

12-2002

Detection and Generalization of Spatio-temporal Trajectories for Motion Imagery

Panayotis Partsinevelos

Follow this and additional works at: <http://digitalcommons.library.umaine.edu/etd>

 Part of the [Databases and Information Systems Commons](#)

Recommended Citation

Partsinevelos, Panayotis, "Detection and Generalization of Spatio-temporal Trajectories for Motion Imagery" (2002). *Electronic Theses and Dissertations*. 581.

<http://digitalcommons.library.umaine.edu/etd/581>

This Open-Access Dissertation is brought to you for free and open access by DigitalCommons@UMaine. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of DigitalCommons@UMaine.

**DETECTION AND GENERALIZATION OF
SPATIO-TEMPORAL TRAJECTORIES
FOR MOTION IMAGERY**

By

Panayotis Partsinevelos

Dipl. Eng. National Technical University of Athens, 1996

A THESIS

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Doctor of Philosophy

(in Spatial Information Science and Engineering)

The Graduate School

The University of Maine

December, 2002

Advisory Committee:

Peggy Agouris, Associate Professor of Spatial Information Science and Engineering, Advisor

M. Kate Beard-Tisdale, Professor of Spatial Information Science and Engineering

Max J. Egenhofer, Professor of Spatial Information Science and Engineering

Mohamad Musavi, Professor of Electrical and Computer Engineering

Anthony Stefanidis, Assistant Professor of Spatial Information Science and Engineering

DETECTION AND GENERALIZATION OF SPATIO-TEMPORAL TRAJECTORIES FOR MOTION IMAGERY

By Panayotis Partsinevelos

Thesis Advisor: Dr. Peggy Agouris

An Abstract of the Thesis Presented
in Partial Fulfillment of the Requirements for the
Degree of Doctor of Philosophy
(in Spatial Information Science and Engineering)
December, 2002

In today's world of vast information availability users often confront large unorganized amounts of data with limited tools for managing them. Motion imagery datasets have become increasingly popular means for exposing and disseminating information. Commonly, moving objects are of primary interest in modeling such datasets. Users may require different levels of detail mainly for visualization and further processing purposes according to the application at hand.

In this thesis we exploit the geometric attributes of objects for dataset summarization by using a series of image processing and neural network tools. In order to form data summaries we select representative time instances through the segmentation of an object's spatio-temporal trajectory lines. High movement variation instances are selected through a new hybrid self-organizing map (SOM) technique to describe a single spatio-temporal trajectory. Multiple objects move in diverse yet classifiable patterns. In

order to group corresponding trajectories we utilize an abstraction mechanism that investigates a vague moving relevance between the data in space and time. Thus, we introduce the spatio-temporal neighborhood unit as a variable generalization surface. By altering the unit's dimensions, scaled generalization is accomplished.

Common complications in tracking applications that include occlusion, noise, information gaps and unconnected segments of data sequences are addressed through the hybrid-SOM analysis. Nevertheless, entangled data sequences where no information on which data entry belongs to each corresponding trajectory are frequently evident. A multidimensional classification technique that combines geometric and backpropagation neural network implementation is used to distinguish between trajectory data.

Further more, modeling and summarization of two-dimensional phenomena evolving in time brings forward the novel concept of spatio-temporal helixes as compact event representations. The phenomena models are comprised of SOM movement nodes (spines) and cardinality shape-change descriptors (prongs).

While we focus on the analysis of MI datasets, the framework can be generalized to function with other types of spatio-temporal datasets. Multiple scale generalization is allowed in a dynamic significance-based scale rather than a constant one. The constructed summaries are not just a visualization product but they support further processing for metadata creation, indexing, and querying. Experimentation, comparisons and error estimations for each technique support the analyses discussed.

DEDICATION

To my parents Emmanuel and Georgia and brothers Antonis and Anna,
to my aunt Dimitra and the Sierros family
and to Fr. George

‘Rejoice, thou who showest philosophers to be unwise,
Rejoice, thou who exposest the experts as irrational...
Rejoice, thou who drawest us from ignorance,
Rejoice thou who enlightenest many with knowledge’

ACKNOWLEDGEMENTS

I would like to acknowledge my advisor Peggy Agouris and my committee members, Anthony Stefanidis, Max Egenhofer, Kate Beard and Mohamad Musavi for all the support and guidance they offered throughout my graduate studies. This work was partially supported by the National Science Foundation under NSF grant numbers: IIS-9702233, DGI-9983445 and IIS-0121269. In addition, I was supported by the Greek State Scholarship's Foundation.

I would like to cordially thank and acknowledge my parents Emmanuel and Georgia and my brother Antonis and sister-in-law Anna, for the immense patience, support and love they offered me in every part of this journey. Also, I would like to thank my family from MA, Nickolas, Dimitra, Constantine, Sunita and young Dimitra for all their help, love and the "Sierros scholarship". Many thanks to Penelope, Panayotis, John, Alexis, Konstantina, Tolis, John, Maria. Also to my beloved aunt Eleni.

Special thanks to all my professors and colleagues in the department for all their help. Especially: Sotiris, Peter and Harris. Also to my undergraduate professors, Dr. Balodimos, Dr. Nakos and Dr. Karras, for their support throughout my studies.

Thanks to all my friends from Greece and the US who helped me in many ways: Manolis, Dimitra, Antonis, Apostolos, Konstantinos, Anastasia, Becky, Tina, Betty, Kiriaki, Vagelis, George, Dimitris, Stefanos, John, Stavros, Efi, Vasilias, Chris, Varnavas, Tina, Vagelis, Panayotis, John. Cheers and many thanks to my teammates from the University of Maine volleyball club that I was a part of for more than three years.

I would like to acknowledge Saint George's Greek Community for their warm support. Especially, Fr Adam, Fr. Peter, Helen, Russell, Marisa, Pr. Elizabeth, Alexis, Elizabeth, Dora, Miltos, John, Helen, Kyriakos, Emily, Polly, Paul, Penelope, Lambros, Rex, Helen, Sophia, Geogria, Sophia, Angela, Athena, Artemis, Elizabeth. Also to many of the people that helped me spiritually, Fr. Apostolos, Fr. Chris, Fr. Efstratios, Fr. Charalambos, Pr. Palaiologou, Antony, Konstantinos, Kostantinos, Andrew, Andrew, Stelios.

I wouldn't be able to accomplish anything without the guidance and love of Fr. George, who among other things he taught me that even though science can be impersonal and faceless the scientist should not.

TABLE OF CONTENTS

DEDICATION.....	ii
ACKNOWLEDGMENTS.....	iii
LIST OF TABLES.....	ix
LIST OF FIGURES.....	x

Chapter

1. INTRODUCTION.....	1
1.1 Summarization vs. Generalization of Datasets.....	3
1.2 Problem Statement.....	4
1.3 Research Approach.....	6
1.4 Generalization of Linear Objects.....	8
1.5 Scope of the Thesis.....	11
1.6 Major Results.....	12
1.7 Intended Audience.....	14
1.8 Organization of Thesis.....	15
2. BACKGROUND.....	17
2.1 Video Analysis.....	17
2.2 Spatio-Temporal Generalization and Video Summarization.....	19
2.3 Spatio-Temporal Trajectories and Spatio-Temporal Data Handling.....	24

3. SINGLE OBJECT TRAJECTORY ANALYSIS.....	27
3.1 Definition of Spatio-Temporal Domain and Spatio-Temporal Trajectories.....	27
3.2 Generalization of Single S-T Trajectory.....	30
3.2.1 Selection of Frames Using a Self-Organizing Map.....	31
3.2.2 SST-SOM Enhancement.....	35
3.2.3 Scaled Generalization.....	38
3.2.4 Error Considerations.....	39
3.3 Summary.....	40
4. GENERALIZATION AND GROUPING OF MULTIPLE TRAJECTORIES.....	41
4.1 Registration of Multiple Trajectories.....	41
4.2 Multiple Spatio-Temporal Trajectory Grouping.....	44
4.3 Application of Method.....	48
4.3.1 Grouping of Distinct Road Segments.....	48
4.3.2 Grouping of Interconnected Road Segments.....	51
4.4 Scaled Generalization.....	52
4.5 Summary.....	54
5. TRAJECTORY CLASSIFICATION.....	56
5.1 Introduction.....	56
5.2 Trajectory Dataset Description and Acquisition.....	57

5.2.1 Dataset Formation.....	59
5.2.2 Dataset Attributes.....	59
5.3 Spatio-Temporal Attribute-Aided Classification	61
5.3.1 Spatio-Temporal Group Formation.....	62
5.3.2 Attribute Space Clustering.....	65
5.3.3 Backpropagation Theory.....	66
5.3.4 Backpropagation Towards Trajectory Classification.....	69
5.3.5 Trajectory Reconstruction.....	72
5.4 Algorithm Summary.....	74
5.5 Summary.....	75
6. SUMMARIES OF 2-DIMENSIONAL PHENOMENA.....	76
6.1 Introduction.....	76
6.2 Spatio-Temporal Helix.....	78
6.3 Movement Capturing Towards S-T Helix Construction.....	82
6.3.1 SOM Towards Spine Movement Capturing.....	83
6.3.2 Directional Change Analysis towards Inner Movement Capturing.....	84
6.3.3 S-T Helix Framework Summary.....	87
6.3.4 Static Phenomena.....	88
6.4 The S-T Helix as a Generalization Tool.....	89
6.5 Reconstruction of a S-T Instance.....	91
6.6 Summary.....	92

7. EXPERIMENTS.....	93
7.1 Single Trajectory Summarization Experiments.....	93
7.2 SST-SOM Trajectory Generalization.....	96
7.3 Errors and Noise.....	101
7.4 Multiple Trajectory Summarization and Generalization.....	107
7.4.1 Unrelated Road Segments.....	107
7.4.2 Related Road Segments.....	116
7.5 Trajectory Classification.....	120
7.5.1 Classification of Branches.....	120
7.5.2 Errors and Comparisons.....	129
7.5.3 Three Trajectory Classification.....	138
7.6 Phenomena Summarization.....	139
8. CONCLUSIONS AND FUTURE WORK.....	146
8.1 Conclusions.....	146
8.2 Future work.....	149
BIBLIOGRAPHY.....	151
BIOGRAPHY OF THE AUTHOR.....	162

LIST OF TABLES

Table 7.1:	Generalization parameters for SST-SOM	
	generalization procedure.....	98
Table 7.2:	Grouping results for temporally close trajectories.....	110
Table 7.3:	Grouping results for temporally distant datasets.....	113
Table 7.4:	Grouping results for related road segments.....	117
Table 7.5:	Statistics of formed groups.....	118
Table 7.6:	Further classification.....	129
Table 7.7:	Errors between SOM nodes and trajectories.....	133
Table 7.8:	Trajectory classification results.....	136

LIST OF FIGURES

Figure 1.1:	Focus of presented framework.....	13
Figure 3.1:	Spatio-temporal domain and spatio-temporal trajectories of moving objects.....	28
Figure 3.2:	Relation of image and trajectory space.....	29
Figure 3.3:	Initial, ordering and refinement phases.....	34
Figure 3.4:	Information abstraction in the S-T domain.....	34
Figure 3.5:	Rough generalization and 3D angle estimation.....	36
Figure 3.6:	Localized SOM solutions.....	37
Figure 3.7:	Densification and thinning.....	37
Figure 3.8:	Densification of nodes generalizing the same trajectory.....	38
Figure 3.9:	A priori eliminated errors.....	39
Figure 4.1:	Registration and representation of product summary.....	43
Figure 4.2:	Definition of S-T group surface.....	44
Figure 4.3:	Spatial extend and spatial direction measures.....	45
Figure 4.4:	Temporal extent.....	46
Figure 4.5:	Distribution of space vs. time.....	47
Figure 4.6:	Change of S-T surface due to uncertainty.....	47
Figure 4.7:	Unrelated road segments.....	49
Figure 4.8:	Groups of generalized trajectories.....	51
Figure 4.9:	Interconnected road segments.....	52
Figure 4.10:	Different generalization volumes.....	53

Figure 4.11:	Representation scheme.....	54
Figure 5.1:	Point trajectory dataset.....	57
Figure 5.2:	Possible misclassification of trajectories.....	58
Figure 5.3:	Misclassification of trajectories due to proximity.....	58
Figure 5.4:	a) Input dataset and b) branch formation.....	63
Figure 5.5:	Average attribute mapping of branches.....	63
Figure 5.6:	Correct classification through branch analysis.....	65
Figure 5.7:	Attribute clustering and branch classification.....	66
Figure 5.8:	A neuron-node produces the output y	67
Figure 5.9:	Two hidden layer feedforward network.....	68
Figure 5.10:	Two hidden layer network for classification.....	71
Figure 5.11:	Backpropagation classification outcome.....	72
Figure 5.12:	Outlier elimination.....	73
Figure 5.13:	Carrier formation and further classification.....	74
Figure 6.1:	Movement of a phenomenon in the spatio-temporal domain.....	77
Figure 6.2:	Spatio-temporal helix.....	79
Figure 6.3:	Schematic definition of spine and prong attributes.....	81
Figure 6.4:	Spatio-temporal trajectory of phenomenon.....	83
Figure 6.5:	SOM nodes describing a S-T trajectory.....	83
Figure 6.6:	Translation of all frames to form a non-moving object.....	84
Figure 6.7:	Comparison of sub-areas between consecutive frames.....	85
Figure 6.8:	Densification of cardinality reference system.....	86
Figure 6.9:	Gradual changes towards significant change.....	87

Figure 6.10:	Overview of helix modeling.....	88
Figure 6.11:	Generalization in spine and prongs.....	90
Figure 7.1:	Generalization of a 140 frame trajectory using a)15, b)25, c)35, d) 45 SOM nodes.....	94
Figure 7.2:	Generalization of a 226 point trajectory using a)15, b)25, c)35, d) 45 SOM nodes.....	95
Figure 7.3:	SOM vs. SST-SOM generalization.....	97
Figure 7.4:	SST-SOM generalization in 2-D and thinning result.....	97
Figure 7.5:	Different generalization solutions in a 419 point dataset.....	99
Figure 7.6:	SST-SOM phases for a 225 point dataset.....	100
Figure 7.7:	Node distances in the 3-D ST domain.....	101
Figure 7.8:	SOM solution to a non-erroneous dataset.....	103
Figure 7.9:	SOM solution for: a) 5% occlusion, b) 10% occlusion.....	103
Figure 7.10:	SOM solution for: a) 20% occlusion, b) 30% occlusion.....	104
Figure 7.11:	SOM solution for: a) 50% occlusion, b) 5% noise.....	104
Figure 7.12:	SOM solution for: a) 10% noise, b) 20% noise.....	105
Figure 7.13:	SOM solution for: a) 30% noise, b) 50% noise.....	105
Figure 7.14:	SOM solution for: a) 10% occlusion and 10% noise, b) 20% occlusion and 20% noise.....	106
Figure 7.15:	SOM solution for 30% occlusion and 30% noise.....	106
Figure 7.16:	Medium distance from nodes to input dataset according to errors.....	107
Figure 7.17:	Unrelated road segment input dataset.....	108
Figure 7.18:	MUSTT using (tgroup, sgroup): a) (50,50) and b) (100,100).....	109

Figure 7.19:	MUSTT using (tgroup, sgroup): a) (30,30) and b) (15,15).....	109
Figure 7.20:	MUSTT using (tgroup, sgroup): a) (80,20) and b) (20,80).....	110
Figure 7.21:	Input dataset for temporally distant trajectories.....	111
Figure 7.22:	MUSTT using (tgroup, sgroup): a) (30,30) and b) (50,50).....	112
Figure 7.23:	MUSTT using (tgroup, sgroup): a) (80,20) and b) (20,80).....	112
Figure 7.24:	MUSTT using (tgroup, sgroup)=(150,150).....	113
Figure 7.25:	SOM solutions for leading trajectories.....	114
Figure 7.26:	Snapshots of product summary.....	115
Figure 7.27:	Input dataset for related road segments.....	116
Figure 7.28:	Separation of road segments.....	117
Figure 7.29:	Grouping trajectories in the separated directions.....	118
Figure 7.30:	Snapshots of product summary.....	119
Figure 7.31:	Distant and entangled input points that form two trajectories.....	120
Figure 7.32:	Classification result for distant trajectories.....	122
Figure 7.33:	Branches and attribute clustering.....	123
Figure 7.34:	BP output and performance, and after BP classification.....	123
Figure 7.35:	Classification when one attribute overlaps.....	124
Figure 7.36:	Classification when both attributes overlap.....	125
Figure 7.37:	Clustering of branches in the 80, 80 overlap case.....	126
Figure 7.38:	Input dataset of spatio-temporally close trajectories.....	126
Figure 7.39:	Classification of adjacent trajectories A.....	127
Figure 7.40:	Classification of adjacent trajectories B.....	128
Figure 7.41:	Removal of distant points.....	130

Figure 7.42:	SOM further classification for first set of trajectories.....	131
Figure 7.43:	SOM further classification for second set of trajectories.....	132
Figure 7.44:	Classification through attribute space clustering.....	134
Figure 7.45:	C-means classification for second dataset.....	135
Figure 7.46:	Surfaces of accuracy for different classification steps, first trajectory pair: a)After c-means, b) After BP, c)After SOM, second trajectory pair: d)After c-means, e)After BP, f) After SOM.....	137
Figure 7.47:	Input dataset, branch formation and clustering.....	138
Figure 7.48:	Resulted classification after the BP algorithm.....	139
Figure 7.49:	Input dataset and SOM generalization of the spine using variable number of nodes.....	140
Figure 7.50:	Phenomenon cardinal segmentation in different temporal instances.....	141
Figure 7.51:	Generalization under different variables A.....	142
Figure 7.52:	Generalization under different variables B.....	143
Figure 7.53:	Generalization under different variables C.....	144
Figure 7.54:	Generalization under different variables D.....	145
Figure 8.1:	Lifeline representation of moving objects and attribute resolution....	149

CHAPTER 1

INTRODUCTION

Information is often conveyed through collections of data that may form large and diverse types of datasets. Advances in sensor technology have resulted in easier collection of valuable multitemporal geospatial information. Reliable satellite, aerial and ground imagery are readily available on demand or periodically. Beyond imagery, the availability of inexpensive and compact positioning and capturing devices provides a plethora of datasets. People use datasets for complex analyses including decision-making processes.

The need to sift through large datasets is evident for varying-level information analysis. Data may be represented at various scales of space and time. For example, finer detail is needed when the user is interested in the phases of the detailed construction of a building than in the generalized expansion of a city.

Various types of datasets may need different methods for their compact representation. Examples include video (Zhang et al., 1997), audio (Christel et al., 1998), and standard maps (Brassel and Weibel, 1988). This thesis focuses on motion imagery datasets. Within the context of this thesis we use the term motion imagery (MI) to refer to

high-resolution (spatial and/or temporal, aerial or ground-level) digital video and still image sequences, captured either by mobile sensors roaming a scene or by a network of fixed sensors, each monitoring a particular area. MI datasets are used extensively as a powerful communication medium primarily because of the rich content they hold. Frequently, MI sets as opposed to still images include additional information as they depict a near continuous data flow.

The development of compact representation schemes proves beneficial for the search, retrieval, interchange, query, and visualization of the information included in datasets (Stefanidis et al. 2001). Compactness refers to the compressed or selective representation of the original to adequately communicate the included information, according to the user and application. Compact representation supports better browsing, dissemination, and further analysis of the corresponding datasets. More specifically for MI datasets:

- Regarding *browsing*, the vast collection of MI datasets on the web and in MI libraries makes the use of compact representations highly important, as they will reveal only the essential and significant parts of the original dataset. Some early efforts in video processing research are addressing creation of information-inclusive abstracts based on image content, image attributes, video structure, voice and text properties. More details on relevant current research are provided in the second chapter of this thesis.
- *Dissemination* of information is also important when the user desires to determine in a short time what took place in a scene, movie, broadcast, or MI in general

(Smith and Kanade, 1995). In addition, shifting through information at a fine detail may make it more difficult to identify low frequency trends. For example, one might be concerned with traffic within an area throughout the day rather than the movement of every single car over an entire day. Coarse resolution datasets communicate better this type of information. Accordingly, for visualization purposes MI compact representations are used to portray the significant parts of a dataset through a concise yet expressive scheme.

- Construction of compact representations may support *further analysis*. Such analysis is essential in order to facilitate behavior-related queries or to examine and reveal possible associations between different datasets and the phenomena they describe (Stefanidis et al., 2001). Efficient analysis and modeling of phenomena captured in MI datasets supports comparison of behavioral patterns and detection of similarities. For example, a MI about weather changes throughout a year could be compared with urbanization tendencies or pollution changes.

1.1 Summarization vs. Generalization of Datasets

Compact representation of datasets is manifested through the use of *generalization* and *summarization*. In the context of this thesis these abstraction tools are defined as follows: Generalization is the process under which an input data of a specific type is transformed into a compact representation of the same type. On the other hand, summarization transforms a series of input data collections of various types into a new representation that combines information from the input datasets.

Frequently, generalization refers to the basic elements that are included in a summary, while the summary is the product combination of generalization and additional products. For example a map may include roads that are represented as lines and bullets that describe major cities. The map as a resulting combination of a series of supportive data descriptions is a summary of the area that represents. The line representations of the roads at various scales are a product of generalization. Usually the generalization products are smaller than the original dataset. The summaries are not necessarily smaller, but should communicate better a large amount of information compared to the original dataset components.

1.2 Problem Statement

The processing, analysis, and management of MI data present well-known challenges, mostly associated with the size and complexity of the space represented (Ardizzone and Hacid, 1999). These challenges are mostly related to large storage requirements, immense processing volumes for dataset analysis, as well as the highly diverse content and themes these datasets may include.

MI datasets are comprised of frames that provide 2-dimensional views (x, y) at discrete time intervals. By piling individual frames on top of each other we form the 3-D spatio-temporal domain (*S-T domain*) with time (t) serving as the third axis. The movement of an object within the S-T domain is manifested as a set of classified points over time. Treating the S-T domain as a near-continuous representation of reality, the trajectory of an object defines a pathway within this 3-D domain, by sequentially

connecting all positions depicting the same object over time. Within the context of this thesis this path is termed as a spatio-temporal trajectory (S-T trajectory). Work on spatio-temporal analysis is evident in Hagerstrand (1970) where the basis of time geography is defined. Geospatial lifeline formation towards spatio-temporal querying and analysis is also seen in Hornsby and Egenhofer (2002).

Motivation is drawn from a family of MI datasets that originates from monitoring applications, where objects move and interact throughout a relatively stable background scene. Some of these applications include vehicles moving in an urban scene, fire spreading in a forest area, etc. Our argument is that crucial elements for MI summarization are the ones describing the behaviors of objects as they change their location and/or shape over time. Analysis of such datasets takes place in order to produce brief summaries of their content.

For summarization purposes, substantial amounts of data and/or information included in MI datasets can be truncated or compressed, while other parts of this information may be defined as significant and should be identified and emphasized. S-T trajectories may be derived from a series of methods including tracking algorithms, GPS receivers or feature extraction analysis. The resulting data are termed as source data and are represented by sets of coordinates that may include, in addition to spatio-temporal location, size, or even color of the extracted objects. In addition, S-T trajectory datasets may include gaps, and often suffer from distortions caused by noise and various errors. The tackling of errors that may be included in the dataset is essential for its accurate representation and further analysis.

1.3 Research Approach

Towards the summarization of monitoring MI datasets we need to address some tasks that are related to the input data comprising the datasets. S-T trajectories contain implicitly substantial information for the behavior of the moving objects they represent in terms of movement and deformation. The presented approach allows the retrieval and analysis of this behavioral information.

The formed S-T trajectories become the base generalization elements. We exploit the geometric properties of S-T trajectories describing the movement of objects, in order to extract important information towards their generalization. Individual trajectories are analyzed to identify critical points, denoting instances where the object's geometry changes (e.g., accelerates, or makes an abrupt turn, etc).

The goal of this work is to support the generation of compact MI datasets that would serve as summaries of the original input. Towards this goal, the primary objective of this thesis is

- to provide the foundation of MI content summaries through generalization of spatio-temporal trajectories describing geometric change, namely movement and deformation.

The primary objective is complemented by the need to support multi-scale generalization within this framework that forms the second objective, which is

- to support dynamic representation both in generalization volume and densification of information.

By multi-scale generalization we indicate two kinds of dynamic information representation. First, dynamic representation refers to the fact that the sampling of the original dataset is based on those parts that convey significant information rather than a stable temporal increment. Significance is defined as the variation of movement and/or deformation of the shape of the moving objects included in the dataset. The second representation attribute refers to summary duration. According to user needs and application, more or less detail of the product summary can be realized.

In addition, tackle possible data errors have to be tackled:

- Data without object correspondence

Sometimes, there is no explicit information that relates each set of coordinates to the object that it belongs to. Thus, detection and separation of object S-T trajectories is required. When data coordinates are mapped as distant in the S-T domain their separation is evident. However, when the data form entangled trajectories their separation may not be trivial and thus a classification procedure has to differentiate them.

- Occlusion and noise

Occlusion in tracking procedures or unavailability of data entries in some instances leads to gaps in the dataset. In addition, noise due to such reasons as tracking misclassification is also evident. Again the analysis of the S-T

trajectories should tackle or integrate a solution for the minimization of such errors.

Thus, additional objectives are:

- to detect and classify neighboring moving object trajectories where there is no information about the object they belong to.
- to handle common errors anticipated in tracking analyses, namely occlusion, noise, misclassification etc.

These objectives refer to commonly anticipated problems inherent in moving object image tracking. In addition, summary data should be such that they can support further analysis like retrieving, querying and similarity matching between different datasets. For example a MI depicting flood occurrences in an area and another depicting land use may be compared to detect similarities and relations.

1.4 Generalization of Linear Objects

Generalization is a well-known cartographic process towards effective map construction that includes some interrelated tasks. According to the application, scale of the map, and quality of input data, these tasks are formed accordingly (Muller et al. 1995, McMaster and Comenetz 1996). More specifically:

- Simplification

Simplification refers to the selection of a subset of representative elements of the same type and the exclusion of some others. In the context of this thesis this term will be referred as clustering.

- Taxonomy

Under this task a series of cartographic elements of the same type are grouped together and are represented through a single cartographic element according to a similarity measure. This task in this thesis is referred as grouping.

- Visualization

The set of graphic representation procedures to convey the map's elements is an essential part of the map formation.

- Deduction

This is the logical deduction of information and we will refer to it as further processing.

There is a series of well-known algorithms for generalization of linear cartographic elements. Most are based on a simplification procedure that selects a subset of points from the points that comprise the line. Some of the algorithms select points based on stable increment sampling and thus result in poor generalization without ample representation of the possible variance areas. On the other hand, the Douglas-Peucker (Douglas and Peucker, 1973) algorithm varyingly samples the line points according to the desirable representation precision.

S-T trajectories are three-dimensional and as discussed before they may include errors. In addition source coordinates often accompany the S-T trajectories. Nevertheless, the above-mentioned algorithms do not handle 3-D data collections like S-T trajectories. In addition, the errors inherent in the data may significantly alter the resulting generalization. Since they are based on single point sampling, a single error might lead to poor representation. Finally, they do not readily tackle multi-dimensional, unclassified and source data that may include composite spatio-temporal coordinates.

In order to address these additional complications we utilize the Self-Organizing Map (SOM) algorithm. The SOM technique is a neural network that represents an input dataset of a given dimension into a fewer dimension dataset and thus is used to perform generalization of lines. The generalization produces a set of nodes that may not be part of the original data. The product output space is indicative of intrinsic statistical features contained in the input patterns. According to the selection of the number of nodes multiple generalization levels are achieved. Possible errors are often efficiently tackled through the inherent generalization properties of the technique that is based on groups of points rather than single points. On the other hand, according to experimental results, the geometry of the input space is not always represented sufficiently, especially when geometrically complex lines are anticipated. Furthermore, the algorithm variables, which influence considerably the solution, are not automatically selected.

Therefore, a new SOM-based method is introduced to adequately approximate S-T trajectories called Spatio-Temporal Trajectory (SST) SOM. This approach offers the advantage of relative invariance with respect to the selection of the initial algorithm

variables, and most importantly approximates better the geometry of complex S-T trajectories.

Thus, the hypothesis is:

“The SST-SOM technique approximates and classifies more accurately the S-T trajectories than the typical clustering and classification techniques.”

Overall, by fulfilling the stated hypothesis we will provide an approach to manage dynamic scene analysis at higher levels of abstraction, and to model and communicate concisely the movement behavior of moving objects.

1.5 Scope of the Thesis

The fulfillment of the objectives of this thesis lies under the assumption that the processed MI datasets display mobile objects with a relatively stable background. Such datasets can be produced by a fixed sensor monitoring a scene, or by moving aerial sensors, like unmanned aerial vehicles, or by sensors periodically revisiting a scene (e.g., satellites). Thus, it is assumed that classification or coarse tracking has identified in each frame (or in selected frames) the approximate outline of mobile objects. In addition, the availability of GPS receivers attached to vehicles is often anticipated in today's transportation technology. Therefore, some required a priori analysis is assumed to exist and some other tasks are beyond the scope of the thesis. Specifically, this thesis does not address:

- Tracking techniques and analysis in general,
- detailed visualization of the product summaries,

- design of a common metadata framework for summary creation and comparison,
- incorporation of the product summaries and defined data types in a database environment.

These aspects are discussed only partially. In addition, we focus only on *geometric properties* of moving objects as opposed to *semantic*. There is substantial complementary work dealing with semantic properties of geographic data (e.g., Hornsby, 1999). Additional temporal increments of interest resulting from query and similarity analysis may be introduced and included in the formed summaries. Query support and similarity matching through formalized data descriptors is introduced and exploited in (Stefanidis et al., 2001).

For a single moving object, the generalization of its S-T trajectory is accomplished through the use of self-organizing maps and their variation STT-SOM. Multiple moving objects introduce more challenges towards the grouping of the formed S-T trajectories. A spatio-temporal grouping technique is introduced, based on moving geometric surfaces. If needed, classification towards the distinction of possible entangled trajectories is tackled by using additional object attributes. Finally, phenomena generalization is accomplished again through self-organization theory. In figure 1.1 the shaded area displays the focus of the analyses addressed in the thesis.

1.6 Major Results

Along with the generalization objective a series of research challenges are tackled throughout this thesis. More specifically:

For a single trajectory, generalization is accomplished using the SST-SOM technique. Compared to a standard SOM process, the STT-SOM approach offers the advantage of invariance to the selection of the initial number of nodes and the additional SOM attributes. Since many local SOMs take place, the initial attribute contribution remains localized where it performs more adequately.

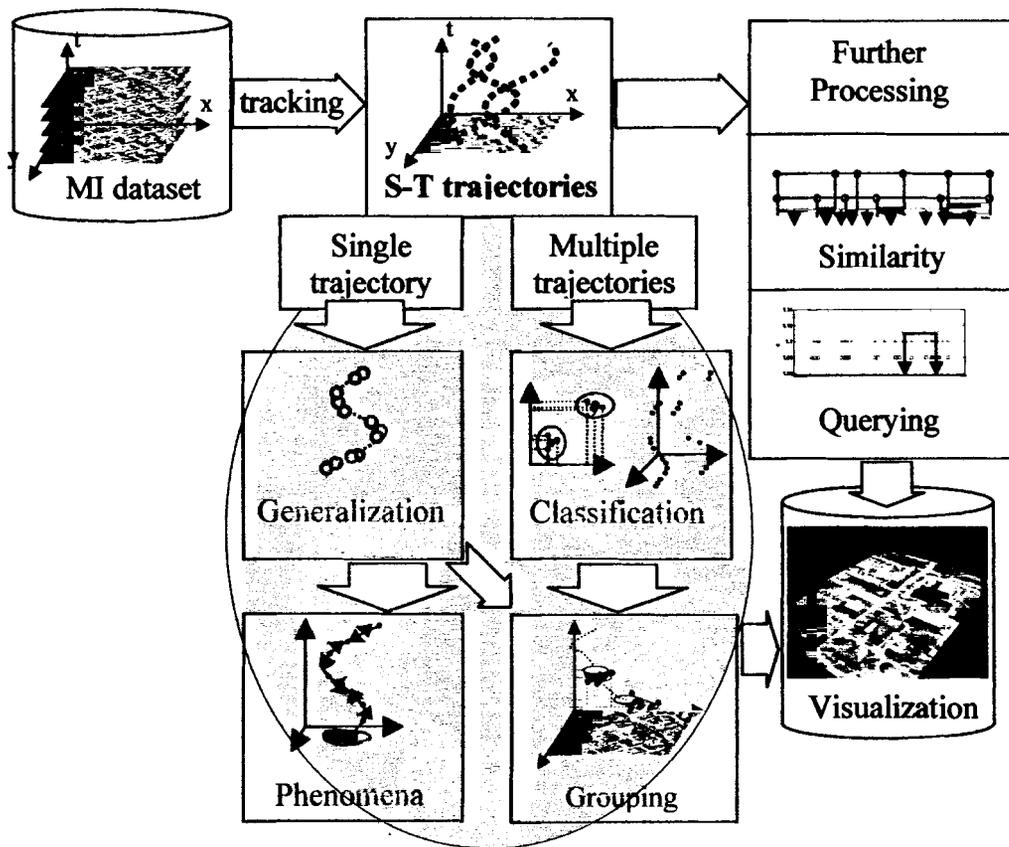


Figure 1.1: Focus of presented framework.

To proceed from a single trajectory into multiple ones, we register the already generalized trajectories by imposing sets of grouping rules. The grouping is performed in

the generalized S-T trajectories and thus considerably fewer computations take place. The resulting generalization supports dynamic scaling to provide more or less detail in the product summary. Possible summary visualization schemes portray the potential information dissemination capabilities of this method.

Classification of multiple trajectories when their ID is not known is an essential task when S-T trajectories are entangled throughout the S-T domain. A multiple step technique is introduced that combines neural networks, geometry and attribute management, in order to offer high confidence in data clustering.

Finally, we investigate the summarization of two-dimensional phenomena evolving over time. The concept of S-T helix (Stefanidis et al., 2002a) is used as compact representation of spatio-temporal events. The helix model is comprised of SOM movement nodes (spines) and cardinality shape-change descriptors (prongs) that generalize the input dataset at varying scales.

Throughout this thesis our focus is the generalization of S-T trajectories. The resulting schemes and supported summaries are not only the basis for a visualization product but may also support further processing.

1.7 Intended Audience

This thesis is intended for scientists and researchers dealing with spatio-temporal data structures, MI datasets and video processing. Related research interests are encountered in moving objects and spatio-temporal databases. The findings of the thesis may form a basis for the establishment of a behavioral analysis framework to accommodate querying

and statistical reasoning. Image tracking, change detection, and data clustering disciplines can also benefit from this work. The anticipated research problems and approaches that are discussed throughout the thesis stand between data analysis and image processing communities.

1.8 Organization of Thesis

The topics that the thesis addresses flow according to single or multiple S-T trajectory analysis, with or without information about the objects they belong to, and with or without consideration of the outline changes of the objects they represent.

More specifically the rest of this thesis is organized as follows:

The second chapter presents a literature review of related work regarding video data, MI summaries, moving object manipulation and spatio-temporal data.

The third chapter presents the theoretical basis of this work, where a single object's trajectory is analyzed and dynamically generalized according to self-organizing map (SOM) theory. A new hybrid approach for enhanced distribution of generalization nodes is proposed and compared to standard SOM.

The fourth chapter copes with multiple object trajectories. When few objects are included in the dataset, registration of single trajectories is used to form the summary elements. However, when more objects are present, a grouping procedure using spatio-temporal surfaces is presented.

The fifth chapter discusses classification and separation analysis between multi-dimensional trajectories, which do not include any information about the objects they represent. Geometric, neural network and transformation analysis is used towards the classification of such datasets.

The sixth chapter deals with the summarization of phenomena that have a substantial spatial extent and change their outline through time. Combined self-organizing map and geometric considerations provide the framework to produce efficient summaries.

In the seventh chapter we introduce the implementation of the issues addressed and analyzed in the previous chapters. Experimental results, comparisons, error metrics, and evaluation properties are also included.

In the eighth chapter we present conclusions as well as discussions on future work.

CHAPTER 2

BACKGROUND

2.1 Video Analysis

The increased availability of MI datasets is bringing forward interesting research issues. Types of MI datasets that are used extensively in research are video data collections. Video data processing and analysis present well-known challenges. Problems inherent in MI and video dataset exploitation include the highly diverse content and themes they can include, the large space they require for storage, and consequently the immense processing time for their analysis. Accordingly, we are faced with the need for efficient methods to model video content and to support content-based queries.

Current research in video databases includes database architecture, data modeling, segmentation and querying. The use of hierarchical data structures provides a higher level of abstraction and leads to computationally less expensive management. For video analysis Oomoto and Tanaka (1993) introduced object-oriented attributes to assist video database management. The objects included in the database share descriptive attributes that facilitate inheritance properties. Hacid et al. (2000) define relationships between contained semantics, where video frames and objects depicted in frames are manipulated

at the same level. Content modeling of video data is discussed in Ardizzone and Hacid (1999), where an object layer is complemented by a schema layer that includes the attributes of objects, while queries are designed based on this framework. A hierarchical video object modeling is discussed in Zhong and Chang (1997), where extraction, indexing and classification of objects take place. In Chang et al. (1998) semantic visual templates are introduced to support video queries based on users' conceptual designs.

Based on indexing schemes, the two most common techniques used in the video database community are *segmentation* and *stratification*. According to segmentation, a video dataset is segmented into its scenes or sub-scenes, and each temporal unit that includes the scene is annotated according to its content. On the other hand, in Aguierre-Smith and Davenport (1992) the stratification method for indexing video data is proposed, using each object or event of interest as independent throughout the video sequence. This technique is object-based, where an object is any significant element accompanied by its temporal duration and description. Densification of information is supported, and overlap of descriptions according to the content of the video dataset is evident. In Hacid et al. (2000) an enhanced stratification technique is introduced where a set of temporal segments is represented by one description. This technique resembles timeline descriptions. Video analysis at the object level is introduced in Hibino and Rundensteiner, (1995) where a visual query language and visualization results for data trend exploration are introduced. The relationships between events are fed into the queries using annotations.

Segmentation of video datasets based on scene detection is an area of substantial research concentration. Histograms of frames or grayscale pixel differences are used. Pixel-based techniques may detect false scene changes due to the localization of the approach. Distribution of color often proves more efficient. In Taskiran and Delp (1998) a histogram-based technique is introduced for scene change detection. Furthermore, Meng et al. (1995) detect scene changes not in the original video but in an MPEG-compressed video dataset. Segmentation of video objects (not scenes) is seen in Mao and Ma (1999). The segmentation method is based on motion in order to extract video objects. Spatial and temporal segmentation both take place using fuzzy c-means clustering techniques and the result is complemented by user semantic interaction. Vasconcelos and Lippman (1997) go further and characterize the content of video. From frame space they transfer to feature space, where shot segmentation and estimation of activity takes place. Finally, categorization according to content (violence, comedy, drama, etc.) is performed.

Because of the diversity in themes, a predefined video database structure using pre-specified objects and attributes is very difficult to be defined. In this thesis, frames and objects are organized and related interchangeably at the same information level, while a type of stratification technique for data representation is used to handle object movement.

2.2 Spatio-Temporal Generalization and Video Summarization

Generalization is a well-known cartographic process and substantial research has been conducted both in classic and modern cartography. Issues include line and region

generalization in model or visualization space and using geometric, fractal and other processes (Jones and Abraham, 1986; Brassel and Weibel, 1988; McMaster and Comenetz 1996; Muller et al., 1995; Stell and Worboys, 1999; Nakos, 1997).

In many cases the result of cartographic generalization of 2-d objects resembles pyramid resampling techniques used in image processing where the image pixel size varies in order to achieve adjustable scales. In the content of this work, generalization is analogous, yet the step of scale change is not stable but relies on significant information that is implied by the area and movement attributes of the moving phenomenon. This information and the quantification of its significance is captured by the algorithms described throughout the thesis.

The development of concise representation schemes is essential for the search, retrieval, interchange, query, and visualization of information included in video datasets. Efforts towards this direction include attempts to summarize video by selecting discrete frames at regular temporal intervals (e.g., every n seconds). However, such an approach typically fails to capture and represent the actual content of the original video dataset. In common monitoring applications it is usual to expect long video segments where few objects move in front of a rather stable background. In geospatial applications such datasets may be produced for example by a sensor on an aerial platform flying over a scene. Alternatively, they may be synthesized by merging multitemporal raster datasets like satellite imagery to describe a phenomenon evolving over time (e.g., a flood).

Common approaches to video analysis tend to emphasize shot detection and the selection of a few representative frames, thus failing to capture the semantic content of a

video (Rui et al., 1998). Summaries can be exploited to support content-based video retrieval. The product summaries can have the form of posters, new videos, synthetic images, or a series of images. Summaries may take advantage of additional features inherent in most video datasets, like text or captioning or sound. For instance, the sound of an explosion is considered important in order to retrieve an action scene from a video dataset.

Work relevant to the approach introduced in this thesis has been performed on the analysis of visual and speech properties to construct “skim” video synopses (Smith and Kanade, 1995). This “skim” video is constructed by merging segments of the original video. Significance is based on the number of objects present in a scene, the words accompanying the scene, the structure of the video scene, and possible text. In Pope et al. (1998) creation of mosaics for the stable background takes place while trajectories of moving objects are extracted. What defines the significance is the maximum number of moving objects, when the camera pans too much, or when image registration fails.

Vasconcelos and Lippman (1998) create summaries based on the dominant motion exhibited in the spatio-temporal domain. The summary has the form of an image where the dominant object is stable and other objects are depicted as blurred versions that demonstrate the motion that took place. This result is not very descriptive in some situations. For example, in monitoring applications this representation would merely yield the full trajectories of the moving objects on the stable background. Lienhart (2000), uses a time-based approach to summarize home videos by using date and speech properties, yet neglecting content of videos.

Summaries can be formulated by extracting keyframes from each scene shot (Yeung and Yeo, 1997; Uchihashi et al., 1999), or may consist of selected important scenes (Christel et al., 1998; Oh and Hua, 2000). In Yeung and Yeo (1997) video posters are proposed as alternatives to describe story content and to represent the original video. In Uchihashi et al. (1999) creation of pictorial video summaries according to defined importance is proposed. Keyframes are selected and resized according to significance. Significance is defined using visual aids like titles, human figures and text to automatically caption summaries. However, interpretation of keyframe selection-type summaries may be difficult because there is no connection between still frames, and the resulting product since it is formed by snapshots and does not adequately describe the original video. In Christel et al. (1998) video skims are videos communicating the content of the represented video. Sometimes there are abrupt transitions between themes/parts of the skim summary. In Oh and Hua, (2000) the summary is based on interesting scenes depending on the purpose of the summary. The user chooses a few scenes, and according to his/her selection, important scenes like the ones selected are revealed. This technique resembles similarity matching and it is user-triggered.

In Gong and Liu (2000) the creation of summaries is based on the content values of frames and not the shot boundaries. Content in this case is represented by color histograms. Acquisition of scenes with variable volume of change takes place, and the number of frames is selected accordingly to represent the final summary. Similarly, content summaries based on color differences is seen in Zhang et al. (1995, 1997). In Russell (2000) a library-like approach is introduced, where pre-selected actions match the

input video, and pre-selected summary design patterns are also used to define the product summary. Summarization alternatives include the use of image templates, statistical features, and histogram-based retrieval and processing (Chang et al., 1998). Sawhney and Ayer (1996) investigate the dominant motion in order to distinguish between moving parts and stable parts of the video sequence and produce video mosaics. In Pfeiffer et al. (1996) movies are summarized mostly based on content using text selection of representative scenes. Significance is based on motion, which is estimated by the radiometric difference of frames, and from the “mood” of the scenes, which is estimated by color and audio properties.

Summaries are sometimes large in size because each segment of the original video is represented. In addition, some representations are unconnected, making interpretation of video content blurry. Most of the techniques presented above select segments from the original video dataset, namely short videos and frames. As it will be demonstrated in this thesis, we provide the basis to construct new videos that form the summary of the original one, using the background as stable and portraying synthetic representations of the objects’ movement behavior. Summaries, as exploited by most of the research community, are the final product of the various analyses, mostly satisfying visualization and dissemination purposes. In our case, summary dataset creation is an intermediate step towards similarity assessment and query support. Finally, our proposed summary creation supports generalization in multiple scales according to user and application needs.

2.3 Spatio-Temporal Trajectories and Spatio-temporal Data Handling

In the trajectory domain, Pfoser and Theodoridis (2000) and Theodoridis et al. (1999) provide a spatio-temporal synthetic dataset generator to simulate movement trajectories. In complementary work Pfoser et al. (2000) analyze index schemes for moving objects using tree structures. Bradshaw et al. (1997) describe a technique for real time trajectory acquisition. Kalman filters are used to describe the motion. Visual delays are described using prediction and they are recovered in order to provide smooth trajectories. Bremond and Medioni (1998) and Medioni et al. (1998) provide a system to extract and recognize moving objects and they use the formed trajectories towards image sequence understanding and scenario recognition. They classify the movement of objects using natural language verbs. Wong et al. (1999) use motion segmentation to extract moving objects and index them based on the trajectories of their center of gravity. Sahouria and Zakhor (1997) provide a framework for the formation of trajectories while queries by user-drawn trajectories are supported.

Topological spatial relations support spatial analysis with focus on relations in a higher information level where further processing is accommodated. Related work in topology reasoning includes processes based on the 9-intersection model, modeling of gradual changes of topological relationships, combined models for direction and topology, and more (Egenhofer and Al-Taha, 1992; Bruns and Egenhofer, 1996). Papadias et al., (1995) introduce minimum bounding rectangles to relate objects and use

R-trees to index their relations. Sorting data according to their spatial occupancy through tree structures is a promising data manipulation scheme (Samet, 1990; Sellis et al., 1987).

In temporal logic Allen and Ferguson (1994) provide the framework of the defined temporal algebra and event-action representation. Keogh and Pazzani (1997) describe the comparison of time series not in a Euclidean-based distance but with the use of dynamic time warping (e.g., non-linear alignment), complemented by a piecewise linear representation of the data series. Agrawal et al., (1993) introduce discrete Fourier transforms of time-series and index them by using R-trees. Equal length of query and database is required for the similarity comparison to take place. Lee et al. (2000) also use minimum bounding rectangles to describe trends in multidimensional data. In Combi (2000) modeling of temporal multimedia data takes place. Frames of videos, temporal visual and textual data are all based on object-oriented video database architecture. Vazirgiannis and Wolfson (2001) query and index moving objects. Sistla et al. (1997) and Wolfson et al. (1999) use a future temporal logic for modeling and querying moving objects. In Agrawal et al. (2000) indexing schemes are introduced in order to support queries of the type “find all points present in a given rectangle area at a given time.”

Work on indexing animated objects is reported in Kollios et al. (1999; 2001), while Tao and Papadias (2001) propose a framework for indexing and querying spatio-temporal data by constructing new tree structures. Chomicki and Revesz (1999) use moving vertices and geometric transformations to portray change in order to describe point and region change. Similarly, database indexing and querying of complex moving

spatial objects is seen in Forlizzi et al. (2000). Retrieval of snapshots towards modeling of continuously moving objects is presented in Tossebro and Guting (2001).

In this thesis we model spatial objects, such as points and regions, by dealing both the change that occurs in movement according to the image coordinate frame and the inner movement that alters the shape of the objects. The produced summaries support manipulation and similarity matching of both one-dimensional and multidimensional sequences of data. The analysis of the framework under which similarity assessment is achieved is not discussed in this thesis but it has been investigated in Stefanidis et al. (2001a, 2001b).

CHAPTER 3

SINGLE TRAJECTORY ANALYSIS

In this chapter we present the basis of our proposed analysis towards MI summarization. We present our approach for the generalization of a spatio-temporal trajectory generated from a single moving object.

3.1 Definition of Spatio-Temporal Domain and Spatio-Temporal Trajectories

MI datasets anticipated in monitoring applications are comprised of frames that define a 3-dimensional space where each frame is registered at the time (t) of its acquisition. The spatial dimensions (x,y) of this space coincide with the image coordinate systems of each individual frame. In that sense, individual frames pile up on top of each other to form the 3-D spatio-temporal domain. A moving object in the S-T domain is depicted by a set of points moving over time. Thus, the trajectory of an object is defined by connecting all positions depicting the same object over time. The formed spatio-temporal trajectory begins at point (x_0^i, y_0^i, t_0^i) and ends at point (x_n^i, y_n^i, t_n^i) , where (x_0^i, y_0^i) are the image coordinates of object i at the time t_0^i when it first appears in the MI stream, and (x_n^i, y_n^i)

are the corresponding coordinates at the time t_n^i when it moves outside the MI stream.
(Figure 3.1).

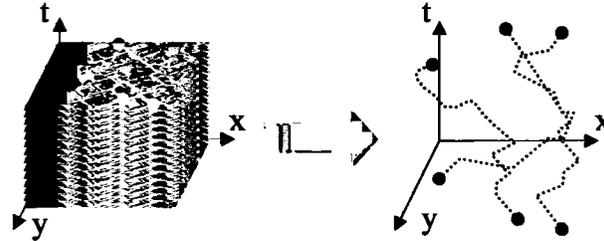


Figure 3.1: Spatio-temporal domain and spatio-temporal trajectories of moving objects.

The key assumption is that for a given MI dataset a classification or coarse tracking has identified in each frame (or in select frames) an approximate representation of mobile objects. This information may be corrupted by various types of errors like occlusions, noise, and misclassified pixels. A spatio-temporal trajectory is produced by sequentially linking all this information and includes inherently a portion of pertinent information for the behavior of a moving object over the corresponding time interval. However, some of this information is redundant and should be truncated for improved analysis, storage, and communication, while some of this information is significant and should be identified and emphasized. In order to capture the significant portion of this immense data flow, an efficient and accurate spatio-temporal generalization of these trajectories is required. The proposed approach proceeds by analyzing the spatio-temporal trajectories of moving objects.

Instead of points, objects can be described by pixel patches, which depict the object's spatial extent in each frame. These patches can be represented by points/pixels of the center of mass of the patch. E.g. for a video sequence of one frame/second frame rate, the movement of two objects is monitored. The first object is equipped with a GPS receiver and the second object is tracked through any of the known tracking algorithms. The area of monitoring is captured by frames of (1000*1000) pixels. In each second of the duration of the video dataset we acquire a GPS coordinate (x,y) pair describing the first object and an image patch describing the second one. GPS and image coordinate systems are linked while the center of mass of the patch is calculated and approximated by a pixel. The summation of all these pixels and their correspondence to the object they belong to, forms the S-T trajectory input dataset. These trajectories are the base units that the described analyses are based on. (Figure 3.2).

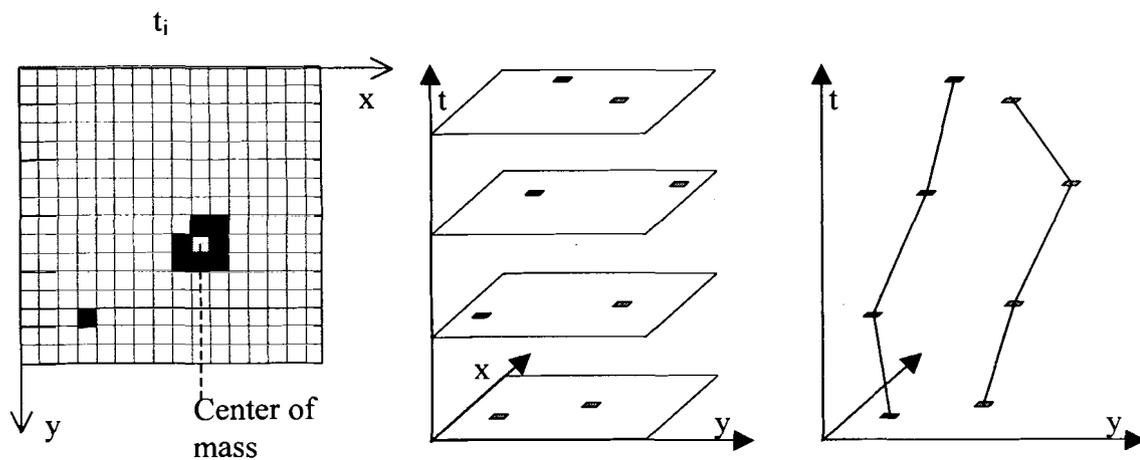


Figure 3.2: Relation of image and trajectory space.

The spatial coordinates $(x,y)_i$ of an object belonging to a trajectory at each temporal instance t_i that a frame exists, correspond to the pixel coordinates of the digital

frame. Therefore image and trajectory spatial coordinates are based on the same reference system.

3.2 Generalization of a Single S-T Trajectory

A spatio-temporal trajectory includes essential movement information that portrays the behavior of a moving object. Towards the generalization of the S-T trajectory we select representative points from the trajectory, which correspond to MI frames. The selection of representative frames is based on the segmentation of trajectory lines into break points termed “nodes”. The nodes are distributed dynamically to capture the information content of regions within the above mentioned 3-D S-T domain. More nodes are assigned where the trajectory presents S-T breakpoints, and fewer nodes are assigned to segments where the spatio-temporal behavior of an object is smooth. Node placement is based on concepts of self organizing maps from neural network theory.

Trajectories can be perceived as paths in the spatio-temporal (S-T) domain. Therefore, we have extended a methodology originally intended to perform road extraction from satellite imagery to capture and generalize spatio-temporal trajectories. The road extraction approach is based on the use of self-organizing maps (SOM) (Kohonen, 1982; Kohonen, 1997; Haykin, 1999) to extract road centerlines (Doucette et al., 1999; Doucette et al., 2001). The SOM technique not only links pixels into road centerlines, but also distributes nodes along the road to generalize the extracted centerline.

Standard SOM algorithms tend to function well on linearized road segments in aerial imagery, as these segments are reasonably smooth lines. However, spatio-temporal trajectories tend to include abrupt variations (e.g. an object may change its velocity and orientation very often in a limited area). This makes the use of standard SOM techniques inadequate for spatio-temporal generalization. To overcome this problem we have developed a hybrid SOM-geometry method termed Spatio-Temporal Trajectory Self-Organizing Map (SST-SOM). Compared to a standard SOM process, our SST-SOM approach offers two important advantages, namely invariance to the selection of the initial number of nodes, and the ability to selectively densify or thin to better capture the complexity of content of the processed dataset.

3.2.1 Selection of Frames Using a Self-Organizing Map

The self-organizing map (SOM) algorithm is a nonlinear and nonparametric regression solution to a class of vector quantization problems, which is used as the method for information abstraction. The SOM belongs to a class of artificial neural networks (ANN) characterized by unsupervised and competitive learning. The *network space* exists independent of the input space, and the objective of the SOM is to define a mapping from the input space of dimension m onto the network space of dimension d , where $m \geq d$.

Its unsupervised character is perceived through the automation of the procedure without any a priori human interaction on the input dataset. The procedure is based on competition between the set of nodes, which attempt to best map the points of the input space. The goal of competitive learning is to reward the node that optimally satisfies a similarity measure between a given input point compared against all nodes. Essentially an

iterative clustering technique, the SOM uses a shrinking neighborhood function over the cluster centers to determine the sequential adjustments that update the clusters positions.

The outcome of this analysis serves two purposes. First a concise and inclusive representation of the input space is performed for visualization purposes. Next, dynamic generalization is achieved through the variation of the solution variables, which yield more or less detailed description of the input space. More nodes provide more detailed representation while fewer nodes yield a more concise description.

To demonstrate, let $p(X)$ describe a probability density function in \mathfrak{R}_1^2 for the input vector,

$$\mathbf{X} = [x, y]^T \in \mathfrak{R}_1^2, \quad (3.1)$$

Each connection or *synapse*, between a component of X and any single node k located in network space has an associated weight. The components of each weight vector are defined in \mathfrak{R}_1^2 , which has the same dimensionality of X , or,

$$\mathbf{W}_k = [w_{k,x}, w_{k,y}]^T \in \mathfrak{R}_1^2. \quad (3.2)$$

By initializing the contents of W in \mathfrak{R}_1^2 for each node, the goal of competitive learning is to reward the node k that optimally satisfies a similarity measure between a given X compared against all W_k . Using the L_2 (Euclidean) norm as the similarity metric, a *winning* node q is determined as,

$$\text{node } q = \arg \min_k \| \mathbf{X} - \mathbf{W}_k \|, \text{ for } k = 1, 2, \dots, K. \quad (3.3)$$

where K is the total number of nodes in \mathfrak{R}_N^1 . The appropriate weight vectors are updated sequentially for each input sample according to Kohonen's learning rule,

$$\mathbf{W}_k(n+1) = \mathbf{W}_k(n) + \eta(t) \cdot h_q(t) \cdot (\mathbf{X}(n) - \mathbf{W}_k(n)) \quad (3.4)$$

Here, $\mathbf{X}(n)$ represents the n -th sample drawn from N total input space samples, $\mathbf{W}_k(n)$ are the node weights at the n -th iteration, and $\mathbf{W}_k(n+1)$ are the updated weights for the n -th iteration. A time variable t is measured in epochs, each of which represents a complete presentation of N input samples to the network. A learning rate function, defined as $0 < \eta(t) < 1$, dynamically controls the relative rate of weight updates. The neighborhood function $h_q(t)$ centered on the winning node, is defined as $0 < h_q(t) \leq 1$. The network nodes adapt to the local density fluctuations in $p(\mathbf{X})$ through *ordering* and *refinement* phases, during which $h_q(t) \rightarrow 0$ as $t \rightarrow \infty$. Multiple epochs (iterations) are typically required for asymptotic convergence of the algorithm. The basic SOM algorithm is summarized as follows (Figure 3.3), (Kohonen, 1982; Kohonen, 1997; Haykin, 1999):

1. Initialize the synaptic weight vectors $\mathbf{W}(n=1)$ for K nodes.
2. Randomly draw an unseen sample $\mathbf{X}(n)$ from the input space.
3. Determine the winning node q using a similarity metric as in equation (3.3).
4. Update \mathbf{W} for winners using equation (3.4).
5. Return to step 2, and iterate until stopping criteria (checked after each epoch) are satisfied.

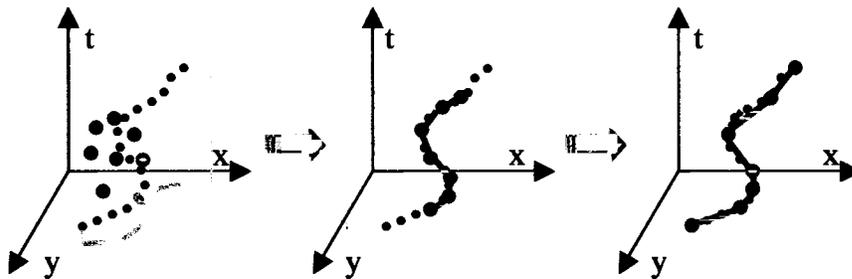


Figure 3.3: Initial, ordering and refinement phases.

The input space in our case is the set of n point coordinates (x, y, t) of the center of mass of the phenomenon as it evolves through time. According to the SOM algorithm a set of neurons-nodes $m < n$ are used to represent the input space. The SOM generalization of a single S-T trajectory is illustrated in figure 3.4, in which a multi-node neural chain is used to abstract the movement fluctuations of a moving object that correspond to MI frames.

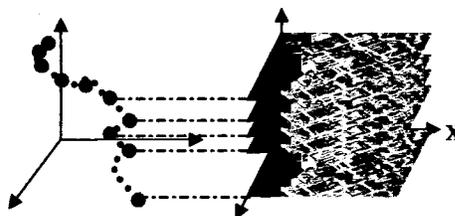


Figure 3.4: Information abstraction in the S-T domain.

Among the advantages of the SOM technique is that the product output space is indicative of intrinsic statistical features contained in the input patterns. In addition, according to the selection of the number of the nodes in the algorithm multiple resolution

is achieved. On the other hand, according to experimental results, the geometry of the input space is not represented sufficiently, especially when complex geometric shapes are generalized. Furthermore, the algorithm is very sensitive to the variables that fluctuate immensely and could lead to poor or no solution. Some of these variables are the learning rate, the extension of the area that the winning node affects and the number of nodes that the generalization will include. This last parameter is the most important one because it defines the volume of generalization that the input space will be represented with and its selection is not automated.

3.2.2 SST-SOM Enhancement

In order to improve the use of SOM for spatio-temporal generalization, SOM nodes should be distributed in a manner that captures the geometric complexity of the trajectory. Complexity is defined in this context as the spatio-temporal variation of the moving object's behavior. In order to detect and quantify this variation the key metric for our analysis is the angle formed between three subsequent points in space (x, y, t). Each pair of points describes a 'state' of spatio-temporal behavior. The angle between two consecutive states is indicative of the local spatio-temporal variation. High deviations of these angles from 180° indicate extreme variations between subsequent states, and require more, densely distributed generalization nodes. In other words, the degree of the generalization is based on the value of the angle in the 3-dimensional space, and a node repositioning and densification process takes place using this angle information. The algorithm proceeds as follows:

1. Perform a fast generalization of a trajectory dataset using a standard SOM. This produces a brief and imprecise map of the input space. The initial number of nodes used in this iteration can be arbitrarily selected.
2. Calculate the angles $\phi(i)$ between SOM nodes to identify locations where densification is needed. This is the case if

$$0^\circ < \phi(i) < \text{max_angle} \quad (3.5)$$

The variable $\text{max_angle} < 180\text{deg}$ can be user defined.

3. Evaluate the number of additional nodes that are to be added to the selected locations. This number is computed according to equation (3.6):

$$n(i) = \frac{\text{max_angle} - \phi(i)}{g} + 2 \quad (3.6)$$

where $n(i)$ is the number of additional nodes in the i location, $\phi(i)$ is the angle between the lines connecting the nodes $(i-1, i, i+1)$, and g is a parameter that determines the generalization volume and can be user defined. (Figure 3.5).

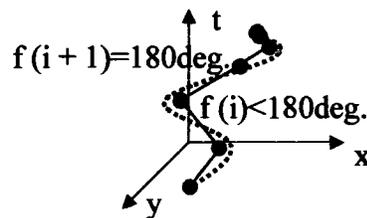


Figure 3.5: Rough generalization and 3D angle estimation.

- 4) Perform new, localized SOM solutions, over short intervals, on the locations that are identified as candidates for densification in steps 2 & 3 above. (Figure 3.6).

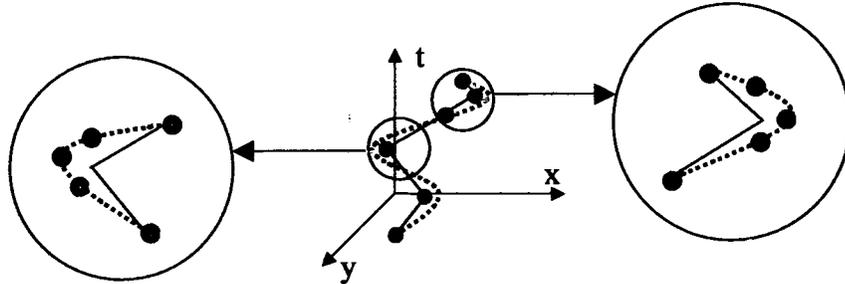


Figure 3.6: Localized SOM solutions.

- 5) Perform thinning to remove nodes that do not contribute to the generalization, e.g. when the angle between states is close to 180 degrees. (Figure 3.7).

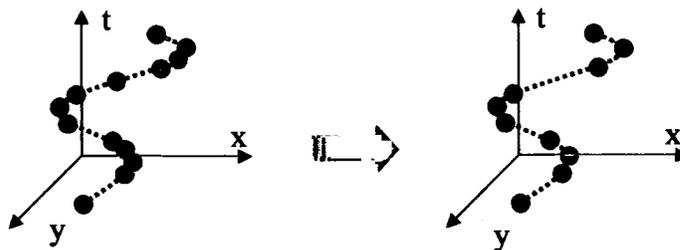


Figure 3.7: Densification and thinning.

The proposed generalization technique bypasses the high precision variable estimation that is needed since a rough descriptor of the shape of the 3d line is only acquired. Then by focusing on the territories that need further generalization SOM behaves in a stable fashion while it tries to represent a smaller and more concise region. In addition, the selection of the number of nodes that are considered able to describe the dataset is *automatically selected*.

3.2.3 Scaled Generalization

Compared to a standard SOM process, the SST-SOM approach offers the advantages of:

- invariance to the selection of the initial number of nodes, and the
- ability to selectively densify or thin to better capture the complexity of content of the processed dataset.

For MI segmentation, storage, and retrieval applications, the most important aspect of the SST-SOM generalization approach is that it accommodates both *dynamic densification* and *spatio-temporal representation*. This densification is based on a dynamic, content-defined scale. In other words, more frames can be used, closer to each other to represent abrupt motions, compared to fewer frames to represent smooth, regular movements. According to the selection of the variables in equation 3.6 more or less nodes can be used to describe the same trajectory based on the application at hand. As demonstrated in figure 3.8 the same trajectory is generalized by using different number of nodes. Apparently the temporal distance between the nodes does not remain constant.

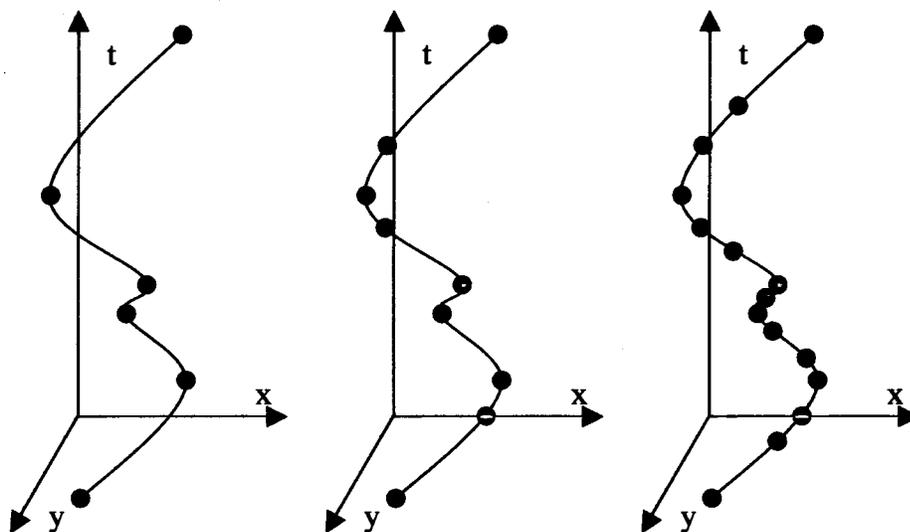


Figure 3.8: Densification of nodes generalizing the same trajectory.

3.2.4 Error Considerations

This analysis proves powerful enough to overcome or restrict the effect of possible errors that occurred in the dataset formation. These errors based on the means of the trajectory acquisition, (GPS or tracking algorithms), include occlusion, noise and misclassification. Since the geometry of the trajectory has a certain continuity, some random errors and noise can be eliminated a priori. In addition two patches or pixels cannot describe the same trajectory in the same temporal instance. Thus, coordinates geometrically irrelevant from the rest of the dataset are discarded. (Figure 3.9).

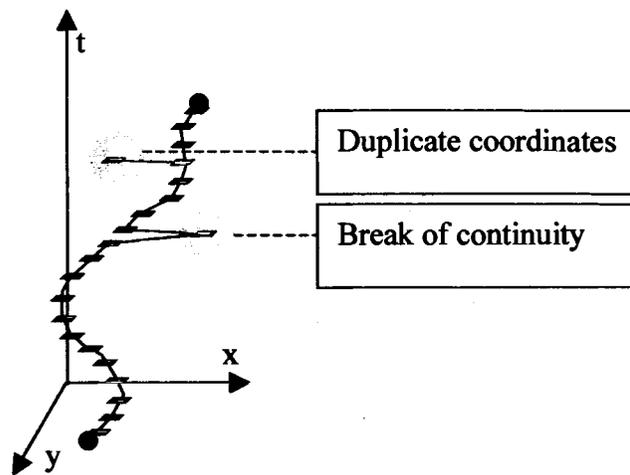


Figure 3.9: A priori eliminated errors.

The remaining of possible errors included in the datasets and are fed into the SST-SOM algorithm, are effectively handled under some limitations. SOM as an iterative and competitive algorithm is a process that generalizes the data and does not explicitly follow each and every one. Noise and misclassification alter the position of the spatial

coordinates in a temporal instance. The final position of each neuron is determined through competition among all neurons for each coordinate pair and the learning rate and winning area determine the neuron movement rate. Therefore, a few errors cannot alter the final result considerably. Occlusion errors result in temporal instances where no information of the trajectory position exists. SOM bridges these information gaps through the connection of neighboring nodes. More details concerning error handling and limitations are demonstrated through the experiments presented in chapter 7.

A possible visualization summary of a MI is a new shorter MI, which includes a base map-image representing the background of the monitored area. The behavior of objects between nodes is represented by rapidly evolving vectors (e.g. moving spots or trace lines). The choice of generalization resolution is a function of the application at hand and specific user needs.

3.3 Summary

In this chapter we presented a technique for the summarization and spatio-temporal scaling of a single trajectory. Using Kohonen's Self Organizing Map neural network we acquired a rough generalization of the spatio-temporal trajectories of moving objects, in the form of a few selected nodes along these trajectories. We introduced the SST-SOM technique, combining SOM with geometric analysis to properly densify these nodes, to better represent the spatio-temporal behavior of objects. Thus, we bypass problems inherently associated with parameter selection in SOM. The presented technique proves a powerful tool for the extraction of generalized information from complex trajectories, displaying high invariance to noise and information gaps in the video stream.

CHAPTER 4

GENERALIZATION AND GROUPING OF MULTIPLE TRAJECTORIES

In chapter 3 we analyzed the generalization of a single trajectory. The consideration of multiple objects brings forward the need to select specific time instances for our MI summaries using nodes from multiple trajectories. One can easily understand that the set of spatio-temporal coordinates of the nodes describing the path of object i and those describing the path of another object j may be totally disjoint. In this chapter, we introduce techniques to accommodate registration of multiple trajectories by using spatio-temporal surfaces that group trajectories according to their spatial and temporal attributes.

4.1 Registration of Multiple Trajectories

According to the density and the dissimilarity of the S-T trajectories and the corresponding nodes, we can follow different strategies for merging a complex scene summary each portraying some disadvantages:

- An obvious solution is to use the nodes from all S-T trajectories and reference all moving objects to every estimated node. This results in a relatively large summary, depending on the number and behavior of the objects.
- Another solution is to define nodes according to the most demanding moving object and project all other node sets to this dominant set. If the behaviors of scene objects are incompatible then the other objects are not efficiently represented.
- Furthermore, by using the SOM a “medium” estimation of node selection upon the whole set of moving objects is obtained. This gives a summary of the whole scene, which does not explicitly depict behavior information for single objects. On the other hand, it provides a technique to unravel mass behavioral attitudes in the scene. For instance if a police car enters the scene, the majority of the moving cars tend to slow down.

To demonstrate the first solution we integrate and merge all the selected nodes from the set of trajectories that comprise a dataset of two moving objects. Co-registration checks in order to handle overcrowded node areas are introduced.

We introduce the concept of temporal neighborhood as the temporal increment under which two or more nodes are considered neighbors. When grouping multiple trajectories, nodes separated by less than this minimum interval dt are merged and substituted by a single “complex” node which is the average of the initial nodes (averaging process).

However, when multiple adjacent nodes are describing longer segments, the summary is approximated locally by a short near-MI segment. Otherwise, it includes distinct sparse instances. Combined, these sparse instances and brief MI segments produce a summary, where temporal resolution changes dynamically, to capture variations of spatio-temporal attributes. On the other hand, it proves inefficient when the number of objects included in the dataset rises.

The co-registration of two trajectories is shown in figure 4.1. The nodes for these two trajectories are denoted by stars and triangles on the left-hand side. The structure of the corresponding summary is also shown in figure 4.1 with elongated segments indicating short videos and small squares and circles indicating sparse instances.

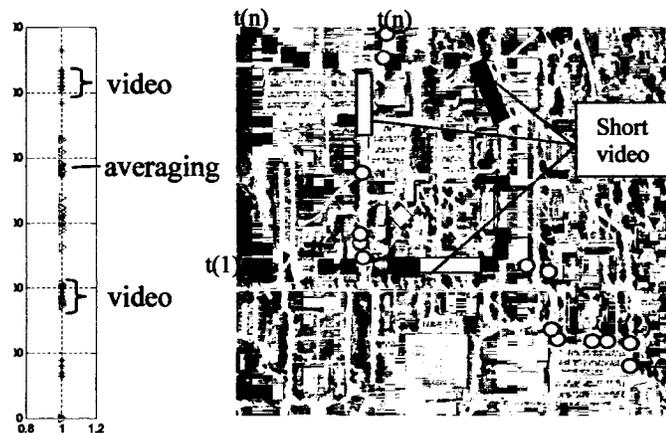


Figure 4.1: Registration and representation of product summary.

This procedure allows us to minimize the number of nodes and the complexity of the produced summary. It is based on the individual nodes of the S-T trajectories and it is

suitable for the summarization of few trajectories. If more trajectories are analyzed the product summary will be relatively large.

4.2 Multiple Spatio-Temporal Trajectory Grouping

In normal monitoring environments, moving objects like cars tend to follow predefined spatial paths namely roads, highways etc. When more than one moving object associates with others, complex relations such as 'group', 'convoy' or 'expected behavior' that are often anticipated, are transposed into the spatio-temporal domain where they can be resolved efficiently. When a set of moving objects retains a series of selected attributes under a series of defined restraints over time then it forms a group. These restraints are both spatial and temporal and they can be represented by a spatio-temporal surface that traverses through time as shown in figure 4.2. All the objects included in the S-T surface over time remain in relative S-T distance and form a group of objects. This S-T surface is termed as S-T neighborhood unit. It is different from the temporal neighborhood concept since it includes spatial and additional attributes that are described below.

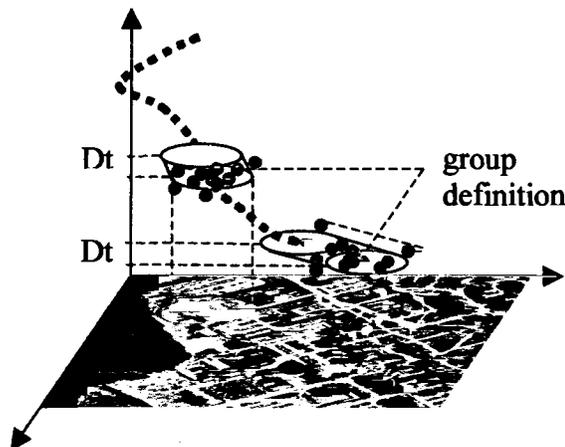


Figure 4.2: Definition of S-T group surface.

Based on the definition of group, generalization of multiple sets of trajectories is performed according to the spatio-temporal relation that the sets of objects retain. The following considerations and attribute descriptions form the basis for the design of the S-T neighborhood unit:

- *Spatial extent*

Two or more objects are considered as a group when they maintain some spatial proximity. This proximity is represented by a circle with a thresholded radius based on the generalization volume desired. Hence, a base 2-D surface in the x-y plane is defined as shown in figure 4.3.

- *Spatial direction*

In cases where the objects are likely to follow a predefined path, like cars follow a road, then the above circle is elongated to the direction of the movement-path to form an ellipse. (Figure 4.3). When an object lies in the defined surface it is a candidate for being part of the current group.

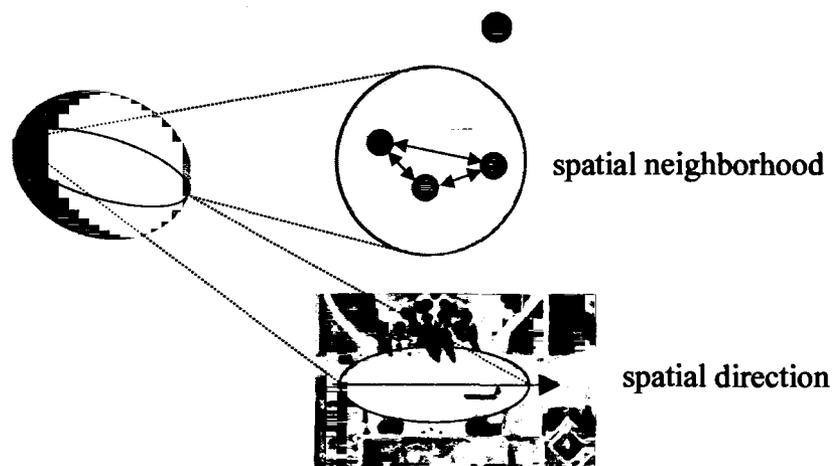


Figure 4.3: Spatial extent and spatial direction measures.

- *Temporal extent*

In the same fashion temporal extent is defined as the temporal duration within which the objects must lie in order to be considered as a group just like the temporal neighborhood. This constraint provides the height of the S-T group surface. (Figure 4.4).

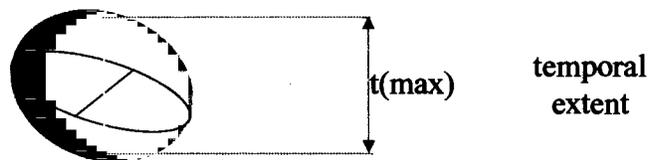


Figure 4.4: Temporal extent.

- *Relation of time/space*

The spatial and temporal constraint need to be linked in a manner that would best describe the application at hand. Therefore, some basic relations are introduced as shown in the figure 4.5. If spatial proximity is the key attribute for the definition of the group then its weight is larger than that of the temporal attribute and the relation of space vs. time can be elongated. Other relations may include elliptic or even perpendicular to the x,y plane surfaces. (Figure 4.5). There is some work for the description of the spatio-temporal surfaces in (Partsinevelos et al., 2000) and in (Hornsby and Egenhofer, 2002).

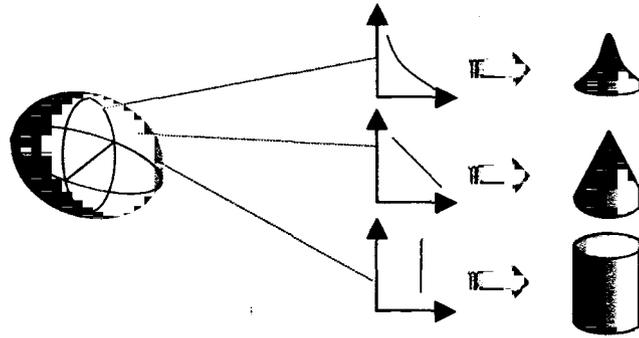


Figure 4.5: Distribution of space vs. time.

- Uncertainty considerations

Until now the geometry of the S-T group surface for a temporal instance is constructed. A dynamic shape reconstruction takes place in order to accommodate uncertainty measures throughout the spatio-temporal domain. E.g. when a crossroad is anticipated, vehicles tend to slow down. As a result the temporal dimension of the surface is enlarged to contain the delays while the spatial dimension is restricted to the applicable area. (Figure 4.6). Accordingly, in a high-speed highway the spatial surface is larger because high speed is anticipated.

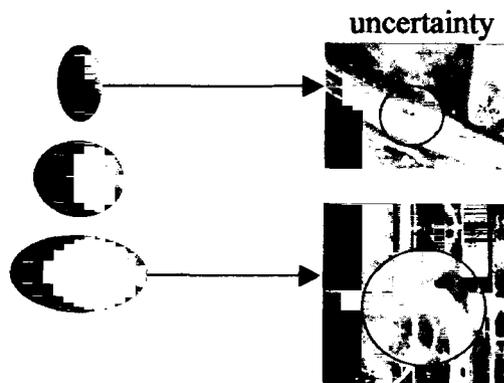


Figure 4.6: Change of S-T surface due to uncertainty.

4.3 Application of Method

The complexity of the definition of the S-T neighborhood unit is apparent. Thus, when there are no specific limitations that define the various thresholds we simplify the process by using simpler parameters. In monitoring applications where the same road network area is monitored it would be rather straightforward to define some initial parameters according to the specific properties of the area. The grouping process will take place according to the defined properties and the user needs. The algorithm constructed to perform the grouping is termed MULTiple Spatio-Temporal Trajectory grouping (MUSTT). A detailed example of the concepts discussed in this chapter is presented in the experiment section. A description of the process according to some design properties is discussed in the next section. We are interested mostly in road networks where vehicles move in rather expected spatial segments such as roads, parking lots driveways etc. The strategy that deals with objects moving in any possible placement in the area is slightly different. E.g. this would be the case of ships moving on the open sea where predefined roads do not usually exist.

4.3.1 Grouping of Distinct Road Segments

When the paths-roads include only single segments and not complex interrelated networks, grouping takes place independently to each segment. Meaning that every moving object traveling in each segment will be bound to this segment and won't traverse to any other. Thus, the segments define the spatial boundaries for the sets of trajectories included in the dataset. As shown in figure 4.7 the three segments have specific enter-exit points and are unrelated to each other.



Figure 4.7: Unrelated road segments.

As input in the MUSTT algorithm we use the classified trajectories of the objects comprising the MI dataset. The algorithm first separates all the roads by simple spatial differentiation. According to the trajectories' spatial extent, each one is assigned to the appropriate road segment.

Next, the basic S-T neighborhood unit is defined by selecting the thresholds and particular properties for each road segment. This unit is imposed to each corresponding segment of the dataset. The algorithm initializes the grouping process in an iterative manner. Every point of each trajectory is examined whether it remains in the same spatio-temporal neighborhood of every other trajectory throughout the road segment. For this

reason the x, y spatial and temporal distances of each pair of points are calculated and compared to the S-T neighborhood unit. After the first iteration the algorithm returns with series of trajectories that can be considered that they belong to the same group.

Nevertheless, each of the trajectories may be included in more than one group. In order to overcome this redundancy we only accept as a successfully classified group the set with the largest number of included trajectories. Thus, the first iteration yields only a single group of trajectories. These exact trajectories are withdrawn from the dataset and the second iteration initiates which results in the second most populated group of trajectories. The iterative process concludes when no more groups can be defined and there are none ore single uncorrelated trajectories remaining on the dataset.

We can now represent each group of trajectories with just one leading average trajectory that describes a general tendency of the group with uncertainty that is defined by the S-T neighborhood unit. The values accompanying the lead trajectory are the medium temporal and spatial starting and ending coordinates with their spatial and temporal range. E.g. the groups of trajectories and numeric descriptions of the trajectories shown in figure 4.8.

The numeric descriptions mean that the first group includes objects 1, 3 and 5 that entered the road at 3:00pm +/-6 minutes and concluded their occupancy of the road segment at 3:45pm +/-3 minutes. The second group includes objects 2 and 4 which entered the scene at 4:00pm +/- 1 minute and exited at 4:30pm +/-1 minute. The last object (#6) entered at 3:00pm but exited at 3:30pm and therefore it couldn't be included in the first or any other group.

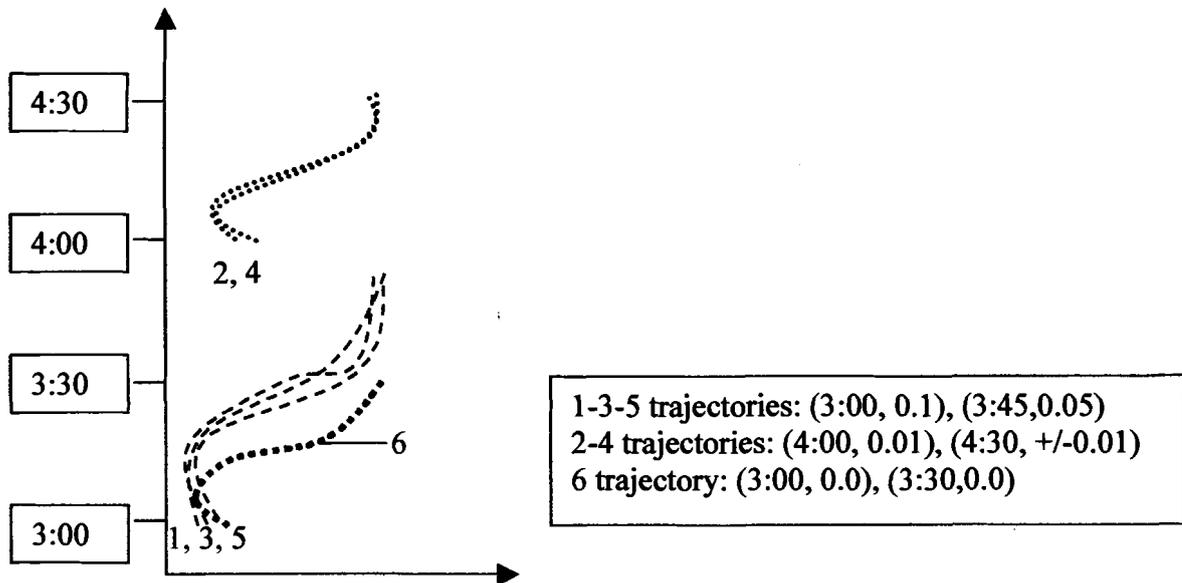


Figure 4.8: Groups of generalized trajectories.

4.3.2 Grouping of Interconnected Road Segments

Common situations include road segments that are connected in a way that objects from one segment can travel to the other, e.g. crossroads, etc. (Figure 4.9).

In these cases we separate the roads in shorter segments so that all resulting segments are unrelated. The points where roads meet become break points that separate the new formed segments. Thus, we now follow the MUSTT algorithm steps considering each of the new segments as a stand-alone unrelated road. The objective of this operation is to form summaries that convey the traffic or moving tendencies more than merely grouping in all the possible moving directions that objects may follow. Experimental results included in chapter 7 portray this type of summarization. The previous section discussed the processing details. As soon as we acquire the formed groups we proceed in the representation scheme as it will be discussed in the next section.



Figure 4.9: Interconnected road segments.

4.4 Scaled Generalization

The generalization procedure focuses only on the nodes of the SST-SOM analysis of each trajectory as discussed in chapter 3. This accelerates the MUSTT algorithm and provides a more concise solution. Computations become condensed and the algorithm converges faster.

By selecting the type of S-T grouping surface and its size, generalization is accomplished, as shown in figure 4.10.

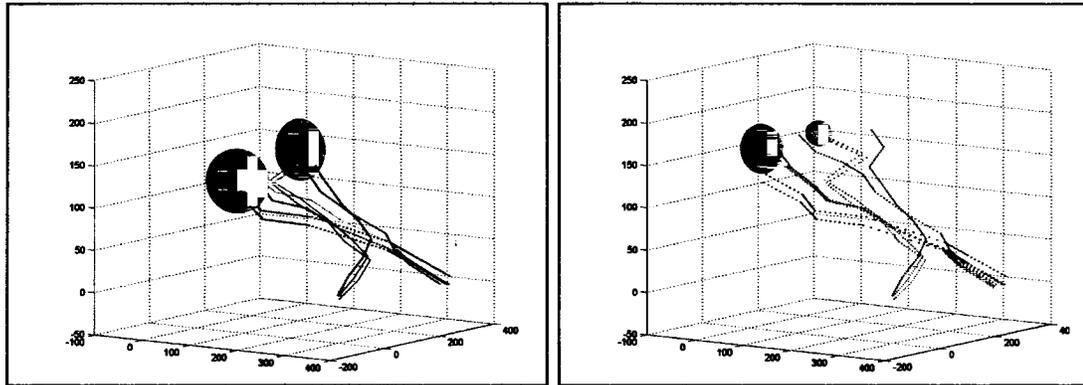


Figure 4.10: Different generalization volumes.

By altering the neighborhood thresholds we get more or less vague representations. Obviously, there is a tradeoff between generalization volume and precision of geometric representation. Using a larger S-T neighborhood unit, more generalization is accomplished, less numbers of leading trajectories are included and a more concise summary can be constructed. Using an infinite temporal neighborhood and defining the spatial extend to include just one road segment, we acquire a single leading trajectory. The final representation scheme of the summary is not discussed in depth in this thesis. A summary representation would include the number of trajectories included in each group, the medium starting and ending times in numeric form, and the S-T unit on the side. According to the number of included objects, a cartographic object will be accordingly large or small and the movement will be shown as a video of the cartographic object moving smoothly in a fast forward fashion between the nodes of interest. This movement in the case of interrelated road segments, yields a more

complicated cartographic object distribution and movement. The following example portrays a possible representation format. (Figure 4.11).

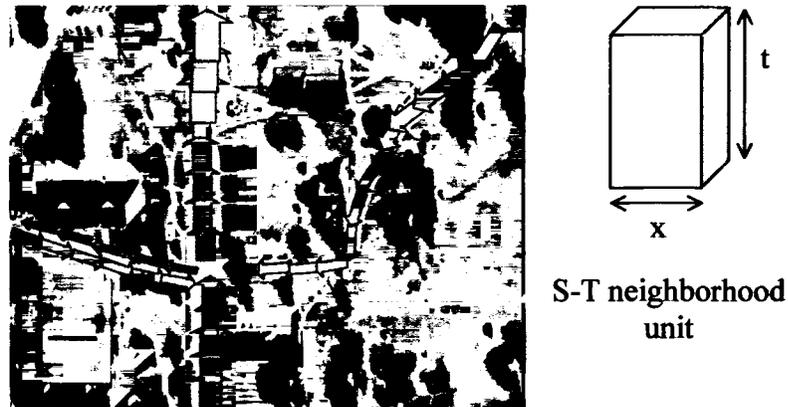


Figure 4.11: Representation scheme.

At a very coarse generalization scale equal to the monitoring time, the number of all objects that used a specific road segment and a medium time of starting-ending of the movement will be shown. According to the system-user needs one can define the group formation according to other techniques. One can group strictly according to each hour of monitoring or irrelevantly to the spatial extend. E.g. to show how many cars entered and exited a city, what were the traffic fluctuations between each hour in a scene etc.

4.5 Summary

In this chapter we tackled the problem of multiple trajectory summarization. Registration of multiple trajectories is accomplished by grouping the points or SOM nodes of the trajectories according to their spatio-temporal relation. This relation is quantified by

introducing the spatio-temporal neighborhood unit. The approach varies slightly between distinct and related road segments while scaled generalization is accomplished by altering the S-T neighborhood unit's dimensions.

CHAPTER 5

TRAJECTORY CLASSIFICATION

This chapter focuses on the classification of moving object trajectories included in a dataset comprised of patch data depicting object position over time. Such datasets are most commonly the product of tracking procedures. Usually, each object is identifiable and it remains associated with its corresponding data patches through a code record in the patch dataset. Problems arise when some objects interrelate in such a manner that we are unable to separate them or there is no information available to ensure their belonging to a class. Thus, data do not explicitly assure an identity since there are no formed trajectories in the dataset but merely pixel coordinates. This exact problem where we have no information about which data belongs to each of the object trajectories is investigated and tackled in the following sections. An attribute-aided technique is introduced, based on attribute space clustering and neural networks classification.

5.1 Introduction

We assume a set of common types of moving objects such as cars and that the number of those objects is relatively small. For a large number of objects computations become

immense and some post processing including clustering on time and space is necessary. Also a relative closeness in the temporal start and end of the object's life is also assumed. If objects do not have any temporal or spatial similarity then their separation is trivial, as explained later in this chapter.

5.2 Trajectory Dataset Description and Acquisition

Let's consider a pair of trajectories. If the trajectories are distinct in space or time then the classification procedure could simply be to separate the points by geometric proximity analysis. On the other hand, if the trajectories are entangled like the ones shown in figure 5.1, classification of each point to its respective trajectory is not a trivial task.

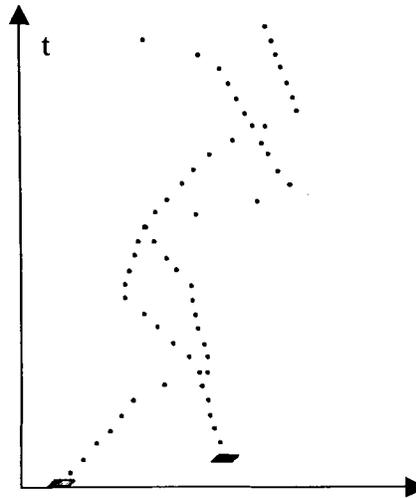


Figure 5.1: Point trajectory dataset.

Situations in which the separation is not obvious, occur when the frame rate is relatively small. As the step between two consecutive images of the same object becomes larger the connection due to proximity weakens. A large frame rate can also be

problematic. Imagine a dataset describing two cars moving side-by-side and then separately. One wouldn't be able to confidently discern between the two cars. Pure geometric analysis that checks the neighboring distance of pairs of points proves inadequate in many cases:

- 1) For the two segments of figure 5.2 one cannot be sure whether the trajectories follow the almost straight path or the curved one.

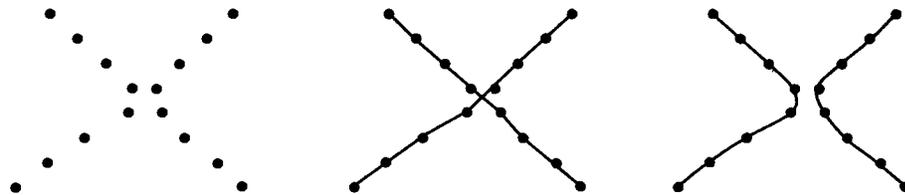


Figure 5.2: Possible misclassification of trajectories.

- 2) As demonstrated in figure 5.3 two points geometrically closer than two others do not guarantee that they belong to the same class. As seen in the figure, even though S_1 distance is larger than S_2 the S_1 link is the valid one and connects the points of the same trajectory.

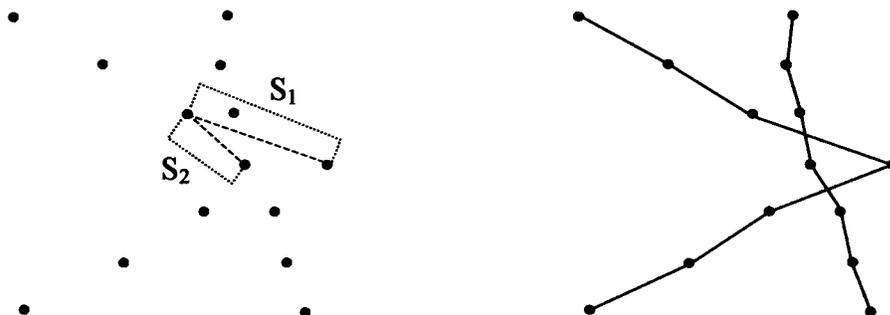


Figure 5.3: Misclassification of trajectories due to proximity.

5.2.1 Dataset Formation

For this classification problem we assume that the number of objects included in the dataset is known. It is relatively simple to infer how many objects are included in the MI dataset. If we randomly select a frame in a temporal instance then by simply counting the number of discrete objects in this frame we can infer the number of objects. Since some noise might be evident, more frames are selected and according to the stated assumptions and noise removal algorithms, we confidently infer the actual number of objects. In the same manner some additional object attributes are estimated while their acquisition precision provides the range of the attribute values as we describe in the remainder of the chapter.

5.2.2 Dataset Attributes

A tracking procedure would commonly yield a number of pixel patches that evolve through time. These patches have some attributes readily attached on them, namely size, shape and color. Sequential geometric neighbors are also evident since MI datasets in most cases presuppose a temporal rate that relates the object's spatial locations through time. All attributes associated with an object are not perfectly determined since occlusion noise and digital image capturing limitations exist.

The input dataset has the form of $(x, y, t, at_1, at_2, \dots)$, where x, y, t are the spatio-temporal coordinates of the trajectory and at_1, at_2, \dots are the attributes attached in the object's description. The key attributes that we use for our analysis are relatively easy to obtain from the dataset or the tracking procedure and include:

- 3 spatio-temporal dimensions

The known spatial (x,y) and temporal (t) dimensions of the center of mass of the object.

- Radiometric dimension

Depending on whether the image is color or black and white this dimension can be 1-d 3-d or 4-d according to the color scheme used. Additional radiometric information is welcome to replace or add to the dimensions included. E.g. thermal imagery can be vital to discern vehicles, humans, etc. At this point we deal with grayscale patches and thus the values of this attribute range between 0 and 255.

- Size and/or shape

The number of pixels describing the object provides an estimation of its size. Shape schemes that classify the shape into predefined categories can be used to portray this attribute. Since we anticipate a small variety of objects that may be part of the scene it is fairly simple to provide a classification map for the shape of the objects. We do not deal with the shape at this point while we use the size as an additional attribute.

As we understand not every single data belonging to the same trajectory will have the same size or the same exact color. Variation is anticipated since errors in tracking and image manipulation are evident. Hence, instead of a stable value, a range of values is assumed to accompany the points of each trajectory.

5.3 Spatio-Temporal Attribute-Aided Classification

The analysis introduced in this section focuses on the separation and differentiation of trajectories, based on both spatio-temporal and attribute coordinates. Having acquired the number of objects and their attributes, the input dataset is 5-dimensional. When one or more attributes are distinct then the separation could be focused on the distinct attribute and the classification would be apparent. When there is some overlap between the attributes, given the spatio-temporal proximity of the trajectories, the same problems as the ones described in section 5.2 still exist. Sequential capturing would fail since there is no evidence whether each point belongs to one or the other trajectory since the attributes are similar.

Spatio-temporal proximity is a key attribute that we take under consideration in order to discern between the different trajectories. When sets of points are geometrically distant then their separation confidence becomes large. In areas where data are geometrically close to each other then utilization of all the attributes takes place to add to the classification confidence. Again, reliance on the differentiation of attributes between consecutive points may lead to severe misclassification of large segments of the trajectories.

The core of the presented procedure is the formation of groups of points that adhere to each trajectory instead of single point-to-point analysis. Thus, a more objective solution is evident since groups of points and their attribute means provide a more reliable basis for the classification procedure. The formed algorithm is called Attribute aided Clustering Classification of Entangled Trajectories (ACCENT).

Experimentation included reduction of dimensionality, neural net transformations, and multi-dimensional clustering. These methods failed to discern between the trajectories since they mostly associate with point-to-point data processing and do not attack the problem in a generalized fashion, which yields a more secure result.

5.3.1 Spatio-Temporal Group Formation

Let's assume that we have a dataset that includes two trajectories as shown in figure (5.4a). The analysis begins by separating groups of points that are spatio-temporally distinct from any other group. This is accomplished by imposing a distance threshold to each pair of points. Points form groups as long as they are farther than the threshold distance from corresponding points of the same temporal instance.

A temporal proximity is also required to describe the connection between the processed points. The groups of points are considered the base units over which the first step of classification takes place and they are termed 'branches', as shown in figure 5.4b. The red rectangles embrace distant point pairs, while the blue rectangles include the formed branches. There is strong confidence that the points included in each branch, belong to the same trajectory even though we don't yet know to which one. In this first step, there is no need for any additional attribute processing.

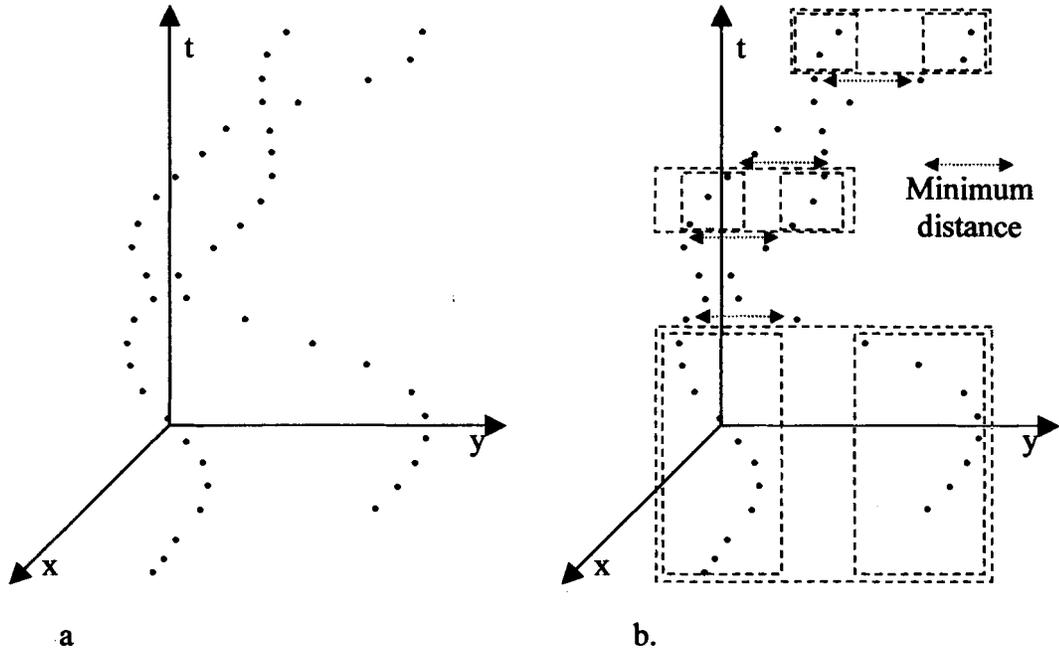


Figure 5.4: a) Input dataset and b) branch formation.

Now a set of branches including the spatio-temporal, color and size coordinates is available. Next, we estimate the mean of the color and size attributes and we map them in a 2-dimensional space. (Figure 5.5).

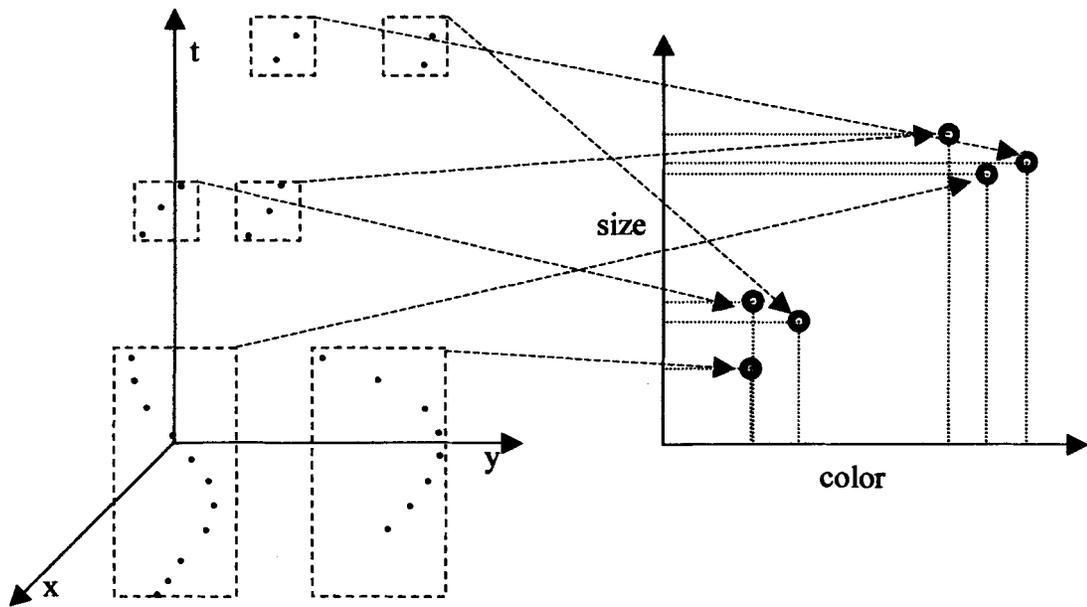


Figure 5.5: Average attribute mapping of branches.

As demonstrated in the figure, the difference among attributes for each trajectory is mapped in the attribute space in a separable manner. This approach works when the varied attributes for each trajectory have different means. In other words, when there is none or some controlled overlap in the attribute values then the above-defined separation is evident. This overlap cannot be defined uniquely and depends on the placement and range of all data coordinates. In the experimental section in chapter 7 we show that even with severe overlap in the attributes we still can differentiate the branch identity.

The branch mean attribute values carry with them single points that may individually be misclassified. To exhibit the advantage of this group branch processing lets assume that a point (*i*) that belongs to trajectory *A* has the following attributes:

$$(x, y, t, c, s) = (10, 10, 10, 124, 24)$$

where (*c*) is the color value and (*s*) is the size value. The average values of the varied attributes (*c, s*) of trajectory *A* are (115 +/-10, 20+/-7), while for trajectory *B* they are (125+/-10, 25+/-7). As easily seen under some spatio-temporal proximity between trajectories *A* and *B* the coordinates of point *i* would classify it under trajectory *B*, even though it belongs to trajectory *A*. On the other hand if point *i* belongs to a branch populated with many points, the average value of the attributes would logically fall near the *A* attribute center. (Figure 5.6).

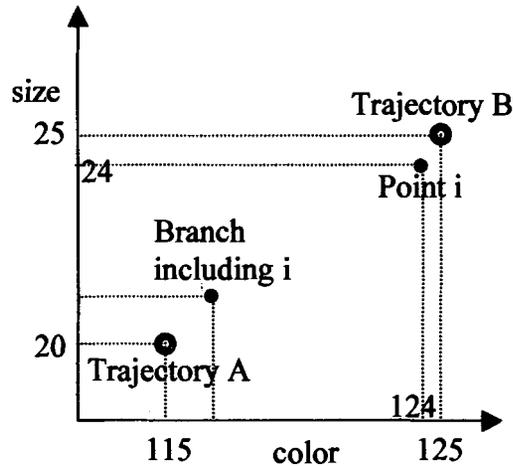


Figure 5.6: Correct classification through branch analysis.

We have to make sure that the number of points included in each branch not to be minimal so that a populated statistical average would be able to classify correctly possible outliers.

5.3.2 Attribute Space Clustering

The following task of the procedure is to divide the separable attribute data according to their positioning in the 2-dimensional attribute space. This task is accomplished by utilizing a simple SOM or k-means algorithm that takes into account not only the separability of the data but also the neighboring of each point to the others. The SOM algorithm is discussed in chapter 3. The result is shown in figure 5.7. Two nodes are used and after the iterative process they converge to the center of the two formed clusters each representing three attribute pairs. The branches are now classified and related to the trajectory they belong to.

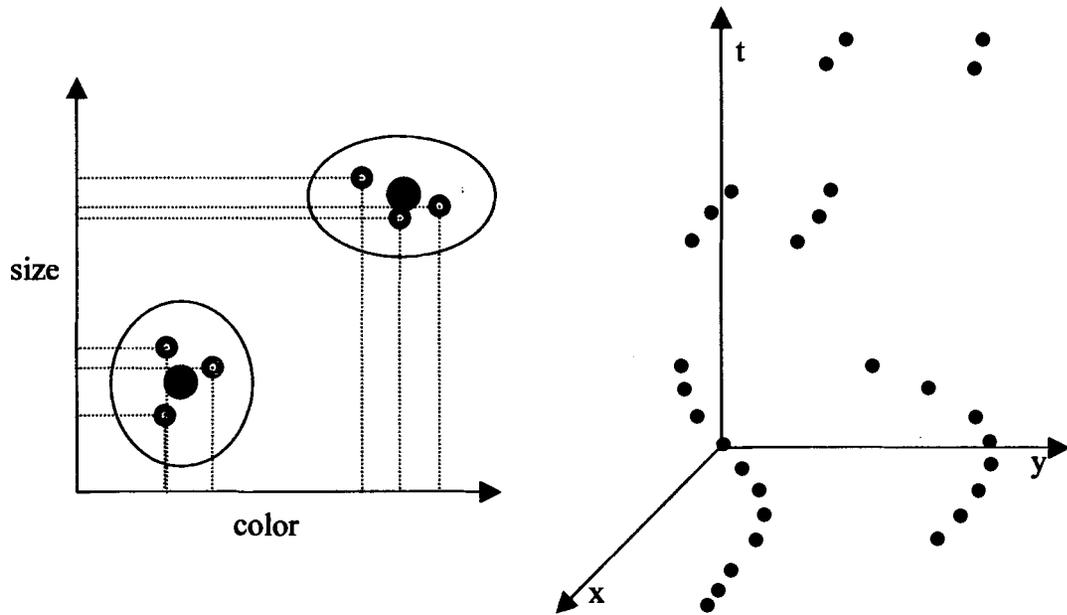


Figure 5.7: Attribute clustering and branch classification.

The remaining points, namely the ones near the intersections between the trajectories are still unclassified. Their classification is accomplished by utilizing a backpropagation neural network.

5.3.3 Backpropagation Theory

The backpropagation neural network (NN) (Haykin, 1999), is a widely used neural network that is applied to numerous diverse applications. Its basic concept is that given an input space it becomes connected with the output space through a series of synaptic neurons that form a series of hidden layers.

A node or neuron is the basic computational unit of a NN. Its function is to receive an input signal and to provide an output through a function that resides in the node. This function is termed as activation function and can be linear, or non-linear such as sigmoidal including hyperbolic tangent and logistic. The input and output values are associated with weights (w) and biases (b) that are altered by the function. If more weights form the input of a neuron, then the function computes the weighted sum of the inputs. In figure 5.8 the input (x_1, x_2, x_3) is fed to the node and the function f yields an output y according to the equation:

$$y = f\left(\sum_{k=1}^j w_k \cdot x_k\right), \quad (5.1)$$

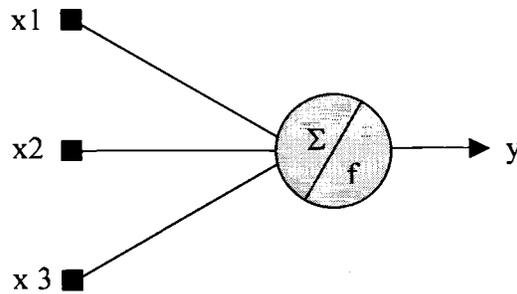


Figure 5.8: A neuron-node produces the output y .

The ordering of the neurons forms layers that are termed hidden layers. In figure 5.9 we discern 2 hidden layers each including k, l , nodes. The input space is n -dimensional while the output includes m nodes.

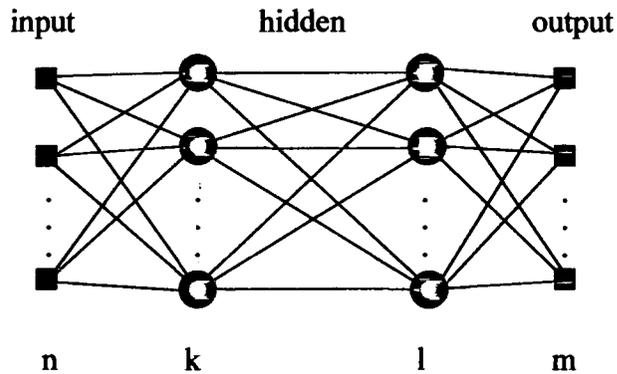


Figure 5.9: Two hidden layer feedforward network.

A feedforward network is a network that does not include loops or circles in its design. The hidden nodes must be under some sort of ordering according to the input and output space. This prerequisite must exist because we actually do not have any target values inside the hidden layers and therefore the network does not know where to converge. The error backpropagation travels reversely in the layers and gives an estimate of performance while inputs and weights under the functions of the nodes provide in-between target values for the hidden layers.

In order to achieve convergence through the training phase a linear or non-linear function connects the various layers of nodes namely input, hidden and output. The input vector travels through the layers and assigns weights to all the connections. These weights according to the function yield results back and forth among the nodes and therefore adjustments are made so that the input will result in the output.

The basic backpropagation algorithm is summarized as follows:

1. Initialization: Give initial synaptic weight and bias values to the connections of the network assuming a uniform distribution.
2. Forward pass: Computation of the function signals through all layers towards the output.
3. Backward pass: Compute the error and correct the synaptic weights of the neuron connections.
4. Iteration: Return to step 2 and 3, and iterate until stopping criteria (checked after each epoch) are satisfied.

During the forward pass the input vector activates the nodes and propagates its effect throughout the hidden layers towards the output space. At this point all connections have stable assigned weights. When compared to the output space an error is formulated which now travels backwards. The weights at this backward pass get changed according to the error and an error rule. By these means the network adjusts its values in a statistical manner and tries to adjust the input towards the output by diminishing the back-propagated error. The forward signals are termed function signals while the backward are termed error signals.

5.3.4 Backpropagation Towards Trajectory Classification

In our case the goal is to classify a multidimensional dataset into separable classes. The input space is five-dimensional as previously described. Figure (5.10). There is a need to train the network in order to learn the specific classification task. Therefore, a set of correct classified input-output relations is required. This training phase adjusts the values

of weights and biases in the network and thus it is important to include as many correct training data as possible to accomplish the best possible performance.

Our input space is formed by the already classified branch points of the trajectories, as described in the previous section. In many cases and according to the application at hand these already classified points comprise a large percentage of the original dataset and the performed classification is considered quite effective. The output space in the training phase consists of vectors of the form given in equation (5.2) for a three trajectory case:

$$\text{training_output} = [1 \ 0 \ 0] \vee [0 \ 1 \ 0] \vee [0 \ 0 \ 1], \quad (5.2)$$

where each vector corresponds to a different class. In other words the network is forced to learn and map the input space in a dimension space of size equal to the number of trajectories which in our case is small. After the training phase we feed input vectors into the network and the network produces a three by one (3×1) vector for each input according to the input coordinates and attributes of each input vector. Then according to the resemblance or value proximity of the output to one of the training vectors of equation (5.2) the network decides in which of the classes each input most likely belongs.

Experiments show that a two hidden layer network comprised of five and three nodes accordingly is capable of performing an adequate classification as shown in figure 5.10. A 5-dimensional input space results in a three possible output layer for a dataset comprised of three trajectories.

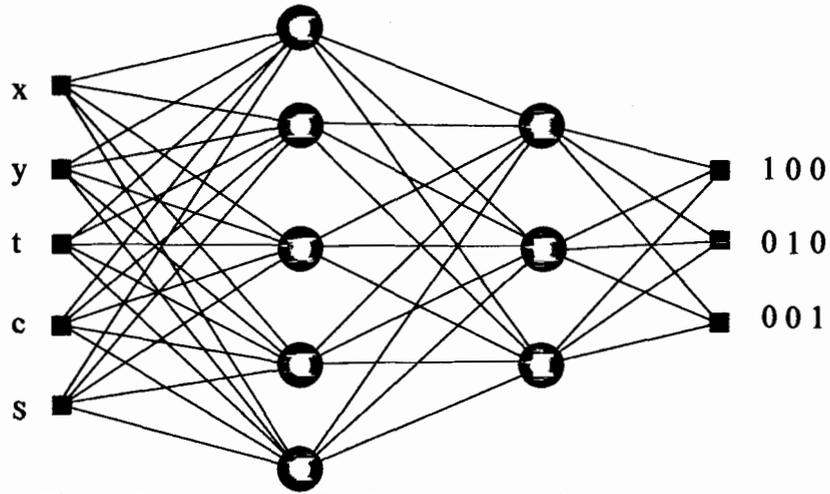


Figure 5.10: Two hidden layer network for classification.

Every input vector can be classified under one of the classes. Nevertheless, not all of the inputs are classified under equal confidence. The output values provide a measure of classification reliability. Therefore, by setting thresholds in the values of the output vectors, the sensitivity of classification is adjusted to include more or less points in each class. More included points results in larger possibility to include misclassified ones. E.g. the thresholds could be for each class of equation (5.2):

Class1: [value1>0.6 value2<0.3 value3<0.3],

Class2: [value1<0.3 value2>0.7 value3<0.3],

Class3: [value1<0.1 value2<0.2 value3>0.9],

Class (1) will probably include more points but the possibility of misclassification rises, while class (3) would have minimal misclassifications yet few included points. Thus, some points will still remain unclassified. The trajectory classification outcome after this step for the trajectories of figure 5.7 is shown in figure 5.11.

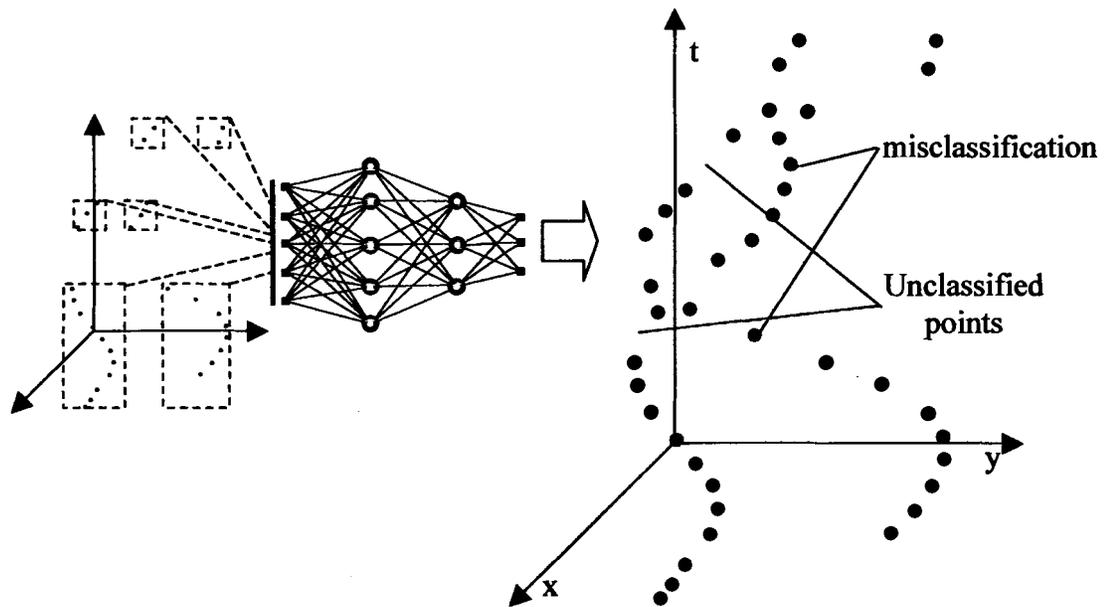


Figure 5.11: Backpropagation classification outcome.

5.3.5 Trajectory Reconstruction

The final classification step includes the separation of each classified point population into a different dataset, which most likely would be incomplete and would include some outliers or misclassified points. Most of the outliers can be easily removed according to simple proximity tests. In figure 5.12 the highlighted points portray the distant misclassified ones, which are eliminated from further processing.

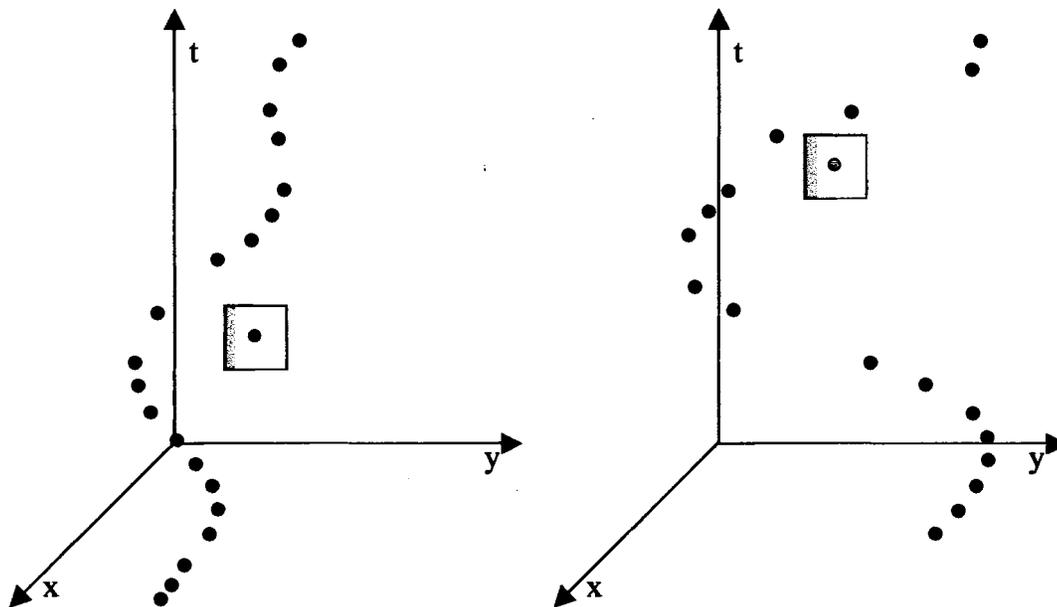


Figure 5.12: Outlier elimination.

Next for each formed dataset imposition of the SOM algorithm takes place. Again, the analysis of the SOM properties and solutions was presented in chapter 3. A better representation of each trajectory is constructed and all the gaps are linked through the SOM algorithm. The SOM node chain is set upon the remaining unclassified points. If a few points portraying an abrupt change in orientation or speed are missing then the neighboring SOM nodes form connecting lines onto which additional unclassified points should lie. These lines are termed 'carriers' and operate as links of points. Since the points between a pair of nodes do not lie on the straight line of the carrier, we need to expand the inclusion space. It is like forming a cylinder around the carrier and testing whether the points lie inside this cylinder or not. For this reason, we estimate the 3-

dimensional distances between the unclassified points and the carriers. In figure 5.13 at the left side, a SOM chain of 7 nodes initializes the procedure, while on the right side the resulting classification due to carrier inclusion is shown to reconstruct the partial trajectories.

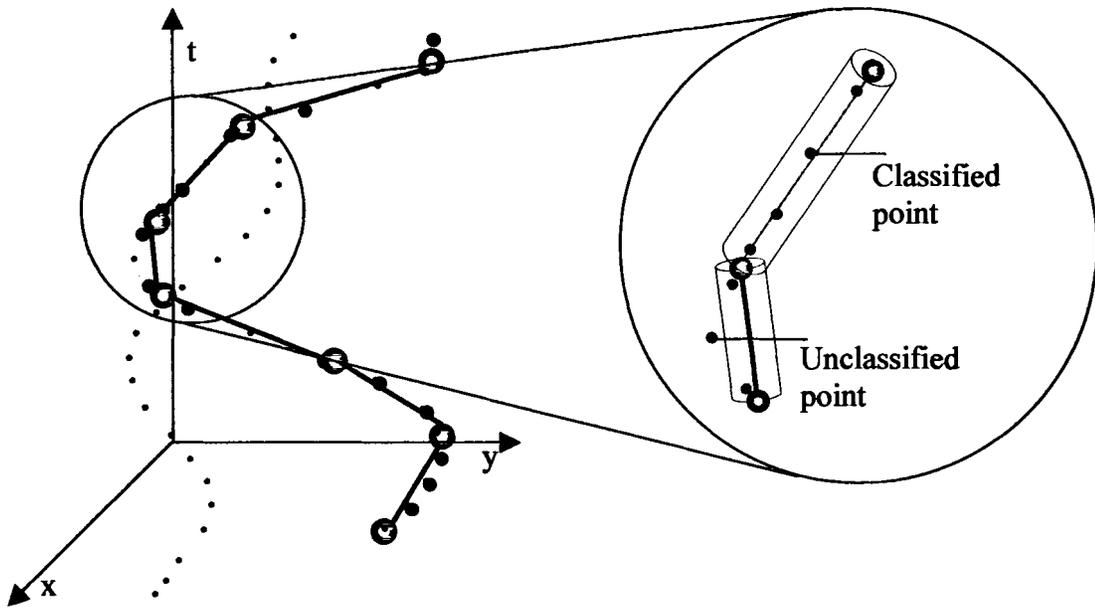


Figure 5.13: Carrier formation and further classification.

The carriers like the branches are more independent and objective selectors of point data since they are not related merely to pairs of points, but they depend on a set of many more points. Since there is no angular information at this point between the nodes and points, we cannot utilize the SST-SOM neural net.

5.4 Algorithm Summary

The ACCENT algorithm steps are summarized as follows:

1. Separation of branches according to spatio-temporal distance.

2. Mapping of branches' average attributes.
3. Clustering of attributes and classification using k-means.
4. Backpropagation classification of remaining points.
5. Separation of trajectories, misclassification corrections and SOM solution.
6. Carrier formation and final point classification.

The experimental chapter 7 demonstrates several examples of the analysis discussed and provides examples of classification errors and misclassification percentages for various situations.

5.5 Summary

In this chapter we introduced an approach for the classification of multiple trajectories, which have uncertain identity as to which objects they belong to. The technique uses additional attributes inherent in the trajectories description to classify some parts of them where the confidence of classification is high. Backpropagation classification and geometric/SOM analysis further classifies the rest of the data with very reliable results.

CHAPTER 6

SUMMARIES OF 2-DIMENSIONAL PHENOMENA

In chapters 3, 4 and 5 we introduced a framework for the summarization of spatio-temporal trajectories considering point datasets (or patches related to a point), where moving objects are reduced to a point representation, ignoring their spatial extent and the variations of their outlines. In this chapter we move beyond this simplification, extending the framework introduced in the previous chapters to accommodate the spatial extent of objects. This allows us to consider not only the movement but also the deformation of spatial objects, introducing a new more comprehensive spatio-temporal model. This is a key development to support the analysis of spatio-temporal phenomena that have certain spatial extent and change their position and/or extent over time.

6.1 Introduction

Phenomena that are tackled in this chapter include slowly moving (e.g. urbanization trends depicted in a series of monthly satellite images) or rapidly evolving (e.g. hurricanes depicted in hourly or daily datasets), and they take place over a fixed area (e.g. flooding) or may be constantly changing their location (e.g. a moving fire front).

The movement of a region within the S-T space is manifested as a set of neighboring classified points moving over time. The trajectory of an object defines a pathway within the 3-D S-T space, by connecting all positions depicting the same region over time. (Figure 6.1).

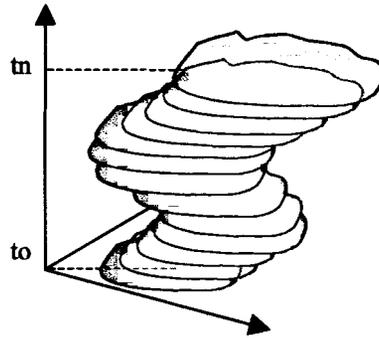


Figure 6.1: Movement of a phenomenon in the spatio-temporal domain.

By exploitation of the geometric properties of S-T trajectories of moving objects we extract important information for dynamic information generalization and creation of MI dataset summaries. We address two primary types of change inherent in the movement of a phenomenon. First, the object changes its location according to an external reference frame. This kind of change can be represented by a trajectory describing the movement of the center of mass of the object as it evolves in time. The second type of movement refers to an internal reference frame and it describes the changes of the shape of the object that occur through time.

An efficient summarization framework has to support the capability to handle multiple levels of generalization of MI datasets. This thesis introduces a generalization approach that accommodates multi-scale spatio-temporal representation. As discussed in

chapter 3, the multi-scale analysis is based on dynamic scale selection. Considering the above, we again treat summaries both as products for visualization purposes and as intermediate datasets that are suitable for further qualitative and quantitative similarity analysis.

6.2 Spatio-Temporal Helix

We assume phenomena that change gradually between two consecutive MI frames and they spread relatively evenly in the (x,y) plane. At this point, no holes in the area depicting the phenomena and no possible splits of the area are examined. We consider a generalized form of representation that includes two types of change, external and internal:

1. When a phenomenon changes geographic location in its entirety and has no evident stable base around which it spreads then the movement of the center of mass of its area is described by a spatio-temporal trajectory. This trajectory forms a component of change that outlines the average moving tendency of the region through time.
2. Inner changes of the area describing the phenomenon are also significant and should be represented. Therefore, a change occurring in the area of the phenomenon between frames corresponding to $t(i)$ and $t(i+1)$ instances, relates to either expansion or shrinkage. This change relates to either specific spatial segments or to the entirety of the phenomenon.

The representation we introduce is indicative of just the general shape deformation tendency of the area and it does not compensate for the actual shape of the phenomenon. Shape descriptions and indexes are seen in (Liu and Geiger, 1999) and are not tackled in this paper. In addition, the focus remains on geometric change of moving objects as opposed to *semantic*.

We make use of the concept of the spatio-temporal helix (*STH*) as a compact description of an object's spatio-temporal behavior (Stefanidis et al., 2002a; 2002b). It comprises a central spine and annotated prongs. We also include the initial instance of the phenomenon description. More specifically:

- The central spine models the spatio-temporal trajectory described by the center of the object as it moves over a temporal interval.
- The protruding prongs express expansion or collapse of the object's outline at a specific time instance.

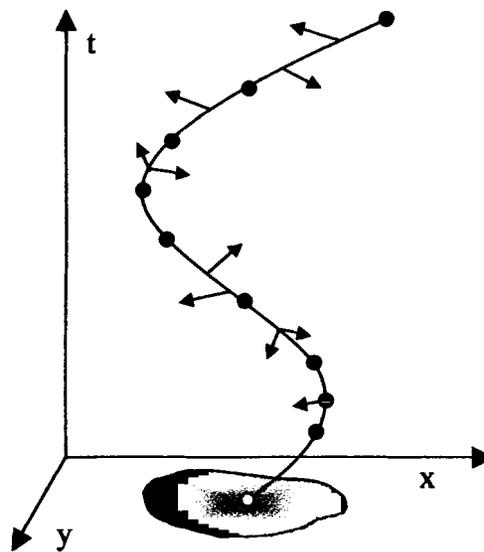


Figure 6.2: Spatio-temporal helix.

A phenomenon can be described by a set of nodes corresponding to instances of change according to the defined components of the S-T helix. Equation (6.1).

$$Ph = (ns^{(1)}, ns^{(2)}, \dots, ns^{(n)}, np^{(1)}, np^{(2)}, \dots, np^{(m)}), \quad (6.1)$$

where $ns^{(1\dots n)}$ are the nodes describing the spine of the phenomenon and $np^{(1\dots m)}$ are the nodes describing its prongs.

As a spatio-temporal trajectory, a *spine* comprises of a sequence of (x, y, t) coordinates. Its nodes correspond to breakpoints along this trajectory, namely points where the object altered its speed and/or orientation. (Figure 6.3). Accordingly, each node $ns^{(i)}$ is modeled as:

$$ns^{(i)} = (x, y, t, q_s, q_o)^{(i)}, \quad (6.2)$$

where:

- (x, y, t) are the spatio-temporal coordinates of the node,
- q_s is a speed qualifier classifying the node as an *acceleration* (q^a), *deceleration* (q^d) *one*, and
- q_o is a rotation qualifier that can be expressed as the *azimuth of the movement*.

Each prong is a model of the local expansion or collapse of the outline at the specific time instance where this event is detected, and is a horizontal arrow pointing away from or towards the spine. (Figure 6.2). It is modeled as:

$$np^{(i)} = (t, r, a_1, a_2)^{(i)}, \quad (6.3)$$

where:

- t is the corresponding temporal instance (intersection of the prong and the spine),
- r is the magnitude of the outline modification, expressed as a percentage of the change between areas or medium distance between consecutive object outlines, with positive numbers expressing expansion (corresponding arrows pointing *away from* the spine) and negative numbers indicating collapse (arrows pointing *towards* the spine),
- a_1, a_2 is the range of azimuth or cardinality where this modification occurs; with each azimuth measured as a left-handle angle from the North (y) axis. (Figure 6.3)

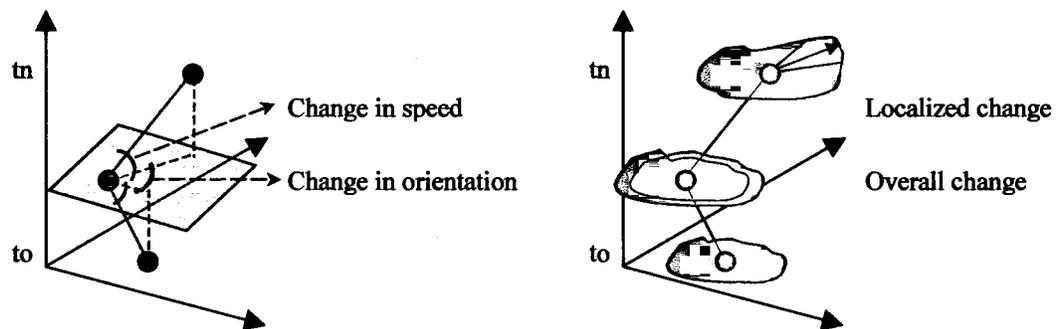


Figure 6.3: Schematic definition of spine and prong attributes.

We can have more than one prong at the same instance, as it is possible for an object to be expanding in one direction while shrinking in another at the same time. While in general prongs correspond to small ranges over an outline, by properly assigning values to the azimuth parameters of a prong we can also model global expansion/collapse ($a=360$).

Combined, spine and prongs comprise a concise signature of an object's spatio-temporal behavior. They express external (spine) and internal (prongs) processes and allow efficient spatio-temporal modeling and support complex analysis. In order to go further in the generalization and representation process we address each component with different analyses that yield dynamic scaled generalization in both (x,y) and t coordinates.

6.3 Movement Capturing Towards S-T Helix Construction

In this section we introduce the analysis, which leads to capturing the spine and prong properties.

First, we estimate the areas of the phenomenon in each instance by using a simple outline edge detection algorithm. Then, we compute the center of mass for the areas in each frame. The center of mass in a 2-dimensional surface is given by adding the x and y coordinates and dividing them with the area of the phenomenon. In a classified image we simply compute the regions where the phenomenon resides and count the included pixels, equation (6.4).

$$x_{center} = \frac{\sum_1^n x_i}{A}, \quad y_{center} = \frac{\sum_1^n y_i}{A}, \quad (6.4)$$

where $x_{(center)}$ and $y_{(center)}$ are the coordinates of the center of mass, $x_{(i)}$, $y_{(i)}$ are the coordinates of each pixel included in the region, and A is the area of the region describing the phenomenon. After detaching the center of mass the spatio-temporal trajectory is formed. (Figure 6.4).

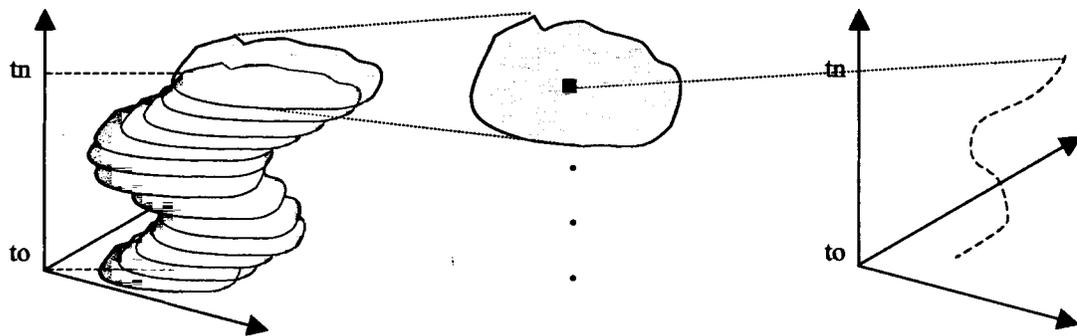


Figure 6.4: Spatio-temporal trajectory of phenomenon.

6.3.1 SOM Towards Spine Movement Capture

Based on the theory of SOM and the SST-SOM algorithms discussed in chapter 3, the SOM solution provides a generalization of the movement of the region. SOM generalization of a single S-T trajectory is illustrated in figure 6.5, in which a multi-node neural chain is used to abstract the movement fluctuations of an object corresponding to the center of the region we model.

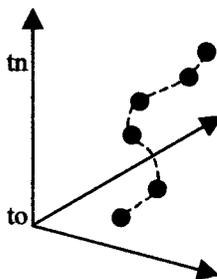


Figure 6.5: SOM nodes describing a S-T trajectory.

Next, translation of all frames according to a common center of mass takes place. The translation is performed by adding to all the pixels of frame j the correction coordinates defined by the equations (6.5):

$$\Delta(X, Y)_{j,k} = (x, y)_k - (x, y)_j, \quad (6.5)$$

where $(x, y)_k$ is the point where all j regions are referenced to.

The algorithm is now ready to represent the perimeters of the regions in each temporal instance as will be described in the next section.

6.3.2 Directional Change Analysis Towards Inner Movement Capture

In accordance with the previous step, every solution node for frame i is translated to the next frame $(i+1)$ by Δx , Δy . (Figure 6.6). Therefore, it is considered as a non-moving object and the solution proceeds by comparing pairs of subsequent areas.

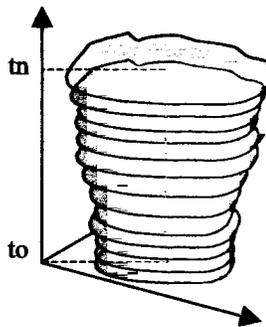


Figure 6.6: Translation of all frames to form a non-moving object.

In order to capture the relation between two instances of an objects' movement we use one of the known cardinality models (Egenhofer & Franzosa, 1991). Here the known cardinal directions (NSEW) and their subspaces are used to assist capture of change. A series of sub-areas are formed by imposing the cardinality reference frame on the center of mass of each area. Comparison of the formed sub-areas for each pair of frames-areas takes place to provide a numeric or percentage change tendency. (Figure 6.7).

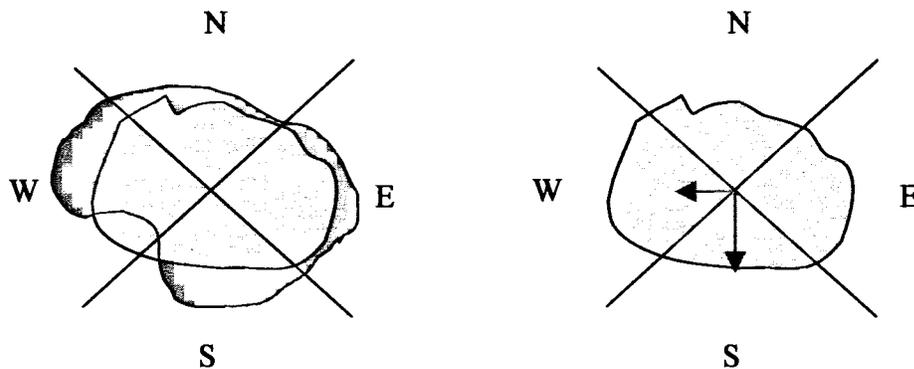


Figure 6.7: Comparison of sub-areas between consecutive frames.

A set of thresholds defines which percentage or volume of change should be included in the summary. As demonstrated in figure 6.7 changes in North and East are considered too small to be represented by the prongs. In some other application the threshold could be set so that the North and East changes would be included in the summary. Accordingly densification of the cardinality framework yields a more detailed representation of change as seen in figure 6.8.

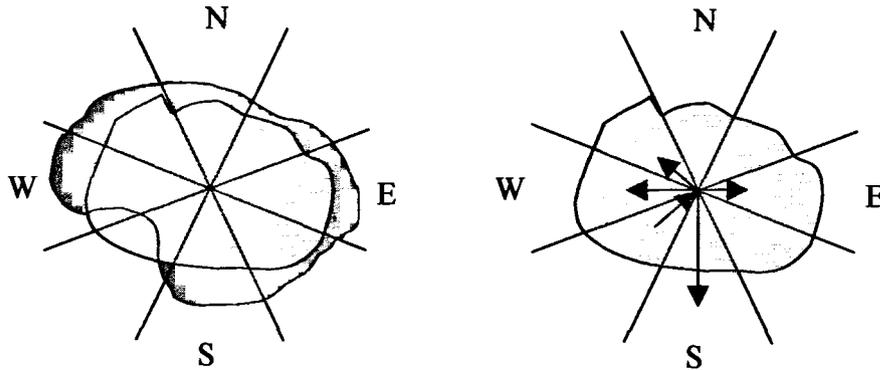


Figure 6.8: Densification of cardinality reference system.

Prongs are defined by comparing areas between two corresponding sub-areas or by measuring the medium distance between two consecutive sub-outlines as formed by the cardinality reference system.

Furthermore, we define several types of additional outline descriptor nodes. These nodes depict attributes that may be of interest to the phenomenon mapping. Directional change captures the localized abrupt changes of the phenomenon's spatial occupancy through time. The non-abrupt non-localized changes of the area of the phenomenon are captured through simple geometric computations that set thresholds on the values of the area modifications through time. These types of change include:

1. When there are gradual modifications that sum up to a significant change between $t_{(i)}$ to $t_{(i+n)}$. (Figure 6.9).

$$\sum_i^n \left(\frac{|A(i+1) - A(i)|}{|A(i)|} \cdot 100 \right) \geq P\%, \quad (6.6)$$

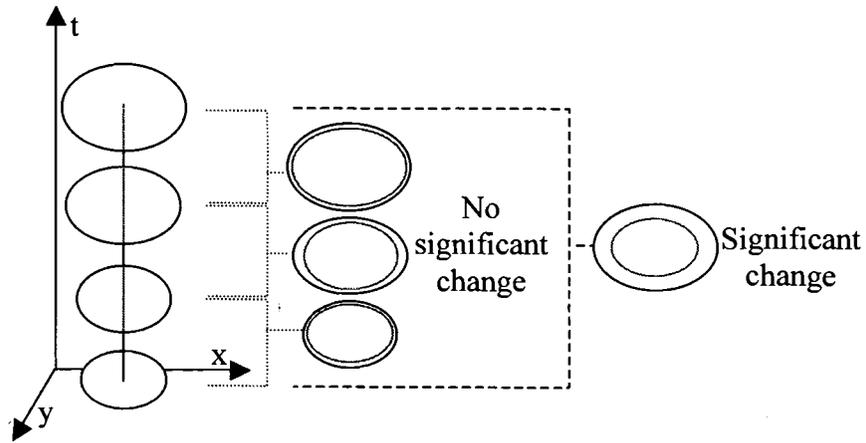


Figure 6.9: Gradual changes towards significant change.

2. When there is a transition between expansion and shrinkage of the area. In this case in order to avoid transitions of insignificant volume we set a $P\%$ threshold to the changes that occurred since the previous transition.

$$A(i+1) \cdot A(i) < 0 \wedge \sum_i^n \left(\frac{|A(i+1) - A(i)|}{|A(i)|} \times 100 \right) \leq P\%, \quad (6.7)$$

6.3.3 S-T Helix Framework Summary

The above defined nodes complemented with the prong vectors form a concise data representation that resembles DNA chain sequences. The phases of the analysis are summarized as follows. (Figure 6.10).

1. Compute center of mass.
2. Perform SOM for the center of mass S-T trajectory.

3. Translate and register the areas to a common (x,y) reference center.
4. Perform directional change analysis for each frame.
5. Compute the additional geometry nodes.

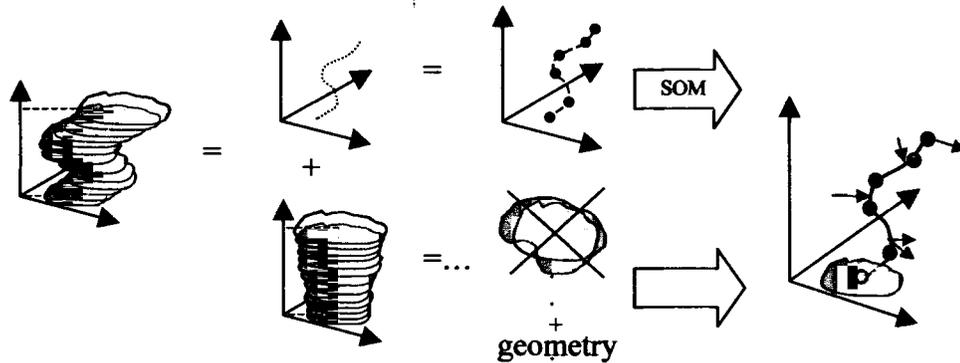


Figure 6.10: Overview of helix modeling.

6.3.4 Static Phenomena

Up to now, we considered phenomena that change with respect to a base location. For phenomena such as floods in lakes, urbanization in an existing town, tumor growth etc. we do not consider external movement of the center of mass. Thus, we are not concerned with the first type of change discussed in the previous sections. Consequently, we need to compute the area of each frame and then estimate the directional change and norm of the abrupt changes between each pair of consecutive frames complemented with the geometry considerations. It is noted that in this case the center of mass may not coincide with the center axis where the cardinality reference system is settled. The axis in this case is perpendicular to the center of mass of the phenomenon area as depicted on the first frame.

6.4 The S-T Helix as a Generalization Tool

The S-T helix supports scaled generalization of the resulting data by varying the attributes of the algorithms for each of the components discussed. Specifically, the SOM algorithm supports flexible representation precision by changing the initial number of nodes that are used to generalize the 3-d trajectories. More nodes are used in order to acquire a more precise representation of the movement of the phenomenon while fewer nodes are used when a more concise representation is required (figure 6.11, blue nodes). If the number of the nodes equals the number of frames-instances then we have a perfect match in the representation of the phenomenon movement.

The prongs capture the local abnormalities according to the densification of the cardinality framework. In addition, the threshold of the change significance we set between the formed areas determines the generalization volume of the final product (arrow prongs).

Finally, based on the thresholds for which overall change constitutes significance we obtain variable generalization of the phenomenon (figure 6.11, yellow nodes). For a more accurate and detailed description a small threshold percentage would be adequate. By using a minimal threshold percentage we get a near complete description of the shape changes of the phenomenon.

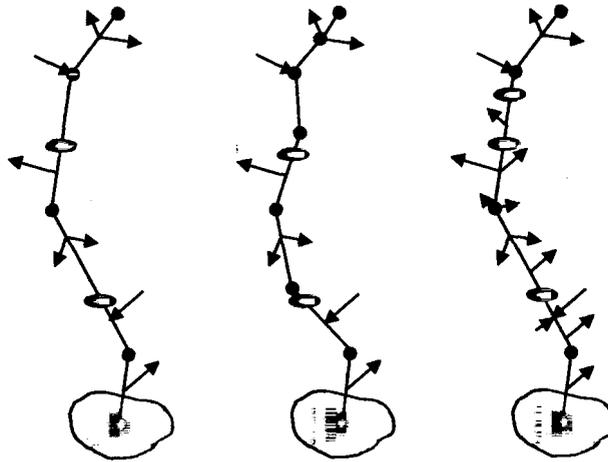


Figure 6.11: Generalization in spine and prongs.

In summary, the generalization parameters are listed below:

1. Number of movement nodes,
2. Percentage of global inner change,
3. Percentage of local inner change.

The result is similar to cartographic generalization. However, the step of the scale change according to our analysis is not stable but relies on the significance of information that is implied. This significance is defined by the area and movement attributes of the moving phenomenon and is captured by the algorithms defined above.

6.5 Reconstruction of a S-T Instance

In the definition of the S-T helix we included the initial instance of the phenomenon description. Thus, the S-T helix constitutes a model map of the described phenomenon. As we explained in the previous section, this map can either represent accurately the input space by using adequate generalization parameters, or represent the input space in an abstract manner and thus lower the computational volume for post processing.

According to our model in order to reconstruct the phenomenon for a random temporal instance we rely on the neighbor spine and prong descriptions equation (6.8):

$$\text{Create_instance} = Ph^{(t)} = (x, y, q_s, q_o, r_1), \quad (6.8)$$

The (x, y) coordinates are estimated through interpolation between the previous and next spine nodes where we have a full description of the movement of the phenomenon. The qualifiers for each instance t , are deduced by the qualifiers of the previous spine node according to equation.(6.9).

$$(x_t, y_t, q_s, q_o) = \left(x_1 + \frac{(x_2 - x_1) \cdot (t - t_1)}{t_2 - t_1}, y_1 + \frac{(y_2 - y_1) \cdot (t - t_1)}{t_2 - t_1}, q_{s,t_1}, q_{o,t_1} \right), \quad (6.9)$$

where (x_1, y_1, t_1) and (x_2, y_2, t_2) are the coordinates of the previous and next spine nodes.

The azimuths and norms of the prongs describe local temporal instances and thus they cannot be interpolated to provide a prong description for a random temporal instance. However, the prongs that describe overall changes (expansion or shrinkage)

throughout the phenomenon can be interpolated to the temporal instance of interest and provide a metric of expansion or shrinkage r_1 .

In order to acquire the history of a temporal increment $\Delta t=(t_1, t_2)$ we consider the temporal instances of the starting and ending time complemented by the spine and prong nodes between these two instances as described by equation (6.10).

$$\text{Create_history} = Ph_{t_1}^{t_2} = (Ph^{(t_1)}, \sum_{t_1}^{t_2} ns^{(i)}, \sum_{t_1}^{t_2} np^{(j)}, Ph^{(t_2)}), \quad (6.10)$$

Again according to generalization variables we may have more or less detail in the approximation of the position and shape tendencies of the phenomenon at hand.

6.6 Summary

In this chapter we defined a framework for the spatio-temporal analysis of motion imagery (MI) datasets depicting two-dimensional phenomena evolving in time. More specifically, we introduced the concept of a spatio-temporal helix as a concise representation of spatio-temporal events, modeling their path in space and the variations of their outline. We presented in some detail the automated algorithms developed to support the automated generation of spatio-temporal helixes and discussed their potential to support spatio-temporal analysis.

CHAPTER 7

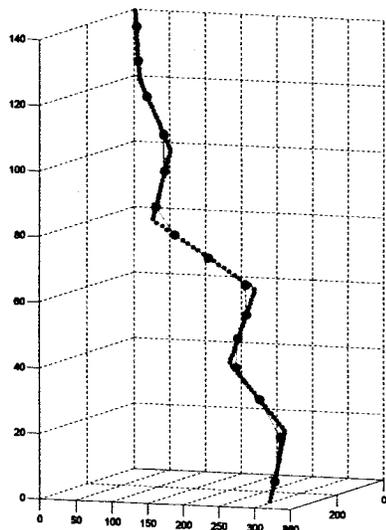
EXPERIMENTS

The approaches described in this thesis have been implemented in the MATLAB environment. Creation of synthetic datasets of moving objects upon digital images took place, in order to use them in the presented experiments. Also random behaviors of spatio-temporal trajectories were generated to describe objects moving with stable or variable velocity.

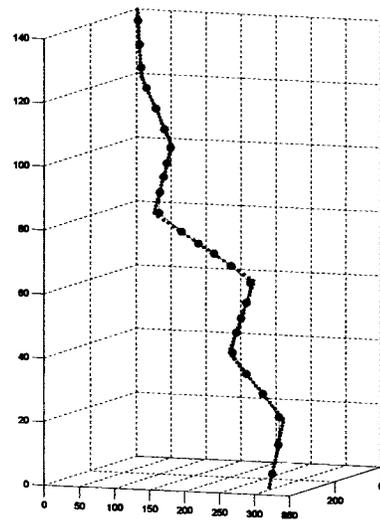
7.1 Single Trajectory Summarization Through Classic SOM

The original SOM algorithm is used to generalize single moving object trajectories as described in chapter 3. The following figures demonstrate the SOM generalization of two trajectories using different number of nodes. The method is bound to the sensitivity according to the number of nodes selected.

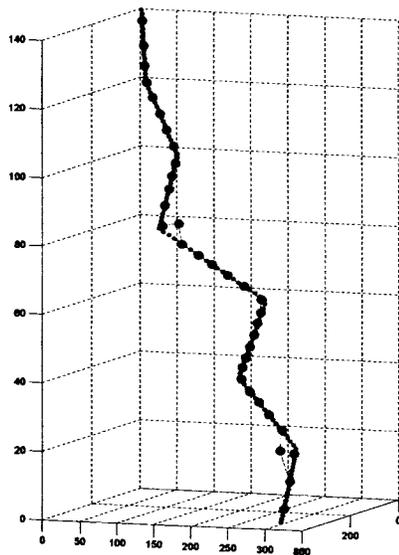
As shown in figure 7.1 a high capture error is evident in the trajectory curves. The blue dots represent the input dataset, while the red nodes are the SOM generalization nodes. The input trajectory is formed from a 140 frame sequence.



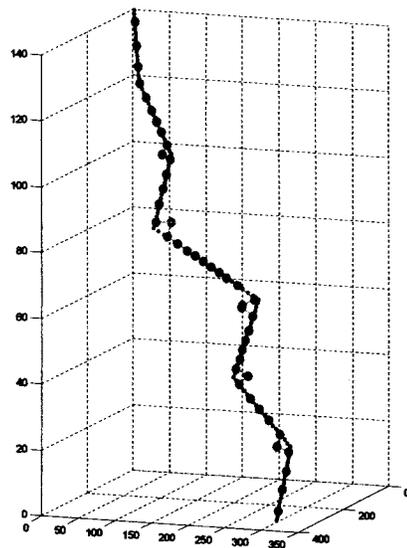
a.



b.

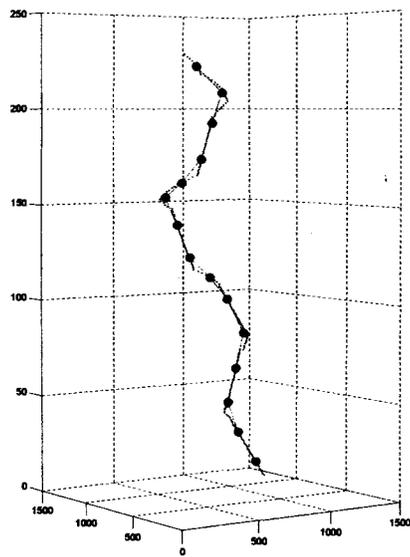


c.

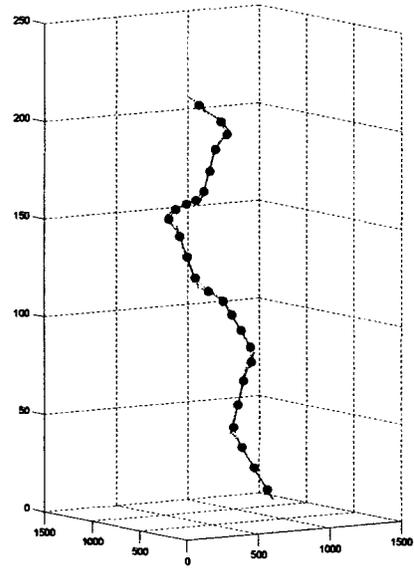


d.

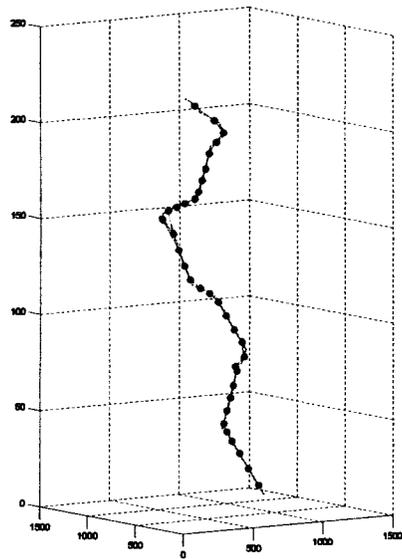
Figure 7.1: Generalization of a 140 frame trajectory using a)15, b)25, c)35, d) 45 SOM nodes.



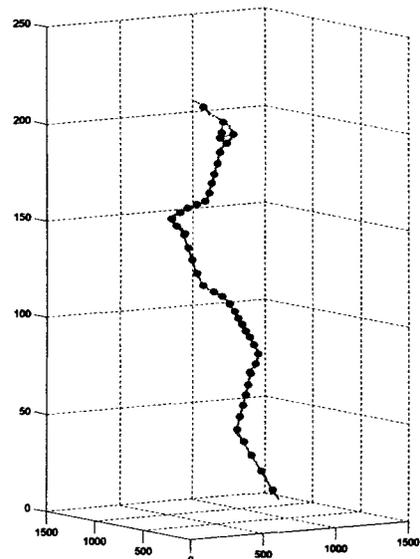
a.



b.



c.



d.

Figure 7.2: Generalization of a 226 point trajectory using a)15, b)25, c)35, d) 45 SOM nodes.

Again, as shown in figure (7.2) the trajectories are poorly represented by the SOM node sequence. In figures 7.1c,d and 7.2d over-parameterization urges the nodes to abruptly separate from the input dataset.

7.2 SST-SOM Trajectory Generalization

The parameters involved in this experimental section are mostly associated with the equation (7.1) introduced in chapter 3.

$$\#of_nodes = \frac{angle_max - angle_node}{g} + 2 \quad (7.1)$$

By altering the variables (*angle_max*) and (*g*) we get different generalization results. Another variable that is used is the number of initial SOM nodes that forms the rough generalization representation upon which the rest of the process relies.

In figure 7.3 we discern that the SST-SOM nodes describe the trajectory much better than the standard solution. The SOM algorithm (green nodes on figure 7.3 left) was unable to capture some areas of the original trajectory. On the other hand, the red nodes on figure (7.3) represent better the original dataset. The precision measurements vary according to the complexity of the trajectories. By using equation 7.2 for this example the SOM solution yielded a deviation of 89 pixels while the SST-SOM solution yielded a 9 pixel deviation.

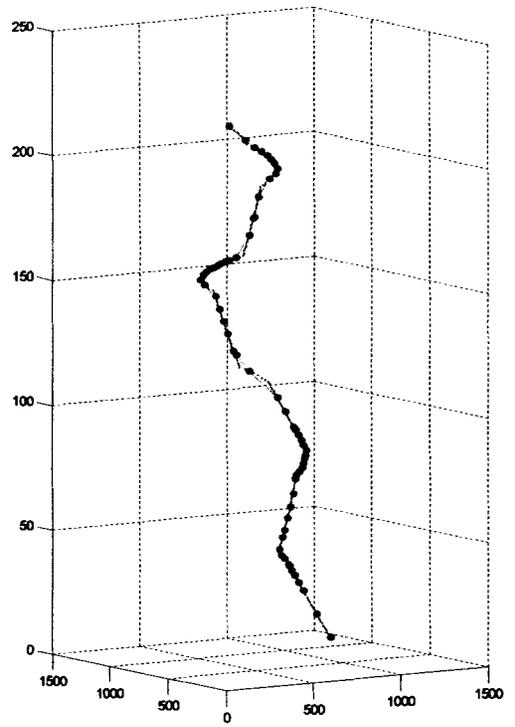
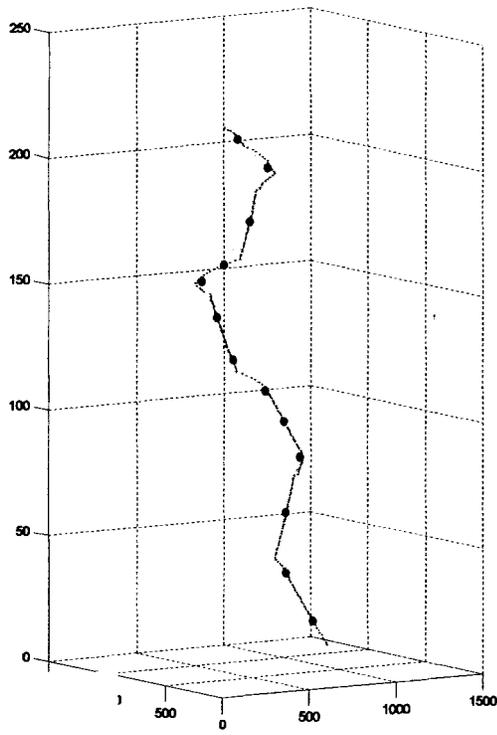
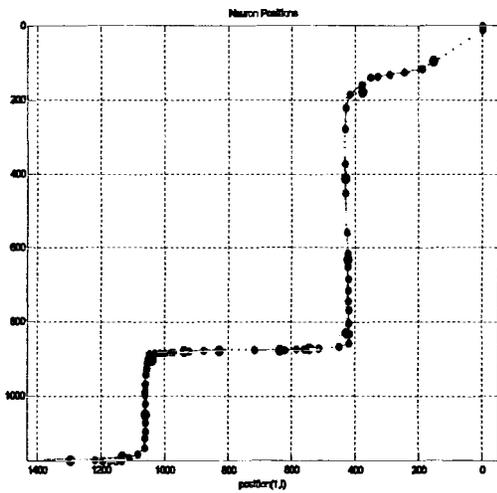
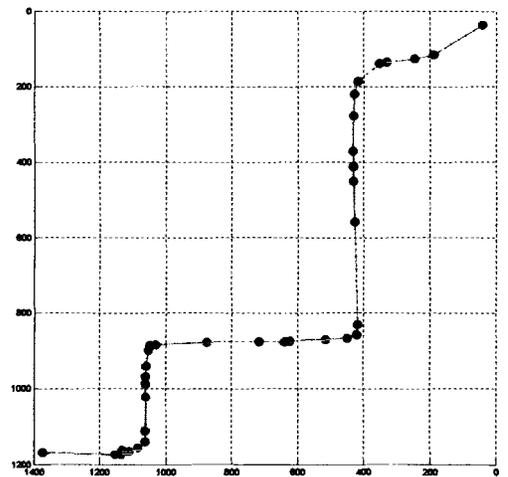


Figure 7.3: SOM vs. SST-SOM generalization.



a



b

Figure 7.4: SST-SOM generalization in 2-D and thinning result.

In figure 7.4a, the original dataset is represented by blue dots, the SOM solution is represented by green nodes, while the SST-SOM solution is depicted by red nodes. In figure 7.4b the red nodes form the final generalization output after the thinning phase. Accordingly, in the remaining figures (7.5) we acquire the following generalization results by using the parameters listed in table (7.1).

Figure reference	Initial nodes	g	angle_max	Resulting nodes	After thinning
-	13	8	195	67	37
a	25	15	195	59	34
b	25	10	190	45	32
c	25	15	190	40	24

Table 7.1: Generalization parameters for SST-SOM generalization procedure.

As demonstrated from the experiments the (angle_max) and (g) variables play the core role towards the generalization volume. The initial number of nodes is also an important variable, as it defines a general frame upon which further analysis is based.

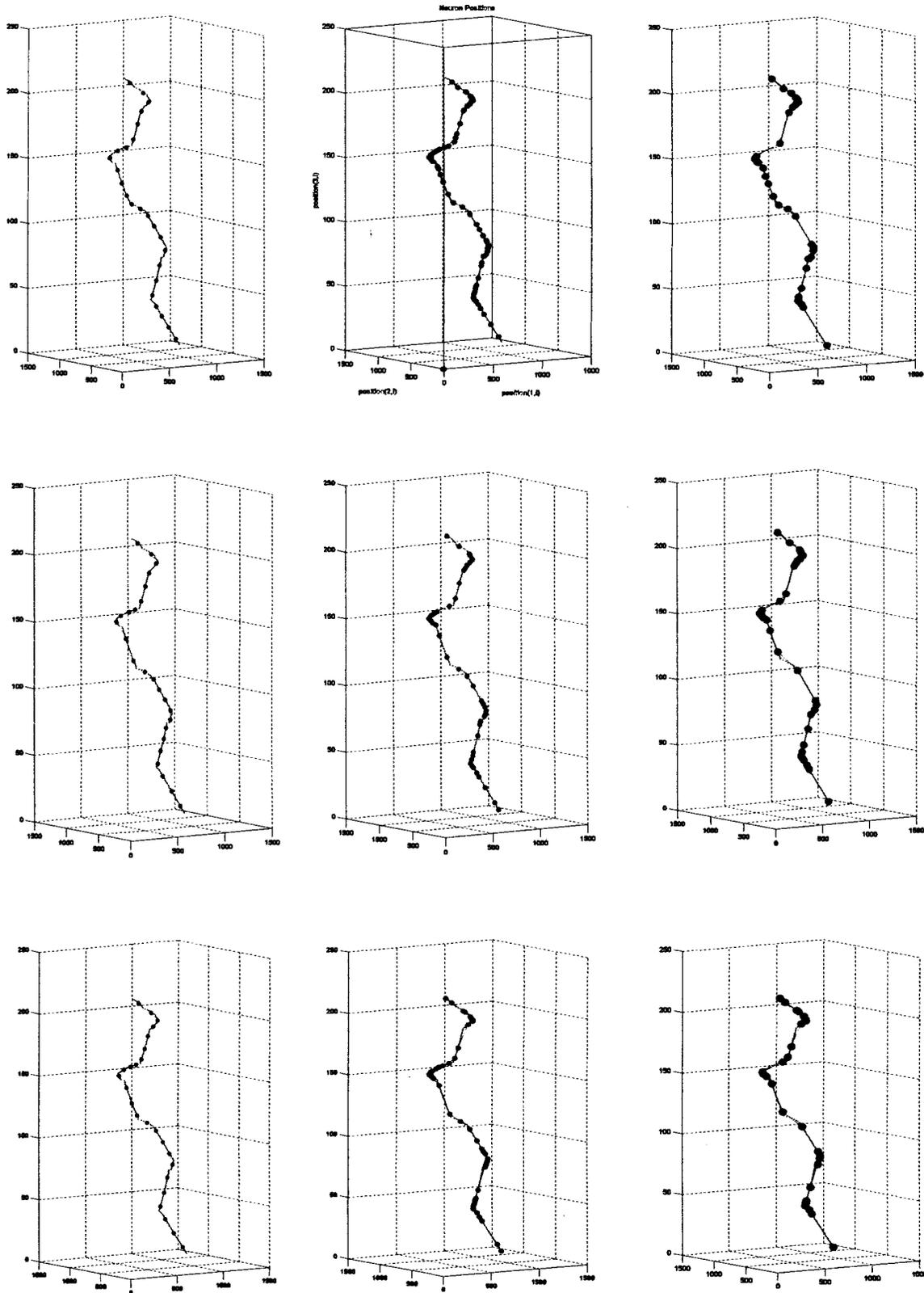


Figure 7.5: Different generalization solutions in a 419 point dataset.

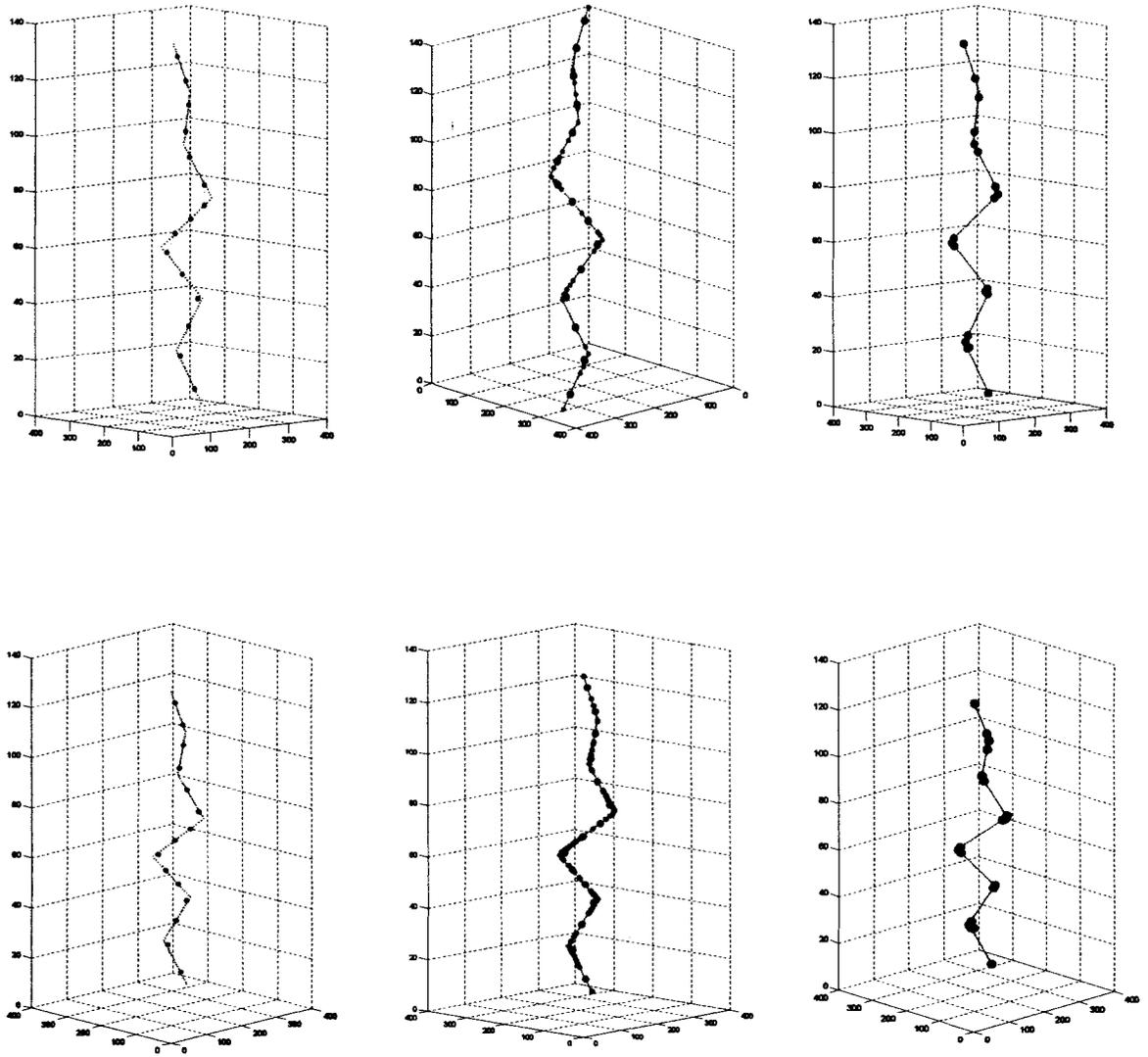


Figure 7.6: SST-SOM phases for a 225 point dataset.

In figure 7.6 the same dataset is generalized using 35 and 60 nodes. Note that the thinning procedure yields a 20 node chain in both cases. That means that the trajectory is smooth enough so two generalizations may initially give a different number of nodes, yet thinning will eventually capture the variances which in that case are limited.

7.3 Errors and Noise

In order to quantify the geometric similarity matching between trajectories we use a measure of the spatio-temporal distance d_{ST} between two trajectories or solutions i and j . It is provided by the equation (7.2):

$$d_{ST} = \sqrt{\frac{\sum d_n^{ij^2} + d_m^{ji^2}}{m+n}} \quad (7.2)$$

where m is the number of SOM nodes along trajectory i , n is the number of SOM nodes along trajectory j and d^{ij} is the distance between the trajectory and the generalization nodes in the ST domain.

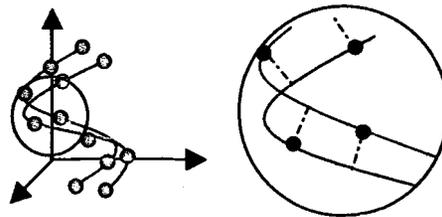
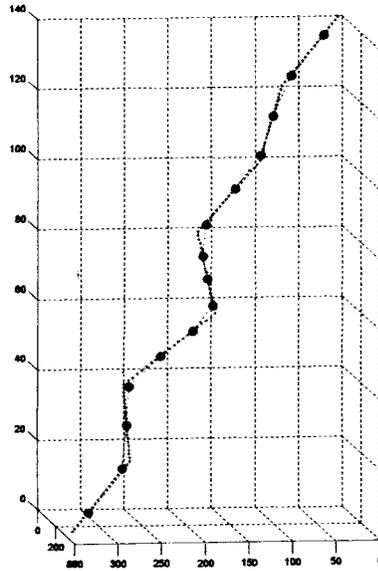


Figure 7.7: Node distances in the 3-D ST domain.

We adopt this RMS type of error to demonstrate the precision of the SOM method and its behavior with respect to occlusion and noise. Each point in the dataset corresponds to an observation of the same object in different MI frames. These points may not be present in a continuous rate in the dataset since there might be some information gaps (e.g. the object was not visible or not extracted in every frame). In addition, misclassification and noise are anticipated in trajectory datasets as discussed in chapter 3.

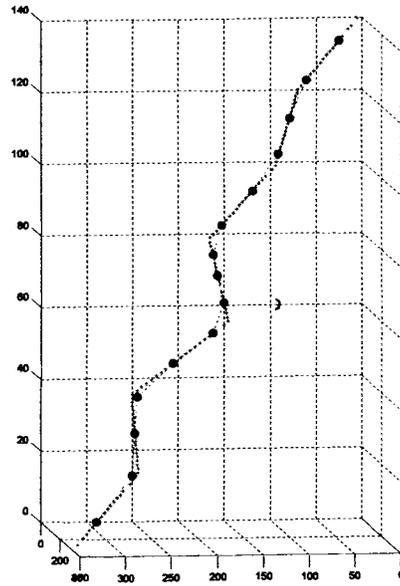
Occlusion error was simulated by randomly erasing point data from the trajectory dataset according to the percentage of occlusion. Noise was added by altering the point data in a random x and y position not exceeding the limits of the trajectory spatial coordinate boundaries. For the following figures we provide the RMS error and the medium distance between the nodes and the dataset as a precision measure.

The input dataset without any occlusion or error is shown in figure (7.8) (blue dots) while the SOM solution is depicted in red nodes. The first number in the parenthesis accompanying the figure is the RMS error and the second is the medium distance from each node to the dataset.

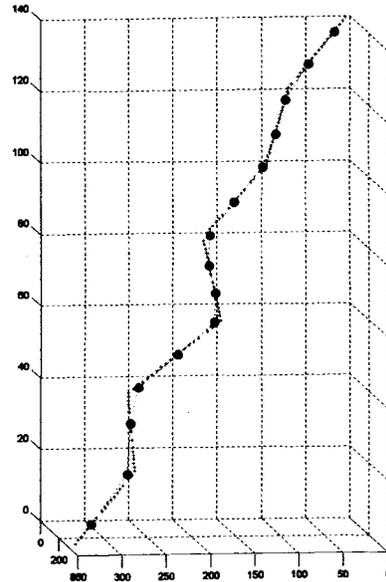


0% occlusion, 0% noise (3.7, 13.9)

Figure 7.8: SOM solution to a non-erroneous dataset.

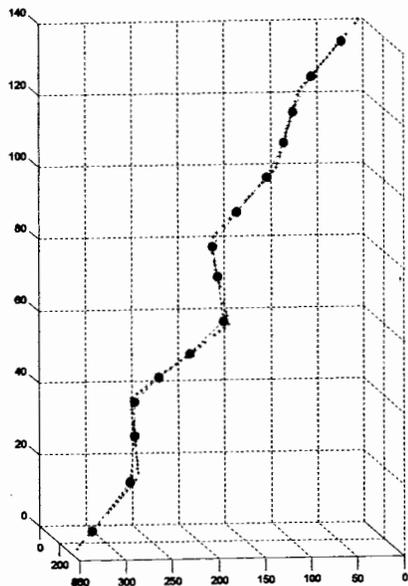


a. 5% occlusion, (3.9, 15.5)

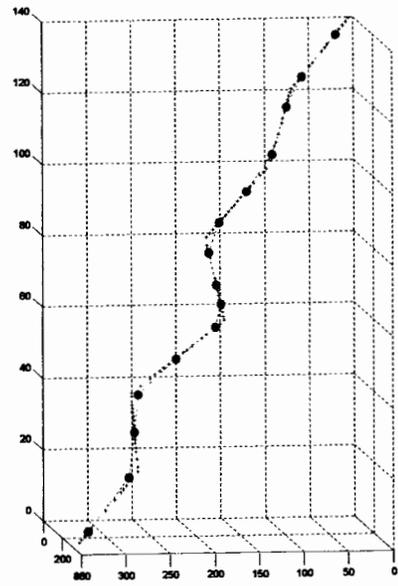


b. 10% occlusion, (3.9, 14.9)

Figure 7.9: SOM solution for: a) 5% occlusion, b) 10% occlusion.

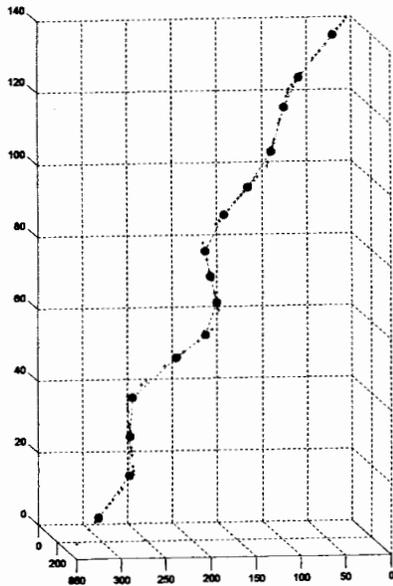


a. 20% occlusion, (3.9, 15.0)

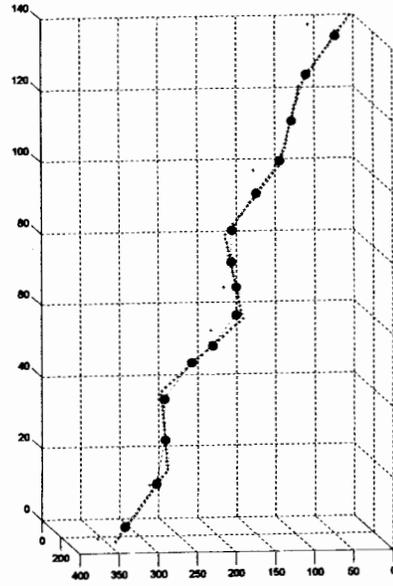


b. 30% occlusion, (3.9, 15.0)

Figure 7.10: SOM solution for: a) 20% occlusion, b) 30% occlusion.

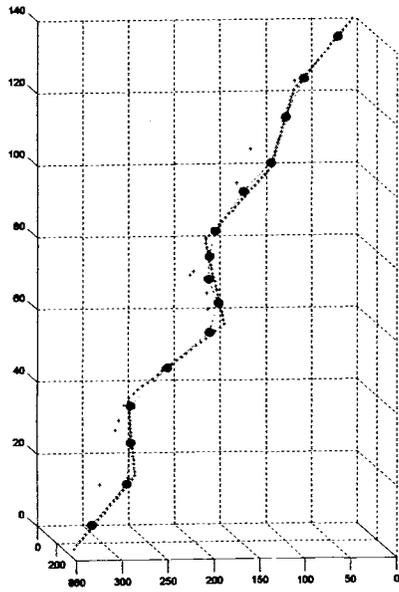


a. 50% occlusion, (4.3, 18.2)

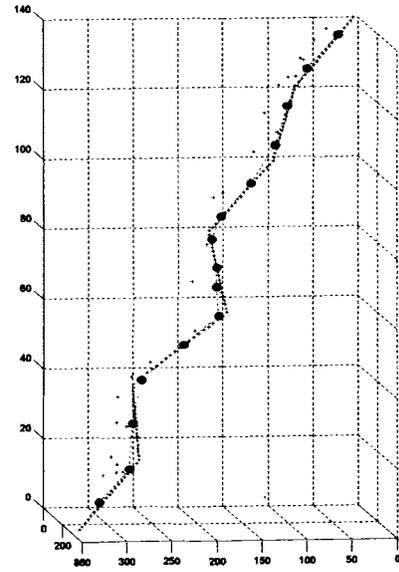


b. 5% noise, (4.6, 21.1)

Figure 7.11: SOM solution for: a) 50% occlusion, b) 5% noise.

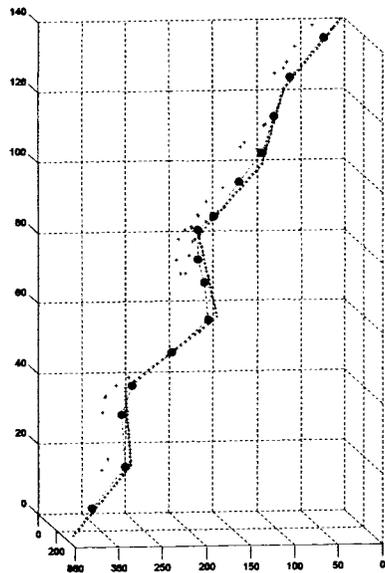


a. 10% noise, (4.3, 8.6)

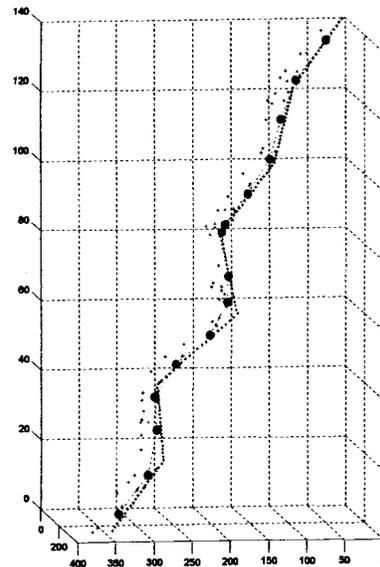


b. 20% noise, (4.6, 21.2)

Figure 7.12: SOM solution for: a) 10% noise, b) 20% noise.

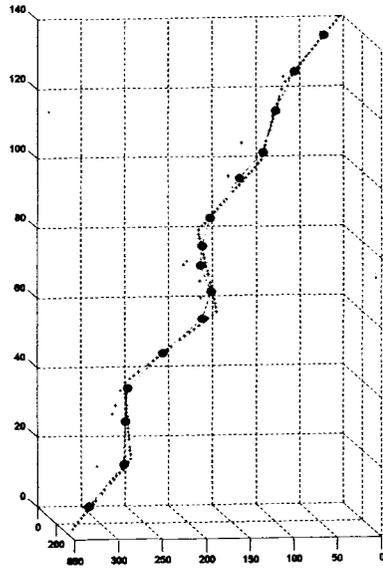


a. 30% noise, (4.8, 23.0)

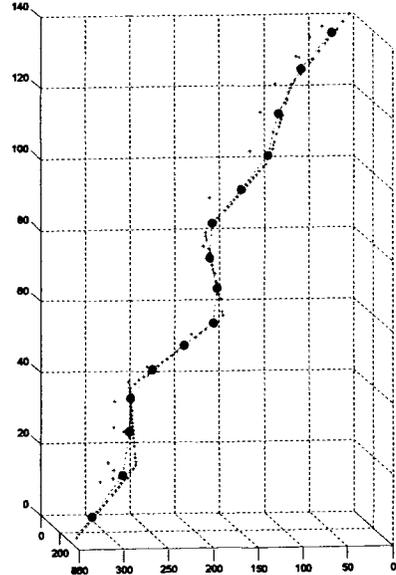


b. 50% noise, (6.2, 38.1)

Figure 7.13: SOM solution for: a) 30% noise, b) 50% noise.

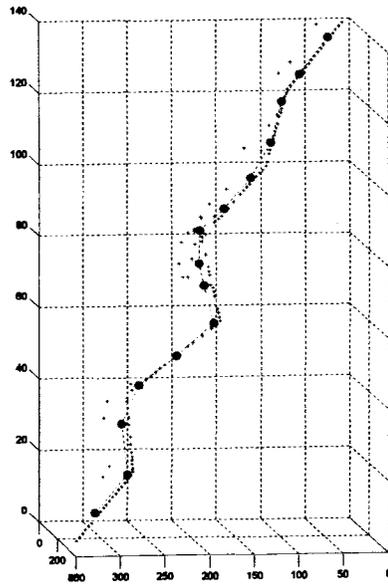


a. 10% occlusion+10% noise, (4.3, 18.5)



b. 20%occlusion+20% noise, (4.5, 20.3)

Figure 7.14: SOM solution for: a) 10% occlusion and 10% noise, b) 20% occlusion and 20% noise.



30%occlusion +30%noise (5.6, 31.5)

Figure 7.15: SOM solution for 30% occlusion and 30% noise.

The results are shown in the chart of figure (7.16). The bubble sizes correspond to the resulting error.

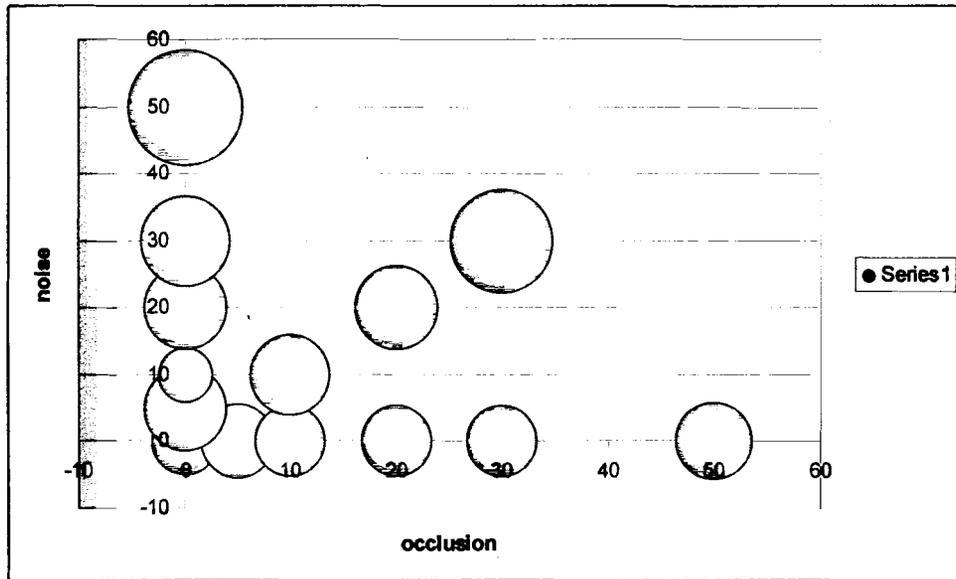


Figure 7.16: Medium distance from nodes to input dataset according to errors.

Usually, the errors are larger when occlusion and noise rise. But for small occlusion and noise percentages results of similar quality with the original dataset are revealed. In some cases a higher noise or occlusion percentage yields smaller errors. This is evident since the occluded data or the noisy ones are such that the generalization properties of SOM perform better to describe the original dataset.

7.4 Multiple Trajectory Summarization and Generalization

7.4.1 Unrelated Road Segments.

Based on the described model in chapter 4, for the grouping of trajectories traveling in distinct road segments we have the following input dataset shown in figure 7.17.

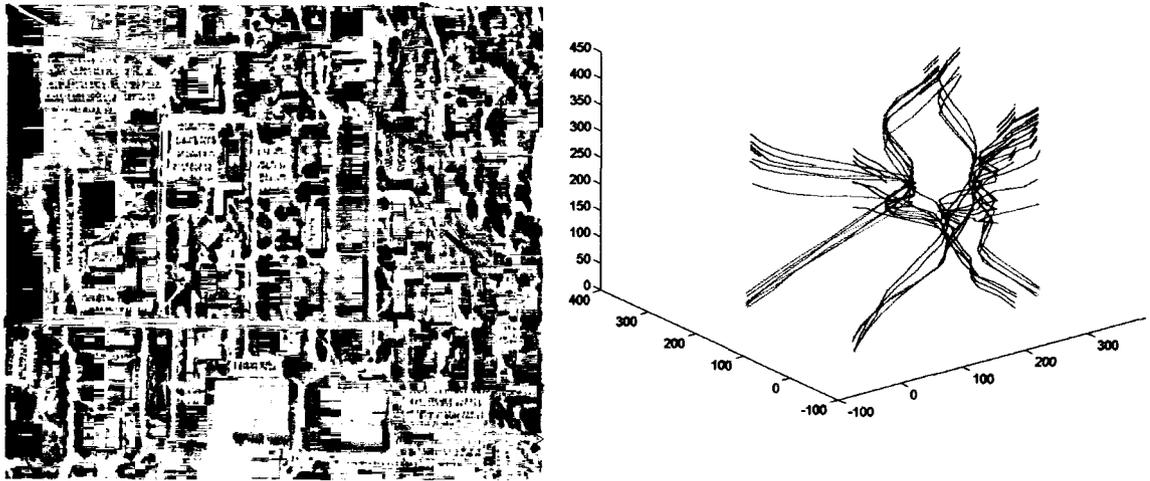


Figure 7.17: Unrelated road segment input dataset.

The dataset is comprised of 30 trajectories moving in both directions in the three road segments shown in figure 7.17 left. For the following examples we use a spatio-temporal neighborhood unit of cubical shape. The spatial dimensions are given by the ‘sgroup’ variable, while the temporal dimension (height) is given by the ‘tgroup’ variable. Different colors portray different resulting groups. For visualization or further processing each trajectory of the same color is represented by a single medium or leading trajectory. The black colored trajectories are the ones that couldn’t be classified as part of any group.

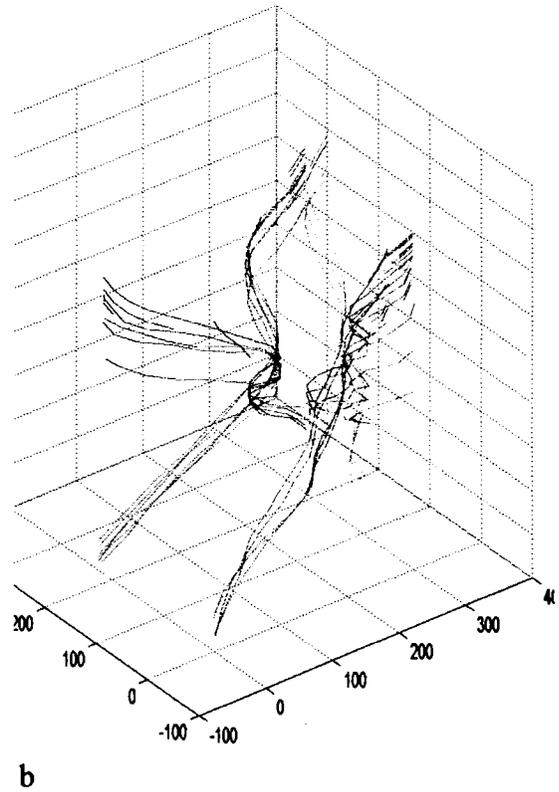
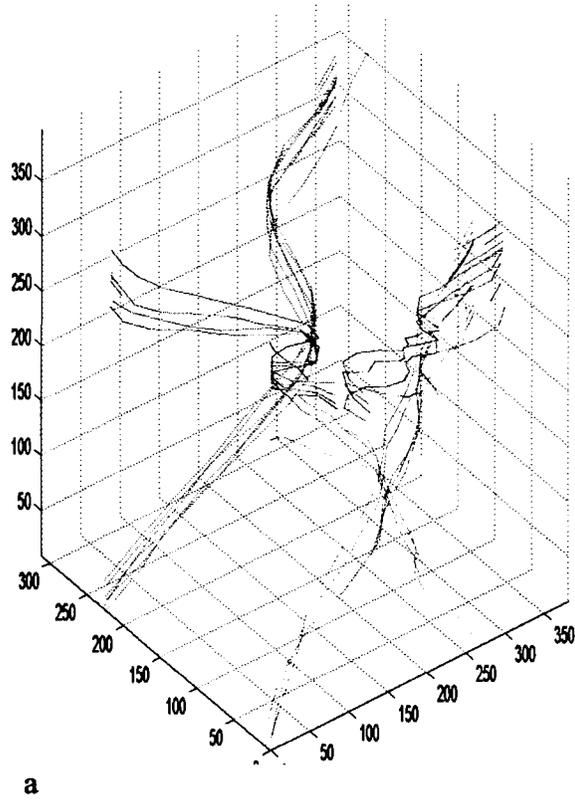


Figure 7.18: MUSTT using (tgroup, sgroup): a) (50,50) and b) (100,100).

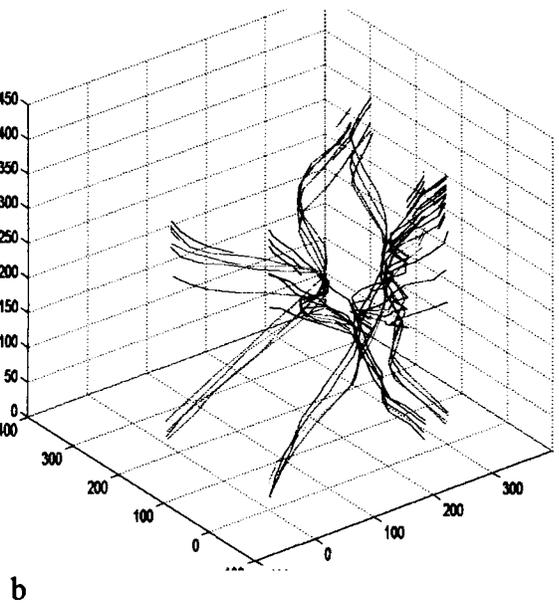
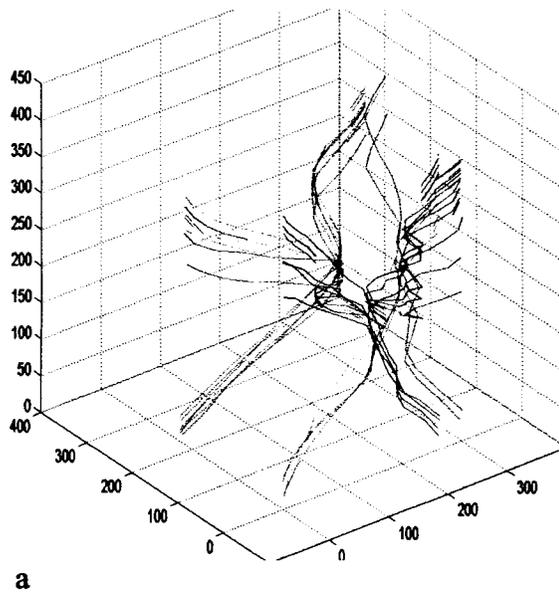


Figure 7.19: MUSTT using (tgroup, sgroup): a) (30,30) and b) (15,15).

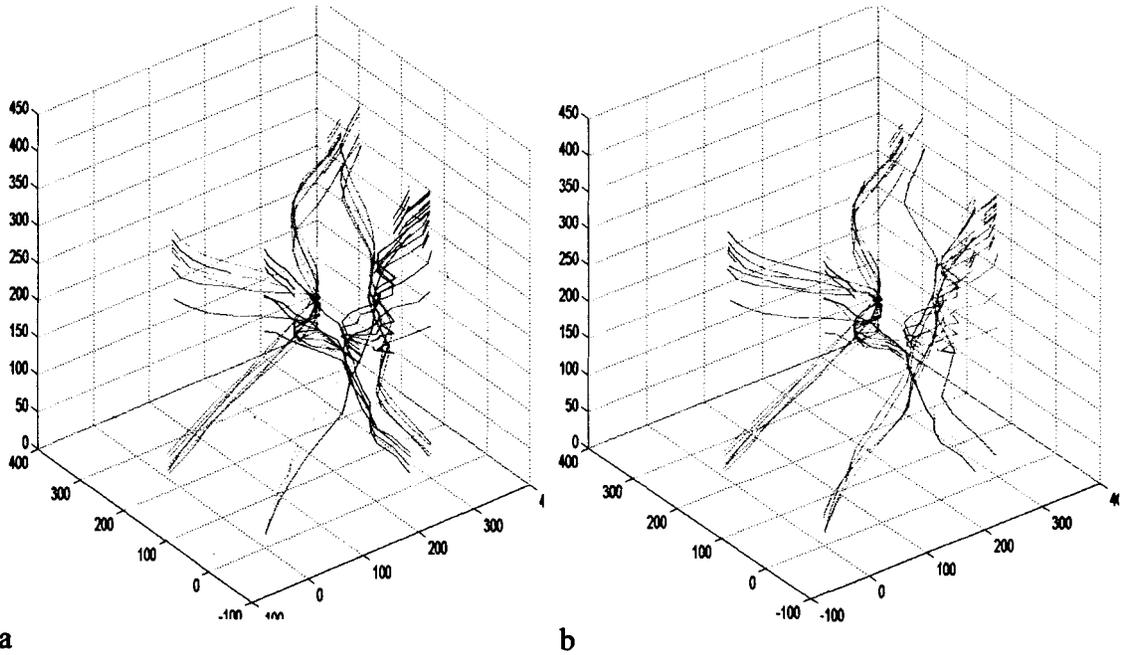


Figure 7.20: MUSTT using (tgroup, sgroup): a) (80,20) and b) (20,80).

In table (7.2) sets of trajectories are grouped together to form the various classes according to the (tgroup, sgroup) S-T neighbor unit.

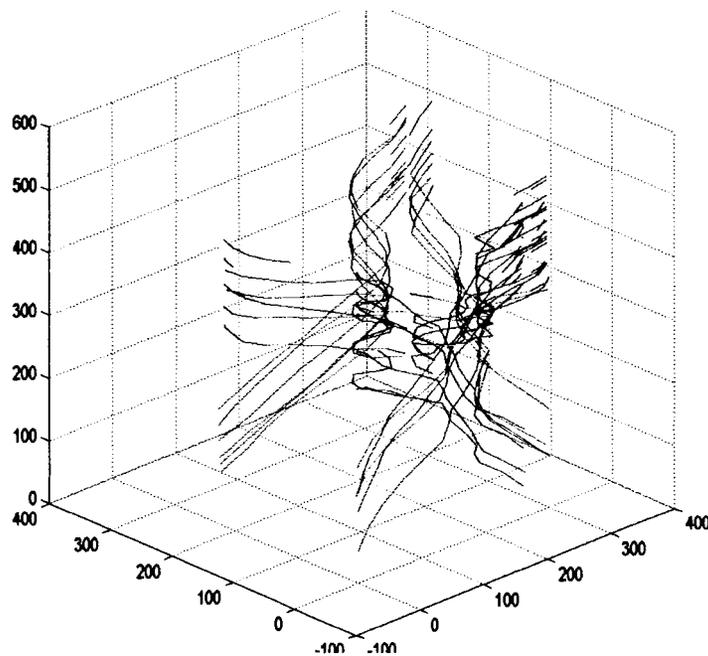
tgroup	sgroup	Class1	Class2	Class3	Class4	Class5	Class6
100	100	6-10,29,30	11-16	1-5	20-24	25-28	17-19
50	50	6-9,30	20-24	1,2,4,5	11-13,16	17-19	25,26,27
30	30	6-9,30	2,4,5	11,12,13	20-22	17,19	23,24
15	15	4,5	8,9	11,13	20,21		
80	20	7-9,30	2,4,5	11,13	17,19	20,21	23,24
20	80	6-9,10,30	20-24	1,2,3,4,5	11-16	17-19	25,26,28

Table 7.2: Grouping results for temporally close trajectories.

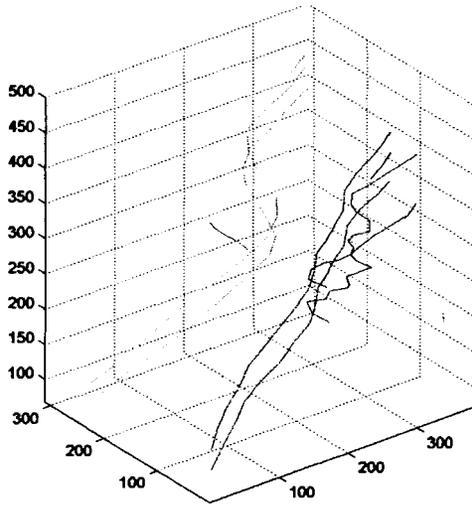
In most cases the number of classes was 6 and this is what was anticipated since there are 3 road segments and two directions for each segment. In addition, the temporal distance between the trajectories was relatively small. In the case of (15, 15) unit

definition the MUSTT algorithm grouping was quite poor because of the restricted dimensions of the S-T neighborhood unit.

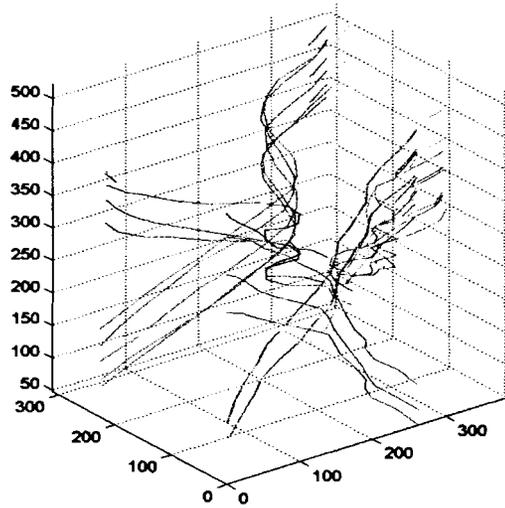
For the next set of experiments the trajectories are now more separated in their temporal distribution. Grouping will accordingly provide more classes than the previous set of experiments. Nevertheless, if we select a larger (tgroup) variable the algorithm would be able to form fewer groups.



7.21: Input dataset for temporally distant trajectories.

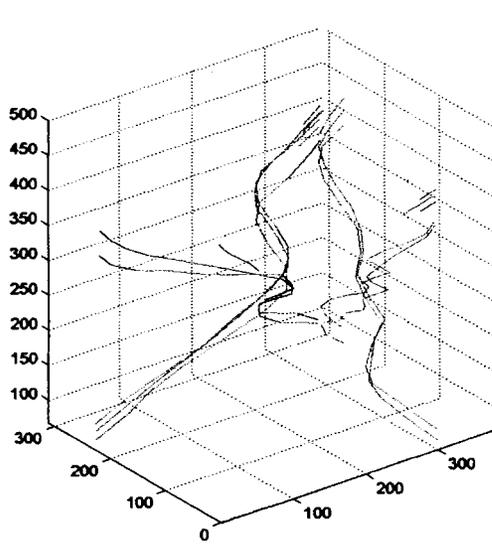


a

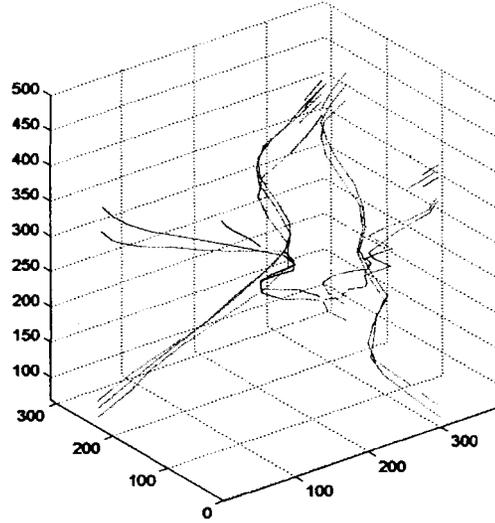


b

Figure 7.22: MUSTT using (tgroup, sgroup): a) (30,30) and b) (50,50).



a



b

Figure 7.23: MUSTT using (tgroup, sgroup): a) (80,20) and b) (20,80).

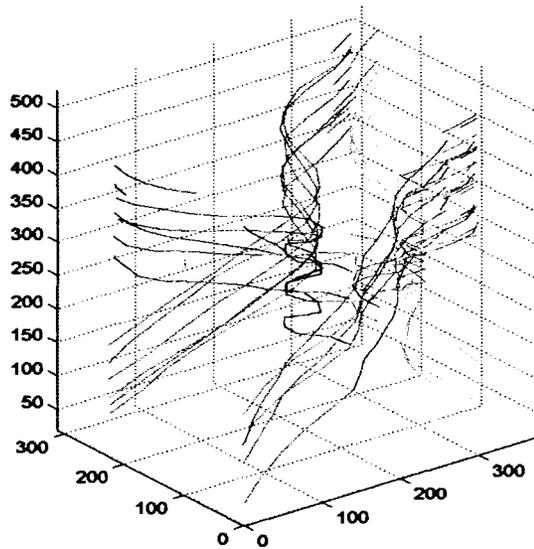


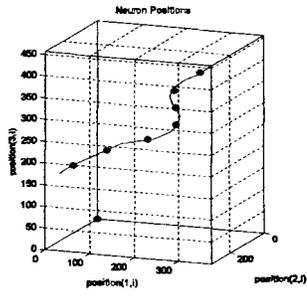
Figure 7.24: MUSTT using $(tgroup, sgroup)=(150,150)$.

In table (7.3) the groups of trajectories are listed according to the grouping variables.

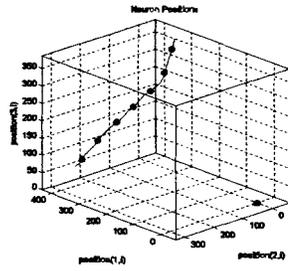
tgroup	sgroup	Class1	Class2	Class3	Class4	Class5	Class6	Class7
50	50	6-9,30	11-13,16	1,2,5	25-27	18,19	20,21	23,24
30	30	11,12	8,30	23,24				
80	20	7,8,9,30	2,4	11,13	17,19	20,21	23,24	
20	80	9,10,30	11,13	1,2	18,19	21,23		
150	150	6-10,29,30	11-16	1-5	20-24	25-28	17-19	

Table 7.3: Grouping results for temporally distant datasets.

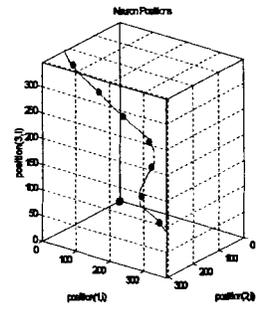
In figure (7.25) the SOM solutions for the leading trajectories of each group form the basic units where visualization is based. In the parenthesis accompanying each figure, the number of trajectories included in the group, the starting time, ending time and the range of the group are the variables that will provide additional information for the summary visualization.



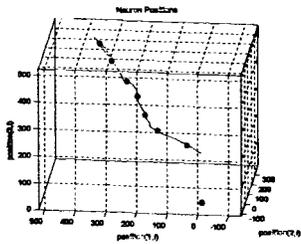
(7,155,455,150)



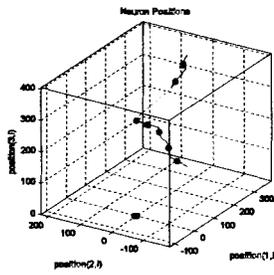
(6,67,387,150)



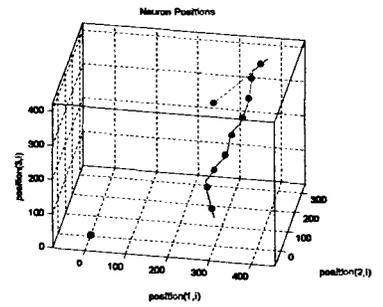
(5,88,348,150)



(5,180,520,150)



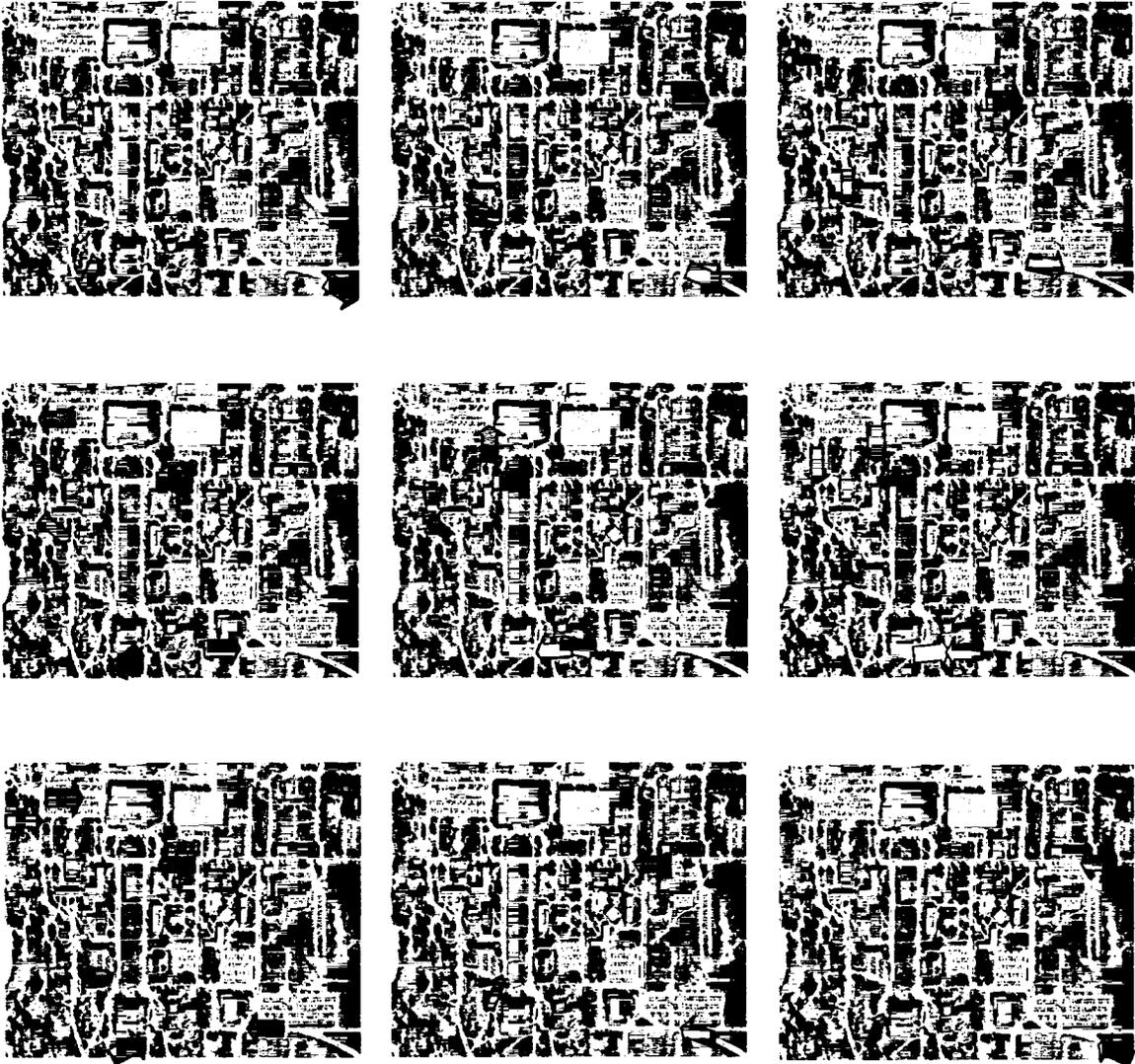
(4,47,407,150)



(3,157,497,150)

Figure 7.25: SOM solutions for leading trajectories.

Figure 7.26: Snapshots of product summary.



The visualization part is not explicitly discussed in this thesis. However, a rough visualization product is shown in the following figures. The actual summary is a video showing a set of flying vectors according to the movements of the objects included. Each vector's size is estimated according to the number of objects it represents. In figure (7.25), snapshots of the summary video show the placement of the flying vectors.

7.4.2 Related Road Segments

As discussed in chapter 4 in the case of related road segments such as those seen in figure 7.27, we divide the dataset into separate segments. (Figure 7.28).

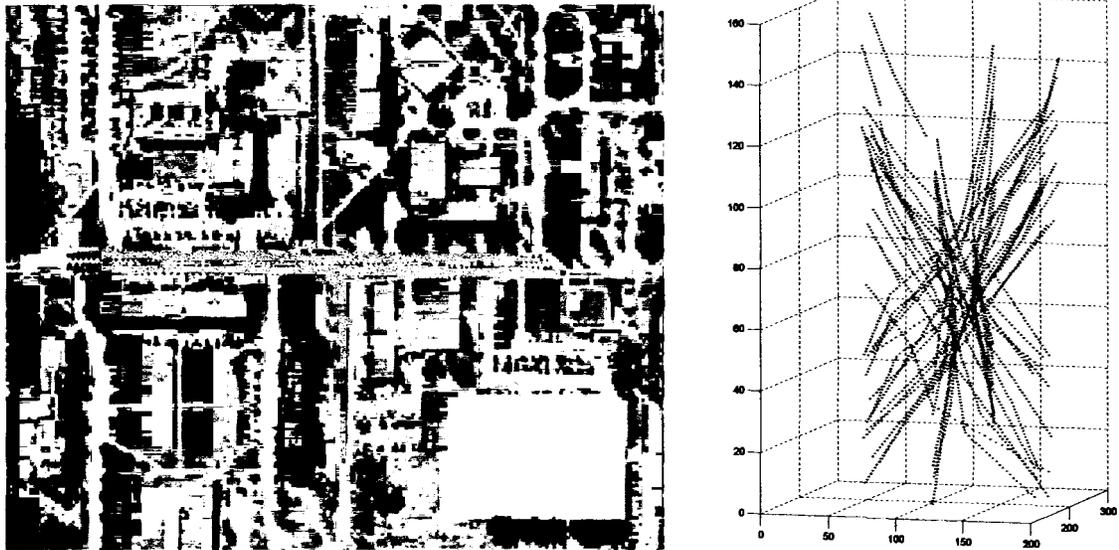


Figure 7.27: Input dataset for related road segments.

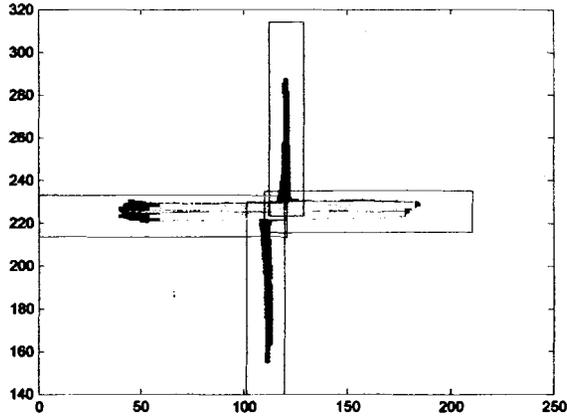


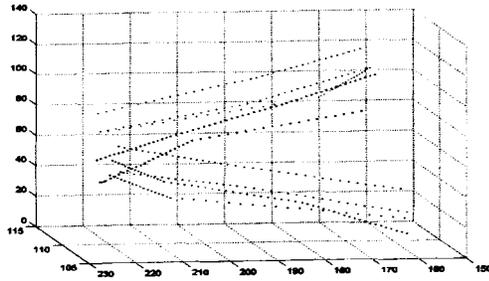
Figure 7.28: Separation of road segments.

Again, we proceed with the MUSST algorithm procedure for distinct road segments. The number of separated segments is 65. The results are shown in the following figures. The (tgroup, sgroup) variables are set to 30 units.

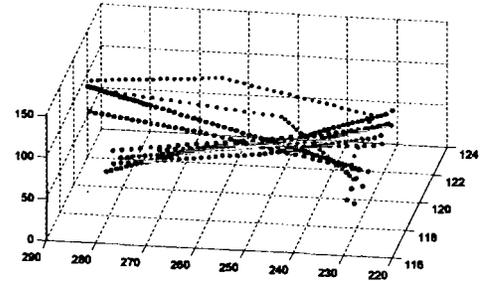
In table 7.4 the trajectories are grouped together to form the various classes according to the (tgroup, sgroup) S-T neighbor unit.

	Class1	Class2	Class3	Class4
North	6-9	1,4,5	2,3	
South	5-9	1-3	2,4	
East	1-7,13,14,20,21,23,25-28	8,9,11,12,15,24	16-18,29	
West	2-4,7,14,16,21,22	8,9,11,12,17,20	1,6,13,15	18,19,23

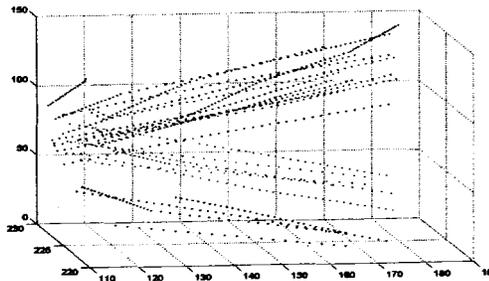
Table 7.4: Grouping results for related road segments.



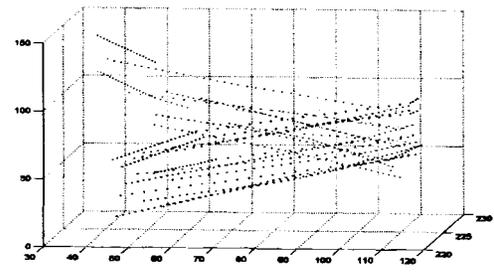
North



South



East



West

Figure 7.29: MUSTT trajectories in the separated directions.

In table 7.5 the formed groups include leading trajectories, which provide the starting-ending time. In each record the first variable is the number of represented points, while the second and third provide the starting and ending time.

N	(4,31,69)	(3,82,120)	(2,38,80)	
S	(5,35,72)	(2,88,125)	(2,59,100)	
E	(16,89,127)	(6,47,77)	(4,1,20)	
W	(8,12,52)	(6,78,106)	(3,55,90)	(4,48,88)

Table 7.5: Statistics of formed groups.

The snapshots of a potential video visualization are shown in figure 7.30.

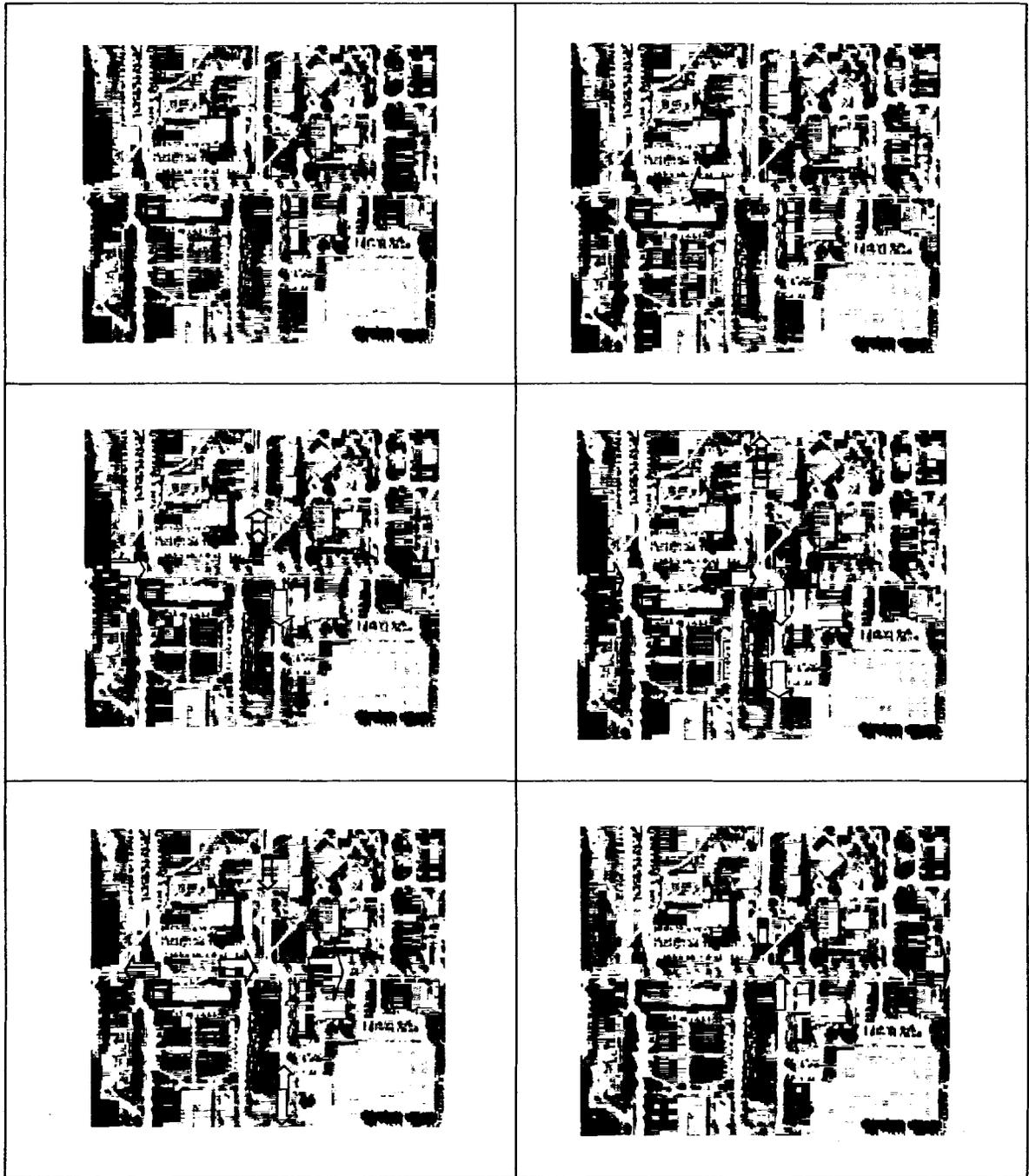


Figure 7.30: Snapshots of product summary.

7.5 Trajectory Classification

According to the theoretical aspects discussed in chapter 5 the following experiments show the capabilities and indicate the limits of the ACCENT algorithm analysis.

7.5.1 Classification of Branches

The input dataset are comprised of data points including coordinates of the form: (x, y, t, color, shape). In figure (7.31a) the trajectories are distant, while in (7.31b) they are entangled and thus make the task of classification more demanding. The different colors indicate the two trajectories for visualization purposes, yet the input dataset does not contain any explicit information of the identity, geometry or form of the trajectories.

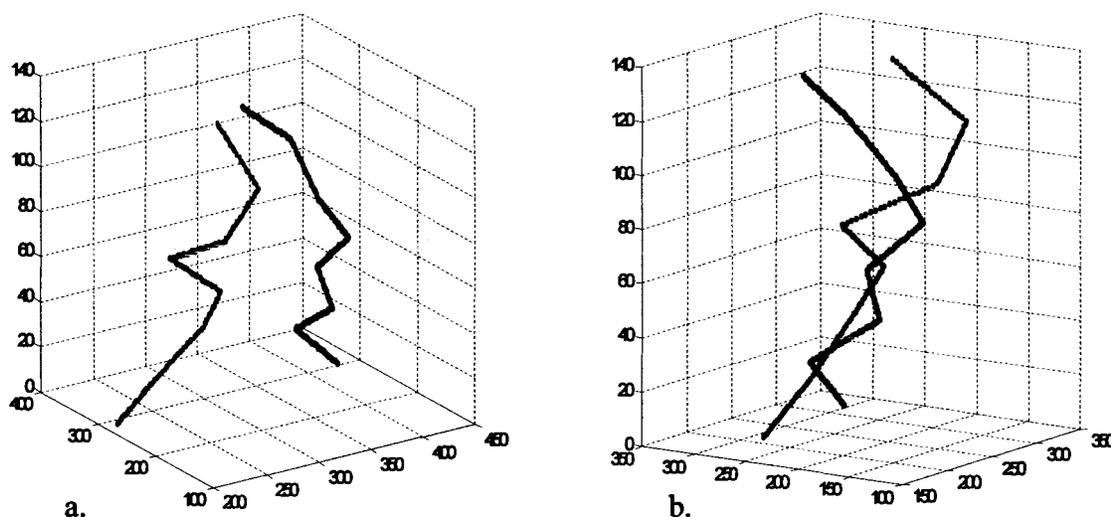


Figure 7.31: Distant and entangled input points that form two trajectories.

In the following experiments we use a pair of attribute percentages that accompany the datasets. The first percentage is the similarity of the color between the two trajectories, and the second percentage portrays the similarity between the sizes of objects as they are captured throughout the MI dataset. Since we control the test data, we know the range of the values of attributes for both color and size. In order to demonstrate the extent of the method's ability to classify, we use different percentages of similarity between the attributes.

The output classification is then compared to the known classified trajectories and two percentages are used to evaluate the outcome. First, the percentage of wrongly classified points shows how many points of those that were actually classified are classified under the wrong class and from now on it will be referenced as PM. Second, the percentage of unclassified points simply shows how many points couldn't be classified under any class and this percentage will be referenced as PU. These percentages are utilized both after the back-propagation process and the SOM carrier analysis as discussed in chapter 5.

In the case of distant trajectories including data of different attribute similarity percentages we easily separate the data since the branch selection includes the trajectories on their whole. Figure (7.32).

The numbers accompanying the figures show the pair of attribute overlap percentages. The outcome as expected is identical for both cases. It is apparent that since the trajectories are distant the classification is mostly based on the geometric spatial

separation and the result is flawless. Actually, a geometric proximity test would also result in a perfect classification.

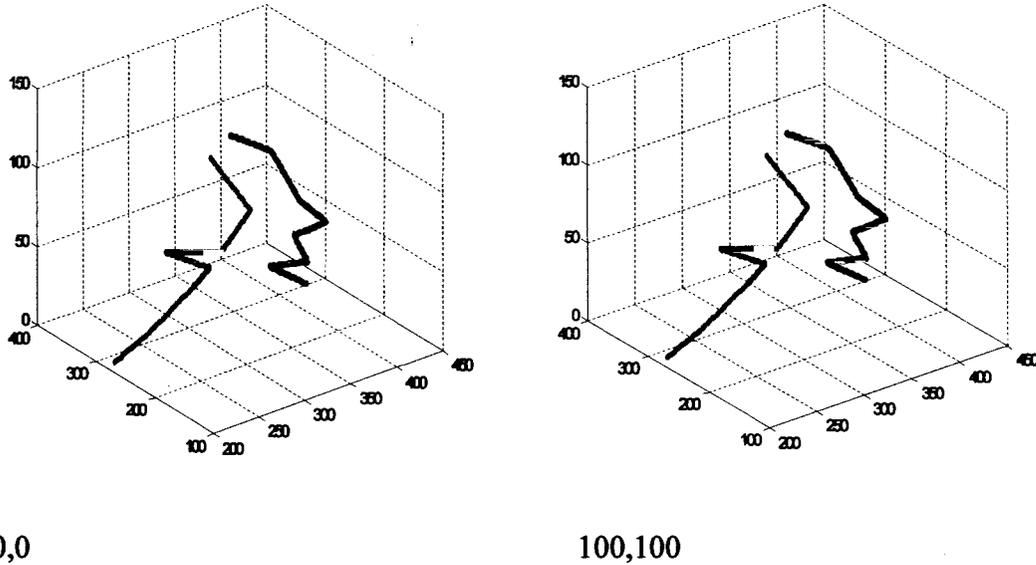


Figure 7.32: Classification result for distant trajectories.

In the case of the geometrically close trajectories the following figures portray each step of the process for the case of (0,0) attribute overlap. Figure (7.33) left shows the branch formation, while figure (7.33) right shows the k-means center and cluster formation on the attribute space. Mapping of the output space and backpropagation (BP) convergeness error are seen in figure (7.34) left. Finally, the resulting classification outcome is shown in figure (7.34) right.

Again in this case the classification errors PU and PM was 0 because there was no overlap in the attributes. The numbers in the left side of the following figures represent the overlap percentage of the color and shape attributes, and the values on the right side indicate the PM and PU classification errors.

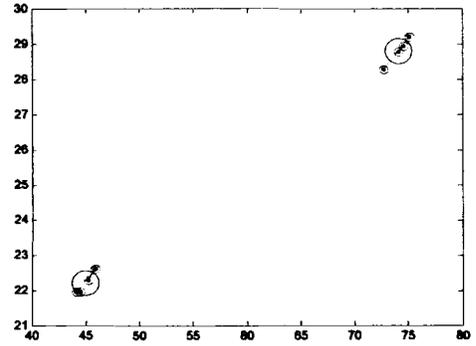
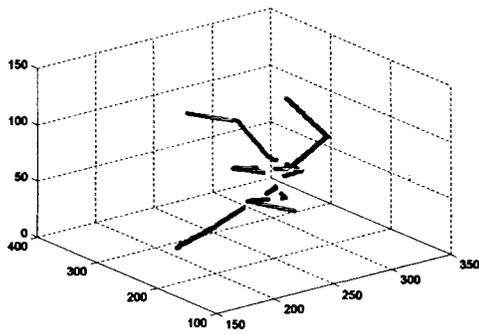


Figure 7.33: Branches and attribute clustering.

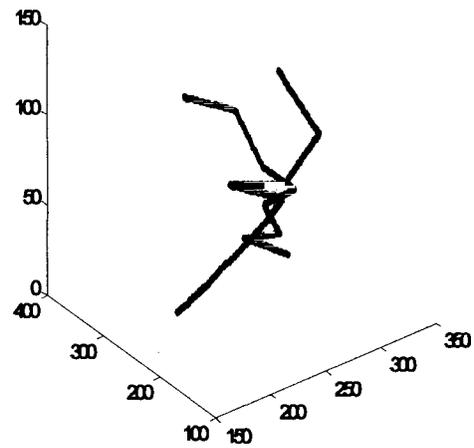
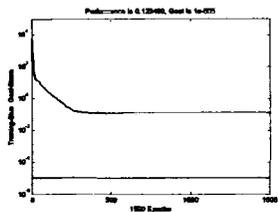
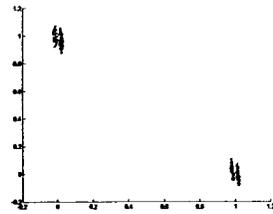
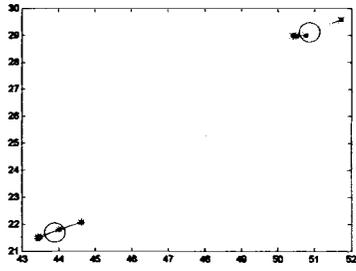
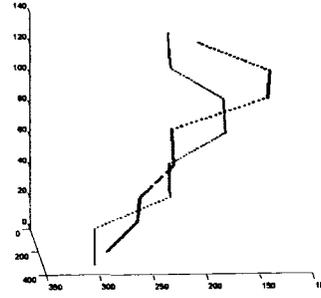


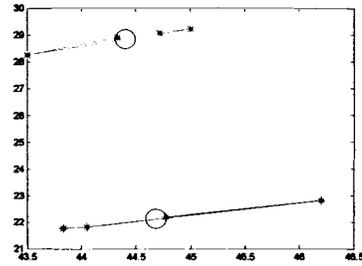
Figure 7.34: BP output and performance, and after BP classification.



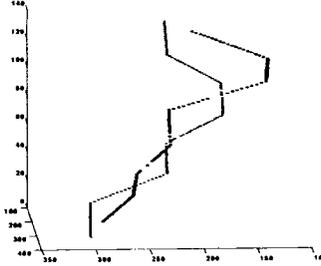
40,0



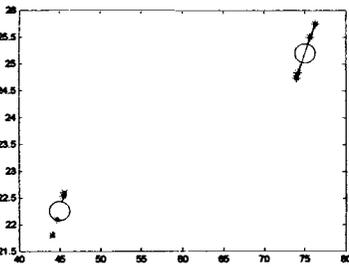
1,0



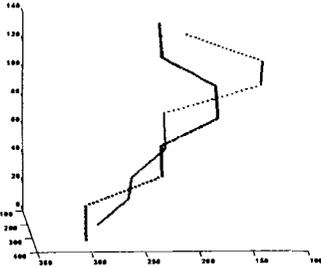
100,0



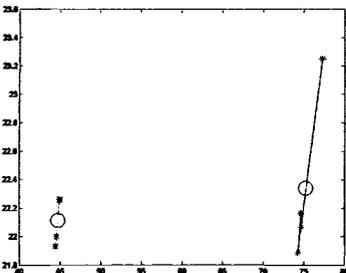
4,0



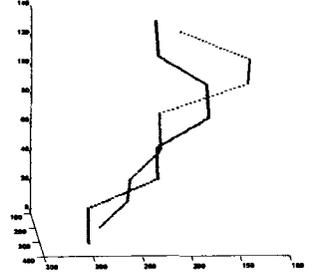
0,40



0,0

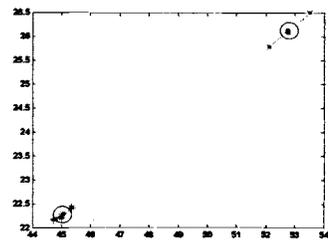


0,100

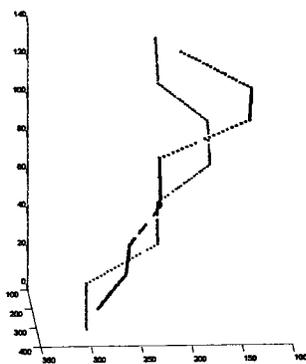


0,0

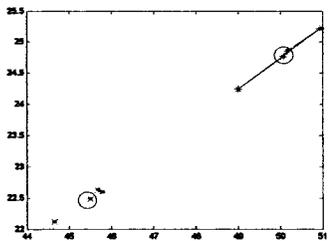
Figure 7.35: Classification when one attribute overlaps.



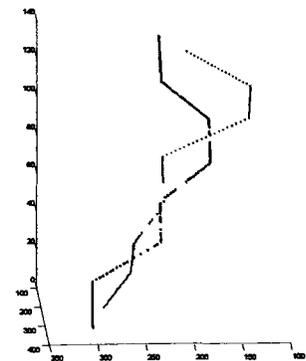
20, 20



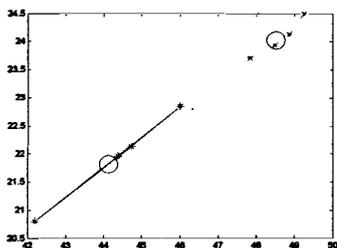
7, 2



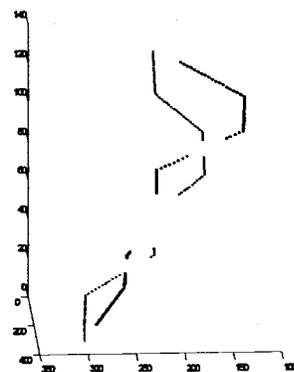
40, 40



15, 3



60, 60



25, 1 or 17, 4

Figure 7.36: Classification when both attributes overlap.

In the cases of (80,80) overlap, BP classification failed to give any decent results. The final classification outcome results purely from the branch clustering step. Figure (7.37). The (MU, MS) pair is (32, 5). In the case of (100,100) overlap the branches were misclassified and the error was very high.

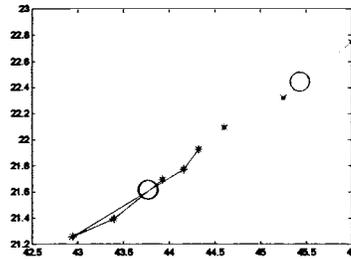


Figure 7.37: Clustering of branches in the 80, 80 overlap case.

For two trajectories that travel in the same road and are temporally close the ACCENT algorithm is not so efficient, since it is difficult to discern the differences even for a human operator. The following figures demonstrate this extreme situation for the input dataset of figure (7.38). The formed branches are shown in figure (7.38) right.

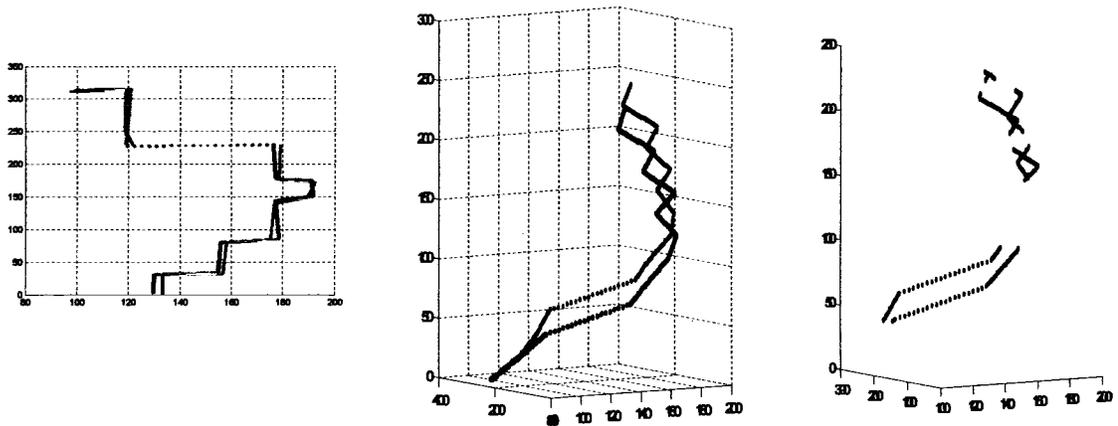


Figure 7.38: Input dataset of spatio-temporally close trajectories.

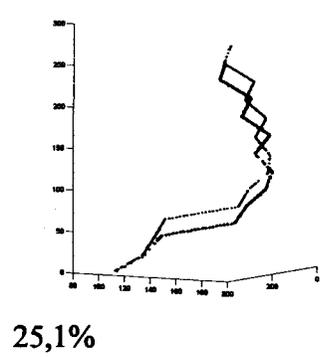
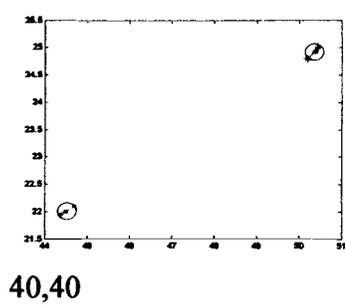
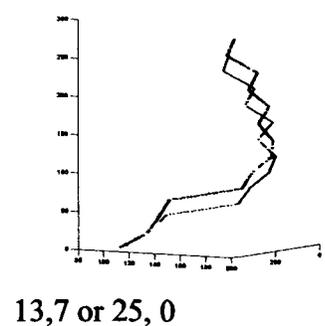
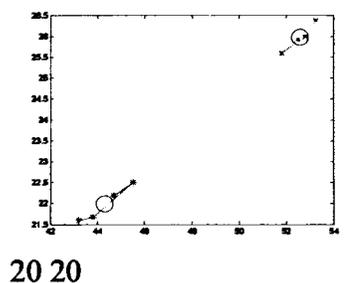
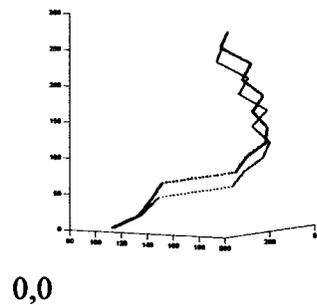
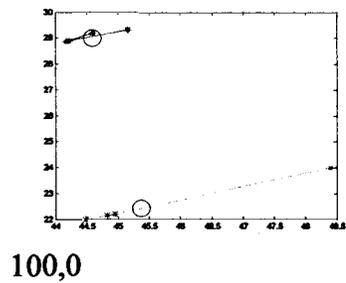
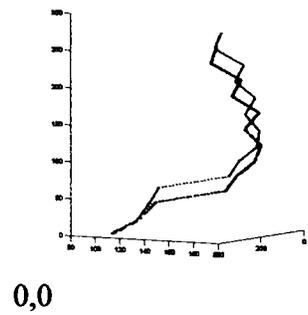
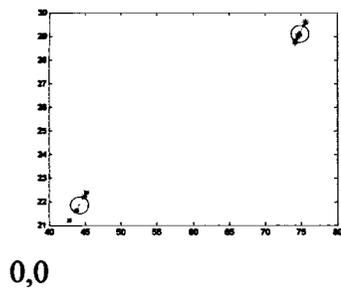


Figure 7.39: Classification of adjacent trajectories A.

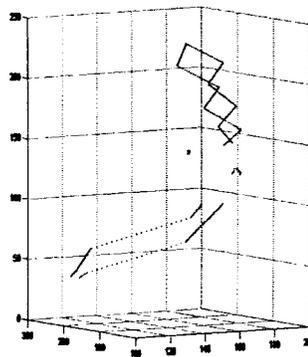
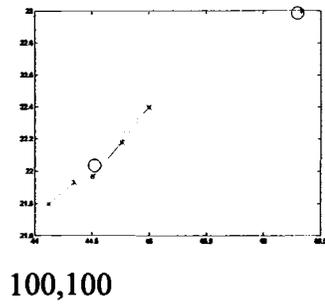
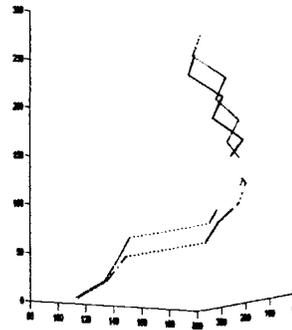
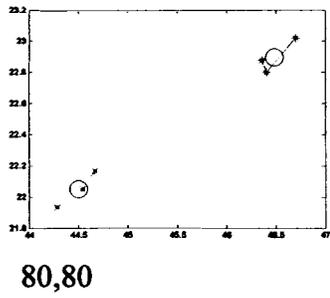
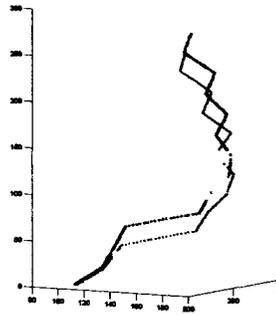
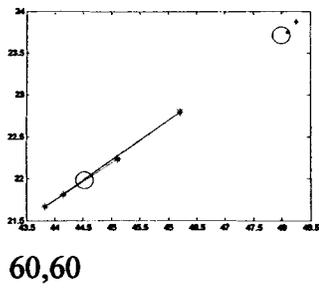


Figure 7.40: Classification of adjacent trajectories B.

7.5.2 Errors and Comparisons

After the branch selection and given that the points of the branches are successfully classified, there is a trade off between the percentages PM and PU. In general, if PU is large then there are not many points to be misclassified and thus PM remains lower. Accordingly, if PU is small then many of the remaining points get classified and PM can take larger values.

Finally, for the situations where we had a large PM or PU, we use the steps of SOM imposing, carrier formation and point classification and we get the classification percentages included in the table (7.6) for each of the two trajectory sets of figures 7.31 and 7.38.

	Set1 Trajectory 1 PU, PM	Set1 Trajectory 2 PU, PM	Set 2 Trajectory 1 PU, PM	Set 2 Trajectory 2 PU, PM
20,20	0,0.5	0,0	4,3	6,6
40,40	1,0	8,0	10,1	3,1
60,60	1,0	6,0	No improvement	No improvement
80,80	2,3	20,0	No improvement	No improvement

Table 7.6: Further classification.

In figure 7.41 we see some preprocessing through which some irrelevant data can be removed for further processing.

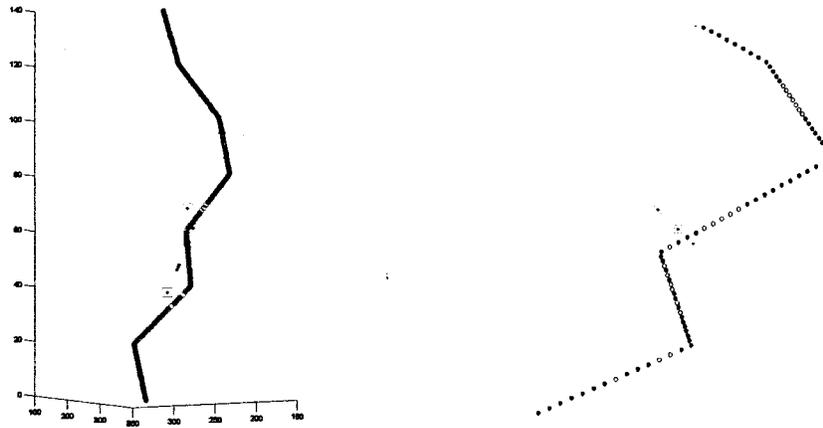


Figure 7.41: Removal of distant points.

In figure 7.42, (first set of trajectories) and 7.43, (second set of trajectories) we present the SOM algorithm in some special situations where SOM is able or unable to correctly classify the unclassified point data.

In the classification products that emerged from the BP algorithm and after imposing a SOM generalization algorithm we compare the resulting representation errors as discussed in section 7.3 with those emerging from the SOM algorithm on the original fully classified dataset. Thus, we can infer how much actual deterioration we may have on the summary formation. The results are given in table (7.7).

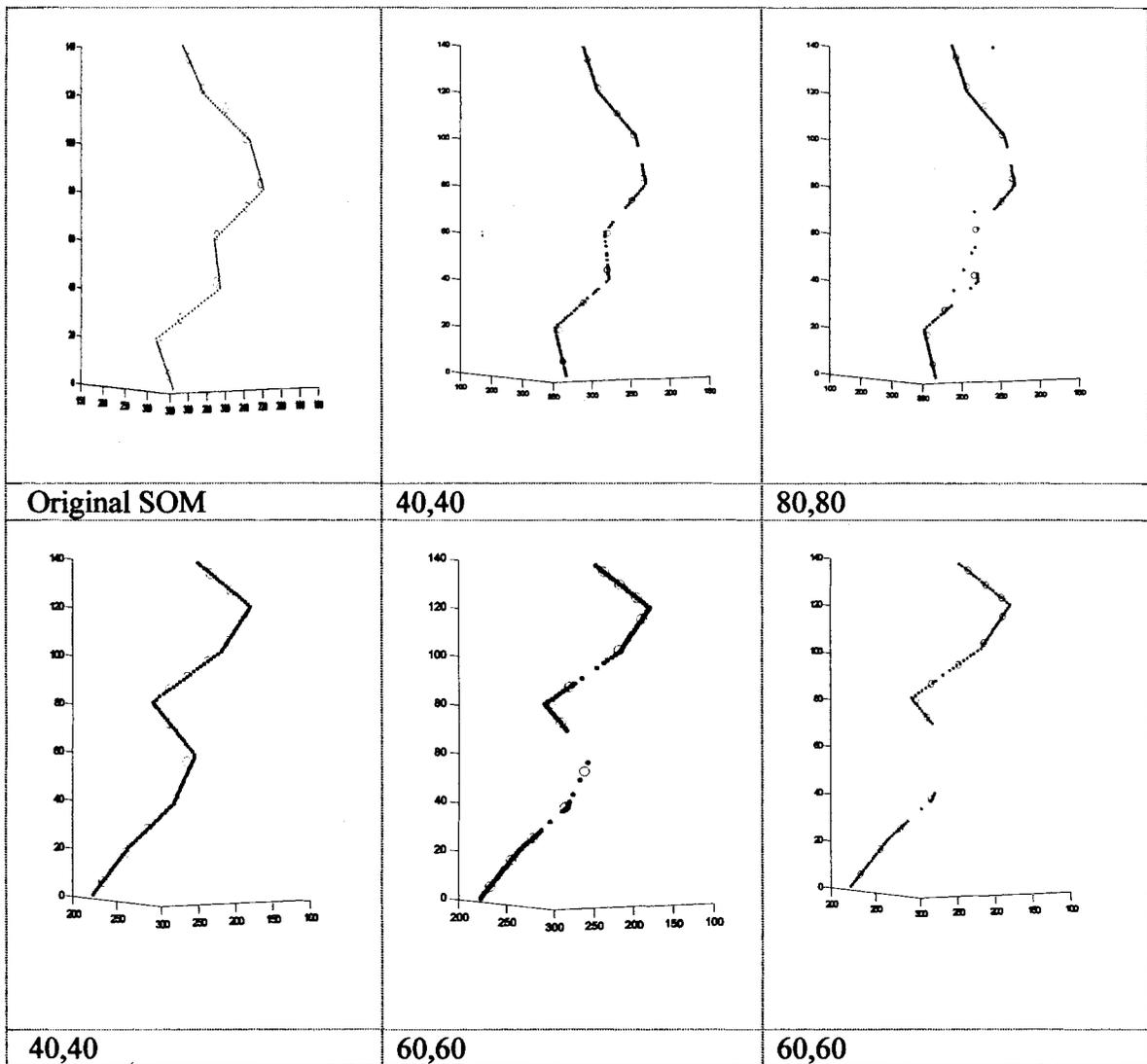


Figure 7.42: SOM further classification for first set of trajectories.

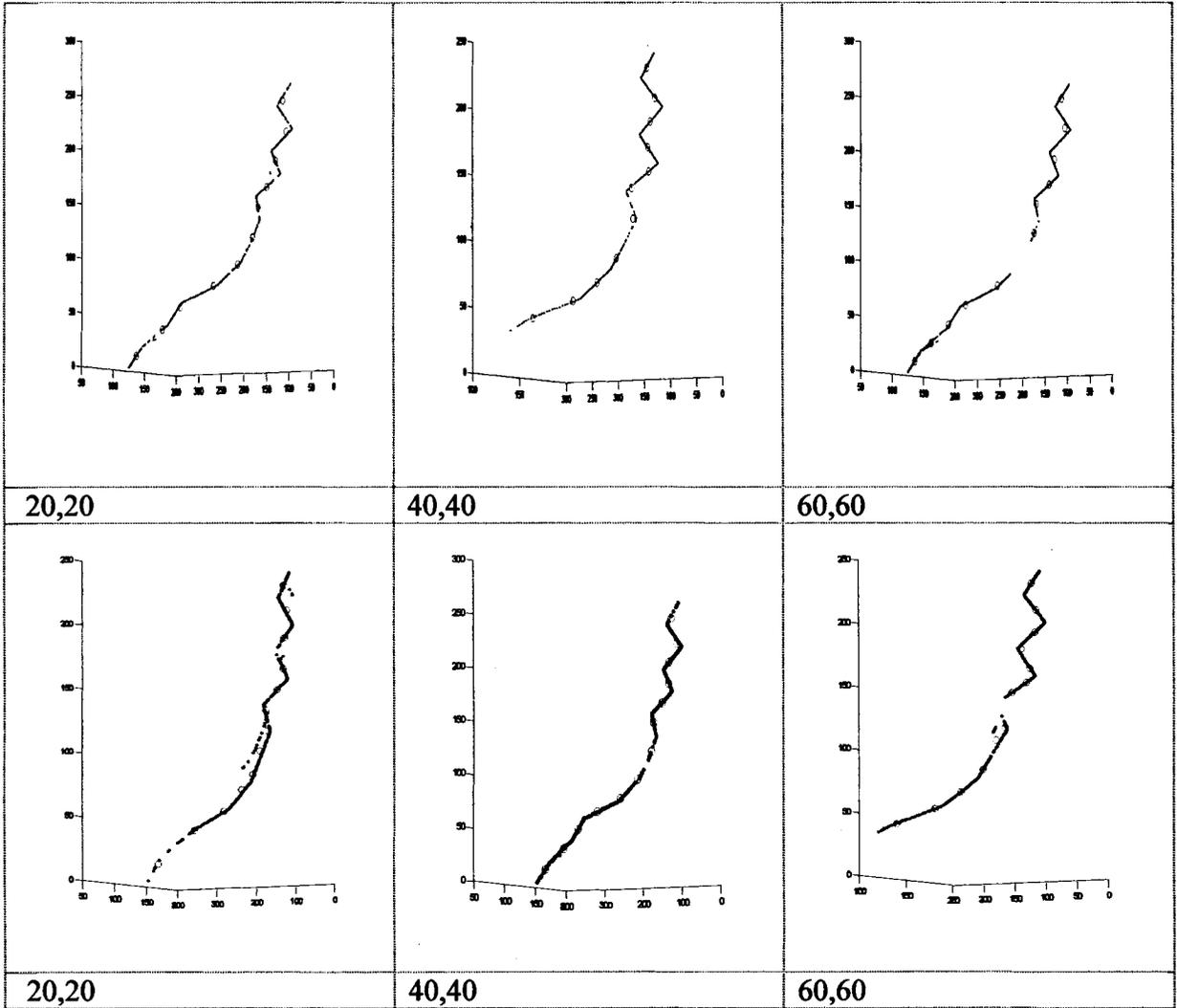


Figure 7.43: SOM further classification for second set of trajectories.

	Error RMS Set 1 Trajectory 1	Error RMS Set 1 Trajectory 2	Error RMS Set 2 Trajectory 1	Error RMS Set 2 Trajectory 2
SOM original	2.9	2.9	4.6	4.7
20,20	3.0	3.1	5.0	5.2
40,40	3.0	3.1	4.7	4.2
60,60	3.0	2.8	5.3	4.4
80,80	7.8	6.1	>>	>>

Table 7.7: Errors between SOM nodes and trajectories.

As it can be inferred the SOM errors in most cases do not deteriorate, since the generalization properties of SOM are able to bridge the occluded or misclassified data points through the connection of the nodes.

To compare our method with a common clustering classification that might be utilized to classify the trajectories we used c-means algorithms for both the 5-Dimensional input space and the 2-Dimensional attribute space. In the 5-D space we got very poor results. The 2-D c-means results for the dataset of figure (7.31) were much better and are presented through the following figures. In this case we have only PM classification errors.

In the (80, 80) and (100, 100) cases the misclassification is so large that we cannot discern any of the trajectories. Accordingly for the trajectory of figure 7.38 we get the following results shown in figure (7.45).

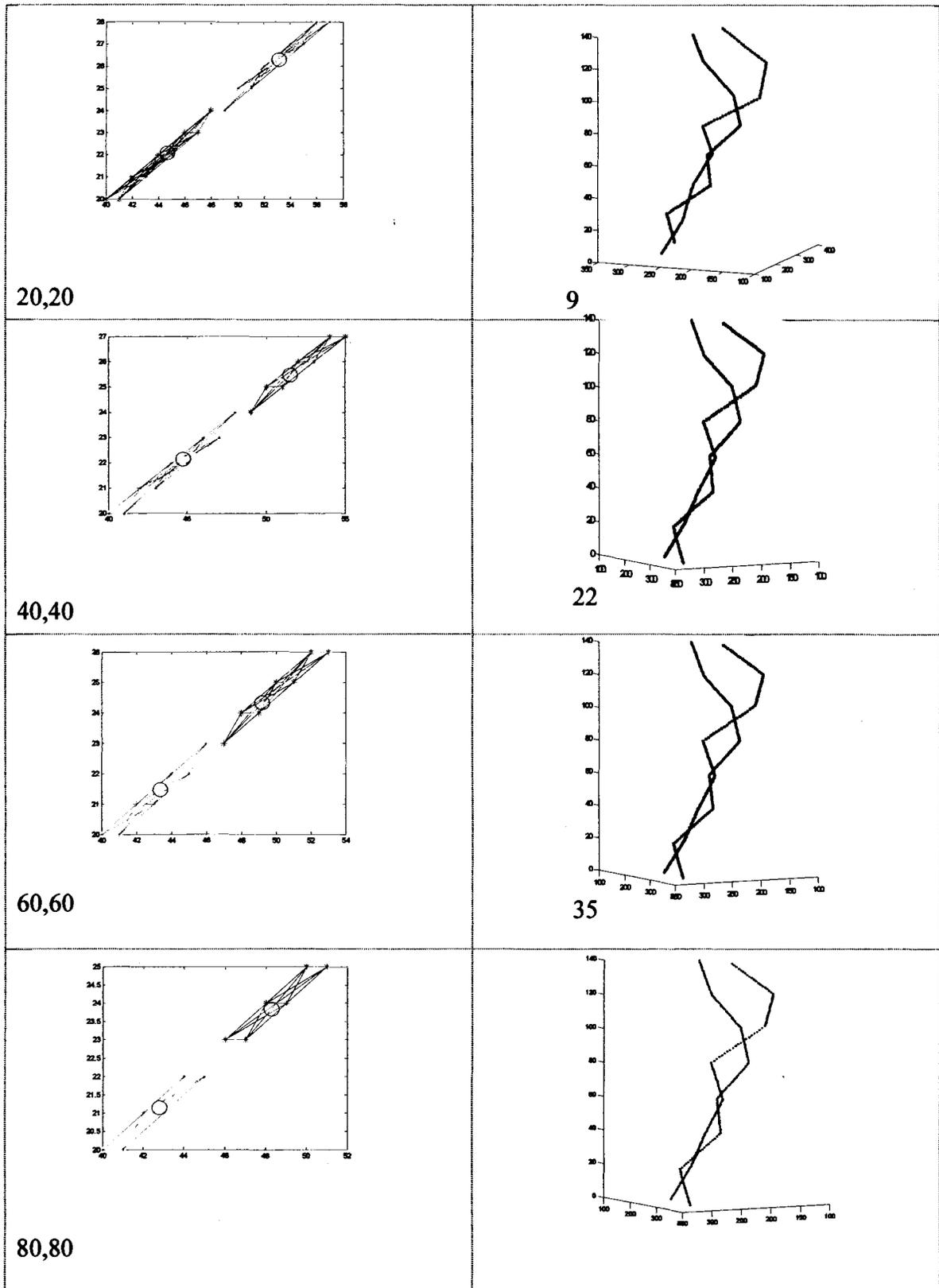


Figure 7.44: Classification through attribute space clustering.

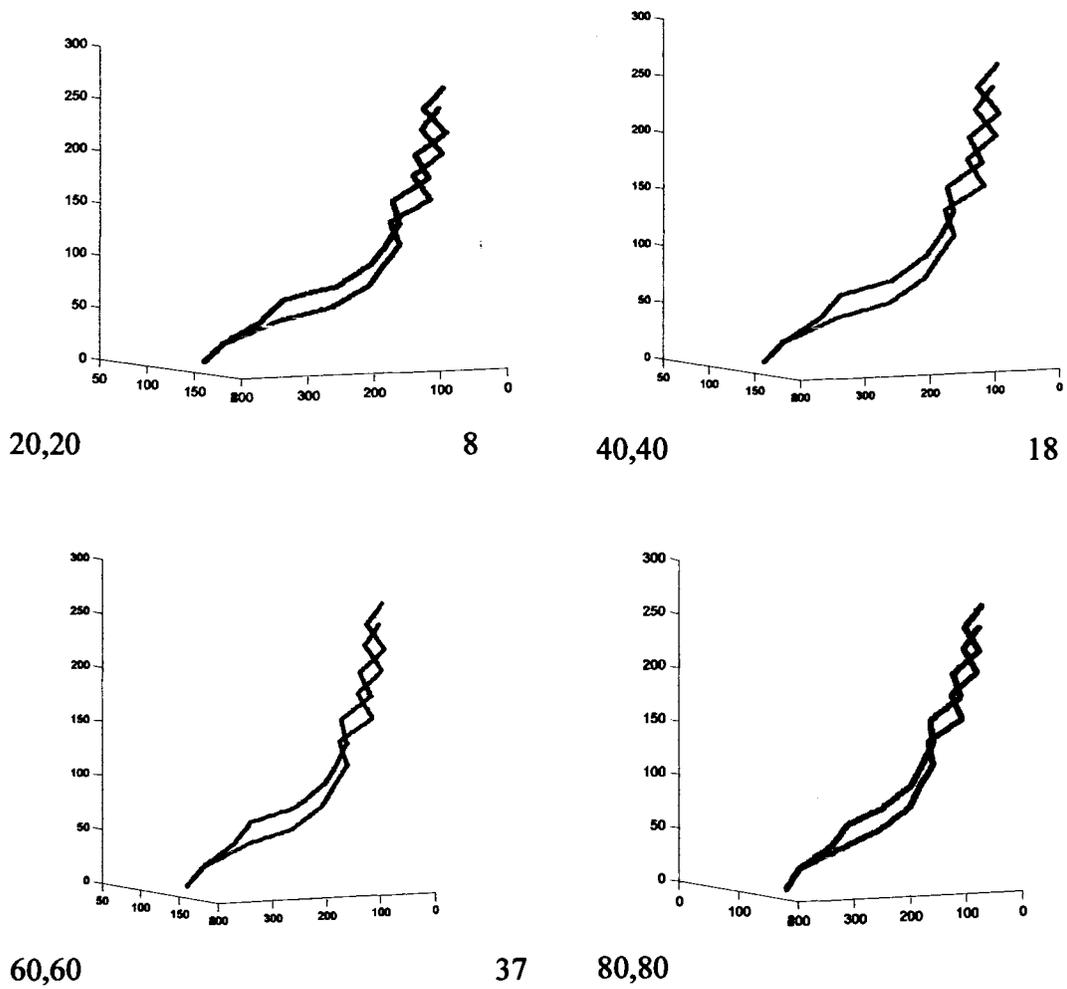


Figure 7.45: C-means classification for second dataset.

Again, for (80,80) and (100,100) there is no outcome that would allow any kind of reasonable classification. All the results presented so far for the trajectory classification are grouped in table 7.8.

As we can deduce from the results and figure 7.46, our analysis yields better results in most of the cases. Even after the BP analysis and before the further SOM implementation the results outperform other clustering techniques. When there is an overlap in the attributes between 40% and 60% the results are very satisfying.

Set 1	After BP (1-PU)	After BP (1-PM)	After SOM (1-PU)	After SOM (1-PM)	Pure cmeans (1-PM)
20,20	93	98	100	100	91
40,40	85	97	96	100	78
60,60	83	96	97	100	65
80,80	70	95	91	98	-
100,100	68	100	68	100	-
Set 2	After BP (1-PU)	After BP (1-PM)	After SOM (1-PU)	After SOM (1-PM)	Pure cmeans (1-PM)
20,20	86	93	95	96	92
40,40	75	99	93	99	82
60,60	83	93	83	93	63
80,80	75	93	75	93	-
100,100	70	100	70	100	-

Table 7.8: Trajectory classification results.

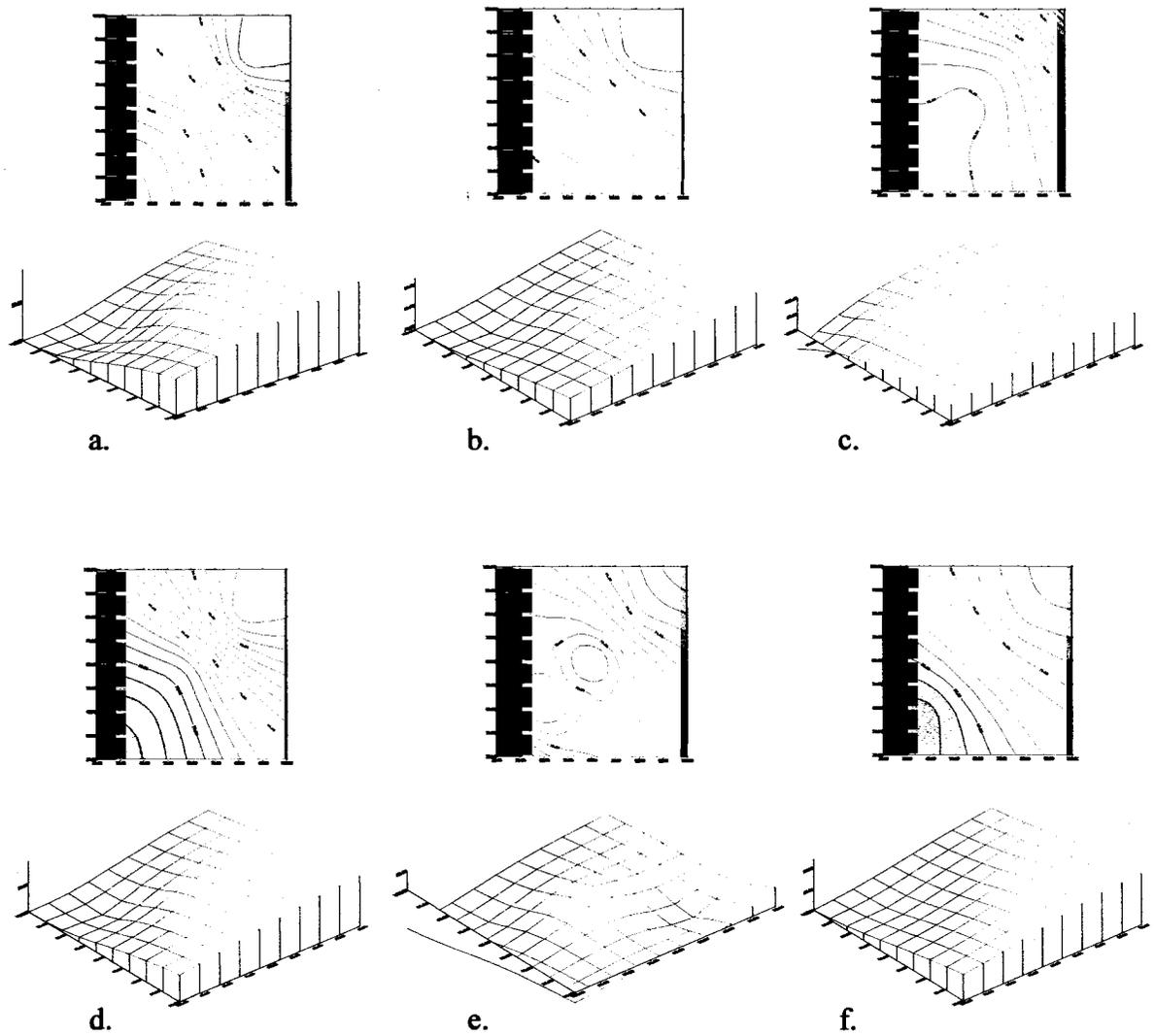


Figure 7.46: Surfaces of accuracy for different classification steps, first trajectory pair:

a)After c-means, b) After BP, c)After SOM, second trajectory pair: d)After c-means,

e)After BP, f) After SOM.

7.5.3 Three Trajectory Classification

In the case we have more than two trajectories, computations rise exponentially. Nevertheless, the ACCENT algorithm is capable to provide sufficient results compared to other clustering techniques. For a three trajectory application, the input dataset, the branch formation and the attribute space clustering are shown in figure 7.47.

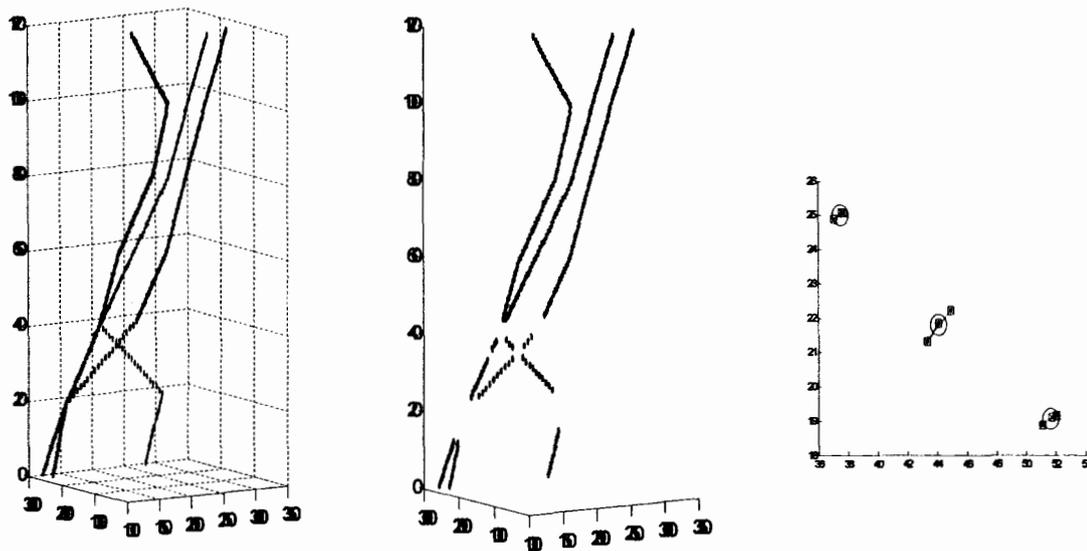
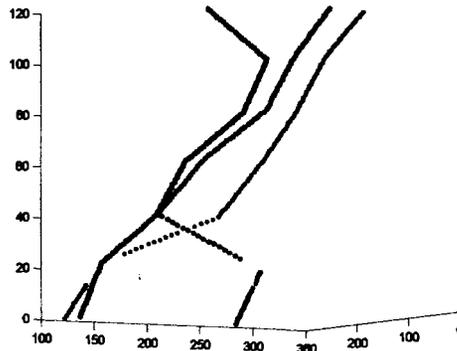


Figure 7.47: Input dataset, branch formation and clustering.

The result after the BP classification step is shown in figure 7.48. The attribute overlap between the three is (30, 40) for any pair of trajectories.



PU=5%, PM=1%

Figure 7.48: Resulted classification after the BP algorithm.

7.6 Phenomena Summarization

In figure 7.49 we present the description of the phenomenon evolving in the S-T domain and its corresponding spine slightly rotated and generalized by the SOM algorithm. This dataset forms the input space and is comprised of 70 frames. The cardinality segmentation for the phenomenon areas is shown in figure 7.50 as depicted in different temporal instances.

Generalization variations are demonstrated in the next set of figures. According to the selected generalization variables we acquire more or less detailed representation of the input dataset in both the spine and prongs descriptors. The blue nodes represent the SOM nodes while the stars denote the overall percentage based inner change, namely expansion (green stars) or shrinkage (red stars). Accordingly, the prong vectors either characterize local expansion (green vectors) or shrinkage (red vectors). The numbers in the parenthesis accompanying the figures describe the number of SOM nodes, the degree

of cardinal division (4, 8, 16, etc), the prong percentage of significance and the area change percentage of significance as described in chapter 6.

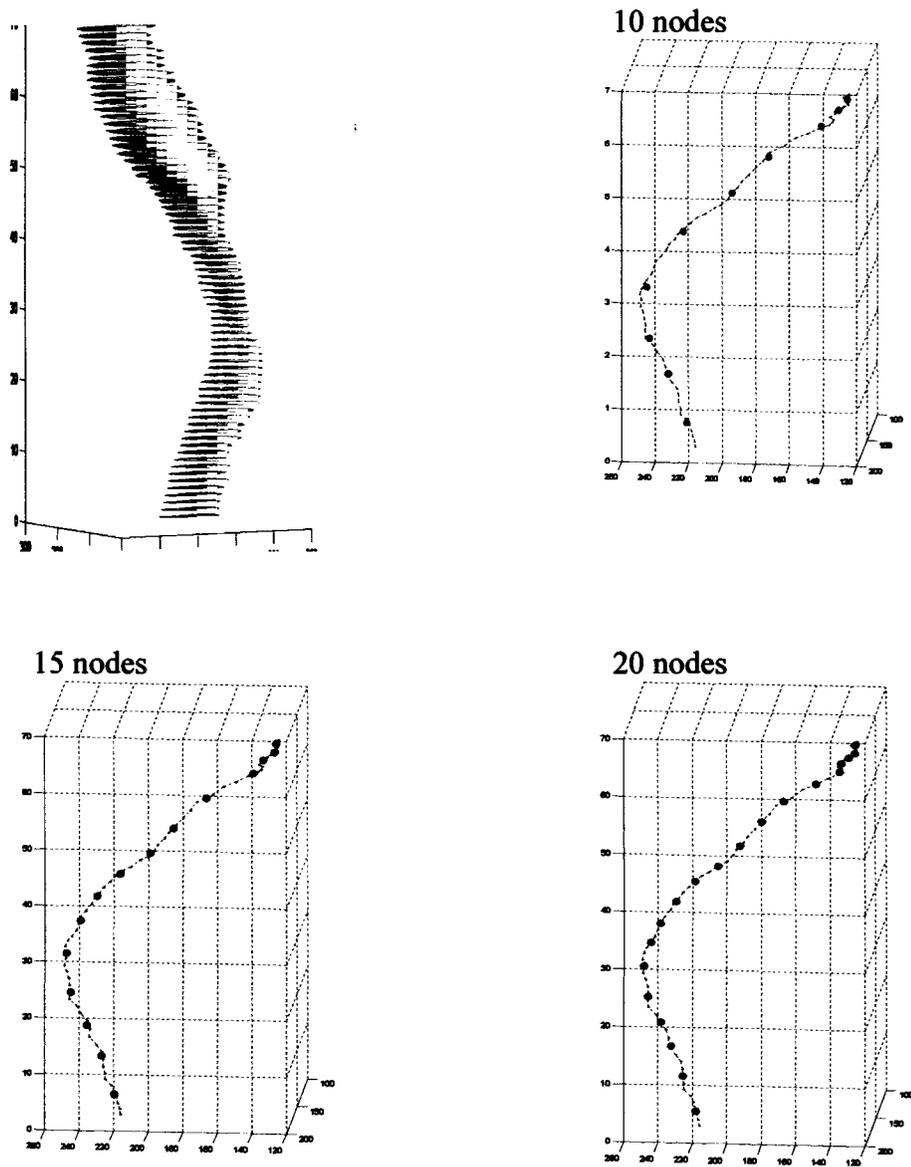


Figure 7.49: Input dataset and SOM generalization of the spine using variable number of nodes.

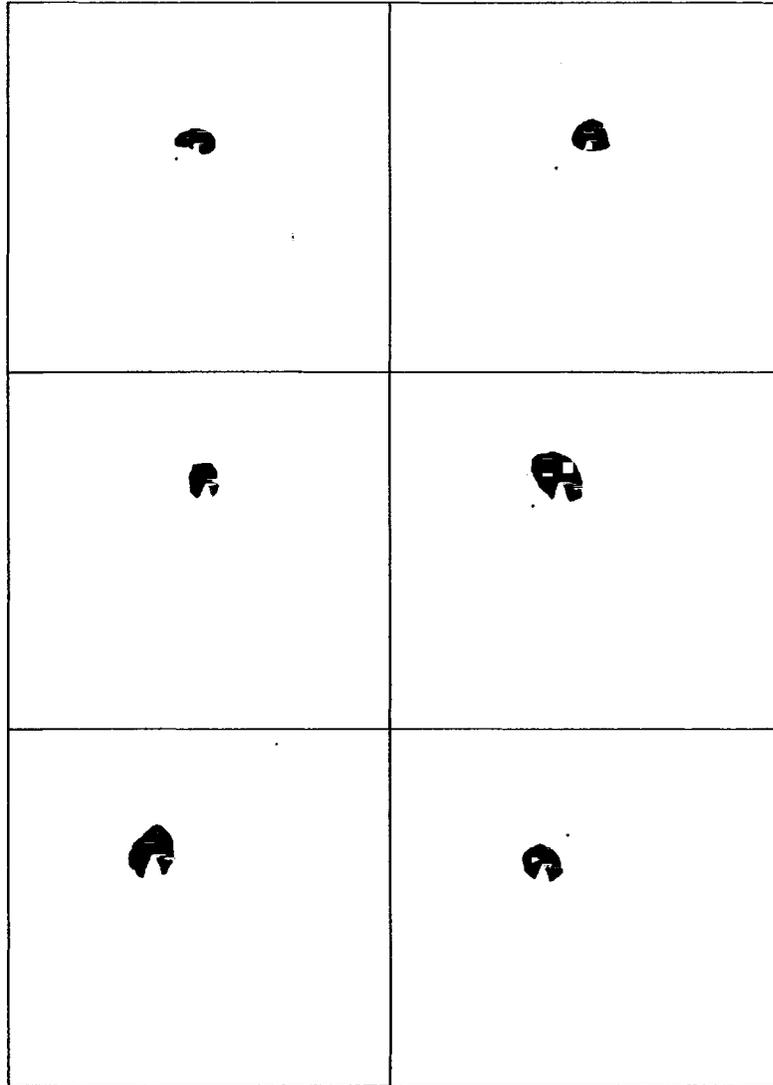
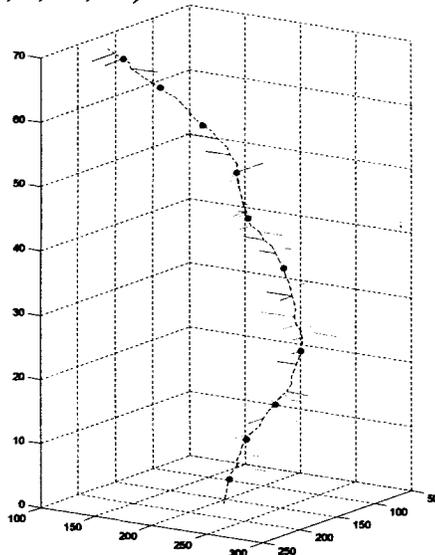
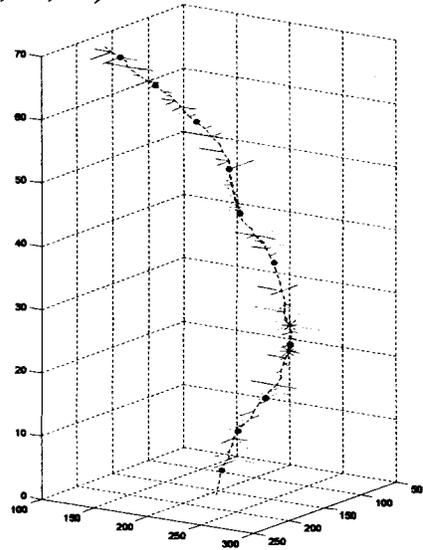


Figure 7.50: Phenomenon cardinal segmentation in different temporal instances.

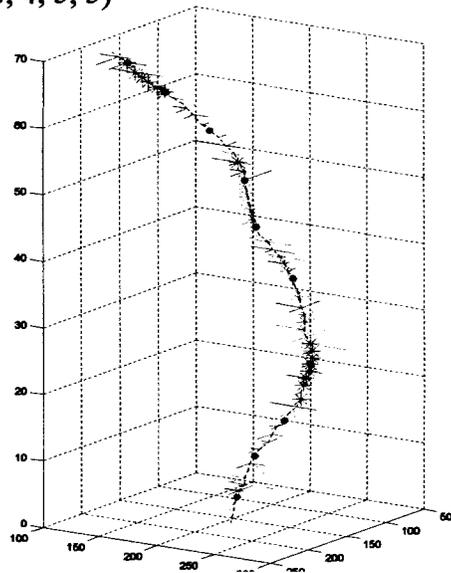
(10, 4, 20, 20)



(10, 4, 10, 10)



(10, 4, 5, 5)



(15, 4, 20, 20)

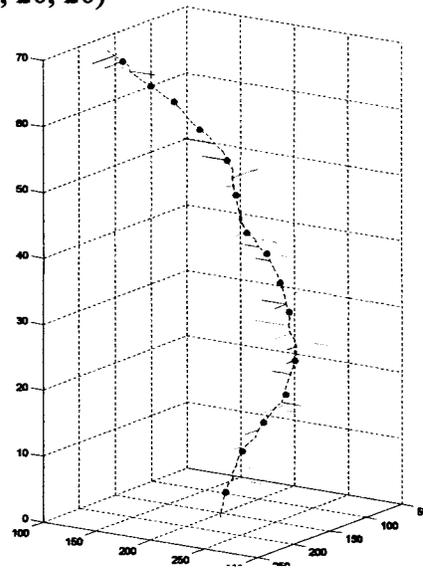
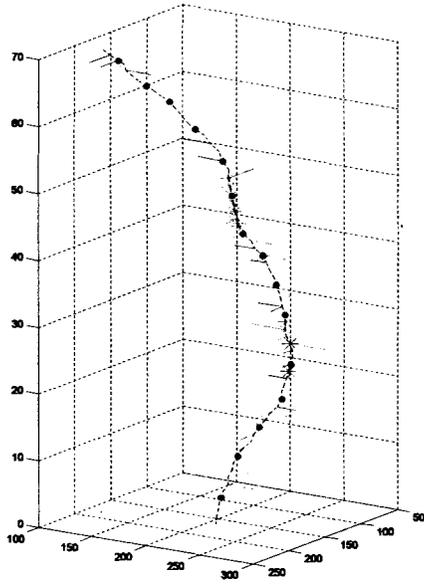
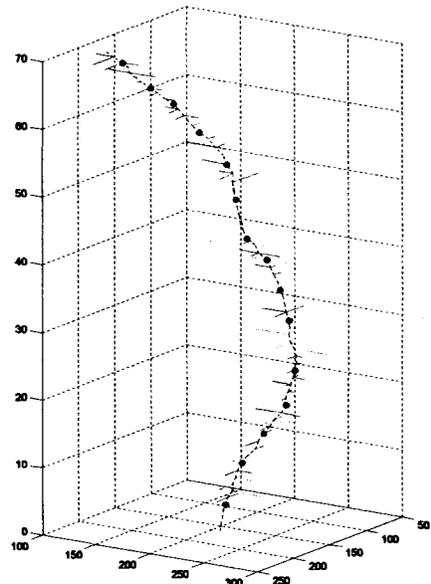


Figure 7.51: Generalization under different variables A.

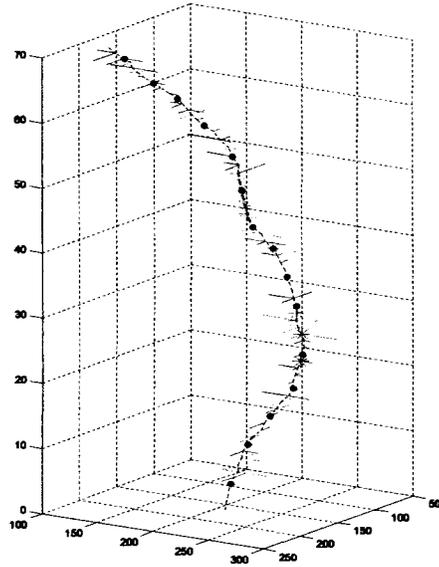
(15, 4, 20, 10)



(15, 4, 10, 20)



(15, 4, 10, 10)



(15, 4, 5, 5)

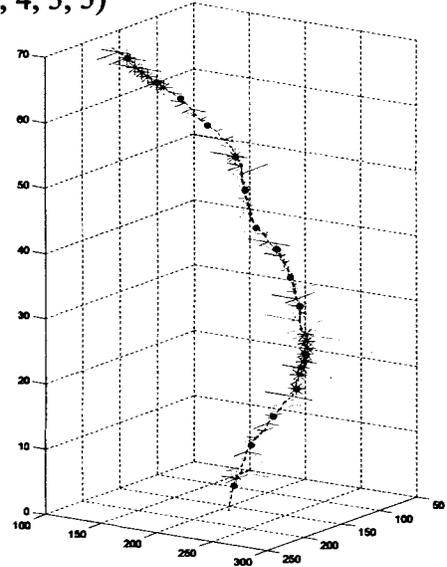
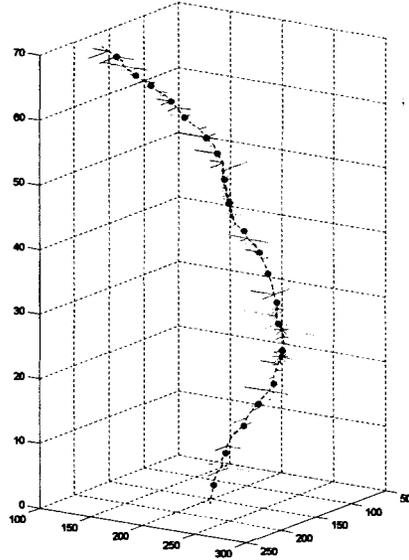
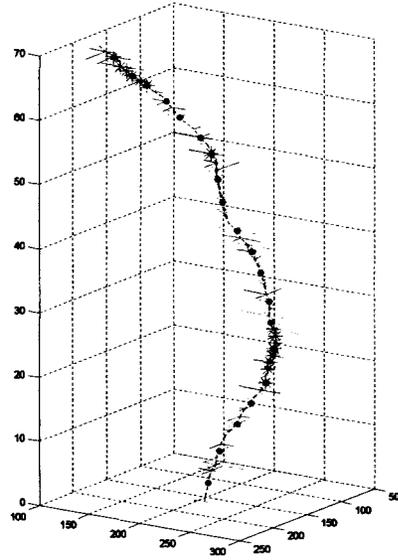


Figure 7.52: Generalization under different variables B.

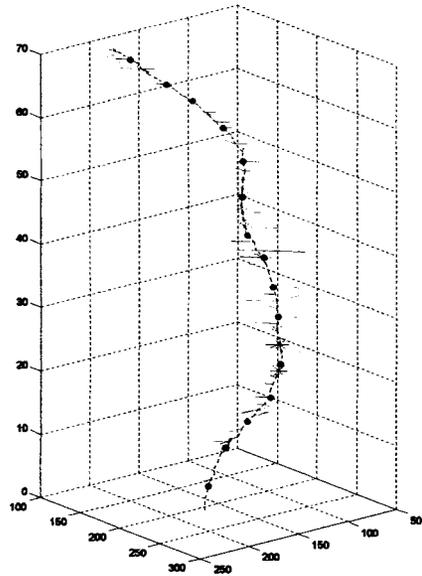
(20,4,10,10)



(20,4,5,5)



(15, 8, 10, 10)



(15, 8, 15, 15)

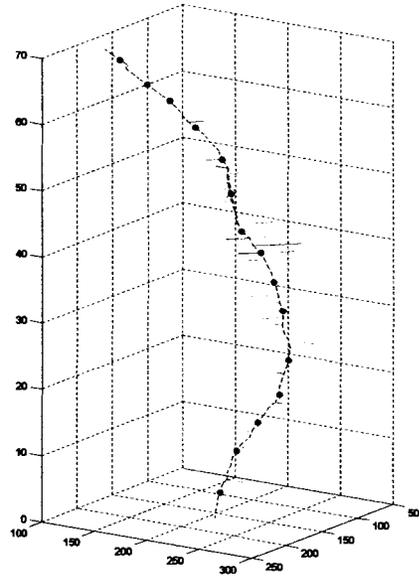
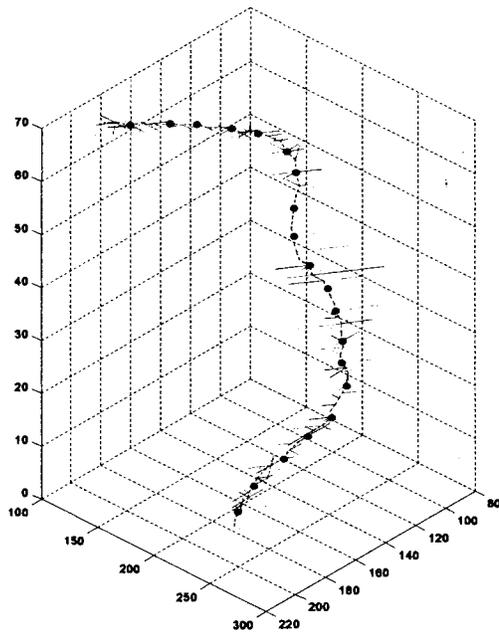
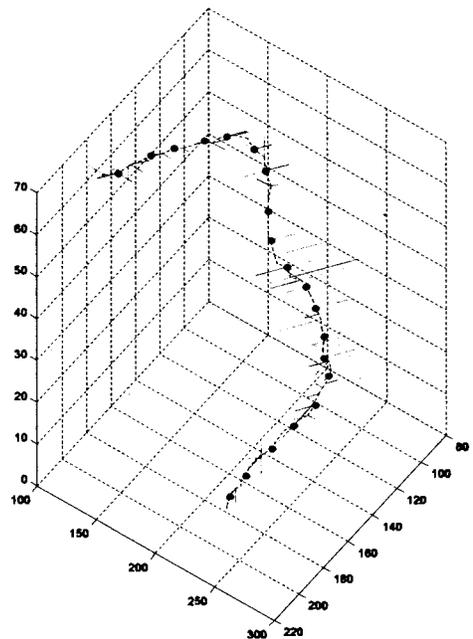


Figure 7.53: Generalization under different variables C.



20,8,5,30



20,8,10,30

Figure 7.54: Generalization under different variables D.

As seen in figures 7.51, 7.52, 7.53, and 7.54 the prongs and overall change nodes in larger percentage thresholds define a subset of the prongs and nodes of a smaller percentage threshold. On the other hand, this relation is not evident in the SOM nodes where competitive learning forces the nodes to change their placement to better describe the 3-D trajectory.

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

8.1 Conclusions

In this thesis we discussed the concept of summarization of motion imagery datasets based on the movement and deformation properties of objects included in the MI dataset. Data summaries are important for a number of reasons including compact dissemination of information, reduction of storage requirements, faster processing and browsing, and revealing of generalized tendencies. These advantages are particularly important for monitoring applications, where large datasets are produced, and most of the recorded trajectories follow typical patterns.

Throughout the theoretic and experimental findings we demonstrated that our generalization and classification techniques outperformed classic SOM and clustering techniques. In section 7.2 we showed that hybrid-SOM accurately describes spatio-temporal trajectories than typical SOM. In addition, our attribute-aided clustering technique proved much better than clustering techniques like SOM or k-means (section 7.5.2). Thus, we satisfied the formulated hypothesis. Additionally, there are several research issues addressed and tackled throughout this thesis. These contributions include:

The presented generalization framework is not based on a stable increment, but it is relevant to the information importance of the movement. By information importance we mean the volume of spatio-temporal variation of attributes, namely velocity and direction. As soon as the scale primitives are set, multiple summary versions of the original dataset are evident by altering the generalization attributes. Thus, one can travel in the significance scale and retrieve a summary with more or less detail. While one might be interested in the brief movement of each car in a scene, another might need to examine the traffic tendencies of a whole hour, day, or month.

Compared to a standard SOM process, our described hybrid-SOM trajectory analysis approach offers the advantages of invariance to the selection of the initial number of nodes and the additional SOM attributes. Since many local SOMs take place, the initial attribute contribution remains localized where it performs adequately. Self-organizing maps prove adequate in dealing with versatile environments including multiple dimensional data, corrupted and occluded data.

In order to move from a single trajectory to multiple trajectory generalization, we considered registration and grouping of trajectories. Thus, we introduced the spatio-temporal neighborhood unit, which defines the space and time under which sets of trajectories could be grouped together. This grouping didn't take place in the whole trajectory dataset, but only in the trajectories' SOM nodes for time efficiency purposes. By altering the S-T neighborhood unit's dimensions, scaled generalization is again supported.

Multi-dimensional classification of trajectories was another research task that was adequately tackled. Due to the elongated character of the data, other types of conventional clustering failed. Reduction of dimensionality competence suffered when the multiple attribute ranges included overlaps. Yet, the combined geometric and backpropagation network solution proved efficient in such classification applications.

Finally we moved in the modeling and summarization of two-dimensional phenomena evolving over time. We introduced the concept of S-T helixes as compact representations of spatio-temporal events. The helix model comprised of SOM movement nodes (spines) and cardinality shape-change descriptors (prongs).

The constructed summaries are not just a visualization product but they support further processing. While most of the research as seen in chapter 2, deals with the visualization part, our work spans in the realm of metadata structure formation. The constructed data can be stored in a database and can be queried. As seen in (Stefanidis et al., 2001) the formed data, support complex spatiotemporal analysis and are suitable for video queries.

The various SOM nodes can be grouped separately in a lifeline type data sequence and they may describe metrics like topology, distance etc. Furthermore, for each specific attribute we can have a hierarchical arrangement of nodes to describe the various levels of generalization. This arrangement of nodes within individual lifelines is shown in figure 8.1. They represent several attributes of an object's spatiotemporal progression and they can be exploited to develop metrics to evaluate the similarity of multiple movements on their entirety, or considering specific attributes only.

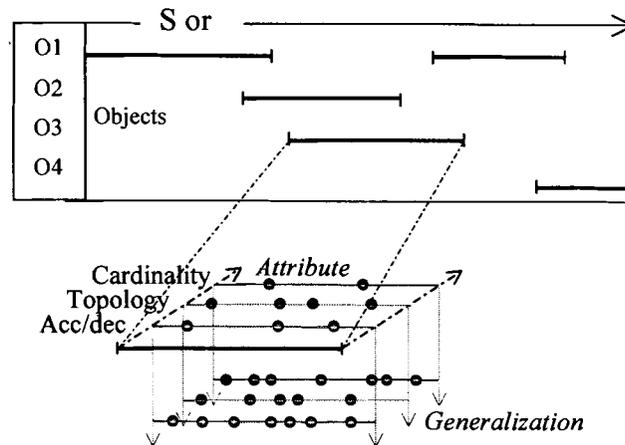


Figure 8.1: Lifeline representation of moving objects and attribute resolution.

8.2 Future Work

In figure 1.2 we introduced the focus of this thesis. The non-shaded areas outline a first indication on the research areas that complementary work would be beneficial. More specifically:

While we focus on the analysis of video datasets, the framework can be generalized to function on additional types of spatiotemporal dataset. It is often desirable to compare data summaries from different scenes describing events that take place in different areas at different time instances. In order to handle such comparisons we have to be able to provide a similarity framework to accommodate temporal, spatial and attribute registration and comparison.

In addition to the information that each spatio-temporal trajectory carries, more information is inherent in a dynamic scene. The constructed data are complemented by

additional instances that include user-defined criteria or application feedback rules. Through querying or rule asserting in the original dataset our summary can be enriched to convey additional information.

In the behavioral domain a lot of potential is evident since movement and interrelation of movements can be modeled to describe the behavior of objects. Expected behavior in this content is introduced in (Partsinevelos et al., 2000) where deviations from expected behaviors are identified. In addition, tracking and object extraction enhancement can take place by using spatio-temporal analysis parallel to the procedures discussed in this thesis.

Visualization is an important product of the summary formation. Further research with experimentation can be conducted to provide a concise and user friendly GUI to accommodate the applications discussed in the thesis. In the phenomena summarization 3-dimensional objects and their movement modeling is a potential research domain. Split incidents, multiple and complex surfaces should be included in the modeling and metadata formation processing in such a way that they could be adequately stored in a database.

BIBLIOGRAPHY

- Agarwal, P. K., L. Arge and J. Erickson, 2000. Indexing Moving Points. *Proceedings of the Nineteenth ACM Symposium on Principles of Database Systems*, Dallas, Texas: 175-186.
- Agrawal, R., C. Faloutsos and A. Swami, 1993. Efficient Similarity Search in Sequence Databases. *Proceedings of the Fourth International Conference on Foundations of Data Organization and Algorithms*, Chicago, *Lecture Notes in Computer Science 730*, Springer Verlag: 69-84.
- Aguierre-Smith, T.G. and G. Davenport, 1992. The Stratification System: A Design Environment for Random Access Video. *Third International Workshop on Network and Operating System Support for Digital Audio and Video*, San Diego, California: 250-261
- Allen, J. F. and G. Ferguson, 1994. Actions and Events in Interval Temporal Logic. *Journal of Logic and Computation*, 4(5): 531-579.
- Ardizzone, E. and M. Hacid, 1999. A Semantic Modeling Approach for Video Retrieval by Content. *Proceedings IEEE International Conference on Multimedia Computing and Systems*, 2: 158-162.

- Bradshaw, K., I. Reid and D. Murray, 1997. The Active Recovery of 3D Motion Trajectories and Their Use in Prediction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(3): 219-234.
- Brassel, K. and R. Weibel, 1988. A review and conceptual framework of automated map generalization, *International Journal of Geographical Information Systems*, 2(3): 229-244.
- Bremond, F. and G. Medioni, 1998. Scenario Recognition in Airborne Video Imagery. *IEEE Workshop on the Interpretation of Visual Motion*, Santa Barbara, California.
- Bruns, H. and M. Egenhofer, 1996. Similarity of Spatial Scenes. *Seventh International Symposium on Spatial Data Handling (SDH '96)*, Delft, The Netherlands M-J. Kraak and M. Molenaar (eds.), 4A: 31-42.
- Chang, S. F., W. Chen and H. Sundaram, 1998. Semantic Visual Templates - Linking Features to Semantics. *Proceedings of the Fifth IEEE International Conference on Image Processing*, Chicago, 3: 531-535.
- Chang, W., G. Sheikholeslami, J. Wang and A. Zhang, 1998. Data Resource Selection in Distributed Visual Information Systems. *IEEE Transactions on Knowledge and Data Engineering*, 10(6): 926-946.
- Chomicki, J. and P. Revesz, 1999. A Geometric Framework for Specifying Spatiotemporal Objects. *Proceedings of the Sixth International Workshop on Temporal Representation and Reasoning (TIME '99)*, Orlando, Florida, C. Dixon and M. Fisher (eds.): 41-46.
- Christel, M.G., M. A. Smith, C. R. Taylor and D. B. Winkler, 1998. Evolving Video Skims into Useful Multimedia Abstractions. *Proceedings of the Conference on*

Human Factors in Computing Systems (CHI '98), Los Angeles, California, C. Karat, A. Lund, J. Coutaz and J. Karat (eds.): 171-178.

Combi, C., 2000. Modeling Temporal Aspects of Visual and Textual Objects in Multimedia Databases. *Proceedings of the Seventh International Workshop on Temporal Representation and Reasoning*, (TIME '00), Cape Breton, Nova Scotia, Canada, S. Goodwin, A. Trudel (eds.): 59-68.

Doucette, P., P. Agouris, M. Musavi and A. Stefanidis, 1999. Automated Extraction of Linear Features from Aerial Imagery using Kohonen Learning and GIS. *Integrated Spatial Databases, Digital Images and GIS International Workshop*, (ISD '99), Portland, Maine, *Lecture Notes in Computer Science*, Springer Verlag, 1737: 20-33.

Doucette, P., P. Agouris, A. Stefanidis and M. Musavi, 2001. Self-Organized Clustering for Road Extraction in Classified Imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 55(5-6): 347-358.

Douglas, D. H. and T. K. Peucker, 1973. Algorithms for the reduction of the number of points required to represent a line or its caricature. *The Canadian Cartographer*, 10(2):112-122.

Egenhofer, M. and K. Al-Taha, 1992. Reasoning about Gradual Changes of Topological Relationships. *Theory and Methods of Spatio-Temporal Reasoning in Geographic Space*, Pisa, Italy A. Frank, I. Campari, and U. Formentini (eds.), *Lecture Notes in Computer Science*, Springer-Verlag: 196-219.

Forlizzi, L., R. H. Gutting, E. Nardelli, and M. Schneider, 2000. A Data Model and Data Structures for Moving Object Databases. *Proceedings of the ACM SIGMOD International Conference on Management of Data*, Dallas, Texas: 319-330.

- Gong, Y. H. and X. Liu, 2000. Video Summarization Using Singular Value Decomposition. *International Conference on Computer Vision and Pattern Recognition, (CVPR '2000)*: 2174-2180.
- Hacid, M., C. Declair and J. Kouloumdjian, 2000. A Database Approach for Modeling and Querying Video Data. *IEEE Transactions on Knowledge and Data Engineering*, 12(5): 729-750.
- Haykin, S., 1999. *Neural Networks*. Upper Saddle River, New Jersey, Prentice Hall.
- Hibino, S. and E. Rundensteiner, 1995. Interactive Visualizations for Exploration and Spatio-Temporal Analysis of Video Data. *Workshop on Intelligent Multimedia Information Retrieval, (IJCAI '95)*, Montreal, Canada.
- Hagerstrand, T., 1970. What about People in Regional Science? *Papers of the Regional Science Association*, 24: 7-21.
- Hornsby, K., 1999. Identity-Based Reasoning About Spatio-Temporal Change, *Ph.D. Thesis, Department of Spatial Information Science and Engineering*, University of Maine.
- Hornsby, K. and M. Egenhofer, 1999. Shifts in Detail Through Temporal Zooming. *Proceedings of the Tenth International Workshop on Database and Expert Systems Applications*, Florence, Italy, IEEE Computer Society, Los Alamitos, California, A. Tjoa, A. Cammelli, and R. Wagner (eds.): 487-491.
- Hornsby, K. and M. Egenhofer, 2002. Modeling Moving Objects over Multiple Granularities. *Annals of Mathematics and Artificial Intelligence*, (1-2): 177-194.

- Jones, C. and I.M. Abraham, 1986. Design Considerations for a Scale-Independent Cartographic Database, *Second International Symposium on Spatial Data Handling*, Duane Marble (ed), Seattle, Washington: 384-398.
- Keogh, E. J. and M. J. Pazzani, 1999. Scaling up Dynamic Time Warping to Massive Datasets. *Principles and Practice of Knowledge Discovery in Databases*, Prague, Czech Republic, J.M. Zytow and J. Rauch (eds.):1-11.
- Kohonen, T., 1982. Self-organized Formation of Topologically Correct Feature Maps. *Biological Cybernetics*: 59-69.
- Kohonen, T., 1997. *Self-Organizing Maps*. Springer-Verlag.
- Kollios, G., D. Gunopulos and V.J. Tsotras, 1999. On Indexing Mobile Objects. *Proceedings of the Eighteenth Symposium on Principles of Database Systems*, Philadelphia, Pennsylvania: 261-272.
- Kollios, G., V. Tsotras, D. Gunopulos, A. Delis and M. Hadjieleftheriou, 2001. Indexing Animated Objects Using Spatiotemporal Access Methods, *IEEE Transactions on Knowledge and Data Engineering*, 13(5): 758-777.
- Lee, S. L., S. J. Chun, D. H. Kim, J. H. Lee and C. W. Chung, 2000. Similarity Search for Multidimensional Data Sequences, *IEEE International Conference on Data Engineering*, San Diego, California: 599-608.
- Lienhart, R., 2000. Dynamic Video Summarization of Home Video, *Proceedings of Storage and Retrieval for Media Databases SPIE*, 3972, San Jose, California: 378-389.

- Mao, J. and K-K. Ma, 1999. Semantic Spatio-Temporal Segmentation for Extracting Video Objects. *IEEE International Conference on Multimedia Computing and Systems*, Florence, Italy, 1: 738-743.
- McMaster, R.W. and J. Comenetz, 1996. Procedure and Quality Assessment Measures for Cartographic Generalization, *GIS/LIS 1996 Proceedings*, Denver, Colorado: 775-785.
- Medioni, G., R. Nevatia and I. Cohen, 1998. Event Detection and Analysis from Video Streams. *DARPA Image Understanding Workshop*, Monterey, California: 63-72.
- Meng, J., Y. Juan and S. F. Chang, 1995. Scene Change Detection in a Mpeg Compressed Video Sequence. *SPIE Symposium on Electronic Imaging: Science and Technology- Digital Video Compression, Algorithms and Technologies*, San Jose, California, A. Rodriquez, R. Safranek and E. Delp (eds.), 2419: 14-25.
- Muller, J., R. Weibel, J. Lagrange and F. Salge, 1995. Generalization: State of the Art and Issues. *GIS and Generalization, Methodology and Practice*, Taylor & Francis, London, Great Britain, J.C. Muller, J.P. Lagrange and R. Weibel (eds.): 3-17.
- Nakos, B., 1997. Fractal geometry theory in performing automated map generalization operations, *Position Paper for the Second Workshop on Progress in Automated Map Generalization*, Gävle, Sweden.
- Oh, J. and K. A. Hua. 2000. An Efficient Technique for Summarizing Videos using Visual Contents. *Proceedings IEEE International Conference on Multimedia and Expo*, New York, NY: 1167-1170.

- Oomoto, E. and K. Tanaka, 1993. OVID: Design and Implementation of a Video-Object Database System. *IEEE Transactions on Knowledge and Data Engineering*, 5(4): 629-643.
- Papadias D., Y. Theodoridis, T. Sellis and M. Egenhofer, 1995. Topological Relations in the World of Minimum Bounding Rectangles: A Study with R-Trees. *SIGMOD '95*, San Jose, CA, M. Carey and D. Schneider (eds.), SIGMOD RECORD 24 (2): 92-103.
- Partsinevelos, P., P. Agouris and A. Stefanidis, 2000. Modeling Movement Relations in Dynamic Urban Scenes. *International Archives of Photogrammetry & Remote Sensing*, Amsterdam, Netherlands, 33: 818-825.
- Pfeiffer, S., R. Lienhart, S. Fischer and W. Effelsberg, 1996. Abstracting Digital Movies Automatically. *J. Visual Communication and Image Representation*, 7(4): 345-353.
- Pfoser, D. and Y. Theodoridis, 2000. Generating Semantics-Based Trajectories of Moving Objects. *International Workshop on Emerging Technologies for Geo-Based Applications*, Ascona, Switzerland: 59-76.
- Pfoser, D., C. Jensen and Y. Theodoridis, 2000. Novel Approaches in Query Processing for Moving Object Trajectories, *Proceedings of the Twenty-sixth International Conference on Very Large Databases*, Cairo, Egypt, Morgan Kaufmann: 395-406.
- Pope, A., R. Kumar, H. Sawhney and C. Wan, 1998. Video Abstraction: Summarizing Video Content for Retrieval and Visualization. *Proceedings Thirty-Second Asilomar Conference of Signals, Systems & Computers*: 915-919.

- Rui, Y., T.S. Huang and S. Mehrotra, 1998. Exploring Video Structure Beyond the Shots. *Proceedings IEEE Intl. Conference on Multimedia Computing and Systems*, Austin: 237-240.
- Russell, D. M., 2000. A Design Pattern-Based Video Summarization Technique: Moving from Low-Level Signals to High-Level Structure. *Proceedings of the Thirty-third Hawaii International Conference on System Sciences*, 3: 48.
- Sahouria, E. and A. Zakhor. 1997. Motion Indexing of Video. *International Conference on Image Processing*, 2: 526-529.
- Samet, H., 1990. *The Design and Analysis of Spatial Data Structures*. Addison-Wesley, Reading, MA.
- Sawhney, H. S. and S. Ayer, 1996. Compact Representations of Videos Through Dominant and Multiple Motion Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(18): 814-830.
- Sellis, T., N. Roussopoulos and C. Faloutsos, 1987. The R+-tree: A Dynamic Index for Multi-Dimensional Objects. *Proceedings of the Thirteenth International Conference on VLDB '87*, Brighton, England: 507-518.
- Sistla, A. P., O. Wolfson, S. Chamberlain and S. Dao, 1997. Modeling and Querying Moving Objects. *Proceedings of the Thirteenth International Conference on Data Engineering (ICDE'97)*, Birmingham U.K., IEEE Computer Society: 422-432.
- Smith, M. and T. Kanade, 1995. Video Skimming for Quick Browsing based on Audio and Image Characterization, *Technical Report CMU-CS-95-186*, Computer Science Department, Carnegie Mellon University.

- Stefanidis, A., P. Partsinevelos and P. Agouris, 2001. Using Lifelines for Spatiotemporal Summaries, *DG.O 2001 Proceedings, National Conference on Digital Government Research*, Los Angeles: 1-8.
- Stefanidis, A., P. Partsinevelos, K. Eickhorst and P. Agouris, 2001. Spatiotemporal Lifelines in Support of Video Queries, *Proceedings DEXA Workshop 2001, Query Processing and Multimedia Issues in Distributed Systems (QPMIDS)*, Munich: 865-869.
- Stefanidis, A., P. Agouris and P. Partsinevelos, 2002a. SpatioTemporal Helixes for Event Modeling, *DG.O 2002 Proceedings, National Conference on Digital Government Research*, Los Angeles: 219-224.
- Stefanidis, A., P. Agouris and P. Partsinevelos, 2002b. Summarizing the Content of Motion Imagery Datasets, *International Archives of Photogrammetry & Remote Sensing*, Corfu, 34 (5): 515-518.
- Stell, J.G. and M.F. Worboys, 1999. Generalizing Graphs using Amalgamation and Selection. *Proceedings of the Sixth International Symposium, (SSD '99), Hong Kong, China, Lecture Notes in Computer Science 1651*, R. Güting, D. Papadias and F. Lochovsky (eds.), Springer Verlag: 19-32.
- Tao Y. and D. Papadias: 2001. MV3R-Tree: A Spatio-Temporal Access Method for Timestamp and Interval Queries. *Proceedings of the Twenty-seventh International Conference on Very Large Databases*, Roma, Italy: 431-440.
- Taskiran, C. and E. J. Delp, 1998. Video Scene Change Detection Using the Generalized Trace. *Proceedings of IEEE International Conference on Acoustic, Speech and Signal Processing*, Seattle, Washington: 2961-2964.

- Theodoridis, Y., J. R. O. Silva and M. A. Nascimento, 1999. On the Generation of Spatiotemporal Datasets. *Proceedings of the Sixth International Symposium, (SSD '99), Hong Kong, China, Lecture Notes in Computer Science 1651*, R. Güting, D. Papadias and F. Lochovsky (eds.), Springer Verlag: 147.
- Tossebro, E. and R.H. Guting, 1999. Creating Representations for Continuously Moving Regions from Observations. *Proceedings of the Seventh International Symposium on Spatial and Temporal Databases*, Redondo Beach, California: 321-344.
- Uchihashi, S., J. Foote, A. Girgensohn and J. Boreczky, 1999. Video Manga: Generating Semantically Meaningful Video Summaries. *Proceedings of ACM International Conference on Multimedia*, Orlando, FL: 383-392.
- Vasconcelos, N. and A. Lippman, 1997. Towards semantically meaningful feature spaces for the characterization of video content. *International Conference on Image Processing*, Washington DC, 1: 25-29.
- Vasconcelos, N. and A. Lippman, 1998. A Spatiotemporal Motion Model for Video Summarization, *International Conference on Computer Vision and Pattern Recognition*, Santa Barbara, California: 361-366.
- Vazirgiannis, M. and O. Wolfson, 2001. A Spatiotemporal Model and Language for Moving Objects on Road Networks. *Proceedings of the Seventh International Symposium on Spatial and Temporal Databases*, Redondo Beach, California: 20-35.
- Wolfson, O., A. P. Sistla, B. Xu, J. Zhou and S. Chamberlain, 1999. DOMINO: Databases fOr MovINg Objects tracking. *Proceedings of the ACM SIGMOD*

Symposium on the Management of Data, Philadelphia, Pennsylvania, A. Delis, C. Faloutsos and S. Ghandeharizadeh: 547-549.

Wong, K. Y., M. Spetsakis and E. G. M. Petrakis, 1999. Motion Segmentation and Indexing for Video Database, *Vision Interface'99 (VI' 99)*, Quebec, Canada: 482-489.

Yeung, M. and Boon-Lock Yeo, 1997. Video Visualization for Compact Presentation and Fast Browsing of Pictorial Content. *IEEE Transactions on Circuits and Systems for Video Technology*, 7(5): 771-785.

Zhang, H.J., C. Y. Low and S. W. Smoliar, 1995. Video Parsing and Browsing using Compressed Data, *Multimedia Tools and Applications*, 1: 89-111.

Zhang, H. J., J. Wu, D. Zhong and S. W. Smoliar, 1997. An Integrated System for Content-Based Video Retrieval and Browsing, *Pattern Recognition*, 30(4): 643-658.

Zhong, D. and S.-F. Chang, 1997. Video Object Model and Segmentation for Content-Based Video Indexing, *IEEE International Symposium on Circuits and Systems*, Hong Kong, 2: 1492-1495.

BIOGRAPHY OF THE AUTHOR

Panayotis Partsinevelos was born in Athens, Greece (Hellas) on November 15, 1973. He graduated from the 1st Pefki High School in 1991. He attended the National Technical University of Athens and graduated in 1996 with high honors receiving a Diploma in Surveying Engineering. His interests were related to geodesy and photogrammetry and his thesis was titled: "Adjustment of mathematical surface models in real surfaces. Application in parabolic antennas." After receiving various graduate level scholarships he entered the Spatial Information Science and Engineering Ph.D. program at The University of Maine in the fall of 1997. He served both as a Graduate Research Assistant and Graduate Teaching Assistant and his research interests related to digital image and video processing and spatio-temporal data summarization. Throughout his studies he authored several refereed publications. He is a member of the American Society for Photogrammetry and Remote Sensing and the Technical Chamber of Greece.

After receiving his degree, Panayotis will return to Greece to join the army and then pursue a career preferably in academia. Panayotis is a candidate for the Doctor of Philosophy degree in Spatial Information Science and Engineering from The University of Maine in December 2002.