

HELSINKI UNIVERSITY OF TECHNOLOGY

Faculty of Electronics, Communications and Automation

Pekka Rossi

Life Cycle Analysis of Convective Cells through Image Processing and Data Fusion

Master's thesis for the degree of Master of Science in Technology
submitted for inspection in Espoo on the 18th of February 2009

Supervisor: Prof. Heikki Koivo

Instructor: Vesa Hasu, D.Sc (Tech)

HELSINKI UNIVERSITY OF TECHNOLOGY ABSTRACT OF THE MASTER'S THESIS

Author: Pekka Rossi

Title: Life Cycle Analysis of Convective Cells through Image Processing and Data Fusion

Date: the 18th of February 2009

Number of pages: 100

Faculty: Faculty of Electronics, Communications and Automation

Professorship: T-61 Computer and Information Science

Supervisor: Prof. Heikki Koivo

Instructor: Vesa Hasu, D.Sc (Tech.)

Today numerical weather models can predict large scale weather phenomena with a reasonable accuracy. Still, these models are too coarse for small scale rapidly changing weather phenomena, such as thunderstorms. Therefore, doing short term local forecasts, i.e. nowcasting, is a challenging task for the contemporary weather forecasting. State-of-the-art remote sensing instruments and computer vision techniques are the key to this challenging task.

This thesis discusses nowcasting of thunderstorms, i.e. convective cells, through different computer vision techniques that are applied to spatially and temporally accurate weather radar and lightning data. Emphasis is on object-oriented convective cell tracking, which is widely accepted as an important concept regarding the nowcasting of convective cells.

Conventionally, the nowcasting of convective cells is performed through weather radar data. In this thesis, we propose a novel cell tracking method, which fuses both weather radar and important lightning information. The aim of the data fusion is to consolidate the tracking, as more information is incorporated in the procedure. The functioning of the algorithm is tested with several case studies.

The proposed tracking algorithm provides an important tool for several applications. Primarily, convective cell tracking is applied to monitoring and predicting movement of hazardous thunderstorms. It can also be used for analyzing cell properties and life cycle. Therefore, this thesis examines also convective cell properties and derives descriptive statistic of the convective cell by means of the proposed tracking algorithm. The results are based on tests, which are carried out through an extensive case material provided by the Finnish Meteorological Institute.

The thesis elaborates also on the lightning properties of the convective cell. The information extracted by the tracking algorithm is applied to analyze the relationship between lightning and different radar parameters within the cell. In addition, probabilistic reasoning is applied to determine possible lightning hazard of individual cells.

Finally, this thesis proposes a new fuzzy logics model for analyzing cell life cycle phases. The model provides an automated method, which mimics expert made reasoning and infer whether the cell is intensifying or dissipating.

Keywords: meteorology, nowcasting, thunderstorm, weather radar, convective cell, computer vision, digital image processing, data fusion, object tracking, life cycle analysis, fuzzy logics, probabilistic reasoning, expert system

Tekijä: Pekka Rossi

Työn nimi: Konvektiosolujen elinkaarianalyysi kuvankäsittelyn ja datafuusion avulla

Päivämäärä: 18.2.2009

Sivumäärä: 100

Tiedekunta: Elektroniikan, tietoliikenteen ja automaation tiedekunta

Professori: T-61 Informaatiotekniikka

Työn valvoja: Prof. Heikki Koivo

Työn ohjaaja: TkT Vesa Hasu

Nykyaikana numeeriset säänennustusmallit pystyvät ennustamaan suuren skaalan sääilmiöitä merkittävällä tarkkuudella. Nämä mallit ovat kuitenkin liian karkeita pienen skaalan sääilmiöille, kuten paikallisille ukkosmyrskyille. Paikallinen, lyhyen ajan säänennustaminen eli lähietkiennustaminen on haastava meteorologinen ongelma. Tämä vaatii erityisesti ajallisesti ja paikallisesti tarkkojen modernien kaukokartoitusinstrumenttien sekä tietokonenäköön perustuvien menetelmien soveltamista.

Tämä diplomityö käsittelee paikallisten ukkosmyrskyjen eli konvektiosolujen lähietkiennustamista. Erityisesti tarkastelemme oliopohjaista konvektiosolun jäljitystä, joka on yleisesti käytetty lähestymistapa ukkoson lähietkiennustamisessa.

Perinteisesti konvektiosolun lähietkiennustamiseen sovelletaan säätutkadataa. Työ esittelee uuden konvektiosolujen jäljitysmenetelmän, joka hyödyntää sekä säätutka- että salamainformaatiota. Koska salamadata antaa tärkeää lisäinformaatiota ukkosmyrskyjen paikasta ja liikkeestä, uusi datafuusiopohjainen menetelmä parantaa algoritmin toimintavarmuutta. Työssä testataan algoritmin toimintaa useiden esimerkkitaustojen avulla.

Suunniteltu jäljitysmenetelmä tarjoaa tärkeän apuvälineen moniin käytännön tarkoituksiin. Ensisijainen sovelluskohde on vaarallisten konvektiosolujen monitorointi sekä liikkeen ennustaminen. Lisäksi menetelmää voidaan soveltaa ukkoson elinkaaren ja ominaisuuksien analysointiin. Tämä työ tarkastelee ukkosolun tilastollisia ominaisuuksia uuden jäljitysmenetelmän avulla. Menetelmällä tuotettua informaatiota sovelletaan myös solun salamoinnin ja erilaisten tutkaparametrien välisen yhteyden analysointiin. Lisäksi työssä suunnitellaan probabilistiseen päättelyyn perustuva malli, jonka avulla voidaan tarkastella yksittäisen konvektiosolun salamariskiä.

Työssä suunnitellaan myös uusi sumeaan logiikkaan perustuva automaattinen asiantuntijamalli, jonka avulla voidaan antaa informaatiota konvektiosolun elinvaiheista. Mallin päätehtävä on analysoida asiantuntijan tavoin konvektiosolun voimistumista tai heikkenemistä.

Avainsanat: meteorologia, lähietkiennustaminen, säätutka, datafuusio, ukkonen, konvektiosolu, tietokonenäkö, digitaalinen kuvankäsittely, oliopohjainen jäljitys, elinkaarianalyysi, sumea logiikka, asiantuntijamalli, probabilistinen päättely

Preface

This thesis was carried out within PiPo-project in collaboration with the Finnish Meteorological Institute, Helsinki University of Technology Department of Automation and System Technology and Vaisala Oyj.

First and foremost, I express gratitude to my instructor Vesa Hasu for invaluable guidance during the thesis work – his advices, patience and optimism has been an enormous help to get through this challenging but also rewarding task. Additionally, I wish to thank my supervisor Prof. Heikki Koivo for the encouraging feedback he has given throughout the project.

For an excellent collaboration, I would like to address my thanks to the staff at the Finnish Meteorological Institute. Particularly, I am grateful to Elena Saltikoff and Antti Mäkelä for meteorological guidance and discussions. In addition, I thank Markus Peura and Jarmo Koistinen for interesting comments related to computer vision problems in weather radar meteorology. Moreover, I thank Harri Hohti and the Finnish Meteorological Institute for providing the data for the thesis.

Finally, I acknowledge the whole Control Engineering Group for excellent research facilities as well as for creating a supportive and cozy working atmosphere.

Pekka Rossi,

February 2009

Abbreviations

AN	Auto-Nowcast system
CAPPI	Constant Altitude Plan Position Indicator
CC	Cloud to Cloud lightning
CG	Cloud to Ground lightning
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
FMI	the Finnish Meteorological institute
FMI LLS	Finnish Meteorological Institute Lightning Location System
GDBSCAN	Generalized Density-Based Spatial Clustering of Applications with Noise
GPS	Global Positioning System
IIR	Infinite Impulse Response
IMPACT	IMProved Accuracy by Combined Technology
LTI	Linear and Time Invariant
MCS	Mesoscale Convective system
MHT	Multiple Hypothesis Tracker
NSSL	National Severe Storm Laboratory
NWP	Numerical Weather Prediction
PCA	Principal Component Analysis
pdf	probability density function
PPI	Plan position Indicator
PRF	Pulse Repetition Frequency
Radar	RAdio Detecting And Ranging device
SAFIR	Surveillance et Alerte Foudre par Interférométrie Radioélectrique
SCIT	Storm Cell Identification and Tracking

SPROG	Spectral PROGnosis
TITAN	Thunderstorm Identification, Tracking, Analysis, and Nowcasting
TREC	Tracking Radar Echoes by Correlation
UTC	Coordinated Universal Time
VHF	Very High Frequency

Nomenclature

Operators

\ominus	morphological dilation
\oplus	morphological erosion
\bullet	morphological closing

Symbols

A_e	effective area of a radar antenna
A_i	area of i th polygon
a_i	coefficient of the finite impulse response part of a LTI filter
a_{ij}	element of assignment matrix
A_p^k	assignment matrix of track p in frame k
b_i	coefficient of the infinite impulse response part of a LTI filter
c	speed of light
\hat{c}_d^i	fuzzy logic model output at the i th time step
$\hat{\hat{c}}_d^i$	filtered fuzzy logic model output at the i th time step
C_i	cluster i
c_{ij}	combinatorial correspondence cost between i th and j th object
c_{ik}	path coherence cost of track i in frame k
D	diameter of backscattering object

D_a	distance between antennas
$d(\mathbf{x}, \mathbf{y})$	metric norm between objects \mathbf{x} and \mathbf{y}
d_∞	max norm
d_p	p -norm
d_A	distance between two polygons
dBZ	basic unit of logarithmic radar reflectivity factor
Δt	time interval
f_t	temporal partial derivate of reflectivity pattern
f_u	horizontal partial derivate of reflectivity pattern
f_v	vertical partial derivate of reflectivity pattern
G	radar gain
g_{ij}	decision boundary between i th and j th class
g_i	discrimination function of i th class
h	spatial extent of radar pulse
H_{EW}	East-West oriented magnetic field
H_{NS}	North-South oriented magnetic field
J	Lucas-Kanade optical flow cost function
K	magnitude of complex dielectricity
l_{T_p}	track length
\mathbf{m}_i	centroid of cluster C_i
$minCard$	minimum weight in the N_{pred} -neighborhood of a core object
$N_\varepsilon(p)$	ε -neighborhood of point p
n_f	number of consecutive frames used in the clustering
N_{pred}	radius N_{pred} -neighborhood

$N_{NPred}(o)$	N_{pred} -neighborhood of object o
O_p^k	objects of p th track in the k th frame
o_i	object i
o_{im}	object i in frame m
$P(\omega_i)$	prior probability of class ω_i
$P(\omega_i \mathbf{x})$	posterior probability of class ω_i
$p(\omega_i \mathbf{x})$	likelihood function of class ω_i
P_e	error probability
Pr	received power at radar antenna
PRF	Pulse Repetition Frequency
P_t	radar transmission power
P_σ	power radiated back from a backscattering object
r	range
r_{\max}	radar pulse length
$r_{A_{\max}}$	cell area change ratio
R_i	decision region of i th class
S_{inc}	non-isotropic power density at range r
S_{iso}	isotropic power density at range r
t	time
T	altitude threshold of $EchoTop_{i,j,T}$
T_p^k	track p in the frame k
t_j	estimated time of arrival at station i
t_{mj}	measured magnetic direction bearing by station i
T_r	reversal temperature

$u_A(x)$	fuzzy membership function of set A
$u_{A \rightarrow B}(\mathbf{x}, y)$	membership function of fuzzy implication
\mathbf{v}	velocity vector
V_c	contributing volume
$wCard(o)$	weighted cardinality of object o
V_{EW}	voltage induced in the East-West oriented loop
V_{NS}	voltage induced in the North-South oriented loop
$V_\varepsilon(\mathbf{x})$	a circle of radius ε centered on point \mathbf{x}
$\tilde{\mathbf{v}}_k^i$	estimated cell velocity of track k in frame i
$\hat{\mathbf{v}}_k^i$	measured cell velocity of track k in frame i
v_{\max}	maximum velocity in the reflectivity pattern
w_i	weight coefficient
\mathbf{x}	feature vector
X	universe in fuzzy logic, dataset in clustering
y_k	output of LTI filtering
z	radar reflectivity factor
Z	logarithmic radar reflectivity factor
α	phase difference
γ	forgetting factor
ε	radius of ε -neighborhood
λ	wavelength
μ	minimum number of points in the ε -neighborhood of a core object
θ_i	estimated bearing of magnetic field direction at i th station
θ_m	measured bearing of magnetic field direction

θ_{mi}	measured bearing of magnetic field direction at i th station
σ	backscattering cross section area
$\sigma_{az_i}^2$	expected azimuthal error of magnetic direction at i th station
σ_{ij}^2	expected error of time of arrival at j th station
Σ_i	covariance matrix of point data
τ	radar pulse length
χ^2	cost function to be minimized in the magnetic direction estimation
$\Psi(A)$	morphological operation on set A
Ω	window size
ω_i	class i

Contents

Preface	iii
Abbreviations	iv
Nomenclature	v
Contents	x
Chapter 1: Introduction	1
1.1 Background	1
1.1.1 From numerical weather forecasting models to nowcasting	1
1.1.2 Convective cell	1
1.1.3 The market niche of nowcasting	2
1.1.4 Remote sensing – the key to nowcasting	3
1.1.5 Computer vision based forecasting – a new approach	4
1.1.6 Lightning and weather radar data fusion	4
1.2 Objectives and scope of the thesis	5
1.2.1 Convective cell tracking	5
1.2.2 Life cycle analysis	5
1.3 Organization of the thesis	5
Chapter 2: Basic concepts and fundamentals	6
2.1 Convection in the atmosphere	6
2.1.1 Cumulus stage	6
2.1.2 Mature stage	7
2.1.3 Dissipating stage	8
2.2 Storm electrification and lightning	8
2.2.1 Graupel-ice-mechanism	9
2.2.2 Inductive electrification	9
2.3 Multicellular storms	10
Chapter 3: Data and instruments	12
3.1 Weather radar	12
3.1.1 Pulsed weather radar characteristics	12
3.2 Radar equations	13
3.2.1 Point target radar equation	14
3.2.2 Radar equation for distributed targets	15
3.2.3 Weather radar equation	15

3.3	Weather radar data representation and visualizing	17
3.3.1	Plan position indicator (PPI)	17
3.3.2	Constant altitude plan position indicator (CAPPI)	17
3.3.3	EchoTop	18
3.3.4	Anomalies and error sources	18
3.4	Locating lightning	18
3.4.1	Magnetic direction finding	19
3.4.2	Time of arrival technique	20
3.4.3	Interferometry based technique	21
3.4.4	Applied lightning data	22
Chapter 4:	Related work	23
4.1	Computer vision	23
4.2	Tracking as a computer vision problem	23
4.2.1	Object detection	24
4.2.2	Object representation	25
4.2.3	Point target correspondence tracking	26
4.3	Computer vision based methods in nowcasting	29
4.3.1	Grid-based motion estimation methods	29
4.3.2	Tracking methods in nowcasting	30
4.4	Life cycle analysis and cell temporal development	32
Chapter 5:	Applied methodologies	34
5.1	Data Clustering	34
5.1.1	Definitions and basic concepts of clustering	34
5.1.2	Density-based clustering	37
5.2	Convective cell identification	42
5.2.1	Reflectivity cell identification	42
5.2.2	Morphological preprocessing	42
5.2.3	Reflectivity cell representation with polygons	46
5.2.4	Flash cell identification	47
5.3	Convective cell tracking algorithm	48
5.3.1	Definition and basic concepts	49
5.3.2	Tracking through the density-based clustering	49
5.3.3	Improving the tracking algorithm with displacement velocity	50

5.3.4	Velocity initiation	51
5.3.5	Dealing with splitting and merging	52
5.3.6	Utilizing lightning data in the tracking	53
5.4	Fuzzy logic modeling of the cell development	54
5.4.1	Basic concepts and definitions	55
5.4.2	Fuzzy inference	56
5.4.3	Fuzzification	58
5.4.4	Defuzzification	58
5.4.5	Designed fuzzy model for the life cycle analysis of convective cells	59
Chapter 6:	Convective cell tracking and life cycle analysis	62
6.1	Visual performance of the tracking	62
6.1.1	Early tests with the tracking algorithm – a case study on Aug 27 th 2006	62
6.1.2	Understanding the parameters used in the clustering	63
6.1.3	The meaning of number of consecutive frames included in the clustering – a case study on Aug 26 th 2007	65
6.1.4	Importance of displacement – a case study on Aug 9 th 2005	66
6.1.5	Considerations on data quality – a case study on Aug 9 th 2005	67
6.2	Time series of convective cell tracks	68
6.2.1	Attaching information to the cells	68
6.2.2	Preprocessing of the time series	69
6.2.3	Cell analysis through time series plots	70
6.3	Descriptive statistics of convective cells	72
6.3.1	Storm duration	72
6.3.2	Typical life cycle time series	74
6.3.3	Cell development and lightning	76
6.3.4	Relationship between lightning and radar parameters	78
6.4	Fuzzy logic modeling	83
Chapter 7:	Conclusions	87
7.1	Future improvements	88
Appendix A:	Fuzzy logic model	90
Appendix B:	Bayesian classification for two multinormal distributions	94
References		96

Chapter 1: Introduction

1.1 Background

1.1.1 From numerical weather forecasting models to nowcasting

Efficient computers and numerical models have revolutionized modern weather forecasting. Today highly complex numerical weather models can predict weather several days ahead with a reasonable accuracy. These numerical weather prediction (NWP) models perform well if we want to forecast the large scale general conditions within a time frame of 12hrs. Especially weather phenomena exceeding spatial resolution of 100 km, such as weather fronts, are predicted well by these numerical models (Ruosteenoja 1996). As an example, meteorologists are able to describe general weather conditions of the next day reasonably well by the models and their expertise.

Despite the fact that these conventional weather forecasting models do relatively well, they are too coarse for phenomena having size less than 100 km. This is due to a deficient knowledge of small scale weather phenomena and their complex and chaotic nature; only a little error in the initial conditions will lead to a totally erroneous prediction. Besides, the number of meteorological observations is insufficient, which implies that the observation network offering the initial conditions for these models is too sparse. Thus, rapidly changing local weather phenomena such as thunderstorms, local shower, sea breeze and squall lines are unpredictable through the large scale numerical models.

In many cases it would be highly valuable to predict weather just a few hours or even minutes ahead. The need for accurate short term weather forecasting products is growing all the time. *Nowcasting* aims particularly at the prediction of rapidly changing meteorological quantities, such as heavy rain, gusts and lightning with spatial accuracy of only a few kilometers and a short time scale varying from a couple of minutes to hours. Thus, objective of nowcasting differs from ordinary weather forecasting, as the conventional meteorological models deal with spatial scale exceeding hundreds of kilometers and a time frame of days.

1.1.2 Convective cell

In this thesis the main objective is the nowcasting of development and life cycle of *convective cells*, i.e. local thunderstorms. Prediction of a convective cell is a classical nowcasting application due to its short lifetime and unpredictable nature. Because it has a spatial extension of a few kilometers, it is impossible to predict convective cells by means of conventional NWP models.

Local thunderstorms are known to occur everywhere in the world. They are more common at low latitudes, but for example in Finland they are frequent during the summer time. Physically the convective cell is an atmospheric updraft and it can be accompanied by heavy rain, lightning, gusty wind or even hail. Its body is mostly vertical and may

exceed an altitude of 10 km. Byers and Braham (1949) define a local thunderstorm as follows:

The thunderstorm represents a violent and spectacular form of atmospheric convection. In its method of development it appears to be a cumulus cloud gone wild. Lightning and thunder, usually gusty surface winds, heavy rain and occasionally hail accompany it. These phenomena are indicative of violent motions and complex physical processes going on within the cloud.

In radar usage the local thunderstorm is regarded as a cell, which is considered to be “a local maximum in weather radar data that undergoes a life cycle of growth and decay” (AMS glossary of meteorology).

1.1.3 The market niche of nowcasting

Why the nowcasting of convective cells is needed? Broadly speaking, according to the study conducted by Deutsche Bank (Auer 2003) over 80% of all the economic activities are directly or indirectly affected by weather. This study also refers that the demand of weather derivatives has increased constantly during the last decade and in the coming years the annual growth of world market in the weather derivatives is expected to run at double digit rates. The interest is also increasing among the companies that have not previously paid heed to the possible weather exposure.

The nowcasting of convective storms is a unique and yet an important branch in the field of weather forecasting. Especially considering the economical and societal needs, nowcasting of convective storms has a lot of potential. Many different fields could benefit directly from nowcasting products. To mention a few, electric production, water management and agriculture are directly influenced by the convective storms. A thunderstorms producing lightning may cause severe damage to the power distribution network; flash floods are loading the sewerage and water management system; intense hailstorm may ruin the whole crop; and every summer fierce thunderstorm gusts cut down trees, harmfully blocking roads. All these examples may lead to extensive economic losses. Even though we are not able to avoid the occurrence of convective cells, nowcasting enables us to prepare and mitigate the impact of convective storms.

The nowcasting of convective cells is also a safety precaution. Convective thunderstorms are especially dangerous to aviation and marine traffic. In air traffic, the real time monitoring of convective cells is highly valuable since the cells can be very hazardous for airplanes and strong up- and downdrafts may lead to uncontrollable behavior of an airplane. Also subcooled water in convective cells may freeze on the airplane and cause a potential dangerous situation. Therefore, automated systems are needed to decrease the risk of dangerous situations and to make air traffic run better. A significant number of accidents could be avoided if we could take the full advantage of the nowcasting of convective cells. According to Malala (2006) weather is factoring over 23% of all aviation accidents. In US this costs annually an estimated 3 billion US dollars for accident damage, delays and unexpected operating costs.

Additional users of nowcasting include public rescue services, which could utilize these products by giving forewarnings about expected hazards. In addition, for an ordinary consumer it is important to be aware of the surrounding weather for instance during recreational activities.

1.1.4 Remote sensing – the key to nowcasting

Due to the lack of dense and extensive surface weather station network, nowcasting methods are heavily relying on real time *remote sensing* measurements such as weather radar, satellite, and lightning data. For the nowcasting purposes, this data has a sufficient spatial and temporal resolution. In this thesis, data gathered by remote sensing instruments called *weather radar* and *lightning detector* will be used for studying convective cells.

Weather radar is a type of radar which observes *hydrometeors*, i.e. rain, snow and hail. This is the most important instrument for nowcasting due to its large coverage area and high temporal and spatial resolution. The precipitation measured by the weather radar can be visualized through weather radar images (e.g. Fig. 1).

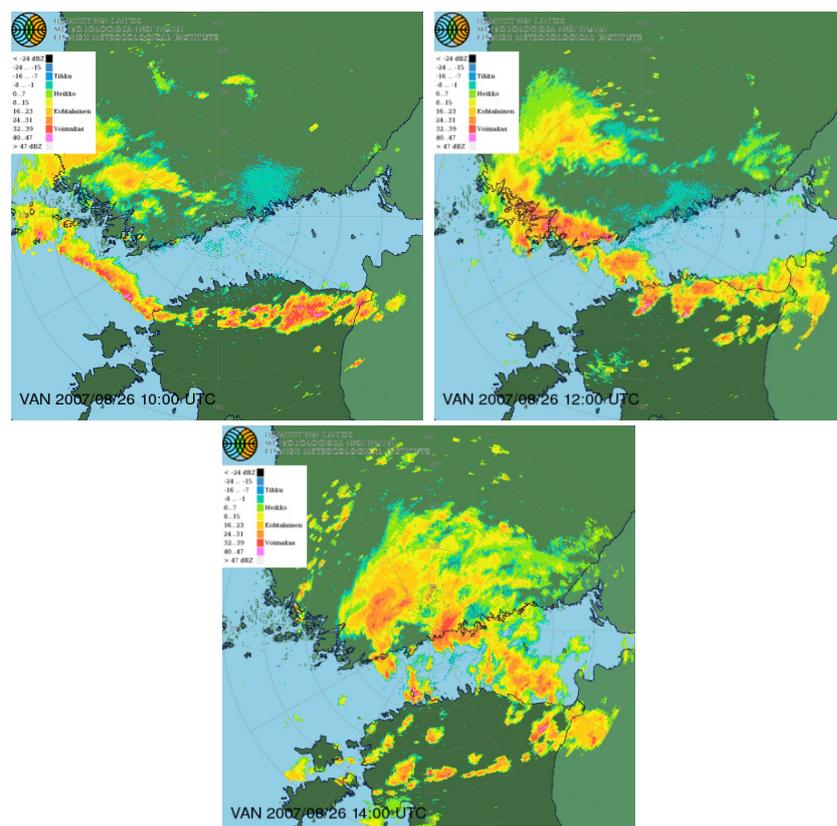


Fig. 1: A sequence of weather radar images added on a geographic map. Colored areas represent precipitation observed by radar (images by courtesy of FMI).

By means of the weather radar, we are also able to identify convective storms and study their spatial and temporal development. In weather radar images, they can be recognized as intense areas featured by a somewhat round, compact shape and an identifiable intensifying and decaying phase.

Lightning detector, in turn, is able to locate lightning flashes produced by the storms. Like weather radar data, also lightning data can be used to identify and study convective storm life cycle and development.

1.1.5 Computer vision based forecasting – a new approach

Remote sensing data can be visualized by digital images and many weather phenomena can be identified from this data. If we take look at a sequence of these images (see Fig. 1), we will find visible patterns of rain and cloud movement. For the human eye, finding of these patterns is not a demanding task, as our brains are well designed for such object recognition. Therefore, radar or satellite image based predictions can be made visually by following image sequences obtained from remote sensing data. This human aided methodology is still a valuable approach for monitoring and predicting forthcoming weather. However, as the amount of data is extremely large, automated systems are needed to ease the prediction task. Since we are dealing with data visualized by digital images, this problem offers an excellent application to the field of computer vision and image processing. In this thesis, different computer vision based methods will be utilized.

In the nowcasting scenario, the question is not only about mimicking the human eye. In addition, through computer aided models we are able to discover important information that goes beyond our visual perception. As an example, different computational techniques are able to acquire features that describe the life cycle of the convective storm and tell us whether the storm is intensifying or dissipating.

1.1.6 Lightning and weather radar data fusion

Due to the tricky nature of convective cells, the forecasting task is demanding. The lifetime of a cell may vary from tens of minutes to several hours. A cell may intensify or it may die before it reaches an observer. For this reason, we need to incorporate as much information as possible to describe the development of the convective storm.

Since both lightning location and weather radar data contain useful information on the convective cell features, the combination of these two data sources is useful. The main goal of this data fusion is to attain more detailed information on the convective cell life cycle and predict its development.

Because we are dealing with small scale fast and rapidly changing phenomena, information sources has to fulfill the requirements of high spatial and temporal resolution. As an example, weather satellites produce spatially accurate information but suffer from the lack of temporal resolution. Nevertheless, both weather radar and lightning location data fulfill these requirements.

Combining lightning and weather radar data is also a quality issue. As both of these information sources contain occasional errors, using the combination of these data types is reasonable; both instruments have their independent error sources and hence they are complementary to each other.

1.2 Objectives and scope of the thesis

1.2.1 Convective cell tracking

One of the standard approaches in computer vision is *object tracking*. In this thesis, the concept of object tracking is applied to convective cells. By tracking convective cells, we may capture the trajectory and the movement of the cells, which can be used, as an example, for a cell extrapolation or monitoring task. In addition, we may attach additional information, such as lightning data, to the cells and analyze their features and development.

In contrast to conventional methods, the convective cell tracking is an object-oriented mechanism that extracts information from individual cells. Since convective cells itself are object-like by nature, this is a viable approach.

1.2.2 Life cycle analysis

As mentioned above, convective cell tracking provides information on cell features, such as movement, size, radar pattern intensity or lightning. This information can be applied for analyzing cell life cycle, that is, cell behavior, characteristic and course of development.

In this thesis, we build up a model that exploits the information provided by a convective cell tracking algorithm. The aim is give an automated estimate whether the cell is intensifying or dissipating. Due to the complex nature of convective cells, this is a difficult task. However, convective cells can be analyzed well by human experts and therefore a human-oriented *fuzzy logic* approach will be applied to overcome the problem.

1.3 Organization of the thesis

The rest of the thesis is organized as follows. Chapter 2 and Chapter 3 introduce important basic concepts that serve as prerequisites for the actual objective. This includes discussion on basic physics of convective events in the atmosphere as well as an insight into the instruments that are used for convective cell exploration in this thesis. However, emphasize will be on data since the used methods rely highly on different data processing techniques such as image processing and computer vision.

In Chapter 4, we have a brief insight into related work on convective cell tracking and nowcasting. Different computer vision based methods for convective cell tracking and extrapolation are viewed.

Chapter 5 contains proposed methodologies that are utilized to achieve the objectives of this thesis. Important image and data processing methods for convective cell identification and representation are discussed. Also a clustering based tracking method for convective cell nowcasting and analysis is proposed. In addition, a fuzzy logic model to analyze convective cell behavior and life cycle is introduced in Chapter 5.

To prove feasibility of the proposed methods, analysis and case studies are considered in Chapter 6. Finally, concluding remarks are given in Chapter 7.

Chapter 2: Basic concepts and fundamentals

2.1 Convection in the atmosphere

A well-known phenomenon is that when fluid is heated its density decreases. Consequently, if the heating is applied to air, warmer and lighter air is influenced by positive buoyancy causing the heated air to rise up and to be replaced by cooler air. This phenomenon is called *natural convection* as the air rises by natural buoyancy forces induced by heating.

In the atmosphere, convective situation appear frequently. Under regular conditions, the temperature of the atmosphere decreases with altitude, which implies that the density of the atmosphere decreases downwards and no buoyancy forces are acting on air. Hence, the atmosphere is stable. However, due to different meteorological reasons, the atmosphere may become labile and susceptible to convection.

In the convective situation, positively buoyant, i.e. instable, air close to the ground surface starts to ascend and simultaneously it is replaced by surrounding cooler air. The domain where the convective circulation takes place is called *convective cell*. This is a prerequisite for a thunderstorm and under suitable conditions each of these convective cells may develop a unit having the characteristic of a thunderstorm. Such a deep convection may result in vigorous ascending currents reaching the height of 10km or even more, and heavy rain, lightning and even hail often accompany it.

In Finland convective thunderstorms appear usually during the summer time when prevailing weather is favorable. These thunderstorms need always some external startup impulse to trigger the convection. This triggering force may be, as an example, a cold air weather front or the sun radiation heating the surface of the earth (Tuomi 1993).

If we consider a convective cell with stage of development, we will find that these stages usually repeat themselves. Byers and Braham (1949) studied the life cycle of convective cells and divided the life cycle in three broad stages depending on the convective circulation speed and direction in a cell:

1. Cumulus Stage – characterized by vertical updraft within the cell
2. Mature Stage – both updrafts and downdrafts appear
3. Dissipating Stage – the cell is featured by downdrafts, finally causing the cell to die out

This classical division is still widely used and often cited in the literature. Regardless of its simplicity, it is still a useful conceptual model for the life cycle convective cells. In the following, a more precise description of the features of each life cycle stage is given.

2.1.1 Cumulus stage

The development of every convection cell begins from the cumulus cloud stage. Cumulus cloud is an easily recognizable "puffy" cloud which appears during the summer time and it is often related to fair weather. However, cumulus clouds may reach the thunderstorm

features if the weather factors like moist, instability and temperature gradient are suitable. Still, only a small number of cumulus clouds continue their growth to attain the thunderstorm features.

The cumulus stage is featured by the updraft (Fig. 2.a), which transfers moist and warm air from the ground level into the cell. Usually these updrafts are between 8 – 10 m/s (Mäkelä 2006), but these speeds may reach as high velocities as high as 15 m/s (Byers and Braham 1949). The updraft speeds within a convective cell is an important factor because it is related to the thunderstorm ferocity. Generally, more convective energy there is in the atmosphere higher the updraft speeds are and the likelihood for severe weather features such as lightning and hail increases.

In this stage, the temperature inside the cell is higher than in the environment guaranteeing the uplifting of air. The greatest buoyancy forces are found at upper levels of the cell, where also the greatest temperature differences occur (Byers and Braham 1945). This is natural, since the buoyancy is proportional to the density differences in fluid and thereby proportional to temperature differences. As the cross section in Fig. 2.a shows, the temperature isotherm is higher inside the cell compared to the cell environment.

Some precipitation is observed inside the cell, especially above the freezing level, which may occur as liquid, solid or both. However, as the updraft is carrying precipitation upwards, rain is not observed on the ground at this stage.

2.1.2 Mature stage

While the air inside the convective cell continues ascending, more and more moist condensates forming visible water particles. This is followed by rain particles and above the freezing level snow and hail. Finally, the size with the mass and gravitation of particles exceeds the force of flow dragging particles up, and the particles start fall down relative to the earth: the cell has achieved the mature stage as illustrated in Fig. 2.b.

If the falling particles contain snow or ice, they start to melt when reaching the freezing isotherm. The melting consumes thermal energy from air which lowers the temperature inside the cell. Under the cloud, falling water starts to evaporate as the air beneath the cloud is drier and unsaturated. The evaporation consumes even more heat energy from air. As a result, melting and evaporating precipitation decreases the buoyancy effect and calms down the updraft.

The above conditions apply only for still air and hence they do not occur very often. Under usual conditions, wind shear has effect on the life cycle development. Due to high wind in the upper levels, a cell is slightly tilted and the falling precipitation does not pass through the updraft part of the cell but falls adjacent to the updraft. Hence, falling precipitation is not imposed immediately on the rising current within the cell and the updraft continues longer.

On the ground, first rain showers are observed during the mature stage. When the moving air encounters the surface of the earth, its direction changes from vertical to horizontal producing one of the most characteristic meteorological phenomenon of the thunderstorm: gusty and strong wind blowing outside from the rain shower area. Eventually, the storm cooled ground wind encounters the storm inflow area and cut down the updraft inside the cell (Byers and Braham, 1949)

Strong updraft and hails amplifies the electrification of the convective cell and therefore lightning is intense in the mature phase. A more detailed introduction the electrification process is given in Section 2.2.

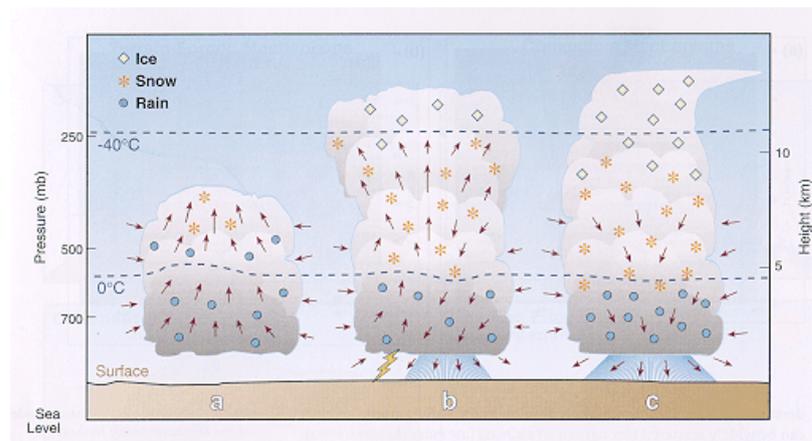


Fig. 2: Convective cell life cycle phases adapted from Byers and Braham (1945). a) Cumulus stage b) Mature stage c) Dissipating phase (Source: National Weather Service)

2.1.3 Dissipating stage

In the dissipating stage, the ground precipitation diminishes until the last residual drops have fallen into the ground. Falling rain and evaporation cool air inside the cell and contributes to the dissipation. Finally, the vanishing updraft turns into a downdraft (Fig. 2.c), which spread throughout the cell body. Since no updrafts occurs in the dissipating phase, storm electrification and consequently lightning decreases and disappears. At the end all that is left is an anvil-shaped cloud in the upper atmosphere consisting of crystallized ice.

2.2 Storm electrification and lightning

A well-known feature of convective cells is *lightning*: an electrical discharge in the atmosphere. Lightning can strike between two clouds or inside a cloud (cloud-to-cloud lightning, intracloud lightning) or between the cloud and the ground (cloud-to-ground lightning). Cloud-to-cloud (CC) lighting occurs more frequently compared to cloud-to-ground (CG) lightning. It is estimated that over 75% of all lightning consists of cloud-to-cloud lightning, from which majority take place inside the cloud (Rakov and Uman 2006).

Physical reasons for thunderstorm electrification are controversial and only a part of the electrification process is known. Prevailing consensus is that convective storm electrification is related to *noninductive graupel-ice mechanism* and *induction charging*

mechanism (e.g. MacGorman and Rust 1998). In addition to these mechanisms, many other theories have been proposed but nowadays graupel-ice combined with inductive electrification is considered as the dominant process of storm charging.

The above electrification mechanisms are related the structure of storm precipitation particles i.e. hydrometeors. Generally, conditions for storm electrification are favorable when strong vertical currents are accompanied with small ice crystals and hail. Electrification may develop also in non-convective clouds but it attains hardly ever the lightning stage (e.g. Mäkelä 2006). Hence, lightning can be regarded as an unambiguous discriminator of convection. Lightning in convective cells is also connected with the storm maturation and therefore it provides important information on the storm life cycle and development.

2.2.1 Graupel-ice-mechanism

This electrification process requires presence of *graupel* particles, which exist frequently in convective cells. The formation of graupel begins when supercooled small water droplets meet crystallized ice. These water droplets may stay supercooled far beyond the freezing point unless they do not have contact with any solid body. Therefore, a contact between a supercooled water droplet and a crystallized ice particle results in the freezing of the water on the surface of the particle. After this, even more droplets stick and freeze on the particle in the process called *riming*. Due to the riming, the size and dimensions of the particle increase resulting in a larger graupel particle.

In the graupel-ice mechanism, collisions between small ice crystals and larger graupel particles cause the storm electrification. The microphysics of the mechanism remains still poorly understood and most of the knowledge is based on empirical results (Rakov and Uman 2006; MacGorman and Rust 1998). The magnitude and sign of charge rely much on different parameters, such as temperature and liquid water content. According to laboratory experiments (Jayaratne et al. 1983), the ice crystal acquire negative charge and the graupel particle positive charge if the collision take place at temperatures higher than $-10\text{ }^{\circ}\text{C}$. If the temperature falls under the *reversal temperature* T_r , which lies generally between -10 and $-20\text{ }^{\circ}\text{C}$, charge signs are reversed.

Due to the convection, particles with different sizes drift apart; heavier graupel particles stay in the lower part of convective cell while the upcurrent carries light ice crystals upwards. This produces an electric field between lower and higher part of the cloud.

2.2.2 Inductive electrification

Also another mechanism called *inductive charging* (e.g. Rakov and Uman 2006) may accelerate and contribute to the storm electrification. This mechanism cannot operate by itself and requires an initial electrical field, which is obtained through the graupel-ice-mechanism described above.

In the ambient electrical field, ice and graupel particles are polarized, that is, the upper part of a precipitation particle is negatively and the lower part positively charged.

Consider now that an ice crystal collides with a graupel particle as the upcurrent transfer light ice crystals upwards (Fig. 3). During the short contact time, some of charge is transferred from the graupel particle to the ice crystal. If the ice crystal rebounds back quickly, the ice particle will obtain an additional positive charge leaving the graupel particle negatively charged. Because the ice crystal carries positive charge upwards and heavier and negative graupel particles fall downwards, the electrical field amplifies as a result. Finally, if the magnitude of the electrical field is strong enough, it may permit a lightning to discharge.

Some researchers have proposed that a convective cell is not able to produce lightning without the aid of inductive charging. However, the importance of this mechanism in thunderstorm charging is controversial, and it is also argued that this mechanism is not capable of producing lightning itself (MacGorman and Rust 1998).

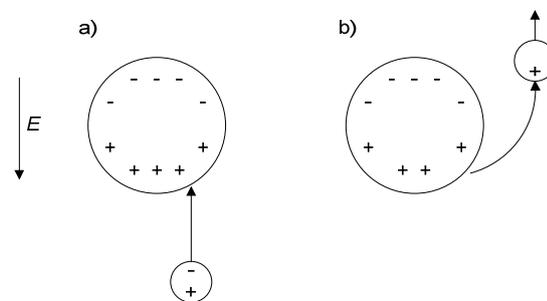


Fig. 3: Inductive charging mechanism. A small polarized ice crystal collides with a larger polarized graupel (a) particle and transfers a unit discharge (b). As a result, the charging increases in the ice crystal and decreases in the graupel particle, which intensifies the electrical field E . (Adapted from MacGorman and Rust 1998).

2.3 Multicellular storms

The discussion above on the convection cell life cycle – growth, maturation and decaying – is an idealized situation. In reality, convective cells are usually accompanied and affected by other neighboring cells. *Mesoscale convective system* (MCS) is an ensemble of convective cells producing widespread contiguous precipitation. Within such a long-lived system, new cells emerge and die constantly. In addition, the dimensions of a MCS are much larger compared to a single-cell system and may reach a size of more than 200 kilometers (e.g. Puhakka 1997).

Zipster (1982) defined four general attributes that makes distinction between the single-cell thunderstorm and the MCS:

1. There must be a group of thunderstorms that have deep convection during the life cycle of a system.
2. The lifetime of the system must be several times larger than the lifetime of an individual cell.
3. The anvils of individual cells within a MCS merge at the upper level to form finally a single cloud shield.

4. The individual downdrafts combine and form a continuous *cold pool* i.e. zone of cool air.

Zipster (1982) divided also the life cycle of the MCS into broad stages: formative, intensifying, mature and dissipating. In the formative stage, individual cells are born and precipitation occur only in the convective regions. In this phase storms appear as disordered clusters.

As the time goes by, storms start to develop and organize. Individual storms grow roughly according the conceptual life cycle model for single-cells described above. However, close neighboring cells in the system may split and merge as the dimensions of individual cells increase. The storm total movement is not only defined by the movement of all individual cell cores within the system but also by new cells emerging in the periphery of the MCS (Puhakka 1997).

In the mature stage, both stratiform and convective precipitations coexist. Downdrafts, rain and evaporation from cell cores cools the surrounding air resulting in a pool of cold air in the lower atmosphere. As the cold pool increases, the cold air becomes to rush out towards the warmer air. The leading edge of the rushing air is called the *gust front* producing a strong burst-like wind (Rauber et al. 2005). The rushing cold air feeds the thunderstorm and triggers new convective cells by forcing the warmer surrounding air to rise up. Usually a new cell emerges in a location where the outflow from the cell is opposite to the inflow and convergence of the rushing air is high (Puhakka 1997). These new cells in the periphery of the original cells keep the MCS alive; when an individual cell in the system starts to decay, it produces a new cell in its vicinity. This mechanism explains the long life cycle of the mesoscale convective systems, which may reach several hours.

Finally, the MCS reaches its dissipating stage: downdrafts dominate the vertical flow within the system and the MCS starts to disappear.

Chapter 3: Data and instruments

3.1 Weather radar

Precipitation is a key factor when examining convective cells and severe weather. Thunderstorms always produce heavy rain or even hail. Heavy precipitation is also an important feature in convective cell recognition and the intensity of the precipitation tells much on the convective cell intensity.

Radar (RADio Detecting And Ranging device) is an instrument that is able to monitor precipitation. It sends electromagnetic microwave radiation into the atmosphere. When radiation encounters an object, such as a raindrop, a part of it is backscattered towards the radar antenna. In the radar, backscattered radiation energy, called *radar echo*, is gathered. The idea of microwave transmission and radar echo gathering is depicted in Fig. 4

Weather radar is an active remote sensing device. Whereas passive remote sensing devices for example in weather satellites usually need external radiation, like visible light or infrared radiation, radar is independent on these external sources. It produces own backscattering energy and therefore it can be used in different weather conditions or during the day and nighttime.

Radar is widely used in different fields. In addition to meteorology, for example aviation, marine and military applies radar for different purposes, like monitoring, navigation and surveillance. In practice, all the radars function according to the same principles and technical framework.

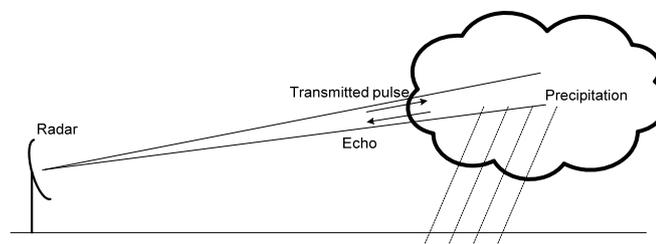


Fig. 4: The principle of weather radar.

3.1.1 Pulsed weather radar characteristics

Modern weather radars are *pulsed radars*, i.e. the radar sends a short microwave pulse and then waits for the echo from a target. Since we know that the microwaves propagate at the speed of light, we can estimate the distance of the backscattering object by measuring the time difference between the transmitted pulse and the radar echo.

Weather radars are usually *single polarization radars*, i.e. the electromagnetic pulse sent by the radar is horizontally polarized. Single polarization radar is also considered in this thesis. However, more advanced *dual polarization radar* is also exploited in different meteorological applications but this radar type is beyond the scope of this thesis.

The *transmitter* of the radar sends pulses of appropriate period, which is usually of class $1\mu\text{s}$. A very important concept regarding the radar functioning is *pulse repetition*

frequency (PRF), which is by definition the number of pulses per second sent in the atmosphere. A typical PRF in weather radars is between 200 Hz and 3000 Hz (Rinehart 2004).

The PRF limits the maximum range of a target that the radar can detect unambiguously. If the time difference between transmitted pulse and radar echo is t , the distance of the target is $r = ct/2$ because the pulse travels back and forth at the speed of light c . If the target is too far, the pulse will be transmitted before the echo gets back to the radar receiver and the echo will be interpreted incorrectly as caused by the last pulse. Thus, given the duration Δt between two pulses the maximum range of the radar detection is

$$r_{max} = \frac{c\Delta t}{2} = \frac{c}{2PRF}. \quad (3.1)$$

Eq. (3.1) implies that there is a tradeoff between the maximum unambiguous detection range and the maximum temporal resolution. By increasing the PRF, we may produce a fast radar scan. Under rapidly changing weather conditions, it is important to update the weather radar image frequently, which requires a fast scan. In addition, to produce different weather radar images, radar scans the atmosphere with several azimuthal and elevation angles, which is time-consuming. Therefore, a relatively fast radar scan is preferred. On the other hand, the fast radar scan reduces the maximum detection range.

Another notable factor that limits the radar detection capability is the pulse length τ , which has a spatial extent $h = c\tau$. If two targets are too close to each other, the radar cannot identify them as individual targets. In order to identify targets as separated objects, echoes from the targets has to be separated also; if the echoes overlap, they are interpreted as single uniform echo. Since the pulse between the two point targets travels back and forth, the distance between these object has to exceed $h/2$ to avoid the confusion.

The third important concept regarding the pulsed radar resolution is *contributing volume* (also called as contributing region, pulse volume, resolution volume or sampled volume) (e.g. AMS Glossary of Meteorology). Contributing volume is a conical frustum determined by one-half the pulse length h and horizontal and vertical beamwidth of the radar. Within the contributing volume all scatterers contribute to the instantaneous radar echo and they cannot be identified as individual objects. Therefore, the contribution volume defines the maximum spatial resolution of a radar scan.

As stated above, the radial extent the contributing volume is $h/2$ since objects closer than this distance are contributing to the same radar echo. Also the width of the cone frustum depends on the range and therefore the contributing volume grows with the distance. For this reason, the pulse resolution decreases as the distance from the radar grows.

3.2 Radar equations

Usually we are not only interested in where precipitation takes place but also in the intensity of the precipitation. Especially considering weather radars, the strength of the radar echo gives important information on the scattering objects. The strength of the radar

echo relies on the pulse strength as well as on the range and physical properties of the backscattering object. Modern weather radars can measure the received signal strength, which in turn can be used to estimate several properties of the precipitation. In order to understand these relations, we need to cover some theory how the radar and target properties affect to the radar echo.

3.2.1 Point target radar equation

Consider a small point-like target, which lies at range r from the radar. Consider also an *isotropic* radar antenna, that is, radar electromagnetic radiation is distributed evenly on the surface of the sphere of radius r . The power density S_{iso} at range r is given as

$$S_{iso} = \frac{P_t}{4\pi r^2}. \quad (3.2)$$

However, a normal weather radar beam is non-isotropic where power is centered in the beam. The radar sends still the same amount of energy in the atmosphere and therefore the power density in the beam can be estimated as follows

$$S_{inc} = S_{iso} G = \frac{P_t G}{4\pi r^2}, \quad (3.3)$$

where G is the *gain* function of the antenna describing how the power density is gained in the beam with respect to the isotropic radiation S_{iso} . When the microwave pulse encounters the target, a part of its energy is backscattered towards the radar. The *backscattering cross section* σ is a coefficient that defines the energy backscattered from the target back to the radar (Puhakka 2000). Power that the target radiates back is given by σ as

$$P_\sigma = \sigma S_{inc} = \sigma \frac{P_t G}{4\pi r^2}. \quad (3.4)$$

The backscattering cross section is a function of several parameters. Not only the shape and type of matter of the object but also the wavelength of the radar has effect on σ (Rinehart 2004).

Usually energy is scattered isotropically (3.2) from the target object. Therefore, power received in the antenna is

$$P_r = A_e \frac{P_\sigma}{4\pi r^2} = A_e \frac{\sigma P_t G}{(4\pi r^2)^2} = A_e \frac{\sigma P_t G}{16\pi^2 r^4}. \quad (3.5)$$

In here, A_e is the effective area of an antenna and describes the proportion that antenna receives from the backscatter power density. It can be expressed with gain and wavelength λ as

$$A_e = \frac{G\lambda^2}{4\pi}. \quad (3.6)$$

Finally, we obtain the point target radar equation that describes the relationship between the transmitted power P_t and received power P_r ,

$$P_r = \frac{P_t G^2 \lambda^2 \sigma}{64\pi^3 r^4}. \quad (3.7)$$

Note that with *point targets* the received power is proportional to $1/r^4$ and thereby received echo weakens significantly as the distance of the target grows.

3.2.2 Radar equation for distributed targets

If we examine targets including many individual targets that fill the radar beam, the received signal is different due the different backscattering properties of distributed targets. Assuming that the distribution of the targets is constant (e.g. drop size distribution of rain) and targets are numerous, the backscattering in the contributing volume V_c can be expressed simply as

$$\sigma = \sum_{j \in V_c} \sigma_j = V_c \sum_{j \in V_c} \frac{\sigma_j}{V_c} = V_c \eta, \quad (3.8)$$

where η denotes *radar reflectivity*. By substituting (3.8) simply into (3.7), we would assume the radar gain G is constant for all the backscattering objects within the V_c . However, we need to take into account that within the contribution volume gain depends on the angle θ between radar echo centre line and a particle. In that case, received energy is obtained by integrating received energy over the V_c :

$$P_r = \frac{P_t G^2 \lambda^2}{64\pi^3} \int_{V_c} \frac{G(\theta)^2 \eta(\theta, r)}{r^4} dV. \quad (3.9)$$

If we assume that the antenna gain function has Gaussian, shape the outcome of the integral (3.9) is

$$P_r = \frac{P_t c \tau \theta^2 G^2 \lambda^2 \eta}{1024\pi^2 \ln(2) r^2}. \quad (3.10)$$

Eq. (3.10) states that the received power from distributed objects P_r is proportional to $1/r^2$. This is a significant difference to the point target radar equation (3.7) that is proportional to $1/r^4$.

3.2.3 Weather radar equation

Weather radar equation is based on the distributed target radar equation (3.10). Since precipitation consists of multiple backscattering targets (hydrometeors), we may use this equation by substituting corresponding radar backscattering cross section into the equation. However, backscattering cross section may be complicated and impossible to calculate analytically (Rinehart 2004). Fortunately, most hydrometeors have relatively simple shape and they can be approximated by spheres. For a spherical object of diameter

significantly *less than radar wavelength* (so called Rayleigh assumption), backscattering cross section can be approximated as follows

$$\sigma = \frac{\pi^5 |K|^2 D^6}{\lambda^4}, \quad (3.11)$$

where D is diameter and K is the magnitude of complex dielectricity coefficient. As (3.11) shows, the backscattering cross section of a spherical hydrometeor is proportional to the sixth power of the diameter. Hence, for example a 2 mm raindrop will backscatter $2^6 = 64$ times more energy than a 1 mm raindrop. Since we know that radar reflectivity of distributed targets can be represented as a summation of all individual radar reflectivities (3.8), we may rewrite (3.10) for hydrometeors as follows

$$P_r = \frac{\pi^3 P_t c \tau \theta^2 G^2 |K| \sum D_i^6}{1024 \ln(2) \lambda^2 r^2}. \quad (3.12)$$

Let us define *radar reflectivity factor* z by summing over the diameters of hydrometeors to the power of six in the contribution volume

$$z = \sum_{V_c} D^6. \quad (3.13)$$

The radar reflectivity factor z is purely a meteorological quantity (Puhakka 2000). Eventually, we obtain the final weather radar equation by replacing summation in (3.12) with the radar reflectivity factor z .

$$P_r = \frac{\pi^3 P_t c \tau \theta^2 G^2 |K| z}{1024 \ln(2) \lambda^2 r^2}. \quad (3.14)$$

This quite general equation is applicable to any radar, provided that hydrometeors are small compared to the wavelength. Upon usual conditions this is not a problem; hydrometeors are typically a couple of millimeters of size and the used wavelength is several centimeters.

In meteorology, intensity of the measured radar reflectivity z [mm^6/m^3] is converted to logarithmic scale

$$Z[\text{dBZ}] = 10 \log_{10} z[\text{mm}^6 / \text{m}^3]. \quad (3.15)$$

This is an important quantity and it is also used in this thesis to represent intensity of precipitation observed by a radar. Additionally, it corresponds quite well with rain intensity from subjective point of view. As an example, reflectivity values exceeding 40 dBZ are usually correspond with heavy rain and therefore this threshold is associated with convection.

3.3 Weather radar data representation and visualizing

3.3.1 Plan position indicator (PPI)

Radar echoes are usually mapped to two dimensional *weather radar images*, where image intensity is represented with varying degrees of brightness. Since the weather radar scans a full 360 degree azimuth coverage, it is convenient to represent weather signal this way. *Plan Position Indicator* (PPI) shows full 360 degree coverage with a fixed elevation angle (Rinehart 2005; Puhakka 2000). The PPI output depends on the elevation angle. With higher elevation angles, the beamline is steeper and the height of the echo is increases fast with distance from the radar.

3.3.2 Constant altitude plan position indicator (CAPPI)

In PPI, the altitude of echoes increases with the distance to the radar. This means that echoes from distant phenomena are obtained from higher altitudes. Therefore, in order to obtain information from a constant altitude, a different method is needed to represent weather radar data.

Constant Altitude Plan Position Indicator (CAPPI) is a combined PPI with different elevation angles so that the height of echoes does not increase with distance. This is a remarkable advantage over the PPI's altitude variation, if we need information from a constant altitude. However, due to limited amount of elevation angles, the CAPPI does not represent constant altitude precisely but this error is usually considered negligible. This effect is illustrated with the saw-tooth line in Fig. 5.

In meteorology, CAPPI is an important product. It can be used for representing rain patterns, which coincides quite well with rain observed on the ground (especially with low altitude CAPPI images). It can be easily combined with geographical maps and used for representing rain relative to geographical locations, as in Fig. 1. In this thesis, CAPPI images having altitude of 500m, denoted as CAPPI 500 m, plays an important role and it act as main radar output when recognizing convective cells.

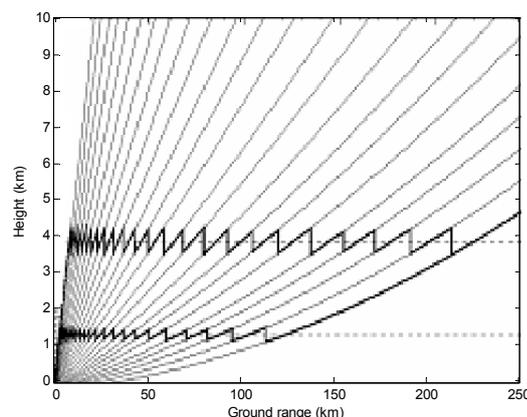


Fig. 5: A schematic illustration of CAPPI. Due to limited amount of elevation angles, CAPPI (marked with the solid saw-tooth line) does not represent constant altitude (the dashed line) precisely (Wikipedia 2008a).

3.3.3 EchoTop

When a radar scans a full 360 degrees in azimuth with successive elevation angles, a three dimensional reflectivity field is obtained and received data can be digitalized in a three dimensional grid. Consider now a three dimensional Cartesian grid in which each volumetric element $Z_{i,j,k}$ represents corresponding value of radar reflectivity factor. EchoTop altitude of threshold T is the highest altitude having an echo stronger than T . If indices $A_{i,j,k}$ is the altitude of the echo $Z_{i,j,k}$, *EchoTop* altitude for given i and j is as follows

$$EchoTop_{i,j,T} = \max_k(A_{i,j,k} | Z_{i,j,k} > T) \quad (3.16)$$

Since a three dimensional grid is projected into two dimensional grid, it is convenient to visualize the output with a digital image. This product is useful when we examine severe weather like convective cells. As it is stated earlier, precipitation may reach really high altitudes in a convective cell during the mature stage and therefore this product can be used for studying the life cycle of the cell. For example, aviation uses this product. Airplanes are advised to avoid high EchoTop regions as they can cause hazardous problems. A typical value of T utilized in radar meteorology is between 20 dBZ and 45 dBZ. In this thesis EchoTop of threshold $T = 20$ dBZ, denoted as EchoTop 20 dBZ, has an important role in the life cycle analysis of convective cells.

3.3.4 Anomalies and error sources

Even under the clear air conditions, weather radar data is usually affected by some error sources. Unfortunately, the hypothesis that atmosphere contain only hydrometeors is untrue. For example, insects or birds can be interpreted erroneously as precipitation. In addition, not only atmospheric objects but also solid objects on the ground have effect on the radar echo. Especially with lower elevation angles, part of radar energy is backscattered from buildings, mountains etc., which can be observed as *ground clutter*. Because the altitude of the radar echo scan increases with the distance, ground clutter occurs especially in the vicinity of radar.

In addition to possible error sources described above, another important factor regarding radar data quality is *attenuation*. Electromagnetic radiation passing through any medium is influenced by attenuation. For example clouds, atmospheric gases and hydrometeors attenuate transmitted pulse and radar echo. Therefore, if strong precipitation is blocking the radar, echoes may have anomalously low values. An example of attenuation is given in Subsection 6.1.5.

In addition to the possible error sources described above, there exist several other sources such as excessive refraction, antenna side lobes, second trip echoes etc. A more precise description of different error sources is given for example by Rinehart (2004).

3.4 Locating lightning

In this thesis, estimated lightning locations is utilized in order to understand intensity and life cycle development of a convective cell. As mentioned earlier in Section 2.1, the first lightning occurs during the mature stage and diminishes when the storm reaches the

dissipating stage. Therefore, knowing where lightning take place is essential to understand storm electrification and relationship between lightning and storm life cycle.

Different techniques are available for storm electrification and lightning detection. All the lightning location systems are founded on measuring some physical impulse emitted by the lightning stroke. This impulse can be, as an example, a visible light flash, a thunder or an electromagnetic pulse emitted by the stroke. For example, the flash or sound of thunder gives us a rough estimate of the direction the lightning stroke originates from. Since we know the velocity of sound in the atmosphere, we may calculate the distance by comparing the time difference between acoustic and visual observation.

Nowadays, modern lightning detection systems utilize electromagnetic impulses emitted by a lightning stroke; lightning emits electromagnetic energy in the frequency range from 1 Hz to 300Mhz or even higher (such as visible light roughly from 10^{14} to 10^{15} Hz) (Rakov and Uman 2006). In this thesis, three different electromagnetic based approaches are introduced briefly: *direction finding*, *time of arrival* and *interferometry* based lightning locating (e.g. Rakov and Uman 2006). These methods are also exploited in the detected lightning data used in this thesis. The data is provided by the Finnish Meteorological Institute Lightning Location System (FMI LLS).

Lightning data are often combined e.g. as a layer on top of the radar imagery. As we will see in Chapter 6, the most intensive and hazardous areas are easier to identify as the highest lightning activity is well correlated with the most intensive cell cores seen by the weather radar. The aim is to connect observed lightning flashes to corresponding convective cell, which provides information on life cycle development and amount of lightning in each cell.

3.4.1 Magnetic direction finding

Magnetic direction finding is based on estimating the direction of electromagnetic impulse emitted by a lightning flash. To calculate the estimated location of the flash, two or more of direction finder instruments are needed. Since one direction finder estimates the angular direction, the location is obtained simply by calculating the intersection point of the direction lines, as depicted in Fig. 6.

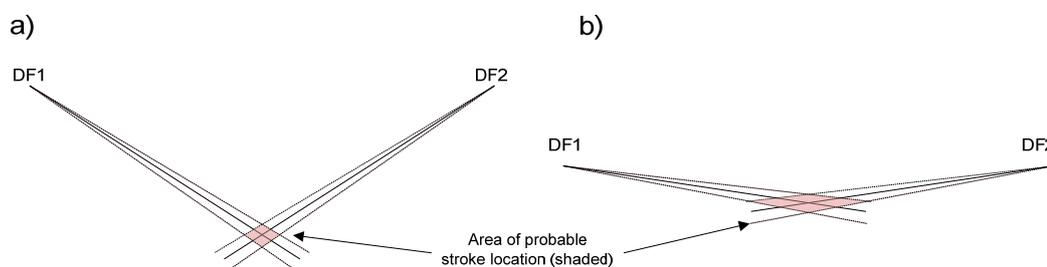


Fig. 6: a) Lightning stroke location when only two direction finders (DFs) are utilized. The solid lines represent estimated azimuth of the stroke and the dashed lines angular error in measurements. The shaded area represents the possible area of uncertainty due to the angular error. b) If the direction lines get parallel to each other, the probable area of stroke location grows.

This method requires that the radiator (lightning flash) is vertically oriented. Under such assumption electric field is vertically and magnetic field is horizontally oriented in the electromagnetic wave, so that the magnetic field is perpendicular to the propagation path. This assumption is adequate for cloud-ground lightning since the discharge is vertically oriented.

The basis of the direction finder consists of two vertical loops oriented along North-South and East-West directions. According to the Faraday's law of induction, changing magnetic field emitted by a lightning induces voltage in the loops and the output voltage is directly proportional to magnetic field vector component that is parallel to the loop plane. Consider now North-South and East-West oriented magnetic field vector components H_{NS} and H_{EW} . The azimuthal bearing θ_m of the electromagnetic field direction can be calculated by the voltage V_{NS} measured in the North-South oriented loop and V_{EW} in the East-West oriented loop as follows (Rakov and Uman 2006)

$$\theta_m = \arctan\left(\frac{H_{NS}}{H_{EW}}\right) = \arctan\left(\frac{V_{NS}}{V_{EW}}\right). \quad (3.17)$$

The detection of a lightning location through this kind of methodology is naturally susceptible to different errors. If only two direction finders are used, the resulting error of detected location may be excessive (shaded area in Fig. 6.b). Naturally, the accuracy improves when more direction finders are included in the location estimation. In larger networks, lightning location is estimated by searching such a location that minimizes the following cost function (MacGorman and Rust 1998; Bevington 1969)

$$\chi^2 = \sum_i \frac{(\theta_i - \theta_{mi})^2}{\sigma_{azi}^2}, \quad (3.18)$$

where θ_{mi} is the measured bearing, θ_i is the estimated bearing, and σ_{azi}^2 is an expected azimuthal error in the measurement by the i th station.

Random errors in direction finders are due to superimposed noise from antenna output and imperfect instrumental processing and digitalizing. In addition, direction finders are often interfered with systematic site errors which originate e.g. from large conducting object or power lines. The self-consistency for each direction finder can be estimated by using other stations as reference (McGorman and Rust 1998).

3.4.2 Time of arrival technique

In this technique, each station identifies and records the time of arrival of the electromagnetic pulse emitted by a lightning. If we have two stations and the exact locations of the stations, by calculating the time difference of arrival, we can estimate the difference in distance from the stroke point to the stations. Therefore, we may define a locus of constant difference in distance, which is a hyperbola for a flat plane. However, if the curvature of the Earth is taken into account, the locus is distorted from hyperbola.

If a third station is added to the system, the location of a strike can be estimated by calculating the intersection point of two hyperbolas defined by the stations. However, this may lead to an ambiguous solution: two hyperbolas often have two intersection points, which both are possible lightning locations. For this reason, this ambiguity is removed by adding a fourth station to the system.

The quality of the time of the arrival technique depends on the lightning intensity. If the lightning is abundant, the likelihood of two simultaneous strokes becomes evident, which naturally may confuse the time of arrival detection (Mäkelä 2006).

In order to improve the accuracy, time of arrival and magnetic field direction finding techniques can be combined. In this hybrid system one instrument locates both direction and time difference of a lightning flash. For example, IMPACT (IMProved Accuracy by Combined Technology) sensors in FMI LLS utilize this hybrid methodology. The accurate synchronized time in each sensor is obtained from the GPS satellites. If at least two IMPACT sensors record observations that are simultaneous enough, the central unit calculates the intersection point of direction lines and adjusts the location by the time differences. The algorithm searches for a location that minimizes a cost function given as (MacGorman and Rust 1998)

$$\chi^2 = \sum_i \frac{(\theta_i - \theta_{mi})^2}{\sigma_{\theta_i}^2} + \sum_j \frac{(t_j - t_{mj})^2}{\sigma_{t_j}^2}, \quad (3.19)$$

where t_{mj} measured time of arrival, t_j is the time of arrival from the trial solution and $\sigma_{t_j}^2$ is expected error in the time at the j th station. Other parameters are as in (3.18).

3.4.3 Interferometry based technique

In the interferometry based technique, two or more closely spaced antennas are connected via a narrowband filter to a receiver (Rakov and Uman 2006). Each of these receivers sends the output to a phase detector, which are used to identify the phase difference of the narrowband signal at each antenna. This phase difference can be utilized to estimate the direction of an electromagnetic pulse.

The phase difference in different antennas depends on direction angle of a radiator relative to the baseline between two antennas. A signal of a wavelength λ propagating from the direction θ_i forms the phase difference α between the antennas and is given by

$$\alpha = \frac{2\pi D_a \cos(\theta_i)}{\lambda}, \quad (3.20)$$

where D_a is the distance between antennas. Since we know the signal wavelength and the phase difference α , the direction angle can be calculated by solving θ_i from (3.20). However, if the distance D_a is too large the measured phase difference α and consequently θ_i is cannot be determined unambiguously. To avoid this aliasing effect, the distance D_a has to fulfill the Nyquist sampling criteria $D_a < 0.5\lambda$.

3.4.4 Applied lightning data

The lightning data utilized in this thesis was provided by the FMI LLS. Lightning location data have been collected in Finland since 1984 when the first automatic ground lightning system was set up (Tuomi and Mäkelä 2007). The current network is composed of two types of lightning detection sensors: eight CG flash IMPACT sensors and three CC flash SAFIR (Surveillance et Alerte Foudre par Interférométrie Radioélectrique) sensors. More precise description of the FMI LLS is given in Tuomi and Mäkelä 2007.

IMPACT sensors exploit the combination of the time of arrival and the magnetic direction finding (Tuomi and Mäkelä 2007). Since two different techniques are combined, ambiguities in the detection can be reduced.

SAFIR sensors, in turn, utilize interferometry technique (Rakov and Uman 2006). Each sensor has five antennas in order to reduce phase difference ambiguities. These antennas exploit VHF (Very High Frequency) bandwidth. Since CC flashes tend to produce pulses at VHF frequencies, by means of the SAFIR sensors we can infer whether the impulse is obtained from a CC stroke or a CG stroke. Both CC and CG data have an important role in this thesis (see Chapter 6).

Due to the small amount of SAFIR sensors, the error of located lightning may be substantial and the efficient CC detection area is rather small. In addition, the VHF exploited in these sensors is not able to reach far; high frequencies tend to propagate along a direct line in the atmosphere and due to the earth curvature signals from distant sources are lost. Conversely, low frequencies used in the IMPACT sensors refract in the atmosphere and therefore CG flashes can be detected at further distances.

Chapter 4: Related work

In this chapter, we discuss contemporary methods that are used in the nowcasting of convective cells. Since the focus is on convective cell tracking, an insight into the general concepts of computer vision and object tracking are given at first.

4.1 Computer vision

Computer vision is concerned with obtaining information from the images and building up models from the information. Basically, computer vision is processing and modeling of data that is obtained from digital images. Therefore, many different disciplines such as statistics, pattern recognition and machine learning are applied in computer vision. The term computer vision also refers to seeing and visual information. This means that the physical models we want construct can be identified from the images by the human eye. As an example, we want to recognize different objects from an image that is usually an easy task for a human. Still, images often contain information that is not possible to observe visually.

When we are building up a computer vision model, some information about the process is required *a priori*. This means that we need to know something about the modeled process in advance. This can be, as an illustration, some information on the physical process of the convective cell.

If a time series of images are considered, we want to extract information related to both spatial and temporal dimensions. Consecutive images are usually somehow different and we would like to study these differences and their relations. Due to the dynamics of an image sequence it contains much more information compared to a single snapshot.

Visually perhaps the most influential effect is motion. This is a demanding and yet an important computer vision application: how to extract and identify motion from images through computer aided models? In this thesis digital image based motion estimation is one of the key factors. We want to find out how motion and the life cycle of convective cells can be studied by means of digital weather radar images.

4.2 Tracking as a computer vision problem

The aim of *object tracking* is to generate the trajectory of an object by identifying it over time in consecutive image frames. Usually object tracking can be divided into two important sub-problems: identifying the object and finding object correspondences between different instants. The latter one is of concern, especially if more than one object is tracked at the same time.

In tracking scenario, an object can be determined as anything that is of interest (Jain et al. 1995). For instance, vehicles in a traffic monitoring images or a person in a surveillance video sequence can be regarded as our object of interest. Usually objects are recognized by some characteristic features e.g. their color, area or movement. However, a part of the object information is changing over time. For example, the object may be moving which means that its position changes. We want to find out the correspondence even though

objects are different in different images. This is executed by searching and linking similar features in different frames. Therefore, a requirement for a feature is temporal representativeness; the location of an object may change in different frames but object color not. For this reason, color can be regarded as an important feature for object identification.

Like many other computer vision problems, tracking of objects is not usually complicated for the human eye. As an example, following a flying flock of birds is an easy task for us but serious problems may arise if the task is given to a computer. Overall, computer based object tracking is a difficult task. Difficulties may arise as an example due to abrupt changes in object motion, nonrigidity, object occlusion, camera motion or changes in ambient conditions such as in illumination. (Yilmaz et al. 2006)

Naturally, in this thesis convective cells are regarded as objects of interest. Our objective is to track convective cells in consecutive weather radar images and utilize tracking information for analyzing life cycle properties of the cells.

In the following, fundamentals and general concepts of computer vision based object tracking are viewed. In Subsection 4.2.3 the concept of *point target correspondence* tracking is introduced which is an important basis for the tracking of convective cells. This is followed by a brief survey of contemporary motion estimation and tracking methods in the contemporary nowcasting in Subsection 4.3.

4.2.1 Object detection

Every tracking method requires an object detection mechanism. A common approach is to perform object detection separately in every single frame. However, exploiting temporal information in the object detection could be also reasonable. As an example, if we want to detect a moving object we may need to make a distinction between moving and still regions in the image sequence. In addition, considering temporal development in the detection process may reduce the number of false detections.

Even in the broad sense, object detection includes various approaches. Yilmaz et al. (2006) outlined the most common detection mechanisms as point detectors, segmentation, background subtraction and supervised learning. In addition to these methods, for example fuzzy logics or different heuristics can be included in the detection process. Naturally, feasible methods depend much on the problem. As we will see in the Subsection 4.3.2, convective cell identification method relies heavily on meteorological expert knowledge.

Owing to a vast number of different approaches, the topic is not further discussed in here. More precise introduction to the contemporary object detection mechanisms are viewed for example in Yilmaz et al. (2006). The methodology used for the convective cell detection in this thesis is introduced in Chapter 5.

4.2.2 Object representation

After the objects have been identified, the resulting objects have to be represented with an appropriate method. This can be, for example, a shape or an appearance. The primary aim of a representation scheme is make object suitable for the used tracking algorithm; different algorithms rely on different assumptions on the objects and hence different representations are needed.

The representation is not only about making data useful for the algorithm. It is also about describing the chosen representation (Gonzalez and Woods 2001). As an example, a boundary can be regarded as an object representation. Object boundary itself, however, can be represented with different representation methods such as simple geometrical shapes or polygons, which usually eases the computational burden in many algorithms.

The following list views briefly representation methods that are commonly employed in the tracking scenario.

1. *Points*. The object is represented simply with a point or a set of points. A common approach is to represent object centroid with a point but also other descriptions can be used. This approach is especially considerable if the objects are rigid and occupy only a small region in an image.
2. *Primitive geometric shapes* are often used to simplify and generalize object appearance. Simple shapes, such as rectangle, triangle or ellipse, can be represented with a very few parameters, but the information included may be totally sufficient to represent object properties in the tracking. Primitive shapes are often suitable for representing rigid and convex objects.
3. *Object silhouette or boundary*. Identified objects can be represented with the segmented pixels, i.e. image silhouette. A less complex alternative is to represent object with its boundary, which can be extracted from the silhouette. Obviously, the silhouette and boundary contain more information on the object than other geometric representation methods. However, including all the object pixels is often unnecessary and computationally expensive.
4. *Polygons* are often used to simplify object boundary and to reduce computational complexity in the algorithms. The goal is to represent the boundary with vertices such that some error criterion between vertices and object boundary is minimized. Convex polygons are often preferred, since they usually reduce computational complexity in spatial data algorithms more than non-convex polygons. In this thesis polygons are used for convective cell representation and therefore digital boundary polygonizing is discussed in Subsection 5.2.3.

5. *Skeletons* are a common approach to represent object structural shape by reducing object silhouettes into a skeleton-like graph. A common approach for skeletonizing is the medial axis transform (see e.g. Gonzalez and Woods 2001).
6. *Articulated shape models*. Articulated objects comprise of a set of subobjects that are held together. As an example, the human body is an articulated object with subobjects such as legs, hands, feet, head and torso. Each subobject can be represented by the methods described above. This approach is reasonable, if we know that the object comprises of rigid subobjects that can be represented for example with simple geometric shapes.
7. *Probability density distribution* estimates appearance or location of an object. They can be either parametric, such as the Gaussian distribution, or nonparametric such as histograms or Parzen windows (Theodoridis and Koutroumbas 2003).

In addition to the described models above, there are a number of different other representation methods. In general, the suitable representation scheme is chosen according to the application. For instance, a point representation is often sufficient for very small objects. For larger objects, usually a geometric shape such as ellipse or polygon is needed. As an example, Dixon and Wiener (1993) utilize ellipses for convective cell representation. In the algorithm presented in this thesis, convective cells are represented with non-convex polygons and ellipses.

4.2.3 Point target correspondence tracking

In the presence of a single object, tracking can be performed through several different approaches (see e.g. Sonka 2007). However, if several objects are tracked simultaneously and independently, the problem will be more complicated, as we need to find track correspondences among a large number of different candidates.

Point target tracking can be formulated as finding the correspondences of the detected objects across the frames (Yilmaz et al. 2006). A candidate solution can be regarded as a set of tracks that describes the motion of each object from the beginning to the end. The task is to find the optimal tracking among different candidate tracks. This is, unfortunately, a nontrivial task.

Generally point target correspondence tracking can be divided into two different classes: statistical methods and deterministic methods. Statistical correspondence methods take into account measurement uncertainties of the object and random perturbations during the object state estimation. Usually object state space model is defined by parameters such as position, velocity and acceleration. The goal is to provide the optimal solution in the statistical sense by giving the state estimate or the posterior probability function for each object.

For the single object tracking, the most conventional statistical approach is *Kalman filtering*. Still, this can be an infeasible approach, since Kalman filtering assumes that all

the probability density functions (pdf) are Gaussian, which is often a too coarse approximation. *Particle filters*, such as *Condensation algorithm* (Isard and Blake 1998), try to overcome this problem by using Monte Carlo approximation in order to estimate the posterior pdf. For multiple object tracking, statistical algorithms such as *Joint Probability Data Association Filter* or *Multiple Hypothesis Tracker* (MHT) have been used widely (Yilmaz et al. 2006; Veenman et al. 2001).

Veenman et al. (2001) argue that using statistical methods for multiple object correspondence tracking have several shortcomings. Firstly, different statistical methods rely on a number of different parameters. The determination of the optimal settings of these parameters is not trivial and for example MHT is quite sensitive to different parameter settings. In addition, statistical methods that are performed over several frames are computationally exhaustive as the complexity grows exponentially with the number of objects in frames.

Deterministic tracking methods try to find correct tracks by optimizing correspondence cost of different track candidates. Usually correspondences are considered only between two consecutive frames which leads to the combinatorial optimization problem called *assignment problem*. The well-known formulation of the problem is as follows (e.g. Papadimitriou and Steiglitz 1998):

There are a set of agents and a set of tasks. Any agent can be assigned to any task, but the cost depends on the assignment. Only one task can be given to one agent. The objective is to carry out all the tasks in such a way that the total cost is minimized.

Let A be a set of agents and B be a set tasks. A task $j \in B$ is given to an agent $i \in A$ with the cost c_{ij} . The assignment is represented with variable x_{ij} , which is 1 if the agent $i \in A$ is assigned to the task $j \in B$ and 0 otherwise. In terms of linear programming, the total cost function to be minimized is formulated as

$$\sum_{i \in A} \sum_{j \in B} c_{ij} x_{ij}, \quad (4.1)$$

with subject to the constraints

$$\begin{aligned} \sum_{i \in A} x_{ij} &= 1 \\ \sum_{j \in B} x_{ij} &= 1 \\ x_{ij} &\geq 0. \end{aligned} \quad (4.2)$$

The assignment problem can be easily generalized to the tracking scenario; objects in a frame at t are considered as agents and the objects in the next frame at $t+dt$ as tasks. The solution is a one-to-one tracking, which can be obtained by an exhaustive greedy search or by more sophisticated method like the Hungarian algorithm (e.g. Papadimitriou and Steiglitz 1998). Even though combinatorial optimization based tracking is not

implemented in this thesis, it is important to bear in mind that the tracking problem can be regarded a combinatorial optimization problem.

In order to prevent unnatural tracks, some assumptions and constraints on the object motion have to be formulated. The optimal solution relies on these constraints that are used to form suitable cost functions. Sethi and Jain (1987) defined the following motion criteria **C1** – **C2** that must be considered in the correspondence optimization. This *individual motion model* relies on common physical laws of inertia and restricts unnatural and abrupt movements of displaced objects.

1. **C1: Proximity constraint.** The location of an object cannot change notably between two frames. If only this constraint is considered, the tracking is obtained by nearest neighbor optimization and distance between two points is considered as the cost c_{ij} between points i and j in (4.1).
2. **C2: Smooth velocity constraint** assumes that the velocity of an object cannot change drastically. To formulate this constraint mathematically Sethi and Jain (1987) defined *path coherence function* which is considered as the cost in (4.1). Let $T_i = \langle \mathbf{x}_{i1}, \mathbf{x}_{i2}, \dots, \mathbf{x}_{ik} \rangle$ be the i th track in the k th frame. If the point \mathbf{x}_{ik+1} is added to the track, the path coherence cost is given as follows

$$c_{ik} = w \left[1 - \frac{(\mathbf{x}_{ik-1} - \mathbf{x}_{ik}) \cdot (\mathbf{x}_{ik} - \mathbf{x}_{ik+1})}{\|\mathbf{x}_{ik-1} - \mathbf{x}_{ik}\| \|\mathbf{x}_{ik} - \mathbf{x}_{ik+1}\|} \right] + (1-w) \left[1 - 2 \frac{\sqrt{\|\mathbf{x}_{ik-1} - \mathbf{x}_{ik}\| \|\mathbf{x}_{ik} - \mathbf{x}_{ik+1}\|}}{\|\mathbf{x}_{ik-1} - \mathbf{x}_{ik}\| + \|\mathbf{x}_{ik} - \mathbf{x}_{ik+1}\|} \right], \quad (4.3)$$

where $0 \leq w \leq 1$. The first part of (4.3) can be viewed as *direction coherence* and the second part as *speed coherence*.

3. **C3: Simple motion model.** Using first-order velocities is definitely enough to approximate the motion of individual objects.

Veenman et al. (2001) extended the work of Jain by defining *common motion constraint* and *global motion constraint*:

4. **C4: Common motion.** The velocity of all objects in a small neighborhood of an object should be similar. This can be done for example by enforcing the average deviation from individual motion models to be at minimum. However, this assumption is not suitable for all tracking problems. As an example, two dense groups of objects moving into opposite directions do not satisfy the common motion requirement.
5. **C5: Global motion.** Overall motion from the initial time frame t_0 to the last frame t_n is uniform and coherent.

In addition to the constraints **C1** – **C5**, we may define some additional constraints such as assumption on object rigidity, maximum velocity or maximum displacement (see. e.g. Yilmaz et al. 2006). These common sense based constraints usually reduce ambiguities in the optimization and make the computational part less complex.

4.3 Computer vision based methods in nowcasting

4.3.1 Grid-based motion estimation methods

The concept of extrapolating radar echoes for predicting precipitation began in the 50s before the aid of computers (Wilson et al. 1998). The estimation was performed visually by observing the movement of radar echo patterns. The first computer based models were introduced in 1960. These models used a simple cross correlation method to estimate a single average velocity vector of the whole precipitation pattern (Wilson et al. 1998). Leese et al. (1971) was the first to obtain differential motions within the pattern but the method was applied for satellite based cloud images.

One of the most classical weather radar based nowcasting method TREC (Tracking Radar Echoes by Correlation) was introduced by Rinehart and Garvey in 1978. This correlation-based method divides each radar scans into equally sized boxes and determines velocity vectors by comparing the boxes; the correlation coefficient is computed between all the possible pair of boxes, and the pair having the highest coefficient defines the motion. To reduce the computational burden and range ambiguities, the search is bounded by the maximum range r . The range r can be obtained from the maximum velocity v_{\max} multiplied with the time interval Δt between the echo patterns.

Due to the simplicity, the TREC has shown many shortcomings and inconsistencies. In the presence of ground clutter or rapid changes in radar patterns, the velocity vectors may show a noisy behavior (Li et al. 1995). The TREC method was improved by Li et al. (1995) by forcing the neighboring velocity vectors to be similar enough, which can be thought as an equivalent constraint to the common motion **C1** used in the correspondence tracking. They also applied variational optimization to guarantee the mass continuity of the radar echo pattern.

In addition to the correlation-based methods, another commonly used approach for extracting motion from images is *optical flow* (see e.g. Peura and Hohti 2004). This is one of the standard computer vision methods for estimating motion in image sequences. The flow in images can be modeled with intensity and temporal gradient of radar image included a simple differential equation

$$\frac{df}{dt} = f_t + f_u u + f_v v = f_t + \nabla f \cdot \mathbf{v}. \quad (4.4)$$

It should be noted that optical flow model is very similar to advection in the atmosphere i.e. air transportation (see e.g. Holton 2004). To be precise, the advection model describes the general flow of any physical quantity, such as rain, whereas optical flow in (4.4) describes the intensity flow of the pattern in images. The advection flow model is one of the basic equations in the numerical weather forecasting and hence the estimation of optical flow in weather radar images is very analogous and reasonable.

For the sake of simplicity, it is often assumed that there is no changes in data i.e. $dt/df = 0$. This yields

$$f_t + \nabla f \cdot \mathbf{v} = 0. \quad (4.5)$$

The key idea is to derive flow \mathbf{v} in (4.5), which can be done through different methods. *Lucas-Kanade method* (Lucas and Kanade 1981) is a popular way to estimate the flow \mathbf{v} from the (4.5). The method assumes that the flow \mathbf{v} is constant inside a small window of size Ω . For each pixel, the flow can be computed by minimizing the squared error function

$$J = \sum_{\Omega} (\nabla f \cdot \mathbf{v} + f_t)^2. \quad (4.6)$$

One of the main advantages of optical flow methods is the computational efficiency. While correlation methods utilize matching, optical flow applies differential computing, which requires fewer calculations (Peura and Hohti 2004). On the other hand, robustness and conceptual clarity are one of the strong points of the correlation-based methods.

4.3.2 Tracking methods in nowcasting

The grid-based methodologies are important tools if we want to extract motion field from the whole image area. However, since we are analyzing convective cells the information contained in individual cells cannot be analyzed through these methods.

In order to extract information from individual cells, we need the concepts of the correspondence tracking introduced in Subsection 4.2.3. While the grid-based methods offer information on the overall motion of the reflectivity pattern, by means of the *object-oriented tracking*, we can provide detailed information on individual convective cell tracks and characteristics. This is an important advantage over conventional grid-based methods and therefore in here the emphasis will be on object-oriented tracking methods. Additionally, they facilitate the life cycle analysis, which is one of the main objectives of the thesis. Moreover, convective cells can be regarded as individual natural objects and hence the use of object-oriented tracking methods for convective cells is justified.

The tracking of individual convective cells began in the 70s in the US at the National Severe Storm Laboratory (NSSL) (Wilson et al. 1998). NSSL developed methods for identifying and tracking the cell centroid in successive images (e.g. Barclay and Wilk 1970). Further improvements were designed by Blackmer et al. (1973) by attempting to handle splitting and merging of the cells.

In 1993, Dixon and Wiener introduced the TITAN algorithm (Thunderstorm Identification, Tracking, Analysis, and Nowcasting), which is still applied in the operational nowcasting purposes (see e.g. Mueller et al. 2003). This algorithm identifies convective cell objects as three dimensional ellipsoids. The tracking problem is defined by the combinatorial assignment problem, which is solved by the Hungarian algorithm (see Subsection 4.2.3). The algorithm deals also with cell splits and mergers.

Also several other methods for convective cell tracking have been proposed during the past years. The main differences between different tracking algorithms lie in the object identification method and the correspondence establishing.

Object identification is usually based on a simple thresholding of a suitable reflectivity factor value suggested by a meteorological expert. A typical threshold used in different algorithms lies in the range from 35 to 50 dBZ. Selecting a suitable threshold for cell identification is a tradeoff between the tracking complexity and the correct cell identification; if a small threshold is used, the algorithms tend to identify also non-convective events such as stratiform precipitation. On the other hand, a large threshold value may lead to nonrigid cells and sudden changes in cell behavior, which usually inflicts a complex and fragmental tracking. In addition, a cell is identified only after it has attained the threshold value and for this reason a high threshold value can be too conservative assumption.

After the thresholding, some shape representation method is applied for the objects (see Subsection 4.2.2). A common approach is to utilize polygons (Rossi and Mäkelä 2008), image silhouettes (Hering et al. 2004) or ellipses (Dixon and Wiener 1993).

In order to find correspondent objects in different images, several approaches have been proposed. This can be performed by a rather simple nearest neighbor matching (e.g. Johnson et al. 1998) or more sophisticated optimization based method (e.g. Dixon and Wiener 1993). In addition, different heuristics for the correspondence matching can be applied. For example, some algorithms (e.g. Hering et al. 2004) assume that cells in consecutive frames overlap, which can be used for linking.

More general correspondence target algorithms described in Subsection 4.2.3 are seldom used for the convective cell tracking. Also general motion constraints **C1** – **C5** for the convective cells are rarely considered explicitly in the literature. In spite of the large variety of different algorithms, some general guidelines similar to the motion constraints **C1** – **C5** can be drawn from the algorithms:

1. *Proximity assumption.* Relatively small change in displacement. During the casual 5 min interval a cell is unable to move far from its origin. The maximum range can be estimated reasonably well by the maximum velocity v_{\max} , which can be obtained for instance by means of the grid-based motion estimation described in Subsection 4.3.1.
2. *Smooth motion assumption.* Since convection is a natural phenomenon, radical variations in the velocity or displacements are unusual. Displacements can be approximated by first-order velocities. Some algorithms (e.g. Hering et al. 2004; Novák and Kyzanrová 2006) assume that the object in the next frame can be predicted reasonably well with displacement velocity of the previous one. This is possible only if the motion is smooth enough.

3. *Common motion.* Convective cells tend to move along the general advection field and cells moving in various directions do not occur. Therefore, in tracking algorithms the cell velocity can be initiated reasonably well by using the velocity of neighboring cells or general advection field.
4. *Splits and merges.* In the real world convective cells split or merge frequently which should be considered in the algorithm. Therefore, one-to-one matching is not necessary and occlusions of convective cells is usually considered as splits or merges.

4.4 Life cycle analysis and cell temporal development

One of the main limiting factors of convective cell nowcasting is the prediction of cell initiation, development and dissipation. Several attempts have been made to solve this awkward problem. Usually only the short range development of a convective cell can be predicted while the comprehensive behavior of the cell is unknown. This is the major shortcoming of the contemporary convection nowcasting methods.

A common approach for the prediction of the storm intensity development is based on information extracted from several past radar echo scans. In 1981, Tsonis and Austin compared a nonlinear and linear intensity and size trending for convective cells through radar data and found that the linear trending achieved the best result. Additionally, their study showed that the trending of radar echo improved only very short range forecasts. This is a reasonable result, since each event has its own course of growth and decay, which makes the trending very difficult. According to Wilson et al. (1998), the study of Tsonis and Austin (1981) proved also that the essence of the physical process is not dictated by the past history of an echo development and more information of the event occurring in the boundary layer must be included in the prediction. It has been suggested that air convergence in the boundary layer is a key to the storm prediction (Wilson et al. 1998). By an appropriate convergence line detection method (e.g. Mueller et al. 2003), it is possible to predict the cell initiation; the thunderstorm is anticipated to initiate in the vicinity of Doppler-radar detected convergence boundaries.

Some of the radar based trending methods are in the operational use. The TITAN tracking algorithm makes a linear weighed trending based on previous radar scans. Another object-oriented tracking method SCIT (Storm Cell Identification and Tracking) (Johnson et al. 1998) gives information also on the cell attributes such as storm top, base and maximum reflectivity. SPROG (Spectral PROGnosis)(Seed 2003) methodology estimates the temporal evolution of precipitation by predicting different scales of radar image spectral components through an autoregressive AR(2)-model.

The trending based methods do not make any assumptions on the physical models of convection. As described in Section 2.1, convective cells often follow certain basic life cycle phases. The *knowledge-based modeling* incorporates prior knowledge of the convective cell life cycle properties into the prediction. Examples of knowledge-based nowcasting are given by Thielen et al. (2000) or Hand and Conway (1996).

Typical examples of operational knowledge-based models are GANDOLF (Pierce et al. 2000) of the U.K. Met Office and the Auto-Nowcast system (AN) (Mueller et al. 2003) by the Nation Centre for Atmospheric Research. In GANDOLF, satellite infrared and radar data are used in a conceptual life cycle model to analyze the cells in all life cycle stages. Likewise, AN incorporates radar and satellite information but exploits also convergence lines detected by a Doppler-radar based algorithm.

In addition to the described approaches above, different nowcasting methods utilizing numerical weather prediction have been proposed. Nevertheless, this topic is no more in the field of computer vision based nowcasting and hence it is not included in this thesis. An insight into the numerical weather prediction in nowcasting of thunderstorms is given for example in Wilson et al. (1998).

It is also worth noting that human made forecasts outperform many automated nowcasting methodologies (e.g. Wilson et al. 1998). However, the use of automated models and human made nowcasting are not mutually exclusive and different algorithms offer auxiliary tools for monitoring and predicting different weather phenomena. Considering this human oriented aspect, the models ought to provide as much information as possible on the monitored process while the definite decision can be made by the human expert.

Chapter 5: Applied methodologies

This chapter contains proposed methodologies that are utilized to achieve the objectives of this thesis. Firstly, we give an insight into the concept of data clustering, which plays an important role in the designed tracking algorithm. This is followed by a discussion on binary morphology techniques for the radar image preprocessing and convective cell identification. Then we introduce a novel clustering based algorithm for convective cell tracking and analysis. Finally, we elaborate on a fuzzy logic model for convective cell life cycle analysis.

5.1 Data Clustering

Data clustering is an important tool in statistical data analysis and widely applied in different fields of science. The goal in the clustering is to classify data into different groups such that every data point in each group has some properties in common while data points in separate groups are different in one way or the other.

In conventional pattern recognition methods, each class has a predefined classification such as “cat” “dog” or “mouse”. If we have, for example, 1000 sample points from each class, we may teach a methodology which can be applied in the recognition of an unknown sample. The approach is called *supervised learning*, that is, we utilize prior knowledge in the learning process.

However, if the supportive prior information is not available, we have to find different classes from the data itself. In the cluster analysis, the goal is to partition data and distinguish different classes based on the structure of the data.

In this thesis, data clustering has an important role in the tracking of convective cells. We focus especially on *density-based clustering* that is utilized in the development of the tracking method for convective cells in Subsection 5.3. The presented approach allows the simultaneous use of lightning data and weather radar in the tracking procedure. This entirely new aspect is important in the convective cell tracking and analysis; often in the convective cell nowcasting only radar data is considered while important information provided by lightning data is ignored. However, due to the several reasons, the radar data may be distorted which usually leads to failure of the tracking. For this reason, a more robust tracking is obtained through the radar and lightning data fusion.

5.1.1 Definitions and basic concepts of clustering

In order to cluster or classify data, we need a mathematical method to represent different data points. We assume that N features x_i are extracted from each data point \mathbf{x} , which can be represented in the vector form

$$\mathbf{x} = [x_1 \quad x_2 \quad \dots \quad x_N]^T. \quad (5.1)$$

This is called a *feature vector* of dimension N .

A *cluster* can be defined as a region in a feature space, where different data points are located relatively densely. Consequently, outside of the cluster relative point density is

sparse. Such a cluster is referred as a *natural cluster* (Theodoridis and Koutroumbas 2003).

There are also other ways to define the cluster. On the one hand, the definition of *natural cluster* is intuitively clear but also very subjective on the other. A more mathematical way to define a cluster is *m-clustering* (Theodoridis and Koutroumbas 2003), in which the data group $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$ is divided into m separate clusters. By the definition, all the data points belong to some cluster C_i , cluster set do not intersect and no cluster is empty:

1. $C_i \neq \emptyset, \quad i = 1, \dots, m,$
2. $\bigcup_{i=1}^m C_i = X,$ (5.2)
3. $C_i \cap C_j = \emptyset, \quad i \neq j, \quad i, j = 1, \dots, m.$

In addition, data points belonging to each cluster share some common properties. These properties are used to describe the similarity between the points and to find the optimal clustering. This naturally depends on the applied algorithm. This thesis considers the definition of the density-based notion (see Subsection 5.1.2).

The definition above is also called as *hard* or *crisp* clustering. An alternative definition for clustering is *fuzzy clustering* (e.g. Karray and De Silva 2004), where a data point may belong to different cluster sets at the same time with a certain degree of membership. The justification for fuzzy clustering is human oriented; in the real world human made reasoning is usually expressed with subjective degrees of likelihood rather than precise truth. Regardless of the subjective clarity of fuzzy clustering, it is not further considered in here. However, the concept of fuzzy logic is applied in this thesis, which is closely related to the fuzzy clustering (see Section 5.4). Note that clustering can be defined in different ways and a suitable approach depends on the application.

In the clustering scheme, we need a mathematical way to measure similarity or dissimilarity between data points. Obviously, a reasonable approach is to measure the the Euclidian distance between the points in the feature space. This is a frequently used method, but distance measure i.e. metrics can be defined by several other functions. Formally, metrics on a set X is a function $d: X \times X \rightarrow \mathbb{R}$, satisfying the following conditions:

1. Positive definiteness:

$$\exists d \in \mathbb{R} : 0 \leq d(\mathbf{x}, \mathbf{y}) < \infty, \forall \mathbf{x}, \mathbf{y} \in X, \quad (5.3)$$

$$d(\mathbf{x}, \mathbf{y}) = 0, \text{ if and only if } \mathbf{x} = \mathbf{y}, \forall \mathbf{x}, \mathbf{y} \in X, \quad (5.4)$$

2. Symmetry:

$$d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x}), \forall \mathbf{x}, \mathbf{y} \in X, \quad (5.5)$$

3. Triangle inequality:

$$d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z}), \forall \mathbf{x}, \mathbf{y}, \mathbf{z} \in X. \quad (5.6)$$

Perhaps the most common way is to define the distance function through the *Minkowski distance* or *p-norm*:

$$d_p(\mathbf{x}, \mathbf{y}) = \left| \sum_{i=1}^N (x_i - y_i)^p \right|^{\frac{1}{p}}. \quad (5.7)$$

In order to satisfy the conditions above, p must be greater than 1. If $p = 2$, we will obtain the well-known Euclidian distance and with $p = 1$ Manhattan distance. If p approaches to infinity, it can be shown that

$$d_\infty(\mathbf{x}, \mathbf{y}) = \lim_{p \rightarrow \infty} \left| \sum_{i=1}^N (x_i - y_i)^p \right|^{\frac{1}{p}} = \max_{1 \leq i \leq N} |x_i - y_i|. \quad (5.8)$$

In addition, it can be shown that d_∞ and d_1 can be viewed as overestimation or underestimation of d_2 that is, $d_\infty(\mathbf{x}, \mathbf{y}) \leq d_2(\mathbf{x}, \mathbf{y}) \leq d_1(\mathbf{x}, \mathbf{y})$.

In addition to the proximity measure between two data points, we need a formulation for measuring difference between two sets of points. Naturally, proximity measures for two data points can be regarded as a special case of set proximity. If C_i and C_j are two set of vectors, the set proximity can include for example:

1. The distance between set centroids \mathbf{m}_i and \mathbf{m}_j

$$d(C_i, C_j) = d(\mathbf{m}_i, \mathbf{m}_j) = \frac{1}{n} \sum_{\mathbf{x} \in C_i, \mathbf{y} \in C_j} d(\mathbf{x}, \mathbf{y}). \quad (5.9)$$

2. The min distance

$$d(C_i, C_j) = \min(d(\mathbf{x}, \mathbf{y})), \mathbf{x} \in C_i, \mathbf{y} \in C_j. \quad (5.10)$$

3. The max distance

$$d(C_i, C_j) = \max(d(\mathbf{x}, \mathbf{y})), \mathbf{x} \in C_i, \mathbf{y} \in C_j. \quad (5.11)$$

The type and shape of clusters is often dependant on the proximity definitions. Different distance measures favour for example compact or elongated clusters.

Many data processing applications have two significant problems: choosing correct features and high dimensionality of data. In a high dimensional feature space, any set of data points can be extremely sparse. In this case, a high number of feature vectors are needed to cover even a small part of the space. The problem is called as *curse of dimensionality* (e.g. Theodoridis and Koutroumbas 2003). Obviously, this has effect on the clustering as the complexity increases with the dimensionality. For this reason, only the most representative features should be considered.

The curse of dimensionality can be illustrated with an example. Consider a set of points distributed evenly on the one dimension interval $[0,1]$, which is further divided in the subintervals of width 0.1. If we require that at least 1 point falls into each subinterval, a set of at least 10 points is needed. However, if we have a two dimensional plane of size $[0,1] \times [0,1]$, which is further divided into subintervals of size 0.1×0.1 , a set of 100 points is required to fill each subinterval. Therefore, in order to retain the same density, the number of required points increases exponentially with dimension.

If the dimensionality is a problem, it is reasonable to reduce the number of dimensions with a suitable technique such as *principal component analysis* (PCA) (e.g. Jolliffe 2002). Through PCA, we are able to produce mutually uncorrelated features. Since information is incorporated into variations of different features, the features having smallest variances can be eliminated. However, the data used in here is low dimensional, and no techniques for dimension reduction are needed.

5.1.2 Density-based clustering

The main reason for our ability to recognize and discover clusters is that within each cluster density of points is considerably higher compared to the neighborhood (Ester et al. 1996). As pointed out in the previous sections, for humans it is usually not a difficult task to recognize dense areas from a two or a three dimensional figure. Actually, this way we defined the natural cluster in Subsection 5.1.1.

Density-based clustering relies on the notion of natural clusters. In this approach, clusters are discovered by the local density of data points. This is a difference compared to the conventional algorithms, which produce the clustering usually through an optimization technique. In addition, the conventional algorithms often need the number of clusters *a priori*.

Clustering algorithms are attractive with large databases. The following requirements for clustering come up especially with large spatial databases (Ester et al. 1996; Sander et al. 1998):

1. *Minimal prior knowledge on the domain of interest to determine input parameters.* Usually suitable values are not known in advance and hence a small number of input parameters is an asset.
2. *Capability to discover arbitrary shaped clusters.* Unlike many stochastic clusters, such as Gaussian distribution based clusters, spatial data clusters may have non-convex distribution.
3. *Good efficiency.* In large databases greedy inefficient algorithms are simply futile.

Unlike optimization-based algorithms, density-based approaches do not necessarily need prior information on the number of different clusters. On the other hand, density-based algorithms need some other parameters which define “density” applied in the clustering.

Density-based clustering provides also an approach to define and discover outliers; points that do not satisfy predefined density conditions are regarded as noise. This definition actually conflicts with the definition of the m -clustering, in which all the points are incorporated in some cluster (see Subsection 5.1.1). While the traditional algorithms try to fit all the data points in clusters, density-based approach deals also with noise. This is a reason why conventional algorithms, such as k -means (see e.g. Theodoridis and Koutroumbas 2003), are very sensitive to outliers.

DBSCAN

A common density-based clustering algorithm is DBSCAN (Density-Based Spatial Clustering of Applications with Noise). The algorithm has an important role in this thesis and it is utilized in the designed tracking method. The following discussion is mainly based on the original article by Ester et al. (1996).

The basic idea of DBSCAN is briefly as follows. Density is considered as the number of data points that fall in the neighborhood of a surrounding data point \mathbf{x} . To be more precise, we will consider a sphere $V_\varepsilon(\mathbf{x})$ centered at \mathbf{x} , where ε is the user defined radius of the sphere. In addition, we consider $N_\varepsilon(\mathbf{x})$ which defines the number of data points lying in $V_\varepsilon(\mathbf{x})$. The user needs to define also the parameter μ , which is the minimum number of points that $N_\varepsilon(\mathbf{x})$ must exceed in order for \mathbf{x} to be a *core point* of the cluster. A cluster can be defined as a set of points, where each point belongs to a neighborhood of at least one core point. In the following, we will go through definitions and more precise description of DBSCAN.

Given a set $X \in \mathbb{R}^N$ and two input parameters ε and μ , the algorithm defines the clustering by searching *maximal density connected* sets on X . The following definitions are used to form the correct clustering.

Definition 1: The ε -neighborhood of a data point

The ε -neighborhood of a point p is defined by a set

$$N_\varepsilon(p) = \{q \in X \mid d(p, q) \leq \varepsilon\}. \quad (5.12)$$

In words, the ε -neighborhood is a set of points that are at most ε apart from p .

Definition 2: Core point

Point p is a *core point* if

$$|N_\varepsilon(p)| \geq \mu. \quad (5.13)$$

By the definition, point p is a core point if the number of points in the ε -neighborhood exceeds μ .

Definition 3: Directly density reachable

Point p is *directly density reachable* from point q if $p \in N_\varepsilon(q)$ and q is a core point.

Definition 4: Density reachable

Given parameters ε and μ point p is *density reachable* from point q , if there is a chain of points $\{p_1, \dots, p_n\}$, $p_1 = q$, $p_n = p$ such that p_{i+1} is directly density reachable from p_i . This is illustrated in Fig. 7.a.

Definition 5: Border point

Point p is a *border point* if it is density reachable from a point $q \in X$ and it is not core point.

For a pair of core points Definition 3 and Definition 4 are symmetric and transitive relations. However, if another point is border point, symmetry does not hold.

Definition 6: Density connectivity

Point p is density connected to point q , if there is a point o such that both p and q are density reachable from point o . This is a symmetric relation. The idea of density connectivity is illustrated in Fig. 7.b.

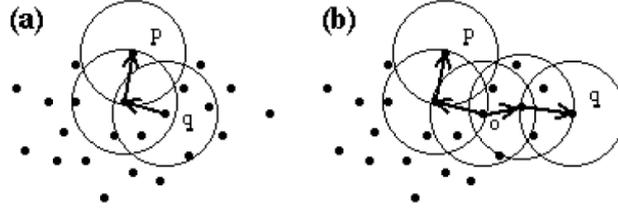


Fig. 7: a) Point p is density reachable from point q . However q is not density reachable from p , since q is a border point. b) Points p and q are density connected to each other via point o . This is a symmetric relation. (Ester et al. 1996)

Definition 7: Cluster

Given parameters ε and μ , a cluster is a set of points $C \subseteq X$ satisfying the following condition: $\forall p, q \in C$, point p is density connected to q .

Definition 8: Noise

Let $\{C_1, \dots, C_k\} \subseteq X$ be the clusters of the dataset X . Point p is *noise*, if $\{p \in X \mid \forall i: p \notin C_i\}$. In words, point p is regarded as noise, if it does not belong to any cluster.

The ability to discover noise is an important advantage over conventional algorithms, because they are often very sensitive to noise. Noise recognition can be also utilized for example in the data preprocessing phase by filtering noise out with DBSCAN.

The search for maximal density connected sets is a two phase method. At first, we search for a seed point p which satisfies the condition of core point. Secondly, we expand the cluster by searching all the points that are density connected to p .

Each point has one of the following states: *undefined*, *cluster point* or *noise*. In the beginning of the algorithm each point is initiated as undefined.

Algorithm 1: DBSCAN

Repeat until all the points in X are defined:

1. Choose an arbitrary point $\mathbf{x}_i \in X$
 - a. If \mathbf{x}_i is undefined and a core point
 - i. Make a new cluster $C_k = \{\mathbf{x}_i\}$
 - ii. Expand cluster by Definition 6. This is performed through Algorithm 2.
 - b. If \mathbf{x}_i is undefined but not a core point
 - i. \mathbf{x}_i is a noise point.

Algorithm 2: Expand cluster

Given a new cluster $C_k = \{\mathbf{x}_i\}$:

1. Expand the cluster with the ε -neighborhood of \mathbf{x}_i : $C_k = \{\mathbf{x}_i, N_\varepsilon(\mathbf{x}_i)\}$. Add $N_\varepsilon(\mathbf{x}_i)$ to an empty auxiliary set S .
2. Repeat until $S = \emptyset$
 - a. Remove the point \mathbf{x}_{i+1} from S
 - b. If point \mathbf{x}_{i+1} is a core point, expand cluster with the ε -neighborhood of \mathbf{x}_{i+1} : $\hat{C}_k = \{C_k, N_\varepsilon(\mathbf{x}_{i+1})\}$
 - c. Increase the set S with points $N_\varepsilon(\mathbf{x}_{i+1})$ that are not already in S .

The DBSCAN algorithm is also independent on the order that points are processed. This is a well-known problem for example in the k -means algorithm (Theodoridis and Koutroumbas 2003).

GDBSCAN

A natural application of density-based algorithms, such as DBSCAN, is the clustering of n -dimensional vector data in a metric space. Sander et al. modified DBSCAN in 1998 by generalizing the algorithm for arbitrary objects, which may include not only spatial but also non-spatial features. As an example, we may cluster arbitrarily shaped geographical regions with non-spatial features like average salaries, unemployment rate or population.

The density-based notion used in DBSCAN can be generalized in two important ways. First of all, we may represent the ε -neighborhood with any symmetric or reflective binary predicate. Secondly, instead of simply counting the number of objects in the ε -neighborhood, we may measure the “density” of the neighborhood by an appropriate weighting function. Hence, GDBSCAN (Generalized Density-Based Spatial Clustering of Applications with Noise) is simply obtained by refining Definition 1 and Definition 2 as follows:

Definition 9: $NPred$ -neighborhood of an object

Let X be a set of objects $\{x_1, \dots, x_n\}$ and $NPred$ a symmetric and reflective binary relation on X . For all $x_i, x_j \in X$ applies

1. Reflectivity: $NPred(x_i, x_i)$
2. Symmetry: $NPred(x_i, x_j) \rightarrow NPred(x_j, x_i)$

Thus, the $NPred$ -neighborhood of an object is defined as follows:

$$N_{NPred}(o) = \{o' \in X \mid NPred(o, o')\} \quad (5.14)$$

If we are clustering for example polygons, an appropriate relation could be “two polygons meet” or “intersect”. If we want to use the normal ε -neighborhood defined in DBSCAN for ordinary vector points, the $NPred$ -neighborhood is as follows:

$$N_{NPred}(o) = \{o' \in X \mid |o - o'| \leq \varepsilon\} \quad (5.15)$$

In order to generalize the DBSCAN algorithm we need also a function for measuring the density defined by the object that lies within the $NPred$ -neighborhood. In DBSCAN, the notion $|N_\varepsilon(p)|$ is used to measure the number of points lying in the vicinity of a point. In GDBSCAN, this notation is replaced by the *weighted cardinality* i.e. the “weight” of objects lying in the $NPred$ -neighborhood of the object o . For example, simply summing up non-spatial features of neighboring objects can be regarded as a weighted cardinality function $wCard(o)$.

Definition 10: Core object

Object o is a *core object* if

$$wCard(o) \geq minCard \quad (5.16)$$

In here $minCard$ is a threshold parameter in order for an object to be a core object. This is a generalized version of Definition 2 of DBSCAN where $|N_\varepsilon(o)|$ can be regarded as $wCard(o)$ and μ as $minCard$.

The rest of the definitions of DBSCAN can be generalized easily by substituting the definitions of ε -neighborhood and core point with the definitions of core object and $NPred$ -neighborhood. The GDBSCAN algorithm itself works similarly to DBSCAN by searching maximal density connected sets. Due the equivalency, the definitions of GDBSCAN are not further represented here. For more precise description of GDBSCAN see Sander et al. (1998).

Disadvantages of DBSCAN and GDBSCAN

The main disadvantage of the density-based clustering is the dilemma of *global density parameter*; due to the inflexible parameterization, it is assumed that the density in all the clusters is similar. For this reason, clusters with varying densities are not identified. As an example, stochastic point data from the Gaussian distribution have a varying density.

Another controversy is the efficiency. The algorithm complexity is of class $O(nq_n)$ where q_n is the runtime of the neighborhood query. However, through spatial indexing e.g. R^* -tree (Beckman et al. 1990) it is possible to reduce the runtime significantly and the complexity of class $O(n \log(n))$ can be achieved (Sander et al. 1998).

5.2 Convective cell identification

The general concepts of object identification in the object tracking scene was discussed in Subsection 4.2.1. In order to track convective cells, we need to identify them at first by using an appropriate data source. In this section, we introduce methods for identifying convective cells from radar and lightning data. Since two different data sources are used for the cell identification, we denote radar image based cells as *reflectivity cells* and lightning data based cells as *flash cells*.

5.2.1 Reflectivity cell identification

As pointed out in Subsection 4.3.2, convective cell identification is usually done through a simple radar image thresholding scheme, in which an expert selects suitable threshold parameters. In this thesis a radar reflectivity factor threshold of 40 dBZ in CAPPI 500 m images is considered as a distinguishing feature between convective and non-convective radar reflectivity factor. For example, non-convective stratiform precipitation rarely reaches values exceeding 40 dBZ (Steiner et al. 1995). An example of the thresholding is given in Fig. 8.

As we see in Chapter 6, in practical convective cell tracking scenario this threshold performs well. However, usually convective cells initiate at lower dBZ levels. Therefore, an adaptive identification of convective cells would be a significant improvement in the future prospects.

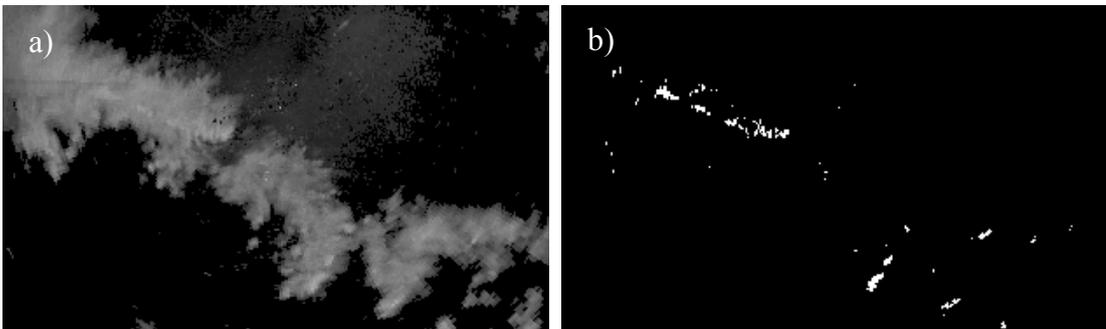


Fig. 8: Convective cell identification through a simple thresholding. a) Original radar image, b) Identified binary regions exceeding 40 dBZ.

5.2.2 Morphological preprocessing

A simple thresholding method is rarely sufficient for convective cell identification itself. The cells are often ragged and consist of multiple small pieces. Therefore, a preprocessing method is needed to unify thresholded areas. The GDBSCAN algorithm described in Subsection 5.1.2 is a manner of binding fragmented areas together as clusters. Another way is to use morphological image processing that is applied in this thesis.

In computer vision, morphology is an important method for extracting components that are useful in object description. It is also an important tool in image preprocessing, such as filtering, thinning and pruning. Mathematically morphology consists of set operations that are applied to the image with the user defined *structuring element*.

Morphology offers a powerful tool to numerous applications. Predominantly it can be used for the following purposes (Sonka 2007):

1. Image pre-processing, such as noise filtering and smoothing
2. Enhancing object structure, such as thinning, thickening and shape simplification.
3. Object segmentation from the background.
4. Calculating descriptive values of objects, such as area and perimeter.

In literature, morphology is presented as a tool of binary image processing, which can be further generalized for gray scale images. After segmenting convective cells from input image, we obtain binary regions that can be processed through morphology. For this reason, general gray scale morphological operations are not covered in here.

Morphological transform is defined by relation of the point set A (e.g. an image) with another small point set B , which is called as a structuring element and expressed with its local origin O (Sonka 2007). Applying morphological operation $\Psi(A)$ to the image A means that the structuring element B is moved systematically along the image. The structuring element B is centered on each pixel corresponding to the O and the operation $\Psi(A)$ is performed for each pixel in the neighborhood defined by B . Therefore, the operation is analogous to the well-known discrete convolution, in which the convolution mask is regarded as a structuring element of the operation. However, the convolution is a linear operation defined by its impulse response whereas morphology is a nonlinear set operation.

Dilation

Dilation is a morphological operation that makes binary objects in an input image thicker and studier. The structuring element defines the shape and width of the thickening layer. Mathematically the dilation on the image A by the structuring element B is defined as (Gonzalez and Woods 2001)

$$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \emptyset\}, \quad (5.17)$$

where \hat{B} is the *reflection* of set B

$$\hat{B} = \{w \mid w = -b, \text{ for } b \in B\} \quad (5.18)$$

and $(\hat{B})_c$ is the *translation* of set \hat{B} . For a given set S translation is as defined as follows

$$S_c = \{c \mid c = s + z, \text{ for } s \in S\}. \quad (5.19)$$

Fig. 9 illustrates the dilation. The input image A consists of two rectangular binary sets. A is dilated with a small rectangular structuring element B . In this case, B and its reflection are equal since B is symmetric with respect to its origin (marked with a black dot in Fig. 9). As shown, the two rectangles have enlarged and united together in the operation.

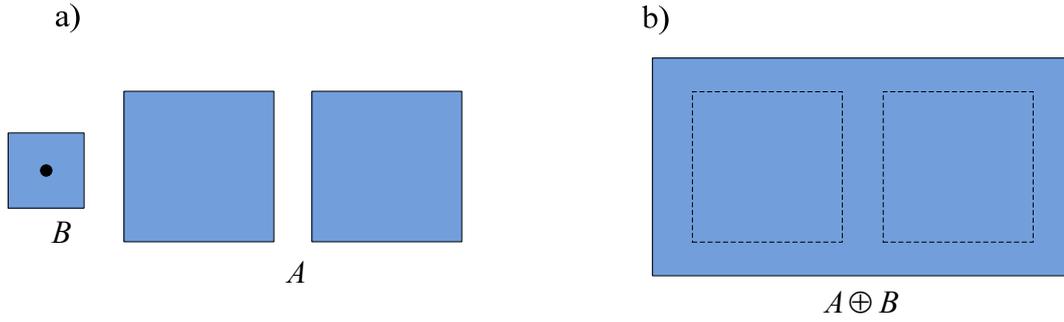


Fig. 9: An example of dilation. a) An input image A and a structuring element B . b) The resulting dilated image.

In the convective cell identification, the dilation can be used for attaching cells together. Through dilation we can attach small distinct pixels to larger entities, eliminate small holes and make ragged areas more uniform. This is also analogous to the GDBSCAN clustering since it can be used for clustering of arbitrary shaped objects such as binary regions.

However, the dilation alone is rarely suitable for the preprocessing of the segmented convective cells. As stated, it increases the size of cells as well as size of small outliers lying outside of larger entities. For this reason, more morphological operations are needed and therefore the concepts of erosion and closing are discussed next.

Erosion

Contrary to the dilation, *erosion* decreases the size of binary regions in the image. Formally, the erosion of the image A by the structuring element B is defined as follows (Gonzalez and Woods 2001)

$$A \ominus B = \{z \mid (\hat{B})_z \subseteq A\}. \quad (5.20)$$

This means that the erosion extracts the set of points z from A such that B , translated by z , is *fully* contained in A . In the border region of A , the structuring element B is only partially contained into A and hence the set A decreases.

It can be also shown that the erosion can be defined by means of the dilation as

$$A \ominus B = (A^c \oplus \hat{B})^c. \quad (5.21)$$

Intuitively, the concept of erosion is clear. In the erosion we shrink binary regions in the input set A , which can be also understood as an enlargement of the complement of A .

Fig. 10 illustrates the erosion operation. As the figure shows, the operation decreases the size of the two larger objects and even removes the smallest binary object.

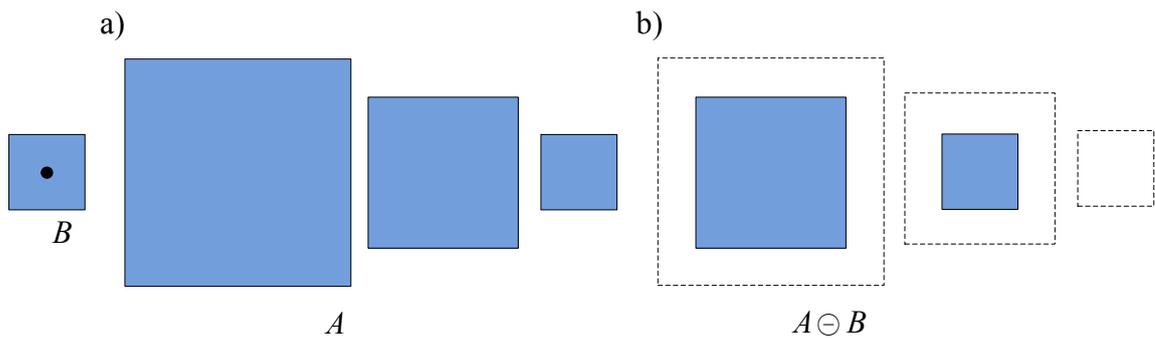


Fig. 10: a) The original image A and the structuring element B . b) The image after erosion.

Closing

Many morphological operations, such as *closing*, are based on dilation and erosion. Like dilation, closing tends to fuse narrow breaks and thin gulfs. Moreover, it fills small open holes and gaps but also smoothes the contour. However, it does not thicken the contour like the dilation.

Mathematically closing can be defined as the dilation which is followed by the erosion (Gonzalez and Woods 2001)

$$A \square B = (A \oplus B) \ominus B \quad (5.22)$$

The interpretation of the closing is simple. In the dilation part, we thicken the edges and fill caps of the contour. The erosion part shrinks and slightly refines the edges. However, since small holes and gaps are filled in the dilation part, erosion is not exposed on these areas anymore. Fig. 11 represents the closing operation applied to the convective binary regions.



Fig. 11: An example of closing. a) Segmented convective regions from input image before morphological preprocessing. The used morphological structuring element is shown in the upper right corner. b) Segmented regions after dilation. c) Segmented regions after closing, which is obtained by eroding (b).

In the convective cell identification, closing has an important status as it is directly applied to the unprocessed segmented convective regions. The user needs to define the shape and size of the structuring element, which can be done for example by a meteorological expert. Questions arise for example on the minimum distance between two segmented areas that are united together as well as on the desired smoothing effect. As an example, too large structuring element merges individual cells together unnecessarily. In this thesis a round disk having radius of approximately 5 km is used as the structuring element.

5.2.3 Reflectivity cell representation with polygons

Reflectivity cell identification results in segmented regions that can be used for the tracking. However, it is computationally more reasonable to use a suitable *object representation* method. As noted in Subsection 4.2.2, this can be, for example, a simple geometrical shape. In this thesis, polygons are used to approximate the reflectivity cells identified from the radar image.

A digital boundary can be approximated with an arbitrary accuracy by using a polygon. For a closed boundary, the approximation is exact when each point in the boundary can be defined with one of the polygonal segments. The goal is to capture the essential information from the segment with the fewest possible polygonal segments (Gonzalez and Woods 2001). This is generally a non-trivial problem, especially if we want to reduce number of border points significantly while retaining all the important information. Generally, different algorithms can be divided into heuristic and optimal approaches.

One of the simplest heuristic methods is the *minimum perimeter polygons* (Sklansky et al. 1972). The procedure is the best illustrated with an example adapted from Gonzalez and Woods (2001). Suppose that the boundary is represented with pixels as shown in Fig. 12. Assume also that the pixel boundary is defined by an enclosure of two walls, and the object boundary is a rubber band placed between the walls. If we let the rubber band to shrink, it will produce a minimum perimeter polygon.

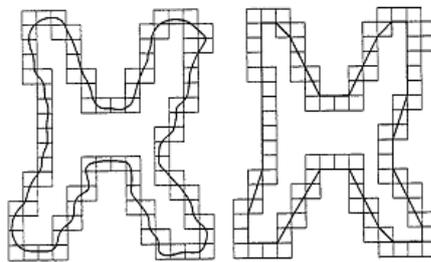


Fig. 12: Rubber band idea (Gonzalez and Woods, 2001).

Another heuristic approach for approximation is *merging* (i.e. Gonzalez and Woods 2001). In merging, all the pixels are initially used as approximating points, after which the algorithm drops out approximation points iteratively until certain conditions are satisfied. A reasonable choice is a point of which elimination will cause minimal increase in the approximation error. The error condition can be, for example, a user defined minimum error between the boundary and the approximation. The error can be defined for example as the sum of Euclidian distances d_2 (5.7) or as the maximum distance d_∞ (5.8).

In merging, the final approximation is chosen by dropping out points one by one. Another technique, called *splitting*, has an opposite approach to the merging (Douglas and Peucker 1973). The splitting technique is initiated with two points from the boundary, which can be for example two most distant points in the boundary. After this, a new point is added

such that error reduction is maximal. This is repeated until desired conditions are achieved.

Heuristic methods may be computationally efficient but lead to a suboptimal solution. More sophisticated methods try to find the optimal solution; the task is to optimize a given digital boundary with a polygon which yields to the minimum error with subject to the input constraints. The optimal solution can be achieved through *dynamic programming* (Bellman 1962). A typical algorithm that represents a dynamic programming solution as well as a suboptimal splitting solution for the digital boundary approximation problem is given by Sato (1992).

Several other methods for the digital boundary approximation have been discussed widely. An excellent overview of different methodologies is given for example by Kolesnikov (2003). In this thesis, reflectivity cell polygons are obtained by the minimum perimeter optimization. The minimum perimeter approach produces usually a very small error; the Euclidian distance error between the boundary and the polygon is at most $d\sqrt{2}$, where d is the minimum possible distance between different pixels (Gonzalez and Woods 2001). Practically, in convective cell identification and analysis this error can be considered negligible. In addition, the computational efficiency is not in the main role in this thesis and hence lossy approximation methods are not further discussed.

5.2.4 Flash cell identification

In addition to cells identified from the radar data, located lightning flashes provide data for convective cell identification. Lightning is known to correlate very well with high radar reflectivity factor values and it is a very clear indication of strong convection (e.g. MacGorman and Rust 1998).

Instead of using located lightning events itself, we produce *flash cells* by grouping spatially and temporally close lightning events into clusters. If we consider a time interval, say 5-15 minutes, and accumulate located lightning events, we will observe clusters formed by the plotted lightning points. As illustrated in Fig. 13, these flash cells are well correlated with high dBZ values of a radar image and correspond with the reflectivity cells identified from the radar data. In addition, the scope of the convective storm can be identified by the lightning locations, as the dimensions of the electrification are of the same size.

To search for these lightning clusters automatically, we apply the DBSCAN algorithm with the Euclidian distance measure. We choose DBSCAN since the areas containing high spatial density of lightning are of interest and the choice of critical density parameters can be done subjectively by an expert. As an example, an expert can require that the density of 1 lightning event per 5 km^2 must be satisfied in a flash cell during the 10 minute interval. Fig. 13 represents accumulated lightning points from a 10 minute interval and clustered the points with parameters $\varepsilon = 2.5 \text{ km}^2$, $\mu = 3$ flashes. Roughly speaking, we seek for the lightning clusters that exceed the density of $(\mu+1)/\pi\varepsilon^2 = 0.2037$

fl./km² during the 10 min interval. In practice, this parameter setting usually does well with intense cloud lightning data.

After the lightning data clustering is performed, some descriptive shape such as polygon or ellipse is fitted to the clusters. These shapes, calculated at time instants (t , $t+dt$, $t+2dt, \dots$), can be considered as counterparts of reflectivity cells and are used for the cell tracking in the similar way as reflectivity cell polygons.

In this thesis, we use fitted ellipses for representing flash cells. The use of ellipse is appealing, since located flashes in a flash cell seem to have a roundish, Gaussian-like distribution. In addition, choosing ellipse is computationally efficient, because the distance between two ellipses can be calculated analytically.

The ellipse fitting is done through PCA (see Subsection 5.1.1). A covariance matrix is estimated for the coordinates of located flashes in each cluster, after which we obtain ellipse main axes by calculating the principal component vectors of each covariance matrix. The length of each principal component vector equals the variance of the flash cell along the corresponding principal axis.

The flash cell detection provides also a possible approach for outlier detection of lightning data. The lightning points not belonging to any cluster do not satisfy the density conditions, and therefore they can be regarded as outliers (white crosses in Fig. 13). An outlier can be caused for example by locationing errors. However, in here the concept of outlier is defined in the sense of density-based flash cells. For instance, a ground flash may strike far from the actual core and the algorithm interprets the flash as an outlier.

Fig. 13 shows identified reflectivity cells and flash cells plotted on top of radar images. Note that the flash cells correspond well with the reflectivity cells. For this reason, both data sources represent primarily the same information and hence they are complementary to each other.

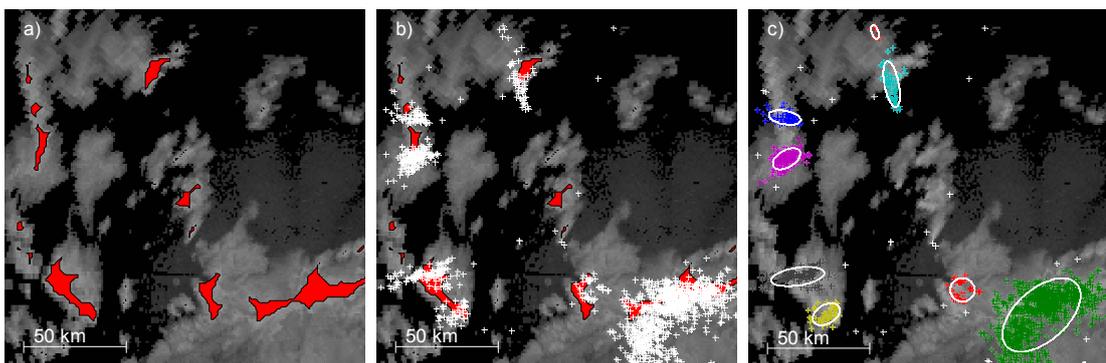


Fig. 13: An example of the reflectivity and flash cell identification. a) Identified reflectivity cells. b) Reflectivity cells with located lightning flashes. c) Identified flash cells and fitted PCA ellipses.

5.3 Convective cell tracking algorithm

Like in other object-oriented tracking algorithms (see Subsection 4.3.2), the aim of the designed tracking algorithm is to establish links between identified convection cells at

successive time instants. However, the presented methodology provides a novel way to track convective cells through different data sources, which consolidates the tracking. The basic idea of the algorithm is founded on the density-based GDBSCAN clustering algorithm introduced in Subsection 5.1.2 and a track is considered as a cluster with respect to the place and time.

5.3.1 Definition and basic concepts

Definition 11: Cluster

Cluster C_i is a set of objects o_p^k , where k is the time frame of p th object.

Definition 12: Cluster connectivity

Two clusters C_{i_1} and C_{i_n} are connected to each other if there is an ordered chain of clusters $\{C_{i_1}, \dots, C_{i_n}\}$ such that $C_{i_j} \cap C_{i_{j+1}} \neq \emptyset \forall j \in \{1, \dots, n-1\}$, and each cluster is connected to itself.

Definition 13: Track

A track T is a maximal set of clusters $C_{i_k}, k=1, \dots, n$, where each cluster $C_{i_l}, l=1, \dots, n$ is connected any other cluster $C_{i_m}, m=1, \dots, n$.

Definition 14: Track length

Track length is defined as the maximal time difference between different objects belonging to the same track

$$l_T = \max_k \{k \mid o_p^k \in C_i, C_i \in T\} - \min_k \{k \mid o_q^k \in C_j, C_j \in T\}. \quad (5.23)$$

As noted in Section 2.3, convective cells and especially MCSs tend to split and merge frequently which results in complex tracks. The track is called *complex* if a cluster C_j overlaps with more than one distinct cluster that include objects in the same time frame (see Fig. 15). However, sometimes splitting or merging does not occur. Therefore, if a track is not complex, it is called as a *simple* track. Dealing with splits and mergers in the designed tracking algorithm is discussed below in Subsection 5.3.5.

5.3.2 Tracking through the density-based clustering

Each cell is associated with three important features, which are utilized in the tracking: position, area and velocity. Also some additional data, such as mean or maximum dBZ value of a cell, can be calculated. However, this metadata does not have any effect on the tracking itself but it can be used for other purposes such as convective cell analysis.

In the proposed tracking approach, we apply the GDBSCAN algorithm, where the minimum Euclidian distance between two polygons is considered as the distance measure and polygon area as the cardinality function $wCard(o)$. The idea is that a cell identified at two or more consecutive time instants form a cluster with respect to the place and time as illustrated in Fig. 14.a. By clustering we search for the cells that are spatially and temporally close to each other. If a cluster contains cells that occur at two different instants, a track can be established.

However, it is often more convenient to reject the parameter time in the clustering. This is possible since the clustering is calculated only over a short time interval and hence spatially close cells are inevitably also temporally close to each other. In addition, the use of displacement approach presented below in Subsection 5.3.3 reduces the importance of time, since the state of the previous cells is predicted at the time t and projected into the same time plane.

Usually tracking algorithms use only two consecutive frames, e.g. algorithms by Dixon and Wiener (1993), Hering et al. (2004) and Novák and Kyzanrová (2006). However, this may lead to occasional errors in the tracking, especially in the presence of low quality data or occasional radar malfunctioning. To consolidate the tracking, our method allows the use of n_f consecutive frames in the tracking. The importance of multiple frames in the tracking is discussed in Chapter 6, in which we test the algorithm.

5.3.3 Improving the tracking algorithm with displacement velocity

We may improve the functioning of the algorithm by utilizing estimated cell velocities in the tracking. Before the actual clustering is performed, it is reasonable to transfer the antecedent cells at time $(t - dt, t - 2dt, \dots, t - n_f dt)$ along the direction of the cell velocity. This is defined as *displacement approach*. This way, we predict the state of the cell at time t and assume that a new cell emerges in the neighborhood of the predicted cells. A similar displacement approach is also used for example in reflectivity cell tracking algorithms by Hering et al. (2004) and Novák and Kyzanrová (2006). The idea of displacement is illustrated in Fig. 14.b, which corresponds Fig. 14.a but represents displaced objects. By comparing Fig. 14.a and Fig. 14.b, we observe that in Fig. 14.b successive cells overlap and form more distinct and clearly identifiable clusters.

The displacement velocity is calculated by the Euclidian distance between the polygon centroids. However, since the cells may have drastic moves, it is reasonable to smooth the measured velocities. A common approach is to utilize discrete linear and time invariant (LTI) filter of order $\max(m, n)$

$$y_k = b_0 x_k + b_1 x_{k-1} + \dots + b_n x_{k-n} + a_1 y_{k-1} + a_2 y_{k-2} + \dots + a_m y_{k-m}, \quad (5.24)$$

where $a_i, i = 1 \dots m$ are filter coefficients of the finite impulse response part of the filter and $b_j, j = 0 \dots n$ are filter coefficient of the infinite impulse response (IIR) part of the filter. Variable y_k is the filter output and x_k the measured value at the k th time instant. A more profound discussion on the linear time invariant discrete filters can be found for example in Mitra (2006). In here, a simple IIR filter of order 1 is used. In addition, we define *forgetting factor* γ , which is considered as a degree a belief describing how much we count on the current measurement. Therefore, the velocity of the i th track in the k th frame can be estimated by the filter of type

$$\hat{v}_k^i = \gamma \tilde{v}_k^i + (1 - \gamma) \hat{v}_{k-1}^i, \quad (5.25)$$

where $0 \leq \gamma \leq 1$, \tilde{v}_k^i refers to the measured velocity in the k th frame and \hat{v}_k^i to the estimated velocity in the k th frame.

With the used 5 min interval between radar images, this displacement approach is not necessary but often improves the result of the tracking, especially, if the cells are moving relatively fast. The importance of displacement is further discussed in Chapter 6.

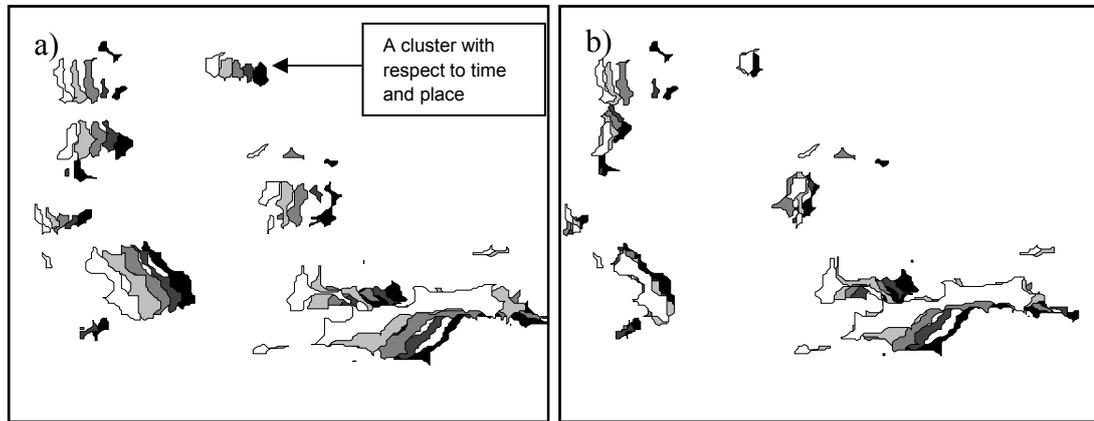


Fig. 14: Convective cell clusters with respect to spatial and temporal coordinates extracted from 5 consecutive image frames. Different shades of gray indicate different time instants such that the cells in current frame are white. a) Cells without displacement. b) Cells with displacement. Cell clusters are more compact, distinct and hence more identifiable compared to (a).

5.3.4 Velocity initiation

In order to use the displacement in the clustering, we need an estimate for the cell velocity. As described above, this can be done by calculating the distance between the object centroids in two frames. However, in the first frame or when the cell emerges, we do not have a predecessor object to give an estimate for the velocity.

Several approaches can be used to initiate the cell velocity. The first choice is to make an educated guess. We may have prior information, for example, on the precipitation movement or the wind direction that is steering the cells. In addition, the cells tend to move slowly relative to the applied 5 min interval between frames. As we will see in the Chapter 6, the algorithm produces usually satisfactory results even without the aid of displacement, that is, zero velocity is used. Therefore, the initial velocity should be close to zero, particularly, when no prior information is available.

Another possibility is to initiate the cell velocity according to movement of neighboring tracks. Convective cells tend to move along the general advection field and thereby more or less in the same direction. Cells moving in totally opposite directions are rarely met and for this reason other tracks may give a good hint of the cell velocity.

Also the use of the grid-based motion estimation methods presented in Subsection 4.3.1 are potential options. Often the cell track may contain abrupt movements and the use of neighboring cells may result in a spurious initiation. A good grid-based motion field is a robust and safe way to initiate the cell movement. This approach is used for example by Novák and Kyzanrová (2006).

5.3.5 Dealing with splitting and merging

Quite often two or more convective cells seem to merge into a single storm. Also quite often a single-cell storm split into two or more storms. Therefore, splitting and merging has to be considered in the tracking algorithm.

There are different ways to deal with the problem. In the contemporary tracking algorithms, the splitting and merging is usually treated through different heuristics. For example, the TRT algorithm by Hering et al. (2004) is based on overlapping of displaced objects in consecutive frames. In their algorithm, splitting and merging is considered if the relative overlapping of multiple objects is large enough. Novák and Kyzanrová (2006) search for possible splitting and merging if the similarity between linked cells is not high enough. Dixon and Wiener (1993) use combinatorial optimization to find one-to-one linking between the objects, after which the track is split or merged if the forecasted location of an unmatched cell falls in to the area of a track.

In here, we have used two different approaches for the problem. In the first approach, splitting and merging of convective cells is considered as splitting or merging of clusters. As an illustration, consider a case in Fig. 15 comprising of objects o_1 - o_7 included in clusters C_1 - C_5 . For the sake of simplicity, only two consecutive frames are used in the clustering, which means that each object is clustered twice. As described in Subsection 5.3.2, links are established between the objects belonging to the same cluster but having different time labels. For instance, o_2 belongs to the both clusters C_1 and C_2 , and therefore a one-to-one link is established between the objects o_1 and o_2 as well as between o_2 and o_3 . In the cluster C_3 we have a different situation. The objects o_4 and o_5 belong to the same cluster C_3 at t_3 , but in different clusters C_4 and C_5 at t_4 . Therefore, a split occurs and o_3 is linked to both o_4 and o_5 . The merging is analogous to the splitting: it is considered if objects belong to different cluster in the current frame but to the same cluster in the next frame.

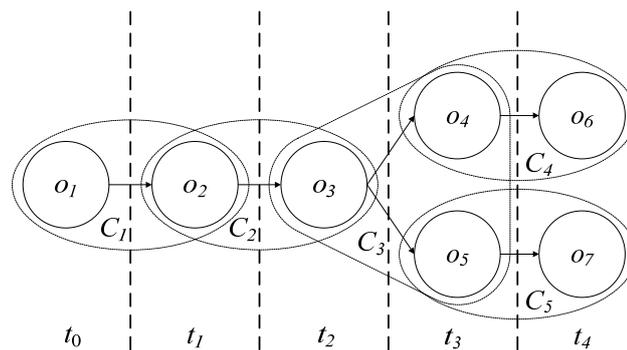


Fig. 15: An illustration of a track over the time $t_0 \dots t_4$, which comprises of objects o_1 - o_7 included in clusters C_1 - C_7 . The solid arrows represent track links between the objects and the dashed boundaries clusters. A split occurs at the frame t_3 , since objects o_4 and o_5 belong to the same cluster C_3 at t_3 but different cluster C_4 and C_5 at t_4 .

The second way to deal with the problem is to reject complex tracks and produce only simple tracks. There are a number of reasons for this. Firstly, it is pretty hard to distinguish different life cycle phases from a complex track. They have a complex graph-

like structure and therefore representing cell properties is difficult, for example, using a time series plot. Secondly, calculating descriptive cell statistics is very inconvenient for the complex tracks. Thirdly, splitting especially is often due to a new cell emerging in the neighborhood of the old cell. Therefore, it is reasonable to consider the case as a new track instead of a split. Finally, the actual life cycle of the cell is usually governed by the largest object and therefore the influence of splits and merges caused by smaller objects can be considered negligible.

Determining simple tracks in the case of object occlusion is not trivial. Consider again the example in Fig. 15, where the objects o_4 and o_5 belong to the same cluster C_3 at t_3 , but in different clusters C_4 and C_5 at t_4 . Since we want to create simple tracks now, we cannot split the track by linking the object o_3 to both o_4 and o_5 . We have to decide, which of the objects are linked together, so that only one-to-one links between objects in different frames are allowed. Because the task is to find the optimal one-to-one matching between a two set of objects, the problem can be viewed as the assignment problem defined earlier in the Subsection 4.2.3.

For the cost required in the assignment problem (4.1), we adopt the function used by Dixon and Wiener (1993) in the TITAN tracking algorithm. They use the cost function for three dimensional ellipsoids but in here we modify it for two dimensional polygonal objects as

$$c_{ij} = w_1 d_{cent} + w_2 d_A, \quad (5.26)$$

where d_{cent} measures the Euclidian distance between polygon centroids and d_A is the difference between polygon areas defined by

$$d_A = |A_i^{1/2} - A_j^{1/2}|, \quad (5.27)$$

where A_i and A_j are the areas of the i th and j th polygon. Coefficients w_1 and w_2 are weights and can be set for example to 1 (Dixon and Wiener 1993). As stated in Subsection 4.2.3, a very common approach is to solve the optimization problem through the Hungarian algorithm. However, in this case the number of objects is usually very small and therefore a greedy search is possible. In addition, we may use a suboptimal solution by linking the objects having the smallest cost directly. As the number of objects is small, our experiments suggest that the suboptimal approach produces good results and even the optimality is achieved frequently.

5.3.6 Utilizing lightning data in the tracking

The same tracking algorithm can be used for the reflectivity cell tracking as well as for the flash cell tracking. If the algorithm is used for the flash cell tracking, we track fitted flash cell ellipses calculated at $(t, t+dt, t+2dt, \dots)$. However, if we are dealing only with lightning data, the time difference between instants can be chosen arbitrarily as lightning data is independent on the fixed 5 min interval.

In order to reduce errors in the radar data, we can combine these two data types and track them simultaneously. The purpose of this is to improve the continuity of the tracking. Due to the attenuation, reflectivity cells may disappear and the track is lost if we rely only on the radar data. The use of flash cells in the tracking is illustrated in Subsection 6.1.5.

One possible approach for this data fusion is to simply combine flash cells with the reflectivity cells and perform the tracking. This method guarantees the continuity but the outcome of the tracking can be disordered. Widespread lightning may lead to an excessive merging of cell tracks. Another way is to emphasize the radar data. Because radar data is more precise, lightning data is taken into account if and only if the reflectivity cell based track is lost. Then the algorithm is less vulnerable to occasional errors in the radar data, and widespread flash cells do not interfere with the tracking too much.

5.4 Fuzzy logic modeling of the cell development

The term *soft computing* concerns theories such as neural networks, probabilistic reasoning, evolutionary computing and especially *fuzzy logic*. In contrast to conventional hard logic, the role model for fuzzy logic is the human mind (Zadeh 1968); in the real world human made reasoning is usually expressed with subjective degrees of likelihood rather than precise truth (see also the definition of fuzzy clustering in Subsection 5.1.1). This stems from the fact that human beings tend to analyze the world with terms “imprecise”, “vague” and “ambiguous”. The complex world seems to be very difficult to categorize and define.

In the conventional “crisp” computing, we encounter uncertainty and ambiguity constantly, and therefore modeling phenomena through crisp definitions and rules is difficult without any loss of generality. Hence, the crisp computational models can be described with the adjectives such as inflexible, clumsy and stupid whereas natural creatures like us are flexible, adaptive and clever (Koivo 2000). Fuzzy logic is a methodology that tries to overcome the problems encountered in the crisp computing by adding some tolerance of imprecision and uncertainty into the models. The discipline of fuzzy logics is a broad theory including for example fuzzy set theory, fuzzy logic and fuzzy measuring. It is also widely applied in the atmospheric sciences as the studied phenomena are usually highly complex and nonlinear. An extensive review of different soft computing techniques applied to the atmospheric phenomena is given by Chattopadyay (2006).

Fuzzy logic is tempting when data can be expressed with linguistic variables such “hot”, “cold” or “warm”, which can be further described with additive terms such as “not at all”, “more or less” or “very”. In this section, we build up a model for convective cell development by mimicking human made expert reasoning. Fuzzy logic fits well to the problem, since the development of convective cells is highly dependent on the experts and it can be well described with linguistic terms like “intensify” or “dissipate”. As an example, a meteorologist would be a good reference to give an estimate of a thunderstorm

state and course of development. This expertise can be included into the fuzzy logic model to automate the expert statements.

The fuzzy logic model in this section utilizes the data acquired from the tracking algorithm introduced in Subsection 5.3. The primary aim is to test the use of fuzzy logic in the convective cell nowcasting and to provide discussion on fuzzy logics as a considerable choice for the future prospect. Therefore, this thesis includes only a very brief insight into fuzzy logic and reasoning. For more extensive coverage, see e.g. Koivo (2000), Karray and De Silva (2004), Berthold and Hand (1999).

5.4.1 Basic concepts and definitions

In the classical set theory an object either belongs or does not belong to a set A . Hence the set A forms a crisp boundary between the set and the universe X , which can be described with the characteristic *membership function* u formulated as

$$u_A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A. \end{cases} \quad (5.28)$$

An alternative definition for the crisp set theory is *fuzzy set theory*, where a point x may belong to different sets with a certain degree of membership. Formally, the point x is associated with a set A with the membership level of $u_A(x)$ as

$$u_A : X \rightarrow [0,1], j = 1, \dots, m. \quad (5.29)$$

If the value of $u_A(x)$ is close to the zero, the degree of membership in the set A is weak. Conversely, if $u_A(x)$ is close or equals one, the level of membership of the point x in the set A is strong. Therefore, the fuzzy set theory can be viewed as an extension of the conventional set theory.

In the classical set theory, union and intersection (AND- and OR-connectives or logical conjunction and disjunction) are one of the fundamental set operations. In the case of fuzzy set theory, we may define these operations through several approaches. A popular way is to define the fuzzy intersection and union through *min-norm* and *max-norm*:

$$\begin{aligned} u_{A \cap B}(x) &= \min(u_A(x), u_B(x)), \\ u_{A \cup B}(x) &= \max(u_A(x), u_B(x)). \end{aligned} \quad (5.30)$$

Another classical choice for these set operations is *product* and *bounded sum* (Berthold and Hand 1999)

$$\begin{aligned} u_{A \cap B}(x) &= u_A(x) \cdot u_B(x), \\ u_{A \cup B}(x) &= u_A(x) + u_B(x) - u_A(x) \cdot u_B(x). \end{aligned} \quad (5.31)$$

The output of the fuzzy intersection or union defines the degree of membership of x in a fuzzy set resulting from the fuzzy set operation. For example, consider fuzzy sets $A =$ “young” and $B =$ “reckless”. If a person x is young with the degree of membership

$u_A(x) = 0.6$ and reckless with $u_B(x) = 0.7$, then by using the min-norm (5.30) the degree of membership of x being young *and* reckless is $u_{A \cap B}(x) = 0.6$.

In addition to the definitions (5.30) and (5.31) above, we may use several other ways to define the union and the intersection (see e.g. Karray and De Silva 2004). However, all the different definitions of union and intersection generalize in *T-norm* and *S-norm* respectively. It can be also shown that the min-norm is the largest possible T-norm and the max-norm the smallest possible S-norm (Karray and De Silva 2004). For the properties of the S-norm and the T-norm, see e.g Karray and De Silva (2004) or Koivo (2000).

5.4.2 Fuzzy inference

The original purpose of fuzzy logic was to model human thinking and reasoning through *fuzzy rules*. These rules can be obtained principally by the following approaches (Koivo 2000):

1. Human experts provide rules. This is especially appealing in the presence complex systems, such as convective systems, which can be analyzed by linguistic terms and concepts. Suitable expert knowledge has to be available for this approach.
2. Data driven methods for learning rules from data. This may include the use of neural networks (e.g. Karray and De Silva 2004; Bishop 1995) or specialized fuzzy learning algorithms (e.g. Berthold and Hand 1999). However, this approach is out of our focus and thus it is not covered in here.
3. A composite method by combining 1. and 2. in an appropriate way.

The first rule forming approach plays an important role in the development of the fuzzy logic model in this thesis. The other methods are out of our focus and hence they are not further covered in here.

How can we use fuzzy sets for reasoning? In classical logic, we conduct reasoning through a simple argument form called *Modus Ponens*:

Premise	x is A
Implication	If x is A , then y is B
Consequent	y is B

In the fuzzy reasoning, we use *Generalized Modus Ponens*, that is, A and B are fuzzy sets, and x and y are symbolic names, such as linguistic words defining the object (Koivo 2000). In order to generalize the Modus Ponens for fuzzy sets, we need to define *fuzzy implication*. Let A be a fuzzy set of n input variables x_1, x_2, \dots, x_n in the universe X^n and the B a fuzzy set of the output variable y in the universe Y . A fuzzy implication $A \rightarrow B$ is a relation in $X^n \times Y$, which can be understood as a fuzzy IF-THEN rule and results in a fuzzy set B' . A very common way is to interpret the fuzzy implication through the *min-operation*

$$u_{B'}(\mathbf{x}, y) = u_{A \rightarrow B}(\mathbf{x}, y) = \min(u_A(\mathbf{x}), u_B(y)), \mathbf{x} \in X^n, y \in Y, \quad (5.32)$$

where $u_A(\mathbf{x})$ is the membership function of $u_A(x_1), u_A(x_2) \dots u_A(x_n)$ connected through AND- and OR-connectives. The min-operation is applied as a fuzzy implication in this thesis. Note that the min-rule is somewhat unintuitive because the logical implication is not a symmetric relation in general. However, in practice this approach provides usually good, robust results and it is easy to compute (Karray and De Silva 2004). In addition to the min-rule, there exist several other approaches for the fuzzy implication (see e.g Koivo 2000; Karray and De Silva 2004)

A fuzzy logic system usually contains several rules. If we have M rules in the system, we need a consistent method for using different fuzzy sets $B'_i, i = 1 \dots M$ resulting from the different rules. The output of the system can be represented either as M distinct fuzzy sets (Koivo 2000) or by combining the different sets adequately. The latter one is usually done by connecting the set B_i through OR-connectives i.e. the fuzzy union. A common way is to apply the max-norm as the fuzzy union. Thus, the overall membership function of the complete rule base can be formulated as follows

$$u_{A \rightarrow B}(\mathbf{x}, y) = \max_i (u_{A \rightarrow B_i}(\mathbf{x}, y)) = \max_i (\min(u_{A_i}(\mathbf{x}), u_{B_i}(y))), \mathbf{x} \in X^n, y \in Y \quad (5.33)$$

To illustrate the functioning of a multi-rule system, consider the following example adapted from Koivo (2000) and Karray and De Silva (2004). We want to model a phenomenon through two input variables x_1 and x_2 , which both can be described with two fuzzy sets A_{11}, A_{12} and A_{21}, A_{22} respectively. The fuzzy logic model consists of the following two rules

1. **IF** (A_{11} and A_{12}) **THEN** B_1
2. **IF** (A_{21} and A_{22}) **THEN** B_2 .

The functioning of the model is represented in Fig. 16.

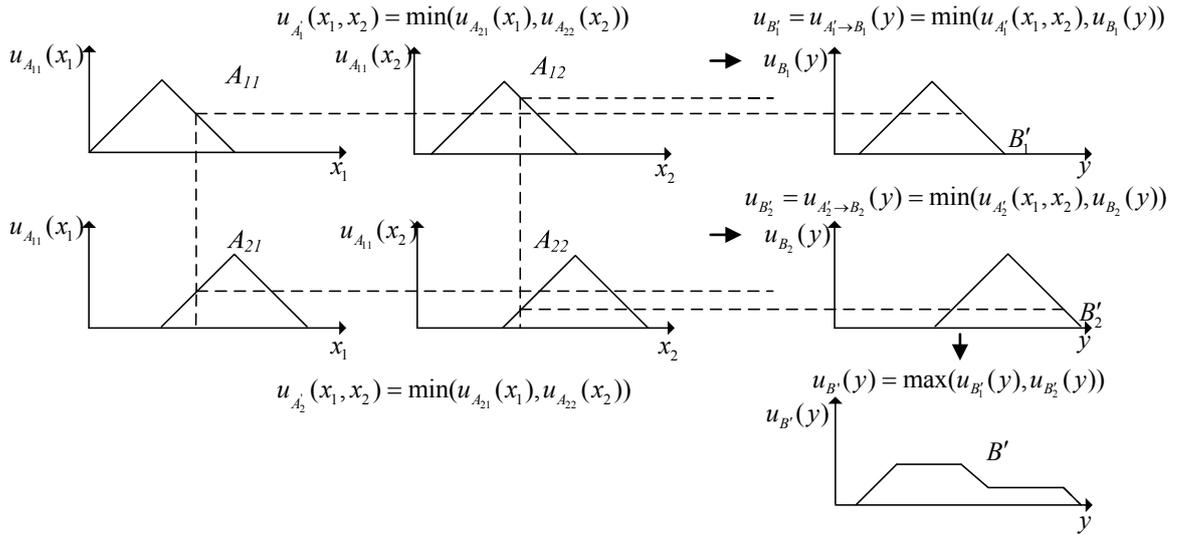


Fig. 16: An illustration of fuzzy logic decision making. The fuzzy AND-connective is processed through the min-operation. The fuzzy set resulting from the rules 1 and 2 are combined by the max-operation.

5.4.3 Fuzzification

Before we can build up a fuzzy model, we need a method for describing the model parameters with fuzzy sets. The concept called *fuzzification* is a process where crisp quantities are converted into fuzzy sets. As the aim is to model the human thinking, intuition is often applied. In this thesis, we trust in our intuition and apply it as the fuzzification mechanism. In addition, there exist several algorithms or logical procedures for the fuzzification. For other approaches, see e.g. Karray and De Silva (2004).

5.4.4 Defuzzification

In many applications a fuzzy set itself is not a suitable output. For example, an air traffic control gives forewarnings about intense storms exceeding a certain threshold of intensity. Hence, in order to give a warning, the air traffic control probably wants know a crisp value of the storm intensity rather than a fuzzy set. Therefore, the fuzzy set resulting from the fuzzy inference is mapped to a crisp value through an operation called *defuzzification*.

Intuitively very appealing method for defuzzification is *centre of gravity defuzzification*, that is, we calculate the centre of gravity of the output fuzzy sets B'_i . Another choice is *middle of maximum*, which calculates the centre of output maximum. For other defuzzification methods, see e.g. Koivo (2000). An extensive discussion on defuzzification theory and different methods is given in Leekwijck and Kerre (1999).

Note that the concept of defuzzification is also analogous to some probabilistic methods. For example, in the Bayesian inference the aim is to describe the model output as a probability distribution. Like a fuzzy set, the whole probability distribution itself is not always a good output and therefore some crisp descriptive value, such as the expected value of the distribution, is preferred. This can be viewed as the corresponding operation to the defuzzification.

5.4.5 Designed fuzzy model for the life cycle analysis of convective cells

A fuzzy logic based expert system was designed for the convective cell time series data acquired from the tracking algorithm. The primary objective of the system is to model and detect convective cell life cycle phases through an expert oriented approach. This aspect results from the fact that usually a human expert is able to perceive easily different life cycle stages from lightning and radar parameters. Therefore, mimicking this expertise is a reasonable basis for the system.

The second objective of the model is to simplify and extract necessary information for the end-user applications. Since the tracking algorithm itself produces several information outputs, such as lightning and different radar parameters, single informative and easily understandable parameter describing the stage of a cell would be an asset.

The following criteria were considered in development of the fuzzy logic system. The exact documentation of the model including all the fuzzy rules and membership functions is given in Appendix A.

1. *Cell area change ratio.* Cell area has an important role in the life cycle modeling. As we will see later in the Chapter 6, a typical convective cell reaches its maximum area at the midpoint of the life cycle. This is followed by the decay phase and the decreasing of the cell area. In order to give an estimate of relative decreasing of cell area, we will define the parameter cell area change ratio as

$$r_{A_{\max}}(i) = \frac{A(i)}{\max_{j=1 \dots i}(A(j))} \quad (5.34)$$

Cell area change ratio lies in the interval [0,1] and describes the development. In the intensifying phase, the cell area is increasing and the rate is equal or close to 1. After the cell has attained its maximum area, the area starts to decrease and the rate surges. Therefore, small values of $r_{A_{\max}}$ can be interpreted as the decay of a cell.

2. *EchoTop 20 dBZ average difference.* This parameter is obtained as follows. At first, we calculate the cell maximum EchoTop 20 dBZ change during a user defined time interval, after which the change is divided by the duration of the interval. Large positive values of the parameter stand for cell intensification as the cell intrudes into the higher altitude levels. Conversely, large negative values indicate dissipation. In here, the interval of 20 min is considered. If a shorter interval than 20 min is used, noisy behavior can outweigh the actual course of development. Larger intervals, on the other hand, are conservative and uninformative as they lose the necessary information included in rapid development changes.
3. *Duration of EchoTop continuous 20 dBZ increasing/decreasing.* This parameter defines the duration of the continuous increase or decrease of the cell maximum

EchoTop 20 dBZ. If the growth is long and continuous, we can infer that the growth is not random and results from storm intensification. The opposite conclusions can be drawn from the duration of EchoTop 20 dBZ decreasing.

4. *Beginning of lightning.* The first lightning stroke is often a clear indication of the storm intensification. The first flash appears usually before the cell has attained the maximum EchoTop 20 dBZ value, which implies that the cell is intensifying (see Subsection 6.3.4).
5. *Duration of continuous increasing/decreasing of lightning.* Like the duration of EchoTop 20 dBZ, also continuous growth/decrease of lightning is a sign of intensification/decay.

The model output is also filtered by first-order IIR-filter (see Subsection 5.3.4) of type

$$\hat{c}_d^i = \gamma \tilde{c}_d^i + (1 - \gamma) \hat{c}_d^{i-1}, \quad (5.35)$$

where \tilde{c}_d^i refers to the defuzzified model output and \hat{c}_d^i to the filtered model output at the i th time step. Centre of gravity was applied as the defuzzification mechanism.

Parameters describing maximum CAPPI 500 m values within the cells were not considered in the fuzzy model. Usually CAPPI 500 m values do not provide any additional information to EchoTop 20 dBZ, which seems describe the cell development better. However, CAPPI 500 m information is included into the cell maximum area change ratio, as the cells itself are defined through the CAPPI 500 m data.

The use lightning frequency absolute changes was also discarded. According to our observations, the intensity of lightning in cell may change abruptly and drastically within a small time interval. For example, in the intensifying phase 5 min lightning frequency increment may be of couple of strokes or of hundreds of strokes. Therefore, it is difficult to deduce life cycle phases based on the intensity of lightning. An appropriate scaling method, e.g. logarithmic scaling (Yeung et al. 2007), could ease the problem.

As discussed above, all of the available information is not included in the fuzzy analysis model. The use of all the possible data sources is not always an advantage. The principle called *Occam's razor* (named after the 14th –century English logician and Franciscan friar William of Ockham) can be used as an objection. Wikipedia states the principle simply as follows (Wikipedia 2008b):

The explanation of any phenomenon should make as few assumptions as possible, eliminating those that make no difference in the observable predictions of the explanatory hypothesis or theory.

In the modeling scenario this can interpreted as: among equally good models, we should choose the one with fewest assumptions. Hence, we should not include any additional

assumption into the model if it does not significantly improve the model performance. For example, the rejection of CAPPI 500 m data is justified by this principle.

In addition, a complex fuzzy model suffers also from the curse of dimensionality defined in Subsection 5.1.1. In the presence of multiple rules, we have to cover the whole input-output space with rules. In other word, every input-output combination has to be considered by some rule. This is can be complicated if the dimensionality of the input space is high.

Chapter 6: Convective cell tracking and life cycle analysis

In this thesis, two important data sources were considered: FMI Vantaa C-band radar and lightning location system FMI LLS. CAPPI 500 m and EchoTop 20 dBZ were used as the radar derived parameters. Lightning data consisted of located CG and CC flashes.

Tracking tests in Section 6.1 were carried out with reflectivity cells identified by CAPPI 500 m images. Also flash cells identified by CG and CC lightning flashes were applied to the tracking task.

Convective cell statistical and life cycle analysis in Section 6.3 was conducted with both radar derived data types and CG flash data.

6.1 Visual performance of the tracking

The tracking method was tested in several case studies. The method was validated visually, since it is difficult to test the performance of the algorithm automatically and usually a human observer offers the best reference to the algorithm. The use of human reference is justified, because the main goal of the computer vision based tracking algorithm is to provide an alternative choice for the human observer. The following discussion views a few examples illustrating the functioning and performance of the algorithm.

6.1.1 Early tests with the tracking algorithm – a case study on Aug 27th 2006

The first tracking tests were carried out by tracking radar based reflectivity cells with data acquired on Aug 27th 2006. On that day, the southern Finland and Estonia encountered heavy convective cells accompanied with intense showers and lightning. More than 1,600 ground flashes and 2,200 cloud discharges were recorded during that day.

Fig. 17 represents the first tracking example, in which a set of convective cells are tracked in the province of Proper Finland (located in the southwestern part of Finland). The tracking was performed over two hours between 11:00 – 13:00 UTC (local time is +3 hrs UTC).

In order to improve the matching between the candidate objects, the displacement was considered in the tracking. However, the forgetting factor γ in (5.25) was set to 1, i.e. the measured displacement velocity vector was not filtered with the previous estimates. Nevertheless, as Fig. 17 shows, the tracking is very consistent. The number of consecutive frames n_f used to find object correspondences was 3.

Parameter *minCard* was set to 10 pixels², which corresponds roughly to the area of 2.5 km². This is a rather small area considering the dimension of a usual convective cell. However, this area describes only roughly the cell area exceeding the threshold of 40 dBZ, that is, only the most intense core of the cell, while the actual cell area is naturally much larger. By Definition 9 in Subsection 5.1.2, the GDBSCAN algorithm rejects smaller objects than *minCard* if they do not fulfill the density conditions. Therefore, small individual objects are removed from the tracks as outliers.

In the test, the parameter $NPred$ was set to 2 pixels (around 1 km), which is also a quite small value. By the definition, $NPred$ determines the radius of the density-based neighborhood. Since too large value of $NPred$ usually causes unnecessary and incorrect merging of cells, a small value of $NPred$ is preferred. Besides, as the displacement approach is used, consecutive displaced objects in successive ought to be close to each other.

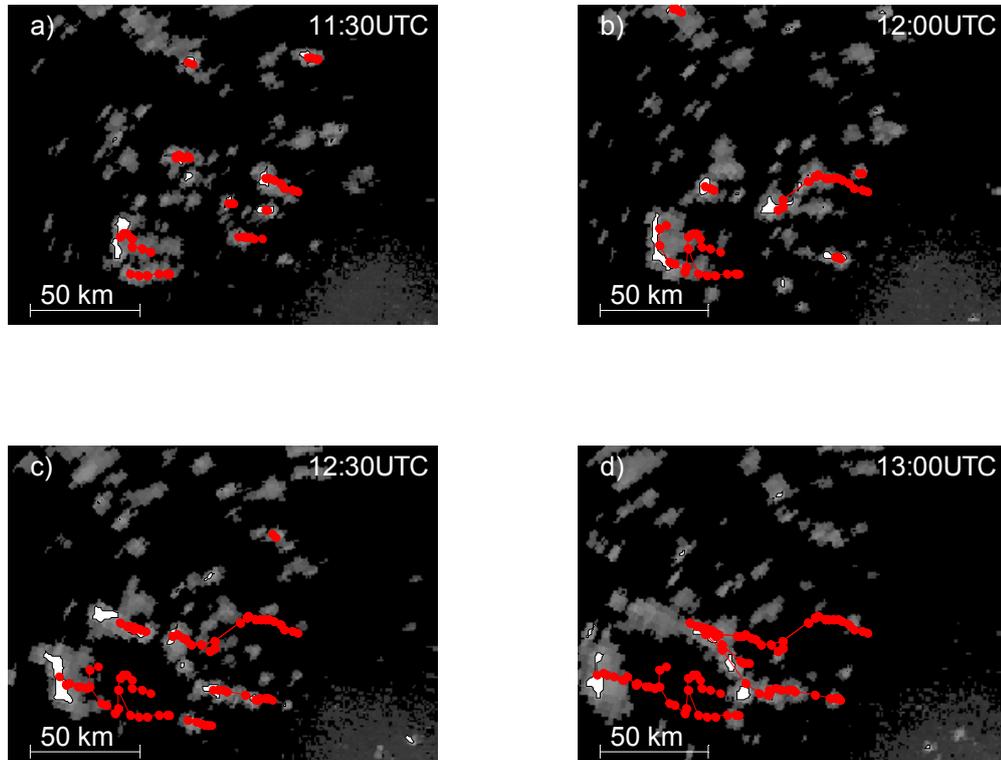


Fig. 17: Tracking example on Aug 27th 2006. A set of convective cells in the western Finland was tracked over two hours. Tracks are marked with red dotted lines in the images. Generally, the tracking performs well and it is consistent with the visual perception. The applied Vantaa radar is not in the image area.

6.1.2 Understanding the parameters used in the clustering

Next, we will study the effect of different parameter values on the clustering through a simple example. The algorithm parameterization was introduced in Subsection 5.1.2 and in Section 5.3. Fig. 18.a depicts convective cell polygons (blue areas) extracted from two consecutive radar image frames and it shows at least three identifiable convective cell clusters.

If the parameter setting is $NPred = 2.5$ km, $minCard = 2$ km², we obtain a good result; all the cluster correspond well to the visual perception in Fig. 18.b. If the parameter $NPred$ is increased, clusters start to merge. The parameter $NPred$ defines object neighborhood and therefore more distant objects are included in the same cluster as the size of the neighborhood increases. In Fig. 18.c, we have increased the parameter $NPred$ to 7 km. The two topmost and clearly distinct clusters have merged, which is apparently an incorrect result.

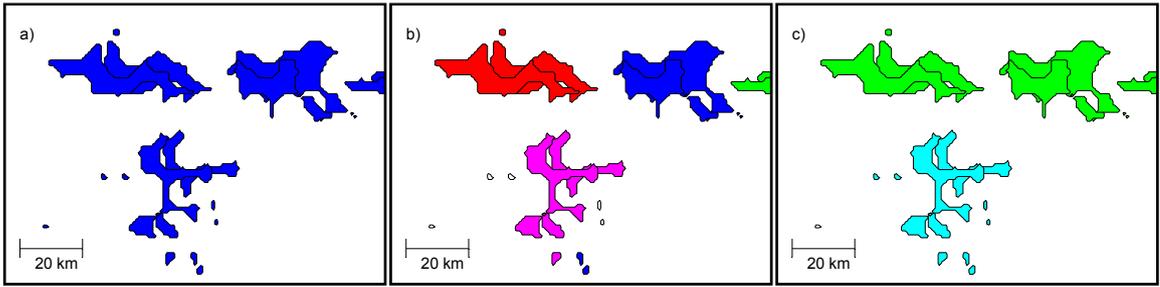


Fig. 18: The meaning of the parameter $NPred$. a) Original data b) Correct clustering with $NPred = 2.5$ km, $minCard = 7$ km². c) Incorrect clustering $NPred = 7$ km, $minCard = 7$ km². Clusters join because the parameter $NPred$ is too large. Small white polygons are identified as outliers.

Also the parameter $minCard$ has effect on the clustering. The parameter defines the minimum amount of weight (e.g. polygon mass) in the neighborhood of an object in order for the object to achieve the status of a core object and consequently to form a cluster. For example, an excessive value of $minCard$ results in an empty clustering as all the objects are regarded as noise. For this reason, the parameter should correspond with the natural dimension of convective cells.

However, an arbitrarily small $minCard$ is not reasonable. There are two primary reasons for this. At first, even the smallest individual and distant objects are regarded as core objects and consequently as clusters. As stated earlier, objects should be included in cluster if and only if the cardinality of the neighborhood is high enough. Linking small objects into clusters is justified if they lie in the proximity of larger entities i.e. they are classified as border objects.

Secondly, a small value of the parameter may lead to the *single link effect*, that is, a small object may link larger entities together (e.g Theodoridis and Koutroumbas 2003). This phenomenon is illustrated with an example. In Fig. 19.a, we have added three small artificial polygons to the original data. With the parameter values $NPred = 2.5$ km, $minCard = 7$ km² the result is still good. If $minCard$ is decreased to 2 km², the single link effect occurs by merging two large and clearly distinct clusters together. Due to the small value of $NPred$, even the smallest objects are considered as core objects and hence they link larger entities unnecessarily together.

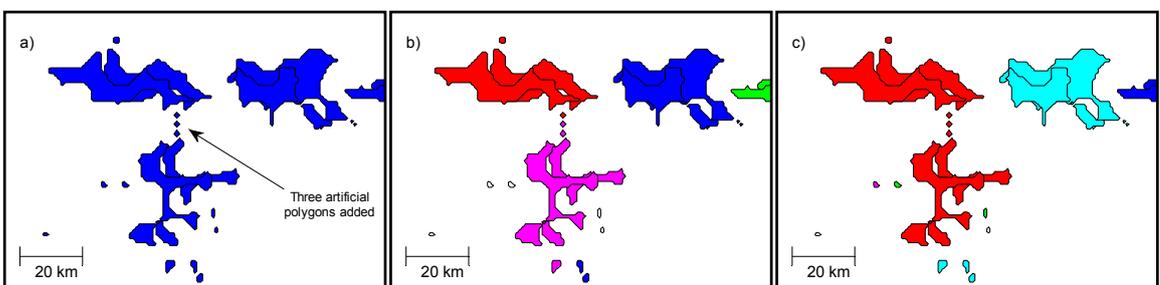


Fig. 19: The meaning of the parameter $minCard$. a) Original data, where three small artificial polygons are added to the data (see the arrow in the figure) b) Correct clustering with $NPred = 2.5$ km, $minCard = 7$ km² c) $NPred = 2.5$ km, $minCard = 2$ km². The artificial polygons link two large clusters unnecessarily together because the parameter $minCard$ is too small.

6.1.3 The meaning of number of consecutive frames included in the clustering – a case study on Aug 26th 2007

This subsection examines the significance of the parameter n_f introduced in Subsection 5.3.2. The parameter determines the number of consecutive frames used in the clustering and plays an important role when building up a robust tracking algorithm. Usually convective cell tracking algorithms consider only two consecutive frames. Because of this, the tracking can be intolerant of errors in radar data. For example, attenuation or radar malfunctioning may result in missing data and consequently in severe problems in the tracking. Therefore, a more robust tracking is obtained by increasing n_f .

On Aug 26th 2007 a long but relatively narrow convective system passed over the southern Finland. About 5,000 cloud discharges were located and the highest local ground flash densities were about 8 fl./(100 km² day) (Rossi and Mäkelä 2008).

At first, the tracking was performed by considering only two consecutive frames in the clustering. Fig. 20 presents the tracking at 10:30, 10:40 and 10:45 UTC. All the tracks at 10:45 are discontinued. This is due to an error in radar data at 10:40 UTC when almost the whole reflectivity pattern disappears erroneously in the radar image.

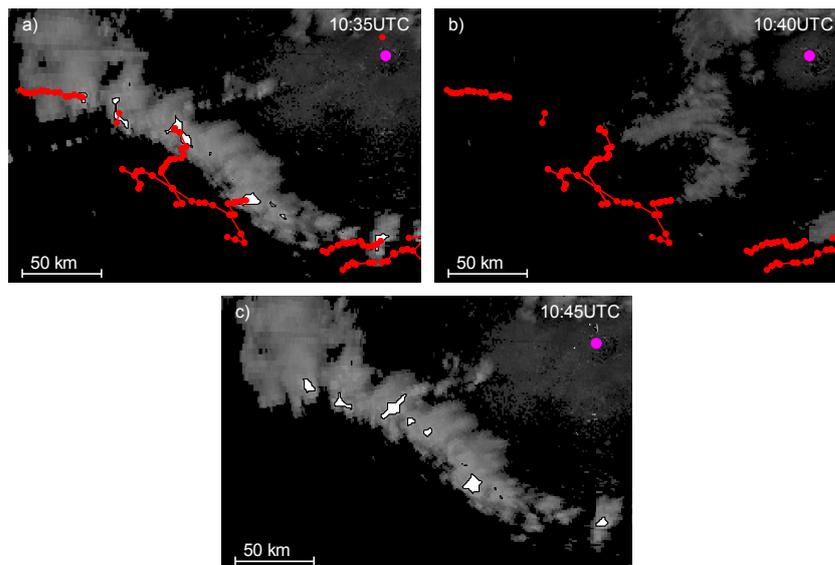


Fig. 20: Cell tracks on Aug 26th 2007 a) at 10:35 UTC, b) at 10:40 UTC and c) at 10:45 UTC. Only two consecutive frames are used in the clustering. All the tracks are discontinued at 10:45 because the radar reflectivity factor pattern vanishes erroneously at 10:40. The magenta dot indicates the location of the Vantaa radar.

When n_f was increased by one, the result was significantly better as Fig. 21 shows. Most of the tracks were continued because at 10:45 the two previous frames (10:40 and 10:35) are included in the clustering. Even though the cells at 10:40 are missing, the tracking is continued with the cells at 10:35. This “backup information” provides the continuity of the tracking in this case.

Naturally we may not increase the number of consecutive frames n_f arbitrarily. If the parameter is too large, tracks may show unnatural behavior and even distant cells are

clustered together. According to several test runs with an extensive data set, a suitable value of n_f is two or three, depending on the quality of the data.

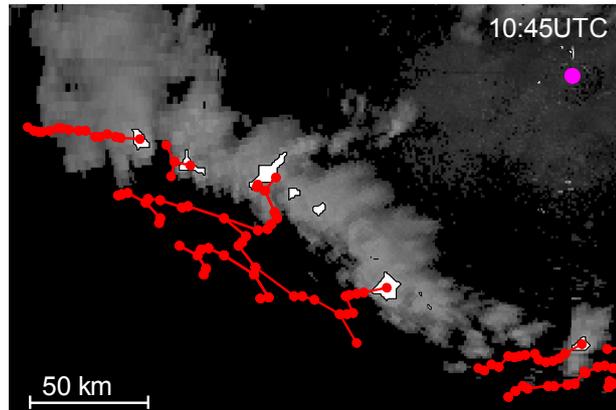


Fig. 21: Robust tracking by increasing number of consecutive frames in the clustering. The tracking continues at 10:45 even though there is an error in the radar data at 10:40.

6.1.4 Importance of displacement – a case study on Aug 9th 2005

In here, we discuss the meaning of the displacement introduced in Subsection 5.3.3. The use of the displacement is important, especially when the cells are moving fast. The importance of the displacement was tested visually through several test runs with and without the displacement. Generally, with the 5 min time resolution, the benefit of the displacement approach is often negligible. Usually the velocity of the cells is relative low compared to the 5 min interval and most likely the cells in the successive frames belong to the same cluster even if the displacement is not used. However, some problems may arise if the cells are moving at a high speed or the interval between the frames is higher.

Since we do not know the number of correct tracks in advance, an automatic test of the tracking is hard. Nevertheless, a heuristic approach is to examine the number of identified tracks and their lengths. One of the goals of the correspondence tracking (see Subsection 4.2.3) is to minimize the number of tracks as a large number indicate also a large number of discontinuities (Veenman et al. 2001). In addition, discontinued tracks are short and hence they should be avoided. These assumptions do not explicitly measure the optimal tracking, since they do not accommodate the motion criteria defined in Subsection 4.2.3, but give some clue on the tracking performance. During the test, the parameters $NPred$ and $minCard$ were kept constant.

The displacement approach was tested with data acquired on Aug 9th 2005. That day, the southern Finland encountered an extensive convective system bringing heavy showers and intense lightning. The local ground flash densities were about 30 fl./($100 \text{ km}^2 \text{ day}$), and more than 20,000 cloud discharges were located during that day (Rossi and Mäkelä 2008). The system propagated very fast, which made the tracking task difficult.

Table 1 represents the number of different tracks and track lengths in the test. If the displacement is ignored, the number of different tracks is higher and the mean length of

the tracks is lower. This implies that less discontinued tracks occur with the displacement approach.

However, in Table 1, the number of tracks with the 15 min interval is smaller than with the 5 min interval, if the displacement is not used. This is an unintuitive result. An explanation for this is noise, such as sea and ground clutter, which is misinterpreted as convection. If we assume that the average amount of noise is constant in every frame, the amount of noise and consequently the number of incorrect cells is larger with the shorter time interval. In this case, the effect of noise probably outweighs benefits obtained from the high time resolution.

Table 1: Performance of the tracking.

Method / Interval between two frames	Number of tracks	Mean length of the tracks
Tracking, no displacement / 5 min	378	13.11 min
Tracking with displacement / 5 min	255	18.37 min
Tracking, no displacement / 15 min	315	19.07 min
Tracking with displacement / 15 min	264	19.66 min

6.1.5 Considerations on data quality – a case study on Aug 9th 2005

In here, we examine the importance of data fusion in the tracking. The test was carried out with the data of Aug 9th 2005 by tracking reflectivity cells, flash cells and the combination of these two data types.

Fig. 22.a shows the reflectivity tracks at 16:30 UTC. Several individual cells are crossing the Helsinki region and heading west. In general, the tracks are uniform and consistent. However, a couple of complex reflectivity cell systems are present in the lower left corner. These systems are susceptible to merging and splitting and therefore the track looks a bit disoriented. Even so, the general trend of the tracking is evident.

Fig. 22.b represents the same situation but only 15 minutes later. The cells have vanished from the figure. This is not due to decaying of the cells because lightning is still present indicating the existence of convection. The reason for the disappearance is probably attenuation since an intense precipitation is blocking the radar (marked with the magenta dot in Fig. 22).

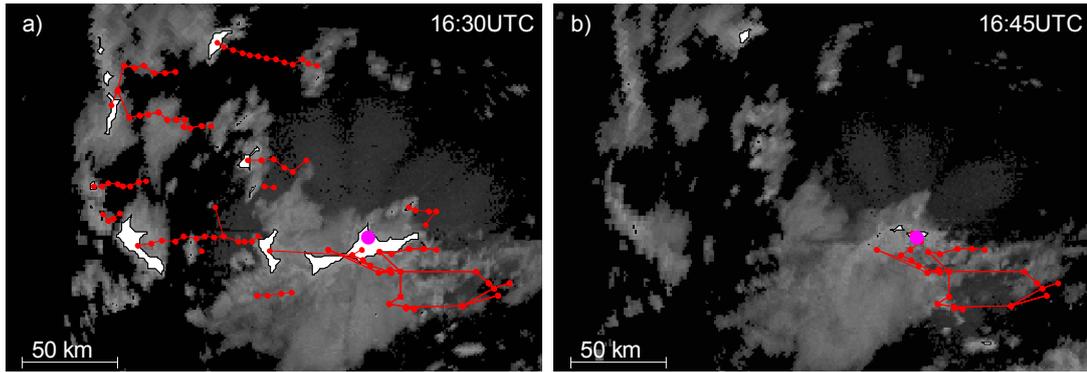


Fig. 22: Reflectivity cell tracks on Aug 9th 2005 at a) 16:30 UTC, b) at 16:45 UTC (local time is UTC+3 hours). Attenuation produces vanishing of convective cells at 16:45 UTC. The radar is marked with a magenta dot.

Despite the attenuation, we may try to capture the correct tracks by utilizing lightning data. Fig. 23.a represents the tracks obtained by tracking the corresponding flash cells. The tracks still exist because lightning continues even though the radar undergoes a severe attenuation.

The tracking in Fig. 23.b utilizes the combination of reflectivity cells and flash cells. The tracks remind closely to those in Fig. 22. However, in contrast to Fig. 22, the tracking continues in the upper left corner as the flash cells still remain.

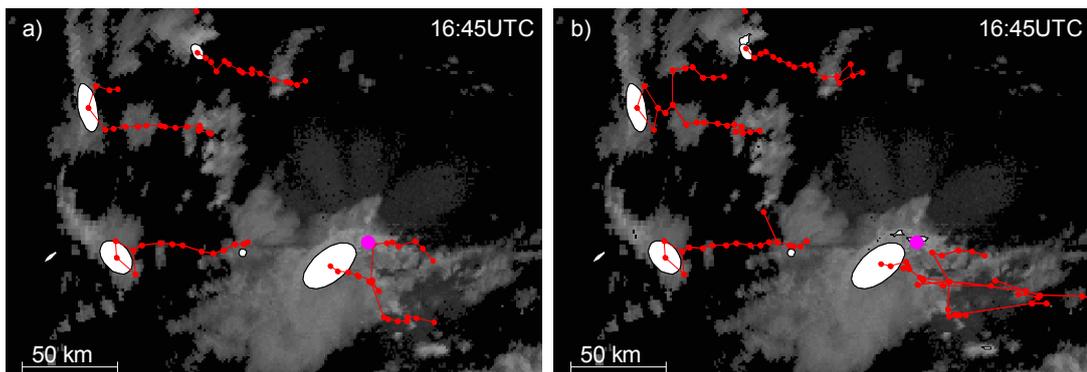


Fig. 23: The tracking at 16:45 a) by flash cells, b) by combining flash cells and reflectivity cells. The tracking continues even though the reflectivity cells disappear due to attenuation.

6.2 Time series of convective cell tracks

As pointed out earlier in Subsection 4.3.2, one of the most significant advances of the object-oriented tracking methods is their ability to capture the history of individual convective cells. After the tracking, we may attach important information, such as located lightning flashes, to the cells. This information is useful in the behavior and life cycle analysis of convective cells.

6.2.1 Attaching information to the cells

In order to present the cell behavior and characteristics, we have to attach lightning, CAPPI 500 m and EchoTop 20 dBZ information to the cells.

Lightning data were attached by the nearest neighbor method; the assumption was that the “mother cell” lies in the proximity of the located flash and thus it can be attached to the

closest cell. However, as discussed in Section 3.4, lightning data contain errors. To avoid the interference of possible outliers and location errors, flashes with the distance over 20 km to the closest cell were rejected. This is a reasonable assumption, as usually flashes occur within or close to the cell boundary. According to test runs, the rejection limit of 20 km was eligible. Still, especially a ground flash may strike occasionally far from its origin and therefore some of the ground flashes can be misinterpreted as noise.

In order to examine the information included in the cell radar parameters, we need to consider the pixels falling into the cell region. Since the number of these pixels may be large, it is convenient to calculate some descriptive value of all the pixels inside a cell. This can be, as an example, the mean or maximum of the pixel values. Also using a percentile is an option. The percentile is the point, below which a chosen percentage of observations fall, e.g. the 50th percentile equals to the median and the 100th percentile is the maximum value.

Using the mean value usually filters out and smoothes possible noisy pixels. On the other hand, it often underestimates the cell characteristics. For example in the mature phase, usually only a part of the EchoTop 20 dBZ pixels within the cell penetrate into the higher altitudes. These high pixel values contain also the most important information regarding the cell life cycle phase. As a consequence, smaller pixel values may outweigh high EchoTop 20 dBZ values in the mean operation and the interesting information is deteriorated. Therefore, in here the mean is discarded as a too conservative and uninformative descriptive value.

The same justification can be applied for the use of the maximum CAPPI 500 m values. When the cell starts to mature, only part of pixels falling into the cell attain really high values while the periphery of the cell is dominated by lower reflectivity values. This makes the maximum a more useful descriptive value.

From now on in this thesis, the cell EchoTop 20 dBZ and CAPPI of 500 m values stand for the maximum value of the pixels falling in the cell area.

6.2.2 Preprocessing of the time series

In order to mitigate the effect of noisy behavior, cell parameter time series were filtered appropriately. For this, we use the concept of LTI-filtering introduced in Subsection 5.3.3. CAPPI 500 m and EchoTop 20 dBZ time series were filtered by a moving average filter with the window size of 15 min, that is, we set the coefficient in (5.24) as $b_0 = b_1 = b_2 = 1/3$ and the rest of the coefficient as zero. The filtering window of 15 min seems to smooth most of the noise and does not interfere with the life cycle information. A wider filtering window would inflict too large phase lag in the filtering output and eliminate important life cycle information.

The lightning frequency was filtered through the first-order IIR-filter as defined in (5.35). For the rapidly changing and impulsive lightning data, this filter type is intuitively appealing. The 5 min flash rate can be considered as an occasional sample measure of the

intensity of electric field in a convective cell. When the lightning ceases, the electric field within the cell should not disappear suddenly after a while but dissipate exponentially. This can be achieved by using the IIR-filtering. In addition, the used filtering is consistent with the cell life cycle and it does well with the fuzzy inference model used in this thesis.

6.2.3 Cell analysis through time series plots

In its simplest form, all the cell parameters can be represented with a time series plot, which itself is a useful and informative tool in for the cell analysis. Examples of such a graph are given in Fig. 24 and Fig. 25. In Fig. 24, the cell is intensifying in the beginning as the storm size, EchoTop 20 dBZ, CAPPI 500 m and lightning increase rapidly. Between 40-70 min the cell has achieved its mature phase, which is followed by the decreasing of the parameter values indicating dissipation. The cell is potentially dangerous due to its high values of the CAPPI 500 m and EchoTop 20 dBZ as well as lightning it is producing. This is a typical example of an idealized convective cell described earlier in Section 2.1. These cells clearly consist of intensifying, mature and dissipating phase, which can be seen as increase and decrease of the cell parameters.

The single-cell storms, as in Fig. 24, are easier to analyze than MCSs, whose awkward structure can be observed also in the time series plot. Usually the graphs of MCSs, as shown in Fig. 25, contain several ups and downs indicating the multi-phase evolution of the storm. Thus, it is very difficult to analyze whether the storm is really dissipating or not. As noted in Section 4.4, Wilson et al. (1998) has argued that the storm evolution is not defined unambiguously by the development of reflectivity pattern. Especially physical events occurring in the boundary layer, such as convergence lines, produce new cells to the MCS and have effect on the life cycle. This assumption explains also the complexity of many long-lived convective systems; when the cell seems to die out, a new cell emerges in the neighborhood of the old dissipating cell and the system starts to intensify again. This is often due to the rushing cold air front caused by the cell's downdraft reaching the ground. The cold air front propagates horizontally forcing the surrounding warmer air to rise and triggering the formation of new cells.

As noted in Section 2.3, MCSs tend to live several times longer than single-cell systems. Our results support this hypothesis. This is natural since new cells emerge in the vicinity keeping the system alive. In addition, the area of such system usually seems to reach several times higher rates compared to single-cell systems. Vigorous MCSs are frequently accompanied with intense lightning.

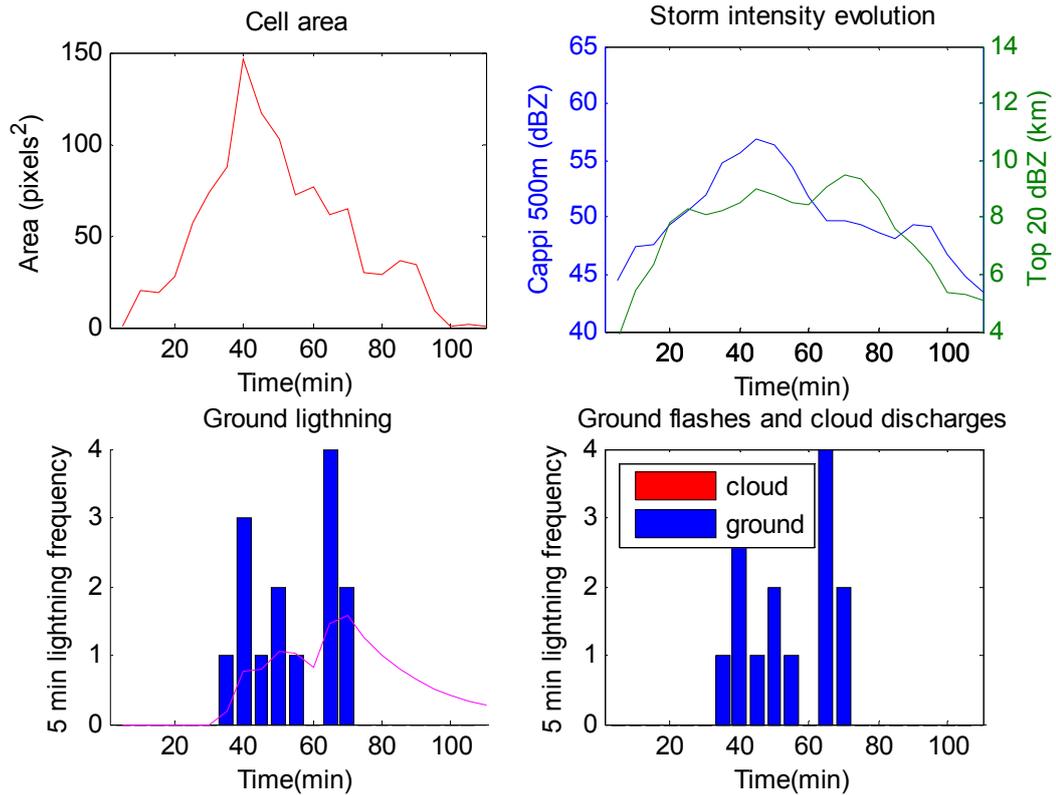


Fig. 24: Evolution of the parameters of a single-cell storm. The intensifying and dissipating phases can be observed from the image. The lifetime is usually less than two hours.

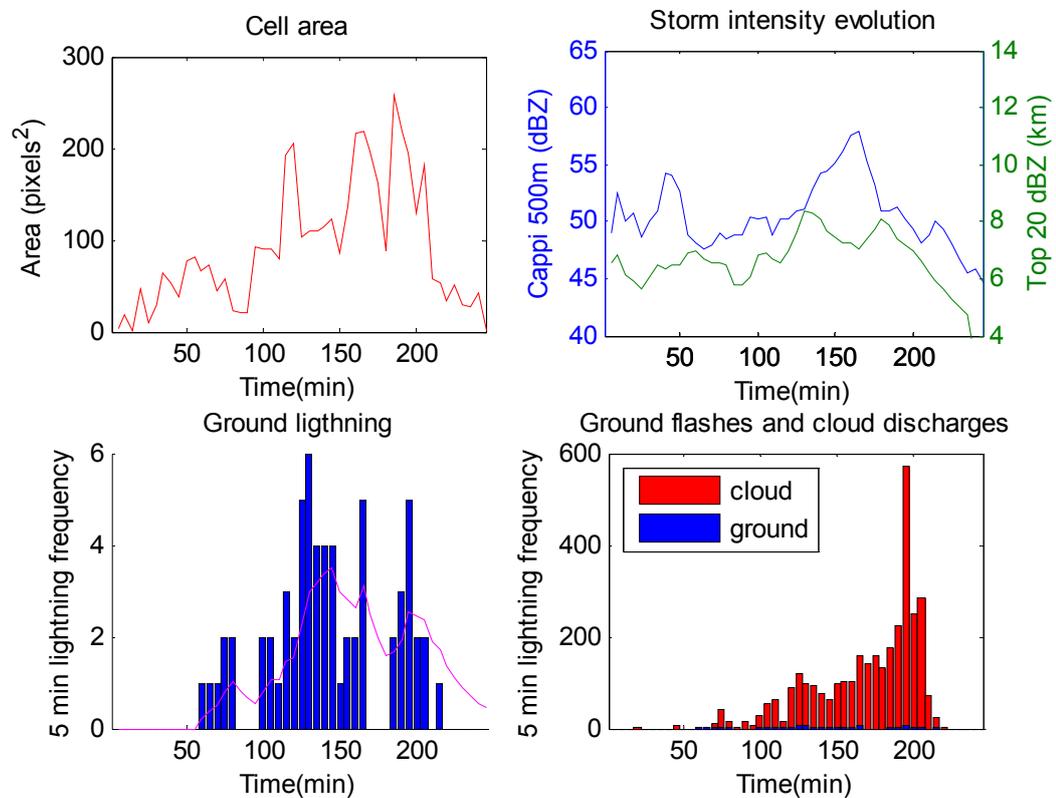


Fig. 25: An intense MCS. The lifetime of such a system may exceed several hours.

6.3 Descriptive statistics of convective cells

Statistical analysis of convective cells was studied through the tracking method. In order to understand typical behavior of convective cells in Finland, we derived descriptive statistics for the cells. We compared also the result to the prevailing consensus of the convective cell life cycle development.

The test consisted of 12 days with convective behavior. The dates were:

9.8.2005, 26.8.2006, 27.8.2006, 28.8.2006, 9.5.2007, 30.5.2007, 15.6.2007, 2.7.2007, 17.7.2007, 14.8.2007, 26.8.2007 and 27.8.2007.

The tracking algorithm extracted 4035 convective cell tracks of which 3024 were rejected because of the following reasons:

1. The length of the track was too short. Short tracks were due to incorrect cell identification or discontinuities in the cell tracking. Often the cell was identified during single radar image only, resulting in a trivial track.
2. The storm existed in the beginning or in the end of the sequence of radar images resulting in an incomplete track.
3. The storm entered or exited the radar image area resulting in an incomplete track.

The same rejection criteria were used also by Dixon and Wiener (1993) in their test with the TITAN convective cell tracking algorithm.

6.3.1 Storm duration

The primary descriptive statistic of convective cells is the lifetime duration. In order to analyze the lifetime of the convective cell statistically, we need to know the distribution of the duration. This is represented in Fig. 26 in the form of a histogram.

The storm duration seems to follow an exponentially decreasing distribution, indicating that the short-lived cells are far more frequent than persistent long-lived cells. This result contradicts with the current consensus. Some studies have suggested that a mean life cycle of convective cells is around 20 min (Wilson 1998). Dixon and Wiener (1993) calculated the distribution of storm duration with the TITAN tracking algorithm and achieved the mode lifetime of 12-24 minutes. However, complex MCSs may last up to several hours (e.g. Puhakka 1995).

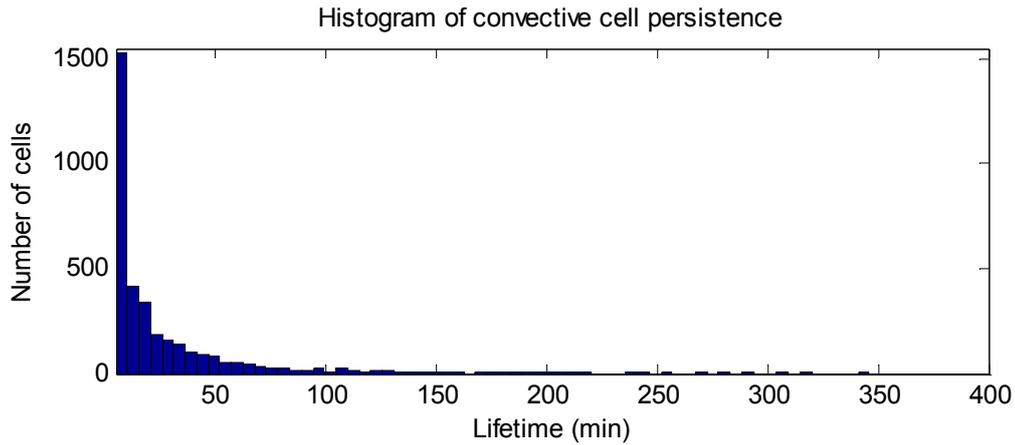


Fig. 26: Histogram of storm persistence.

An explanation for this inconsistency is based on the used tracking algorithm. The algorithm is able to capture only part of the cell life cycle. Probably the most of the cells attain the required intensity of 40 dBZ in CAPPI 500 m images only for a short duration. Therefore, the short-lived cells outnumber the other cells and the real distribution is distorted.

However, if we consider only storms that produce lightning, the results are more consistent with the theoretical convective cell life time. In Fig. 27, we have considered only convective cells producing more than 3 CG flashes. The cells with lightning have larger radar reflectivity factor values and the tracking is able to capture relatively longer part of the life cycle.

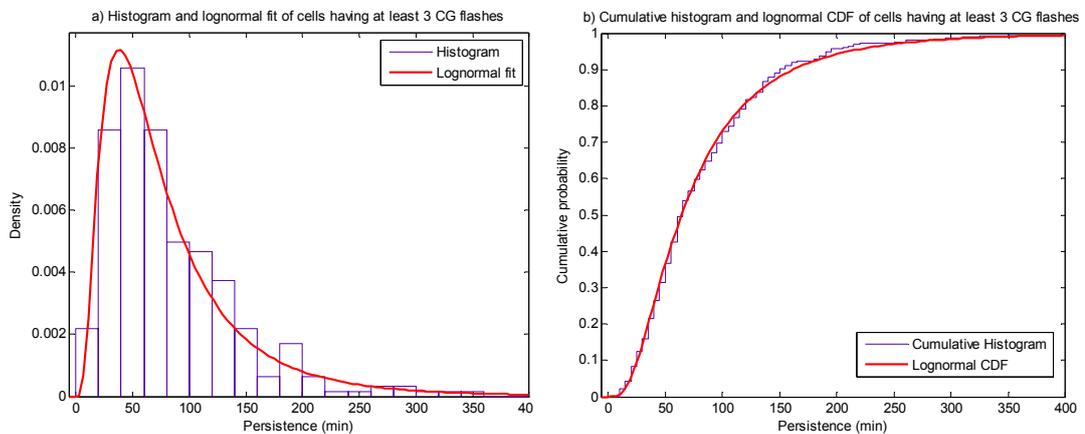


Fig. 27: Lognormal probability density function and cumulative probability function fitted to the histogram of cells having at least 3 CG flashes.

The maximum value of the histogram in Fig. 27.a is around 50 min. The standard deviation of the distribution is 63 min indicating a high variability in the lifetime.

The data was fitted to the lognormal distribution (Wikipedia 2008c), which seems to follow the histograms adequately. For the estimated lognormal fit represented in Fig. 27.b, the expected and mode lifetime of a cell producing at least 3 CG flashes is 83 min and 40 min respectively. The cumulative distribution in Fig. 27.b shows that 90% of all

cells exceeds the lifetime of 25 min. Also 90% of all cells have the lifetime shorter than 175 min.

If the cells are divided into several groups according to the sum of produced CG flashes, we obtain the lognormal probability density distributions and cumulative distributions as represented in Fig. 28. The expected value and mode of the lifetime increase as the cell lightning activity increases. This is an expected result, since the long-lived cells have more time to produce lightning. The variability of the lifetime increases with cells having high lightning activity.

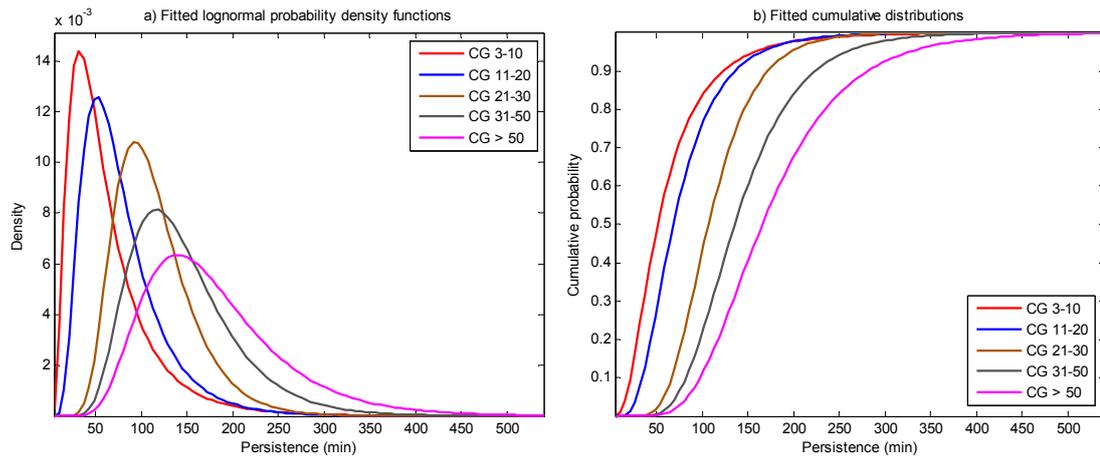


Fig. 28. Fitted lognormal probability density functions and cumulative distributions with increasing lightning activity

6.3.2 Typical life cycle time series

In order to understand a typical life cycle of convective cells, an average life cycle was calculated. However, different convective cells have the lifetime of varying lengths. Therefore, a simple average of all the parameter time series does not produce very representative result in the sense of cell life cycle development. In order to understand the average behavior of each cell in each life cycle phase, we defined *normalized life cycle* by normalizing the duration of each cell linearly to the interval 0...1. Since each cell was normalized to the same interval, we could examine the characteristic behavior of convective cells regardless of large variations in their lifetimes.

In Fig. 29, we have calculated the average of normalized life cycles for cell area, EchoTop 20 dBZ, CAPPI 500 m reflectivity and 5 min CG flash rate activity. Cells persisting less than 30 min were rejected in the study, since a relative long 5 min sampling interval makes the comparison of phase differences between short time series indefinite. As Fig. 29 shows, on average cells reach the maximum approximately at 50 % of the life cycle. This signifies that on average the intensity maximum of a life cycle is reached in the midpoint of the lifetime, after which the cell starts to decay. The shape of the normalized evolution of EchoTop 20 dBZ and CAPPI 500 m parameters are parabolic indicating rapid increase and decrease in the parameter values in the beginning and in the end of the life cycle, respectively. Hence, we infer that cell intensification is fast in the beginning and decelerates while the cell matures. Finally, the intensification turns into

dissipation, which is most evident in the end of the life cycle. The average cell EchoTop 20 dBZ in the midpoint of the life cycle is approximately 7.3 km.

The average growth and decay of cell area and lightning activity are more static leading to a triangular shaped life cycle. This result does not explicitly indicate the cell behavior shown in Fig. 29.d. Usually lightning does not seem to grow statically, but rather impulsively. However, the result shows that the most intense CG period is usually reached in neighborhood of the life cycle midpoint.

The outcome of the average normalized life cycle is slightly different if we categorize the cell time series based on the storm persistence. Fig. 30 represents average normalized life cycle durations for the cells having a life cycle of 25-150 min, 100-225 min and over 150 min. As expected, the average values increase with the cell duration, implying that long-lived cells tend to be more intense and potentially dangerous than short-lived cells. In addition, the behavior of persistent cells is less smooth. This is probably due to statistically small amount of long-lived cells, but also the characteristic alternating life cycle of long-lived cells may have effect on the result; long-lived MCSs usually have several intensifying and dissipating phases resulting in less smooth time series.

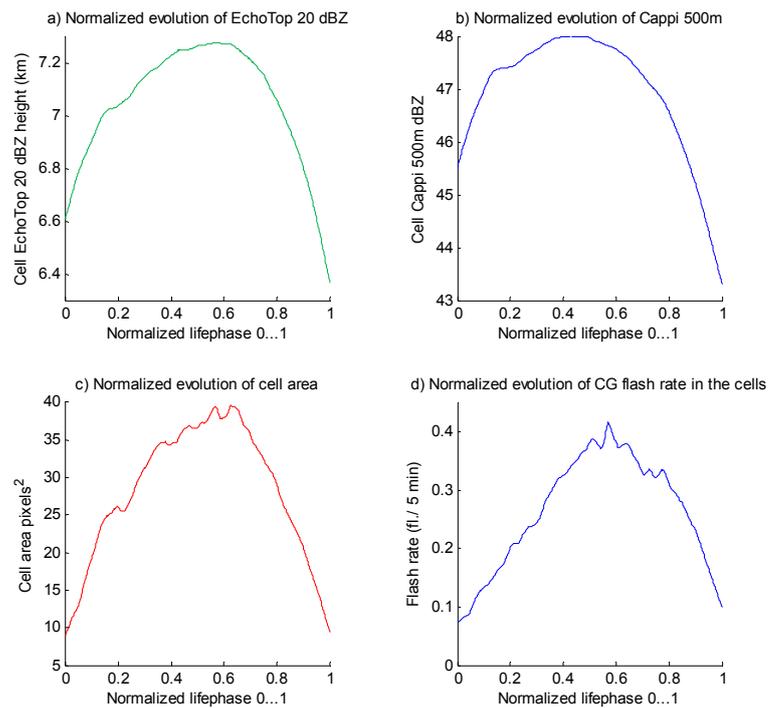


Fig. 29: Average normalized evolution of a) EchoTop 20 dBZ, b) CAPPI 500 m, c) area and d) CG flash rate.

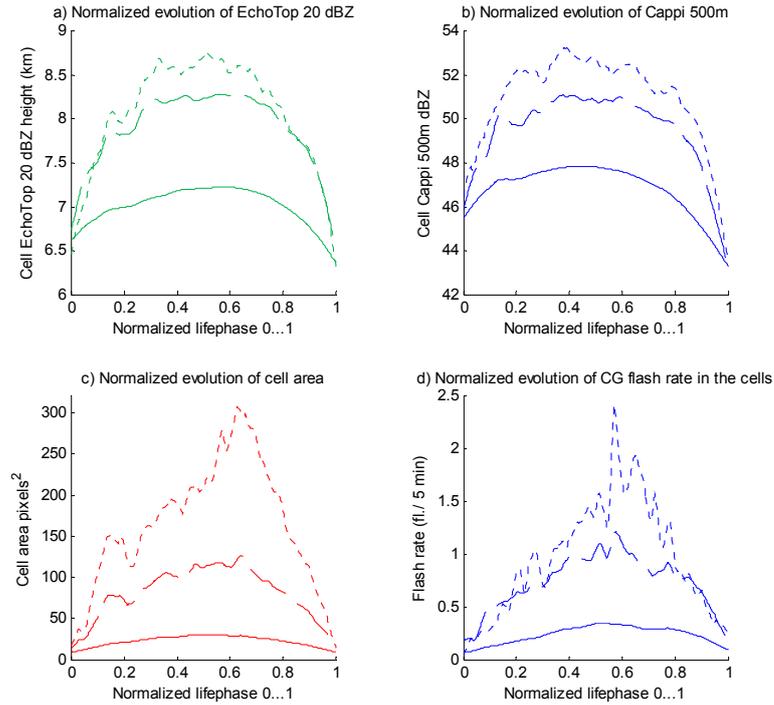


Fig. 30: Average normalized evolution categorized by the cell duration. The classes 25-150 min, 100-225 min and over 150 min are represented with solid, dashed and dotted line respectively. Represented time series in the figures are a) EchoTop 20 dBZ, b) CAPPI 500 m, c) area and d) CG flash rate

6.3.3 Cell development and lightning

Lightning is known to indicate severe convection and the most dangerous convective cells include lightning almost without exception. In order to attain the lightning phase, cells need to reach high updraft speed, cloud top altitude and intense rain (see Section 2.2). Contemporary literature in radar meteorology views the cell development usually with respect to different parameters derived from the radar reflectivity factor. In this subsection, we study the relationship between lightning and the radar derived life cycle of the convective cell. According to several tests, EchoTop 20 dBZ describes well the cell development. Although EchoTop 20 dBZ does not explicitly measure the actual height of the cell, it illustrates the vertical structure of radar reflectivity factor within the cells and thereby cell development. In the following, we assume that the life cycle phases are predominantly dictated by EchoTop 20 dBZ. Therefore, in order to examine the relationship between lightning and cell development, 5 min CG flash rate was compared with EchoTop 20 dBZ development. The cloud flash data was ignored in this study, because the effective cloud flash detection range that overlaps with the Vantaa radar image area is rather small, and hence the number of cells with a qualified cloud discharge rate was too low. In addition, we have ignored the cells producing less than 3 CG flashes. As an example, the active lightning period of cells producing only one CG is trivial and all the descriptive values of the lightning phase are mapped to the same value. Also cells persisting less than 30 min were rejected, since it is not reasonable to compare phase differences of short time series.

In Fig. 31.a, we have estimated the histogram of the time difference between the maximum EchoTop 20 dBZ and the first lightning flash in a cell. As the histogram shows, the first flash strikes usually before the cell have reached the maximum EchoTop 20 dBZ implying that the cell is still in the intensifying phase. According to the cumulative histogram in Fig. 31.b, over 82 % of all lightning periods start before the EchoTop 20 dBZ maximum. However, the mean and mode values of the distribution are close to zero signifying only a small difference between EchoTop 20 dBZ maximum phase and the first ground flash.

Fig. 31.c represents the histogram of the time difference between the maximum EchoTop 20 dBZ phase and the maximum lightning phase. The variance of the distribution is relatively large, which implies that the correlation between maximum CG flash rate and EchoTop 20 dBZ rate is not very large. Still, the mean and mode of the distribution are close to zero, indicating that the maximum CG flash rate and the maximum EchoTop 20 dBZ are expected to occur at the same time. However, the maximum lightning phase is suggested to occur after the maximum EchoTop 20 dBZ (personal communication with Elena Saltikoff on Nov 11th 2008, Saltikoff 2008), which conflicts with the result represented herein.

Fig. 31.e corresponds to Fig. 31.a but the histogram is estimated for the time difference between the cell EchoTop 20 dBZ height maximum and the final lightning flash of the cell track. The result implies that the lightning continues over the mature phase until the dissipating phase. According to the cumulative histogram in Fig. 31.f, more than 93 % of all lightning periods last beyond the maximum EchoTop 20 dBZ height.

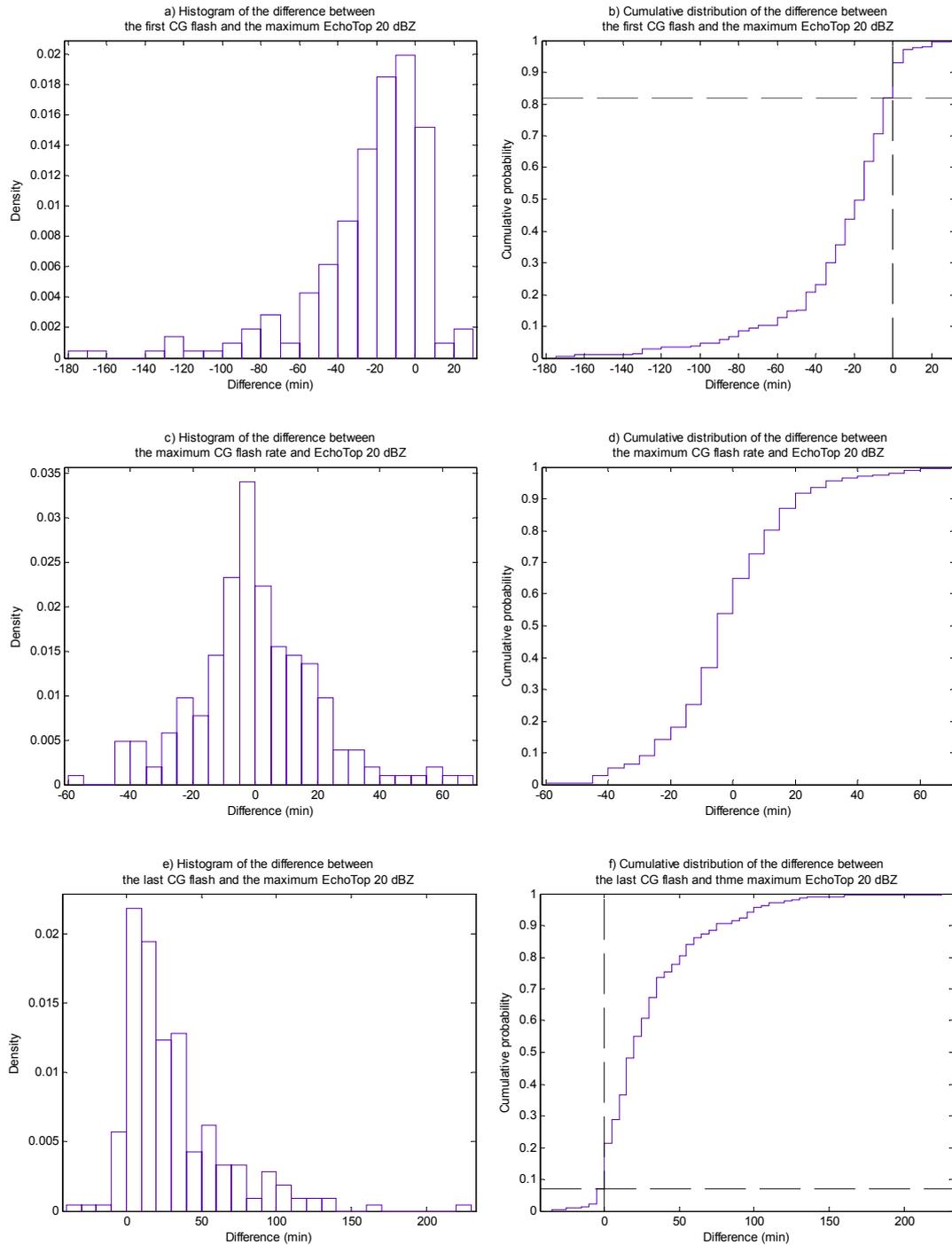


Fig. 31: Histogram and cumulative distribution of the time difference between the first CG flash and EchoTop 20 dBZ maximum in (a) and (b), the maximum CG flash rate and EchoTop 20 dBZ in (c) and (d) and the last CG flash and EchoTop 20 dBZ maximum in (e) and (f).

6.3.4 Relationship between lightning and radar parameters

This subsection studies the relationship between lightning and radar parameters in convective cells. Usually intense storms with lightning activity feature with high EchoTop 20 dBZ and CAPPI 500 m values. In order to verify this result, we examined radar parameter distributions of the cells with and without lightning. Let ω_1 and ω_2 denote the classes “lightning activity” and “no lightning activity”, respectively. Fig. 32 represents histograms of cell track maximum EchoTop 20 dBZ height of these two

classes. As expected, cells producing lightning reach higher maximum EchoTop 20 dBZ values; the mean values are 9.5 km and 7.3 km for ω_1 and ω_2 respectively. However, the margin between the mean values is not large and the distributions are heavily overlapping. This means that in the statistical sense, a clear distinction between these cell types cannot be made. If we compare the CAPPI 500 m reflectivity values, the outcome is even worse; the margin is even smaller and the major part of all the reflectivity values belongs to both distributions.

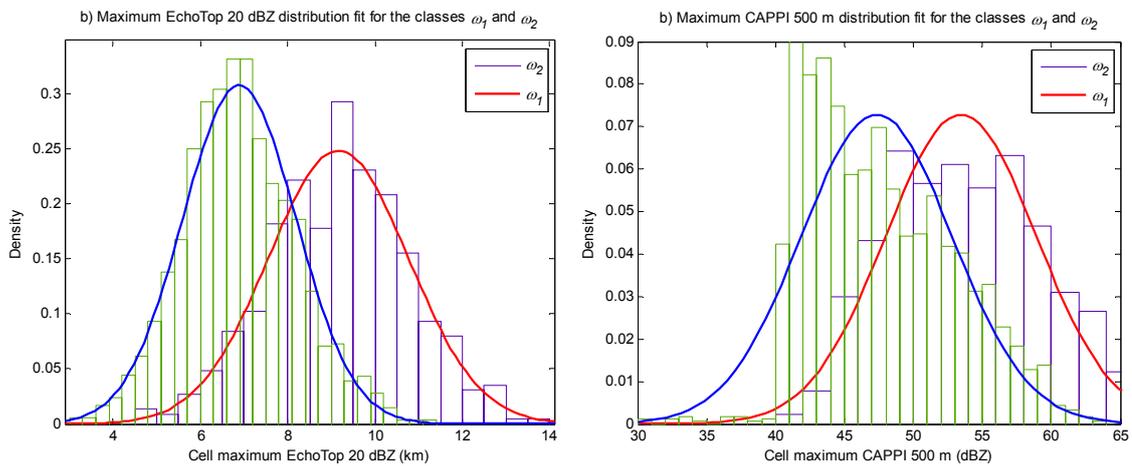


Fig. 32: a) Cell maximum EchoTop 20 dBZ and b) CAPPI 500 m parameter histograms and fitted normal distributions for the classes ω_1 and ω_2 . In the histograms the class ω_1 is represented with the blue bars and ω_2 with the green bars.

According to Fig. 32 both parameters, excluding CAPPI 500 m of the class ω_2 , seem to follow the normal distribution adequately. CAPPI 500 m of the class ω_2 follows the normal distribution poorly due to the applied cell identification mechanism; the cells are defined only by using a single reflectivity threshold of 40 dBZ. Therefore, the cells not attaining the required threshold are not included in the data. In spite of this, the normality assumption is considered because it makes the computational part of the following analysis more convenient.

The estimated distributions in Fig. 32 can be understood as the marginal distributions of the joint probability density distribution defined by the EchoTop 20 dBZ and CAPPI 500 m. In here, we suppose that the joint distribution can be estimated with the two-dimensional normal distribution. This is a justified assumption, since the marginal distributions of a multivariate normal distribution (Appendix B) are also normal. Fig. 33 represents the maximum of joint probability density distributions $p(\mathbf{x}|\omega_1)$ and $p(\mathbf{x}|\omega_2)$ estimated for cell maximum EchoTop 20 dBZ and CAPPI 500 m data.

In the nowcasting scenario, an end-user is not usually interested in convective storms in general but rather in certain type of cells, such as those producing lightning. Therefore, a desired application would be a classification model that identifies whether or not cells produce lightning based on the radar measurements. By this, we could predict potentially dangerous cells beforehand by giving a forewarning about a convective cell that will

probably produce lightning. Possible end-users of the classification model could be, for example, aviation, industry or power distribution maintenance.

For the classification task, we may utilize the estimated probability distributions in Fig. 33. Statistically optimal way for the classification is to find a decision rule that minimizes the probability of the incorrect classification i.e. error probability (Theodoridis and Koutroumbas 2003). For this we may apply *Bayesian Decision Theory* (e.g. Bishop 1995). We will consider again the two classes, ω_1 and ω_2 , and the feature vector \mathbf{x} consisting of the two radar parameters. The *a posteriori* distribution of the class i is given by the well-known *Bayes rule*

$$P(\omega_i | \mathbf{x}) = \frac{p(\mathbf{x} | \omega_i)P(\omega_i)}{p(\mathbf{x})}, \quad (6.1)$$

where

$$p(\mathbf{x}) = p(\mathbf{x} | \omega_1)P(\omega_1) + p(\mathbf{x} | \omega_2)P(\omega_2). \quad (6.2)$$

The posterior distribution describes the probability of a class ω_i given the feature vector \mathbf{x} . If we know the prior probabilities $P(\omega_i)$ and the likelihood functions $p(\mathbf{x} | \omega_i)$, we may estimate the probability of lightning with respect the measured cell maximum EchoTop 20 dBZ and CAPPI 500 m values using the Bayes rule. In this case, we get for the prior probabilities $P(\omega_1) = 0.2$ and $P(\omega_2) = 0.8$, which means that out of ten convective cells only two produce lightning. The maximum of likelihood functions $p(\mathbf{x} | \omega_i)$ is depicted in Fig. 33.

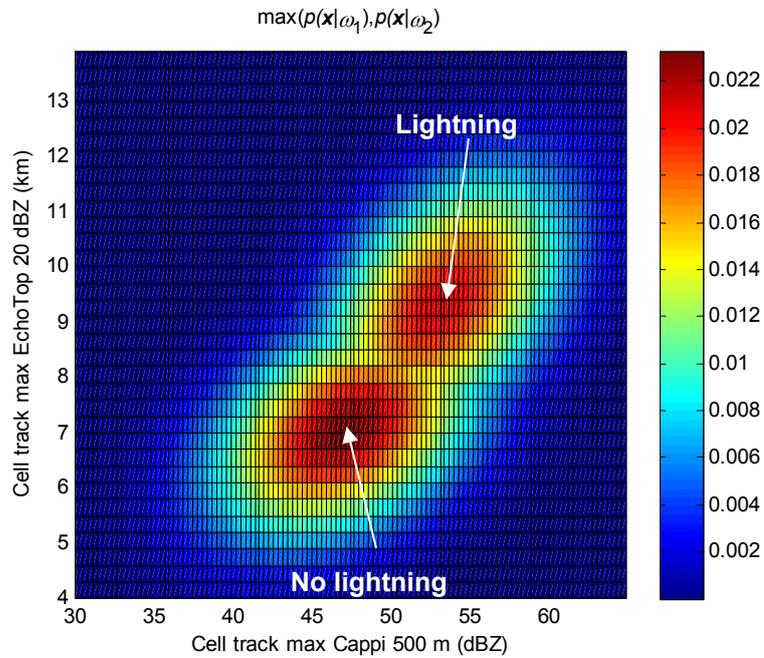


Fig. 33: Joint probability density distributions of the cells producing and not producing lightning plotted in the same figure.

The error probability to be minimized is defined as follows. Consider a decision boundary that partitions the feature space in two regions R_1 and R_2 , in which we choose ω_1 , if the measured feature vector falls in R_1 , and ω_2 , if the feature falls in R_2 . An incorrect classification occurs if \mathbf{x} falls in the region R_1 although it belongs to ω_2 or if \mathbf{x} falls in R_2 although it belongs to ω_1 . Formally, the error probability is

$$P_e = P(\mathbf{x} \in R_2, \omega_1) + P(\mathbf{x} \in R_1, \omega_2) \quad (6.3)$$

and using the Bayes rule

$$\begin{aligned} P_e &= P(\mathbf{x} \in R_2 | \omega_1)P(\omega_1) + P(\mathbf{x} \in R_1 | \omega_2)P(\omega_2) \\ &= P(\omega_1) \int_{R_1} p(\mathbf{x} | \omega_1) d\mathbf{x} + P(\omega_2) \int_{R_2} p(\mathbf{x} | \omega_2) d\mathbf{x}. \end{aligned} \quad (6.4)$$

From (6.4), we can see that the error is minimized if we partition the feature space into the decision regions R_1 and R_2 such that

$$\begin{aligned} R_1 &: P(\omega_1 | \mathbf{x}) > P(\omega_2 | \mathbf{x}) \\ R_2 &: P(\omega_1 | \mathbf{x}) < P(\omega_2 | \mathbf{x}). \end{aligned} \quad (6.5)$$

This means that we choose the class that has the largest *a posteriori* probability $P(\omega_i | \mathbf{x})$. The boundary between R_i and R_j is denoted as the *decision boundary* g_{ij} . If we have two normally distributed classes ω_i and ω_j , the optimal decision boundary g_{ij} is of quadratic type (Appendix B)

$$g_{ij}(\mathbf{x}) = \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b} \mathbf{x} + c = 0. \quad (6.6)$$

In our classification task, the distributions for both classes were estimated with two dimensional normal distributions as represented in Fig. 33. Fig. 34.a shows the estimated a posteriori probability of lightning $P(\omega_1 | \mathbf{x})$ calculated by the Bayes rule (6.1), and Fig. 34.b illustrates the maximum of likelihood functions $p(\mathbf{x} | \omega_i)$. The optimal decision boundary is represented with the white line in Fig. 34. As Fig. 34.a shows, the optimal decision boundary does not follow exactly the minimum of $\max(p(\omega_1 | \mathbf{x}), p(\omega_2 | \mathbf{x}))$ and it is biased towards the class ω_1 . This is due to the different prior probabilities; as estimated above, the majority of storms does not produce lightning and therefore the classification favors the class ω_2 .

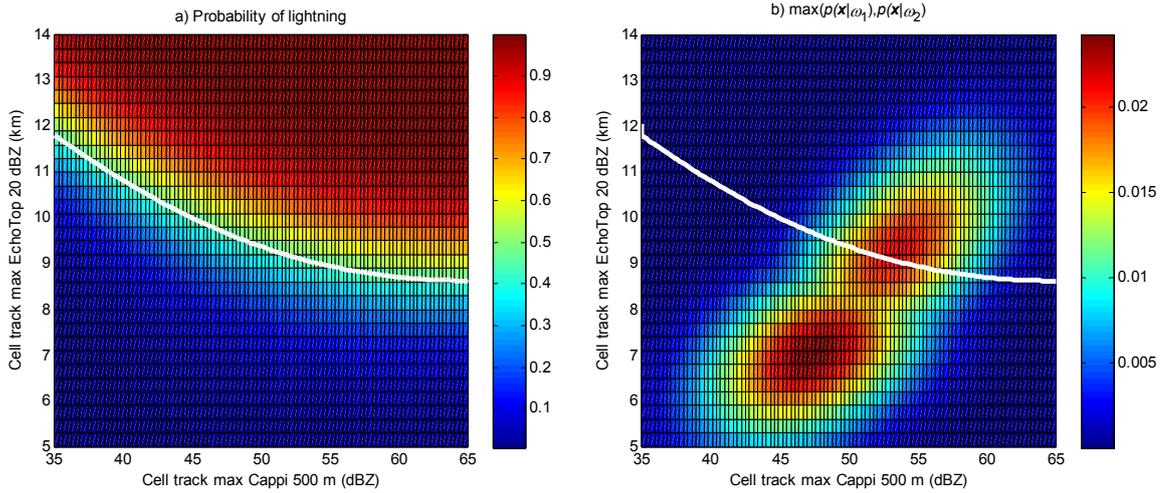


Fig. 34: a) Probability of lightning. b) Maximum of the likelihood functions $p(x|\omega_1)$ and $p(x|\omega_2)$.

The maximum CAPPI 500 m reflectivity has only a small effect on the decision because the optimal decision boundary goes almost horizontally in Fig. 34. Considering the histogram of CAPPI 500 m values in Fig. 32 this was an expected result. A high risk of lightning is obtained roughly when EchoTop 20 dBZ exceeds approximately 9 km.

In order to measure the performance of our model, the error probability of the classification was calculated. By (6.4), we achieved the error probability $P_e = 0.1842$, that is, more than 18 % of all cells are classified incorrectly. Considering the prior probabilities $P(\omega_2) = 0.8$ and $P(\omega_1) = 0.2$, this is a pretty poor result. Almost the same result is achieved by simply assigning all the cells into the class ω_2 . Because a big part of the classes are overlapping and the likelihood ratio $P(\omega_1)/P(\omega_2)$ is large, the used training data have only a small influence on the classification. One should note also that this is only a theoretical estimate for the performance. In order to obtain a more realistic result, the model should be validated with test data that is totally independent on the model training data. Due to the lack of test data, this test was not considered.

When we applied the algorithm only for the cells that exceeded the lifetime of 30 min, the importance of the training data increased remarkably. The prior probabilities were $P(\omega_1) = 0.47$ and $P(\omega_2) = 0.53$, which means that the uncertainty without any measurements is large and the prior knowledge is almost unimportant. The Bayes classification for this data yielded the result in Fig. 35. Since both classes are almost equiprobable, the optimal decision boundary in the feature space goes directly between the likelihood functions in Fig. 35.b. The error probability is $P_e = 0.239$, which is a significant improvement with respect to the prior probabilities. Still, the model classifies more than 23 % percent of cells incorrectly and therefore we may conclude that the classification solely through radar reflectivity factor data is difficult.

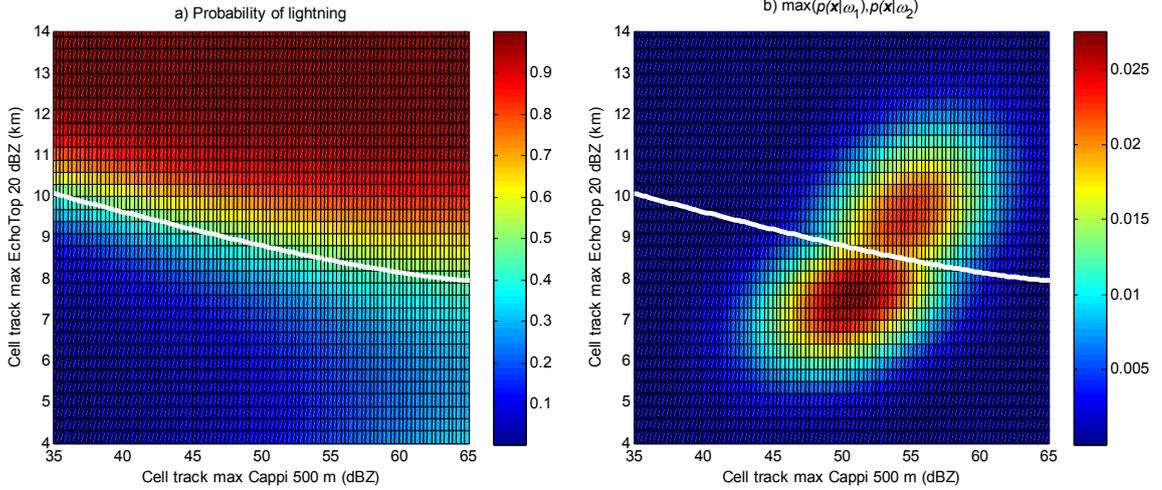


Fig. 35 a) Probability of lightning. b) Maximum of the likelihood functions $p(x|\omega_1)$ and $p(x|\omega_2)$ for the cells exceeding the lifetime of 30 min.

The proposed classification model offers a good basis for the future prospects. As an example, a more realistic assumption on the used probability distribution would increase the margin between the classes and consequently the model performance. Still, the normality assumption makes the computational part of the classification very convenient. In addition, according to Fig. 32.b the distribution of CAPPI 500 m in ω_2 seems to follow truncated normal distribution in which all of the parameter values under 40 dBZ are excluded. Therefore, we conclude that the actual distribution could be normal but due to the inflexible cell identification mechanism the dataset is incomplete. A more efficient and adaptive cell identification mechanism would complete the dataset and improve the classification.

6.4 Fuzzy logic modeling

The fuzzy logic model introduced in Subsection 5.4.5 was tested. The aim of this model is to give an automated guess of the course of storm development and to facilitate the end-user by providing a single informative value to describe the current state of the storm. The model output is represented with a continuous index $-1 \dots 1$, where -1 stands for “clear dissipation” and 1 for “clear intensification”.

The model was tested visually through several test runs. Naturally, it would have been more formal to test the model systematically for example by minimizing a certain error function. However, as the main objective of this fuzzy model is to mimic the human expertise and intuition, the visual validation is a reasonable approach. A detailed description of the fuzzy logic model rules and the applied membership functions is given in Appendix A.

The model performs especially well if it is applied to single-cell storms as represented in the example of Fig. 36. As the figure states, all variables intensifies until 30 min. At 40 min, the storm attains the turning point and the cell area surges while other parameters act more stationary. This appears as a partial decrease in the model output approximately ten

minutes later. At 60 min, all the inputs suggest decreasing and the model interprets this as clear dissipation. The cell dies out 20 min later.

A more complicated example is given in Fig. 37. As expected, this multi-cell system is less easy to analyze than the single-cell example in Fig. 36. Usually the storm parameters of such MCS contain several ups and downs indicating its multi-phase evolution and multi-cell structure. For this reason, the model does not handle the case properly. The system comprises of three individual cells: the first in the very beginning persisting up to 45 min, the second between 45-115 min and the third between 115-155 min.

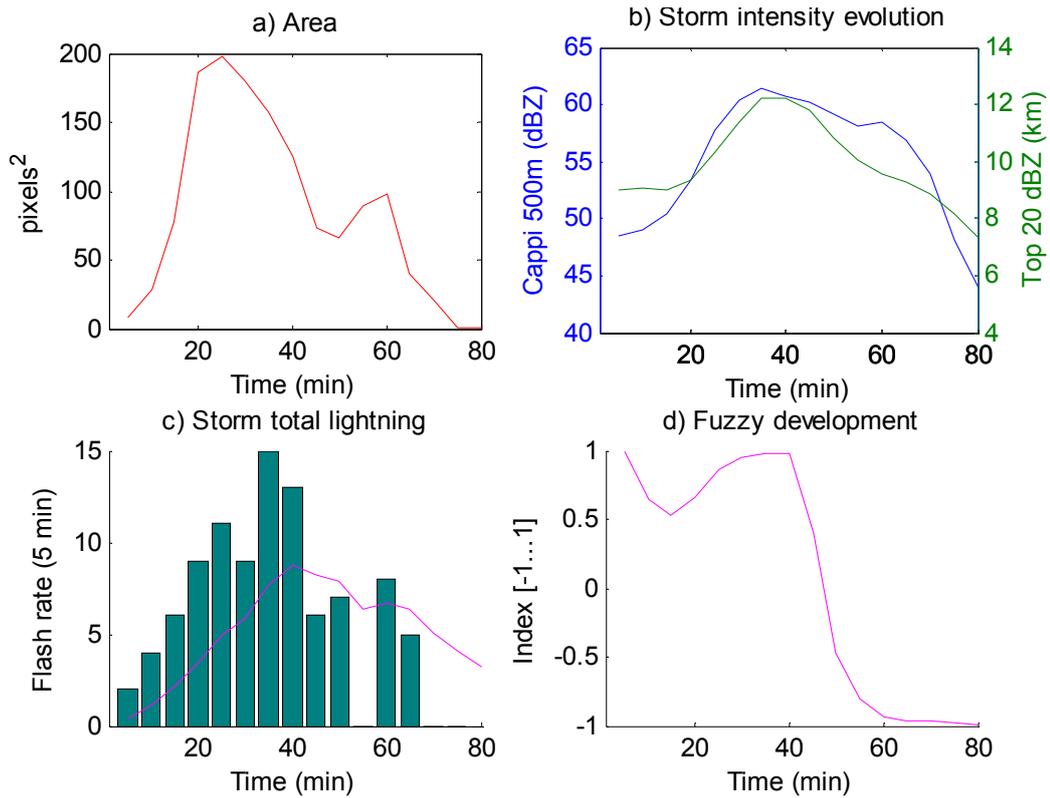


Fig. 36: a) The cell area, b) EchoTop 20 dBZ and CAPPI 500 m values, c) total lightning and d) fuzzy development index of a single-cell convective storm. The fuzzy development index behaves nicely indicating clear dissipation 20 min before the cell dies out.

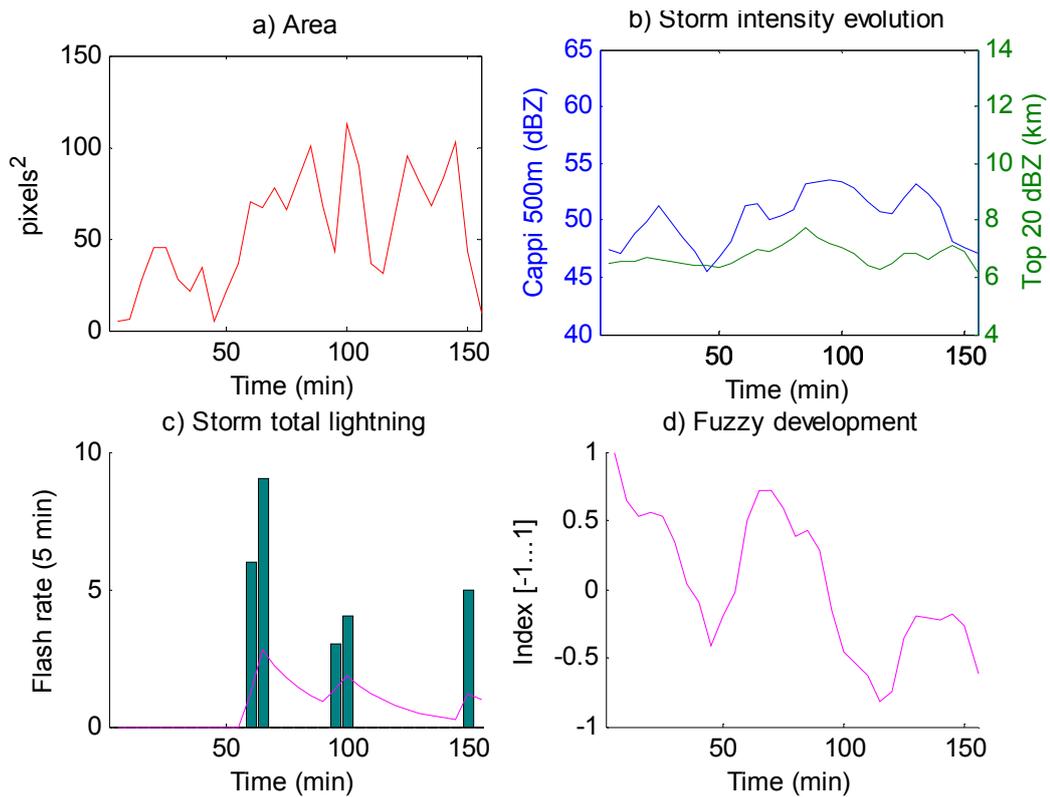


Fig. 37 The cell area, b) EchoTop 20 dBZ and CAPPI 500 m values, c) total lightning and d) fuzzy development index of a complex storm. The outcome of the fuzzy development also more complicated compared to the single-cell storm in Fig. 36.

The quality of weather radar data has also has also effect on the functioning of the fuzzy logic model. Fig. 38 shows the parameters and fuzzy development of a storm located in the periphery of the radar image area. Since the radar contribution volume of radar scan increases with distance, the resolution of EchoTop 20 dBZ values decreases and the resulting times series is usually “flattened”.

The quality of EchoTop 20 dBZ height is poor also in the neighborhood of the radar. The highest possible elevation angle of radar scan is usually too low and only a part of the storm vertical structure is captured. Therefore, the EchoTop 20 dBZ height is undervalued near the radar as represented in Fig. 39.

The parameter CAPPI 500 m is less sensitive to these anomalies and therefore it could be an option in these extreme cases. However, EchoTop 20 dBZ usually seem to provide better information on the storm evolution. A good discussion on the EchoTop quality is given in Delobbe and Holleman (2006).

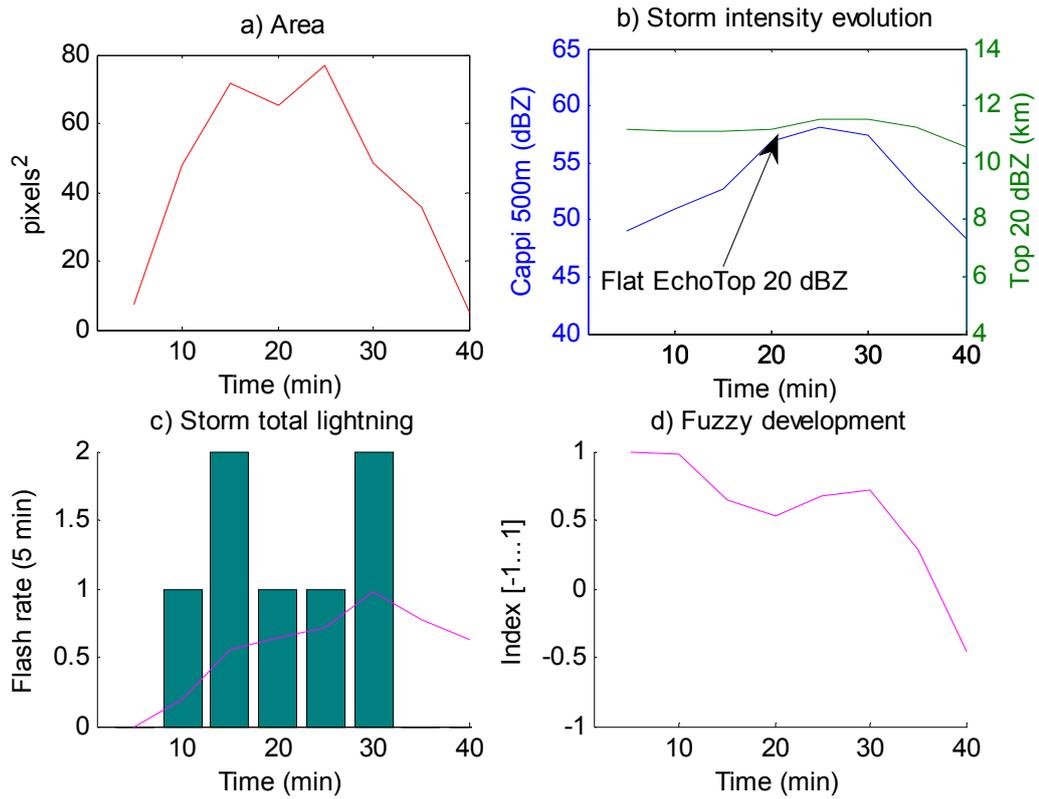


Fig. 38: Behavior of the cell parameters in the periphery of the radar image. The parameter EchoTop 20 dBZ behaves inconsistently

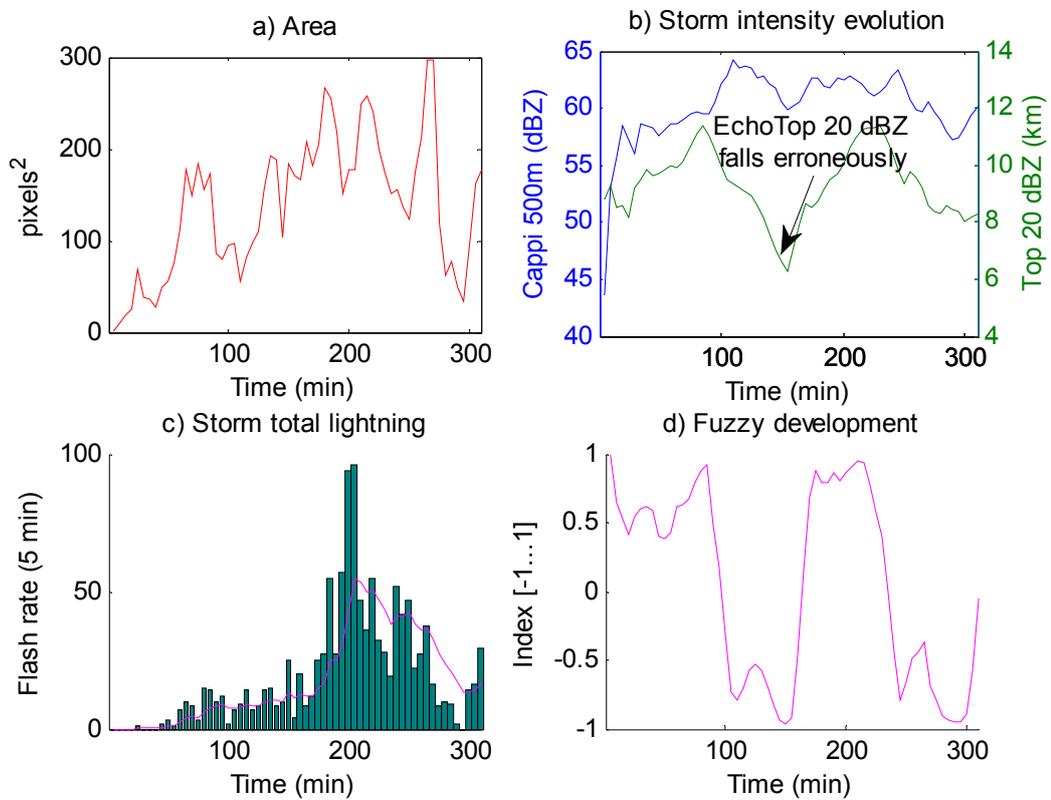


Fig. 39: Behavior of the cell parameters in the neighborhood of the radar. Parameter EchoTop 20 dBZ falls when the cell approaches the radar as the highest radar scan is unable to capture the real EchoTop 20 dBZ value.

Chapter 7: Conclusions

This thesis discussed the potential of computer vision in the nowcasting and life cycle analysis of convective cells. The nowcasting of convective cells is a challenging task and several solutions with varying success have been suggested to solve this awkward problem. This thesis showed the importance of different remote sensing data sources and efficient data processing techniques in the convective cell nowcasting scenario. The presented methods in this thesis provide valuable tools for convective cell monitoring, life cycle analysis and prediction.

The primary achievement of this thesis is a clustering based algorithm for simultaneous tracking of multiple convective cells. Even though several authors have proposed different convective cells tracking methods, the presented study shows significant improvements. At first, the tracking is consolidated by fusing lightning and weather radar data. In many algorithms in the literature, the tracking is performed solely with reflectivity pattern while the important information given by lightning is ignored. Secondly, the use of multiple consecutive frames provides the continuity of the tracking even in the presence of occasional missing radar frames. On the basis of several test runs, the functioning of the algorithm is robust even in the presence of errors in data. This is an important asset, especially considering possible operative applications, because radar data is susceptible to different errors, such as attenuation and radar malfunctioning

Thirdly, the presented algorithm exploits efficiently the spatial and temporal structure of the data. In many cell tracking algorithms (e.g. Hering et al. 2004; Novák and Kyzanrová 2006), the smallest objects are filtered out as noise by rejecting objects having a size less than a certain threshold. Ignoring smallest objects is reasonable, since they are frequently produced by non-convective phenomena such as ships or sea clutter. However, such a methodology may be too strict and it may remove objects that are crucial for the tracking. A better approach is to consider a small object as a border object, which extends the tracking, especially in the beginning and in the end of a track. Before filtering out the smallest objects, we should study the spatial and temporal structure of the data for example through the density-based clustering. The main reasons for this are as follows:

1. A small piece of cell can lie in the vicinity of a larger cell and hence it can be a part of a larger entity. Thus, prior to ignoring the small areas, one should examine whether the cell can be reasonably linked to another cell.
2. Small cells can be either in the beginning or end of their life cycles. Therefore, we will extend the tracking by taking into account the smallest objects as border objects. By simply ignoring the smallest areas, the tracks are shorter.
3. A small object may act as an important link within a track. Therefore, this is regarded as a way of making the algorithm more robust.

4. Only objects with an infeasible spatial and temporal structure should be ignored. This implies that the size of an object should not be considered as the only feature to determine noise. For example, sea clutter resembles convective cells occasionally in individual images. On the other hand, the occurrence of sea clutter is very random. Therefore, the clutter should be ignored as it does not have a suitable spatial and temporal structure to produce a good track.

Convective cell life cycle analysis and behavior was studied through the designed tracking algorithm. According to the average normalized evolution, convective cells tend to follow the hypothetical life cycle viewed in the literature. Usually, convective cells are characterized by clear intensification, maturation and dissipation phase, which is observed as the development of different cell parameters. However, complex MCSs are more difficult to analyze and they have several intensification and development phases. On average, long-lived MCSs are more intense and dangerous than short-lived single-cell storms.

One of the most characteristic features of convective storms is lightning. The relationship between lightning and radar parameters was studied through a two dimensional Bayesian classification model. The proposed classification approach gives encouraging results and different radar reflectivity parameters improve lightning prediction and classification. In addition, the classification method provides an important tool for different end-users such as aviation and industry.

A novel prototype fuzzy logic model was designed for convective cell life cycle analysis. With an ideal single-cell storm, the model performs well and identifies the cell life cycle phases. With a more complex MCS, the performance can be weak. Overall, the prediction of cell development is a difficult task. However, new expert-based features describing the storm state can be easily added to improve the model and therefore the model is a promising starting point for the future.

7.1 Future improvements

The proposed methods provide a considerable basis for the future prospects. Still, improvements are needed to reach a comprehensive knowledge on convective cell behavior. An important result of the work conducted herein is the identification of the main bottlenecks in the nowcasting of convective cells. The following discussion covers the desirable future improvements to overcome the problems encountered in the convective cell nowcasting:

1. A robust and adaptive cell identification method. Since the applied cell identification method exploits only single reflectivity threshold, it offers a very inflexible solution; the cell is observed only during the existence of reflectivity factor values exceeding 40 dBZ and therefore only part of its life cycle is captured. Inflexible identification also distorts the storm statistics. An ideal identification algorithm would recognize the cells even in the early stage and cover not only a region of a certain threshold but also the whole reflectivity

pattern included in the cell. A better identification would also serve the performance of the tracking algorithm.

2. Better knowledge on the phenomena, such as convergence lines, occurring in the boundary layer. By an adequate boundary layer detection mechanism, we could predict the initiation of new storms (Wilson et al. 1998).
3. More efficient use of storm vertical structure. Vertical structure of the storm includes plenty of information on the life cycle phases and hence that information would be an advance in the fuzzy logic model. In this thesis, the only parameter describing the vertical structure is EchoTop 20 dBZ.
4. Improvement in radar data quality. In this thesis, only single weather radar is considered, which naturally has effect on the tracking as well as on the fuzzy logic modeling and the cell statistics. For example, FMI's Doppler weather radar network consists of 8 radars (FMI 2008) allowing the use of composite weather radar data. The use of this composite data would be highly beneficial regarding the data quality. In addition, quality information could be included in the composite data by a quality weighted composition approach (Peura and Koistinen 2007).

Additionally, to improve the lightning classification, we would also need more representative features to decrease the overlapping between the distributions of the classes ω_2 and ω_1 . As stated earlier in Section 2.2, the prevailing consensus is that lightning is related to the presence of certain type of hydrometeors, such as hail and graupel. Therefore a better knowledge on the hydrometeor content of the storm could be highly beneficial. The use of more advanced instruments, such as dual polarization radar (e.g. Rinehart 2004), could be a key to better results.

Appendix A: Fuzzy logic model

This appendix introduces the membership functions and the rules utilized in the designed fuzzy logics model of Subsection 5.4.5. The following parameters are taken into account in the model (see Subsection 5.4.5):

1. Cell area change ratio.
2. 20 min EchoTop 20 dBZ average difference.
3. Duration of continuous EchoTop 20 dBZ increasing/decreasing.
4. Beginning of lightning.
5. Duration of lightning increasing/decreasing.

In here, centre of gravity is considered as the defuzzification method. Fuzzy AND is evaluated by the min-norm and fuzzy OR by the max-norm (5.30). All the membership functions of the input variables are represented in Fig. 40.

Next, we will introduce an auxiliary fuzzy variable *state* that defines the state of EchoTop 20 dBZ development or alternatively the state of lightning development at time t . The membership functions of this parameter are given in Fig. 41. The parameter *state* is calculated with the auxiliary fuzzy model. The input values of the auxiliary model are the duration of continuous EchoTop 20 dBZ increasing/decreasing or the duration of lightning increasing/decreasing. In addition, the auxiliary model takes the previous state as an input variable. The output parameter *state* describes if the cell EchoTop 20 dBZ or lightning is increasing or decreasing. As an example, in order to change state from “increase” to “decrease” continuous duration of EchoTop 20 dBZ has to be “negative” and the previous state has to be “increase”.

The rules of the auxiliary model are given in Table 2, where the parameter *input* refer the duration EchoTop 20 dBZ increasing/decreasing or the duration of lightning increasing/decreasing.

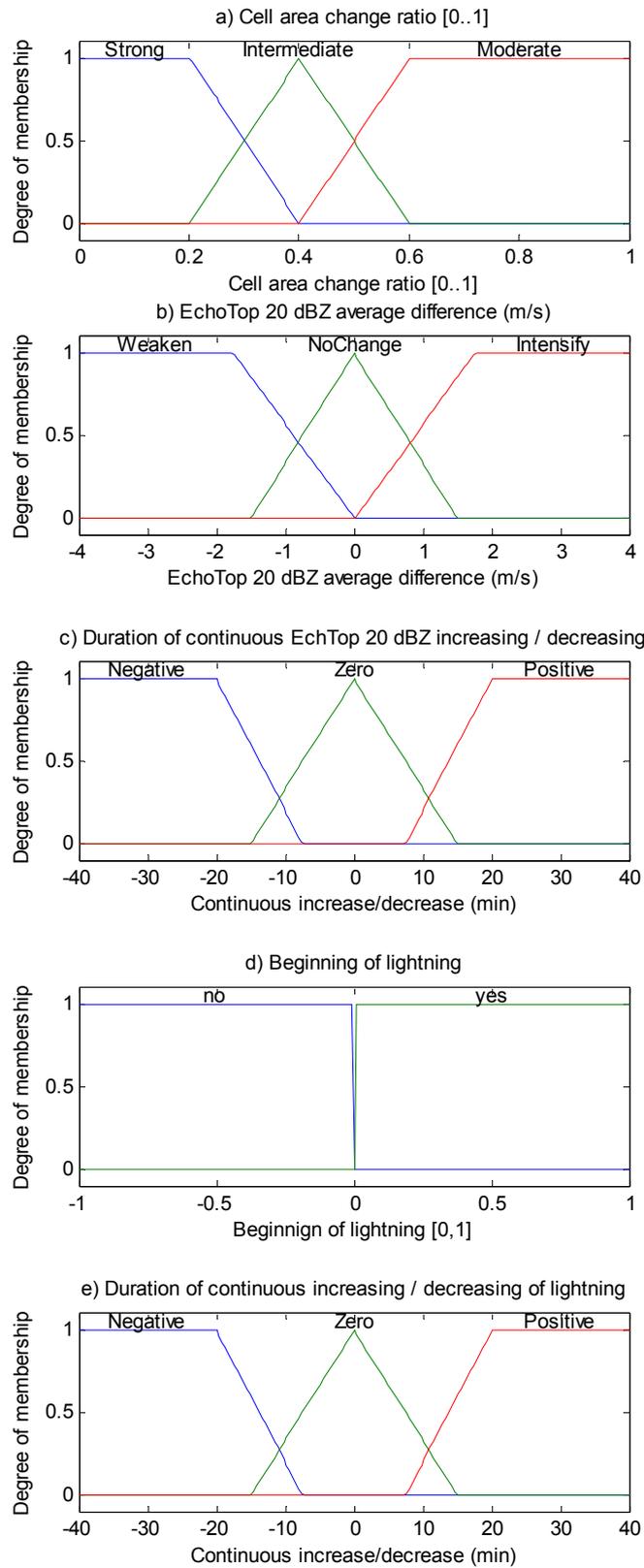


Fig. 40: Membership functions of the input variables listed above. Note that Beginning of lightning is actually a crisp binary value.

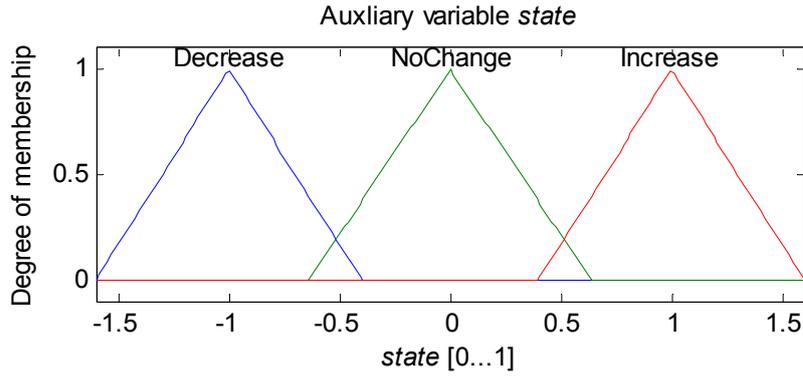


Fig. 41: Membership functions of the auxiliary model output

Table 2: Rules of the auxiliary fuzzy model.

1.	IF (<i>input</i> is Positive and <i>state</i> (<i>t</i> -1) is Decrease) THEN (<i>state</i> (<i>t</i>) Increase)
2.	IF (<i>input</i> is Negative and <i>state</i> (<i>t</i> -1) is Increase) THEN (<i>state</i> (<i>t</i>) Decrease)
3.	IF (<i>input</i> is Negative and <i>state</i> (<i>t</i> -1) is Decrease) THEN (<i>state</i> (<i>t</i>) Decrease)
4.	IF (<i>input</i> is Positive and <i>state</i> (<i>t</i> -1) is Increase) THEN (<i>state</i> (<i>t</i>) Increase)
5.	IF (<i>input</i> is Positive and <i>state</i> (<i>t</i> -1) is No Change) THEN (<i>state</i> (<i>t</i>) Increase)
6.	IF (<i>input</i> is Negative and <i>state</i> (<i>t</i> -1) is No Change) THEN (<i>state</i> (<i>t</i>) Decrease)
7.	IF (<i>input</i> is Zero and <i>state</i> (<i>t</i> -1) is Decrease) THEN (<i>state</i> (<i>t</i>) Decrease)
8.	IF (<i>input</i> is Zero and <i>state</i> (<i>t</i> -1) is Increase) THEN (<i>state</i> (<i>t</i>) Increase)
9.	IF (<i>input</i> is Zero and <i>state</i> (<i>t</i> -1) is No Change) THEN (<i>state</i> (<i>t</i>) No Change)

After the variable *state* is obtained for both continuous duration of EchoTop 20 dBZ, we will estimate the final model output variable *development* with the fuzzy model given in Table 3. This model can be viewed as the actual expert reasoning model, whereas the auxiliary model preprocesses the input variables to the expert modeling. The membership functions describing the fuzzy output is given in Fig. 42.

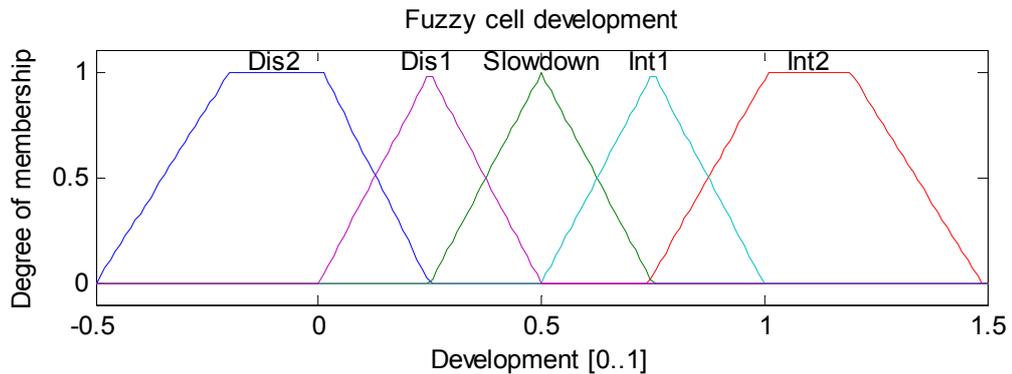


Fig. 42: Fuzzy cell development [0...1]. In the figure abbreviations Dis2 stand for “Clear Dissipation”, Dis1 for “Dissipation”, Int1 for “Intensification” and Int2 for “Clear Intensification”.

Table 3: Rules of the actual fuzzy logic model.

1.	IF (EchoTop 20 dBZ state is Decrease) and (EchoTop 20 dBZ average difference is Weaken) THEN (development is Clear Dissipation)
2.	IF (EchoTop 20 dBZ state is Decrease) or (EchoTop 20 dBZ average difference is Weaken) or (lightning state is Decrease) THEN (development is Dissipate)
3.	IF (EchoTop 20 dBZ state is Increase) and (EchoTop 20 dBZ average difference is Intensify) THEN (development is Clear Intensification)
4.	IF (EchoTop 20 dBZ state is Increase) or (EchoTop 20 dBZ average difference is Intensify) or (lightning state is Increase) THEN (development is Intensify)
5.	IF (EchoTop 20 dBZ state is No Change) and (EchoTop 20 dBZ average difference is No Change) THEN (development is Slowdown)
6.	IF (1stLightning is yes) THEN (development is Clear Intensification)
7.	IF (EchoTop 20 dBZ average difference is Weaken) and (lightning state is Decrease) THEN (development is Clear Dissipation)
8.	IF (EchoTop 20 dBZ average difference is Intensify) and (lightning state is Increase) THEN (development is Clear Intensification)
9.	IF (EchoTop 20 dBZ average difference is No Change) and (lightning state is NoChange) THEN (development is Slowdown)
10.	IF (Area change ratio is Strong) THEN (development is Clear Dissipation)
11.	IF (Area change ratio is Intermediate) THEN (development is Dissipate)

Appendix B: Bayesian classification for two multinormal distributions

This appendix includes the derivation of the optimal decision boundary for two normally distributed classes. As stated in Subsection 6.3.4, statistically optimal decision boundary is obtained if we partition the feature space into the decision regions R_1 and R_2 such that

$$\begin{aligned} R_1 &: P(\omega_1 | \mathbf{x}) > P(\omega_2 | \mathbf{x}) \\ R_2 &: P(\omega_1 | \mathbf{x}) < P(\omega_2 | \mathbf{x}), \end{aligned} \quad (\text{B.1})$$

where $P(\omega_i | \mathbf{x}) = p(\mathbf{x} | \omega_i)P(\omega_i) / p(\mathbf{x})$ is the posterior probability of the class ω_i . However, the term $p(\mathbf{x})$ can be ignored because it does not have effect on the classification.

The boundary where the posterior distributions are equal between the i th and the j th class is denoted as the *decision boundary* g_{ij}

$$g_{ij} = \{ \mathbf{x} | g_i - g_j = 0 \} = \{ \mathbf{x} | P(\omega_i | \mathbf{x}) - P(\omega_j | \mathbf{x}) = 0 \}, \quad (\text{B.2})$$

where the function g_i denotes the *discrimination function* of the i th class. Consider now that the likelihood $p(\mathbf{x} | \omega_i)$ for the i th class is the N -dimensional multivariate normal distribution

$$p(\mathbf{x} | \omega_i) = \frac{1}{(2\pi)^{N/2} |\Sigma_i|^{1/2}} e^{-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)}, \quad (\text{B.3})$$

where $\boldsymbol{\mu}_i$ is the estimated mean and Σ_i is the estimated covariance matrix. Instead of using the posterior probability $P(\omega_i | \mathbf{x})$ itself, we may apply any monotonic function to the posterior probabilities in (B.1). Since the normal distribution has the exponential term, it is convenient to apply the natural logarithm as

$$\begin{aligned} g_i(\mathbf{x}) &= \ln(P(\omega_i)p(\mathbf{x} | \omega_i)) \\ &= -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) + \ln(P(\omega_i)) - \frac{1}{2} \ln(|\Sigma_i|) - \frac{N}{2} \ln(2\pi). \end{aligned} \quad (\text{B.4})$$

The last term is independent on the class hence it can be removed from the discrimination function. Therefore, g_i can be refined as follows

$$g_i(\mathbf{x}) = -\frac{1}{2}(\mathbf{x}^T \Sigma_i^{-1} \mathbf{x} - \boldsymbol{\mu}_i^T \Sigma_i^{-1} \mathbf{x} - \mathbf{x}^T \Sigma_i^{-1} \boldsymbol{\mu}_i + \boldsymbol{\mu}_i^T \Sigma_i^{-1} \boldsymbol{\mu}_i) + \ln(P(\omega_i)) - \frac{1}{2} \ln(|\Sigma_i|). \quad (\text{B.5})$$

Since the covariance matrix Σ_i is symmetric, the inverse covariance matrix Σ_i^{-1} is also symmetric and $\boldsymbol{\mu}_i^T \Sigma_i^{-1} \mathbf{x} = \mathbf{x}^T \Sigma_i^{-1} \boldsymbol{\mu}_i$. Therefore, we obtain

$$\begin{aligned} g_i(\mathbf{x}) &= -\frac{1}{2} \mathbf{x}^T \Sigma_i^{-1} \mathbf{x} + \boldsymbol{\mu}_i^T \Sigma_i^{-1} \mathbf{x} - \frac{1}{2} (\boldsymbol{\mu}_i^T \Sigma_i^{-1} \boldsymbol{\mu}_i + \ln(|\Sigma_i|)) + \ln(P(\omega_i)) \\ &= \mathbf{x}^T \mathbf{A}_i \mathbf{x} + \mathbf{b}_i^T \mathbf{x} + c_i, \end{aligned} \quad (\text{B.6})$$

where

$$\begin{aligned}\mathbf{A}_i &= -\frac{1}{2}\Sigma_i^{-1}, \\ \mathbf{b}_i &= \boldsymbol{\mu}_i^T \Sigma_i^{-1}, \\ \mathbf{c}_i &= -\frac{1}{2}\left(\boldsymbol{\mu}_i^T \Sigma_i^{-1} \boldsymbol{\mu}_i + \ln(|\Sigma_i|)\right) + \ln(P(\omega_i)).\end{aligned}\tag{B.7}$$

Now we can rewrite the optimal decision boundary g_{ij} as follows

$$g_{ij}(\mathbf{x}) = g_i(\mathbf{x}) - g_j(\mathbf{x}) = \mathbf{x}^T (\mathbf{A}_i - \mathbf{A}_j) \mathbf{x} + (\mathbf{b}_i - \mathbf{b}_j) \mathbf{x} + (c_i - c_j) = \mathbf{x}^T \mathbf{A} \mathbf{x} + \mathbf{b} \mathbf{x} + c = 0. \tag{B.8}$$

As (B.8) states, the optimal decision boundary is of quadratic form. However, if covariance matrices are equal for both classes, then $\mathbf{A}_i - \mathbf{A}_j = 0$ and the decision boundary is a linear hyperplane in the feature space.

References

- AUER, J. 2003, *Weather derivatives heading for sunny times*, Deutsche Bank Research, Frankfurt am Main.
- BARCLAY, P.A. and WILK, K.E., 1970. Severe thunderstorm radar echo motion and related weather events hazardous to aviation operations. Norman, Oklahoma, United States: National Severe Storms Laboratory.
- BECKMANN, N., BEGEL, H., SCHNEIDER, R. and SEEGER, B., 1990. The R*-tree: an Efficient and Robust Access Method for Points and Rectangles, *Proc. ACM SIGMOD Int. Conf. on Management of Data*. 1990, pp322-331.
- BELLMAN, R.E., 1962. Applied dynamic programming. Princeton (NJ): Princeton University Press.
- BERTHOLD, M. and HAND, D.J., 1999. Intelligent data analysis : an introduction. Berlin: Springer.
- BEVINGTON, P.R., 1969. Data reduction and error analysis for the physical sciences.
- BISHOP, C.M., 1995. Neural networks for pattern recognition. Oxford: Clarendon.
- BLACKMER, R.H., DUDA, R.O. and REBOH, E., 1973. Application of pattern recognition to digitized Weather Radar Data.
- BYERS, H.R. and BRAHAM, R.R., JR., 1949. The thunderstorm : report of the thunderstorm project. Washington.
- CHATTOPADHYAY, S., 2006. Soft Computing Techniques in combating the complexity of the atmosphere- a review.
- DELOBBE, L. and HOLLEMAN, I., 2006. Uncertainties in radar echo top heights used for hail detection. *Meteorological Applications*, **13**(4), pp. 361-374.
- DIXON, M. and WIENER, G., 1993. TITAN: Thunderstorm Identification, Tracking, Analysis, and Nowcasting—A Radar-based Methodology. *Journal of Atmospheric and Oceanic Technology*, **10**(6), pp. 785-797.
- DOUGLAS, D.H. and PEUCKER, T.K., 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Canadian Cartographer*, **10**, pp. 112-122.
- E. R. JAYARATNE, C. P. R. SAUNDERS, J. HALLETT, 1983. Laboratory studies of the charging of soft-hail during ice crystal interactions. *Quarterly Journal of the Royal Meteorological Society*, **109**(461), pp. 609-630.

ESTER, M., KRIEGEL, H., S, J. and XU, X., 1996. A density-based algorithm for discovering clusters in large spatial databases with noise, 1996, AAAI Press pp226-231.

FINNISH METEOROLOGICAL INSTITUTE, 2008. FMI radar network. Available: http://www.fmi.fi/weather/rain_9.html.

GONZALEZ, R.C., WOODS, R.E., 2002. Digital image processing. Upper Saddle River, NJ: Prentice Hall.

HERING, A.M., MOREL, G., C., SENESI, S., AMBROSETTI, P. and BOSCACCI, M., 2004. Nowcasting thunderstorms in the Alpine region using a radar based adaptive thresholding scheme, *Proc. of the 3th European Conf. on Radar Meteorology (ERAD 2004)*, 2004, pp1-6, Visby, Sweden

HOLTON, J.R., 2004. An introduction to dynamic meteorology. Elsevier: Amsterdam.

ISARD, M. and BLAKE, A., 1998. CONDENSATION – Conditional Density Propagation for Visual Tracking. *International Journal of Computer Vision*, **29**, pp. 5-28.

JAIN, R., KASTURI, R. and SCHUNCK, B.G., 1995. Machine vision. New York (NY): McGraw-Hill.

JOHNSON, J.T., MACKEEEN, P.L., WITT, A., MITCHELL, E.D., STUMPF, G.J., EILTS, M.D. and THOMAS, K.W., 1998. The Storm Cell Identification and Tracking Algorithm: An Enhanced WSR-88D Algorithm. *Weather and Forecasting*, **13**(2), pp. 263-276.

JOLLIFFE, I.T., 2002. Principal component analysis. New York: Springer.

KARRAY, F. and DE SILVA, C., 2004. Soft computing and intelligent systems design : theory, tools and applications. New York: Pearson/Addison Wesley.

KOIVO, H., 2000. AS-74.115 Soft Computing in Dynamical Systems.

KOLESNIKOV, A., 2003. *Efficient algorithms for vectorization and polygonal approximation*. Ph. D. thesis, Joensuu Yliopisto: Joensuu.

LEEKWIJCK, W.V. and KERRE, E.E., 1999. Defuzzification: criteria and classification. *Fuzzy Sets and Systems*, **108**(2), pp. 159-178.

LEESE, J.A., NOVAK, C.S. and CLARK, B.B., 1971. An Automated Technique for Obtaining Cloud Motion from Geosynchronous Satellite Data Using Cross Correlation. *Journal of Applied Meteorology*, **10**(1), pp. 118-132.

LI, L., SCHMID, W. and JOSS, J., 1995. Nowcasting of Motion and Growth of Precipitation with Radar over a Complex Orography. *Journal of Applied Meteorology*, **34**(6), pp. 1286-1300.

LUCAS, B.D. and KANADE, T., 1981. An iterative image registration technique with an application to stereo vision, *Proc. Seventh International Joint Conference on Artificial Intelligence*, 1981, pp674–679.

MACGORMAN, D.R. and RUST, D.W., 1998. The electrical nature of storms. New York: Oxford University Press.

MÄKELÄ, A., 2006. *Ukkonen sääilmiönä sekä vertailu säätuoka- ja salamanpaikanninhavaintojen välillä Suomessa kesällä 2005. Pro gradu tutkielma*, Helsingin Yliopisto: Helsinki.

MALALA, R., 2006. The Impact of Weather On Aircrafts Accidents, *Proc. of the 2006 EUMETSAT Meteorological Satellite Conference*, 2006, EUMETSAT.

MITRA, S.K., cop. 2006. Digital signal processing : a computer based approach. New York: McGraw-Hill Higher Education.

MUELLER, C., SAXEN, T., ROBERTS, R., WILSON, J., BETANCOURT, T., DETTLING, S., OIEN, N. and YEE, J., 2003. NCAR Auto-Nowcast System. *Weather and Forecasting*, **18**(4), pp. 545-561.

NOVÁK, P. and KYZNAROVÁ, H., 2006. Cell-oriented forecasts of Czech weather radar data, *Proc. of the 4th European Conf. on Radar Meteorology (ERAD 2006)*, 2006, Barcelona, Spain

PAPADIMITRIOU, C.H. and STEIGLITZ, K., 1998. Combinatorial optimization : algorithms and complexity. Mineola (NY): Dover.

PEURA, M. and KOISTINEN, J., 2007. Using radar data quality in computing composites and nowcasting products, *33rd Conference on Radar Meteorology*, 2007, Cairns, Australia.

PEURA, M. and HOHTI, H., 2004. Optical flow in radar images, *Proc. of the 3th European Conf. on Radar Meteorology (ERAD 2004)*, 2004, pp. 454-458, Visby, Sweden

PIERCE, C.E., HARDAKER, P.J., COLLIER, C.G. and HAGGETT, C.M., 2000. GANDOLF: a system for generating automated nowcasts of convective precipitation. *Meteorological Applications*, **7**, pp. 341-360.

PUHAKKA, T., 2000. Tutkameteorologian perusteet. Helsinki: Helsingin yliopisto.

PUHAKKA, T., 1995. Pilvifysiikka. Helsingin yliopisto, Meteorologian laitos: Helsinki.

RAKOV, V.A., 2006. Lightning : physics and effects. Cambridge: Cambridge University Press.

RAUBER, R.M., 2005. Severe and hazardous weather : an introduction to high impact meteorology. Kendall / Hunt: Dubuque, IA.

RINEHART, R.E. and GARVEY, E.T., 1978. Three-dimensional storm motion detection by conventional weather radar. *Nature*, **273**, pp. 287–289.

RINEHART, R.E., cop. 2004. Radar for meteorologists : or you, too, can be a radar meteorologist. Part 3. Nevada, MO: Rinehart Publications.

ROSSI, P. and MÄKELÄ, A., 2008. A clustering-based tracking method for convective cell identification and analysis, *Proc. of the 5th European Conf. on Radar Meteorology (ERAD 2008)*, 2008, Helsinki: the Finnish Meteorological Institute.

RUOSTEENOJA, K., 1996. Meteorologian perusteet. Helsinki: Helsingin yliopisto, Meteorologian laitos.

SALTIKOFF, E., 2008. Personal communication on Nov 11th 2008.

SANDER, J., ESTER, M., KRIEGEL, H. and XU, X., 1998. Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications. *Data Mining and Knowledge Discovery*, **2**(2), pp. 169-194.

SATO, Y., 1992. Piecewise Linear Approximation of Plane Curves by Perimeter Optimization. *Pattern Recognition*, **25**(12), pp. 1535-1543.

SEED, A.W., 2003. A Dynamic and Spatial Scaling Approach to Advection Forecasting. *Journal of Applied Meteorology*, **42**(3), pp. 381-388.

SETHI, I.K. and JAIN, R., 1987. Finding trajectories of feature points in a monocular image sequence. *IEEE Trans. Pattern Anal. Mach. Intell.*, **9**(1), pp. 56-73.

SKLANSKY, J., CHAZIN, R.L. and HANSEN, B.J., 1972. Minimum Perimeter Polygons of Digitized Silhouettes. *IEEE Transactions on Computers*, **21**(3), pp. 260-268.

SONKA, M., 2007. Image processing, analysis, and machine vision. Mason, OH: Thomson.

STEINER, M., HOUZE, R.A. and YUTER, S.E., 1995. Climatological Characterization of Three-Dimensional Storm Structure from Operational Radar and Rain Gauge Data. *Journal of Applied Meteorology*, **34**(9), pp. 1978-2007.

THEODORIDIS, S. and KOUTROUMBAS, K., 2003. Pattern recognition. London: Academic Press.

TSONIS, A.A. and AUSTIN, G.L., 1981. An evaluation of extrapolation techniques for the short-term prediction of rain amounts. **19**, pp. 54-56.

TUOMI, T.J., 1993. Ukkonen ja salamat. Helsinki: Tähtitieteellinen yhdistys Ursa.

TUOMI, T.J. and MÄKELÄ, A., 2007. Lightning observations in Finland 2007. Helsinki: Ilmatieteen laitos.

VEENMAN, C.J., REINDERS, M.J.T. and BACKER, E., 2001. Resolving motion correspondence for densely moving points. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **23**, pp. 54-72.

WIKIPEDIA, 2008a. Constant Altitude Plan Position Indicator. Available: <http://en.wikipedia.org/wiki/CAPPI>

WIKIPEDIA, 2008b. Log-normal distribution. Available: http://en.wikipedia.org/wiki/Lognormal_distribution

WIKIPEDIA, 2008c. Occam's razor, Available: http://en.wikipedia.org/wiki/Occam's_Razor.

WILSON, J.W., CROOK, N.A., MUELLER, C.K., SUN, J. and DIXON, M., 1998. Nowcasting Thunderstorms: A Status Report. *Bulletin of the American Meteorological Society*, **79**(10), pp. 2079-2099.

YEUNG, L.H.Y., LAI, E. S. T. and CHIU, S. K. S., 2007. Lightning Initiation and Intensity Nowcasting based on Isothermal Radar Reflectivity - A Conceptual Model, *Proc. 33rd Conference on Radar Meteorology*, 2007, Cairns, Australia.

YILMAZ, A., JAVED, O. and SHAH, M., 2006. Object tracking: A survey. *ACM Comput.Surv.*, **38**(4), pp. 13.

ZADEH, L.A., 1968. Fuzzy algorithms. *Information and Control*, **12**(2), pp. 94-102.

ZIPSER, E.J., 1982. Use of a conceptual model of the life cycle of mesoscale convective systems to improve very-short-range forecasts. *Nowcasting*, , pp. 191–204.