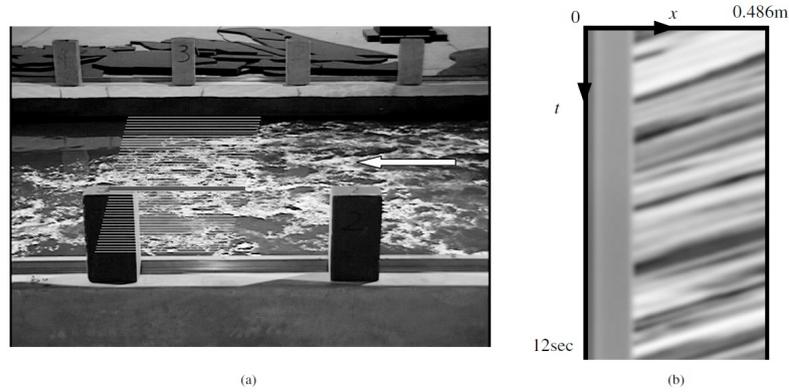


Figure 3.6: Laboratory flume flow visualized by tracer; (a) oblique-angled image and the location of searching line (b) an example of STI for a thick searching line indicated in (a). Figure taken from Fujita et al. [27].

The velocity components, according to Fujita's and Tsubaki's method, are calculated using the orientation computed along the interrogation line as:

$$U = \frac{S_x}{S_t} \tan \phi \quad (3.26)$$

where U is the average velocity, S_x is the length scale of the pattern in m/pixel, S_t is the unit time scale of the time axis in sec/pixel and ϕ is the mean orientation of the pattern along the interrogation line.

The main advantages of the STIV method against PIV techniques, are that (1) it is faster in terms of calculation speed compared to PIV methods (Fujita mentions it can be 10 times faster), (2) it requires less memory because the computation is only carried out using two adjacent frames whereas, PIV techniques require more than 2 adjacent frames.

On the other hand, the STIV method measures the mean velocity for downstream flow cases whereas, LSPIV techniques measure two-dimensional instantaneous velocity. Also, the STIV method cannot represent the detailed flow structure because it is a one-dimensional measurement method.

3.2.4 Methods Based on Statistical Estimation

Another approach, is using statistical estimation methods to derive estimations about the motion field. One of the most known approaches in this class is the one presented by

3. PROBLEM STATEMENT AND RELATED WORK

Chang et.al. [28], whose method involves estimating a local flow probability distribution function at each pixel using the STAR model and the data from a spatio-temporal neighbourhood and then feeding the set of distribution functions into a global optimization framework.

Statistical Estimation methods for determining flow vectors should be adopted since the relative positions of neighbouring points change faster in a fluid than on a rigid body. In order to estimate the dynamic fluid motion a stochastic fluid motion model based on the characteristics of fluid motion, must be build. Chang et.al. [28], for example, considers the flow vector at a pixel at a certain time as a random variable with a probability distribution function, which indicates the probability of the pixel to be displaced in each position of the destination neighbourhood. The flow vector at each pixel in the local spatio-temporal region can be considered as a random sample from this distribution function. Once the estimation of the probability distribution function is calculated, the flow vector of the pixel is more likely to be consistent with a vector with a high probability and thus yielding the flow vector.

The local distribution function, computed previously, defines the probability of every potential velocity vector at a pixel. However, the motion field of a dynamic fluid actually has a unique velocity vector everywhere, so Chang et.al. [28] proposed an optimization-based method to extract a dense motion field with a unique velocity vector everywhere from the set of local distribution functions. This leads to a smooth and dense motion field, which describes the dynamic motion of fluids.

Our approach is based on Chang's method, so we will present the theoretical background of this method in following Chapter, along with our diversifications on this method and the impact they have in the performance of this method.

3.3 Pattern Recognition in Fluid Flows

Pattern Recognition in Fluid Flows addresses the need of analysing and visualizing the fluid flow velocity data sets in order to extract important structures present in the flow, such as shock waves, vortex cores, boundary layer separation and reattachment lines, flow topology, and boundary layer characteristics.

Automatic extraction and visualization only of these important structures has many applications in a wide variety of fields where there is a need to automatically select and

3.4 Related Work on Pattern Recognition in Fluid Flows

visualize regions of interest. For example, one of the main applications is aerodynamics, where vortices are typically considered as the most important structures in flow fields.

In fluid dynamics, such structures provide information about the fluid motion and its nature, for example, vortices in turbulent flows helps to understand and explain the inverse cascade phenomenon in which the energy of the flow concentrates itself in a few large vortex-like structures. The vorticity of the fluid carries important information about the nature of the flow measuring the swirl in a fluid.

3.4 Related Work on Pattern Recognition in Fluid Flows

In this section, a number of feature extraction techniques will be discussed that have been specifically designed for certain types of features. These techniques are often based on physical or mathematical (topological) properties of the flow. Features that often occur in fluid flows are vortices, parallel flow, converging and diverging flows. Those techniques can be categorized into the following classes:

- **Methods based on Image Processing**
- **Methods based on Vector Field Topology**
- **Methods based on Physical Characteristics**

3.4.1 Methods based on Image Processing

Basic image processing techniques can be used for feature extraction from scientific data. Edges or boundaries of objects are found by detecting sudden changes in the data values, which are denoted by high gradient magnitudes, in these point of interest. Therefore, basic image segmentation techniques, such as thresholding, region growing, and edge detection can be used for feature detection.

However, in computational fluid dynamics, often grid types are used. Many techniques from image processing cannot be easily adapted for use with grids. Furthermore, the fact that motion pattern recognition deals with motion patterns, i.e. vector patterns, which

3. PROBLEM STATEMENT AND RELATED WORK

are not characterized by a known range of vector values or intensity patterns, results in the basic image processing techniques to be ineffective.

One way to convey image processing to vector fields is to define the convolution on vector fields. In order to do so, a product of vectors has to be defined as the convolution has to calculate the product of vectors. There have been several approaches using the product of vectors, such as the one proposed by Heiberg [29], who defines scalar convolution based on the scalar product of two vectors, or the one introduced by Ebling and Scheuermann [31], used Clifford Algebra to express the product of vectors.

Vector Pattern Matching:

Heiberg [29], defines a scalar convolution on vector fields based on the scalar product of two vectors:

$$s_n(r) = \int \int \int_{\Omega} U(\xi) * P_n(r - \xi) d\xi \quad (3.27)$$

where s_n is the filter response, U is the normalized vector field and P_n the filter mask with direction n . This convolution is referred as the scalar convolution.

Heiberg et. al. [30], uses this approach combined with different filter masks of known patterns in order to identify blood flow patterns in the human heart, using 3-Dimensional vector fields (Figure 3.7). The vector field is normalized and the filter mask weighted with a rotational symmetric function. The filter mask is rotated in six directions evenly distributed over a hemisphere and the convolutions of the six filters with the field of interest are calculated. Finally, using a tensor of orientation created based on the squared filter responses and the directions, they calculate the direction and similarity between the filter patterns and the field (Figure 3.8). One of our approaches, in fluid flow pattern recognition, presented in the following Chapter, is modification his approach, modified so that it can be used for pattern recognition in 2-Dimensional fluid flow motion fields for natural phenomenon, such as rivers and streams. Heiberg's approach will be presented in the next Chapter along with the proposed modifications.

Advantages and Disadvantages of this method: The advantages of Heiberg's approach are that it is a robust method insensitive to noise, due to the fact that is regional and does not look only at local gradients. Also, this method is not limited in the choice of only certain types of features to be examined. On the other hand, the disadvantage of

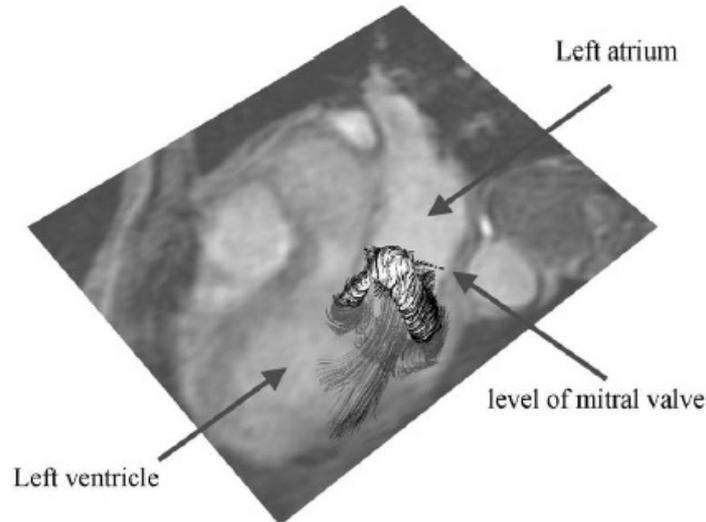


Figure 3.7: Results of flow characterization of blood flow in the human heart and the aorta as presented by Heiberg et.al. [30]. A complete vortex ring can be seen below the mitral valve (the valve between the left atrium and the left ventricle) in a healthy normal heart. The vortex core is shown as a white isosurface and streamlines are released from the isosurface. The time in the heart cycle is the beginning of the atrial contraction, when blood flows from the left atrium to the left ventricle. Streamlines were also released in the valve orifice.

this method is that it only works on symmetric filter masks. Also, the accuracy of this method has not yet been tested for 2-Dimensional field, leaving a window for research.

Clifford Convolution in Vector Pattern Matching:

Ebling and Scheuermann [31], used Clifford Algebra to express the product of vectors. In Clifford Algebra [32], a multiplication of vectors supplies us with sinus and cosinus of the angle between the two vectors and the plane in which the angle is measure. Therefore, Clifford Convolution for pattern matching of the vector field and vector field masks gives the direction of the structure.

Having found the direction of the structure then the vector filter masks can be rotated in the appropriate direction. The scalar convolution of the rotated filter mask and the vector field is computed as a similarity measure. In 2-D space three directions are enough

3. PROBLEM STATEMENT AND RELATED WORK

Cut view						
Top view						
Pattern	Vortex	Swirl (right hand)	Swirl (left hand)	Converging	Diverging	Parallel flow

Figure 3.8: Patterns used by Heiberg et. al. [30]

for extracting similarity measures for each filter mask, whereas in 3-D space there is a need for six directions evenly distributed on the circle or the sphere. The similarity values can be visualized using iso-surface algorithms like matching cubes.

Advantages and Disadvantages of this method: The advantages of this method are that it is not limited to any filter masks and it is insensitive to noise and at least as precise as the one of Heiberg. On the opposite this method has not yet tested for irregular grids and the only accuracy tests were made for a gas furnace chambers(Figure 3.9).

3.4.2 Methods based on Vector Field Topology

Another approach to feature extraction is the topological analysis of 2D linear vector fields, which is based on detection and classification of critical points. These are the points where the vector magnitude is zero. The concept of critical points in feature extraction was introduced by Helman and Hesselink [34], who presented a method in which by computing the eigenvalue and eigenvectors of the velocity gradient tensor, the critical points can be classified and tangent curves can be computed.

According to Post, Vrolijk et. al. [35], critical points can be clustered to attract- ing/repelling nodes, saddle point, and spiral/saddle(Figure 3.10). Using this informa- tion, a schematic visualisation of the vector field can be generated. Critical points are

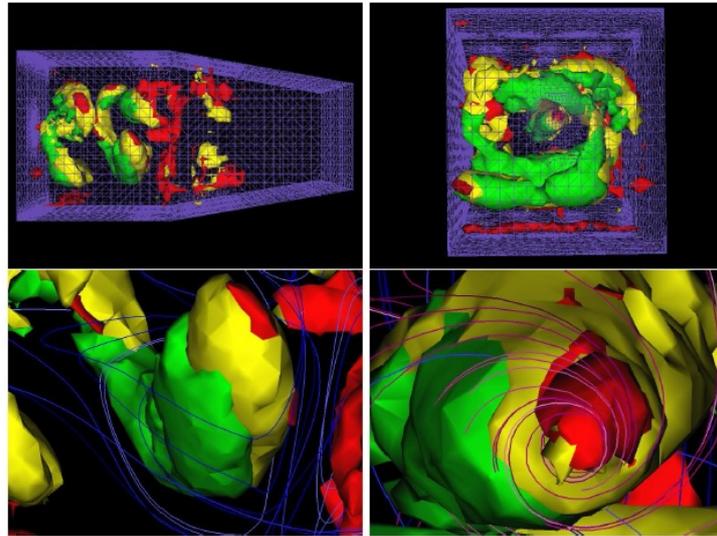


Figure 3.9: Parts of a gas furnace chamber as present by the method of . Pattern matching of the vector field of the chamber and $3 \times 3 \times 3$ (red), a $5 \times 5 \times 5$ (yellow) and a $8 \times 8 \times 8$ (green) rotation mask was computed. The similarities are visualized using a marching tetrahedra with an isovalue of 0.5. On the right, the main inflow of the gas is shown. At the bottom, some streamlines are drawn additionally. Figure was taken from Ebling and Scheuermann [31].

widely used in fluid flow vector field characterization due to the fact that their patterns can represent the flow features present in a fluid flow and can be expressed in the form of a graph known as a topological skeleton(Figure 3.11).

One of the approaches, which combines image processing methods and critical point methodology is the one presented by Schlemmer et. al. [33], in 2007. Their approach for analysing 2-Dimensional flow field data is based on the idea of invariant moments. Their approach supports the computation of moments at multiple scales, facilitating fast pattern extraction and recognition. This can be done for critical point classification, but also for patterns with greater complexity.

Moment Invariants in Vector Pattern Matching:

Moment Invariants, a technique known from image understanding, can be used for pattern recognition in scalar data. Schlemmer et. al. [33] presented an approach for

3. PROBLEM STATEMENT AND RELATED WORK

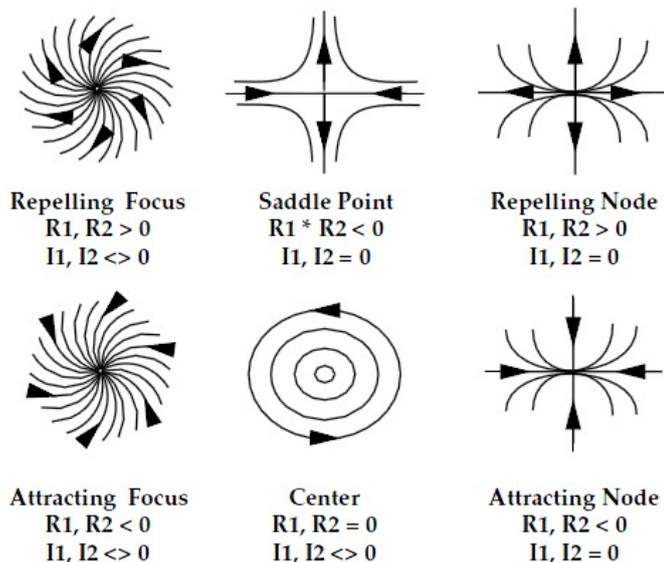


Figure 3.10: Vector field topology: critical points classified by the eigenvalues of the Jacobian. $R1$ and $R2$ correspond to the real parts of the eigenvalues, whereas $I1$ and $I2$ correspond to the imaginary parts. Figure taken from Post, Vrolijk et. al. [35].

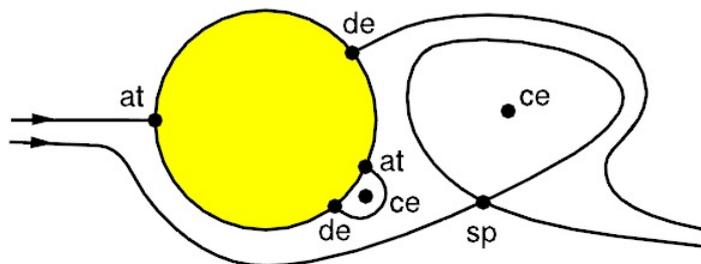


Figure 3.11: Vector field topology: a topological skeleton of a flow around a cylinder, where at: attachment, ce: center, de: detachment and sp: saddle point. Figure taken from Post, Vrolijk et. al. [35]

analysing 2-Dimensional flow field data using the concept of complex invariant moments generalized to vector-valued functions:

$$c'_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x + iy)^p (x - iy)^q f(x, y) dx dy \quad (3.28)$$

3.4 Related Work on Pattern Recognition in Fluid Flows

where $f : \mathbb{R}^2 \rightarrow \mathbb{C} \simeq \mathbb{R}^2$ be a map from \mathbb{R}^2 with $f \neq 0$, $(p + q)$ is the order of f with $p, q \in \mathbb{N}$ and $i = \sqrt{-1}$. By application of the binomial theorem complex moments of arbitrary order can be represented as linear combinations of regular moments:

$$c'_{pq} = \sum_{j=0}^p \sum_k^q \binom{p}{j} \binom{q}{k} (-1)^{q-k} i^{p+q-j-k} m_{j+k, p+q-j-k} \quad (3.29)$$

with m being the regular moment:

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy \quad (3.30)$$

Schlemmer et. al., derived a a basis B of order two of invariant moments, incorporating translation and scale invariance:

$$B = \{c_{01}, c_{00}c_{02}, c_{11}c_{02}, c_{10}c_{02}^2, c_{20}c_{02}^3\} \quad (3.31)$$

Using this basis they define abbreviations for their complex-valued basis elements which will be used as a similarity measure between the filter mask and the field:

$$\Psi_1 = c_{01}, \Psi_2 = c_{00}c_{02}, \Psi_3 = c_{11}c_{02}, \Psi_4 = c_{10}c_{02}^2, \Psi_5 = c_{20}c_{02}^3 \quad (3.32)$$

The function f is created by convolving, mirrored filter masks with the field enhanced by a Fast Fourier Transform. This function is then inserted in moments presented in eq.(3.41), whose computation result is used as similarity measure.

In order to describe all the vector patterns present in the field, a moment pyramid can be constructed which contains the moment values for different scales of the filter masks.

Moments in Critical point Visualization: According to Schlemmer et. al. [33], the method presented allows the observation of critical points which are considered to be a special class of patterns. Invariant moments have certain properties for critical point features , i.e., clockwise rotation, convergence, divergence, or saddles. Figure 3.12 presents the invariant moment values for proto-typical flow features as computed using the aforementioned equations.

Looking at the results, as Schlemmer mentions, we can see that only the first-order invariant moment $\Psi_1 = c_{01}$ is non-zero for most of the observed features and that rotation

3. PROBLEM STATEMENT AND RELATED WORK

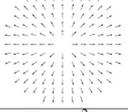
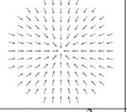
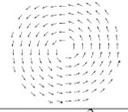
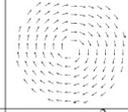
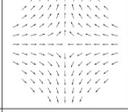
homogeneous flow	source / divergence	sink / convergence	counter-clockwise rotation	clockwise rotation	compressed rotation	saddle
						
$\Psi_1 = 0$ $\Psi_{2,3,4,5} = 0$	$\Psi_1 = \frac{2}{3\sqrt{\pi}}$ $\Psi_{2,3,4,5} = 0$	$\Psi_1 = -\frac{2}{3\sqrt{\pi}}$ $\Psi_{2,3,4,5} = 0$	$\Psi_1 = i\frac{2}{3\sqrt{\pi}}$ $\Psi_{2,3,4,5} = 0$	$\Psi_1 = -i\frac{2}{3\sqrt{\pi}}$ $\Psi_{2,3,4,5} = 0$	$\Psi_1 \approx -i0.1677$ $\Psi_{2,3,4,5} = 0$	$\Psi_1 = 0$ $\Psi_{2,3,4,5} = 0$

Figure 3.12: Invariant moment values for proto-typical flow features. Figure taken from Schlemmer et. al. [33].

patterns have a only imaginary value in c_{01} . These facts can be used to form a simple algorithmic approach for classifying these kind of features.

Advantages and Disadvantages of this method: One advantage of this method is that is invariant to translation, scaling and rotation. Also, this method is able to recognize structures that are reversed with respect to any axis. On the other hand, the need for a moment pyramid as look up table adds computational cost to this method. Another disadvantage is the need of using circular regions for our comparison in order to succeed rotation invariance limits the range of the region to be recognized.

3.4.3 Methods based on Physical Characteristics

In this class of methods, can be categorized methods which use physical characteristics of the features in order to detect them, for example the term of vorticity or the curvature characteristic in order to detect vortical structures. This methods mainly focus on the detection of vortical structures, such as vortices, and can be distinguished in two major categories:

- *Velocity-gradient based methods*
- *Curvature based methods*

Velocity-gradient based methods:

One of the physical characteristics of vortices or vortical structures is their vorticity. The magnitude of the vorticity vector ω can be used for detecting vortices and vortical flow but this classification can be false in cases concerning wall bound flows. This is due

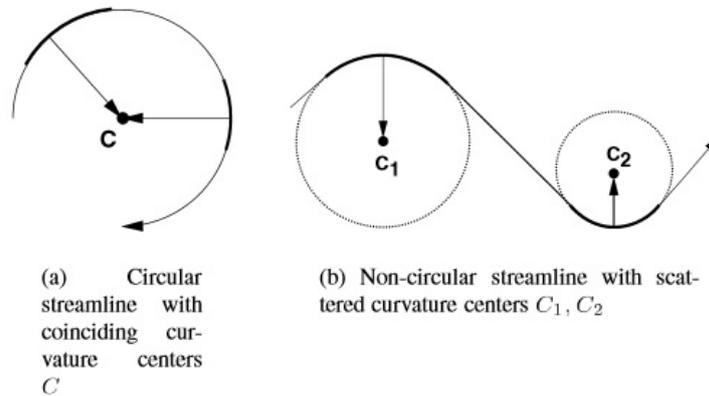


Figure 3.13: Streamlines with centers of curvature. This figure was taken from Sadarjoen et al. [38].

to the fact that $|\omega|$ does not allow the distinguish between rotation caused by shear due to topological formations and the one caused by swirling motion.

Approaches belonging to this class of methods, use velocity gradients to estimate the existence of vortices. For example, Chong et al. [37] defined a vortex core as a region where the eigenvalues of ∇u are complex, where u is the velocity field. Other approaches, make use of the second derivative for identifying the curvature of a vector field and thus, the vortex, or use the vorticity combined with wavelet transforms to estimate position and size of vortices in two-dimensional vector fields.

Drawbacks of this method: Most of the algorithms based on velocity gradients the are noise sensitive, since they use local estimates of the derivative.

Curvature based methods:

Curvature-based methods estimate curvature in the flow field in order to detect vortical flow or strongly curving flow.

One of the most known approaches using the concept of curvature to detect vortex cores is the one introduced by Sadarjoen et al. [38]. In this approach the curvature at each point of the input field is calculated and used for estimating the distance and direction to a vortex center.

3. PROBLEM STATEMENT AND RELATED WORK

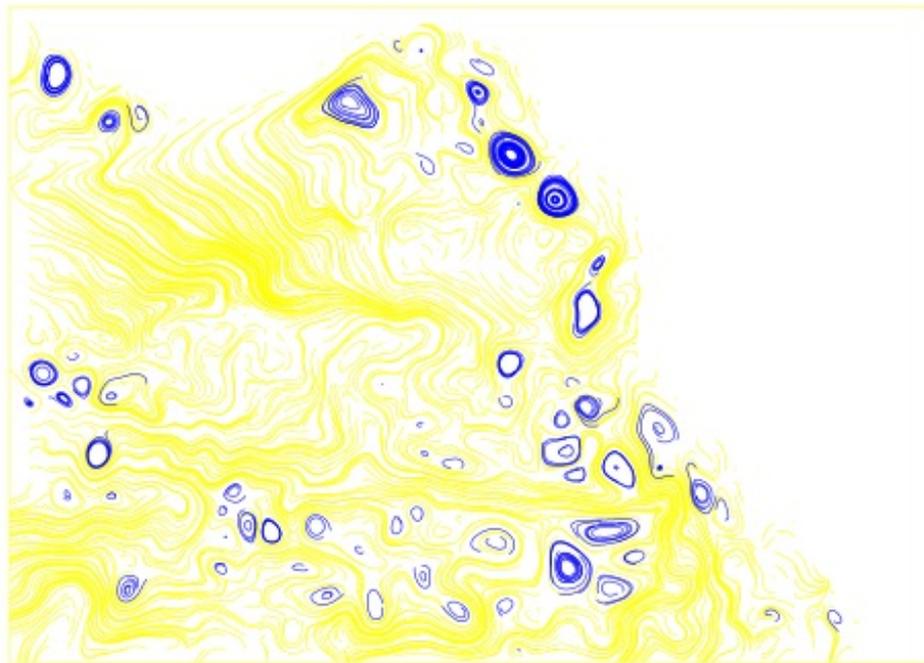


Figure 3.14: Result of this method for an image taken in the Pacific Ocean with selective streamlines, (blue) indicates the detected vortices, (yellow) other motions. This figure was taken from Sadarjoen et al. [38]

The curvature is estimated from streamline points using the differential geometric properties of streamlines. The CCD is estimated as follows: at a streamline they determine the center of curvature of its osculating circle. The curvature centers are used to denote the presence or absence of a vortex according to Sadarjoen:” *If the streamline is (nearly) circular, the curvature centers will be concentrated in a region or even coincide, which indicates the presence of a vortex(Figure 3.13a). If the streamline is not circular, the curvature centers will be scattered in space, which indicates the absence of a vortex (Figure 3.13b)*”.

The detected vortex centers are then gathered to form a curvature center distribution (CCD), which can be considered to be a vortex core likelihood. High value of curvature centers, high likelihood, in this field may indicate a vortex(Figure 3.14).

Advantages of this method: This method compared to the velocity gradient-based methods, has the advantage that is noise insensitive due to the fact that the characteri-

3.4 Related Work on Pattern Recognition in Fluid Flows

zation is not local.

Drawbacks of this method: The drawback of this approach, as Heiberg et. al. [30] presents in his review of these methods, is that in helical flow cases the curvature cannot be estimated properly as well as the position of the vortex center in cases of abrupt flows containing curved flows adjacent to vortices.

3. PROBLEM STATEMENT AND RELATED WORK

Chapter 4

Proposed Approach

In this chapter, we will present our approach for the topic of Motion Estimation in Fluid Flows, by introducing a new version of the method presented by Chang et.al. [28], which belongs to the Statistical Estimation based methods for fluid flow motion estimation. Furthermore, we will be introducing two approaches for the subject of Motion Pattern Recognition in Fluid Flows. The first is a new version for the method introduced by Heiberg et. al. [30], and belongs to the Image Processing based methods for the Motion Pattern Recognition in Fluid Flows. The other approach presented belongs also to the Image Processing class and is a simple correlation based method based on the Mean Squared Error(M.S.E.).

4.1 2-D Statistical Estimation of Fluid Flows

Our approach for the subject of Motion Estimation in Fluid Flows, is based on the Statistical Estimation approach introduced by Chang et.al. [28]. Chang's approach is the most representative method of the Statistical Estimation class and succeeds in estimating the fluid flow motion field successfully surpassing algorithms such as Lucas and Kanade's multi-scale algorithm. The algorithm can be categorized in the class of Differential Methods for Optical Flow extraction and consists of 2 basic steps: (1)The displacement probability step, in which a local distribution indicating the probability of each displacement in a local region is formulated and computed, and (2) the global optimization step, which computes a global unique velocity field, and thus, yielding a smoothed flow field.

4. PROPOSED APPROACH

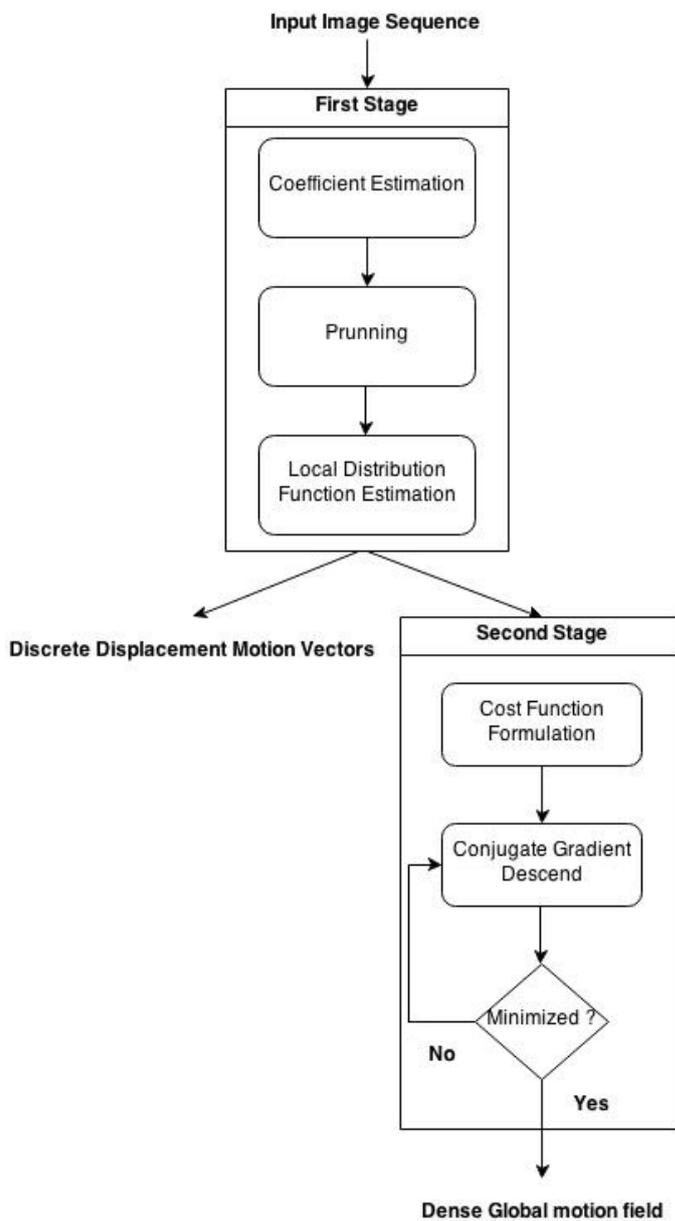


Figure 4.1: Flow diagram presenting the the two basic steps/stages of the presented approach. The first stage results in a motion field with discrete displacements whereas the second stage in a smooth global motion field, which if implemented for every pixel will result in a dense optical flow field.

These two basic steps and their components as well as their affect on the estimated motion field are represented in the form of flow diagram, in Figure 4.1. As we can see the first stage, which is the no global optimization step, calculates the transaction coefficients and thus, the local distribution functions and will result in a motion field whose motion vectors will have discrete valued magnitudes. On the other hand, the second stage, the global optimization step, uses the information of the first stage formulating a cost function whose minimization will lead to a smooth global motion field, which if computed for every pixel will result in a dense optical flow field.

What is the difference of our approach?

First of all, in the amount of data used for the motion estimation, Chang et. al. uses the spatio-temporal neighbourhood(3-Dimensional) for the estimation step, whereas our approaches uses only the spatial neighbourhood(2-Dimensional). This aspect decreases the information used as well as the computational cost but affects the quality of the estimation.

The second difference is in the global optimization step, in which a basis function is used to convert the estimation into into a differentiable function, in order to impose a global velocity field. Chang et. al. uses a tensor product of cosine functions as the basis function, whereas in our approach, a Gaussian function is selected and compared with the cosine version.

Furthermore, in order to succeed the best displacement estimation for the velocity field in the global optimization, a conjugate gradient algorithm is preferred. However, the authors unfortunately do not specify which of the existing conjugate gradient methods is used, so we had to select which algorithm to use.

Moreover, we have provided with all the theoretical background needed for the least squares process along with the connection between the coefficients and the brightness constraint equation, which were absent from Chang's approach.

Lastly, we have incorporated an additional aspect to the presented approach, the tracer identification and visualization step. Our approach supports the identification of artificial(e.g. floating balls) as well as natural(e.g. foam) tracers and the calculation of their motion field. The motion field of the tracer elements, can be associated with the motion field of the fluid flow and thus, resulting in the computation of the fluid flow motion field average velocity magnitude.

4. PROPOSED APPROACH

4.1.1 The Statistical Estimation Basis

The dynamic nature of the fluid flow, according to Chang, results in the relative positions of neighbouring points to change faster in a fluid than on a rigid body. Therefore, a stochastic fluid model can be used to determine the fluid motion field. The basis of this stochastic model, is that the flow vector at a pixel at a certain time can be considered as a random variable with a probability distribution function.

This probability distribution function at a pixel (x_s, y_s) at a certain time t can be represented as, $\phi_{x_s, y_s, t}(x_d, y_d)$, with $(x_d, y_d) \in D_s$, where D_s is the set of destination pixels, which is often a neighbourhood of the pixel (x_s, y_s) . For this distribution function we can apply that $\sum_{x_d, y_d \in D_s} \phi_{x_s, y_s, t}(x_d, y_d) = 1$.

Assuming that all the pixels belonging to the spatial region, centered at (x_s, y_s, t) , all share the same probability distribution function for their flow vectors, then the flow vector at each pixel in the local spatial region can be considered as a random sample from this distribution function¹.

Therefore, as Chang mentions, the statistics of the flow vectors in the local region can be used as an approximation to this distribution function. Then according to Chang: *"once we have the estimation of the probability distribution function, the flow vector at the center pixel (x_s, y_s) is more likely to be associated to the vector with the highest probability"*.

4.1.2 Estimation of the Distribution function

The local distribution functions can be estimated using the Spatio-Temporal Autoregressive Model(STAR), which is a version of the Autoregressive model(AR) generalized to include the spacial neighbourhood:

$$\mathbf{s}(x, y, t) = \sum_{i=1}^m A_i \mathbf{s}(x + \Delta x_i, y + \Delta y_i, t + \Delta t_i) + \mathbf{n}(x, y, t) \quad (4.1)$$

where \mathbf{s} is a state or appearance variable at time t , A_i are $r \times r$, ($\mathbf{s} \in \mathbb{R}^r$) matrix constants which characterize the sequence, $\Delta x_i, \Delta y_i$ and Δt_i specify the neighbourhood structure of the model and \mathbf{n} is drawn from a zero-mean noise distribution. AR models are used for

¹**1st Difference:**Chang et. al use the spatio-temporal region.

predicting the value of a state variable s at time t . The prediction is a linear combination of m previous values.

For the case of images, and by using the brightness constancy assumption, the state variables can be replaced with image pixel intensity. The coefficient A_i indicates the degree of correlation between $\text{pixel}(x, y)$ at time t and $\text{pixel}(x + \Delta x_i, y + \Delta y_i)$ at time $t + \Delta t_i$. By assuming a casual neighbourhood, then the coefficient A_i can be considered as proportional to the probability of the event the fluid at $\text{pixel}(x + \Delta x_i, y + \Delta y_i)$ to come from the $\text{pixel}(x, y)$, after a time interval $\Delta t_i > 0$, considering the $\text{pixel}(x, y)$ as the source of the flow:

$$\mathbf{I}(x, y, t) = \sum_{i=1}^m A_i \mathbf{I}(x + \Delta x_i, y + \Delta y_i, t + \Delta t_i) \quad (4.2)$$

where $I(x_s, y_s, t)$: image intensity at $\text{pixel}(x_s, y_s)$. To estimate the probability distribution function at a $\text{pixel}(x_s, y_s)$, we have to estimate the probability the fluid $\text{pixel}(x_s, y_s)$ actually ending up at the $\text{pixel}(x_s + \Delta x_i, y_s + \Delta y_i)$ in the next frame, so $\Delta t_i = 1$ and therefore the local probability distribution at the $\text{pixel}(x_s, y_s)$ becomes:

$$\phi_{x_s, y_s, t}(x_s + \Delta x_i, y_s + \Delta y_i) \approx A_i \quad (4.3)$$

where $(x + \Delta x_i, y + \Delta y_i) \in D_s$.

Connection between the STAR model and the gradient constraint equation:

According to the pre-mentioned information, the STAR model can express the intensity of a $\text{pixel } \vec{x} = (x_s, y_s)$ based on the probability of displacement of the pixel to a destination region in the next frame (eq. 4.2). The equation (4.2) describing this effect can be expressed as follows:

$$\begin{aligned} \mathbf{I}(x, y, t) &= \sum_D A_i \mathbf{I}(\vec{x} + \vec{v}_i, t + 1) \\ &= \sum_D A_i (\mathbf{I}(\vec{x}, t) + \nabla I \vec{v}_i + \mathbf{I}_t(\vec{x}, t)) \\ &= \sum_D A_i \mathbf{I}(\vec{x}, t) + \sum_D A_i \nabla I \vec{v}_i + \sum_D A_i \mathbf{I}_t(\vec{x}, t) \end{aligned} \quad (4.4)$$

4. PROPOSED APPROACH

Based on the fact that the coefficients A_i are actually transition probabilities for a casual neighbourhood then we have $\sum_D A_i = 1$ and equation(4.4) can now be further analysed as follows:

$$\begin{aligned} \text{Equation(4.4)} &= \mathbf{I}(\vec{x}, t) + \sum_D A_i \nabla I \vec{v}_i + \mathbf{I}_t(\vec{x}, t) \\ &= \mathbf{I}(\vec{x}, t) + \nabla I \sum_D A_i \vec{v}_i + \mathbf{I}_t(\vec{x}, t) \end{aligned} \quad (4.5)$$

where $\vec{v}_i = \left(\frac{dx_i}{dt}, \frac{dy_i}{dt} \right) = (u_i, v_i)$.

From this equation we can derive that every possible flow vector is determined by its coefficient A_i . If we assume that all the flow vectors are samples of a general family of vectors denoted by \vec{v} then eq.(4.5) becomes:

$$\begin{aligned} \mathbf{I}(x, y, t) &= \mathbf{I}(\vec{x}, t) + \nabla I \sum_D A_i \vec{v}_i + \mathbf{I}_t(\vec{x}, t) \\ &= \mathbf{I}(\vec{x}, t) + \nabla I \vec{v} + \mathbf{I}_t(\vec{x}, t) \\ &= \nabla I \vec{v} + \mathbf{I}_t(\vec{x}, t) = 0 \end{aligned} \quad (4.6)$$

We end up with the *gradient constraint equation*, which means that the coefficients A_i form an extra constraint on the family of the allowed vectors denoted by the *gradient constraint equation*.

4.1.3 Coefficient Estimation Process

As mentioned previously, all pixels in a spatial neighbourhood N_s , centered at (x_s, y_s, t) , all share the same distribution function, which means that for every pixel in N_s we get a similar equation like equation(4.2). The distribution coefficients can be solved using the system of normal equations for least-squares when the number of pixels in N_s is larger than the number of pixels in D_s .

Least Squares Method for Coefficient Computation:

Let $A = [A_1, A_2, \dots, A_D]^T$ be an $D \times 1$ array of the transition probabilities of the pixel (x_s, y_s) , to the possible displacement positions at the destination neighbourhood D_s at the next frame, and $k_s = [k_{s1}, k_{s2}, \dots, k_{sD}]^T$, k_s be a $D \times 1$ array of the pixel intensities at the possible displacement positions in the destination neighbourhood D_s for the pixel

(x_s, y_s, t) . For each pixel belonging to the spatial neighbourhood N_s , there are equivalent arrays, such as the ones described. Our goal is to find the coefficients which will give us the best estimation for the displacement probabilities.

The Least Squares criterion is expressed as follows:

$$J(A) = \sum_{s=1}^N (m_s - k_s^T A)^2 \quad (4.7)$$

with $m_s = I(x_s, y_s, t)$ denoting the intensity of pixel $(x_s, y_s) \in N_s$ and N being the number of elements in the spatial neighbourhood N_s .

Minimizing the previous equation we get:

$$\sum_{s=1}^N k_s (m_s - k_s^T A) = 0 \Rightarrow \left(\sum_{s=1}^N k_s k_s^T \right) A = \sum_{s=1}^N (k_s m_s) \quad (4.8)$$

Furthermore, if we define K as the table with all the vectors with values the intensities for all the possible positions of a transition in the destination neighbourhood D_s for each of the spatial neighbourhood N_s , centered at (x_s, y_s) , and M as the table with the intensities of the pixels in the spatial neighbourhood N_s :

$$K = \begin{bmatrix} k_1^T \\ k_2^T \\ \vdots \\ k_N^T \end{bmatrix} = \begin{bmatrix} k_{1,1} & k_{1,2} & \cdots & k_{1,D-1} & k_{1,D} \\ k_{2,1} & k_{2,2} & \cdots & k_{2,D-1} & k_{2,D} \\ \cdots & \cdots & \ddots & \cdots & \cdots \\ k_{N-1,1} & k_{N-1,2} & \cdots & k_{N-1,D-1} & k_{N-1,D} \\ k_{N,1} & k_{N,2} & \cdots & k_{N,D-1} & k_{N,D} \end{bmatrix} \quad M = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_N \end{bmatrix} \quad (4.9)$$

where $k_{s,i} = I(x_s + \Delta x_i, y_s + \Delta y_i, t + 1)$ is the intensity at the destination pixel $(x_s + \Delta x_i, y_s + \Delta y_i) \in D_s$ at the next frame, D is the number of elements in the destination neighbourhood D_s and N is the number of elements at the spatial neighbourhood N_s .

Based on the above, equation(4.8) becomes:

$$(K^T K) A = K^T M \Rightarrow A = (K^T K)^{-1} K^T M \quad (4.10)$$

The solution of equation(4.10) yields the coefficients A_i , and thus, the transition probabilities of the destination neighbourhood D_s for the pixel (x_s, y_s) .

However, as Chang denotes, the least squares method does not guarantee that $\sum_{i=1}^D A_i = 1$ then the local distribution function for a given pixel will be equal to the transition probability for the pixel in the next frame, to the sum of the transition probabilities for the

4. PROPOSED APPROACH

destination neighbourhood:

$$\phi_{x_s, y_s, t}(x_s + \Delta x_i, y_s + \Delta y_i) \approx \frac{A_i}{\sum_{j=1}^D A_j} \quad (4.11)$$

4.1.4 Coefficient Pruning

Due to the random and noisy nature of the motion of fluids we need to have a good estimation set for the parameters of the local distribution function to get the best fitting for the input data.

To improve the fitting data rates, Chang et. al. [28] suggest the application of a pruning process to the coefficients setting the coefficients that are not needed to zero. The pruning is based on Schwarz's Bayesian Criterion (SBC) which is implemented along with a binary search algorithm in order to find the number of non-zero coefficients. The number of non-zero coefficients is constantly changing until the SBC criterion stops minimizing:

$$SBC = |\Omega| \ln \sigma_a^2 + p \ln |\Omega| \quad (4.12)$$

where $|\Omega|$ is the data set size, p is the number of coefficients and σ_a^2 is the estimated innovation variance. Algorithm 1 describes the pruning process.

For the computation of the SBC the estimated innovation variance is expressed as follows:

$$\sigma_a^2 = \frac{1}{|\Omega|} \sum_{i=1}^{|\Omega|} (x_i - \bar{x}_i)^2 \quad (4.13)$$

where x_i is the intensity of the pixel, \bar{x}_i is the estimated intensity for the pixel computed as follows:

$$\bar{x}_i = \frac{\sum_{j=1}^D A_j I(x_i + \Delta x_j, y_i + \Delta y_j, t + 1)}{D} \quad (4.14)$$

in which D is the number of the coefficients.

Data: Coefficient array A
Result: Pruned coefficient array A^{pr}
Sort the array into descending form;
Calculate the SBC value, sbc_0 ;
Set half of the coefficients with smallest magnitude to zero;
Execute least-squares again for the remaining coefficients to obtain a new SBC value, sbc_1 ;
if $sbc_0 > sbc_1$ **then**
 while $sbc_0 > sbc_1$ **do**
 Set $sbc_0 = sbc_1$;
 Set half of the coefficients with smallest magnitude to zero;
 Execute least-squares again for the remaining coefficients to obtain a new SBC value, sbc_1 ;
 end
 Restore the coefficients set previously to zero;
 The non-zero coefficients are the output array A^{pr} ;
else
 Restore all the coefficients set to zero;
 Set half of the coefficients with biggest magnitude to zero;
 Execute least-squares again for the remaining coefficients to obtain a new SBC value, sbc_1 ;
 if $sbc_0 > sbc_1$ **then**
 while $sbc_0 > sbc_1$ **do**
 Set $sbc_0 = sbc_1$;
 Set half of the coefficients with smallest magnitude to zero;
 Execute least-squares again for the remaining coefficients to obtain a new SBC value, sbc_1 ;
 end
 Restore the coefficients set previously to zero;
 The non-zero coefficients are the output array A^{pr} ;
 else
 The coefficients does not need to be pruned;
 end
end

Algorithm 1: Pruning Coefficient Algorithm

4. PROPOSED APPROACH

4.1.5 Individual Motion Vector

So far, we have computed the local distribution function which indicates the transition probability for each possible transition to a destination neighbourhood D_s for the pixel (x_s, y_s) of the spatial neighbourhood N_s .

This information can be used, even at this point, to extract the motion vector of the pixel. We can simply assume as the transition destination the position associated to the highest probability (coefficient with the largest value). The motion vector is extracted by simply taking the difference between the coordinates of the pixel's position at the current frame and the coordinates of the selected position in the destination neighbourhood.

4.1.6 Global Optimization

The motion field of a liquid actually shows a unique velocity vector everywhere, meaning that each pixel should be characterized by a single velocity vector which can be extracted using the information of distribution functions we found. Essentially, all we need is a general optimization framework which finds simultaneously the velocity vectors for all pixels (and thus, a global motion field), choosing one from all the conceivable ones, that best describes the motion of the pixel.

According to Chang's approach this can be accomplished by using a Bayesian framework which allows the calculation of a posterior distribution for each pixel through which we can choose the appropriate distribution that will give us the velocity vector best describing the pixel's movement. This velocity vector shall be obtained by maximizing this posterior distribution. The maximization of the posterior distribution is equivalent to the minimization of the following cost function:

$$CostFunction = - \sum_s \log (\bar{\phi}_{x_s, y_s, t}(x_s + \Delta x_s, y_s + \Delta y_s)) + \lambda \sum_s (\|\mathbf{u}_x(x_s, y_s)\|^2 + \|\mathbf{u}_y(x_s, y_s)\|^2) \quad (4.15)$$

where $\mathbf{u}_x(x_s, y_s) = \mathbf{u}(x_s + 1, y) - \mathbf{u}(x_s, y_s)$, $\mathbf{u}_y(x_s, y_s) = \mathbf{u}(x_s, y + 1) - \mathbf{u}(x_s, y_s)$, \mathbf{u} is the global velocity vector, λ is a smoothing coefficient and $\bar{\phi}$ is a differentiable version of the previously discrete distribution function described as follows:

$$\bar{\phi}_{x_s, y_s, t}(x_s + \Delta x_s, y_s + \Delta y_s) = \sum_i [\phi_{x_s, y_s, t}(x_s + \Delta x_i, y_s + \Delta y_i) * h(\Delta x_s - \Delta x_i, \Delta y_s - \Delta y_i)] \quad (4.16)$$

in which $(x_s + \Delta x_i, y_s + \Delta y_i) \in D_s$ and $h(x, y)$ is a differential basis function which satisfies these conditions: i) $h(0, 0) = 1$, ii) $h(x, y) = 0$ when $|x| \geq 1$ or $|y| \geq 1$, iii) $h_x(0, 0) = 0$ and $h_y(0, 0) = 0$, iv) $h_x(x, y) = 0$ if $|x| = 1$ or $|y| = 1$ and $h_y(x, y) = 0$ if $|x| = 1$ or $|y| = 1$.

In order to avoid landing on a local minima for the cost function minimization process we need to find the optimal solution. To achieve this, as Chang proposes and implements, a heuristic optimization procedure can be used which combines a conjugate gradient algorithm with a variable support region for the basis function $h(x, y)$. The support region for the basis function shirking at constant rate during the minimization process until the support region reaches size 1.

As for the basis function $h(x, y)$, in our approach we used a 2-Dimensional Gaussian function instead of the tensor product of cosine functions used by Chang et. al. [28]¹. The Gaussian function and its partial derivatives used in this method are expressed as follows:

2-D Gaussian distribution function:

$$h(x, y) = A_{con} \cdot \exp \left(- \left(\frac{(x - x_o)^2}{2\sigma_x^2} + \frac{(y - y_o)^2}{2\sigma_y^2} \right) \right) \quad (4.17)$$

which can also be expressed as:

$$h(x, y) = A_{con} \exp \left(- (a \cdot (x - x_o)^2 + 2 \cdot b \cdot (x - x_o) \cdot (y - y_o) + c \cdot (y - y_o)^2) \right) \quad (4.18)$$

where x_o, y_o is the center of the distribution, σ_x^2 and σ_y^2 are variances computed from the support region, A_{con} is a constant and

$$a = \frac{\cos^2 \vartheta}{2 \cdot \sigma_x^2} + \frac{\sin^2 \vartheta}{2 \cdot \sigma_y^2}, \quad b = -\frac{\sin^2 2\vartheta}{4 \cdot \sigma_x^2} + \frac{\sin^2 \vartheta}{4 \cdot \sigma_y^2}, \quad c = \frac{\sin^2 \vartheta}{2 \cdot \sigma_x^2} + \frac{\cos^2 \vartheta}{2 \cdot \sigma_y^2} \quad (4.19)$$

in which ϑ is the rotation angle, which is zero in our approach. The partial derivatives of equation(4.18) are expressed as follows:

$$\frac{\partial h}{\partial x_{dir}} = A_{con} \cdot \exp \left(- (a \cdot (x_{dir})^2 + 2 \cdot b \cdot (x_{dir}) \cdot (y_{dir}) + c \cdot (y_{dir})^2) \right) \cdot (- (2 \cdot a \cdot x_{dir} + 2 \cdot b \cdot y_{dir})) \quad (4.20)$$

¹**2nd Difference:** Chang et. al use a tensor product of cosine functions, which when tested in our approach did not yield as good results as the 2-D Gaussian function.

4. PROPOSED APPROACH

$$\frac{\partial h}{\partial y_{dir}} = A_{con} \cdot \exp\left(-\left(a \cdot (x_{dir})^2 + 2 \cdot b \cdot (x_{dir}) \cdot (y_{dir}) + c \cdot (y_{dir})^2\right)\right) \cdot \left(-\left(2 \cdot c \cdot y_{dir} + 2 \cdot b \cdot x_{dir}\right)\right) \quad (4.21)$$

where $x_{dir} = x - x_o$ and $y_{dir} = y - y_o$.

The role of the support region Gaussian function:

The support region in the Gaussian function determines the dispersion function and indirectly the range of displacements. Decreasing the support region and thus reducing the dispersion values will result also on a limitation of the range of the displacement values. This limitation will give a homogeneity in the motion vectors describing the flow field.

4.1.7 Conjugate Gradient Method

The elements of interest, however, which are substantially the parameters distinguishing the value of the cost function, are displacement values Δx_s and Δy_s . The goal is to find the best estimates for these parameters. This will result in the minimization of the cost function yielding the best estimated motion vectors describing the flow field.

The minimization of the cost function is achieved, as mentioned earlier, by using a Conjugate Gradient algorithm, which at each iteration calculates the cost function and updating, according to the result, the motion vector used to describe the pixel motion. In our approach¹, we have used the Standard Conjugate Gradient algorithm for Nonconvex problems, as presented by R. Pytlak [39].

The purpose of the Conjugate Gradient method is the optimization problem of minimizing a continuous differential equation:

$$\min \{f(x) : x \in \mathbb{R}^n\}$$

where $f : \mathbb{R}^n \mapsto \mathbb{R}$ is a differential function which in our case is the cost function. A non-linear conjugate gradient method generates a sequence $x_k, k \geq 1$ starting from an initial hypothesis $x_1 \in \mathbb{R}^n$ using the recursion:

$$x_{k+1} = x_k + \alpha_k d_k$$

¹**3rd Difference:** Chang et. al use a fast conjugate gradient algorithm, but unfortunately they do not provide information on which of the versions existed was used.

with α_k being a positive step value, found using *line search* and d_k is the search direction defined as:

$$d_{k+1} = -g_{k+1} + \beta_k d_k, \quad d_1 = -g_1$$

where β_k is the update parameter of the CG and $g_k = \nabla f(x_k)^T$ with $\nabla f(x_k)^T$ being the function derivative in x_k . In this case, in the position x_k of the sequence we can define the displacement sequences Δx_k and Δy_k with update functions:

$$\begin{aligned} \Delta x_{k+1} &= \Delta x_k + \alpha_k d_{x_k} \\ \Delta y_{k+1} &= \Delta y_k + \alpha_k d_{y_k} \end{aligned}$$

in which d_{x_k}, d_{y_k} are search directions derived by taking the partial derivative of the cost function with respect to the directional components x, y :

$$\begin{aligned} d_{x_{k+1}} &= -g_{x_{k+1}} + \beta_k d_{x_k} \\ d_{y_{k+1}} &= -g_{y_{k+1}} + \beta_k d_{y_k} \end{aligned}$$

Moreover, the update parameter of the CG, β , is derived from the first non-linear method CG, suggested by Fletcher and Reeves:

$$\beta_k = \frac{\|g_{k+1}\|^2}{\|g_k\|^2}$$

The algorithm used incorporates the *line search* to update the value of α and is presented in Algorithm 2.

When the algorithm finishes, we will have calculated the optimum α_k and so we would have found the optimal displacements which minimize the cost function. The update process in the algorithm is repeated as long as the Cost Function Minimization

4. PROPOSED APPROACH

Condition is satisfied.

Step 1: Choose an arbitrary $x_1 \in \mathbb{R}$ and set $d_1 = -g_1$, $k = 1$;

Step 2: Find an α_k satisfying the "strong" Wolfe restrictions by setting an upper limit for the value of α_k , which cannot satisfy the Cost Function Minimization condition:

Wolfe's "strong" restrictions:

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \mu \alpha_k g_k^T d_k \quad (1)$$

$$|g(x_k + \alpha_k d_k)| \leq \eta |g_k^T d_k| \quad (2)$$

Minimization Condition:

$$f(x_k + \alpha_k d_k) \leq f(x_k)$$

If the Wolfe's restrictions and minimization condition are satisfied then we increase the α_k and repeat Step 2. ;

Step 3: If $\|g_{k+1}\| = 0$ then we stop and the current α_k is the best estimate.

Otherwise, we compute the update parameter, β_k , and we continue to the update along the search direction, $d_{x_{k+1}}$ and $d_{y_{k+1}}$;

Step 4: Increase k by 1 and go to Step 2;

Algorithm 2: The standard conjugate gradient algorithm for nonconvex problems

An additional Restriction in Algorithm 2:

Despite the restrictions used in the CG algorithm, we added an additional restriction that the displacement values must be less or equal to the size of the neighbourhood destination . This was done due to the fact that while the displacements of the pixels had values within these limits, there were 1-3 pixels whose displacement values were exceeding the destination neighbourhood limits. This is probably due to the rate of increase of the coefficient, α_k , which may have resulted in overcoming the convergence point and so the minimization of the cost function was a result of a local minimum position.

4.1.8 Tracer Identification and Motion Visualization

The approach presented can be adapted to tracer identification and motion extraction. Tracers, as mentioned in Chapter 3.2.3, are either artificial objects, such as balls, or natural things, for example leaves, that are used to extract the motion field of the fluid flow.

The advantages of tracers in fluid flows are several, such as the minimization of the number of motion vectors of interest. The less data we have to examine the less computational time is spent on the extraction of the motion field.

Also, due to the fact that fluids tend to have specific range of intensity values, correlation-based techniques may be affected while comparing values of pixels belonging to the fluid which will result in bad motion estimation results. Tracers are region limited, with specific intensity values and with a specific pattern. These characteristics result in more robust estimation of the motion field.

On the other hand, the tracers used must be selected carefully so that they can be distinguished from the fluid flow. Also, their motion must resemble the motion of the fluid in order to extract the motion field of the fluid flow.

Tracer identification

Tracers, as mentioned earlier, must be distinctive in order to be identified from the flow field. In our method, we use the simple concept of intensity variation between a tracer and a fluid flow. The only condition, in order to identify a tracer in the flow field, is the tracer to have higher intensity values compared to the fluid. For example, white foam can be used as a tracer pattern, its intensity values are greater compared to fluids, such as water in rivers or streams. Foam has close to white, colour patterns while water tends more to black(in grayscale images), in image processing this can be translated as follows: *white-like colour patterns correspond to higher intensity values whereas black-like colour patterns correspond to lower intensity values.*

We can use a simple thresholding approach to identify the pixels which potentially consist the tracer, along with a condition which identifies a tracer pattern.

The threshold used is expressed as follows:

$$T = \frac{\text{Highest intensity value} - \text{Average intensity value}}{\alpha_{normal}}$$

4. PROPOSED APPROACH

where T is the threshold value and α_{normal} is a normalization constant, $\alpha_{normal} \in [1, \infty)$.

The condition the pixel to belong to the tracer pattern is expressed as follows: If the number of pixels in the spatial neighbourhood in which the pixel belongs, having an intensity greater or equal to the threshold value, is greater than the total number of pixels belonging in the spatial neighbourhood divided by 2, then the pixel neighbourhood corresponds to a tracer.

4.2 2-D Flow Characterization Using Vector Pattern Matching

The first approach on the topic of Motion Pattern Recognition in Fluid Flows, is based on the methodology presented by Heiberg et. al. [30]. Their method is based on Vector Pattern Matching using a set of idealized patterns for each structure type to be classified. Similarity to these patterns is defined as a scalar convolution on vector fields based on the scalar product of vectors. Based on the convolution result an outer product tensor is constructed whose eigenvectors and eigenvalues denote the degree of correlation between the structures and the patterns.

What is the difference of our approach?

First of all, in the amount of information used for the pattern recognition step. Heiberg et. al. use in their approach a 3-Dimensional vector field as well as 3-Dimensional idealized patterns to classify the vector field. In our approach the input vector field is 2-Dimensional as well as the patterns used, which decreases the computational cost.

However, the use of the 2-Dimensional space along with Heiberg's methodology has not been tested thoroughly using field data, only the theoretical background exists. Our goal was to examine whether the implementation of the theory on fluid flows, such as rivers or stream, would yield satisfying results.

The second difference is that our approach is focused only on the identification of vortical and plane wave homogeneous flows, whereas Heiberg's method is able to identify also swirling flow and diverging or converging flow. We chose to use only the most basic

motions to simplify our search and make sure our approach is accurate enough. Other flow types can be included in the future.

Finally, the most obvious difference is the field data used. Heiberg's method is implemented for identifying blood flow types in the human heart, whereas our approach is implemented for fluid flows in natural phenomena, such as rivers and streams. This shows that this method can be used in various fields of science.

Why to select this method?

As mentioned in Chapter 3.4.1, Heiberg's approach has many advantages. The most notable one is that it is robust in terms of noise sensitivity due to the fact that it is regional. Also, it is independent of the orientation of the structure and can be adapted to a variety of search patterns. Lastly, as mentioned earlier it can be used in various scientific fields, such as medicine (as presented by Heiberg), aerodynamics and hydrology.

Method Formulation:

The method presented can be formulated into five basic steps which are summarized as follows: (1) Normalize the vector field, (2) Create the set of filters for the structures we want to look for, (3) Convolve with each filter, (4) Create tensor field and (5) Calculate eigenvalues of each tensor in the tensor field and take largest eigenvalues as similarity output.

4.2.1 Normalization of the vector field

The first step of the methodology, is to normalize the vector field, so that all the vectors in a vector field have the same magnitude. The normalization step serves as we want to look for topological similarities instead of similarity between velocity distributions. The normalized input vector field, $\hat{W}(\vec{x})$, is expressed as follows:

$$\hat{W}(\vec{x}) = \frac{W(\vec{x})}{|W(\vec{x})|}$$

where $W(\vec{x})$ is the original vector field and $\vec{x} = (x, y)$.

4. PROPOSED APPROACH

4.2.2 Filter Creation

As mentioned previously, the aim of the algorithm is to find a topological similarity between the reference pattern and the motion field. To accomplish this, the filters must be sensitive to the direction and set in the patterns of interest, which in our case are vortical flow patterns or homogeneous flow patterns. These, as Heiberg has observed, are precisely the qualities found in Quadrature Filters.

Quadrature Filters: The Quadrature Filters are filters with a zero transfer function in one half plane of the frequency domain. An important property of the Quadrature filter is that both the filter and the filter output is complex valued where the real part corresponds to a line/plane detector and the imaginary part to an edge detector. On the non zero half plane of the filter, a suitable directional function needs to be chosen. Quadrature filters can be expressed as follows:

$$\begin{cases} F_k(u) > 0 & \text{if } u \cdot n_k > 0 \\ F_k(u) = 0 & \text{otherwise.} \end{cases}$$

where u is the frequency and n_k is the unit orientation vector of the filter k .

The frequency response of the quadrature filter when $u \cdot n_k > 0$ is:

$$F_k(u) = g(\|u\|) \cdot (u \cdot n_k)^2 \quad (4.22)$$

In cases where the input signal is related to neighbourhoods which are expressed in the form:

$$\xi(s) = f(s \cdot n_\xi)$$

with s being the spatial coordinates and n_ξ is the unit orientation vector, the Fourier transform is non-zero only in the line defined by:

$$u \propto n_\xi$$

And then for equation(4.22), according to the theory of filters, we will take as filter output:

$$\|F_k\| = d \cdot (n_\xi \cdot n_k)^2 \quad (4.23)$$

4.2 2-D Flow Characterization Using Vector Pattern Matching

where d is a constant independent from the orientation of the filter, n_ξ is the unit orientation vector of the input neighbourhood vector field and n_k is the unit orientation vector of the filter k .

We can observe from equation(4.23) that the output of the quadrature filter depends on the orientation of the input and the orientation of the filter. This satisfies our need for topological comparison in form of vector orientation between the reference pattern and the comparison structure.

4.2.2.1 Vortical Flow Filters

The filters representing vortical and turbulent flow, based on the theory of Quadrature filters, are expressed as follows:

$$F_k(\vec{x}) = \gamma \cdot G(\vec{x}) \cdot \varphi(r) \quad (4.24)$$

where G is a function describing the pattern, φ is a weight function which acts as a spatial limitation function and γ is a normalization constant so that $(F_i, F_i) = 1$.

More specifically we have,

- *Weight Function φ :*

$$\varphi(r) = \begin{cases} 1, & |r| < R \\ e^{-\sigma(r-R)^2}, & |r| \geq R. \end{cases} \quad (4.25)$$

where R and σ determine the size of the pattern and $r = \sqrt{\vec{x} \cdot \vec{x}}$, with $\vec{x} = (x, y)$.

- *Pattern Function G :*

$$G(\vec{x}) = n_k \times \vec{x} \quad (4.26)$$

with n_k being the unit orientation vector of the filter, indicating the symmetry axis of the filter.

Thus, equation(4.22) which describes the filter for vortical and turbulent flow can be written as:

$$F_k(\vec{x}) = \gamma \cdot \varphi(r) \cdot n_k \times \vec{x} \quad (4.27)$$

4. PROPOSED APPROACH

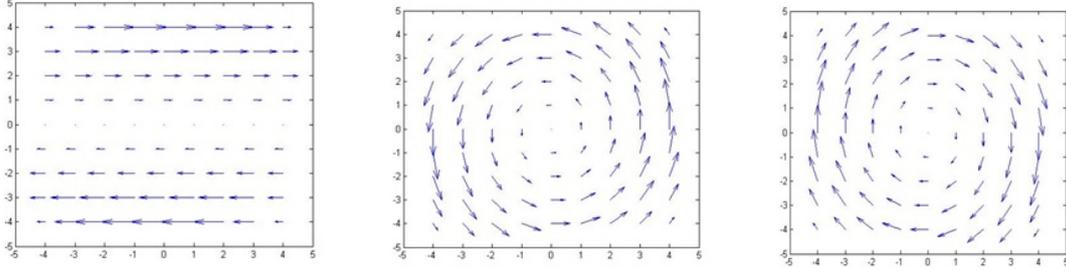


Figure 4.2: The reference filter patterns used for identifying vortical and turbulent flow, (left) filter with orientation vector n_1 , (center) with orientation vector n_2 and (right) with orientation vector n_3 .

The filters used for the identification of vortical flows can be seen in the Figure 4.2. The filters used have the following unit orientation vectors:

$$n_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad n_2 = \begin{pmatrix} \frac{-1}{2} \\ \frac{\sqrt{3}}{2} \end{pmatrix}, \quad n_3 = \begin{pmatrix} \frac{1}{2} \\ \frac{-\sqrt{3}}{2} \end{pmatrix}$$

The reason these specific unit orientation vectors for the filters have been selected as well as the number of filters used, is explained in the following subsection(4.2.3).

4.2.2.2 Plane Wave Homogeneous Flow Filters

The filter equation used to describe the plane wave homogeneous flow is similar to equation(4.27). The difference is in the role of the weight function and the spatial coordinates.

The role of the weight function is to form the motion pattern by giving appropriate weights to the motion vector of each element and also limiting it spatially. In homogeneous flows all vectors are distributed evenly in space and in parallel directions. Each element will move in a straight direction to the original position and the direction will be parallel to that of the other elements that constitute the structure. Therefore, all the elements should be associated with a weight according to its spatial coordinates so that we can form a homogeneous flow pattern. The equation describing the filter, as Andreas Andersson describes in his master thesis [44], can now be expressed as:

$$F_{k_{homogeneous}}(\vec{x}) = \psi(\vec{x}) \cdot n_k \quad (4.28)$$

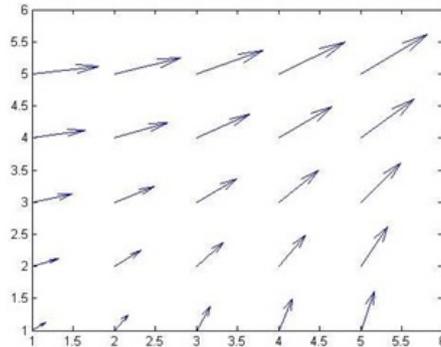


Figure 4.3: Example of a filter pattern describing a homogenous flow pattern with orientation vector n_{homog} .

An example of a homogeneous filter pattern can be seen in Figure 4.3, whose unit orientation vector is:

$$n_{homog} = \begin{pmatrix} \frac{2}{\sqrt{3}} \\ \frac{\sqrt{3}}{2} \end{pmatrix}$$

As will be presented in the following subsection, in order to identify a plane homogeneous flow we can only use the filter response in the case of vortical flow patterns.

4.2.3 Filter Convolution

The convolution between the normalized field motion \hat{W} and the filter F_n describing the reference pattern is defined as:

$$h_n(x) = \int_{\mathbb{R}^2} \hat{W}(x) F_n(x - \xi) d\xi$$

where n is the symmetry axis of the filter.

Filter response:

Andreas Andersson [44], in his thesis, proved that the response of vortical filters for the case of vortical flows will be proportional to the scalar product between the orientation of the vortex core filter and the orientation of the vortical flow pattern, whereas for the

4. PROPOSED APPROACH

case of plane flows will be zero. Based on his work we can denote the following filter response equations:

For the case where the input is an ideal vortical flow field, the flow field $W_{\hat{a}}$ is defined as:

$$W_{\hat{a}}(x) = \psi(r)\hat{a} \times x$$

with $\hat{a} \in \mathbb{R}^2, \hat{a} \neq 0$ and $\psi : \mathbb{R}_+ \mapsto \mathbb{R}_+$.

The filter equation for the case of vortical flow identification is expressed as:

$$F_{\hat{n}}(x) = \phi(r)\hat{n} \times x$$

with $\phi : \mathbb{R}_+ \mapsto \mathbb{R}_+$ is a radially symmetric decay function, as presented in equation(4.25).

The convolution between these two vector fields is defined as follows:

$$\begin{aligned} (F_{\hat{n}}, W_{\hat{a}}) &= \int_{\mathbb{R}^2} F_{\hat{n}}(x)W_{\hat{a}}(x)dx \\ &= \int_{\mathbb{R}^2} \phi(r)\psi(r) (\hat{a} \times x) \cdot (\hat{n} \times x) dx \end{aligned} \quad (4.29)$$

The term $(\hat{a} \times x) \cdot (\hat{n} \times x)$ can be expressed as¹ :

$$\begin{aligned} (\hat{a} \times x) \cdot (\hat{n} \times x) &= ((\hat{a} \times x) \times \hat{n}) \cdot x \\ &= ((\hat{a} \cdot \hat{n})x - (x \cdot \hat{n})\hat{a}) \cdot x \\ &= (\hat{a} \cdot \hat{n})x \cdot x - (x \cdot \hat{n})(\hat{a} \cdot x) \\ &= |x|^2 (\hat{a} \cdot \hat{n}) - \hat{n}^T x x^T \hat{a} \end{aligned}$$

Using the above, equation(4.29) is expressed as:

$$(F_{\hat{n}}, W_{\hat{a}}) = \int_{\mathbb{R}^2} \phi(r)\psi(r)|x|^2 dx (\hat{a} \cdot \hat{n}) - \hat{n}^T \int_{\mathbb{R}^2} \phi(r)\psi(r) x x^T dx \hat{a} \quad (4.30)$$

The second integral of the equation is actually 2-by-2 matrix M where :

$$x x^T = \begin{bmatrix} x \\ y \end{bmatrix} \begin{bmatrix} x & y \end{bmatrix} = \begin{bmatrix} x^2 & xy \\ yx & y^2 \end{bmatrix} \quad (4.31)$$

¹Gunnar Sparr [40], in Theorem 6 in Chapter 5 of "Linjar Algebra", proved that $(a \times b) \times c = (a \times c)b - (b \times c)a$.

4.2 2-D Flow Characterization Using Vector Pattern Matching

Due to symmetry of the integrand (the patterns used must be axis symmetric) it follows that:

$$\int_{\mathbb{R}^2} \phi(r)\psi(r)xydx = \int_{\mathbb{R}^2} \phi(r)\psi(r)yxdx = 0$$

whereas, for the diagonal elements we have:

$$\int_{\mathbb{R}^2} \phi(r)\psi(r)x^2dx = \int_{\mathbb{R}^2} \phi(r)\psi(r)y^2dx = \frac{1}{2} \int_{\mathbb{R}^2} \phi(r)\psi(r)|x|^2dx$$

And thus, equation(4.30) can be defined as:

$$\begin{aligned} (F_{\hat{n}}, W_{\hat{a}}) &= \frac{1}{2} \int_{\mathbb{R}^2} \phi(r)\psi(r)|x|^2dx(\hat{a} \cdot \hat{n}) \\ &\propto \hat{n} \cdot \hat{a} \end{aligned} \quad (4.32)$$

Thus, we find that the result of the convolution of the vortex core filter $F_{\hat{n}}$ and the idealised vortical flow field $W_{\hat{a}}$ is proportional to the scalar product between the orientation of the vortex core filter and the orientation of the idealised vortical flow pattern.

The vortex core filter should yield a zero response for non vortical flow fields, such as plane waves of homogeneous flow fields. As mentioned earlier, Andersson [44] has presented that a plane wave homogeneous flow is described by equation(4.28):

$$F_{k_{homogeneous}}(\vec{x}) = \psi(\vec{x}) \cdot \hat{a}_k$$

In this case, the filter response will be zero:

$$\begin{aligned} (F_{k_{homogeneous}}, W_{\hat{a}}) &= \int \phi(r)\psi(x) (\hat{n} \times x) \cdot \hat{a}_k dx \\ &= \int \phi(r)\psi(x) (\hat{n} \times \hat{a}_k) \cdot x dx \\ &= (\hat{n} \times \hat{a}_k) \int \phi(r)\psi(x) x dx \\ &= (\hat{n} \times \hat{a}_k) \cdot 0 = 0 \end{aligned}$$

As we can see, only by using the vortical flow filters output we can detect, apart from the vortical flow fields, the homogeneous plane wave fields.

4. PROPOSED APPROACH

4.2.4 Outer Product Tensor

Instead of convolving with each vortical flow filter and examining their outputs, we can generalize the detection process, according to Heiberg's approach, by creating a tensor of orientation whose eigenvalues and eigenvectors will denote the presence of a vortical flow field or a homogeneous one.

What is a Tensor of orientation:

A tensor is a table representation of the vector space and is used to represent the direction of a vector in the vector space. Every vector \vec{x} is characterized by sub-vectors with coordinates x_i and their association with a vector space that can be expressed in the form:

$$\vec{x} = \sum_{j=1}^n T_{ij} x_j$$

The association of a vector \vec{x} to a tensor is:

$$T = r^{-1} \vec{x} \vec{x}^T$$

where $r = \sqrt{\vec{x} \cdot \vec{x}}$. The tensor matrix T contains the representation in the form of directions for the coordinates in any coordinate system in vector space.

For the association of a vector in the 2-Dimensional space a symmetric tensor matrix of second order is used. A second order symmetric tensor matrix is a 2×2 matrix with three unique elements $x_{1,1}$, $x_{1,2}$ and $x_{2,2}$ (due to symmetry $x_{1,2} = x_{2,1}$). The uniqueness in a tensor translates into degrees of freedom in terms of direction.

H. Knutsson [41], proves that the minimum number of Quadrature filters for representing direction in the 2 - Dimensional space in the form of tensor matrix is 3 and, according also to Knutsson [42] will have as orientation, the following unit orientation vectors:

$$n_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, n_2 = \begin{pmatrix} \frac{-1}{2} \\ \frac{\sqrt{3}}{2} \end{pmatrix}, n_3 = \begin{pmatrix} \frac{1}{2} \\ \frac{-\sqrt{3}}{2} \end{pmatrix}$$

Knutsson [42], also shows that the tensor produced by the filter outputs is the following:

$$T = \sum_k F_k \left(n_k n_k^T - \frac{1}{m-1} I \right) \quad (4.33)$$

4.2 2-D Flow Characterization Using Vector Pattern Matching

with m being the dimensionality of the tensor T , n_k is the orientation of the filter k and F_k is the output of the filter k .

From linear algebra it is known that the dimensionality of a $n \times n$ tensor matrix is $\frac{n(n+1)}{2}$, which means that in the case of 2×2 tensor the dimensionality is 3, and so the tensor is formed as:

$$T = \sum_k F_k \left(n_k n_k^T - \frac{1}{2} I \right)$$

This equation according to Knutsson [42], is a generalization of the following equation:

$$T_{gen} = \sum_k F_k (n_k n_k^T) \quad (4.34)$$

This equation yielded better results when used in our implementation compared to the previous one.

4.2.5 Eigenvector and Eigenvalue Computation

From linear algebra it is known that every table can be expressed as a combination of products of the eigenvectors and their corresponding eigenvalues. This corresponds also to a tensor:

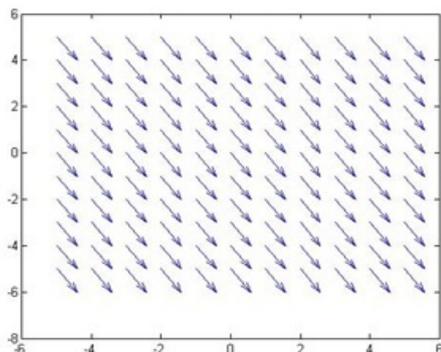
$$T = \sum_i^n \lambda_i \hat{e}_i \cdot \hat{e}_i^T$$

The fact that the tensor used is symmetrical, ensures that the eigenvalues will be real and thus, can be classified according to their width. For the case of our second class tensor the number of eigenvalues extracted is 2.

The eigenvalues of the tensor can be used as a similarity measure between the filter pattern and the input motion field. Even more, according to Heiberg et. al. [30] the corresponding eigenvectors can provide us with information about the orientation of the structure.

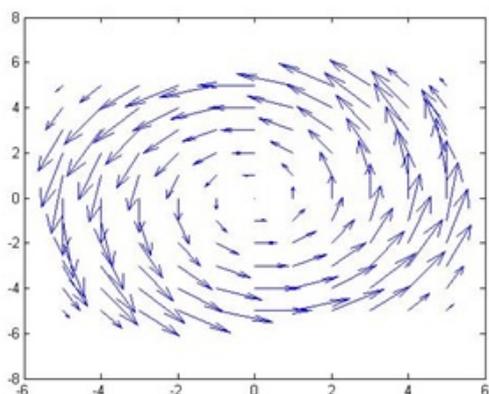
Based on this observation, we can define value intervals for the eigenvalues which will be used as a means of classification specifying whether the input motion field, shows similarity with the reference pattern or not.

4. PROPOSED APPROACH



Eigenvalues	Values
λ_1	0.000
λ_2	0.000

Figure 4.4: Field movement which shows no vortex form. The result of the eigenvalues correctly classifies the given flow pattern as non-vortical motion.



Eigenvalues	Values
λ_1	0.000
λ_2	5.1459e+08

Figure 4.5: Field movement which shows vortex form. The result of the eigenvalues correctly classifies the given flow pattern as vortical motion pattern.

The classification according to the values of the eigenvalues, as defined in our methodology, is as follows:

$$\begin{cases} \text{vortex pattern,} & \text{if } \lambda_1 \geq 0 \text{ and } \lambda_2 > 0 \\ \text{homogeneous flow pattern,} & \text{if } \lambda_1 = 0 \text{ and } \lambda_2 = 0 \\ \text{uncategorized flow patterns,} & \text{otherwise.} \end{cases}$$

The validity of the threshold used can be seen in the figures 4.4 and 4.5. The classification was made using the vortical flow filters presented earlier.

4.3 2-D Flow Characterization based on Mean Squared Error

Finally, the algorithm used is summarized and presented in algorithm 3.

Data: Flow vector field
Result: Flow Classification
Normalize the input flow field;
Create the filters representing the reference patterns, i.e. the vortical flow filters;
Convolve the filter with the input motion field for each velocity component separately;
Construct the Tensor of orientation using the convolution outputs;
Calculate the eigenvalues of the Tensor and their corresponding eigenvectors;
if $\lambda_1 \geq 0$ *and* $\lambda_2 > 0$ **then**
 | **Classify the flow field as vortex;**
else
 | **if** $\lambda_1 == 0$ *and* $\lambda_2 == 0$ **then**
 | **Classify the flow field as a homogeneous flow field;**
 | **end**
end

Algorithm 3: The Flow Characterization algorithm using Vector Pattern Matching

4.3 2-D Flow Characterization based on Mean Squared Error

The second approach presented for the topic of Flow Characterization is based on vector pattern matching using the Mean Squared Error(M.S.E.) as a means of similarity between the reference pattern and the structure.

Method Formulation:

The method introduced can be formulated into five steps also, which are summarized as: (1)Normalize the vector field, (2)Construct the reference patterns, (3)Normalize the pattern vector field, (4)Compute the M.S.E. and (5)Classify the structure based on the M.S.E. value computed between the structure and each reference pattern.

What is the difference of this approach?

The difference of this method, compared to the previous one, is that the vector field

4. PROPOSED APPROACH

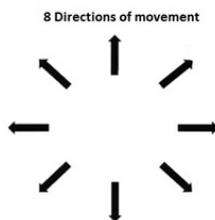


Figure 4.6: The eight sub-directions of movement for homogeneous flow.

of the reference pattern is compared to the vector field of the structure using the Mean Squared Error. The simplicity of this method allows us to look for a variety of flow patterns such as subclasses of homogeneous flow, for example rightward, upward or downward homogeneous flow.

4.3.1 Reference Pattern Creation

We can skip the presentation of the Normalization step, due to the fact that it is done the same way as the one presented in the previous subsection.

The reference patterns used are, also created in the same way as the filter flow patterns used in the previous methodology. The patterns created, consist of the vector field for both velocity components. Therefore, if normalized also, can be correlated with a structure.

For the case of vortical flow patterns, we use the patterns displayed in Figure 4.2 (the center pattern corresponding to a rightward vortex and the pattern on the right side which corresponds to a leftward vortex). The patterns, as mentioned previously, are created using equation (4.27), for each velocity component, which yields the velocity vectors.

Homogeneous flow patterns are created using equation (4.28) with $\psi(x) = x$. The homogeneous flow can be divided into eight sub-directions displayed in Figure 4.6. The factor that denotes each direction is the unit orientation vector used. The patterns associated with these sub-classes of homogeneous flow are depicted in Figure 4.7 and their corresponding unit orientation vectors are the following:

4.3 2-D Flow Characterization based on Mean Squared Error

Pattern	Unit Orientation Vector
Upward motion	$n_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$
Downward motion	$n_2 = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$
Leftward motion	$n_3 = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$
Rightward motion	$n_4 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$
Upward and Rightward motion	$n_5 = \begin{pmatrix} \frac{2}{\sqrt{3}} \\ \frac{\sqrt{3}}{2} \end{pmatrix}$
Downward and Leftward motion	$n_6 = \begin{pmatrix} -\frac{2}{\sqrt{3}} \\ -\frac{\sqrt{3}}{2} \end{pmatrix}$
Downward and Rightward motion	$n_7 = \begin{pmatrix} \frac{2}{\sqrt{3}} \\ -\frac{\sqrt{3}}{2} \end{pmatrix}$
Upward and Leftward motion	$n_8 = \begin{pmatrix} -\frac{2}{\sqrt{3}} \\ \frac{\sqrt{3}}{2} \end{pmatrix}$

4.3.2 Normalization of the Reference Vector field

During the calculation of the mean square error we take into account the magnitude of the vector. Furthermore, as mentioned previously, the vectors of the input motion field have been normalized, i.e. the vector values $\in [-1, 0, 1]$. To achieve the best comparison result we need to normalize the vector field of the reference pattern in order the result of the mean square error function to be unaffected by the velocity magnitude values.

Let $F_k(\vec{x})$ be the motion field of the reference pattern then the normalized field motion $\hat{F}_k(\vec{x})$ is denoted as follows:

$$\hat{F}_k(\vec{x}) = \frac{F_k(\vec{x})}{|F_k(\vec{x})|}$$

4. PROPOSED APPROACH

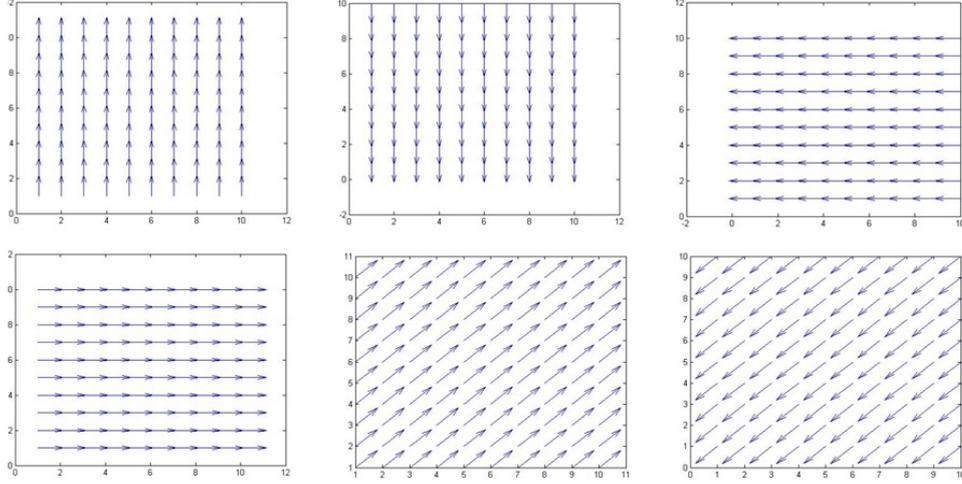


Figure 4.7: The first six of the eight possible direction patterns of homogeneous flow with normalization. The unit orientation vectors used (from up left to right order) are: (1) n_1 , (2) n_2 , (3) n_3 , (4) n_4 , (5) n_5 and (6) n_6 .

4.3.3 Mean Squared Error Calculation

The measure of similarity between the reference pattern and the motion field of the structure used in the presented method is the Mean Squared Error (Mean Squared Error) and is defined as follows:

$$MSE_k = \frac{1}{n} \cdot \sum_{i=1}^n \left(\hat{F}_{k,i} - \hat{W}_i \right)^2 \quad (4.35)$$

where n is the number of vector to be compared, \hat{W} is the normalized input vector field and \hat{F}_k is the normalized vector field of the reference pattern k .

4.3.4 Pattern Classification

The Mean Squared Error is computed for each reference pattern. The results are then sorted in ascending form and then the structure is classified to the pattern associated with the lowest M.S.E. value.

In order to make the method more accurate we use a threshold on the M.S.E. values setting as an accepted classification result, the selected M.S.E. values being less than equal to a user defined threshold value which is expressed as follows:

4.3 2-D Flow Characterization based on Mean Squared Error

Accepted Classification:

$$\text{if } M.S.E.\text{value} \leq \max(\text{of all } M.S.E.\text{values}) - a_{par} * \text{mean}(\text{of all } M.S.E.\text{values})$$

with $a_{par} \in [0, \infty)$. The M.S.E. values used are the M.S.E. values of the patterns that the structures have classified. Essentially, the threshold serves as an extra constraint on the classification output insuring its validity.

In conclusion, all the aforementioned steps are formulated in algorithm 4. Even this simple approach yields satisfying results in identifying and classifying an input structure.

Data: Flow vector field

Result: Flow Classification

Normalize the input flow field;

Create the reference patterns, i.e. the vortical flow filters;

Normalize the vector field of the reference pattern;

Calculate the Mean Squared Error between the normalized vector field of each reference and the normalized vector field of the input structure;

Sort in ascending form the Mean Squared Error values ;

Select as classification pattern the one corresponding to the lowest M.S.E. value;

if $M.S.E.\text{ value} \leq \text{Threshold value}$ **then**

 | **Classify the flow field to the selected pattern;**

else

 | **Do not classify the structure;**

end

Algorithm 4: The Flow Characterization algorithm using the Mean Squared Error.

4. PROPOSED APPROACH

Chapter 5

Results

In this chapter we present the results of our work. First of all, on the topic of motion estimation, the dynamic nature of fluid flows results in the presence of multiple types of motions to be present within a fluid flow motion field. Even accurate and well tested algorithms, such as Lucas and Kanade's algorithm or Horn and Schunck's result in different motion fields when implemented to fluid flows.

Instead, they seem to coincide on the main motion types present in the flow. So we will compare the results of our approach in comparison with those of other algorithms based on the main motion types detected. Another way to test the accuracy is to allow a user to define the basic motions observed and then test whether the algorithm succeeds in identifying them within the flow.

Furthermore, for the subject of Flow Characterization, the best way to test the accuracy of the presented algorithms is to examine the classification results of each algorithm and check whether the algorithm classifies the structure to the correct reference pattern or not. By testing the validity of each algorithm through a series of image sequences, representing different types of flows, we can check its accuracy.

Finally, we present information concerning the performance of each algorithm introduced, although we focused our research more on the topic of implementing an accurate algorithm and less on the optimization step. Even though the algorithms show satisfying performance results.

5. RESULTS

5.1 Algorithm Accuracy

In this section, the accuracy of each introduced algorithm is tested. We can consider an algorithm as accurate when the output data correspond to an expected accurate value. As mentioned previously, the accuracy of the algorithm presented on the topic of motion estimation, will depend on how accurately represents the main motion types present in the flow, compared to other algorithms and a user observations.

Also, the accuracy of the algorithms concerning the Flow Characterization field, is tested based on the observation of the validity of their results.

5.1.1 2-D Statistical Estimation of Fluid Flow Fields

The algorithm presented, is tested using image sequences representing fluid flows in natural phenomena, such as rivers or streams. The resulted motion field is compared to the ones of known accurate algorithms, such as Lucas and Kanade's, as well as user observations. The results of the algorithm can be compared during both of the two basic stages of implementation:(a) the local distribution function computation stage, and (b)the global optimization stage.

The results of the first stage of the algorithm can be compared with the results of correlation based algorithms such as Three Step Search algorithm. This is due to the fact that the displacement vectors are computed locally using block neighbourhood yielding discrete displacement vectors similar to the block matching approach used in Block Matching algorithms.

On the other hand, the second stage imposes a differential framework combined with the local estimation of the first stage resulting in a smoothed global motion field. Thus, the extracted motion field can be combined with other differential approaches such as Lucas Kanade's algorithm.

5.1.1.1 First Stage Results

In this stage, as mentioned in Chapter 4.1, the local distribution functions are computed, each displacement position in the destination neighbourhood is associated with a coefficient, which is considered to be a sample of the local distribution function. These coefficients correspond to displacement probabilities and so we can set as a destination

5.1 Algorithm Accuracy

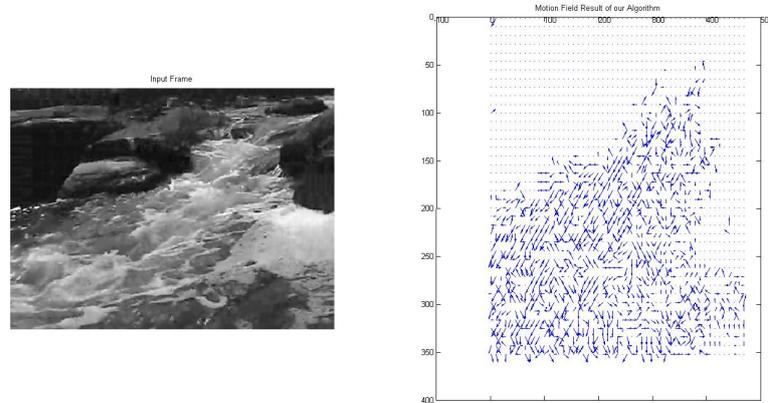


Figure 5.1: Small Stream Flow, (left)Input frame and (right)Resulted motion field, N_s size: 9×9 and D_s size: 5×5 .

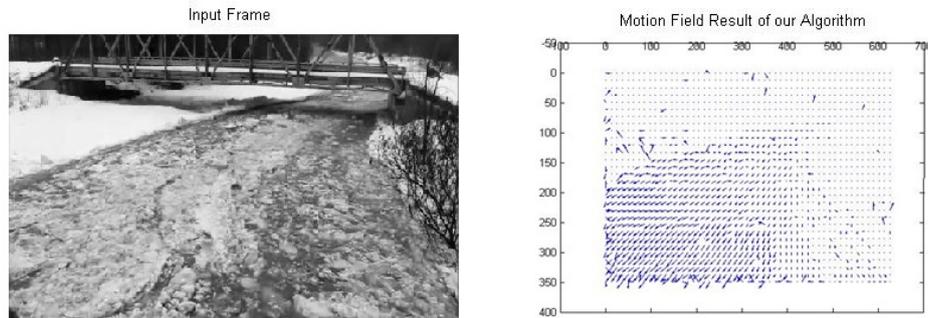
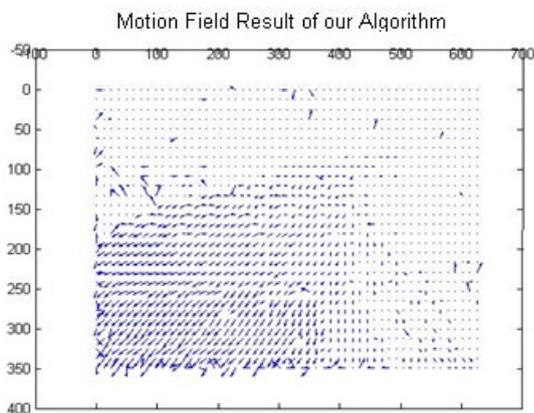


Figure 5.2: River Flowing covered with Ice, (left)Input frame and (right)Resulted motion field, N_s size: 9×9 and D_s size: 5×5 .

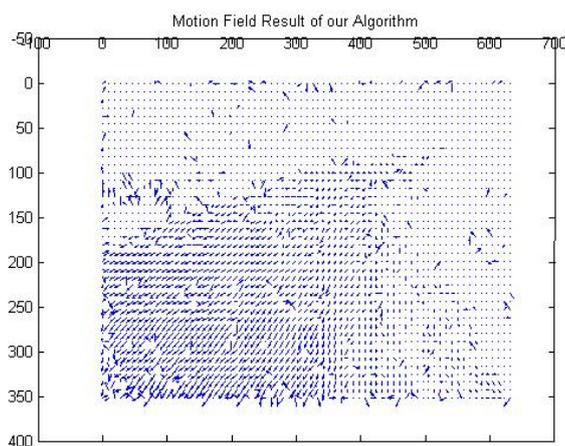
position the position associated to the coefficient with the highest probability value. Only two adjacent frames are enough to extract the motion field information.

Figures 5.1 and 5.2 show the resulted motion field of our algorithm for the cases of a small stream flow and an river flow covered in ice. The resulted motion field displayed contains the motion vectors of the central pixel of each block. As we can see the algorithm, even at this early stage, succeeds in finding the main flow motion and stationary regions in the scene such as river banks or rocks show no motion as they

5. RESULTS



(a) N_s size: 9×9 and D_s size: 5×5 .



(b) N_s size: 9×9 and D_s size: 8×8 .

Figure 5.3: River Flowing covered with Ice, we can observe that as the size of the destination neighbourhood rises so does the estimation error resulting to more erroneous motion vectors.

should have.

The quality of the estimated flow is determined by the size of the destination neighbourhood D_s as well as the noise present in the input image sequence. Due to the limited range of intensity values associated to a fluid flow, neighbouring pixels tend to show very close intensity values. In cases of large destination neighbourhoods the number of

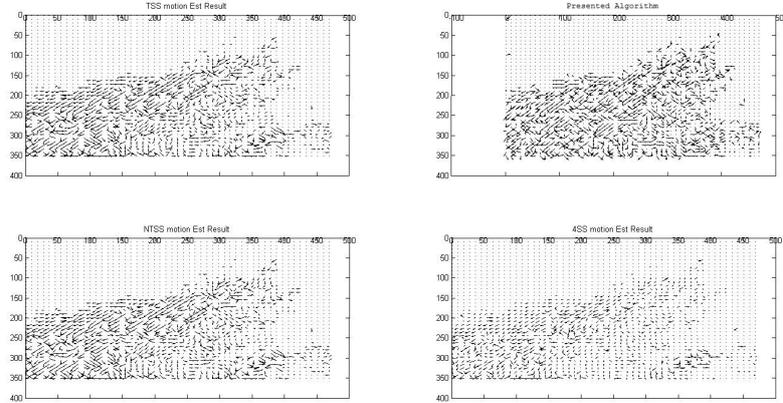


Figure 5.4: Small Stream flow, (left and up) Three Step Search algorithm, (right and up) Presented algorithm, (left and down) New Three Step Search algorithm and (right and down) Four Step Search algorithm, N_s size: 9×9 and D_s size: 5×5 , Similarity rates: 63.37% with TSS, 64.15% with NTSS and 63.14% with FSS.

coinciding intensity values rises and thus affecting the estimation. Figure 5.3 shows the affect of changing the destination neighbourhood size. As the destination neighbour gets wider the estimation error rises. So we have to select the window size of the destination neighbourhood wisely.

First Stage Accuracy Comparison

As mentioned earlier, we can compare the motion field resulted from this stage to Block Matching algorithms, due to the locality of the estimation and the discrete nature of the displacement vectors. Our algorithm is compared to 3 of the most known Block Matching algorithms: (a) Three Step Search algorithm (TSS), (b) Four Step Search algorithm (FSS) and (c) New Three Step Search algorithm (NTSS).

Figures 5.4 and 5.5 show the comparison results, for the previous flow cases presented, between these three algorithms and ours along with the velocity vector magnitude correspondence between our approach and the others. The similarity rate is computed as follows:

$$\text{Similarity rate} = \frac{u \text{ velocity component comparison} + v \text{ velocity component comparison}}{2}$$

5. RESULTS

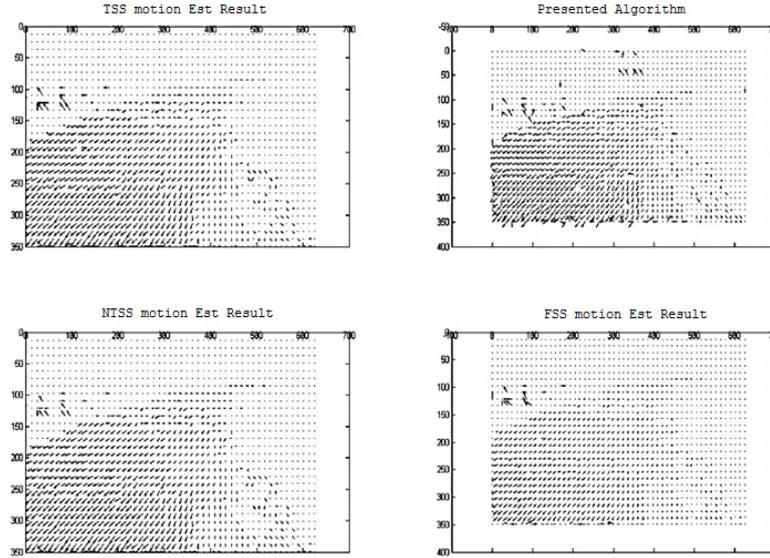


Figure 5.5: River Flowing covered with Ice, (left and up) Three Step Search algorithm, (right and up) Presented algorithm, (left and down) New Three Step Search algorithm and (right and down) Four Step Search algorithm, N_s size: 12×12 and D_s size: 7×7 , Similarity rates: 85.74% with TSS, 85.01% with NTSS and 82.59% with FSS.

where (u, v) are the x-axis and y-axis velocity components and for which the comparison of the velocity vectors is done as:

$$\text{velocity component comparison} = \frac{\text{number of equal vectors}}{\text{total number of vectors}} * 100\%$$

From the first look we can see that the main motion detected by our algorithm presents great similarities to the ones detected by the other algorithms. The similarity between the detected motion vectors ranges up to 64% for the case of a small stream flow with various flow types due to the presence of obstacles. Whereas, for more 'clear' flows such as the one of a river (Figure 5.5) where the flow is not obstructed and shows homogeneous direction our algorithm yields up to 85% resemblance with the other algorithms.

The following table displays the similarity rates between our approach and the three other approaches, based on the variations of the size of the spatial neighbourhood N_s . The comparison is based on the correspondence between the velocity vector magnitude

5.1 Algorithm Accuracy

between the approaches. Even between the three well tested and known for their accuracy approach the similarity rates range at 91.52% for the case of the river flow image and about 83.36% for the stream flow image.

N_s	D_s	TSS	NTSS	FSS	Image Flow
9×9	7×7	79.98%	79.43%	77.71%	River flowing covered in ice.
12×12	7×7	85.74%	85.01%	82.59%	River flowing covered in ice.
14×14	7×7	87.73%	87.25%	85.02%	River flowing covered in ice.
9×9	7×7	64.02%	64.19%	60.10%	Small Stream flow.
12×12	7×7	70.63%	70.67%	63.51%	Small Stream flow.
14×14	7×7	74.76%	74.94%	67.17%	Small Stream flow.

Through a number of comparisons between the motion vector magnitudes estimated by our algorithm and the ones estimated by the other approaches for various cases of flow images, such as rough multi-directional flows or homogeneous flows, we have concluded that our algorithm shows similarity ranging from 52% up to 96%.

The factors that determine the similarity rate are the neighbourhood block size as well as the type of flow, for example, abrupt flows where the fluid flow meets obstacles such as streams (Figure 5.4), result in abrupt multi-directional flow fields which lead to the drop of similarity rate even between accurate and well tested algorithms. Bigger neighbourhood sizes lead to fewer vectors to be estimated which results in fewer erroneous estimations.

5.1.1.2 User defined accuracy control

Due to the fact that even the most accurate and well tested algorithms show inconsistencies on the estimated flow, we used a user defined accuracy control where the user sets the correct, by clicking in the image, direction of the main flow, as observed, and the motion vectors corresponding the preferred direction as displayed with black points.

Figure 5.6 shows the result of this approach for the case of the river flowing covered with ice. The correct motion of the river is downwards. When the user selects the downward direction then the motion vectors corresponding to this selection (black colour) are

5. RESULTS



Figure 5.6: River Flowing covered with Ice, (circle)starting point and (x-mark)destination point (left)User selects the correct direction of motion, downward motion and (right)User selects the wrong direction of motion, upward motion.

displayed forming the pattern of the river. This shows that our approach has accurately estimated the main flow field. On the other hand, when the user selects the upward direction as the correct flow(false observation) then our algorithm yields almost no motion vectors.

5.1.1.3 Tracer Identification and Motion Association

As we mentioned in subsection 4.1.8 of Chapter 4, our approach supports tracer identification and motion visualization. Figures 5.7 and 5.8 show the results of our algorithm for the cases of a real river flow(Achelooos River) and an artificial river flow respectively.

In the case of the real river flow, we use a natural tracer, the foam floating on the surface of the water produced by the water motion. On the other hand, on the artificial river flow, an artificial tracer is used. White balls were placed on the flow to be used as tracer elements. Our approach successfully identifies the tracers in both cases, and associates the corresponding motion vectors. The use of tracers dramatically reduces the computational cost but we need to choose wisely the tracer element.

5.1.1.4 Second Stage Results

The second stage of our approach imposes a differential framework to the previous stage combining the information of the distribution functions with a differential function and a smoothing constraint in order to derive a smoothed global motion field. In our approach the differential function used, as mentioned earlier, is a 2-D Gaussian function.

Figure 5.9 shows the difference in the resulted motion field between the first and second stage, for the case of a small stream flow with abrupt motion. As we can observe

5.1 Algorithm Accuracy

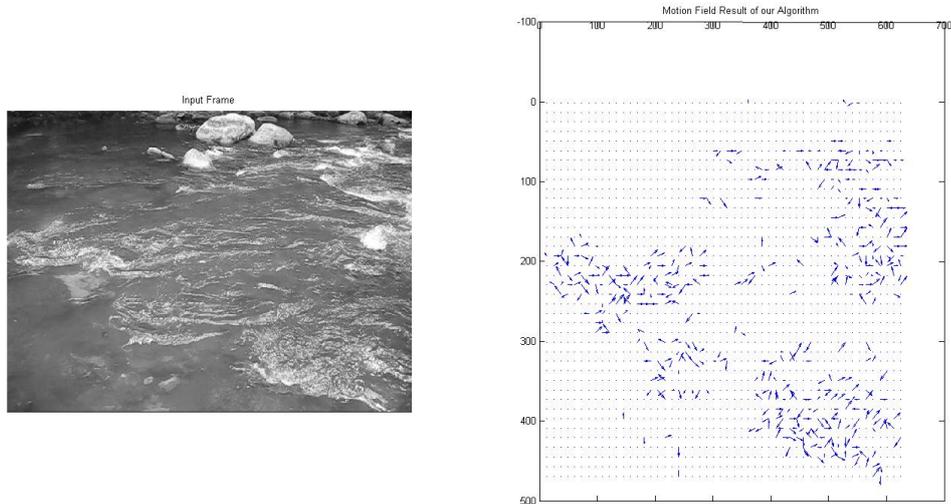


Figure 5.7: Acheloos River Flowing, Tracers used: surface foam, N_s size: 12×12 , D_s size: 7×7 .

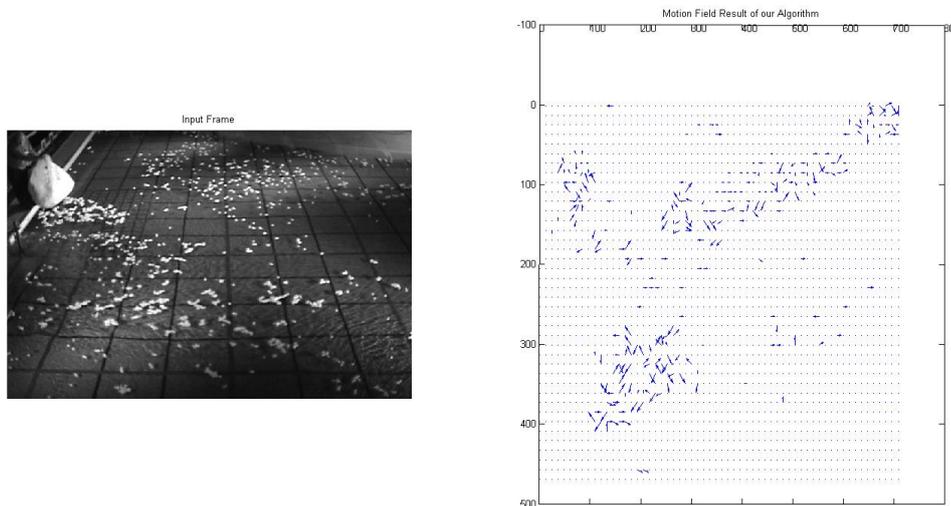


Figure 5.8: Artificial River Flowing, Tracers used: White balls, N_s size: 12×12 , D_s size: 7×7 .

5. RESULTS

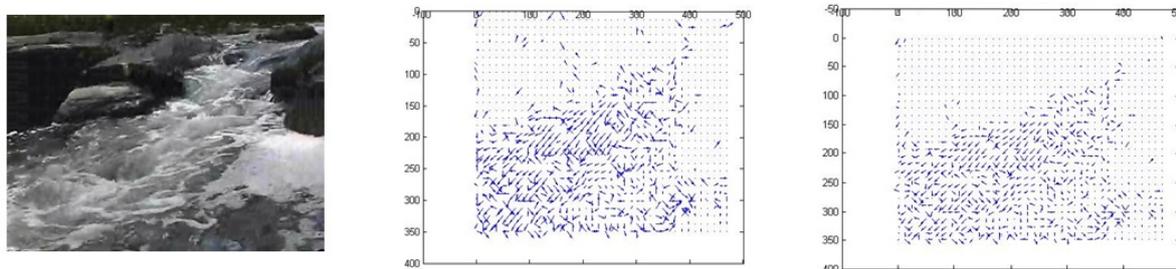


Figure 5.9: Small Stream flow, (left)Input frame, (center) Resulted motion field from the First Stage, (right)Resulted motion field from the Second Stage, N_s size: 12×12 , D_s size: 7×7 .

the flow field is more smoothed compared to the previous one and the vectors seem to have uniformly distributed values.

The addition of the differential framework leads to the classification of the algorithm to the differential methods, such as Lucas and Kanade's. We compared our approach with two of the most known differential methods, Lucas - Kanade's and Horn-Schunck's methods. Both of them have been well tested and used for many optical flow problems including the problem of motion field extraction in fluid flows.

Figures 5.10 and 5.11, show the resulted motion fields of the three approaches for the cases of a small stream flow and a river flowing covered in ice. Comparing the resulted motion fields we can derive the following observations: (1) All the three algorithms successfully display the main motion directions for both flow cases, the abrupt stream flow and the homogeneous river flow, (2)All the three algorithms disagree on the subsequent motion types, such as vortical flow and on the velocity vector magnitude range of values and (3) Our algorithm seems to show less sensitivity to vortical motion compared to the other ones. We can associate this drawback to two factors, the first is the loss of information due to the transition from the 3-Dimensional space to the 2-Dimensional and the second is the differential basis function used, Chang uses a product of cosine functions whereas we use a 2-D Gaussian function, which along with the support region used affects.

The fact that all three algorithms coincide on the main motion types present in the flow, lead to the conclusion that our algorithm is accurate enough for identifying main

5.1 Algorithm Accuracy

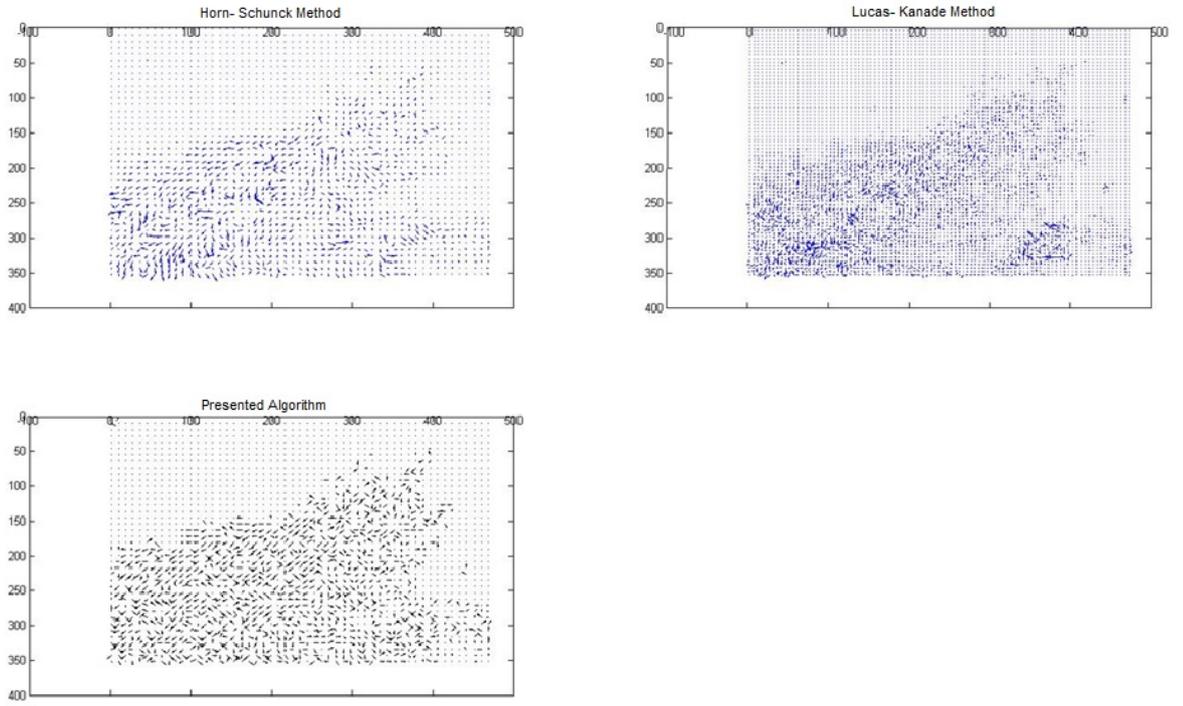


Figure 5.10: Small Stream flow, (left and up) Horn- Schunck’s Method, (right and up) Lucas- Kanade’s Method, (left and down) Our approach with basis function a 2-D Gaussian function and N_s size: 12×12 , D_s size: 7×7 .

motion flow types in fluid flow.

Parameters affecting the accuracy of the algorithm:

The parameters affecting the accuracy of our approach are first of all the components of the estimation function (eq. 4.15), such as the support region size or the value of the smoothing coefficient λ which affect the output of the algorithm. Also, the size of the destination neighbourhood must be selected wisely, due to the fact that it constraints the range of the displacement. Thus, in abrupt and fast flows we must use a wide destination neighbourhood so that we can include the fast transition, but with a cost on the estimation error.

5. RESULTS

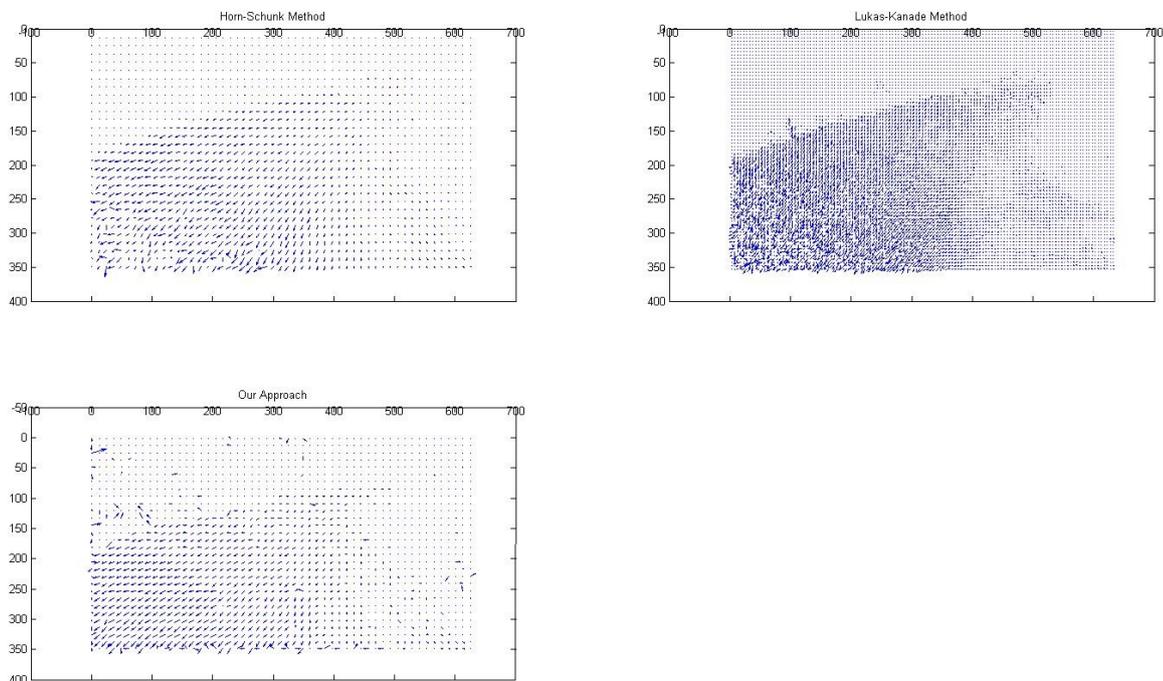


Figure 5.11: River Flowing covered in ice, (left and up) Horn- Schunck’s Method, (right and up) Lucas- Kanade’s Method, (left and down)Our approach with basis function a 2-D Gaussian function and N_s size: 12×12 , D_s size: 7×7 .

Finally, the differential function used can also affect the accuracy. We must use differential function which are affected by the local distribution functions uniformly and they do not cloak the importance of the distribution functions.

5.1.2 2-D Fluid Flow Characterization Using Vector Pattern Matching

To test this approach, we have used the resulted motion field images from Lucas and Kanade’s algorithm since it has shown to be the most sensitive one on vortical flow as well as on homogeneous flows. In order to identify a flow pattern we need to use the motion field information of all the pixels of the image.

5.1 Algorithm Accuracy

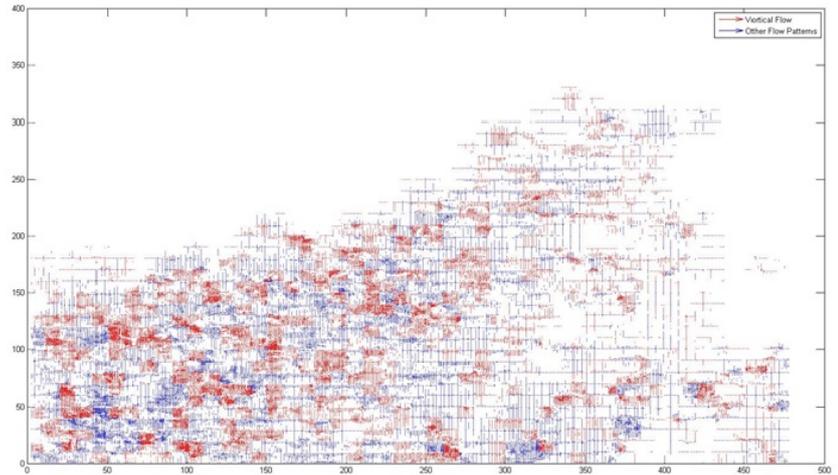


Figure 5.12: Small Stream Flow, (left) initial image, (right) Resulted motion field from Lucas- Kanade's algorithm, with red marked vectors corresponding to detected vortical flow patterns.

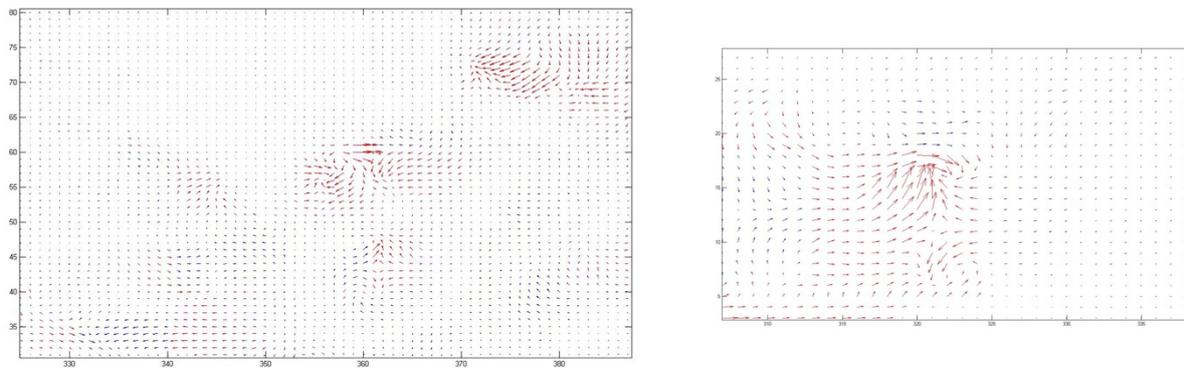


Figure 5.13: Small Stream Flow, (red)the detected vortical flow pattern, (blue) Other flow patterns.

Our approach is able to identify homogeneous flows and vortical flows, with the latter containing the cases of full shaped vortexes as well as structures displaying vorticity.

Figures 5.12 and 5.13 show the resulted classification of vortical flow for the case

5. RESULTS

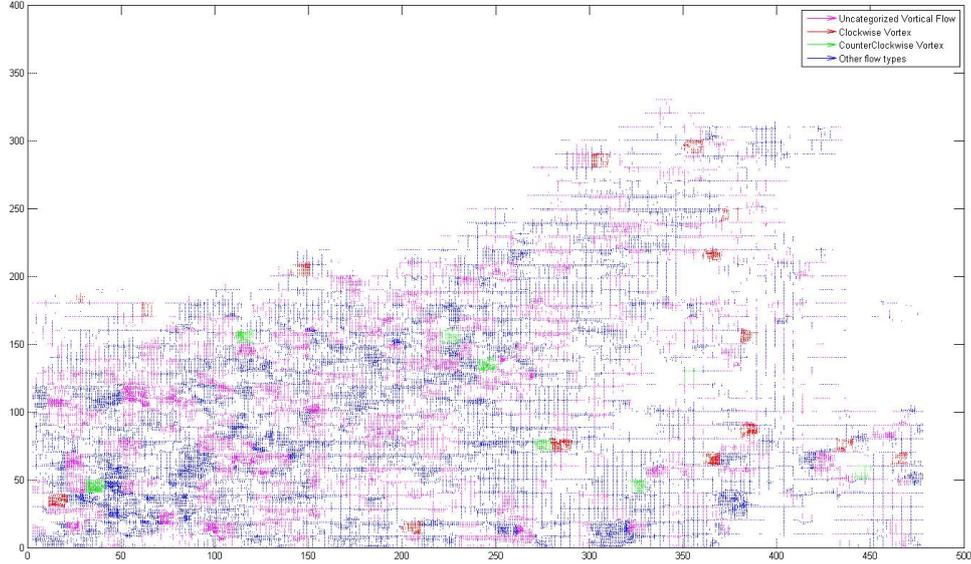


Figure 5.14: Small Stream Flow, (red mark) clockwise vortexes, (green mark) counter-clockwise vortexes, (pink mark) uncategorised vortical flow patterns and (blue) other flow types, Search window size: 10×10 and Threshold values: $T_{left} = 6, T_{right} = 8$.

of the abrupt stream flow. In abrupt flows we can expect the presence of vortices and as we can observe in Figure 5.12, our approach succeeds on identifying a large number of them. Figures 5.13 and show the accuracy of our approach as it has identified only vortical flows. The figures are zoomed areas of the previous image.

If we introduce a threshold on gradient of each velocity vector component we can even define whether the vortex has a clockwise or a counter clockwise rotation. The threshold used is defined as follows:

$$\begin{cases} \text{CounterClockwise Vortex,} & \text{if } \text{grad}(y - \text{axis}) \geq 0 \text{ and } -T_{left} \leq \text{grad}(x - \text{axis}) \leq T_{left} \\ \text{Clockwise Vortex,} & \text{if } \text{grad}(y - \text{axis}) \leq 0 \text{ and } -T_{right} \leq \text{grad}(x - \text{axis}) \leq T_{right} \\ \text{Unidentified Vortical flow,} & \text{Otherwise} \end{cases}$$

The threshold values (T_{left}, T_{right}) must be selected wisely and according to the size of the the search area. Figure 5.14 displays the discrimination classification for the case of the small stream flow. Figures 5.15, 5.16 and 5.17 show the classification results. Our approach provides satisfying results but there is classification error as can be observed

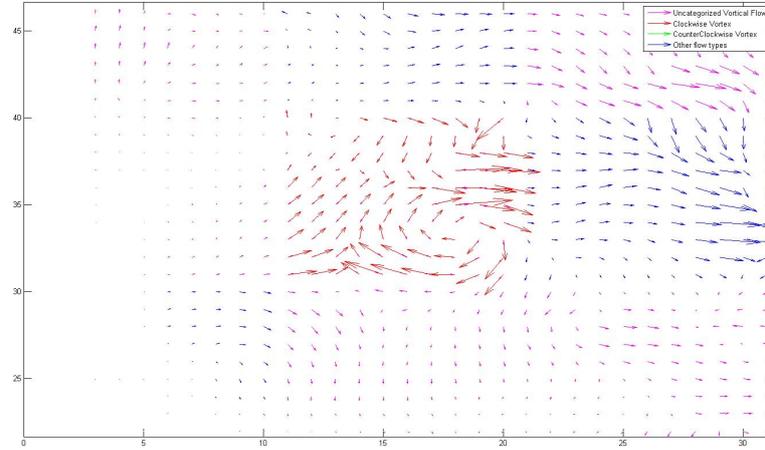


Figure 5.15: Small Stream Flow, (red mark)the clockwise vortex detected.

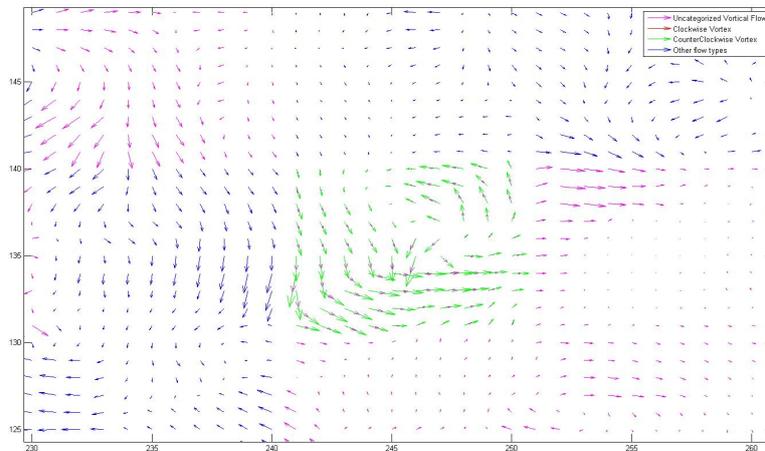


Figure 5.16: Small Stream Flow, (green mark)the counter-clockwise vortex detected.

in figure 5.17 which can be minimized by the appropriate selection of the search window size and the threshold values.

Figures 5.18 and 5.19 show the resulted classification for homogeneous flows for the stream image. As we can observe again our approach successfully identified homogeneous

5. RESULTS

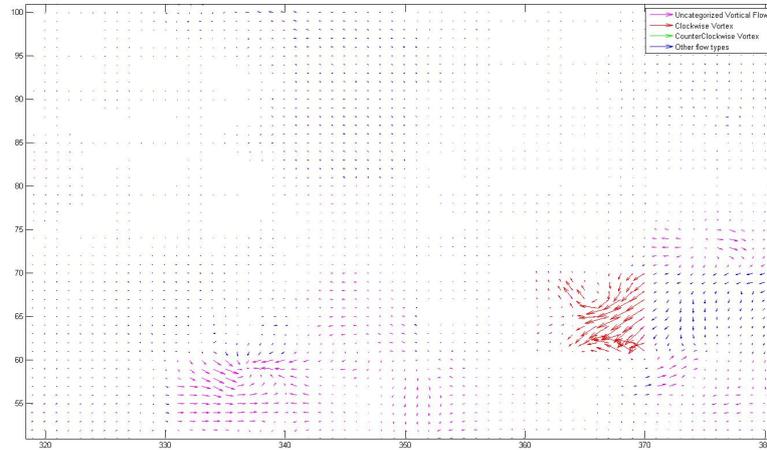


Figure 5.17: An example of the importance of correct thresholding, we can observe that despite the fact that a vortex has been identified as a vortex the threshold value did not allow the vortex (black mark) to be further classified (clockwise or counter clockwise).

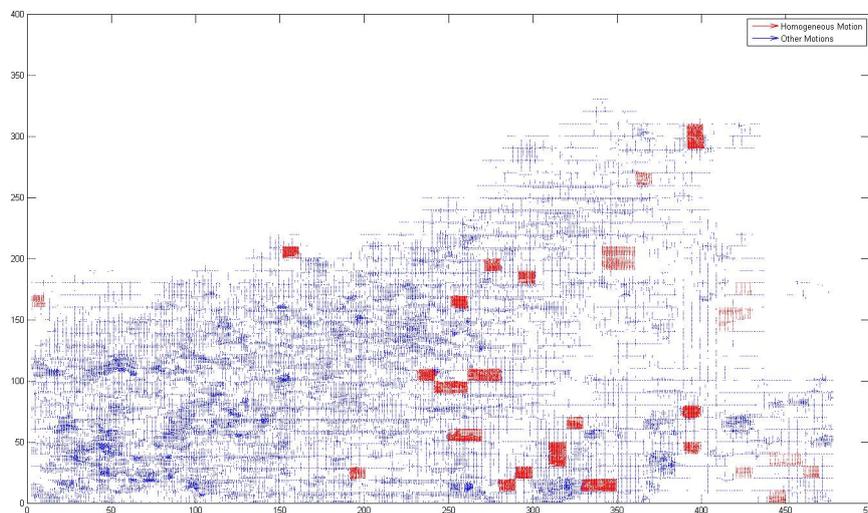


Figure 5.18: The homogeneous flow structures, red mark, identified in the Small Stream motion field, using the condition that the eigenvalues λ_1, λ_2 being zero.

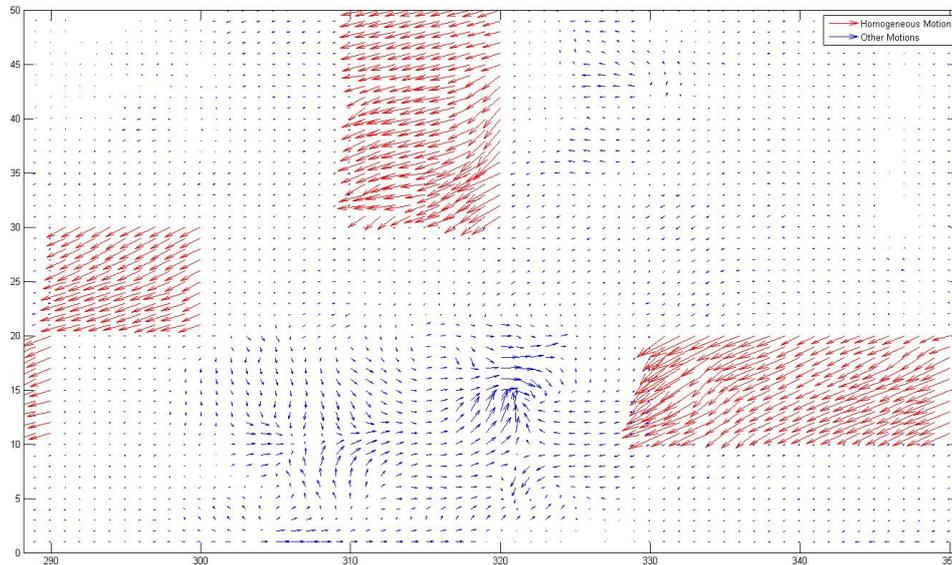


Figure 5.19: Enhanced display of the previous figure. The structures are correctly identified as homogeneous flow patterns. Search window size and filter pattern size: 10×10 .

motion fields. The rate of identified homogeneous structures can be increased by carefully selecting the search window size.

For the case of the flowing river covered in ice we can expect the number of homogeneous flow structures detected to be large. As we can see in figures 5.20 and 5.21, our algorithm succeeds in identifying the majority of the existing homogeneous flow structures.

For the case of the vortical flow patterns, figures 5.22 and 5.23 display the output of our approach, the red marks correspond to the vortical flow structures detected. We can observe that our approach yields satisfying results, identifying structures, which can be further improved with further thresholding the resulted eigenvalues.

Parameters affecting the accuracy of the algorithm:

The only parameter affecting the accuracy of this approach is the size of the filter pattern which is associated with the size of the search window (they have the same size). The size of the search window should be selected with care in order to contain all the

5. RESULTS

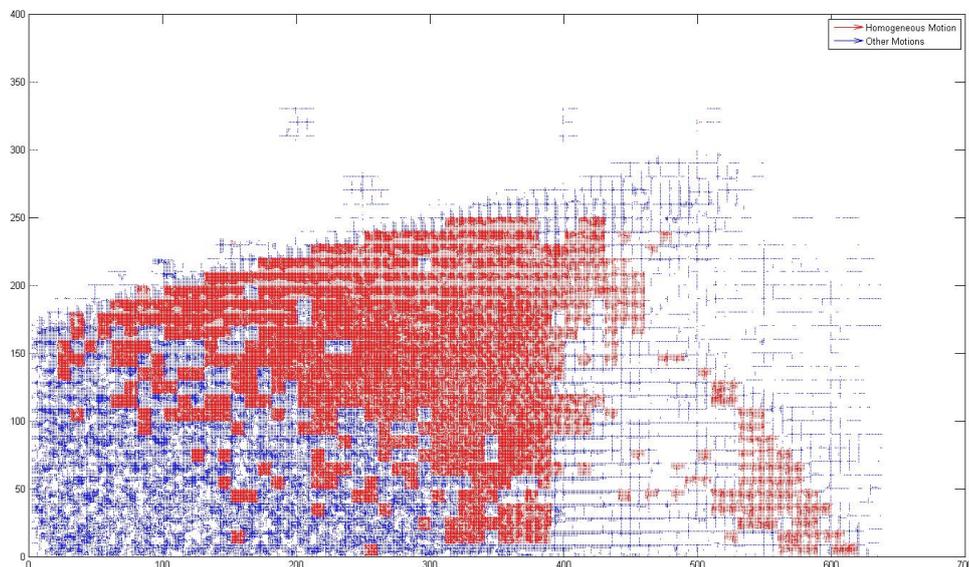


Figure 5.20: Classification output of our approach for the case of a River flowing covered in Ice, displaying the homogeneous flow structures. Initial image displayed in figure 5.2. Search window size and filter pattern size: 10×10 .

information relating the structure to the reference pattern. Small window sizes may lead to less information which will result in the pattern not being identified. On the other hand, large window sizes may lead to inclusion of false information, which may result again to the false classification of the data structure.

5.1.3 2-D Fluid Flow Characterization Using M.S.E.

This approach requires also the use of all the image pixels, so that we can have the all the information available for the classification process. As mentioned in subsection 4.3.1, of Chapter 4, the simplicity of this method allows the use of a variety of patterns. Although this method is more noise sensitive compared to the previous one, it succeeds in yielding very satisfying results, with a good discretization between the two main flow types of interest, vortical and homogeneous flows.

Figure 5.24 shows the output of this approach for the case of the river flow covered

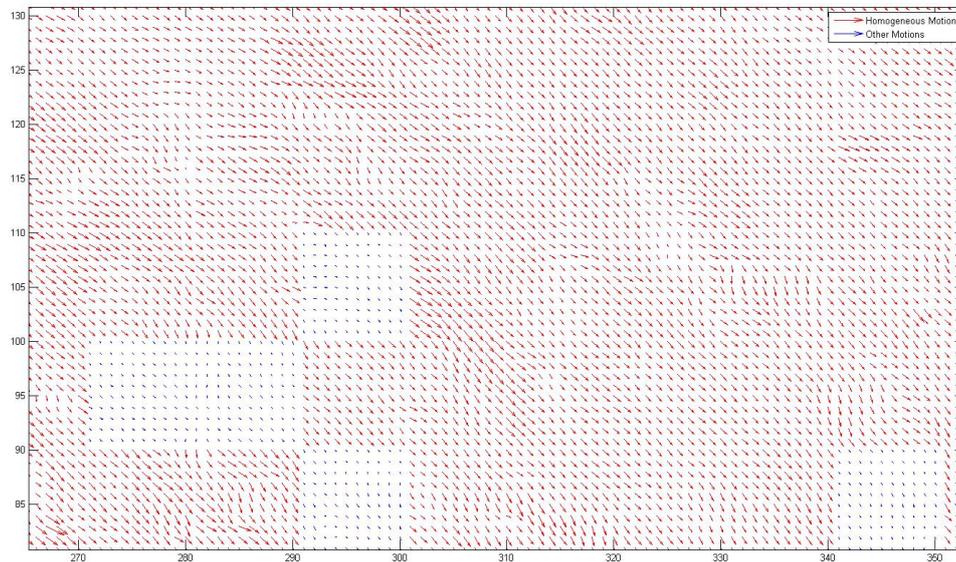


Figure 5.21: Enhanced display of figure 5.20. The structures are correctly identified as homogeneous flow patterns. Search window size and filter pattern size: 10×10 .

in ice, representing the homogeneous flow patterns with all its subclasses (downward, upward, left, right, upward & right, downward & right, downward & left and upward & left)¹, as well as the vortical flow patterns which are divided in clockwise and counter-clockwise vortices.

Figure 5.25 displays enhanced images of regions from the previous figure. As we can observe in figures and this approach succeeds in identifying and classifying correctly the majority of the flow structures. False classifications can be decreased by using smaller or larger window sizes as well as window shapes.

Figure 5.26 presents the output for the case of the small stream flow, with figures 5.27 and 5.28 presenting enhanced regions of this figure. Again it can be observed that this approach classifies satisfyingly the flow structures to the reference flow patterns but yet again there can be misclassifications.

¹The motions (upward or downward) and right, (upward or downward) and left are group (for the case of result images) into diagonally and right, diagonally and left motions respectively.

5. RESULTS

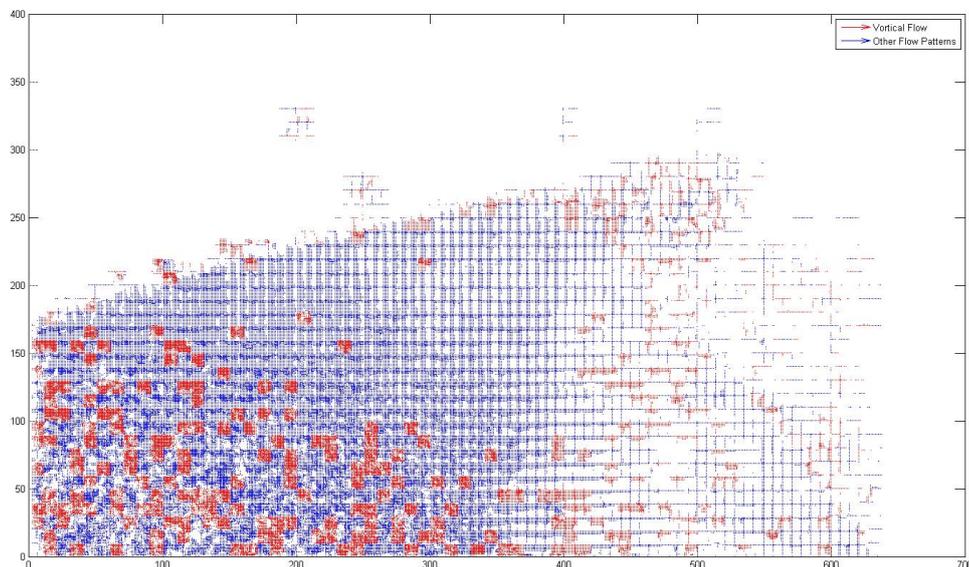


Figure 5.22: Classification output of our approach for the case of a River flowing covered in Ice, displaying the vortical flow structures detected. Initial image displayed in figure 5.2. Search window size and filter pattern size: 10×10 .

We can select to display only the homogeneous flows or the vortical flows or even both. Figures 5.29 and 5.30 show the case of selecting the vortical flows for the stream flow image and shows the vortexes identified.

Parameters affecting the accuracy of the algorithm:

The parameters affecting the accuracy of this approach are first of all, the size of the search window. The size of the search window should be selected wisely in order to contain all the information relating the structure to the reference pattern. Small window sizes may lead to less information which will result in the pattern not being identified. On the other hand, large window sizes may lead to inclusion of false information, which may result again to the false classification of the data structure.

Furthermore, the threshold value on the M.S.E. output of the selected reference pattern should be chosen in such way that it allows the structures showing resemblance to the selected pattern to be classified and structures whose rate of resemblance to the

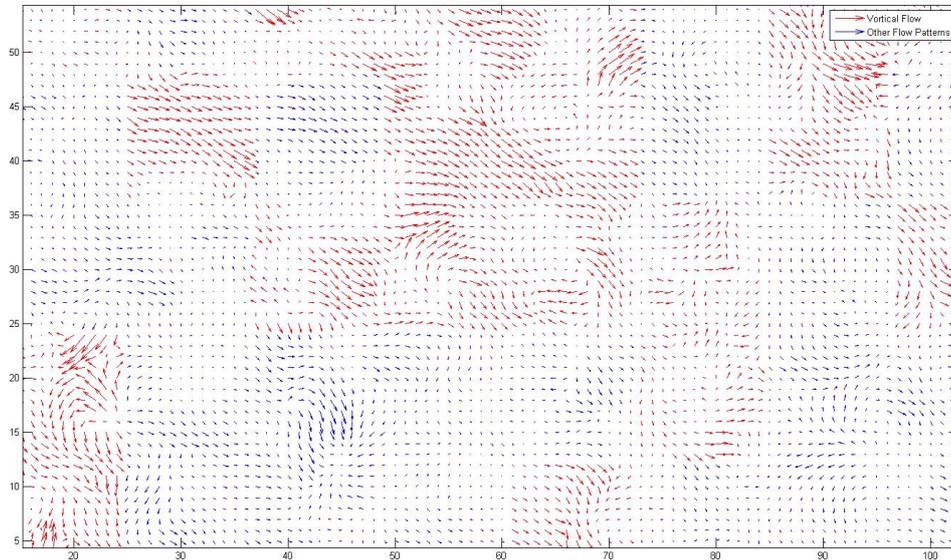


Figure 5.23: Enhanced display of figure 5.22. The structures are correctly identified as vortical flow patterns. Search window size and filter pattern size: 10×10 .

available reference patterns is below the acceptable value no to be classified at all. Also, an additional threshold on the curvature of each structure will improve the discretion between the vortical flow and the homogeneous flow patterns.

Finally, as mentioned earlier, the shape of the search window can be considered as a parameter affecting the classification outcome. Circle shaped windows may yield better results when identifying vortical flows.

5.2 Algorithm Performance

5.2.1 2-D Statistical Estimation of Fluid Flows

The aim of this thesis was to provide an accurate motion estimation algorithm, and thus, no optimization attempts were made. Our approach takes approximately 23 seconds for the first stage and up to 17.1 minutes for the global optimization step. This performance speed is the average computation speed for various image and neighbourhood sizes.

5. RESULTS

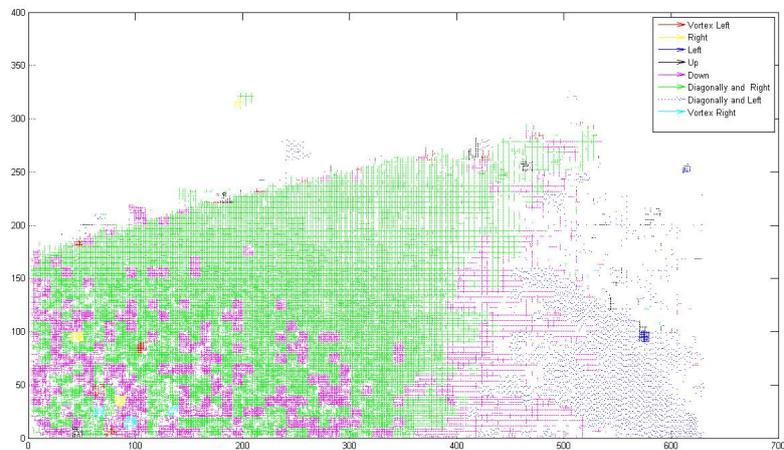


Figure 5.24: River Flowing covered in Ice, initial image is shown in figure 5.2. Search window size: 8×8 . All the possible motions are displayed.

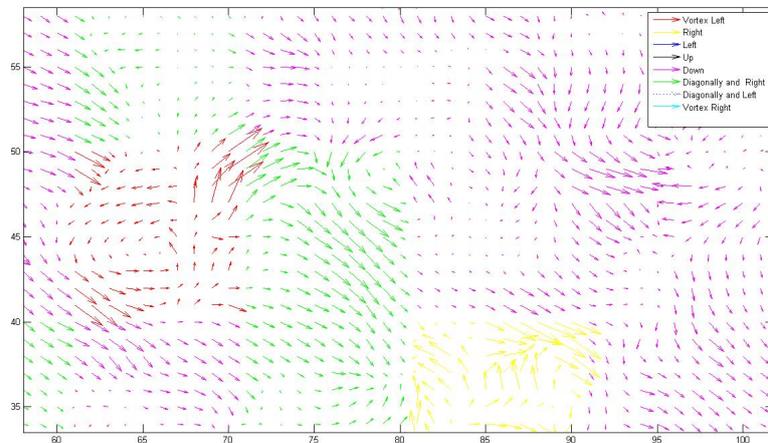


Figure 5.25: Enhanced region of figure 5.24. Search window size: 8×8 . All the possible motions are displayed. The majority of the structures are classified correctly.

The computational speed can be largely improved due to the fact that for the implementation only the most basic components were used due to the lack of information provided by the algorithm's founders.

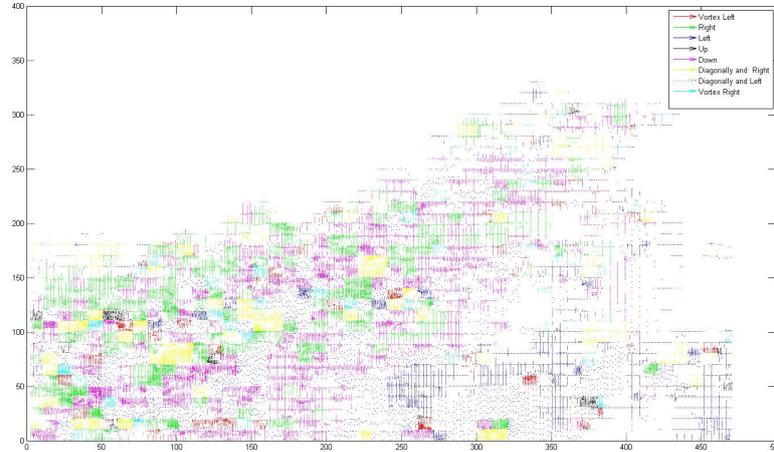


Figure 5.26: Small Stream Flow, initial image is shown in figure 5.12. Search window size: 8×8 . All the possible motions are displayed.

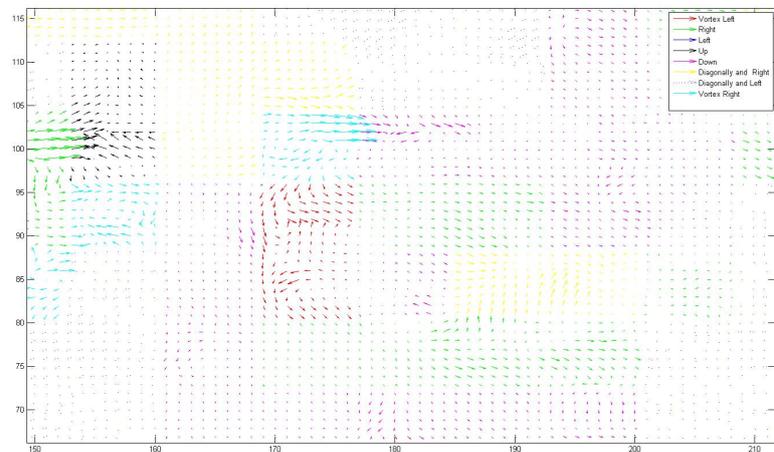


Figure 5.27: Enhanced region of figure 5.26. Search window size: 8×8 . All the possible motions are displayed. The majority of the structures are classified correctly.

The most time consuming component is the Conjugate Gradient component, which consumes approximately the 75% of the computational time. This is due to the fact that

5. RESULTS

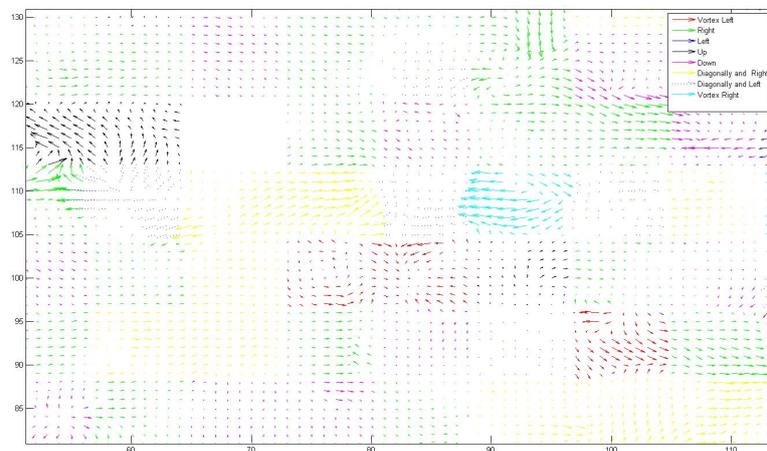


Figure 5.28: Enhanced region of figure 5.26. Search window size: 8×8 . All the possible motions are displayed. The majority of the structures are classified correctly.

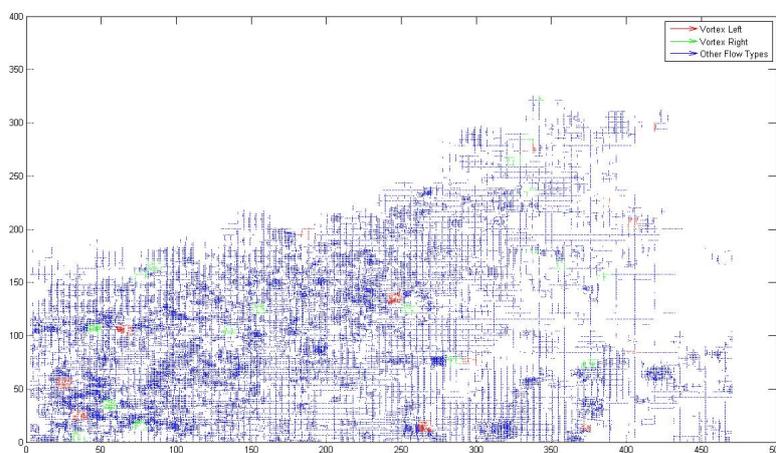


Figure 5.29: Resulted classification only for vortical flows (clockwise vortex (red) and counter-clockwise vortex (green)), for the case of the small Stream flow.

we did not use a fast conjugate gradient algorithm but one of the most accurate ones to improve the accuracy. Also, the update step α was given a small magnitude to insure

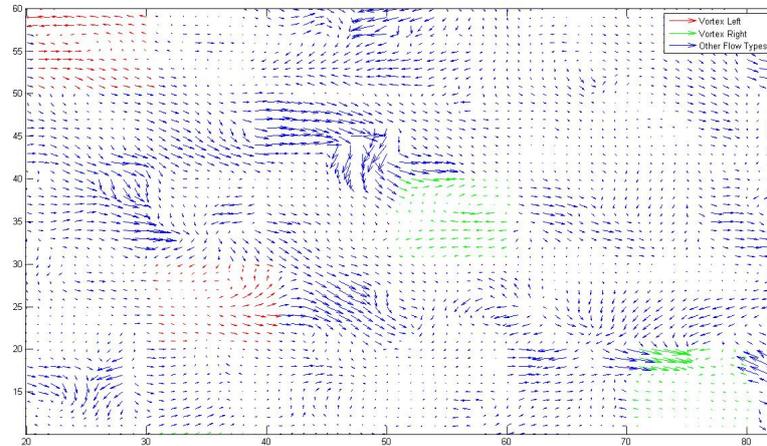


Figure 5.30: Enhanced region of figure 5.29. Search window size: 8×8 . Clockwise vortex(red) and counter-clockwise vortex(green).

the global minimization position is found. This leads to further burden the computation time.

5.2.2 2-D Fluid Flow Characterization Using Vector Pattern Matching

As mentioned earlier, the aim of this thesis is to provide accurate approaches instead of focusing to the computational speed. This also, applies for the case of the pattern recognition algorithms. No optimization attempts were made, however this approach proved to have satisfying computational speed results even at this early stage.

The average computational speed for a variety of image sizes and with a fixed filter size as well as search window size(for both cases, a 10×10 size was used), was approximately 160,35 seconds. The computational speed is affected by the search window size, larger windows will decrease the computational time but will affect the accuracy of the classification. We have found that the window size which achieve a balance on the computational speed affect as well as the accuracy is a 10×10 search window size.

Also, as one can imagine the image size also affects the computational time, larger images are more time consuming. For example, an image with dimensions 480×360 takes

5. RESULTS

approximately 128.231 seconds, with search window size 10×10 whereas an image with dimensions 640×360 takes approximately 195.245 seconds.

5.2.3 2-D Fluid Flow Characterization Using M.S.E.

Again, for the case 2-D Fluid Flow Characterization Using M.S.E. proposed approach no optimization attempts were made, however even this approach, although clearly correlation based, did not proved to be very time consuming.

The average computational speed for a variety of image sizes and with a fixed search window size, 10×10 , is about 3,5 minutes. For example, an image with dimensions 480×360 takes approximately 140.01 seconds, with search window size 10×10 whereas an image with dimensions 640×360 takes approximately 250.245 seconds.

Again, as mentioned previously, the computational speed is affected by the search window size, larger windows will decrease the computational time but will affect the accuracy of the classification. The window size which achieve a balance on the computational speed affect as well as the accuracy is a 10×10 search window size.

5.3 Vector Pattern Matching Vs Mean Squared Error

Comparing the two presented approaches we can see that the Vector Pattern Matching approach surpasses the M.S.E. approach both in accuracy and performance.

In terms of accuracy the Vector Pattern Matching approach gives more solid classifications compared to the M.S.E. but the M.S.E. approach is more flexible in the variety of patterns to be displayed. In order to achieve this many pattern discretions in the Vector Pattern Matching approach there has to be additional constraints.

On the topic of performance it is obvious that the Vector Pattern Matching approach is about 24% faster than the M.S.E. approach. Both approaches however, have not been optimized so the computational time can be further reduced and thus increasing the performance.

5.3 Vector Pattern Matching Vs Mean Squared Error

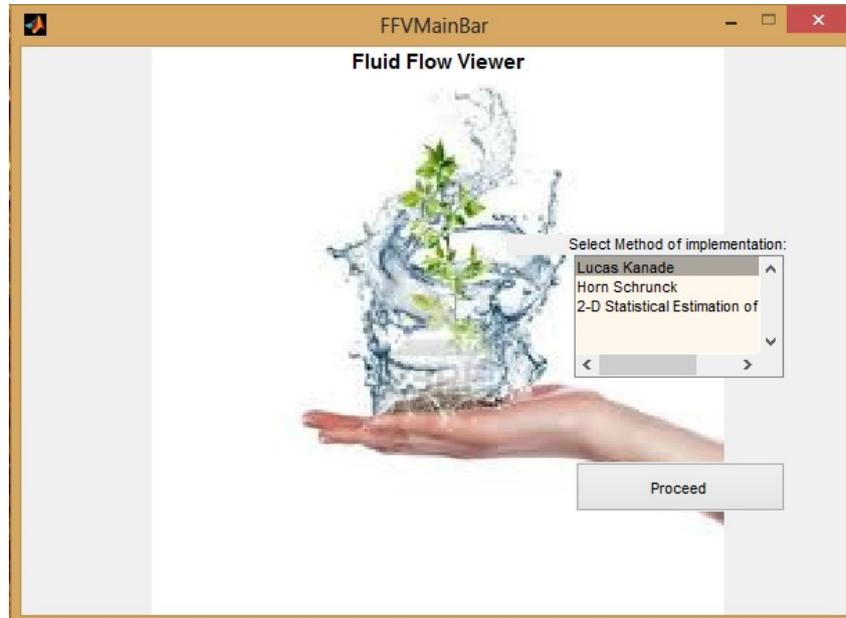


Figure 5.31: Initialization window of Fluid Flow Viewer.

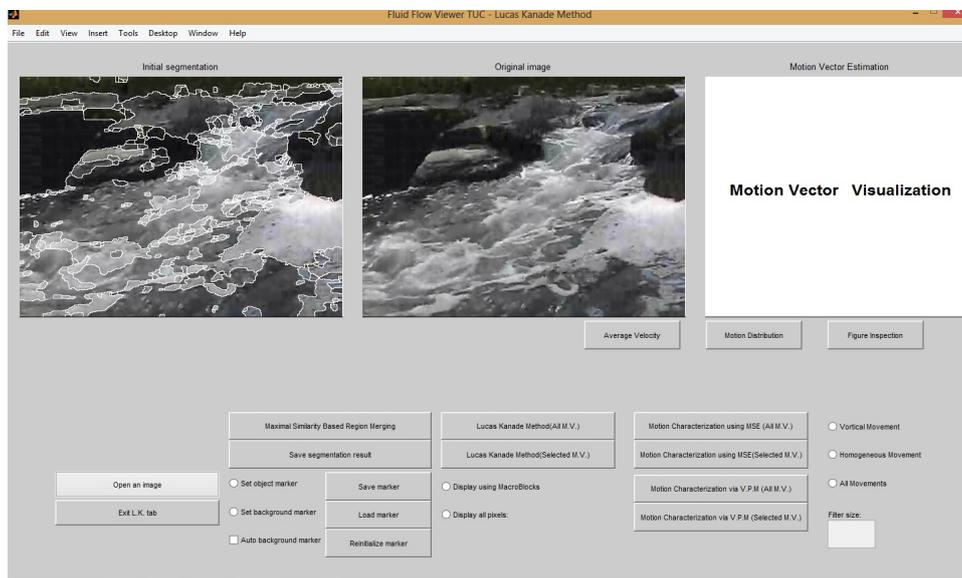


Figure 5.32: The main process panel of F.F.V., in this figure the selected method is Lucas- Kanade.

5. RESULTS

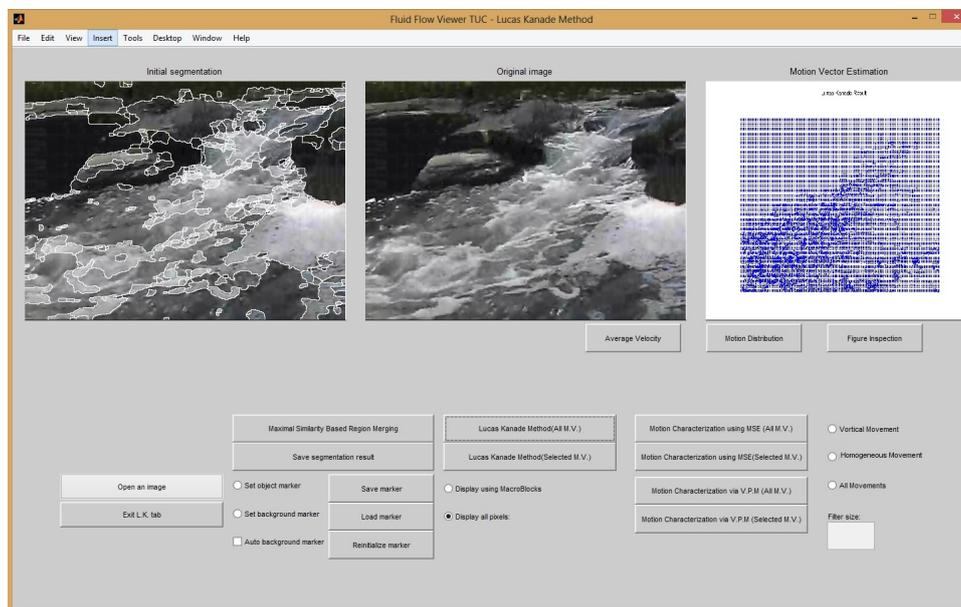


Figure 5.33: Lucas- Kanade result with all the image pixels present for the case of the small stream flow image. The whole output figure can be displayed when the user pushes the figure display button. This allows a more detailed examination of motion field.

5.4 Fluid Flow Viewer

All the aforementioned approaches, can be combined into a graphical user interface which will allow the extraction, visualization and processing of motion estimates for fluid flows. In this section we introduce a first approach on this subject and present the Fluid Flow Viewer(F.F.V.). Such tools, can be very useful to hydrologists studying the properties of the flow and using the information derived for the extra extraction of flow characteristics.

This graphical tool presents the user with the following possibilities:

- Motion field extraction using one of the available methods(*2-D Statistical Estimation, Lucas-Kanade, Horn- Schunck*).
- Computation of the average velocity of the motion field(pixel per frame).
- Identification and Classification of the flow types present in the flow field using one of the available methods(*2-D Flow Characterization using Vector Pattern Matching, 2-D Flow Characterization using Mean Squared Error*).

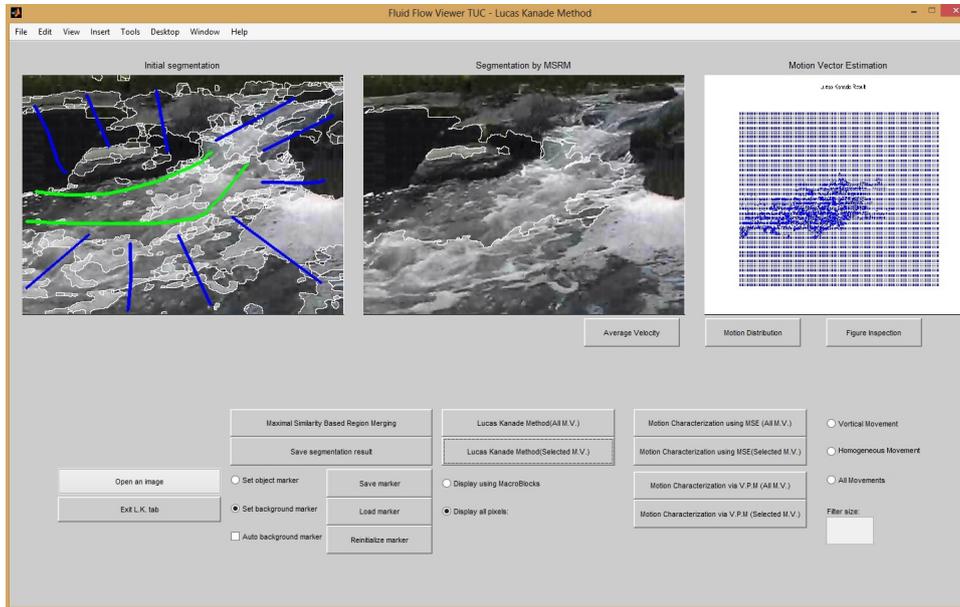


Figure 5.34: Lucas- Kanade result with all the image pixels present for the case of the small stream flow image. Only the user defined region's motion field is displayed. The green mark denotes the desired region and the blue mark denotes the region's background(undesired regions).

- Presentation of the flow type's distribution in the form of histogram.
- Selective processing, with the user defining the region of interest and implementing the aforementioned possibilities on this specific region.

First of all, the user must select the methodology of interest(Figure 5.31) and then the main task panel is displayed(Figure 5.32). As mentioned earlier the user can either work on the whole image(figure 5.33) or in a user defined region.

The region selection is done by selecting(drawing) the region of interest and then by using the Maximal Similarity Region based Merging by J.Ning et.al. [43], the region is isolated from the rest of the image(figure 5.34).

The user can use the motion extraction method with a choice of block implementation(figure 5.35) or all pixel implementation(figure 5.33). The first allows less execution time and displays the main motion types, whereas the second although much slower yields the full motion field with a detailed representation of the existing flow patterns.

5. RESULTS

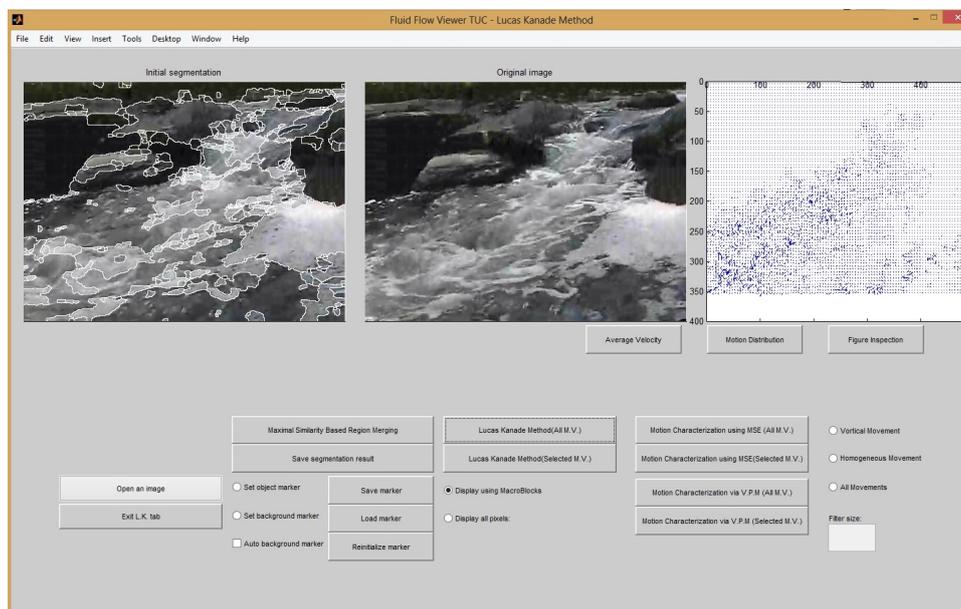


Figure 5.35: Lucas- Kanade result with a Macro-block pixel display for the case of the small stream flow image. The whole output figure can be displayed when the user pushes the figure display button. This allows a more detailed examination of motion field.

Another aspect of this tool, is the motion characterization the user can choose from the two presented methodologies, the 2-D Flow Characterization using Vector Pattern Matching approach(figures 5.36, 5.37) and the 2-D Flow Characterization using Mean Squared Error(figures 5.38, 5.39). The size of the search window and the filter pattern can be defined by the user. This allows the user to calibrate the result according to the image size and the flow motion(more abrupt flows require the use of smaller search windows, in order to identify all the small motion patterns present in the flow. The user can also select the flow type to be identified, homogeneous flows or vortical or even both.

Also, an additional visualization tool is the flow type histogram which presents the flow type analogy in the motion field(figure 5.40) as well as the average velocity vector components for the motion field(figure 5.41).

Finally, the tool also allows the user to open the desirable image and also save the working region for future use.¹

¹These capabilities along with the Maximal Similarity Region based Merging methodology were based on a graphical tool presented by J. Ning and is available on http://www4.comp.polyu.edu.hk/~cslzhang/MSRM/PR_MSRM_website.htm. The GUI presented by J.Ning was used as a the basic structure

5.4 Fluid Flow Viewer

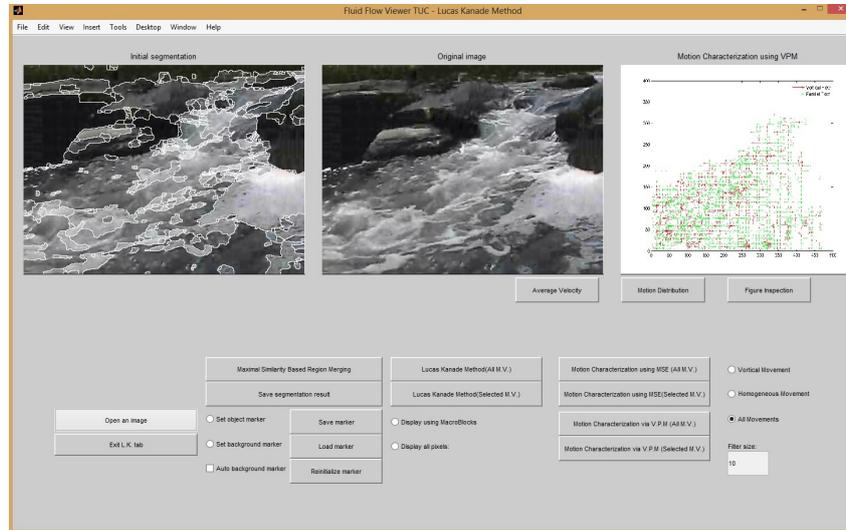


Figure 5.36: 2-D Flow Characterization using Vector Patter Matching result. The whole output figure can be displayed when the user pushes the figure display button. The user defined filter size is 10×10 and also all structures are to be classified to all the motion types available(homogeneous(green mark) and vortical(red mark)) are selected by the user.

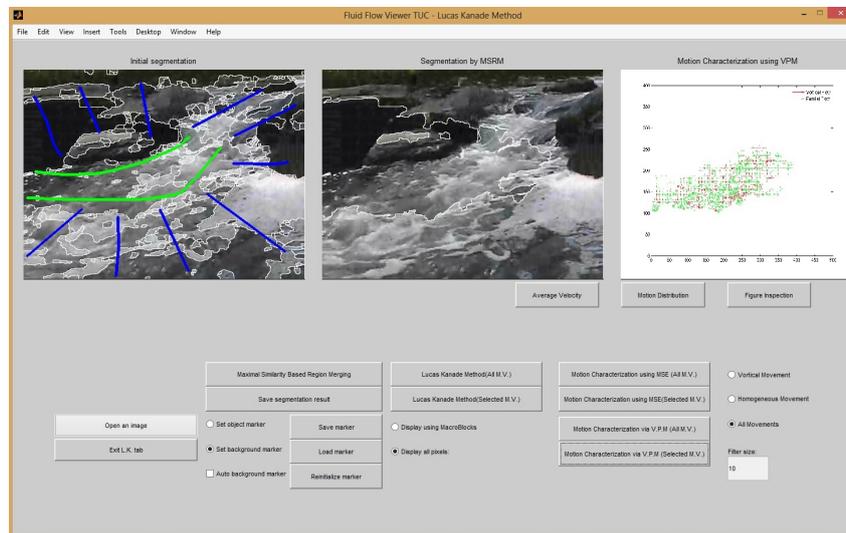


Figure 5.37: 2-D Flow Characterization using Vector Patter Matching result for the user defined region of examination. The same display parameters are present as in the previous figure.

for the development of our graphical tool.

5. RESULTS

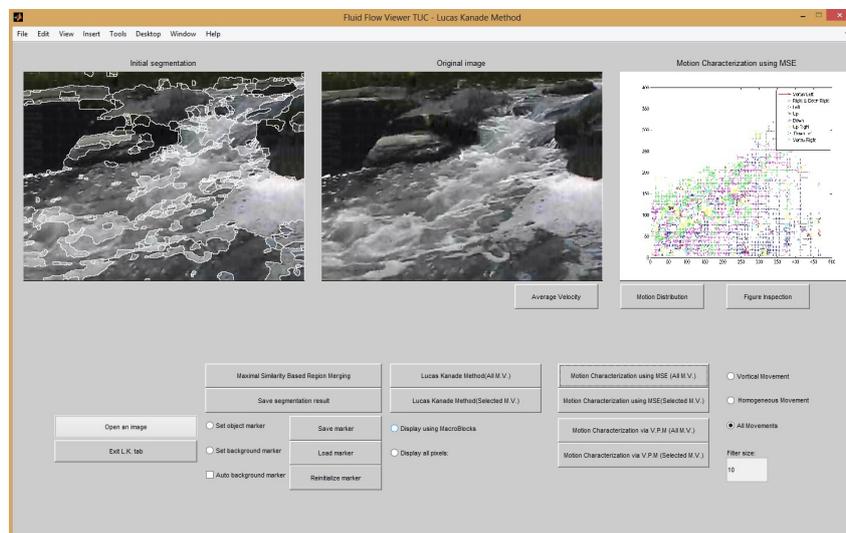


Figure 5.38: 2-D Flow Characterization using M.S.E. result. The whole output figure can be displayed when the user pushes the figure display button. The user defined filter size is 10×10 and also all structures are to be classified to all the motion types available (homogeneous with its subclasses and vortical with its subclasses (clockwise, counter-clockwise)) are selected by the user.

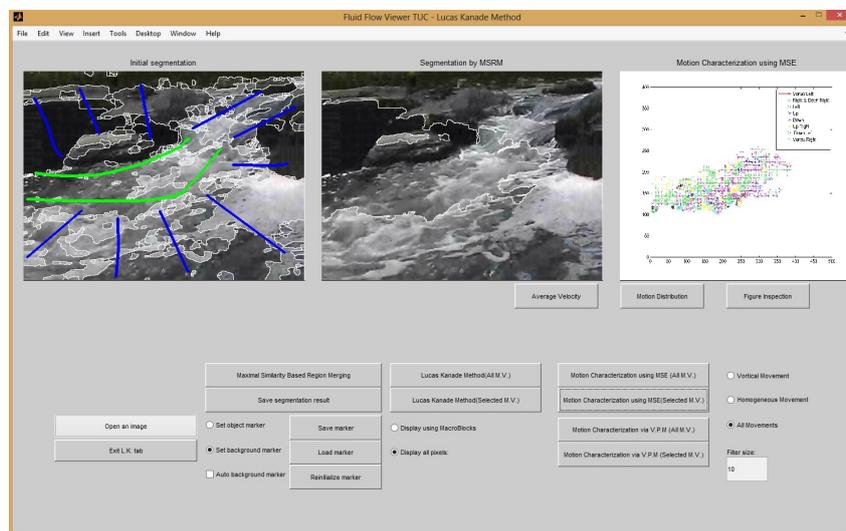


Figure 5.39: 2-D Flow Characterization using M.S.E. result for the user defined region of examination. The same display parameters are present as in the previous figure.

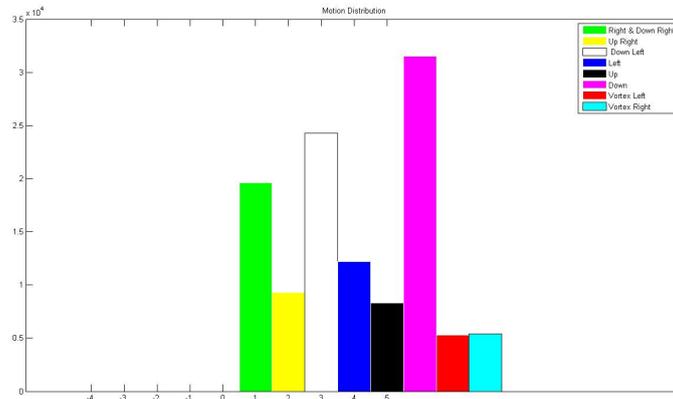


Figure 5.40: Histogram displaying the motion distribution analogy in the motion field. The current histogram displays the motion types analogy for the case of the small Stream flow. The motion classification was made with the 2-D Flow Characterization using M.S.E method. Similar motion distribution histogram can also be constructed with the 2-D Flow Characterization using Vector Patter Matching.

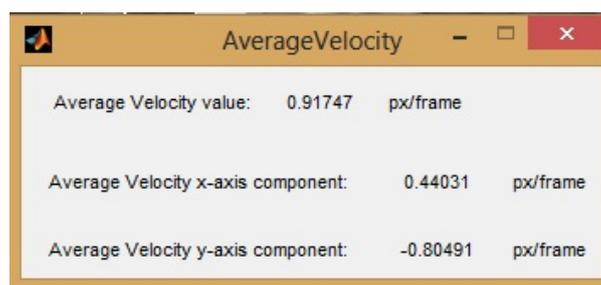


Figure 5.41: Average velocity components values. The current average velocity magnitude was calculated using the motion field information provide by Lucas- Kanade's result. Average velocity magnitudes can be computed also using the two other available methods(2-D Statistical Estimation and Horn- Schunck's methods).

5. RESULTS

Chapter 6

Conclusion

6.1 Conclusion

This thesis presents one approach for the topic of Motion estimation in fluid flows, based on a pre-existing method introduced by Chang et.al. [28]. Our approach follows Chang's framework but has major differences on the amount of data used and the parameters of the estimation process. The input data are only two adjacent frames from an image sequence and outputs the motion field of the flow.

The motion estimation is based on a statistical framework associating each destination position to a transaction likelihood which is considered as a sample of a local distribution function. The estimated local distribution function are combined with a differential framework whose optimization will result in a smooth global motion field.

Furthermore, this thesis introduces two approaches for the subject of Motion Pattern Recognition, with the first being a variation of a pre-existing method introduced by Heiberg et. al. [30], based on vector pattern matching ,and the later being a simple correlation approach based on Mean Squared Error(M.S.E.). Both approaches use as input the motion field of a natural fluid flow phenomena, such as rivers and output the identified pattern structures of vortical and homogeneous flows.

The proposed approaches, although, never been used before for fluid flow pattern recognition, show satisfying results which can be further improved with additional constraints. Moreover, although they are in early implementation stages with no optimization been done they seem to be computationally effective with performance speeds rang-

6. CONCLUSION

ing from seconds up to seconds with the computational speed depending on the search window size and the image size.

6.2 Future Work

6.2.1 2-D Statistical Estimation on Fluid Flows

There can be numerous improvements that can be done to this approach which can ameliorate the estimation accuracy as well as the algorithm's performance.

6.2.1.1 Increasing Accuracy

As mentioned in Chapter 5, the proposed approach estimates successfully the main motion flow types present in the fluid flow but is less sensitive on the identification of vortical flows. In order to succeed better estimation accuracy, which will result in the improvement of vortical flow estimation, we can proceed to the following additions:

- **Spatio-temporal Neighbourhood:**

This improvement used by Chang [28] uses the spatio-temporal neighbourhood for the coefficient estimation process instead of the spatial neighbourhood that we used. This addition improves the estimation accuracy, more data to work with leads to better estimation. However, the increase in accuracy will also increase the computational cost and thus, decreasing the speed performance of the algorithm. We can cope with this negative by using further additions which will improve the performance.

- **Combine with a hierarchical approach:**

Hierarchical approaches are known for their accuracy and also, for their ability to be combined with almost every optical flow computation method. By combining our approach with the hierarchical approach we will be able to improve the estimation accuracy. The hierarchical approach stages combined with the spatio-temporal neighbourhood will leads to even better estimations but with a further burden on the computational speed.

- **Shifting destination neighbourhood:**

Due to the fact that we use the spatial neighbourhood we are limited in specific range of translations defined by the size of the destination neighbourhood. Thus, arises the estimation accuracy problem we mentioned in correlation based approaches:” *The destination neighbourhood size must be chosen carefully according to the range of the expected displacement.*”.

This problem can be addressed by either imposing a spatio-temporal neighbourhood approach, as mentioned previously, or by using a shifting window for the destination neighbourhood, when choosing the spatial approach. This idea is widely used in correlation based approaches and has been proved to improve the accuracy. The basic concept is that the estimation starts with an initial spatial neighbourhood and after the estimation is done, the selected position is used as the basis position to form a new destination neighbourhood around this pixel. If the estimation results are better compared to the previous ones, then this destination neighbourhood is selected as the final transaction destination.

- **Basis Function:**

The accuracy of the approach in the global optimization step is depends largely on the basis function used. The function must be such to provide the differential framework which will lead to a global flow field but also take account the information provided by the local distribution functions and use it for the final estimation. In other words the basis function must be used in accordance with the distribution functions and not ”*shade*” their affect. Therefore, other differential functions may yield better estimation results than the Gaussian used by us or the product of cosine functions used by Chang.

- **Weights on the destination neighbourhood:**

The addition of weights on the destination neighbourhoods which will further increase the estimation accuracy. The choice of weights however, must be done carefully taking in mind the nature of the fluid flow motion.

6. CONCLUSION

6.2.1.2 Increasing Performance

As mentioned in the previous Chapter the aim of this thesis was to provide an accurate motion estimation algorithm, and thus, no optimization attempts were made. The computational speed can be largely improved because for the implementation only the most basic information about the components was used. This opens a window for improvements which can increase the performance of the algorithm:

- **Conjugate Gradient Algorithm:**

The algorithm used although accurate tends to be computationally expensive. This fact combined with a small update step α , increases even more the computational time. As mentioned in Chapter 5 in the performance section, the Conjugate gradient component takes up around 74% of the total execution time.

We can reduce this percentage and thus, increase the performance by using fast versions of the conjugate gradient algorithm(as Chang suggests and uses), along with an appropriate selection for the update step. However, the conjugate gradient version must be selected carefully so that the increase in the computational speed has a small cost in the accuracy. Also, the update step value should be such allowing as less computational time as possible but also succeeding in finding the global minimization position.

- **Fourier space domain:**

We can reduce the computational speed, and thus, increasing the performance by moving the computation of the elements to the Fourier domain. We know for fact that the computation is done much faster and simpler in the Fourier spaced due to the fact that even complicate function can be expressed as a summation of cosine and sine functions which makes their computation easier.

- **Computation Optimization:**

Another aspect affecting the performance is the way the elements are accessed. Approaches, for example, containing *for-loops* in order to access the data increase the computational complexity. By reducing their amount either immediately selecting the element(when a position is known) or by using matrices to save regions of data

and using them for computation we can decrease the computational speed leading to the increase of performance.

6.2.2 2-D Flow Characterization using Vector Pattern Matching

For the case of Vector Pattern Matching we can also, move to improvements that will increase the estimation accuracy as well as the algorithm's performance.

6.2.2.1 Increasing Accuracy

As mentioned the algorithm presented has shown very satisfying results on the identification of flow motion types. However, due to the fact that a 2-D approach has never been implemented before or tested in flow field data, there is space for further improvements in its accuracy:

- **Increase the number of reference Patterns:**

We can increase the number of patterns identified increasing the accuracy of the algorithm. However, the formulation as well as the discrimination characteristics must be selected with care so that the classification can be done simply and in accordance with the methodology. For example, we have shown(Chapter 5, accuracy section) that with a simple threshold in the gradient of the vortical flow pattern we can further discrete the flow pattern into clockwise and counter-clockwise vortexes. But, as mentioned the discretion elements must be selected carefully, even in our case the threshold component must be adapted to the image characteristics.

For this purpose, we can also use existed information such as the eigenvectors which according to the methodology founder Prof. Heiberg contain information about the symmetry axis of the structure to be classified.

We can also create homogeneous plane filters and use their filter response as a means of identification homogeneous flows, further increasing the number of flow types identified.

6. CONCLUSION

- **Pyramid Data Structure on the filter pattern size:**

We can create a pyramid structure containing the classification results for various filter and search window sizes, similar to a hierarchical approach. Combining the classification results we can have classification for patterns without being limited by the size of the search window and thus, increasing the accuracy of our performance.

6.2.2.2 Increasing Performance

Although, the computational speed of this approach is very satisfying, approximately 2,5 minutes per image, can be further reduced by implementing one of these additions:

- **Use the convolution result for the classification:**

By using the convolution result as a similarity measure exclude the step of the orientation tensor. This will result in two improvements: (1) reduce of the computational speed and (2) the better discrimination between the patterns, as it has been presented the tensor is constructed using a set of 3 orientation filters which for the case of a vortical flow contain the two subcategories of a vortex, clockwise and counter-clockwise. In order to further discriminate the classification in the case of a tensor we need to use further constraints which result in computational burden.

However, the classification success of this approach is not guaranteed since it has not been tested yet.

- **Implement the convolution in the Fourier domain:**

Implementing the convolution in the Fourier space using the Fourier Transform (FT) convolution theorem,¹ results in the increase of the computational speed which will result in the increase of the algorithm's performance.

- **Computation Optimization:**

As mentioned, in the previous method, by optimizing the way of the data are accessed and the way their associations are calculated(matrix calculation is easier and less computation expensive) we will increase the computational speed, improving the performance.

¹The convolution in the Fourier space is equal to the product of the Fourier Transforms of the input functions.

6.2.3 2-D Flow Characterization using Mean Squared Error

This correlation based approach can be improved in various ways which will increase the estimation accuracy as well as the algorithm's performance.

6.2.3.1 Increasing Accuracy

The proposed algorithm although, having the ability of using a variety of reference patterns is less accurate compared to the Vector Pattern Matching approach. Motions showing potential vortical nature might be considered as homogeneous patterns based on the M.S.E. result. We have restraint this misclassification cases by using further constraints, i.e. the curvature constraint mentioned during the presentation of the approach. The approach can become yet more accurate in the classification which can be accomplished with the following additions:

- **Shape of search window:**

Changing the shape of the search window will help on the improvement of the classification between the homogeneous and vortical flow patterns. Circle shape windows will show better identification rates for the case of vortical flows due to the circular formation of a vortex. By combining the classification results of a two set of window shapes(rectangular and circular) and thresholding their results we will achieve an increase on the accuracy.

- **Changing the size of the search window:**

By repeating the classification for using different search window sizes and comparing the classification results along with the M.S.E. value for each classification result at each size we might be able to stop being restricted by the size of the search window and achieve better classification results, and thus improving the accuracy.

6.2.3.2 Increasing Performance

As we have mentioned this algorithm is about 25% slower compared to the previous one, taking about 3,5 minutes per image. This is expected since it is a correlation based approach who tend to lack on computational speed but are noise insensitive. In order to increase the performance of this approach we proceed to the following improvements:

6. CONCLUSION

- **Computation Optimization:**

Again as mentioned, by optimizing the way of the data are accessed and the way their associations are calculated, matrix calculation is easier and less computation expensive, and we will improve the performance.

- **Additional Constraints:** We can use additional constraints prior the M.S.E. computation that will exclude structures that will fail to satisfy the acceptance threshold value, as mentioned in Chapter 4. This constraints must be computationally efficient, such as the number of -1,0 and 1 values present, and their implementation should burden the total computational time less than if we had to compute the M.S.E. for the structures we exclude.

6.2.4 Fluid Flow Viewer(F.F.V.)

In the previous Chapter we presented a Graphical tool which combines motion estimation approaches with the motion pattern recognition methods. We have also, presented its capabilities and its layout. This tool, as mentioned, can be a useful tool to hydrologists providing the basic information for the flow motion field and its characteristics. It is a promising tool which with more additions aiming the extraction of more scientific hydrological data can become a scientific tool used by hydrologists for the processing of hydrological data.

The characteristic of user defined flow processing or full flow processing allows the hydrologists to focus on the region of interests leading to more detailed and accurate processing. As a future work we aim at further improving the Segmentation algorithm allowing more precise isolation of the region of interest and thus, improving the detail of processing.

References

- [1] Bernd Jdhne, Handbook of Computer Vision and Applications: Signal processing and pattern recognition, vol. 2, 1999 [xv](#), [6](#)
- [2] S.S. Beauchemin and J.L. Barron, The computation of optical flow, ACM Computing Surveys **27**,no 3, (1996) 433–467 [xv](#), [8](#), [16](#), [17](#)
- [3] Horn B K. and Schunck B. G,1981. Determining optical flow, Artif. Intell. 17, 185–204 [7](#), [9](#), [10](#)
- [4] BERGEN, J. R. BURT, P. J. , HINGORANI. R., AND PELEG, S ,1992, Three-frame algorithm for estimatmg two-component Image motion. IEEE PAMI 14, 9, 886–896. [9](#)
- [5] B.D. Lucas and T. Kanade, An iterative image registration technique with an application to stereo vision, Proc. 7th International Joint Conference on Artificial Intelligence (1981), 674–679, Vancouver (CA) [11](#)
- [6] S.T. Barnard and W.B. Thompson, Disparity analysis of images, IEEE Transactions on Pattern Analysis and Machine Intelligence (1980), 333–340 [15](#)
- [7] T. Koga, K. Iinuma, A. Hirano, Y. Iijima and T. Ishiguro, "Motion compensated interframe coding for video conferencing", Pro. Nat. Telecommun. Conf., New Orleans, pp. G5.3.1- 5.3.5, Nov. 1981. [14](#)
- [8] L. M. Po and W. C. Ma, "A novel four-step search algorithm for fast block motion estimation", In: IEEE Trans. on Circuits and Systems for Video Technology, Vol. 6, No. 3, 313–317, Jun. 1996. [14](#)

REFERENCES

- [9] S. Zhu and K. K. Ma, “A new diamond search algorithm for fast block matching motion estimation”, *IEEE Trans. Image Processing*, Vol. 9, No. 2, pp. 287-290, Feb. 2000. [14](#)
- [10] ADELSON, E H AZZU BERGEN, J R 1985, Spatiotemporal energy models for the perception of motion, *Opt. Soc. Am. A* 2, 2, 284–299 [15](#)
- [11] P. Anandan, A computational framework and an algorithm for the measurement of visual motion, *International Journal of Computer Vision*,(1989), no. 3, 283–310 [17](#)
- [12] H.-H. Nagel. On the estimation of optical flow: Relations between different approaches and some new results. *Artificial Intelligence*, 33:299–324, 1987 [12](#)
- [13] A.J. Chorin and J.E. Marsden. *A Mathematical Introduction to Fluid Mechanics*. Springer-Verlag, New York, 1990. Texts in Applied Mathematics 4. [23](#)
- [14] Richard P. Wildes, Michael J. Amabile, Ann-Marie Lanzillotto, and Tzong-Shyng Leu, Recovering Estimates of Fluid Flow from Image Sequence Data, *Computer Vision and Image Understanding* 80, 246–266 (2000) [25](#), [26](#)
- [15] J. M. Fitzpatrick, The existence of geometrical density-image transformations corresponding to object motion, *Comput. Vision Graphics Image Process.* 44, 1988, 155-174. [25](#)
- [16] Y. Nakajima, H. Inomata, H. Nogawa, Y.Sato, S. Tamura, K. Okazaki, and S. Torii, “Physics-based flow estimation of fluids”, *Pattern Recognition*, vol. 36, pp. 1203–1212, 2003. [26](#)
- [17] B. Jahne and S. Wass, “Optical wave measurement technique for small scale water surface waves”, *Proc. Advances in Optical Instruments for Remote Sensing*, pp. 147–152, 1989. [27](#)
- [18] H. Saikano, “Fluid Motion Estimation Method based on Physical Properties of Waves”, *Comp. Vision and Pat. Recognition(CVPR)*, 2008 [xv](#), [27](#), [28](#), [29](#)
- [19] H.W. Haussecker and D. J. Fleet, “Computing optical flow with physical models of brightness variation”, *IEEE Trans. PAMI*, vol. 23, no. 6, pp. 661–673. [28](#)

REFERENCES

- [20] A. Bradley ,A. Kruger, E. A. Meselhe and M. Muste, Flow measurement in streams using video imagery, *Water Resources Research*, vol. 38, no. 12, 1315 , (2002) [xvi](#), [30](#), [31](#), [32](#)
- [21] Lecordier B, Demare D, Vervisch L M, Riveillon J and Triniti M 2001, Estimation of the accuracy of PIV treatments for turbulent flow studies by direct numerical simulation of multi-phase flow, *Meas. Sci. Technol.* 12 , 1382-91 [30](#)
- [22] R Tsubaki, I Fujita, S Tsutsumi, Measurement of the flood discharge of a small-sized river using an existing digital video recording system, *Journal of Hydro-environment Research*, 5 (2011) 313–321 [31](#)
- [23] Fujita, I., Komura, S., 1994. Application of video image analysis for measurements of river-surface flows. *Annu. J. Hydraul. Eng., JSCE* 38, 733–738 (in Japanese). [31](#)
- [24] Juan C. Ag'on and J. Jiminez (1987), On the performance of particle tracking. *Journal of Fluid Mechanics*,185, pp 447–468 [33](#)
- [25] Stitou A and RiethmullerM L 2001, Extension of PIV to super resolution using PTV, *Meas. Sci. Technol.* 12 1398-403 [xvi](#), [33](#)
- [26] Fujita, I. and Tsubaki, R. (2002). “A Novel Free- Surface Velocity Measurement Method Using Spatiotemporal Images,” *Hydraulic Measurements and Experimental Methods*, ASCE, on CDROM. [34](#)
- [27] Fujita, I., Watanabe., H. and Tsubaki, R. (2007). Development of a non-intrusive and efficient flow monitoring technique: The space-time image velocimetry (STIV) *Intl. J. River Basin Management* Vol. 5, No. 2 (2007), pp. 105 -114 [xvi](#), [34](#), [35](#)
- [28] Johnny Chang and David Edwards and Yizhou Yu, Statistical Estimation of Fluid Flow Fields, *ECCV workshop on Statistical Methods in Video Processing*, Copenhagen (2002),91-96 [v](#), [vii](#), [2](#), [23](#), [36](#), [49](#), [56](#), [59](#), [115](#), [116](#)
- [29] Einar Brandt Heiberg, Automated Feature Detection In Multidimensional Images, *Linkoping Studies in Science and Technology*, Thesis No. 909, Department of Biomedical Engineering and Medicine and Care, Linkfping University, Sweden, 2001. [38](#)

REFERENCES

- [30] Heiberg, Einar and Ebbers, Tino and Wigstrfm, Lars and Karlsson, Matts, Three-Dimensional Flow Characterization Using Vector Pattern Matching. *IEEE Trans. Vis. Comput. Graph.*(2003),num 3, vol 9, 313-319 [v](#), [vii](#), [xvi](#), [38](#), [39](#), [40](#), [47](#), [49](#), [64](#), [73](#), [115](#)
- [31] Ebling, Julia and Scheuermann, Gerik. Clifford Convolution And Pattern Matching On Vector Fields. *IEEE Visualization(2003)*,193-200, url: <http://dblp.uni-trier.de/db/conf/visualization/visualization2003.html> [xvi](#), [38](#), [39](#), [41](#)
- [32] Gerik Scheuermann, Topological Vector Field Visualization With Clifford Algebra, Dissertation, Fachbereich Informatik, University of Kaiserslautern, Germany 1999 [39](#)
- [33] Schlemmer, Michael and Heringer, Manuel and Morr, Florian and Hotz, Ingrid and Hering-Bertram, Martin and Garth, Christoph and Kollmann, Wolfgang and Hamann, Bernd and Hagen, Hans Moment Invariants for the Analysis of 2D Flow Fields. *IEEE Trans. Vis. Comput. Graph.*, no. 6, vol. 13, pp 1743-1750. [xvii](#), [41](#), [43](#), [44](#)
- [34] J. L. Helman and L. Hesselink. Representation and display of vector field topology in fluid flow data sets. *IEEE Computer*, 22(8):27 -36, Aug. 1989. [40](#)
- [35] Frits H. Post and Benjamin Vrolijk and Helwig Hauser and Robert S. Laramée and Helmut Doleisch. Feature Extraction and Visualisation of Flow Fields, 2002 [xvi](#), [40](#), [42](#)
- [36] Scheuermann, Gerik and Kr'oger, Heinz and Menzel, Martin and Rockwood, Alyn P., Visualizing Nonlinear Vector Field Topology. *IEEE TVCG*, 4(2):109•116, Apr. 1998
- [37] M. Chong, A.E. Perry, and B.J. Cantwell, "A General Classification of Three-Dimensional Flow Field," *Physics of Fluids*, vol. A, no. 2, p. 765, 1990. [45](#)
- [38] I.A. Sadarjoen, F.H. Post, B. Ma, D. Banks, and H.G. Pagendarm, "Selective Visualization of Vortices in Hydrodynamic Flows," *Proc. IEEE Visualization 1998*, pp. 419-422, 1998. [xvii](#), [45](#), [46](#)

- [39] Pytlak, Radoslaw “Conjugate gradient algorithms in nonconvex optimization”, Non-convex Optimization and Its Applications, 2008, Chapter 2, Algorithm 2.7, page 87. [60](#)
- [40] G. Sparr, “Linjar Algebra”.Studentlitteratur, 1982. [70](#)
- [41] H. Knutsson. Filtering and Reconstruction in Image Processing. Diss. No. 88, Linkoping University, 1982. [72](#)
- [42] H. Knutsson. “Representing Local Structure Using Tensors,” Proc. Sixth Scandinavian Conf. Image Analysis, pp. 244-251, 1989. [72](#), [73](#)
- [43] Ning, Jifeng and Lei, Zhang and Zhang, David and Wu, Chengke. “Interactive image segmentation by maximal similarity based region merging,” Pattern Recognition, vol. 43, pp. 445-456, 2010. [109](#)
- [44] Andreas Andersson, Master Thesis: “Automatic identification of vortex cores in three dimensional blood flow in the human heart”, Linkonping University, Supervisor: Einar Heiberg. [v](#), [vii](#), [68](#), [69](#), [71](#)