

University of Zagreb

FACULTY OF ELECTRICAL ENGINEERING AND COMPUTING

Borna Bićanić

STATE ESTIMATION AND MULTIPLE MOVING OBJECT TRACKING ON RIEMANNIAN MANIFOLDS

DOCTORAL THESIS

Zagreb, 2023



University of Zagreb

FACULTY OF ELECTRICAL ENGINEERING AND COMPUTING

Borna Bićanić

STATE ESTIMATION AND MULTIPLE MOVING OBJECT TRACKING ON RIEMANNIAN MANIFOLDS

DOCTORAL THESIS

Supervisor: Academic Professor Ivan Petrović, F.C.A.

Zagreb, 2023



Sveučilište u Zagrebu

FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Borna Bićanić

ESTIMACIJA STANJA I PRAĆENJE VIŠE GIBAJUĆIH OBJEKATA NA RIEMANNOVIM MNOGOSTRUKOSTIMA

DOKTORSKI RAD

Mentor: Akademik prof. dr. sc. Ivan Petrović

Zagreb, 2023.

Doctoral thesis was written at the University of Zagreb, Faculty of Electrical Engineering and Computing, Departement of Control and Computer Engineering.

Supervisor: Academic Professor Ivan Petrović, F.C.A.

Thesis contains 123 pages

Thesis no.:

About the supervisor

Ivan Petrović received B.Sc., M.Sc. and Ph.D. degrees in electrical engineering from the University of Zagreb, Faculty of Electrical Engineering and Computing (FER), Zagreb, Croatia, in 1983, 1989 and 1998, respectively.

For the first ten years after graduation he was with the Institute of Electrical Engineering of Končar Corporation in Zagreb, where he had been working as a research and development engineer for control and automation systems of electrical drives and industrial plants. From 1994 he has been working at the Department of Control and Computer Engineering at FER, where he is currently a Full Professor with tenure. Since May 19, 2022 he has been a fellow of the Croatian Academy of Sciences and Arts. He has actively participated as a collaborator or principal investigator on more than 40 national and 25 international scientific projects, where from them ten are funded form FP7 and Horizon 2020 and Horizon Europe framework programmes. He is also co-director of the Centre of Research Excellence for Data Science and Cooperative Systems. He published more than 80 papers in scientific journals and 220 papers in proceedings of international conferences in the area of control engineering and automation applied to control mobile robots and vehicles, power systems, electromechanical systems and other technical systems.

Professor Petrović is a member of IEEE, Croatian Academy of Engineering (HATZ), chair of the Technical committee on Robotics of the International Federation of Automatic Control (IFAC), a permanent board member of the European Conference on Mobile Robots, an executive committee member of the Federation of International Robot-soccer Association (FIRA), and a founding member of the iSpace Laboratory Network. He is also a member of the Croatian Society for Communications, Computing, Electronics, Measurements and Control (KoREMA) and Editor-in-Chief of the Automatika journal and Associate Editor of the Mechatronics journal. He received the award "Professor Vratislav Bedjanič" in Ljubljana for outstanding M.Sc. thesis in 1990 and silver medal "Josip Lončar" from FER for outstanding Ph.D. thesis in 1998. For scientific achievements he received the award "Rikard Podhorsky" from the Croatian Academy of Engineering (2008), "National Science Award of the Republic of Croatia" (2011), the gold plaque "Josip Lončar" (2013), "Science Award" from FER (2015), award of the Croatian Academy of Sciences and Arts (2017), award "Nikola Tesla" from IEEE Croatia Section (2019) and award of the City of Zagreb (2022).

O mentoru

Ivan Petrović diplomirao je, magistrirao i doktorirao u polju elektrotehnike na Sveučilištu u Zagrebu Fakultetu elektrotehnike i računarstva (FER), 1983., 1989. odnosno 1998. godine.

Prvih deset godina po završetku studija radio je na poslovima istraživanja i razvoja sustava upravljanja i automatizacije elektromotornih pogona i industrijskih postrojenja u Končar - Institutu za elektrotehniku. Od svibnja 1994. radi u Zavodu za automatiku i računalno inženjerstvo FER-a, gdje je sada redoviti profesor u trajnome zvanju. Od 19. svibnja 2022. redoviti je član Hrvatske akademije znanosti i umjetnosti. Sudjelovao je ili sudjeluje kao suradnik ili voditelj na više od 40 domaćih i 25 međunarodnih znanstvenih projekata, od čega deset projekata iz programa FP7 i Obzor 2020 i Obzor Europa. Nadalje, suvoditelj je Znanstvenog centra izvrsnosti za znanost o podatcima i kooperativne sustave. Objavio je više od 80 znanstvenih radova u časopisima i 220 znanstvenih radova u zbornicima skupova u području automatskog upravljanja i estimacije s primjenom u upravljanju mobilnim robotima i vozilima te energetskim, elektromehaničkim i drugim tehničkim sustavima.

Prof. Petrović član je stručne udruge IEEE, Akademije tehničkih znanosti Hrvatske (HATZ), predsjednik tehničkog odbora za robotiku međunarodne udruge IFAC, stalni član upravnog tijela European Conference of Mobile Robots, član izvršnog odbora međunarodne udruge FIRA, suutemeljitelj međunarodne udruge „The iSpace Laboratory Network". Član je i upravnog odbora Hrvatskog društva za komunikacije, računarstvo, elektroniku, mjerenja i automatiku (KoREMA) te glavni i odgovorni urednik časopisa Automatika te suradnik glavnog urednika časopisa Mechatronics. Godine 1990. primio je u Ljubljani nagradu „Prof. dr. Vratislav Bedjanič" za posebno istaknuti magistarski rad, 1998. srebrnu plaketu "Josip Lončar" FER-a za posebno istaknutu doktorsku disertaciju, a za znanstvena je postignuća dobio 2008. godine nagradu „Rikard Podhorsky" Akademije tehničkih znanosti Hrvatske, 2011. godine „Državnu nagradu za znanost", 2013. godine zlatnu plaketu "Josip Lončar" FER-a, 2015. godine nagradu za znanost FER-a, 2017. godine nagradu Hrvatske akademije znanosti i umjetnosti, 2019. godine nagradu "Nikola Tesla" Hrvatske sekcije IEEE te 2022. godine nagradu Grada Zagreba.

To my family

Acknowledgements

"Our greatest glory is not in never falling, but in rising every time we fall."

– Confucius

Thanks to my supervisor prof. Ivan Petrović and to my unofficial supervisor prof. Ivan Marković for giving me a chance in LAMOR *group and for all advices during my doctoral study.*

Thanks to my friends and colleagues from the LAMOR *group, by name Josip, Kruno, Igor, Jura, Luka P., Marta, Filip, Antea, Petki, Goran, Anđela, Demijan, Ana, Jelena, Karlo, Nikola, Ivan, Vlaho and Luka T.*

Special thanks to my mother, father and my sisters Barbara and Vesna for all the love and support.

In Zagreb, July 21, 2023

Abstract

State Estimation and Multiple Moving Object Tracking on Riemannian Manifolds

Multi-target tracking is an essential task in autonomous driving. In order to ensure the safety of other participants in traffic, an autonomous vehicle needs reliable data about the surrounding moving objects, such as other vehicles and pedestrians. Therefore, the purpose of this thesis is to tackle some of the plethora of challenging problems of multiple object tracking. First, the thesis presents a novel method for pedestrian tracking from the sequences of images. The visual tracking of moving objects differs from using other sensors in the availability of the objects' appearance, which can be extremely beneficial when dealing with one of the challenges of multi-target tracking – the detection-to-target assignment ambiguity. Nonetheless, fusing appearance cues represented by deep features of the neural networks together with the kinematic cues inside a classical multi-target tracking framework, such as probabilistic data association, proves to be a non-trivial problem due to the nonlinearity of the underlying space of deep embeddings. To this end, this dissertation focuses on the problem of state estimation on Riemannian manifolds – differentiable curved spaces with metric structures. By utilizing the tools of the Riemannian geometry, this thesis proposes a novel filtering method based on the unscented transform for systems whose state-space is a tangent bundle of a Riemannian manifold. By modeling the state-space as a tangent bundle of a manifold, it is possible to simultaneously estimate the position of the object on the manifold and its velocity, which lies in the tangent space of the manifold. In other words, it allows us to use the *constant velocity* motion model for state estimation on Riemannian manifolds. Even though this could be achieved by using an extended Kalman filter for Riemannian manifolds, the advantage of the proposed method is that it uses the unscented transform based on sigma-point representation to propagate the uncertainty of the filter, thus avoiding the tedious calculation of Jacobians necessary for extended Kalman filter. In the final part of the thesis, the state estimation methods for Riemannian manifolds are adapted to the multi-target case resulting in the novel multi-target tracking method for Riemannian manifolds based on probabilistic data association.

KEYWORDS: state estimation, multi-target tracking, Bayesian estimation, Kalman filter, Riemannian geometry, nonlinear estimation

Sažetak

Estimacija stanja i praćenje više gibajućih objekata na Riemannovim mnogostrukostima

Praćenje više objekata jedan je od osnovnih preduvijeta autonomne vožnje. Kako bi se osigurala sigurnost ostalih sudionika u prometu, autonomno vozilo treba pouzdane podatke o okolnim gibajućim objektima, kao što su druga vozila i pješaci. Stoga je svrha ove disertacije suočavanje s nekima od mnoštva izazovnih problema praćenja više objekata. Prvo, rad predstavlja novu metodu za praćenje pješaka iz sekvenci slika. Vizualno praćenje gibajućih objekate razlikuje se od praćenja pomoću drugih senzora u tome što su kod vizualnog praćenja dostupni izgledi objekata, što može biti izuzetno korisno kod rješavanja problema pridruživanja podataka. Unatoč tome, spajanje izgleda predstavljenog dubokim značajkama neuronskih mreža zajedno s kinematičkim stanjem gibajućeg objekta unutar klasičnih metoda praćenja više objekata, kao što je vjerojatnosno pridruživanje podataka, pokazuje se kao zahtijevan problem zbog nelinearnosti prostora u kojem leže duboke značajke. Zbog toga, ova se disertacija usredotočuje na problem estimacije stanja na Riemannovim mnogostrukostima – diferencijabilnim zakrivljenim prostorima s pridruženom glatkom metrikom. Korištenjem alata Riemannove geometrije, ova disertacija predlaže novu metodu filtriranja koja se temelji na transformaciji pomoću sigma točaka za sustave čiji je prostor stanja tangencijalni snop Riemannove mnogostrukosti. Modeliranjem prostora stanja kao tangencijalnog snopa mnogostrukosti, moguće je istovremeno estimirati položaj objekta na mnogostrukosti kao i njegovu brzinu, koja leži u tangentcijalnom prostoru mnogostrukosti. Drugim riječima, omogućuje nam korištenje modela gibanja *konstantne brzine* za estimaciju stanja na Riemannovim mnogostrukostima. Iako bi se to moglo postići korištenjem proširenog Kalmanovog filtra za Riemannove mnogostrukosti, prednost predložene metode je u tome što koristi transformaciju temeljenu na reprezentaciji pomoću sigma točaka za propagaciju nesigurnosti stanja filtra, čime se izbjegava zahtijevno izračunavanje Jakobijevih matrica potrebnih za prošireni Kalmanov filter na Riemannovim mnogostrukostima. U završnom dijelu disertacije, metode estimacije stanja za Riemannove mnogostrukosti prilagođene su za slučaj s više objekata, što je rezultiralo novom metodom praćenja više objekata na Riemannovim mnogostrukostima koja se temelji na vjerojatnosnom pridruživanju podataka.

Ova disertacija predstavlja tri originalna znanstvena doprinosa koji daju metode praćenja više objekata na zakrivljenim prostorima. U nastavku su navedeni i ukratko opisani glavni znanstveni doprinosi ove disertacije.

#1 *Metoda praćenja pješaka zasnovana na združenom integriranom vjerojatnosnom pridruživanju nad dubokim ugrađivanjima.*

Prvi doprinos ove disertacije pokriva novu metodu praćenja pješaka koja se sastoji od dva duboka modela te metode praćenja više objekata na temelju vjerojatnosnog pridruživanja podataka. Za detekciju pješaka korišten je Mask R-CNN duboki model koji je unaprijed utreniran te nakon toga fino podešen na COCO i CityPersons skupovima podataka. COCO skup podataka korišten je jer pruža veliku raznovrsnost u pozama ljudi, a CityPersons jer sadrži maske istinitosti na razini piksela. Za generiranje dubokih značajki izgleda korišten je drugi duboki model koji se temelji na ResNet-18 arhitekturi. Korištena su prva dva rezidualna bloka modela unaprijed treniranog na ImageNet skupu podataka koja su dodatno fino podešena na MOT16 skupu podataka. Za praćenje pješaka primjenjena je metoda združenog integriranog vjerojatnosnog pridruživanja podataka koja koristi model konstantne brzine za estimaciju stanja pojedinih pješaka.

#2 *Metoda filtriranja sigma točkama na tangencijalnom snopu Riemannove mnogostrukosti*

Ovaj doprinos pokriva proširenje metode za estimaciju stanja na Riemannovim mnogostrukostima na temelju transformacije pomoću sigma točaka. Prethodno predložena metoda Kalmanova filtra temeljena na reprezentaciji pomoću sigma točaka za sustave ograničene na Riemannove mnogostrukosti razvijena je za jednostavne modele prijelaza stanja sustava. U sklopu ovog doprinosa, metoda estimacije na Riemannovim mnogostukostima temeljena na reprezentaciji pomoću sigma točaka proširena je na tangencijalni snop Riemannove mnogostrukosti. To proširenje omogućuje primjenu modela gibanja prvog reda, odnosno konstantne brzine, za estimaciju stanja na Riemannovim mnogostrukostima. Također, prednost ove metode u usporedbi s već postojećom metodom za estimaciju stanja na tangencijalnom snopu Riemannove mnogostrukosti pomoću proširenog Kalmanovog filtra je što predložena metoda ne zahtjeva računanje Jakobijevih matrica.

#3 *Metoda za vjerojatnosno praćenje više objekata na Riemannovim mnogostrukostima*

Treći doprinos ove disertacije pokriva novu metodu praćenja više objekata koji su ograničeni na glatke i zakrivljene prostore. Praćenje više objekata pomoću vjerojatnosnog pridruživanja podataka prošireno je na glatke zakrivljene prostore korištenjem Riemannove geometrije. Koraci predikcije i ažuriranja predložene metode za praćenje više objekata dobiveni se primjenom proširenog Kalmanovog filtara ili Kalmanovog filtra temeljenog na

reprezentaciji pomoću sigma točaka za Riemannove mnogostrukosti. Predložena metoda praćenja više objekata na Riemannovim mnogostrukostima podržava i model nultog reda kao i model gibanja prvog reda.

Disertacija je podjeljena u sedam poglavlja. Na početku svakog poglavlja dan je kratak sažetak koji uvodi metode opisane u tom poglavlju. Nakon toga dan je kratak pregled literature usko vezane za navedenu metodu. Nadalje, čitatelja se postepeno uvodi u probleme koje predložena metoda rješava. Na kraju svakog poglavlja dan je kratki sažetak.

Prvo poglavlje disertacije predstavlja uvod u disertaciju. Na početku poglavlja, dana je formulacija problema kojih rješavamo te motivacija. Nakon toga navedeni su te ukratko opisani znanstveni doprinosi disertacije. Na poslijetku poglavlja, opisana je struktura disertacije te dan kratak sažetak svakog poglavlja.

Drugo poglavlje daje pregled Riemannove geometrije neophodan za razumijevanje ove disertacije. Riemannova geometrija pruža nam alate za diferencijalnu analizu na glatkim zakrivljenim prostorima. Prvo se razmatraju opće Riemannove mnogostrukosti. Definira se tangencijalni prostor Riemannove mnogostrukosti te glatki metrični tenzor pomoću kojega možemo izračunati skalarni produkt tangencijalnih vektora na mnogostrukosti. Nadalje, pomoću metričnog tenzora moguće je izračunati duljine krivulja kao i kuteve između krivulja na mnogostrukosti. Nakon toga uvedeni su geodezici, koji su krivulje koje lokalno povezuju točke na mnogostrukosti najkraćim putem. Uvede se dva preslikavanja između mnogostrukosti i njezinog tangencijalnog prostora – eksponencijalno i logaritamsko preslikavanje. Nadalje, razmatra se i geometrija tangencijalnog snopa Riemannove mnogostrukosti koji je i sam Riemannova mnogostrukost. Zatim, navedeno je nekoliko primjera Riemannovih mnogostrukosti kao što su n-dimenzionalna sfera te prostor simetričnih pozitivno-definitnih matrica. Na poslijetku se uvode koncentrirane Gaussove distribucije na tangencijalnom prostoru mnogostrukosti kao generalizacija multivarijatne normalne distribucije.

Treće poglavlje daje pregled stanja tehnike u području estimacije stanja i praćenja više objekata. Poglavlje započinje s uvodom u teoriju Bayesove estimacije. Zatim se uvode Kalmanov filtar i njegova nelinearna proširenja kao posebni slučajevi Bayesovog filtra. Još jedna klasa Bayesovog filtara, von Mises-Fisher filtar, također je predstavljen u ovom poglavlju. Drugi dio ovog poglavlja razmatra proširenje klasičnih metoda estimacije stanja na praćenje više objekata. Prikazana su tri različita pristupa problemu praćenja više objekata. Jedna od te tri grupe metoda praćenja više objekata je praćenje pomoću vjerojatnosnog pridruživanja podataka. Te metode rješavaju problem pridruživanja detekcija objektima na način da generiraju listu međusobno isključivih hipoteza o tome koja detkcija pripada kojem objektu. Zatim, za svaku od tih hipoteza izračuna se njezina vjerojatnost, a potom se za svaku detekciju izračuna vjerojatnost da ta detekcija potječe od svakog pojedinog objekta. Stanje svakog objekta se onda ažurira s otežanom srednjom vrijednosti svih dostupnih detekcija, pri čemu se svaka detekcija oteža s pripadajućom vjerojatnosti pridruživanja. Praćenje pomoću višestrukih hipoteza predstavlja drugu grupu metoda praćenja više objekata. Te metode grade globalno stablo hipoteza asocijacije između detekcija i objekata. Na taj se način odgađa odluka o asocijaciji detekcija u trenutnom koraku algoritma. Na temelju novih detekcija stablo hipoteza se proširuje s novim hipotezama te se zatim uklanjaju one hipoteze

čija je vjerojatnost zanemariva kako bi se osiguralo da je algoritam računalno izvodljiv. Treća grupa metoda praćenja više objekata su metode temeljene na slučajnim konačnim skupovima. Umjesto eksplicitnog rješavanja problema pridruživanja detekcija objektima, kod ovih metoda se cijeli više-objektni sustav modelira pomoću slučajnih skupova. Budući slučajni skupovi imaju slučajan broj elemenata, pri čemu je svaki element slučajni vektor, ti skupovi su pogodni za praćenje više objekata jer implicitno modeliraju vremenski promjenjiv broj objekata kao i nesigurnost pridruživanja detekcija objektima. Na kraju poglavlja, dane su neke od metoda za evaluaciju algoritama praćenja više objekata.

Četvrto poglavlje predlaže metodu za praćenje pješaka pomoću vjerojatnosnog pridruživanja podataka i dubokih značajki. Prvi dio poglavlja predstavlja duboke neuronske modele koji se kasnije koriste za detekciju pješaka i generiranje dubokih značajki. Za detekciju pješaka u nizu slika korišten je Mask R-CNN detektor. Detektor je prethodno uvježban na COCO skupu podataka, a zatim fino podešen na skupu podataka CityPersons. Duboke značajke dobivene su iz drugog dubokog modela, koji se temelji na ResNet-18 arhitekturi. Eksperiment ablacije pokazao je da ugradnje izvađene iz posljednjeg sloja drugog rezidualnog bloka ResNet-18 daju najbolje rezultate. Model duboke korespondencije uvježban je na MOT2016 skupu podataka. Združeni integrirani filtar vjerojatnosnog pridruživanja podataka prilagođen je kako bi se postiglo praćenje pješaka na temelju detekcija dobivenih Mask R-CNN detektorom. Metoda je evaluirana na javno dostupnim MOTChallenge skupovima podataka, gdje je ostvarila #1 rank u 3DMOT2015 kategoriji.

Peto poglavlje uvodi novu metodu za estimaciju stanja na Riemannovim mnogostrukostima. Polazeći od proširenja Kalmanovog filtra temeljenog na reprezentaciji pomoću sigma točaka za sustave koji su ograničeni na Riemannove mnogostrukosti, proširujemo ga korištenjem modela gibanja prvog reda, odnosno modelom konstantne brzine. Kako bi se implementirao model gibanja prvog reda na Riemannovim mnogostrukostima, korištena je geometrija tangencijalnog snopa mnogostrukosti. Predložena metoda filtriranja eksperimentalno je provjerena u Monte Carlo simulacijama na nekoliko različitih Riemannovih mnogostrukosti: jedinične n-dimenzionalne sfere, prostor simetričnih pozitivno-definitnih matrica te prostor dijagonalnih pozitivno-definitnih matrica. Iako predložena metoda postiže slične rezultate kao prošireni Kalmanov filtar s modelom gibanja prvog reda na Riemannovoj mnogostrukosti, njezina prednost je u tome što ne zahtijeva Jakobijeve matrice modela gibanja koje bi mogle biti kompliciranе ili za izračunavanje na pojedinim mnogostrukostima.

Šesto poglavlje predstavlja novu metodu praćenja više objekata za objekte koji se kreću po Riemannovim mnogostrukostima. Predložena metoda dobivena je proširenjem metoda procjene stanja za Riemannove mnogostrukosti uvedenih u petom poglavlju. Vjerojatnosno pridruživanje podataka korišteno je za rješavanje problema pridruživanja detekcija objektima u slučaju praćenja jetnog objekta, a združeno vjerojatnosno pridruživanje podataka u slučaju praćenja više objekata. U eksperimentalnoj evaluaciji razmatrano je nekoliko različitih implementacija filtara vjerojatnosnog pridruživanja podataka na Riemannovim mnogostrukostima: pomoću proširenog Kalmanovog filtra te na temelju Kalmanovog filtra temeljenog na reprezentaciji pomoću sigma točaka. Uz to, korišteni su i modeli gibanja nultog reda i prvog reda. U simulacijama su korištena dva različita primjera Riemannove mnogostrukosti: jedinična sfera te prostor dijagonalnih simetričnih pozitivno definitnih

matrica.

Posljednje poglavlje daje zaključak disertacije te diskutira o potencijalnim budućim smjerovima istraživanja.

KLJUČNE RIJEČI: estimacija stanja, praćenje više objekata, Bayesova estimacija, Kalmanov filter, Riemannova geometrija, nelinearna estimacija

Contents

1 INTRODUCTION 1
- 1.1 Motivation and Problem Statement . . . 2
- 1.2 Original Contributions of the Thesis . . . 3
- 1.3 Outline of the Thesis . . . 3

2 RIEMANNIAN GEOMETRY 6
- 2.1 Tangent Space . . . 7
- 2.2 Riemannian Metric . . . 9
- 2.3 Connection . . . 9
  - 2.3.1 Parallel Transport . . . 11
  - 2.3.2 Curvature . . . 11
- 2.4 Geodesics . . . 11
  - 2.4.1 Exponential and Logarithmic Map . . . 12
- 2.5 Tangent Bundle . . . 12
- 2.6 Examples of Riemannian Manifolds . . . 16
  - 2.6.1 Sphere . . . 16
  - 2.6.2 Space of Symmetric Positive Definite Matrices . . . 18
  - 2.6.3 Space of Diagonal Positive Definite Matrices . . . 19
- 2.7 Statistics on Riemannian Manifolds . . . 20
- 2.8 Summary . . . 21

3 STATE ESTIMATION AND MULTI TARGET TRACKING 22
- 3.1 Bayesian Estimation . . . 23
  - 3.1.1 Kalman Filter . . . 25
  - 3.1.2 Extended Kalman Filter . . . 26
  - 3.1.3 Unscented Kalman Filter . . . 27
  - 3.1.4 Von Mises-Fisher Filter . . . 29
- 3.2 Introduction to Multiple Object Tracking . . . 30
- 3.3 Probabilistic Data Association . . . 32
  - 3.3.1 Integrated Probabilistic Data Association . . . 32
  - 3.3.2 Joint Probabilistic Data Association . . . 33

3.3.3 Joint Integrated Probabilistic Data Association . . . 33
3.4 Multiple Hypothesis Tracking . . . 34
3.4.1 Hypothesis Oriented Multiple Hypothesis Tracking . . . 35
3.4.2 Track Oriented Multiple Hypothesis Tracking . . . 36
3.4.3 Probabilistic Multiple Hypothesis Tracking . . . 38
3.5 Random Finite Sets Approach . . . 38
3.5.1 Random Finite Sets . . . 40
3.5.2 Multi-Target Bayes Filtering . . . 42
3.5.3 Probability Hypothesis Density Filter . . . 46
3.6 Multiple Object Tracking Evaluation . . . 50
3.6.1 Optimal Sub-Pattern Assignemnt . . . 50
3.6.2 Clear MOT Metrics . . . 51
3.7 Summary . . . 53

4 PROBABILISTIC TRACKING OVER DEEP EMBEDDINGS 54
4.1 Related Work . . . 57
4.2 MOT Challenge Benchmark . . . 58
4.3 Detection and Appearance Representation . . . 60
4.3.1 Pedestrian Detection . . . 60
4.3.2 Deep Correspondence Embedding . . . 61
4.3.3 Details of Training Correspondence Embedding . . . 62
4.4 Pedestrian Tracking Method . . . 62
4.5 Experimental Results . . . 63
4.5.1 Validating the Detection and Correspondence Model . . . 63
4.5.2 Pedestrian Tracking Evaluation . . . 64
4.5.3 Comparison With Public Detections . . . 66
4.6 Summary . . . 68

5 STATE ESTIMATION ON RIEMANNIAN MANIFOLDS 69
5.1 Related Work . . . 70
5.2 System Models on Riemannian Manifold . . . 71
5.2.1 Zero-Order Motion Model on Riemannian Manifold . . . 72
5.2.2 First-Order Motion Model on Riemannian Manifold . . . 72
5.3 Kalman Filters on Manifolds . . . 73
5.3.1 Extended Kalman Filtering on Manifolds . . . 74
5.3.2 Unscented Kalman Filtering on Manifolds . . . 76
5.4 Constant Velocity Kalman Filters on Riemannian Manifolds . . . 78
5.4.1 Riemannian Manifold Extended Kalman Filter . . . 79
5.4.2 Riemannian Manifold Unscented Kalman Filter . . . 79
5.5 Experimental Results . . . 80
5.5.1 Zero-Order Model . . . 80
5.5.2 First-Order Model . . . 84
5.6 Summary . . . 87

6 PROBABILISTIC MULTI OBJECT TRACKING ON RIEMANNIAN MANIFOLDS 88

6.1 Related work . . . 89
6.2 PDAF on Riemannian manifolds . . . 90
6.2.1 Prediction step . . . 90
6.2.2 Measurement gating . . . 91
6.2.3 Update step . . . 91
6.3 JPDAF on Riemannian manifolds . . . 92
6.3.1 Prediction step . . . 92
6.3.2 Measurement gating . . . 92
6.3.3 Update step . . . 92
6.4 Results . . . 93
6.4.1 Tracking on the unit sphere . . . 93
6.5 Summary . . . 96

7 CONCLUSION AND OUTLOOK 97

BIBLIOGRAPHY 101

A DERIVATION OF JACOBIANS 110
A.1 Sphere . . . 111
A.2 Diagonal positive definite matrices . . . 115

LIST OF FIGURES 117

LIST OF TABLES 118

ACRONYMS 119

CURRICULUM VITAE 121

PUBLICATIONS 122

ŽIVOTOPIS 123

Chapter 1

Introduction

"A person who never made
a mistake, never tried anything new."

– Albert Einstein

The introduction chapter elaborates the motivation behind the research conducted within this thesis. First, we start by discussing what exactly does the task of the multi-target tracking refer to. There we discuss on the many challenging problems that arise in multi-target tracking applications. Then we proceed with introducing the geometry of the curved spaces in the context of state estimation and multi-target tracking. Subsequently, the original contribution of the thesis are presented together with a short summary of each of them. Finally, we provide the outline and the structure of the thesis and give a short summary of each chapter.

1.1 Motivation and Problem Statement

Many challenging problems arise in multi-target tracking (MTT) compared to classical estimation such as missing detections, false alarm, uncertainty in measurement origin and many others [1]. Most of the state-of-the-art MTT algorithms can be divided in three groups with respect to how they treat the unknown measurement origin (data association): (i) probabilistic data association (PDA), (ii) multiple hypothesis tracking (MHT) and (iii) random finite sets (RFS) tracking. PDA and its variants [2–5] calculate posterior association probabilities between tracks and received detections and then update each target with the weighted sum of detections. MHT methods [6, 7] handle the detection origin uncertainty by creating a tree of possible association hypotheses. The tree is created recursively while the unlikely hypotheses are discarded to reduce the computational load. RFS tracking methods [8–11] are paradigm that does not solve the data assignment problem directly but rather formulate the tracking as filtering on random finite sets [12]. Regardless of the MTT approach, the underlying geometry of the tracked targets state does not have to necessarily reside on a Euclidean space.

Non-Euclidean spaces have recently been addressed in many robotic and computer vision applications [13, 14]. Rigid body pose can be naturally modelled as an element of the Lie group, hence state-of-the-art simultaneous localisation and mapping (SLAM) and pose estimation algorithms rely on the geometry of the underlying space [15–19]. Other examples of using non-Euclidean geometry can be found in state estimation and tracking

on directional only data, which can be accomplished by utilizing directional distributions, such as the von Mises-Fisher or the Bingham distribution [20–24]. In visual tracking, object appearance is very useful in the detection-to-target association procedure which can be represented using hand-crafted features [25–27] or deep neural network features [28–30]. Both directional only data and covariance features lie on smooth metric spaces called Riemannian manifolds (RMs) [14]. Visual tracking applications can use the geometry of those spaces to find the optimal assignment between detected and tracked objects [31], but they often ignore the dynamics of objects and the uncertainty of deep features. Nonetheless, filtering methods involving RM valued systems have been introduced in [32–34].

1.2 Original Contributions of the Thesis

This thesis provides three original scientific contributions which offer multi-target tracking methods for objects that evolve on curved spaces. Hereafter we introduce all contributions and provide a brief elaboration of each of them:

#1 *A method for pedestrian tracking based on joint integrated probabilistic data association over deep embeddings.*

The first contribution of this thesis covers the novel pedestrian tracking method which consist of an Mask R-CNN object detector and probabilistic data association tracker. The Mask R-CNN object detector is pre-trained and fine-tuned on different datasets in order to avoid overfitting.

#2 *A method for sigma point filtering on the tangent bundle of a Riemannian manifold.*

This contribution covers the novel method for state estimation on Riemannian manifolds based on unscented transform. Previously proposed unscented Kalman filter method for systems constrained on Riemannian manifold considered simple state transition model.

#3 *A method for probabilistic multiple object tracking on Riemannian manifolds.*

The third contribution covers the novel multi-target tracking method for targets that are constrained to smooth and possibly curved spaces. The probabilistic data association based multi-target trackers are extended to the smooth, curved spaces by utilising the Riemannian geometry. The prediction and update steps of the novel tracker are obtained by applying extended and unscented Kalman filters for Riemannian manifolds. The proposed Riemannian manifold multi-target tracking method supports both the zero-order as well as first-order motion model.

1.3 Outline of the Thesis

This thesis is organised into seven chapters. Each chapter begins with a brief abstract which aims to concisely present the method proposed in the chapter and its contributions.

Afterwards, the reader is gradually introduced to the problem and the related work in the field. In the end of each chapter, a short summary is given which restates some of the main results of the chapter, obtained experimental results and conclusions that stem from them. The sequel of this section presents the outline of the thesis with a short summary of each chapter.

Chapter 2 This chapter presents the overview of the Riemannian geometry necessary for the understanding of this thesis. The Riemannian geometry provides the tools for the differential analysis on the smooth curved spaces. First, the general Riemannian manifolds are considered followed by some examples of Riemannian manifolds. Additionally, the geometry of the tangent bundle of the Riemannian manifold is also considered. Concentrated Gaussian distributions on the tangent space of the manifold are introduced as an generalisation of the multivariate normal distribution.

Chapter 3 This chapter gives an overview of the state-of-the-art of the state estimation and multi-target tracking. First, the theory of the Bayesian estimation is presented. Then, the Kalman filter and its nonlinear extensions are introduced as special cases of the Bayes filter. Another class of Bayes filter, the von Mises-Fisher filter, is also presented in this chapter. The second part of this chapter considers the extension of the classical state estimation to the multi-target tracking. Three different approaches to multi-target tracking problem are presented. Finally, some multi-target tracking evaluation methods are introduced.

Chapter 4 This chapter proposes a method for pedestrian tracking via probabilistic data association and deep embeddings. The first part of the chapter introduces deep neural models used to detect pedestrians and generate the deep correspondence embeddings. The Mask R-CNN detector was used to detect the pedestrians on the image. The detector was pre-trained on the COCO dataset and then fine-tuned on the CityPersons dataset. Deep correspondence embeddings were extracted from another deep model, which was constructed from the ResNet-18 architecture. The ablation experiment showed that the embeddings extracted from the last layer of the second residual block of the ResNet-18 yield best results. The deep correspondence model was trained on the MOT2016 dataset. The joint integrated probabilistic data association filter was adapted in order to achieve multi-target tracking given the detections obtained by the Mask R-CNN detector. The method was evaluated on the public MOTChallenge benchmark where it was ranked #1 result on the 3DMOT2015 dataset.

Chapter 5 This chapter introduces the novel method for state estimation on Riemannian manifolds. Starting from the unscented Kalman filter for systems that evolve on Riemannian manifold, we extend it using the first-order motion model. In order to implement the first-order motion model on Riemannian manifold, the theory the geometry of the tangent bundle of

the manifold was utilised. The proposed filtering method was experimentally verified in Monte Carlo simulations on two different Riemannian manifolds. Although the proposed method achieves similar results as the extended Kalman filter with the first-order motion model on the Riemannian manifold, it does not need Jacobians of the motion model which might be tedious or even impossible to calculate on some manifolds.

Chapter 6 This chapter presents the novel multi-target tracking method for targets moving on Riemannian manifolds. The proposed method was implemented by extending the state estimation methods for Riemannian manifolds introduced in chapter 5. Probabilistic data association was utilized to solve the measurement-to-target assignment problem in single-target case and joint probabilistic data association in multi-target case. In the experimental evaluation, a few different implementation of the Riemannian manifold probabilistic data association filters were considered: one based on the extended Kalman filter and one based on the unscented Kalman filter. Additionally, both the zero-order and first-order motion models were used. Two different manifolds were used in simulations: the unit sphere and the space of positive definite matrices.

Chapter 7 This chapter presents the conclusion of the thesis and gives a summary of each scientific contribution of the thesis.

Chapter 2

Riemannian Geometry

The purpose of this chapter is to give a brief introduction to the field of Riemannian geometry necessary to understand the rest of the thesis. The Riemannian geometry studies the geometry of smooth spaces equipped with metric tensor that are also known as Riemannian manifolds (RMs). A d-dimensional manifold $\mathcal{M}$ is a connected space such that there exists a neighbourhood U at each point in $\mathcal{M}$ which is homeomorphic to some open subset Ω on $\mathbb{R}^d$ [35]. Hence, there exists a continuous bijective mapping $x : U \to \Omega$ which is called a local coordinate chart. A set of charts $\{U_a, x_a\}_{a=1}^n$ that covers the complete manifold $\mathcal{M}$ is called an atlas of $\mathcal{M}$. Furthermore, if all transitions between the charts in an atlas are differentiable, then the atlas is called differentiable atlas.

Definition 2.1 (Differentiable atlas [35])**.** *An atlas $\{U_a, x_a\}_{a=1}^n$ is called differentiable atlas if for any two charts U_a and U_b such that $U_a \cap U_b \neq \emptyset$, the chart transition*

$$x_b \circ x_a^{-1} : x_a(U_a \cap U_b) \to x_b(U_a \cap U_b)$$

is differentiable of class C^∞.

A manifold $\mathcal{M}$ together with the differentiable atlas $\{U_a, x_a\}_{a=1}^n$ is called a differentiable or smooth manifold. A representation of a point $p \in U_a$ in a chart is given by $(x^1, \ldots, x^d) = x_a(p)$ and $(x^1, \ldots, x^d)$ are called local coordinates of p in chart (U_a, x_a).

Definition 2.2 (Differentiable manifold)**.** *Differentiable, or smooth, manifold is an manifold $\mathcal{M}$ together with the differentiable atlas.*

2.1 Tangent Space

Although differentiable manifolds are generally curved spaces and thus are not Euclidean, they can be well approximated by the Euclidean space at the neighbourhood of each of its points $p \in \mathcal{M}$. Such spaces are called tangent spaces of $\mathcal{M}$ and are denoted with $T_p\mathcal{M}$. Let $x = (x^1, ..., x^n)$ represent a local coordinate chart on some subset $U \in \mathcal{M}$ and let $p \in U$ be a point on a manifold. The partial derivatives $\frac{\partial}{\partial x^i}$ for $i \in (1, \ldots, d)$ at point p span the tangent space $T_p\mathcal{M}$ [35]. In the following, ∂_i will be used as a shorthand notation for $\frac{\partial}{\partial x^i}$. Thus, all tangent vectors $u \in T_p\mathcal{M}$ can be expressed in local coordinates as $u = u^i \partial_i$, where $(u^1, \ldots, u^d)$ are components of u in chart (U, x). Here and in the sequel we use the Einstein notation, i.e. repeating the same index once as a subscript and once as a superscript denotes

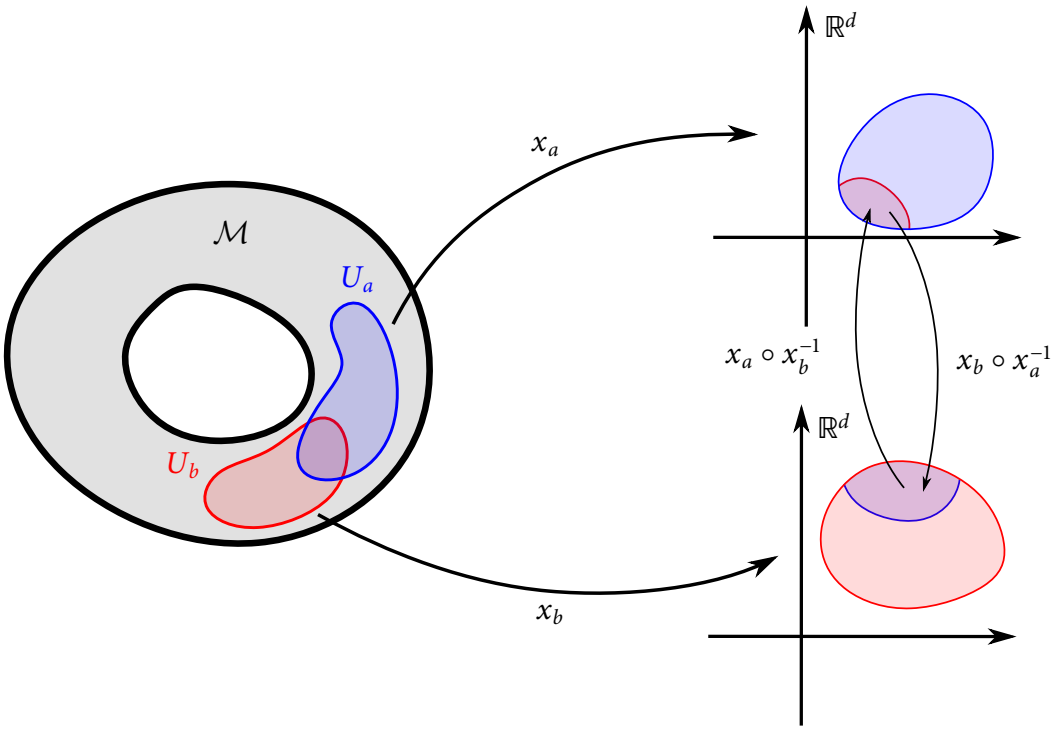


Figure 2.1: Smooth d-dimensional manifold $\mathcal{M}$ with two subsets U_a and U_b with smooth maps x_a and x_b respectively. Map x_a provides local coordinate representation of subset U_a in $\mathbb{R}^d$, while x_b provides local coordinate representation of U_b. The change of coordinates between these two local charts are given by transitions $x_b \circ x_a^{-1}$ and $x_a \circ x_b^{-1}$ which are differentiable of class C^∞ on the set $U_a \cap U_b$.

summation over that index from 1 to the dimension of the space), hence $u^i \, \partial_i$ stands for $\sum_{i=1}^d u^i \, \partial_i$, while

$$g_{ij} u^i v^j = \sum_{i,j=1}^{d} g_{ij} u^i v^j.$$

In numerical applications it is sometimes necessary to switch from one coordinate chart to another. For example, during state estimation on manifold, a state might move out of the domain of a chart and different chart must be used. Let $\bar{x}$ represent the new coordinate chart and let $(u^1, \ldots, u^d)$ be components of $u \in T_p\mathcal{M}$ in coordinate chart x. The coordinate transformation of u is given by [35]

$$\bar{u}^j = \frac{\partial \bar{x}^j}{\partial x^i} u^i, \tag{2.1}$$

where $\frac{\partial \bar{x}^j}{\partial x^i}$ are components of the Jacobian of coordinate change. Furthermore, let $f : \mathcal{M} \to \mathcal{N}$ be a smooth mapping from manifold $\mathcal{M}$ to manifold $\mathcal{N}$. The differential of f at point p is then the linear map $\mathrm{d}f(p) : T_p\mathcal{M} \to T_{f(p)}\mathcal{N}$. Denote with F_j^i components of $\mathrm{d}f(p)$ in local charts x on $\mathcal{M}$ and y on $\mathcal{N}$. Let now $\bar{x}(x)$ be a coordinate change on $\mathcal{M}$ and $\bar{y}(y)$ a coordinate change on $\mathcal{N}$. The coordinate transformation of $\mathrm{d}f(p)$ is then given by [35]

$$\bar{F}_b^a = \frac{\partial \bar{y}^a}{\partial y^i} \frac{\partial x^j}{\partial \bar{x}^b} F_j^i. \tag{2.2}$$

2.2 Riemannian Metric

So far we have defined what the smooth manifolds are and we showed that any smooth manifold can be locally approximated with the Euclidean space. However, to be able to measure lengths of curves and angles between tangent vectors on smooth manifold, it is necessary to introduce the metric structure on differentiable manifold. Since each tangent space $T_p\mathcal{M}$ is Euclidean space, it can be equipped with the scalar product $\langle u, v\rangle_p$. Moreover, if $\mathcal{M}$ is a smooth manifold, it is possible to define $\langle u, v\rangle_p$ in such a way that it depends smoothly on point $p \in \mathcal{M}$ [35].

Definition 2.3 (Metric tensor [35]). *The Riemannian metric g on smooth manifold $\mathcal{M}$ is the bilinear symmetric positive-definite tensor, which induces the scalar product $\langle u, v\rangle_p$ on each tangent space $T_p\mathcal{M}$. This scalar product can be written in coordinate chart x as*

$$\langle u, v\rangle_p = g_{ij}(x(p))\, u^i v^j, \tag{2.3}$$

*where g_{ij} are components of metric tensor g at point p in chart x, while u^i and v^j are respectively components of tangent vectors $u, v \in T_p\mathcal{M}$ in the same coordinate chart x. A smooth manifold $\mathcal{M}$ endowed with the metric tensor g is called the Riemannian manifold (*RM*).*

Equation (2.3) can also be written in matrix notation as $\boldsymbol{u}^T \boldsymbol{G}(x(p))\boldsymbol{v}$, where $\boldsymbol{G}$ is the matrix representation of metric tensor g. The norm of the tangent vector $u \in T_p\mathcal{M}$ is then given by

$$\|u\|_p = \sqrt{\langle u, u\rangle_p}. \tag{2.4}$$

Let now $\gamma(t) : [a, b] \to \mathcal{M}$ be a smooth curve on manifold $\mathcal{M}$. The velocity $\dot{\gamma}(t)$ at each point of the curve is then the tangent vector of $\mathcal{M}$. Thus, it is possible to measure length of $\gamma(t)$ by using the metric tensor g [35]

$$L(\gamma) = \int_a^b \|\dot{\gamma}(t)\|_{\gamma(t)}\, \mathrm{d}t = \int_a^b \sqrt{g_{ij}(x(\gamma(t)))\, \dot{\gamma}^i(t)\, \dot{\gamma}^j(t)}\, \mathrm{d}t. \tag{2.5}$$

Similarly, the energy of a curve is defined by [35]

$$E(\gamma) = \frac{1}{2} \int_a^b \|\dot{\gamma}(t)\|^2_{\gamma(t)}\, \mathrm{d}t. \tag{2.6}$$

Now it is possible to define the distance between two points p and q on manifold $\mathcal{M}$ as the length of the shortest piecewise smooth curve connecting p and q [35]

$$d_{\mathcal{M}}(p, q) = \inf\{L(\gamma); \gamma : [a, b] \to \mathcal{M}\}, \tag{2.7}$$

where $\gamma(a) = p$ and $\gamma(b) = q$. It can be easily showed that this definition satisfies the usual distance axioms [35].

2.3 Connection

It is often necessary to perform various operations on tangent vectors that do not lie on the same tangent space. For this purpose, it is possible to equip the smooth manifold with the

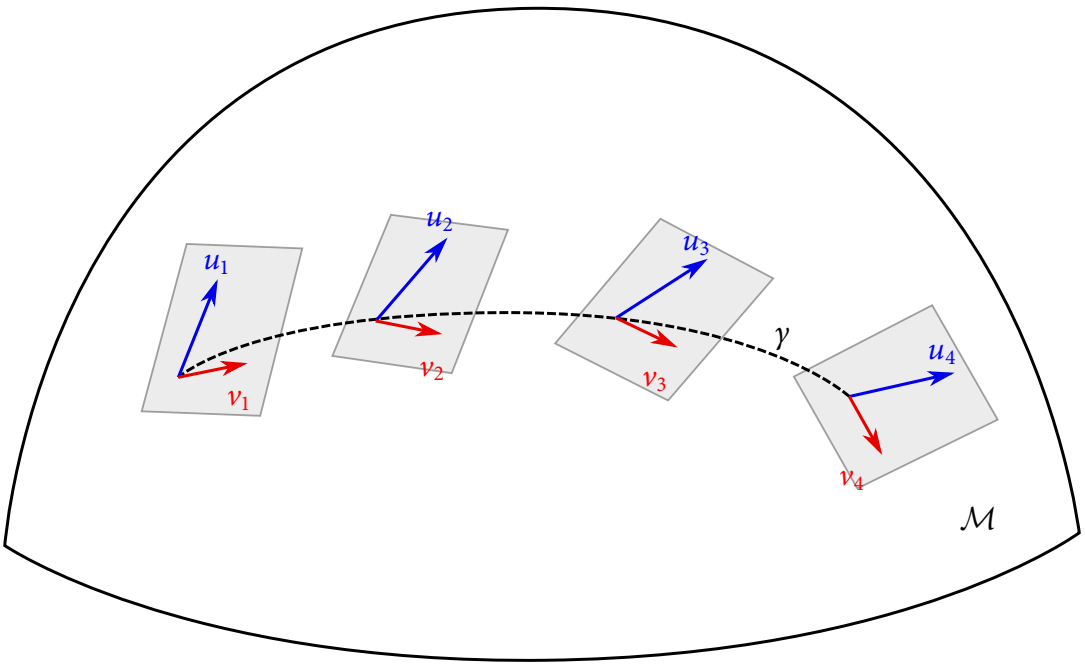


Figure 2.2: Parallel transport of vectors u and v along a curve γ on a manifold.

affine connection ∇. Such connections joins infinitesimally close tangent spaces. However, due to the non-existence of the global coordinate system for general smooth manifolds there is not a unique way to define the affine connection. One example of a connection is the Levi-Civita connection [35], which is often referred to as a covariant derivative. Informally, the covariant derivative is the generalization of the directional derivative from the usual vector calculus.

Definition 2.4 (Covariant derivative). *An affine connection ∇ is called a covariant derivative if*

a) it is compatible with the metric tensor g, i.e. , $\nabla g = 0$

b) it is torsion-free, i.e. , $\nabla_u v - \nabla_v u = [u, v]$ for any two vector fields u and v on $\mathcal{M}$, where $[u, v]$ is the Lie bracket operation on vector fields.

The covariant derivative of the vector field $v = v^i \partial_i$ with respect to vector field $u = u^i \partial_i$ in a coordinate chart $x : U \to \mathbb{R}^d$, where $U \subset \mathcal{M}$, is given by [35]

$$\nabla_u v = \left(u^j \frac{\partial v^k}{\partial x^j} + u^j v^i \Gamma^k_{ij} \right) \partial_k. \tag{2.8}$$

Coefficients Γ^k_{ij} are called Christoffel symbols of the second kind and are given by [35]

$$\Gamma^k_{ij} = \frac{1}{2} g^{kl} \left(\frac{\partial}{\partial x^j} g_{il} + \frac{\partial}{\partial x^i} g_{jl} - \frac{\partial}{\partial x^l} g_{ij} \right), \tag{2.9}$$

and g^{kl} are elements of the inverse of a metric tensor g. The first term in (2.8) is the usual directional derivative while the second term accounts for the curvature of the coordinate system with respect to the covariant derivative.

2.3.1 *Parallel Transport*

By applying the covariant derivative it is possible to define the parallel transport of the tangent vectors along smooth curves on manifold. Let $\gamma : [a, b] \to \mathcal{M}$ be a smooth curve on manifold $\mathcal{M}$ and let $u(t)$ be a smooth vector field defined along $\gamma(t)$. The vector field u is said to be parallel along curve $\gamma(t)$ with respect to covariant derivative, if it satisfies

$$\nabla_{\dot{\gamma}(t)} u(t) = 0. \tag{2.10}$$

A parallel transport between two points p and q on a smooth manifold $\mathcal{M}$, in general, is not uniquely defined, but it depends on the choice of the curve γ connecting p and q. The parallel transport is visualised on the figure 2.2.

2.3.2 *Curvature*

In general, the covariant derivative on the RM does not commute, i.e.

$$\nabla_u \nabla_v w \neq \nabla_v \nabla_u w,$$

where u, v, and w are smooth vector fields on manifold $\mathcal{M}$. This non-commutativity is captured by the Riemann curvature tensor defined by [35]

$$R(u, v)w = \nabla_u \nabla_v w - \nabla_v \nabla_u w - \nabla_{[u,v]} w, \tag{2.11}$$

where $[u, v]$ is the Lie bracket operation on the vector fields u and v. The components of the curvature tensor are given by

$$R^i_{jkl} = \partial_k \Gamma^i_{lj} - \partial_l \Gamma^i_{kj} + \Gamma^i_{km} \Gamma^m_{lj} - \Gamma^i_{lm} \Gamma^m_{kj}, \tag{2.12}$$

2.4 Geodesics

In Euclidean space, there is a concept of a straight line, a shortest line connecting two points. Due to the curvature, there is no such thing as a straight line on a general smooth manifold. Nonetheless, it is possible to generalize the concept of the smooth shortest path connecting close-by points on the manifold by minimizing the length or energy functionals given by eqs. (2.5) and (2.6). Such curves are called geodesics and can be taught of as straight lines on the smooth manifold.

Let $x : U \to \mathbb{R}^d$ be a coordinate chart on $\mathcal{M}$. Then the Euler-Lagrange equations of the energy functional (2.6) are [35]

$$\ddot{x}^i(t) + \Gamma^i_{jk}(x(t))\, \dot{x}^j(t)\, \dot{x}^k(t) = 0. \tag{2.13}$$

Thus, we define:

Definition 2.5 (Geodesic curve [35]). *A smooth curve $\gamma : [a, b] \to \mathcal{M}$ which minimizes the energy functional (2.6) is called geodesic and it satisfies (2.13) with $\dot{x}^i(t) = \frac{\mathrm{d}}{\mathrm{d}t} x^i(\gamma(t))$.*

2.4.1 *Exponential and Logarithmic Map*

Let $\mathcal{M}$ be an RM and let $p \in \mathcal{M}$. It can be shown that there exists a neighbourhood $V = \{v \in T_p\mathcal{M} : \|v\|_p < \varepsilon\}$ for some $\varepsilon > 0$ such that for each tangent vector $v \in V$ there exist a unique geodesic γ passing through p such that its velocity at p equals v [35]. Hence, it is possible to define a diffeomorphic mapping from V to the neighbourhood of $p \in \mathcal{M}$.

Definition 2.6 (Exponential map [35])**.** *Let $\mathcal{M}$ be an* RM *and $p \in \mathcal{M}$. Denote with $\gamma_v(t)$ a geodesic starting at p with velocity $\dot{\gamma}_v(0) = v \in T_p\mathcal{M}$ and let V_p be a subset of $T_p\mathcal{M}$ such that for any $v \in V_p$, the geodesic $\gamma_v(t)$ is uniquely defined on $[0,1]$. The diffeomorphism*

$$\begin{aligned} \mathrm{Exp}_p : V_p &\to \mathcal{M} \\ v &\mapsto \gamma_v(1) \end{aligned}$$

is called the exponential map of manifold $\mathcal{M}$ at p.

For many manifolds it is possible to extend the domain of the exponential to the entire $T_p\mathcal{M}$ for all $p \in \mathcal{M}$. Such manifolds are called geodesically complete manifolds [35]. Informally, the inverse of the exponential map, when it exists, is called the logarithmic map and it maps the neighbourhood of $p \in \mathcal{M}$ to the tangent space $T_p\mathcal{M}$. Note that the logarithmic map does not need to be unique even in the case of geodesically complete manifold. For example, two antipodal points on sphere can be connected by infinitely many geodesics, all of the same length. The logarithmic map is denoted with

$$\begin{aligned} \mathrm{Log}_p : \mathcal{M} &\to T_p\mathcal{M} \\ q &\mapsto v. \end{aligned}$$

When there is a unique shortest geodesic connecting two points p and $q \in \mathcal{M}$, then $\mathrm{Log}_p(q)$ is uniquely defined and following holds

$$d_{\mathcal{M}}(p,q) = \left\|\mathrm{Log}_p(q)\right\|_p. \tag{2.14}$$

2.5 Tangent Bundle

Tangent bundle $T\mathcal{M}$ of a manifold $\mathcal{M}$ is a union of all its' tangent spaces

$$T\mathcal{M} = \bigcup_{x \in \mathcal{M}} T_x\mathcal{M}. \tag{2.15}$$

Thus a pair $p = (x, v)$, where $x \in \mathcal{M}$ and $v \in T_x\mathcal{M}$, forms a point of a tangent bundle $T\mathcal{M}$. A choice of local coordinates $x = (x^1, \ldots, x^n)$ on $\mathcal{M}$ thus naturally induces the local

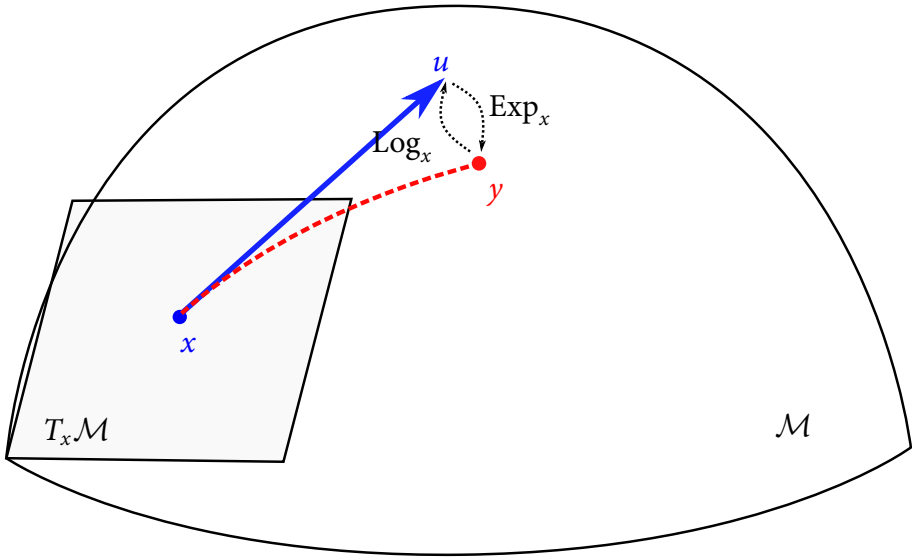


Figure 2.3: Exp and Log maps of a Riemannian manifold $\mathcal{M}$.

coordinates $p = (x^1, ..., x^n, v^1, ..., v^n)$ on $T\mathcal{M}$ and the basis $\left(\frac{\partial}{\partial x^1}, ..., \frac{\partial}{\partial x^n}\right)$ induces the basis $\left(\frac{\partial}{\partial p^1}, ..., \frac{\partial}{\partial p^{2n}}\right) = \left(\frac{\partial}{\partial x^1}, ..., \frac{\partial}{\partial x^n}, \frac{\partial}{\partial v^1}, ..., \frac{\partial}{\partial v^n}\right)$ on the tangent space $T_pT\mathcal{M}$.

Let now $\bar{x}^i(x^1, ..., x^n)$ be a coordinate change on $\mathcal{M}$ and v^i components of a tangent vector $v \in T_x\mathcal{M}$. From (2.1) we have that the components of v change by multiplying it with the Jacobian of the coordinate change $\frac{\partial \bar{x}^i}{\partial x^j}$. Hence the induced coordinate change on $T\mathcal{M}$ is given by

$$(x, v) \to \left(\bar{x}(x), \frac{\partial \bar{x}^i}{\partial x^j}(x) \cdot v^j\right). \tag{2.16}$$

Similarly as in the case of a simple manifold $\mathcal{M}$, the coordinate transformation of tangent vector $\mu \in T_pT\mathcal{M}$ at point $p = (x, v)$ is obtained by multiplying it with the Jacobian of the coordinate change (2.16) which is given by

$$\frac{\partial \bar{p}}{\partial p} = \begin{bmatrix} \frac{\partial \bar{x}^i}{\partial x^j} & 0 \\ \frac{\partial}{\partial x^k}\left(\frac{\partial \bar{x}^i}{\partial x^j} \cdot v^k\right) & \frac{\partial \bar{x}^i}{\partial x^j} \end{bmatrix}. \tag{2.17}$$

Tangent bundle of a Riemannian manifold is a Riemannian manifold as well [35] and can be endowed with the natural metric $\tilde{g}$ called Sasaki metric given the metric g of the manifold $\mathcal{M}$[36, 37]. The tangent space $T_pT\mathcal{M}$ at point $p = (x, u)$ can be split in horizontal and vertical subspaces $H_pT\mathcal{M}$ and $V_pT\mathcal{M}$ [36]. Let $v = \xi^H + \mu^V \in T_pT\mathcal{M}$ be a tangent vector of $T\mathcal{M}$ at point p, where ξ^H is its horizontal component and μ^V vertical component. Both of these components can be represented by tangent vectors of $\mathcal{M}$ at point x. Lifts from the tangent space of $T_x\mathcal{M}$ to horizontal and vertical subspaces of $T_pT\mathcal{M}$ are denoted by $[\cdot]^H : T_x\mathcal{M} \to H_pT\mathcal{M}$ and $[\cdot]^V : T_x\mathcal{M} \to V_pT\mathcal{M}$. Expressions for horizontal and vertical lifts of $v \in T_x\mathcal{M}$ to $T_{(x,v)}T\mathcal{M}$ in local coordinates are [36]

$$v^H = v^i \frac{\partial}{\partial x^i} - v^j v^k \Gamma^i_{jk} \frac{\partial}{\partial v^i}, \tag{2.18}$$

$$v^V = v^i \frac{\partial}{\partial v^i}. \tag{2.19}$$

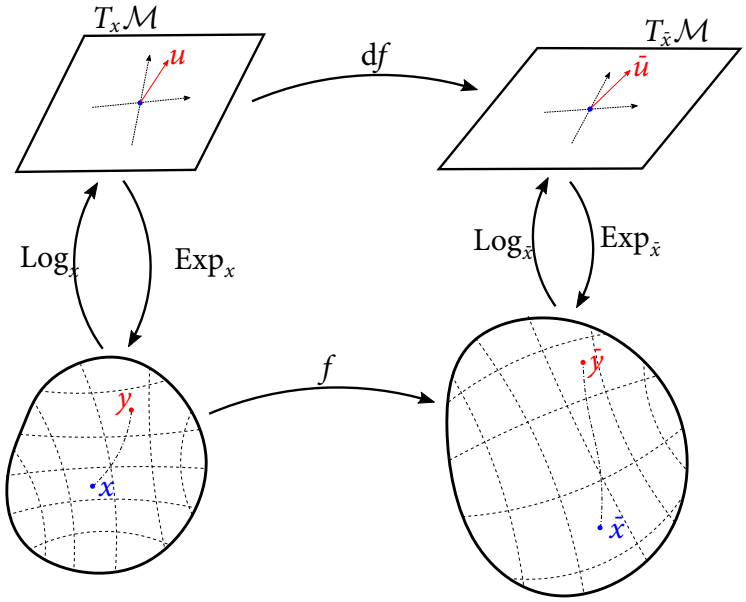


Figure 2.4: An example of the coordinate change on manifold $\mathcal{M}$. In first coordinate system (left) we have two points, x and y, whose coordinates in the other coordinate system (right) are $\tilde{x} = f(x)$ and $\tilde{y} = f(y)$. Log_x maps the neighbourhood of x to the tangent space $T_x\mathcal{M}$ at point x and Exp_x maps the tangent space $T_x\mathcal{M}$ to the neighbourhood of x. Similarly, we have the tangent space $T_{\tilde{x}}\mathcal{M}$ and maps $\mathrm{Exp}_{\tilde{x}}$ and $\mathrm{Log}_{\tilde{x}}$ in the other coordinate system (left) In essence, $T_x\mathcal{M}$ and $T_{\tilde{x}}\mathcal{M}$ are the same tangent space represented in different basis and the transformation between two is given by the Jacobian of the coordinate transformation $\mathrm{d}f : T_x\mathcal{M} \to T_{\tilde{x}}\mathcal{M}$. If, for example, we are given the point x and tangent vector $u \in T_x\mathcal{M}$ and we want to calculate $y = \mathrm{Exp}_x(u)$, but the expression for the exponential map is too complex in the first (left) coordinate system compared to the second (right), we can transform the tangent vector to the second coordinate system by $\tilde{u} = \mathrm{d}f \cdot u$, then calculate $\tilde{y} = \mathrm{Exp}_{\tilde{x}}(\tilde{u})$ and finally transform back to the first coordinate system $y = f^{-1}(\tilde{y})$.

Now let $v = \xi^H + \mu^V$ and $w = v^H + \eta^V$ be two tangent vectors of a tangent bundle at point p. Scalar product of v and w with respect to metric tensor $\tilde{g}$ is given by

$$\langle v, w \rangle_p = \langle \xi, v \rangle_x + \langle \mu, \eta \rangle_x = \xi^T G(x)\, v + \mu^T G(x)\, \eta, \tag{2.20}$$

where $\langle \cdot, \cdot \rangle_x$ is a scalar product on $\mathcal{M}$. Let $\gamma(t) = (x(t), u(t))$ be a smooth curve on $T\mathcal{M}$. γ is a geodesic of $T\mathcal{M}$ if following holds

$$\tilde{\nabla}_{\dot{\gamma}} \dot{\gamma} = 0, \tag{2.21}$$

where $\tilde{\nabla}$ is the covariant derivative of the metric $\tilde{G}$. by decomposing $\dot{\gamma} = \mu^H + v^V$ to its horizontal μ and vertical v components, geodesic equation can be written as [38]

$$\nabla_\mu \mu = R(u, v)\mu, \tag{2.22}$$

$$\nabla_\mu v = 0, \tag{2.23}$$

where $R(u, v)w$ and ∇ are the Riemannian curvature tensor and covariant derivative of $\mathcal{M}$. Solving (2.22) and (2.23) does not yield a closed form solution even if $\mathcal{M}$ has a closed

form solution for its geodesic equation. However, exponential and logarithmic mappings of a tangent bundle can be calculated numerically by using expressions of a base manifold [38]. Exponential map can therefore be obtained using numerical integration of (2.22) and (2.23), while the logarithmic mapping is obtained by optimisation procedure.

Let $p = (x, u) \in T\mathcal{M}$ be a point of the tangent bundle and $w = (v, \mu) \in T_pT\mathcal{M}$ be a tangent vector. The exponential map of tangent bundle can be calculated by repeating following equations N times [38]:

$$x_{k+1} = \mathrm{Exp}_{x_k} \varepsilon v_k, \tag{2.24}$$

$$u_{k+1} = \mathrm{P}_{x_k \to x_{k+1}} (u_k + \varepsilon \mu_k), \tag{2.25}$$

$$v_{k+1} = \mathrm{P}_{x_k \to x_{k+1}} (v_k - \varepsilon R(u_k, \mu_k) v_k), \tag{2.26}$$

$$\mu_{k+1} = \mathrm{P}_{x_k \to x_{k+1}} (\mu_k), \tag{2.27}$$

where $\varepsilon = \frac{1}{N}$ is the step size. Note that the exponential map, parallel transport and curvature tensor in above equations are evaluated on $\mathcal{M}$ and are available in closed form for many manifolds used in robotics.

To calculate the logarithmic mapping of the tangent bundle $\mathrm{Log}_p(q)$, the curve on the tangent bundle is approximated as a discrete set of points $p_0, \ldots, p_{N+1}$, where $p_i = (x_i, u_i)$, $p_0 = p$ and $p_{N+1} = q$. Mid points p_i are initialised in a way that the points $x_i \in \mathcal{M}$ lie on the geodesic connecting x_0 and x_{N+1}, and tangents $u_i \in T_{x_i}\mathcal{M}$ are obtained by linear interpolation of u_0 and u_{N+1}. The points p_i, in general, do not lie on the geodesic of $T\mathcal{M}$, however, by minimising the energy of the discretized curve we can get a good approximation of the geodesic. Let $v_i = (\mathrm{Log}_{x_i}(x_{i+1}) - \mathrm{Log}_{x_i}(x_{i-1}))\frac{N}{2}$ be the central finite difference at point x_i and let $\mu_i = (\mathrm{P}_{x_{i+1} \to x_i}(u_{i+1}) - \mathrm{P}_{x_{i-1} \to x_i}(u_{i-1}))\frac{N}{2}$ be the central finite difference of tangent u_i at point x_i. Tangents v_0 and μ_0 are calculated as a finite differences between (x_0, u_0) and (x_1, u_1), while v_{N+1} and μ_{N+1} are calculated as finite differences between (x_N, u_N) and (x_{N+1}, u_{N+1}) because the central finite difference cannot be used at edge points. The energy of discretized curve is then given by [38]

$$E = \sum_{i=1}^{N} \left(\|v_i\|_{x_i}^2 + \|\mu_i\|_{x_i}^2 \right). \tag{2.28}$$

The gradients of E with respect to x_i and u_i are given by

$$\mathrm{grad}_{x_i} E = \nabla_{v_i} v_i + R(u_i, \mu_i) v_i, \tag{2.29}$$

$$\mathrm{grad}_{u_i} E = \nabla_{v_i} \mu_i. \tag{2.30}$$

The covariant derivatives $\nabla_{v_i} v_i$ and $\nabla_{v_i} \mu_i$ are approximated as central finite differences of v_i and μ_i respectively:

$$\nabla_{v_i} v_i \approx (\mathrm{P}_{x_{i+1} \to x_i}(v_{i+1}) - \mathrm{P}_{x_{i-1} \to x_i}(v_{i-1}))\frac{N}{2}, \tag{2.31}$$

$$\nabla_{v_i} \mu_i \approx (\mathrm{P}_{x_{i+1} \to x_i}(\mu_{i+1}) - \mathrm{P}_{x_{i-1} \to x_i}(\mu_{i-1}))\frac{N}{2}. \tag{2.32}$$

Now the steepest descend or conjugate gradient algorithm may be used to find the geodesic of $T\mathcal{M}$ connecting p and q and the logarithmic map is given by

$$\mathrm{Log}_p q = v_0^H + \mu_0^V. \tag{2.33}$$

2.6 Examples of Riemannian Manifolds

2.6.1 *Sphere*

The proposed approach can be applied to any RM but in this dissertation, as a case study, we take the n-dimensional sphere. Thus, the scalar product of tangent vectors u and v at point $x \in \mathcal{S}_\rho^n$ is $\langle u, v \rangle_x = u^T v$, i.e. the metric tensor, is the identity matrix. The distance between $x, y \in \mathcal{S}_\rho^n$ can be computed as [39]

$$d(x, y) = \rho \arccos \frac{x^T y}{\rho^2} \tag{2.34}$$

defining great circles as the geodesic lines. The exponential map of the tangent vector u at point x is defined as [14]

$$\mathrm{Exp}_x(u) = x \cos \frac{\|u\|}{\rho} + \frac{u}{\|u\|} \rho \sin \frac{\|u\|}{\rho}, \tag{2.35}$$

while the logarithmic map amount to the following expression

$$\mathrm{Log}_x y = d_{\mathcal{M}}(x, y) \frac{\rho^2 y - x^T y\, x}{\|\rho^2 y - x^T y\, x\|}. \tag{2.36}$$

Parallel transport of a tangent vector u from point x to point y along a geodesic is finally given by [14]

$$\mathrm{P}_{x \to y}(u) = u - \frac{\mathrm{Log}_x(y)^T u}{d(x, y)^2} \Big[\mathrm{Log}_x(y) + \mathrm{Log}_y(x) \Big]. \tag{2.37}$$

Riemann curvature tensor of the sphere is

$$R(u, v)w = \langle v, w \rangle\, u - \langle u, w \rangle\, v, \tag{2.38}$$

where $u, v, w \in T_x \mathcal{S}_\rho^n$ for some $x \in \mathcal{S}_\rho^n$.

Parametrisation of a sphere

In this section we provide the local coordinate representation of the n-sphere of radius ρ. The motivation to use local coordinates is so that we could define the covariance of the concentrated Gaussian distribution in those coordinates. If we would use the representation of the ambient Euclidean space, the covariance would lie in the tangent space of the sphere, which is the hyperplane in the ambient space. Hence, the resulting covariance matrix would not be of the full rank.

There are many different ways to parametrise the sphere, for example polar coordinates and stereographic projection. However, it is not possible to cover the whole n-dimensional sphere using only n parameters. Hence we will use 2 different parametrisations: stereographic projection from the north pole and stereographic projection from the south pole. These two coordinate systems form an atlas of the sphere [35]. In the north pole stereographic projection the local coordinates of a point $x = (x^1, \ldots, x^{n+1}) \in \mathcal{S}_\rho^n$ are

$$\xi^i = \frac{\rho\, x^i}{\rho - x^{n+1}}, \qquad i = 1, \ldots, n. \tag{2.39}$$

Note that this local representation is valid for all x except north pole $(0, \ldots, \rho)$. The inverse of the mapping (2.39) is

$$x^i = \frac{2\rho^2}{\rho^2 + \|\xi\|^2}\,\xi^i, \qquad i = 1, \ldots, n, \tag{2.40}$$

$$x^{n+1} = \frac{\|\xi\|^2 - \rho^2}{\rho^2 + \|\xi\|^2}\,\rho. \tag{2.41}$$

The conversion of the tangent vector $u \in T_x\mathcal{S}^n_\rho$ from ambient coordinates to its representation in a north pole stereographic projection the Jacobian of the coordinate transformation (2.39) is used:

$$\tilde{u}^j = \frac{\partial \xi^j}{\partial x^i} u^i, \tag{2.42}$$

where u^i are components of u in ambient coordinates, $\tilde{u}^j$ are components of u in local coordinates and

$$\frac{\partial \xi^j}{\partial x^i} = \begin{cases} \frac{\rho}{\rho - x^{n+1}}\,\delta_{ij}, & i = 1, \ldots, n, \\ \frac{\rho\, x^j}{(\rho - x^{n+1})^2}, & i = n+1, \end{cases} \tag{2.43}$$

where δ_{ij} is the Kronecker delta, i.e. $\delta_{ij} = 1$ if $i = j$ and 0 otherwise. To transform the tangent vector back to ambient coordinates, Jacobian of the inverse mapping $\frac{\partial x^j}{\partial \xi^j}$ is used.

Similarly, the local coordinates of x in the south pole stereographic projection are

$$\eta^i = \frac{\rho\, x^i}{\rho + x^{n+1}}, \qquad i = 1, \ldots, n, \tag{2.44}$$

and is valid for all x except the south pole $(0, \ldots, -\rho)$. The inverse of this projection is given by

$$x^i = \frac{2\rho^2}{\rho^2 + \|\eta\|^2}\,\eta^i, \qquad i = 1, \ldots, n, \tag{2.45}$$

$$x^{n+1} = \frac{\rho^2 - \|\eta\|^2}{\rho^2 + \|\eta\|^2}\,\rho. \tag{2.46}$$

The conversion between these local coordinates is given by following

$$\eta^i = \frac{\rho^2}{\|\xi\|^2}\,\xi^i, \qquad \xi^i = \frac{\rho^2}{\|\eta\|^2}\,\eta^i, \qquad i = 1, \ldots, n. \tag{2.47}$$

To convert tangent vectors from one representation to other, or between ambient coordinates to local coordinates and vice versa, the Jacobian of coordinate transformation must be used. For example, let u^i be components of the tangent vector u in ξ coordinates. Then the components of the vector u in η coordinates are given by [35]

$$\tilde{u}^j = \frac{\partial \eta^j}{\partial \xi^i} u^i. \tag{2.48}$$

The components of the Jacobian in this case are

$$\frac{\partial \eta^i}{\partial \xi^j} = \frac{\rho^2}{\|\xi\|^2}\delta_{ij} - \frac{2\rho^2}{\|\xi\|^4}\xi^i\xi^j. \tag{2.49}$$

The components of the metric tensor for both of these local representations is given by [39]

$$g_{ij} = \frac{4\rho^4}{\left(\rho^2 + \|\xi\|^2\right)^2}\delta_{ij}, \tag{2.50}$$

hence all expressions of the sphere (exponential and logarithmic map, parallel transport and curvature tensor) are same in both coordinate systems. However, it is more convenient to calculate those expressions in ambient coordinates and then convert the result back to local coordinates because the expressions in local coordinates are more complex. The Christoffel symbols of the metric tensor g_{ij} are

$$\Gamma^k_{ij} = -\frac{2}{\rho^2 + \|\xi\|^2}\left(\xi^i\delta_{jk} + \xi^j\delta_{ik} - \xi^k\delta_{ij}\right). \tag{2.51}$$

2.6.2 *Space of Symmetric Positive Definite Matrices*

The space of $n \times n$ symmetric positive definite (SPD) matrices $\mathcal{P}_n$ is a Riemannian manifold of dimension $d = \frac{1}{2}n(n+1)$. The tangent space $T_X\mathcal{P}_n$ of $\mathcal{P}_n$ at point X is a space of a symmetric $n \times n$ matrices denoted by $\mathrm{Sym}(n)$ [40]. Due to the symmetry of SPD matrices, they can be represented by their upper-triangular (or lower-triangular) part. Hence, the point $X \in \mathcal{P}_n$ can be represented in the following way

$$v(X) = D_n^+ \mathrm{vec}(X), \tag{2.52}$$

where $\mathrm{vec}(X)$ is the vectorization operator and D_n^+ is the pseudo-inverse of a duplication matrix D_n [40]. Now the matrices

$$E_{ij} = \begin{cases} 1_{ii}, & \text{if } i = j, \\ 1_{ij} + 1_{ji}, & \text{if } i \neq j, \end{cases} \tag{2.53}$$

where 1_{ij} stands for $n \times n$ matrix whose (i,j)-th element is 1 and others are 0, form the basis of the space of symmetric matrices $\mathrm{Sym}(n)$ that corresponds to parametrisation $v(x)$ [40]. Thus any symmetric matrix X can be written as

$$X = x^\alpha E_\alpha. \tag{2.54}$$

where $\alpha = \imath(i,j)$ and $\imath$ converts a pair of indices (i,j) to a single index[1]. The scalar product $\langle U, V\rangle_X = \mathrm{tr}(X^{-1}UX^{-1}V)$, where $X \in \mathcal{P}_n$ and $U, V \in \mathrm{Sym}(n)$, defines the smooth metric tensor on $\mathcal{P}_n$. The elements of this metric tensor are [40]

$$g_{\alpha\beta}(X) = \mathrm{tr}\left(X^{-1}E_\alpha X^{-1}E_\beta\right) \tag{2.55}$$

[1] In this section, latin indices are assumed to run from 1 to n, while greek indices run from 1 to d.

and the Christoffel symbols are

$$\Gamma^{\alpha}_{\beta\gamma} = -\operatorname{tr}\left(E_\beta X^{-1} E_\gamma X^{-1} E^{*\alpha} X^{-1}\right), \tag{2.56}$$

where $E^{*\alpha} = g^{\alpha\beta} E_\beta$ is the dual basis of $\operatorname{Sym}(n)$ with respect to metric (2.55). The Riemann curvature tensor of $\mathcal{P}_n$ at point X is given by [41, 42]

$$R(U,V)W = \frac{1}{4}\Big\{\left[VX^{-1}, UX^{-1}\right]W + W\left[X^{-1}U, X^{-1}V\right]\Big\}, \tag{2.57}$$

where $U, V, W \in \operatorname{Sym}(n)$ while $[U, V] = UV - VU$ is the matrix commutator. The components of curvature tensor are

$$R^{\alpha}_{\beta\gamma\delta} = \frac{1}{4}\operatorname{tr}\Big\{\left[X^{-1}E^{*\alpha}, X^{-1}E_\beta\right]\cdot\left[X^{-1}E_\gamma, X^{-1}E_\delta\right]\Big\}. \tag{2.58}$$

The metric tensor (2.55) induces the exponential map given by [40]

$$\operatorname{Exp}_X(U) = X^{1/2}\exp\left(X^{-1/2}UX^{-1/2}\right)X^{1/2}, \tag{2.59}$$

where $X \in \mathcal{P}_n$, $U \in \operatorname{Sym}(n)$ and exp is the usual matrix exponential. The inverse of (2.59) is well defined for all $X, Y \in \mathcal{P}_n$ and is given by

$$\operatorname{Log}_X(Y) = X^{1/2}\log\left(X^{-1/2}YX^{-1/2}\right)X^{1/2}, \tag{2.60}$$

where log is the usual matrix logarithm. The distance between points X and Y on $\mathcal{P}_n$ is given by [40]

$$d_{\mathcal{P}_n}(X,Y) = \left\|\log\left(X^{-1/2}YX^{-1/2}\right)\right\|_F = \left(\sum_{i=1}^{n}\ln^2\lambda_i\right)^{\frac{1}{2}}, \tag{2.61}$$

where $\|\cdot\|_F$ is the Frobenius norm and λ_i are eigenvalues of $X^{-1}Y$. The parallel transport of a tangent vector $U \in \operatorname{Sym}(n)$ along a geodesic connecting points X and Y of $\mathcal{P}_n$ is given by [14]

$$\mathrm{P}_{X\to Y}(U) = A\,U\,A^T, \tag{2.62}$$

where $A = Y^{1/2}X^{-1/2}$.

2.6.3 *Space of Diagonal Positive Definite Matrices*

The space of $n \times n$ diagonal positive definite matrices, $\mathcal{D}_n$, is a Riemannian submanifold of $\mathcal{P}_n$ and its dimension is n. It can be parametrised by diagonal elements $(x_1, \ldots, x_n)$. The metric tensor of this manifold can be obtained from the metric tensor of $\mathcal{P}_n$ given by (2.55) by restricting X to bi diagonal matrix. It follows that the metric tensor is

$$g_{ij}(x) = \delta_{ij}\frac{1}{x_i^2} \tag{2.63}$$

and the scalar product of two tangent vectors $u, v \in T_x\mathcal{D}_n$ is

$$\langle u, v\rangle_x = \sum_{i=1}^{n}\frac{u^i\cdot v^i}{x_i^2}. \tag{2.64}$$

The exponential mapping of $\mathcal{D}_n$ is then

$$\mathrm{Exp}_x u = (x_1 \exp \frac{u^1}{x_1}, ..., x_n \exp \frac{u^n}{x_n}) \tag{2.65}$$

and the logarithmic map is given by

$$\mathrm{Log}_x y = (x_1 \log \frac{y_1}{x_1}, ..., x_n \log \frac{y_n}{x_n}). \tag{2.66}$$

The distance between two points on $\mathcal{D}_n$ is

$$d_{\mathcal{D}_n}(x, y) = \sqrt{\sum_{i=1}^{n} \log^2 \frac{y_i}{x_i}}. \tag{2.67}$$

Components of Riemannian curvature tensor of $\mathcal{D}_n$ are all 0, i.e.

$$R(u, v)w = 0 \tag{2.68}$$

for all u, v and $w \in T_x\mathcal{D}_n$.

2.7 Statistics on Riemannian Manifolds

The mean of N points, $\{x_1, ..., x_n\} \in \mathcal{M}$, can be calculated as the Kärcher mean by minimizing the quadratic error [14]

$$\mu = \arg\min_{x \in \mathcal{M}} \frac{1}{N} \sum_{i=1}^{N} d^2_{\mathcal{M}}(x, x_i). \tag{2.69}$$

When points are close enough, a unique global solution called the Fréchet mean can be computed using the Gauss-Newton optimisation. The covariance matrix of points $\{x_1, ..., x_n\}$ in the tangent space $T_\mu\mathcal{M}$ is $\Sigma = \frac{1}{N} \sum_{i=1}^{N} \mathrm{Log}_\mu(x_i)\mathrm{Log}_\mu(x_i)^T$.

To generalise a Gaussian distribution to RMs, it is defined in the tangent space at its mean value [14]

$$\mathcal{N}_{\mathcal{M}}(x \mid \mu, \Sigma) = \frac{1}{\sqrt{\det(2\pi\Sigma)}} \exp\left(-\frac{1}{2} d^2_M(\mu, x)\right), \tag{2.70}$$

where $d_M(\mu, x) = \sqrt{\mathrm{Log}_\mu(x)^T \, \Sigma^{-1} \, \mathrm{Log}_\mu(x)}$ is the generalised Mahalanobis distance.

2.8 Summary

In this chapter we have presented an overview of the Riemannian geometry. We started by stating the definition of a Riemannian manifold as an differentiable manifold coupled with the smooth metric structure. Then, we showed that each differentiable manifold can be approximated with an Euclidean space of the same dimension at the neighbourhood of each of its points. Such Euclidean spaces are called tangent spaces. We proceeded with introducing geodesics – the shortest lines on Riemannian manifolds. Exponential and logarithmic mappings are then introduced as maps between the manifold and its tangent spaces. Next, we consider the space of all tangent spaces of an Riemannian manifold which is called tangent bundle of a manifold. It is shown that the tangent bundle of a Riemannian manifold is Riemannian manifold as well. Finally, some statistical tools necessary for state estimation on Riemannian manifolds are introduced.

Chapter 3

State Estimation and Multi Object Tracking

In the state estimation, the goal is to determine the state of some dynamical system based on the sequence of noisy measurements. Usually, the state is not completely observable by the sensor used to collect measurements. For example, when tracking multiple objects, the state of an object often includes both position and velocity, however, it is not always possible to measure the velocity of the object directly. The measurement noise is present due to the sensor imperfections and some unknown environment conditions, while the process noise is used to account for the uncertainties of the dynamical model of the system.

The multi target tracking (MTT) is the extension of the state estimation where, in addition to the unknown objects' states, the number of objects is unknown and time varying as well. Together with the variable number of targets and noisy data, the MTT systems often cope with the uncertainty in the measurement origin which makes it a very challenging problem. Hence, many modern MTT methods handle the problem of measurement origin uncertainty separately and then proceed with the estimation of each object's state independently.

The rest of this chapter is organized as follows. Section 3.1 introduces the paradigm of Bayesian filtering and Kalman filter (KF) as one example of closed-form solutions to Bayesian estimation problem. Then, section 3.2 gives brief overview on the state-of-the-art MTT algorithms. And finally, some evaluation methods for multi target tracking systems are presented in section 3.6.

3.1 Bayesian Estimation

The aim of Bayes filtering is to estimate the probability density function (PDF) $p_k\left(x_k \mid y_{1:k}\right)$ of the unknown state vector x_k given the sequence of detections $y_{1:k} \triangleq \{y_1, \ldots, y_k\}$. The time evolution of the state can be often described with nonlinear transition function

$$x_k = f_{k-1}(x_{k-1}, u_{k-1}, w_{k-1}), \tag{3.1}$$

where u_{k-1} denotes the control input of the system and w_{k-1} denotes the process noise. Since in target tracking, which is the main interest of this thesis, the control input u_{k-1} is usually unknown, it will be omitted in continuation. The unknown input is therefore treated as the process noise. The state transition (3.1) of the stochastic system can also be expressed with Markov transition density [43]

$$f_{k|k-1}\left(x_{k|k-1} \middle| x_{k-1}\right). \tag{3.2}$$

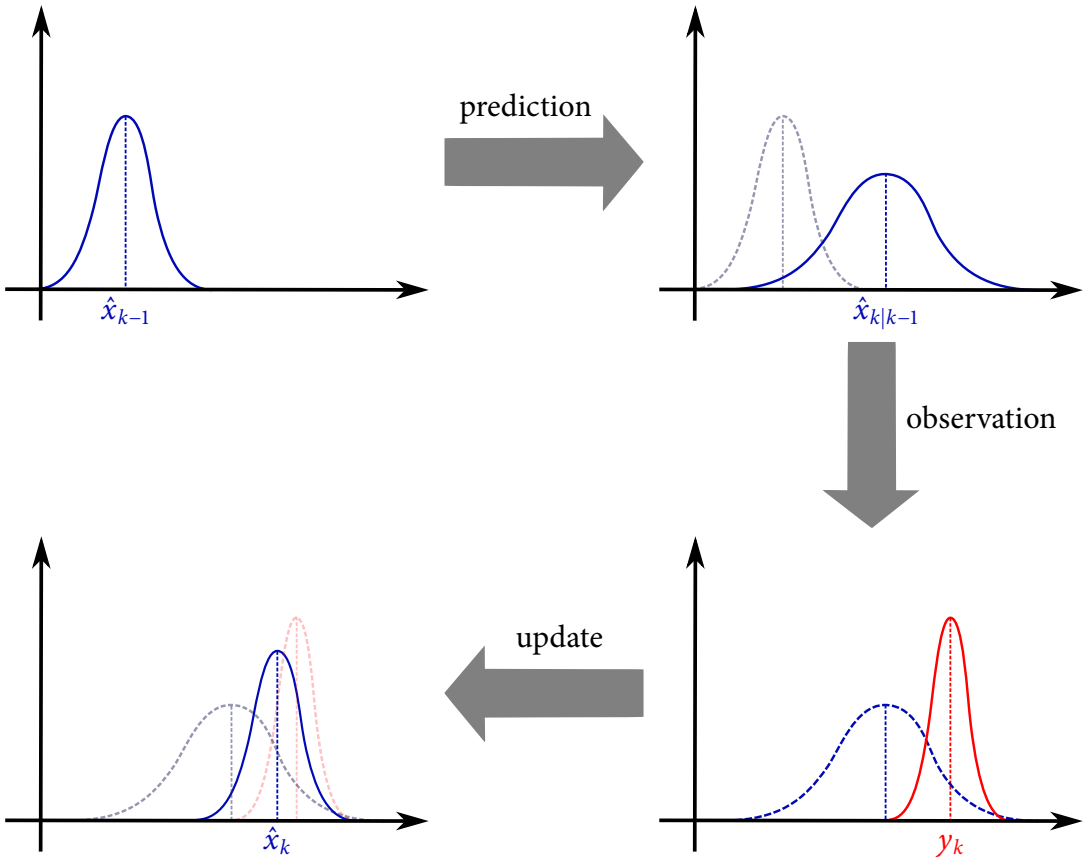


Figure 3.1: This illustration shows the one cycle of the Bayes filter. At the beginning of a cycle (top-left plot), the prior probability density of a state is shown with solid blue line and its maximum a posteriori (MAP) estimate is denoted with $\hat{x}_{k-1}$. The Bayes filter then proceeds with the prediction step which results with a new probability density (top-right) and a new MAP estimate $\hat{x}_{k|k-1}$. It can be seen that the new PDF is wider due to the uncertainties of the dynamical model of the system. Now, the new observation denoted with y_k is collected (bottom-right) and the likelihood function of the unknown state x given this new observation is illustrated with solid red line. Given this likelihood and the predicted PDF, the posterior density calculated using the update step of the Bayes filter is shown with the solid blue line (bottom-left).

Similarly, the observation of the stochastic system can be described with nonlinear model

$$y_k = g_k(x_k, v_k), \tag{3.3}$$

where v_k denotes the measurement noise, or alternatively by a likelihood function [43]

$$g_k\left(y_k \,|\, x_k\right). \tag{3.4}$$

Suppose that the PDF of state is known at time 0 and denote it with $p_0(x_0)$. The Bayes filter then recursively estimates the posterior density p_k in two steps. Firstly, given the prior PDF $p_{k-1}\left(x_{k-1} \,|\, y_{1:k-1}\right)$, the predicted density $p_{k|k-1}\left(x_{k|k-1} \,\middle|\, x_{k-1}, y_{1:k-1}\right)$ is calculated. Then, given the prediction from previous step and the new measurement z_k, the updated density $p_k\left(x_k \,\middle|\, x_{k|k-1}, y_{1:k}\right)$ is obtained. The whole cycle of the filter, including prediction and update step, is illustrated on figure 3.1. The Bayesian filtering utilizes the Markov property which

says that the current state x_k contains the information about its whole past [43]. In other words, the previously mentioned PDFs can be simplified to

$$p_{k-1}\left(x_{k-1} \mid y_{1:k-1}\right) = p_{k-1}(x_{k-1}), \tag{3.5}$$

$$p_{k|k-1}\left(x_{k|k-1} \middle| x_{k-1}, y_{1:k-1}\right) = p_{k|k-1}\left(x_{k|k-1} \middle| x_{k-1}\right), \tag{3.6}$$

$$p_k\left(x_k \middle| x_{k|k-1}, y_{1:k}\right) = p_k(x_k). \tag{3.7}$$

The prediction of the PDF is obtained by Chapman-Kolmogorov equation from the prior density and Markov transition density (3.2) [43, 44]

$$p_{k|k-1}(x_{k|k-1}) = \int f_{k|k-1}\left(x_{k|k-1} \middle| x_{k-1}\right) p_{k-1}(x_{k-1})\, \mathrm{d}x_{k-1}. \tag{3.8}$$

The updated PDF is then obtained using Bayes rule and the total probability theorem

$$p_k(x_k) = \frac{g_k\left(y_k \mid x_k\right) p_{k|k-1}(x_k)}{\int g_k\left(y_k \mid x_k\right) p_{k|k-1}(x_k)\, \mathrm{d}x_k}. \tag{3.9}$$

To obtain an optimal estimate $\hat{x}_k$ of the state x_k given the posterior PDF $p_k(x_k)$, one must apply some of the estimators that minimizes the particular Bayes risk [44]. Such estimate is called Bayes optimal estimate. Most commonly used Bayes optimal estimators are MAP

$$\hat{x}_k^{MAP} = \arg\sup_{x_k} p_k(x_k) \tag{3.10}$$

and expected a posteriori (EAP) estimator

$$\hat{x}_k^{EAP} = \int x_k\, p_k(x_k)\, \mathrm{d}x_k. \tag{3.11}$$

The Bayesian filtering recursion does not admit the closed-form solution in general, hence some approximations are necessary to implement the filter. One popular implementation of Bayes filter is the particle filter, which approximates the densities by randomly generated samples. However, particle filters are not used throughout this thesis and hence are not further studied here, but curious readers are referred to [43, 45]. Under linear Gaussian assumptions (linear system and Gaussian process and measurement noise), it can be shown that if the prior density $p_{k-1}(x_{k-1})$ is Gaussian, the $p_k(x_k)$ is Gaussian as well. This leads to the closed-form solution, the Kalman filter [46] which is presented in section 3.1.1.

3.1.1 *Kalman Filter*

Introduced by Kalman in [46], the KF is an example of Bayesian filters. Although it was originally derived using least squares (LS) technique, it can be shown that it is a closed-form solution to Bayesian filtering recursion when the system is linear Gaussian assumptions [47].

Consider a discrete stochastic linear time-invariant system described with

$$x_k = F_{k-1}x_{k-1} + L_{k-1}w_{k-1}, \tag{3.12}$$

$$y_k = H_k x_k + M_k v_k. \tag{3.13}$$

The time evolution of that system is captured by equation (3.12), where x is the state of the system, w is the zero-mean Gaussian process noise signal of covariance Q_{k-1}, while F_{k-1} and L_{k-1} are matrices of corresponding dimensions. The observations of the system's state are modelled by (3.13), where y is the observation, v is the zero-mean Gaussian measurement noise of covariance R_k, while H_k and M_k are matrices of adequate dimensions. Suppose that the PDF of the state vector at previous is a Gaussian distribution, i.e. $p_{k-1}(x_{k-1}) = \mathcal{N}\left(x_{k-1} \mid \hat{x}_{k-1}, \hat{P}_{k-1}\right)$, the prediction of the KF is then given by

$$\hat{x}_{k|k-1} = F_{k-1}\hat{x}_{k-1}, \tag{3.14}$$

$$\hat{P}_{k|k-1} = F_{k-1}\hat{P}_{k-1}F_{k-1}^T + L_{k-1}Q_{k-1}L_{k-1}^T. \tag{3.15}$$

Similarly, the PDF of the predicted observation is also Gaussian and its mean and covariance are given by

$$\hat{y}_{k|k-1} = H_k\hat{x}_{k|k-1}, \tag{3.16}$$

$$\hat{S}_{k|k-1} = H_k\hat{P}_{k|k-1}H_k^T + M_kR_kM_k^T. \tag{3.17}$$

Given the observation y_k, the update of the KF is

$$K_k = \hat{P}_{k|k-1}H_k^T\hat{S}_{k|k-1}^{-1}, \tag{3.18}$$

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k\left(y_k - \hat{y}_{k|k-1}\right), \tag{3.19}$$

$$\hat{P}_k = \left(I - K_kH_k\right)\hat{P}_{k|k-1}. \tag{3.20}$$

Instead of (3.20), the Joseph form

$$\hat{P}_k = \left(I - K_kH_k\right)\hat{P}_{k|k-1}\left(I - K_kH_k\right)^T + K_kM_kR_kM_k^TK_k^T \tag{3.21}$$

is often used to ensure the positive-definiteness of the covariance matrix.

3.1.2 *Extended Kalman Filter*

Let now consider the non-linear system defined by

$$x_k = f_{k-1}\left(x_{k-1}, w_{k-1}\right), \tag{3.22}$$

$$y_k = h_k\left(x_k, v_k\right), \tag{3.23}$$

where $w_{k-1} \sim \mathcal{N}\left(\cdot \mid 0, Q_{k-1}\right)$ and $v_k \sim \mathcal{N}\left(\cdot \mid 0, R_k\right)$ are respectively the zero-mean non-additive process and measurement noises. The KF introduced in previous section cannot be applied in this case due to non-linear functions f_{k-1} and h_k. One of often used adaptation of the KF to non-linear systems is the extended Kalman filter (EKF). The EKF uses the first order Taylor approximation of the non-linear functions f_{k-1} and h_k in order to propagate the covariance matrix $\hat{P}$. Let F_{k-1} and L_{k-1} be following Jacobians of (3.22)

$$F_{k-1} = \left.\frac{\partial f_{k-1}}{\partial x_{k-1}}\right|_{\hat{x}_{k-1},0}, \qquad L_{k-1} = \left.\frac{\partial f_{k-1}}{\partial w_{k-1}}\right|_{\hat{x}_{k-1},0}, \tag{3.24}$$

and let H_k and M_k be the Jacobians of (3.23)

$$H_k = \left.\frac{\partial h_k}{\partial x_k}\right|_{\hat{x}_{k|k-1},0}, \qquad M_k = \left.\frac{\partial h_k}{\partial v_k}\right|_{\hat{x}_{k|k-1},0} \tag{3.25}$$

The time prediction of the EKF is obtained by the non-linear model (3.22)

$$\hat{x}_{k|k-1} = f_{k-1}\left(\hat{x}_{k-1}, 0\right). \tag{3.26}$$

The covariance matrix $\hat{P}$ is predicted as in linear KF by equation (3.15), where F_{k-1} and L_{k-1} are calculated by (3.24). The prediction of the measurement $\hat{y}_{k|k-1}$ is obtained from the non-linear function (3.23)

$$\hat{y}_{k|k-1} = h_k\left(\hat{x}_{k|k-1}, 0\right). \tag{3.27}$$

Given the predicted measurement (3.27) and the Jacobians (3.25), the rest of the update step of the EKF follows the and the update procedure of EKF is given by equations eqs. (3.17) – (3.21).

3.1.3 *Unscented Kalman Filter*

The need to calculate the Jacobians of the non-linear state transition function f and measurement function g is the main drawback of the EKF. Even when it is possible to differentiate functions f and g, it might still be very tedious. Furthermore, the high non-linearity of the system may lead to the divergence of the first-order approximation used by EKF. Rather than using Jacobians, the unscented Kalman filter (UKF) [48] approximates the PDF $p_k(x_k) = \mathcal{N}\left(x_k \mid \hat{x}_k, \hat{P}_k\right)$ with a set of sample points. Note that, as opposed to the particle filter, the sample points in UKF are chosen deterministically.

Consider the following non-linear system

$$x_k = f_{k-1}(x_{k-1}) + w_{k-1}, \tag{3.28}$$

$$y_k = g_k(x_k) + v_k, \tag{3.29}$$

where w_{k-1} and v_k are zero-mean, mutually independent, Gaussian process and measurement noises respectively. Note that this model uses the additive process and measurement noise, however, other implementations of UKF exists that may use non-additive noise models, for example see [49]. Suppose that the prior PDF is Gaussian with the mean $\hat{x}_{k-1}$ and covariance $\hat{p}_{k-1}$. The goal is to approximate the PDF of the predicted state $p_{k|k-1}(x_{k|k-1})$ with the Gaussian distribution $\mathcal{N}\left(x_{k|k-1} \mid \hat{x}_{k|k-1}, \hat{P}_{k|k-1}\right)$. In order to achieve this, the prior Gaussian distribution must be mapped with the non-linear state transition model (3.28), which, in general, does not result in a Gaussian distribution. To alleviate this problem, the unscented transform is applied [48]. The prior distribution $p_{k-1}(x_{k-1})$ is approximated with $2n+1$ samples called sigma points, where n is the dimension of the system. The sigma points are generated by the following equations

$$\sigma_0 = \hat{x}_{k-1}, \tag{3.30}$$

$$\sigma_j = \hat{x}_{k-1} + \alpha\sqrt{\kappa}A_j, \qquad j = 1, \ldots, n, \tag{3.31}$$

$$\sigma_{n+j} = \hat{x}_{k-1} - \alpha\sqrt{\kappa}A_j, \qquad j = 1, \ldots, n, \tag{3.32}$$

where A_j is the j-th column of the Cholesky factor of the covariance matrix, $\hat{P}_{k-1} = A\,A^T$. Each sigma point has its first order and second order weights given by following equations

$$w_0^{(1)} = \frac{\alpha^2\kappa - n}{\alpha^2\kappa}, \qquad w_0^{(2)} = w_0^{(1)} + 1 - \alpha^2 + \beta, \tag{3.33}$$

$$w_j^{(1)} = w_j^{(2)} = \frac{1}{2\alpha^2\kappa}, \qquad j = 1, \ldots, 2n. \tag{3.34}$$

The parameters α and κ control the spread of the sigma points, while β depends on the distribution and the optimal value for the Gaussian distribution is $\beta = 2$. There is also a simpler parametrization of UKF given in [48] where the sigma points and corresponding weights are calculated according to

$$\sigma_0 = \hat{x}_{k-1}, \qquad w_0 = \frac{\kappa}{n+\kappa}, \tag{3.35}$$

$$\sigma_j = \hat{x}_{k-1} + \sqrt{n+\kappa}A_j, \qquad w_j = \frac{1}{2(n+\kappa)}, \qquad j = 1, \ldots, n, \tag{3.36}$$

$$\sigma_{n+j} = \hat{x}_{k-1} - \sqrt{n+\kappa}A_j, \qquad w_j = \frac{1}{2(n+\kappa)}, \qquad j = 1, \ldots, n. \tag{3.37}$$

Given the sigma point representation of the prior Gaussian PDF, the unscented transform can be calculated in the following way. Firstly, the transformed mean is obtained as the weighted mean of transformed sigma points

$$\hat{x}_{k|k-1} = \sum_{i=0}^{2n} w_i^{(1)} f_{k-1}(\sigma_i). \tag{3.38}$$

The transformed covariance is then calculated by using second order weights

$$\hat{P}_{k|k-1} = \sum_{i=0}^{2n} w_i^{(2)} \left[f_{k-1}(\sigma_i) - \hat{x}_{k|k-1}\right]\left[f_{k-1}(\sigma_i) - \hat{x}_{k|k-1}\right]^T + Q_{k-1}, \tag{3.39}$$

where Q_{k-1} is the process noise covariance matrix.

Now the another unscented transform is applied to obtain the predicted measurement and the innovation covariance. Given the predicted probability density function $p_{k|k-1}(x_{k|k-1}) = \mathcal{N}\left(x_{k|k-1} \mid \hat{x}_{k|k-1}, \hat{P}_{k|k-1}\right)$ the new set of sigma points $\bar{\sigma}_j$ and weights $\bar{w}_j^{(1)}$ and $\bar{w}_j^{(2)}$ are calculated using the same procedure as before. The predicted measurement is calculated by

$$\hat{y}_{k|k-1} = \sum_{i=0}^{2n} \bar{w}_i^{(1)} g_k(\bar{\sigma}_i), \tag{3.40}$$

while the innovation covariance is given by

$$\hat{S}_{k|k-1} = \sum_{i=0}^{2n} \bar{w}_i^{(2)} \left[g_k(\bar{\sigma}_i) - \hat{y}_{k|k-1}\right]\left[g_k(\bar{\sigma}_i) - \hat{y}_{k|k-1}\right]^T + R_k, \tag{3.41}$$

where R_k is the measurement noise covariance matrix. Additionally, the cross-covariance is

$$C_{xy} = \sum_{i=0}^{2n} \bar{w}_i^{(2)} \left[\bar{\sigma}_i - \hat{x}_{k|k-1}\right]\left[g_k(\bar{\sigma}_i) - \hat{y}_{k|k-1}\right]^T, \tag{3.42}$$

The Kalman gain of the UKF is given by

$$K_k = C_{xy}\hat{S}_{k|k-1}^{-1} \tag{3.43}$$

and the updated PDF is $\mathcal{N}\left(x_k \mid \hat{x}_k, \hat{P}_k\right)$, where

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k\left(y_k - \hat{y}_{k|k-1}\right), \tag{3.44}$$

$$\hat{P}_k = \hat{P}_{k|k-1} - K_k\hat{S}_{k|k-1}K_k^T. \tag{3.45}$$

3.1.4 *Von Mises-Fisher Filter*

The probability density given by

$$f\left(x;\mu,\kappa\right) = C_n(\kappa)\exp\left(\kappa\mu^T x\right), \tag{3.46}$$

where x and μ are points on the unit sphere $\mathcal{S}^n$ and $\kappa \in \mathbb{R}_+$ is called the von Mises-Fisher (vMF) distribution. The normalization constant is

$$C_n(\kappa) = \frac{\kappa^{(n-1)/2}}{(2\pi)^{n+1}I_{(n-1)/2}(\kappa)}, \tag{3.47}$$

where $I_p(\cdot)$ denotes the p-th order modified Bessel function of the first kind [20] which is given by the expression

$$I_p(x) = \frac{1}{\pi}\int_0^\pi e^{x\cos\theta}\cos(p\theta)\,\mathrm{d}\theta - \frac{\sin(p\pi)}{\pi}\int_0^\infty e^{-x\cosh t - pt}\,\mathrm{d}t. \tag{3.48}$$

The positive constant κ is the concentration parameter of the vMF distribution and controls the spread of the distribution. The vMF distribution belongs to the exponential family of distributions which is a very important property [24]. The expectation of the vMF distribution, or the directional mean, is given by

$$\mathbb{E}\left[x\right] = A_{n+1}(\kappa)\mu, \tag{3.49}$$

where

$$A_n(\kappa) = \frac{I_{(n+1)/2}}{I_{(n-1)/2}}. \tag{3.50}$$

The proof of (3.49) can be found in [24].

Special cases of the von Mises-Fisher distributions are distribution on the unit circle (also known as the von-Mises distribution) and the vMF distribution on the unit sphere. The normalization constant and the ratio of Bessel functions (3.50) in the latter case simplify to

$$C_2(\kappa) = \frac{\kappa}{4\pi\sinh\kappa}, \tag{3.51}$$

$$A_2(\kappa) = \frac{1}{\tanh\kappa} - \frac{1}{\kappa}. \tag{3.52}$$

motion and observation model

In von Mises-Fisher filtering it is often assumed that the object moves slowly compared to the sensor's sampling rate. Markovic *et al.* in [24] stated that in that case it is practical to approximate the Markov transition density with the vMF density

$$f_{k|k-1}\left(x_{k|k-1}\,\middle|\,x_{k-1}\right) = f\left(x_{k|k-1}; x_{k-1}, \kappa_\tau\right), \tag{3.53}$$

where κ_τ models the uncertainty of the process. As in [24], the von Mises-Fisher distribution is chosen as the measurement model given by

$$p\left(y_k \,|\, x_k\right) = f\left(y_k; x_k, \kappa_o\right), \tag{3.54}$$

where κ_o models the uncertainty of the measurement process.

state prediction and update

The prediction and update equations for vMF filter are obtained from Bayesian filtering equations (3.8) and (3.9). Under the assumption that both the prior distribution $p_{k-1}(x_{k-1})$ and the Markov transition density $f_{k|k-1}\left(x_{k|k-1}\,\middle|\,x_{k-1}\right)$ are vMF densities, the resulting predicted density $p_{k|k-1}$ unfortunately is not vMF. In [24] Markovic *et al.* proposed to approximate the resulting density with the von Mises-Fisher density by moment-matching technique. Hence, given the prior density $p_{k-1}(x) = f\left(x; \hat{\mu}_{k-1}, \hat{\kappa}_{k-1}\right)$, the predicted density is also vMF density $p_{k|k-1}(x) = f\left(x; \hat{\mu}_{k|k-1}, \hat{\kappa}_{k|k-1}\right)$, where

$$\hat{\mu}_{k|k-1} = \hat{\mu}_{k-1}, \tag{3.55}$$

$$\hat{\kappa}_{k|k-1} = A_n^{-1}\left(A_n(\kappa_\tau)A_n(\hat{\kappa}_{k-1})\right). \tag{3.56}$$

The function A_n^{-1} is the inverse of the function (3.50) that can be computed numerically [24]. Given the new measurement y_k and using the von Mises-Fisher measurement model, the update equations of the vMF filter are given by [24]

$$\hat{\mu}_k = \frac{\hat{\kappa}_{k|k-1}\hat{\mu}_{k|k-1} + \kappa_o y_k}{\hat{\kappa}_k}, \tag{3.57}$$

$$\hat{\kappa}_k = \left\|\hat{\kappa}_{k|k-1}\hat{\mu}_{k|k-1} + \kappa_o y_k\right\|. \tag{3.58}$$

3.2 Introduction to Multiple Object Tracking

MTT is a challenging problem where the goal is to simultaneously estimate the states of numerous objects that are present in the field of view of a sensor [50]. In addition to the uncertainty of the underlying motion model that each object follows, there are many more The objects may disappear from the scene as well as new objects may appear, hence the number of objects is time-varying and has to be estimated together with the states of individual objects. Detections obtained by the sensor are in no particular order and often cannot be easily associated with previous detections. Moreover, the detections can be cluttered with spurious data that cannot be distinguished from true detections. Thus,

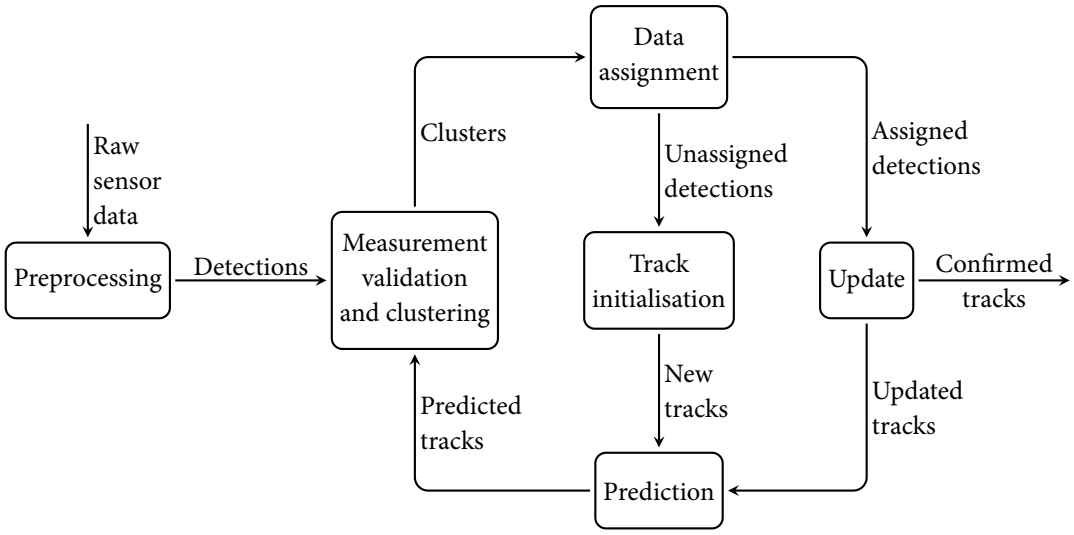


Figure 3.2: Typical flowchart of an MTT algorithm. New detections are first validated and then separated together with predicted tracks in independent clusters. Data association is then performed for each cluster and the tracks are updated with assigned detections. New tracks are initialised based on unassigned detections. Finally, the prediction step is performed on both updated and new tracks to predict their position in the next step.

most of the MTT methods include some kind of data association that tries to match new detections with tracked objects. The flowchart of the typical multi target tracking algorithms is shown on figure 3.2. In most applications it is assumed that the objects are small enough so they can be treated as points. However, sometimes such assumption is not possible and each object can generate it's own cluster of detections, and furthermore, detections of multiple objects may be merged in one detection due to the finite resolution of the sensor. These two problems of having multiple detections originating from the same object and multiple objects sharing same detections pose a tremendous challenge to already difficult problem of data association. In this thesis, the following premises about the MTT problem are made:

a) objects move and generate detections independently of each other,

b) new objects may appear on the scene and existing objects may disappear,

c) each object may either be detected and generate one detection or be undetected in each frame,

d) each detection may belong to at most one object,

e) spurious detections are independent of each other and of ground truth objects.

The first multi target tracking methods emerged in the second half of the previous century and were applied in various areas such as air traffic control, air defence and missile defence [6]. Initial MTT solutions involved the use of single target trackers in parallel for each target. Such trackers are known in literature as nearest neighbour (NN) filters because each target is updated with the closest detection. The main limitation of NN is that each single target filter is run independently of each other, hence the same detection

could be used to update more than one target. The global nearest neighbour (GNN) filter is the improvement of NN, where the measurement-to-target association is done globally, usually by applying Hungarian algorithm [51] to find the optimal assignment. Nevertheless, GNN is still prone to divergence when there are many targets tightly spaced. Thus, many MTT algorithms have been developed in recent years and multi target tracking is still actiev research field today. Current state-of-the-art MTT algorithms can be divided in three different categories [1] (i) probabilistic data association (PDA), (ii) multiple hypothesis tracking (MHT) and (iii) random finite set (RFS) approaches. The rest of this section covers some examples of MTT algorithms belonging to these three groups.

3.3 Probabilistic Data Association

In [2] Bar-Shalom and Tse proposed the probabilistic data association filter (PDAF) for single object tracking in a presence of false alarms. Rather than choosing only one measurement to update the object state, the PDAF takes into account all available measurements that might have originated from the object. It is assumed that the false alarms are independent and identically distributed according to uniform distribution over the surveillance region. Further, assume that at most one of the received measurements may belong to the target and denote with $\mathscr{H}_k^i$ event that the i-th measurement at time k is correct. Also, denote with $\mathscr{H}_k^0$ that all measurements at time k are clutter. The association probabilities are given by [2]

$$\beta_k^0 \triangleq \Pr\left(\mathscr{H}_k^0\right) = C\left(1 - p_d p_g\right), \tag{3.59}$$

$$\beta_k^i \triangleq \Pr\left(\mathscr{H}_k^i\right) = C\lambda^{-1} p_d p_g g\left(y_{k,i} \,|\, \hat{x}_k\right), \tag{3.60}$$

where p_d is the detection probability, p_g is the gating probability, λ is clutter density, $g\left(y_{k,i} \,|\, \hat{x}_k\right)$ is the likelihood of measurement $y_{k,i}$ given that the target is detected and gated and that its state is $\hat{x}_k$, and finally, C is the normalization constant. Due to the fact that these events are mutually exclusive and exhaustive, the association probabilities are [2]

$$\beta_k^0 = \frac{1 - p_d p_g}{1 - p_d p_g + \sum_{j=1}^{m_k} \lambda^{-1} p_d p_g g\left(y_{k,i} \,|\, \hat{x}_k\right)}, \tag{3.61}$$

$$\beta_k^i = \frac{\lambda^{-1} p_d p_g g\left(y_{k,i} \,|\, \hat{x}_k\right)}{1 - p_d p_g + \sum_{j=1}^{m_k} \lambda^{-1} p_d p_g g\left(y_{k,i} \,|\, \hat{x}_k\right)}. \tag{3.62}$$

3.3.1 *Integrated Probabilistic Data Association*

The PDAF assumes that the object is always present in the field of view of a sensor. However, in many real-world application the object may disappear from the scene and then reappear later. To adapt the PDAF for such environment, a heuristics for track initialisation and termination is necessary. Alternative is to use integrated probabilistic data association filter (IPDAF) introduced by Musicki *et al.* in [52]. Instead of simply assuming that the object is present, the existence probability is estimated in IPDAF. Then the track can be terminated when its existence probability falls below some threshold.

3.3.2 *Joint Probabilistic Data Association*

Joint probabilistic data association filter (JPDAF) [3] is an extension of PDAF to the multi-target scenario. Similarly to the PDAF, it is assumed that the number of targets is known and they cannot disappear. At each sensor scan, obtained detections are validated for each track thus forming a validation matrix with binary values that indicate if a measurement could have originated from a target. In order to calculate a posteriori data association probabilities it is necessary to generate all possible joint measurement to track associations. In case of large number of targets and measurements, data association problem can become intractable for a real-time application. To avoid this problem, it is proposed in [5] to consider only a fixed number of possible associations that are most likely. To find most likely association, joint data association problem is formulated as a series of integer linear programs.

3.3.3 *Joint Integrated Probabilistic Data Association*

To allow tracking of a time varying number of targets, a joint integrated probabilistic data association filter (JIPDAF) is proposed [4, 53]. JIPDAF predicts the target state individually for each track. If we construct an appropriate target motion model, then the state of each target can be propagated using the standard Kalman filter prediction step. Additionally, the target existence probability prediction is given by

$$p_{k|k-1}\left(\mathcal{H}^j \,\middle|\, Y_{1:k-1}\right) = p_S\, p_{k-1}\left(\mathcal{H}^j \,\middle|\, Y_{1:k-1}\right), \tag{3.63}$$

where p_S is the target survival probability which is assumed to be independed of the target's state, $Y^k = \{y_i^k\}_{i=1}^{m_k}$ is the set of all observations at time k where m_k is the number of observations at time k. $Y_{1:k}$ is the set of all observations up to and including time k and $\mathcal{H}^j$ is the hypothesis that track j exists.

Let $\nu_{i,j} = y_i - \hat{y}_j$ denote the innovation of the i-th measurement to the track j, where y_i is the measurement and $\hat{y}_j$ is the predicted measurement for track j. Time superscripts are omitted here for clarity. The target state is then corrected by the Kalman filter update equation

$$\hat{x}_k = \hat{x}_{k|k-1} + K_{j,k}\nu_j, \tag{3.64}$$

where $\nu_j = \sum_{i=1}^{m_k} \beta_{i,j}\,\nu_{i,j}$ is the weighted innovation, $\beta_{i,j}$ are posterior association probabilities, and $K_{j,k}$ is the Kalman gain for target j. The update of the covariance matrix slightly differs from the original Kalman update step [3]

$$P_{j,k} = P_{j,k|k-1} - (1-\beta_{0,j})K_{j,k}S_{j,k}K_{j,k}^T + \tilde{P}_{j,k}, \tag{3.65}$$

$$\tilde{P}_{j,k} = K_{j,k}\left[\sum_{i=1}^{m_k}\beta_{i,j}\nu_{i,j}\nu_{i,j}^T - \nu_j\nu_j^T\right]K_{j,k}^T, \tag{3.66}$$

where $S_{j,k}$ is the innovation covariance of the target j.

The combinatorial computational complexity of the JIPDA can be alleviated by discarding the assignment hypotheses that are unlikely. Since the innovation of the measurement is a zero-mean normal distribution, the measurement validation can be achieved by selecting only those measurement that lie in the confidence ellipsoid of the target [2]. A priori

likelihood function of a measurement i given state of a target j after validating with the gating probability p_G is given by

$$g_{i,j} \triangleq g(y_i \mid \hat{y}_j) = p_G^{-1} \mathcal{N}(y_i; \hat{y}_j, S_j), \tag{3.67}$$

when y_i is inside the validation gate and zero otherwise.

To calculate a posteriori association probabilities $\beta_{i,j}$, all possible joint association events must be considered. In each event, one target can be associated with at most one detection, and each detection cannot be assigned to more than one target. Let $\mathcal{A}$ denote the set of all joint association events. Since those events are exhaustive and mutually exclusive, the probability of the joint event $\mathcal{A}_i$ is given by [4]

$$\begin{aligned} p(\mathcal{A}_i \mid Y_{1:k}) = C \times & \prod_{j \in T_0} (1 - p_D\, p_G\, p_{\mathcal{H}^j}) \\ & \times \prod_{j \in T_1} \left(p_D\, p_G\, p_{\mathcal{H}^j}\, g_{i,j}\, \lambda^{-1} \right), \end{aligned} \tag{3.68}$$

where C is the normalization constant, p_D is detection probability, λ is clutter density, T_0 and T_1 are sets of tracks assigned with no measurements and with one measurement in $\mathcal{A}_i$ and $p_{\mathcal{H}^j} = p_{k|k-1}(\mathcal{H}^j | Y_{1:k-1})$.

Let $\mathcal{H}_i^j$ be the hypothesis that the measurement i belongs to target j and $\mathcal{H}_\varnothing^j$ the hypothesis that the target j was not detected. The posterior probabilities of individual track existence and measurement association can be obtained by [4]

$$p(\mathcal{H}^j, \mathcal{H}_i^j \mid Y_{1:k}) = \sum_{\mathcal{A} \in \mathcal{A}(i,j)} p(\mathcal{A} \mid Y_{1:k}), \tag{3.69}$$

$$\begin{aligned} p(\mathcal{H}^j, \mathcal{H}_\varnothing^j \mid Y_{1:k}) = & \frac{(1 - p_D p_G) p_{\mathcal{H}^j}}{1 - p_D p_G p_{\mathcal{H}^j}} \\ & \times \sum_{\mathcal{A} \in \mathcal{A}(\varnothing,j)} p(\mathcal{A} \mid Y_{1:k}), \end{aligned} \tag{3.70}$$

where $\mathcal{A}(i,j)$ is the set of all events that assign measurement i to track j, while $\mathcal{A}(\varnothing, j)$ is the set of all events in which track j was missed. Given probabilities (3.69) and (3.70), the a posteriori track existence probability is computed as

$$\begin{aligned} p_k\left(\mathcal{H}^j \middle| Y_{1:k}\right) = & p(\mathcal{H}^j, \mathcal{H}_\varnothing^j \mid Y_{1:k}) \\ & + \sum_{i \in \{\mathcal{M}_{i,j}=1\}} p(\mathcal{H}^j, \mathcal{H}_i^j \mid Y_{1:k}), \end{aligned} \tag{3.71}$$

where $\mathcal{M}_{i,j}$ is the element of the validation matrix, while a posteriori association probabilities are given by

$$\beta_{i,j} = \frac{P(\mathcal{H}^j, \mathcal{H}_i^j \mid Y_{1:k})}{p_k(\mathcal{H}^j \mid Y_{1:k})}. \tag{3.72}$$

3.4 Multiple Hypothesis Tracking

Multiple hypothesis tracking (MHT), introduced by Reid [6], is a deferred decision approach to data association based multi-target tracking [1]. To deal with the unknown

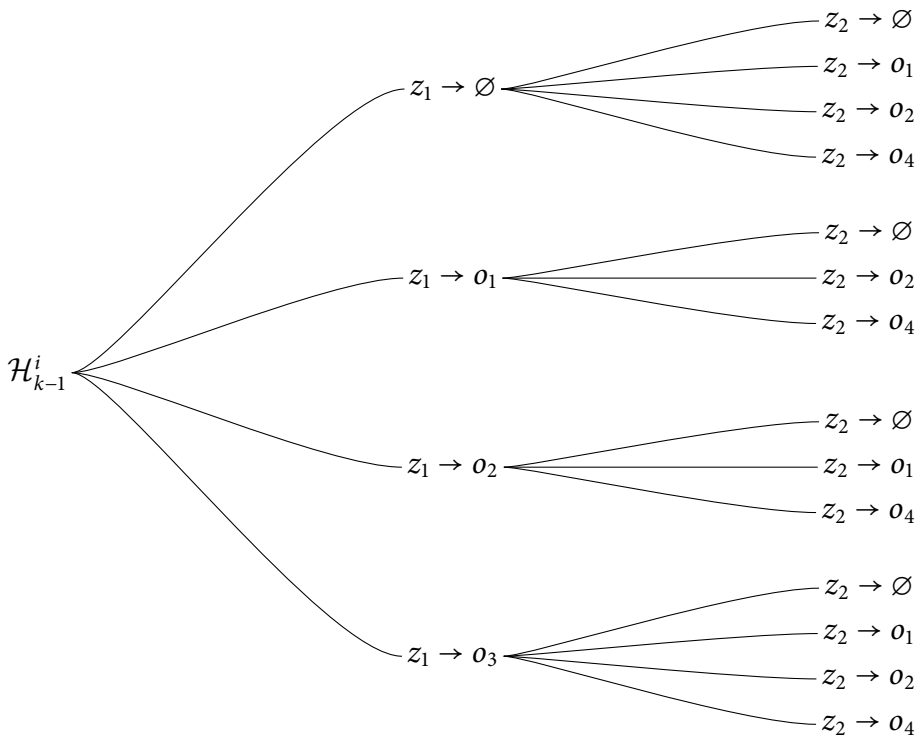


Figure 3.3: The illustration of hypothesis tree generation of the hypothesis oriented multiple hypothesis tracking. In this example the prior hypothesis $\mathcal{H}^i_{k-1}$ contains two targets o_1 and o_2. Given two measurements z_1 and $z_2 \in Z_k$, new hypotheses are constructed by branching from $\mathcal{H}^i_{k-1}$. First, 4 branches are created by assigning z_1 to o_1, o_2, considering it as a false alarm, and finally, by assigning it to a new target denoted with o_3. Then, the same process is applied to each branch for the measurement z_2. Note that measurements z_1 and z_2 are not assigned to the same object in any branch.

measurement-to-tracks association, MHT keeps a set of possible association hypotheses until more data is collected[54]. After a new set of measurements is available, a new set of hypotheses is created from the existing hypotheses and their posterior probabilities are updated using Bayes rule. To avoid exponential growth of a hypotheses, a heuristic pruning method is needed to discard unlikely hypotheses. By creating and pruning of hypotheses, the MHT algorithms inherently handles track initiation and termination, and hence accommodates to an unknown and time-varying number of targets. Based on the most likely hypothesis, a single-target Bayes filter is used to update the states of each targets using measurements associated with them.

Multiple hypothesis tracking algorithms can be divided into two groups – hypothesis oriented multiple hypothesis tracking (HOMHT) and track oriented multiple hypothesis tracking (TOMHT). The MHT algorithm proposed by Reid in [6] is the hypothesis-oriented, which means that it keeps a number of global hypotheses between consecutive sensor scans [1].

3.4.1 *Hypothesis Oriented Multiple Hypothesis Tracking*

The algorithm proposed by Reid in [6] is an example of the hypothesis oriented multiple hypothesis tracking (HOMHT) method. The HOMHT maintains the list of possible

measurement-to-target association hypotheses. After each sensor scan, the set of hypotheses is expanded with new measurements. Denote with $\mathcal{H}_{k-1}$ the set of global hypotheses at time $k-1$ and with Y_k the set of measurements collected at time k. Starting from the prior set of hypotheses $\mathcal{H}_{k-1}$, new hypotheses of the HOMHT are formed by iteratively combining measurements form the new scan with them [6]. Let $\bar{\mathcal{H}}_k^i$ be the set of hypotheses after the i-th measurement from Y_k has been applied and initialize $\bar{\mathcal{H}}_k^0$ with $\mathcal{H}_{k-1}$. The set of hypotheses $\bar{\mathcal{H}}_k^{i+1}$ is formed by creating branches from every hypothesis in $\bar{\mathcal{H}}_k^i$. The following assumptions are taken into account when generating new hypotheses. Each measurement may be assigned to one target, considered as a false alarm, or may be assigned to a new target. Each target may only be associated with one measurement from the same sensor scan. The whole procedure is illustrated on figure 3.3. Denote with Ψ_k the set of hypotheses about the measurement-to-target associations in time step k. Then, each global hypothesis $\mathcal{H}^i{}_k$ may be considered as a joint hypothesis that some prior hypothesis $\mathcal{H}_{k-1}^a$ and some association hypothesis Ψ_k^b are true. The probability of each global hypothesis $\mathcal{H}_k^i$ can then be evaluated recursively by utilizing Bayes rule [6]

$$P\left(\mathcal{H}_k^i \,\middle|\, Z_k\right) = \frac{1}{C} \cdot P\left(Z_k \,\middle|\, \mathcal{H}_{k-1}^a, \Psi_k^b\right) \cdot P\left(\Psi_k^b \,\middle|\, \mathcal{H}_{k-1}^a\right) \cdot P(\mathcal{H}_{k-1}^a), \tag{3.73}$$

where C is the normalization constant. The first term in (3.73) is the product of likelihoods of measurements in Z_k given prior hypothesis $\mathcal{H}_{k-1}^a$ and association hypothesis Ψ_k^b [6].

3.4.2 *Track Oriented Multiple Hypothesis Tracking*

Instead of generating and maintaining the tree of global hypotheses, the TOMHT constructs the tree for each target [1, 54]. TOMHT approaches can be further split to tree based and non-tree based [1]. In the rest of this section, only the tree-based variant is considered. Each branch of a target tree of an TOMHT represents one hypothesis for that target. As new measurements are available, each target tree is expanded by assigning different measurements to it or by making a hypothesis that the target is missed by the sensor in current scan. This whole process is illustrated on figure 3.4. From the list of all target hypotheses at present time, the best global hypothesis can be obtained by solving the multi-dimensional assignment problem [1]. The multi-dimensional assignment problem refers to the problem of assigning n consecutive sets of sensor scans to targets in such a way that it obtains highest total track score. Each global hypothesis must fulfil following constraints: each measurement can by assigned to at most one target and each target can be associated with at most one detection from one sensor scan.

The track score that is used to rank global hypotheses in TOMHT is the logarithm of likelihood ratio normalized by the probability density of false alarms [1]. Let $\mathrm{LLR}_{i,k-1}$ denote the score of the i-th target at time $k-1$. The score of i-th target at time k in hypothesis h_a can then be computed by [1]

$$\mathrm{LLR}_{i,k} = \mathrm{LLR}_{i,k-1} + \Delta\mathrm{LLR}_{i,k}. \tag{3.74}$$

Depending on the detection assigned to i-th target in hypothesis h_a, the incremental track score is given by

$$\Delta\mathrm{LLR}_{i,k} = \log \frac{p_d g_k\left(y_{j,k} \,\middle|\, \hat{x}_{i,k|k-1}\right)}{\lambda_c} \tag{3.75}$$

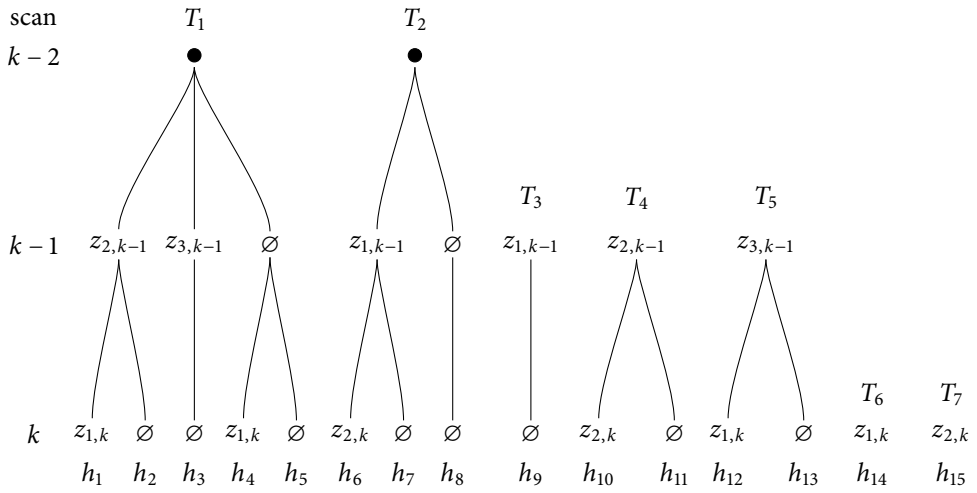


Figure 3.4: Formation of target trees of an tree-based TOMHT. In this example there are two confirmed targets T_1 and T_2 present at time $k-2$. There are three detections collected at time $k-1$ denoted with $z_{1,k-1}$, $z_{2,k-1}$ and $z_{3,k-1}$, and two detections obtained at time k denoted with $z_{1,k}$ and $z_{2,k}$. The tree is generated for each target by assigning different detections or by assuming to it in time $k-1$ and k. Missed detections are also considered and are denoted with $\varnothing$ in the target tree. Additionally, new tentative target trees are initiated for each detection. Figure adapted from [1].

if j-th measurement is assigned to i-th target, or by

$$\Delta\text{LLR}_{i,k} = \log(1 - p_d p_g) \tag{3.76}$$

in the case of missed detection. Detection probability, gating probability and false alarm density in above expressions are denoted respectively by p_d, p_g and λ_c, while $g_k\left(y_{j,k} \,\middle|\, \hat{x}_{i,k|k-1}\right)$ denotes the measurement likelihood. Track scores of new targets are initialised to [1]

$$\text{LLR}_{\text{init}} = \log \frac{\lambda_{\text{init}}}{\lambda_c}, \tag{3.77}$$

where λ_{init} denotes the density of new targets.

In a tree-based TOMHT approach, the hypothesized target is represented by the target tree so the N-scan pruning can be performed resulting in a reduced number of tracks figure 3.4), as opposed to non-tree based TOMHT which cannot perform N-scan pruning.

Frank et al. proposed a graphical model representation of the TOMHT filter [55], where the track posterior distribution of TOMHTs was formulated as a factor-graph. This formulation allowed the approximate calculation of track marginal probabilities, which may be used for online parameter estimation. In [56] the track-oriented MHT logic has been extended to a through the wall radar tracking, where in addition to the direct returns there are also several returns caused by the reflections. Thus the multiple passes through the MHT logic are performed in order to update the tracks with direct as well as multipath detections. Another example of multi-target tracking with multipath detections is presented in [57], where the multiple-detection MHT was proposed. The developed algorithm was tested in the simulation for multi-target tracking using over-the-horizon radar (OTHR). Many other improvements and extensions of MHT algorithms can be found in [1, 54, 58–61].

3.4.3 *Probabilistic Multiple Hypothesis Tracking*

In [62] the probabilistic multiple hypothesis tracking (PMHT) was developed. As opposed to the traditional MHT algorithms, PMHT approaches the data association problem in a probabilistic manner, thus resulting in a much lower computational load as compared to the algorithm proposed in [6]. Further, the PMHT algorithm does not require the enumeration of measurement-to-track assignments and pruning because all measurements are assigned to all tracks thus it is an optimal empirical Bayesian multi-target tracker. PMHT algorithm was also applied to the multi-sensor tracking but it is optimal only when the all sensors are identical. Thus, in [63] Krieg and Gray implemented the general multi-sensor PMHT algorithm (msPMHT). In addition to measurement-to-track assignment and target states, the general msPMHT also estimates the sensor from which each measurement is collected. This allows the application of the algorithm in the scenarios in which the source of measurements are unknown or when the parameters of sensors are unknown. In [64] the maximum likelihood PMHT (ML-PMHT) was applied in a scenario of a sharply maneuvering targets implemented in both the Cartesian and delay-bearing measurement space. Furthermore, the maneuvering-model parametrization was introduced for ML-PMHT to deal with rapidly-maneuvering targets. Choi and Hong presented the PMHT algorithm that can simultaneously estimate target states and signal-to-noise ratio, the PMHT-S filter [65].

3.5 Random Finite Sets Approach

During last decade a great deal of progress has been achieved in the finite set statistics [12] resulting in a variety of RFS based multi-target tracking algorithms, one of which is probability hypothesis density (PHD), introduced by Mahler in [66]. Since PHD filter propagates only the first moment of a multi-target density it alleviates the computational load making it suitable for real-time applications. Although PHD recursion cannot be analytically solved in general case, in [8] Vo and Ma showed that under Gaussian mixture approximations the closed-form solution may be obtained for linear single-target motion and observation model, resulting in a Gaussian mixture (GM) implementation of the PHD filter. The main drawback of the PHD filter is that it rests on the assumption that the number of tracked objects is Poisson distributed thus making it unreliable in scenarios with high number of targets. To overcome this problem a cardinalized probability hypothesis density (CPHD) filter was introduced, which releases the premise of a Poisson distributed number of targets. In [9], analytic implementation of a CPHD filter is presented – the Gaussian mixture CPHD filter, while in [67] Mahler et al. developed an adaptive CPHD filter that can learn the unknown clutter rate and detection model while filtering. In [68] the GM CPHD filter was integrated with the digital road maps in order to track the ground moving targets. Lindgren et al. presented a CPHD filter with the target spawning model in [69]. Although, the prediction step in a CPHD filter may become computationally intractable with the general spawning model, it is shown that for certain cardinality distributions of the spawning model, computationally efficient expressions for the prediction step are derived. In Gaussian mixture PHD and CPHD algorithms, the appearance of the new targets is

modelled by Gaussian mixtures, thus it demands a large number of components to cover the entire surveillance area in the case when new targets may appear anywhere. Beard et al. proposed a partially uniform target birth model[1] for GM implementation of a PHD and CPHD filters in [70]. In [71] the cubature Kalman GM PHD and CPHD (CKF-GM-PHD and CKF-GM-CPHD) filters were implemented. Further, the adaptive gating was proposed for developed filters in order to reduce the computational complexity without the significant impact on the tracking performance. Although the GM approximation of the PHD/CPHD filters leads to the closed-form solution, it is also possible to implement them using sequential Monte Carlo (SMC) methods thus avoiding assumptions of a linear target motion and measurement model as shown in [72]. Another interesting application of the PHD filter is presented in [21], where Marković et al. developed a von Mises mixture implementation of the PHD filter to tackle the problem of tracking on a unit circle.

In [12] Mahler introduced another approximation to the multi-target Bayes recursion – the multi-target multi-Bernoulli (MeMBer) filter. As opposed to PHD and CPHD, MeMBer propagates full multi-target density with the assumption that the filtering density is a multi-Bernoulli RFS. Mahler also presented the Gaussian mixture implementation of the MeMBer filter. However, Vo et al. showed analytically that the proposed MeMBer recursion overestimates the number of targets resulting in a high number of false tracks [10, 73]. In addition, they proposed a improved approximation to the MeMBer recursion that deals with the cardinality bias – cardinality balanced MeMBer filter. Furthermore, the Gaussian mixture as well as sequential Monte Carlo implementations were provided. In [74] an adaptive version of the MeMBer filter is developed in order to accommodate to unknown and time-varying clutter rate and detection profile.

Neither of the RFS methods mentioned above are not designed to output whole target trajectories but rather to yield current estimated states in each time step. Hence, labeled RFS methods were introduced in [75], where δ-generalized labeled multi-Bernoulli (δ-GLMB) filter. In [76] an further improvements of the δ-GLMB filter were achieved by truncating the sums of a large number of terms by using K-shortest path algorithm as well as using the PHD filter as a efficient look-ahead strategy to reduce the number of computations. Additionally, in [77] is presented even more efficient implementation of a GLMB filter that combines prediction and correction into a single step, thus requiring only one truncation process per iteration. Reuter et al. presented the labeled multi-Bernoulli (LMB) filter in [11]. The LMB filter can be considered as a generalization of the above mentioned MeMBer filter with two important advantages – it outputs trajectories, and it does not suffer from a cardinality bias. Labeled multi-Bernoulli filter is also an efficient approximation of a δ-GLMB filter because it exploits GLMB's accuracy while avoiding its computational load due to the exponential growth in the number of components. Olofsson et al. applied the LMB filter to the sea ice tracking [78], moreover, they used spatial indexing in order to further simplify the clustering process of the filter, providing efficient partitioning of the filter into smaller and independent parts.

The main advantage in a random finite set approach is that it provides a way to systematically treat non-standard measurement models like extended targets and merged

[1] In RFS multi-target tracking literature the spontaneous target appearance is termed target birth, while target spawning refers to the new targets being created by existing targets

measurements as well as non-standard multi-target motion models such as interaction among the targets [12]. In [79] and [80] extended targets are considered. Such targets may generate multiple detections in a single sensor scan, hence it can be considered as a RFS observation. The PHD and CPHD filtering algorithms are developed, and Gaussian mixture implementations are given, as well as the particle implementations. Beard et al. implemented both GLMB and LMB multi-target filter in [81], where they modelled the extended targets by gamma Gaussian inverse Wishart (GGIW) distribution. They also showed that the GLMB filter performs better than the extended target CPHD filter. In [82], Scheel et al. applied the extended target LMB filter for a laser-based tracking of vehicles using separable likelihood model. In [83], the generalisation of a GLMB tracking method was applied to MTT with merged measurements. Merged measurements may appear due to the closely spaced targets in regards to the sensor resolution. In [84], a different type of a non-standard MTT problems is studied – the interaction among the targets. The goal was to develop the method for a pedestrian tracking that would systematically deal with the interaction among them. Thus the social force model was developed that encapsulates various factors that affect the human motion. Based on a developed social force model, the SMC and GM implementations of the social force PHD filter were provided.

3.5.1 *Random Finite Sets*

A random finite set (RFS) X on $\mathcal{X} \subseteq \mathbb{R}^d$ is a finite-set valued random variable which means that it takes values in the space of all finite subsets of $\mathcal{X}$, $\mathcal{F}(X)$. An RFS consists of n unordered random object states (random vectors) $x_1, ..., x_n \in \mathcal{X}$, where n is a discrete random variable. In other words, the RFS is a set of random vectors, such that the number of its elements is the random variable as well. Due to this property RFSs are a useful statistical tool in multiple object tracking in cluttered environment.

An RFS can be completely described by the discrete cardinality distribution

$$\rho(n) = \Pr\left(|\mathrm{X}| = \mathrm{n}\right) \tag{3.78}$$

that captures the cardinality (number of points) of RFS X and a set of symmetric joint probabilistic densities

$$p_n(x_1, ..., x_n) \tag{3.79}$$

that characterize the spatial distribution of points given the cardinality n. The probability distribution of the RFS X on $\mathcal{X}$ is the probability measure P on $\mathcal{F}(X)$ defined as

$$P(\mathcal{S}) = \Pr\left(\{\mathrm{X} \in \mathcal{S}\}\right), \tag{3.80}$$

where $\mathcal{S}$ is any Borel subset[2] of $\mathcal{F}(\mathcal{X})$. The probability density π of the RFS X on $\mathcal{X}$, if it exists, is given by [12]

$$P(\mathcal{S}) = \int_{\mathcal{S}} \pi(X)\mu(\,\mathrm{d}X), \tag{3.81}$$

where μ is a dimensionless measure on $\mathcal{F}(\mathcal{X})$ and the integral in (3.81) is a set integral defined by [12]

[2] Borel set is any set in a topological space that can be formed from open sets through the operations of countable union, countable intersection and relative complement.

$$\int_{\mathcal{S}} f(X)\mu(\mathrm{d}X) \quad = \quad \sum_{n=0}^{\infty} \frac{1}{n!} \int 1_{\mathcal{S}}\big(\{x_1, ..., x_n\} \big) f\big(\{x_1, ..., x_n\} \big) \lambda^n(\mathrm{d}x_1, ..., \mathrm{d}x_n), \quad (3.82)$$

where $1_{\mathcal{S}}$ is the indicator function for $\mathcal{S}$ and λ is a dimensionless Lebesgue measure on $\mathcal{X}$.

As it is case with ordinary probability densities, it is also possible to define the moments of RFSs. The first moment of the RFS X on $\mathcal{X}$, which is very significant in a multi target tracking, is defined by [12]

$$V(S) = \mathbb{E}\big[|X \cap S|\big] \quad (3.83)$$

for any $S \subseteq \mathcal{X}$. If V can be written as

$$V(S) = \int_S v(x)\ \mathrm{d}x \quad (3.84)$$

for any $S \subseteq \mathcal{X}$ and $v : \mathcal{X} \to [0, \infty)$, then v is called the intensity function or the probability hypothesis density (PHD) and V is called the intensity measure. Thus, the intensity measure gives the expected number of points of X that are contained in region S, while the intensity function gives the density of the expected number of points of X at x. There are also moments of higher order but they are not used often.

Poisson random finite set

A Poisson RFS X on $\mathcal{X}$ is the RFS which is completely defined by its intensity function v. The cardinality of a Poisson RFS is a Poisson discrete random variable with mean $\lambda = \int v(x)\ \mathrm{d}x$ and its elements are spatially distributed identically and independent of each other with probability density $v(\cdot)/\lambda$. The probability density of a Poisson RFS is given by

$$\pi\big(\{x_1, ..., x_n\} \big) = K^n e^{-\lambda} \prod_{i=1}^{n} v(x_i), \quad (3.85)$$

where K is the unit of volume on $\mathcal{X}$.

Independent identically distributed clusters

Independent and identically distributed clusters are a generalization of the Poisson RFS. They are uniquely defined by the intensity function $v(\cdot)$ and its cardinality distribution $\rho(\cdot)$ which can be any discrete distribution that satisfies

$$N = \sum_{n=0}^{\infty} n\rho(n) = \int v(x)\ \mathrm{d}x, \quad (3.86)$$

where N is expected cardinality. The probability density of a IID cluster X on $\mathcal{X}$ is

$$\pi\big(\{x_1, ..., x_n\} \big) = K^n n! \rho(n) \prod_{i=1}^{n} \frac{v(x_i)}{N}. \quad (3.87)$$

bernoulli random finite set

The Bernoulli RFS is either empty with the probability $1 - r$ or has a single element which is distributed by the probability densitiy $p(\cdot)$ on $\mathcal{X}$. Hence, the cardinality distribution of a Bernoulli RFS is a Bernoulli random variable with parameter r. The probability density π of a Bernoulli RFSX on $\mathcal{X}$ is given by

$$\pi(X) = \begin{cases} 1 - r, & X = \varnothing \\ Krp(x), & X = \{x\} \\ 0, & \text{otherwise} \end{cases} \tag{3.88}$$

where K is the unit of volume in $\mathcal{X}$. Since the Bernoulli RFS X on $\mathcal{X}$ is completely characterized by parameters r and $p(\cdot)$, its probability density is often abbreviated by (r, p).

multi-bernoulli random finite set

A multi-Bernoulli RFS X on $\mathcal{X}$ is a union of many independent Bernoulli RFSs X_i given by their existence probability r_i and probability density $p_i(\cdot)$ on $\mathcal{X}$

$$X = \bigcup_{i=1}^{M} X_i, \tag{3.89}$$

where M is a number of independent Bernoulli RFSs. The expected cardinality of a multi-Bernoulli RFS is given by $N = \sum_{i=1}^{M} r_i$. The probability density of X is given by

$$\pi(X) = \begin{cases} \prod_{i=1}^{M}(1 - r_i), & n = 0 \\ K^n \prod_{i=1}^{M}(1 - r_i) \sum_{1 \le i_1 \ne \ldots \ne i_n \le M} \prod_{j=1}^{n} \frac{r_{i_j} p_{i_j}(x_j)}{1 - r_{i_j}}, & n \le M \\ 0, & n > M \end{cases} \tag{3.90}$$

where $n = |X|$. Since the multi-Bernoulli RFS is fully defined by its parameters r_i and p_i, its probability density is often abbreviated by $\{(r_i, p_i)\}_{i=1}^{M}$.

3.5.2 *Multi-Target Bayes Filtering*

Since the number of targets and detections vary with the time and because the order of target state does not matter due to the unknown data association, it is natural to model multi-target state and detections as RFSs. Suppose that the target states evolve in a state space $\mathcal{X} \subseteq \mathbb{R}^{n_x}$ where n_x is the dimension of the state space, and the target states are partially observed in the observation space $\mathcal{Z} \subseteq \mathbb{R}^{n_z}$ where n_z is the dimension of the observation space. Then the multi-target state and detections at time k are realisations of a RFS and can be written in following form [44]

$$X_k = \{x_{k,1}, \ldots, x_{k,n(k)}\} \subset \mathcal{X}, \tag{3.91}$$

$$Z_k = \{z_{k,1}, \ldots, z_{k,m(k)}\} \subset \mathcal{Z}, \tag{3.92}$$

where $n(k)$ is the number of targets at time k, while $m(k)$ is the number of detections. The multi-target state X_k takes values on the space $\mathcal{F}(\mathcal{X})$ which is called multi-target state space and Z_k takes values on $\mathcal{F}(\mathcal{Z})$ which is called multi-target observation space.

In the rest of this section, the multi-target state transition and observation models are considered, then the multi-target Bayes recursion is given and finally, some methods for state extractions are proposed.

state transition model

The multi-target state transition model must incorporate the disappearance and appearance of objects in addition to the single-object dynamical model to allow the time-varying number of targets. The RFS theory allows us to model complex multi-target transition models, e.g. interaction between objects, however it may result in a probability densities that are hard to work with. Hence, it is often assumed, as it will be in this work too, that the targets move independent of each other and that the target appearance and disappearance is independent of other objects.

Let $X_{k-1} = \{x_{k-1,1}, \ldots, x_{k-1,n}\}$ be the multi-target state at time $k-1$. Then each object with the state x_{k-1} either continues to exist at time k with survival probability $p_{S,k}(x_{k-1})$ and evolves to the new state x_k according to the (single-target) probability density $f_{k|k-1}(x_k|x_{k-1})$ or it dies with probability $1 - p_{S,k}(x_{k-1})$. It is easy to see that the state transition of the single objects can be modelled as the Bernoulli RFS

$$S_{k|k-1}(x_{k-1}). \tag{3.93}$$

Then the state transition of all targets is the multi-Bernoulli RFS given by [44]

$$T_{k|k-1}(X_{k-1}) = \bigcup_{x_{k-1} \in X_{k-1}} S_{k|k-1}(x_{k-1}), \tag{3.94}$$

with parameters $\left\{\left(p_{S,k}(x_{k-1}),\ f_{k|k-1}(\cdot|x_{k-1})\right) : x_{k-1} \in X_{k-1}\right\}$. The probability density of the RFS of transitioned objects is given by [44]

$$\pi_{T,k|k-1}(X_k|X_{k-1}) = K_s^{|X_k|}(1 - p_{S,k})^{X_{k-1}} \sum_{\tau \in \mathcal{T}(X_k, X_{k-1})} q_{S,k,\tau}^{X_k}, \tag{3.95}$$

where

$$q_{S,k,\tau}(x) = \frac{p_{S,k}(\tau(x)) f_{k|k-1}(x|\tau(x))}{1 - p_{S,k}(\tau(x))}, \tag{3.96}$$

and $\mathcal{T}(W, X)$ is the set of all one-to-one mappings from W to X with convention that the summation is zero when $|W| > |X|$ and unity when $W = \varnothing$.

RFS considered above covers time-evolution and deaths of existing targets. If the appearance of new objects can be modelled by an RFS Γ_k than the multi-target state at time k is given by

$$X_k = T_{k|k-1}(X_{k-1}) \cup \Gamma_k, \tag{3.97}$$

and the multi-object transition density can be written as

$$f_{k|k-1}(X_k|X_{k-1}) = \sum_{W\subseteq X_k} \pi_{T,k|k-1}(W|X_{k-1})\pi_{\Gamma,k}(X_k \setminus W). \tag{3.98}$$

Multi-target transition model can also incorporate the target spawning (new targets generated by existing targets) [8, 69] but it not considered in this work and explicit formulae for multi-target transition densities can be found in [12, 44].

OBSERVATION MODEL

Similar to the multi-target transition model, here will only the standard multi-target detection model be considered. At the time k each object $x_k \in X_k$ is either detected with probability $p_{D,k}(x_k)$ according to the likelihood $g_k(z_k|x_k)$, or missed with probability $1-p_{D,k}(x_k)$. Thus, each target generates the detection which is a Bernoulli RFS

$$D_k(x_k) \tag{3.99}$$

with parameters $\big(p_{D,k}(x_k), g_k(\cdot|x_k)\big)$, and the collection of all target generated detections is a multi-Bernoulli RFS

$$\Theta_k(X_k) = \bigcup_{x_k\in X_k} D_k(x_k), \tag{3.100}$$

with parameter set

$$\Big\{\big(p_{D,k}(x_k),\ g_k(\cdot|x_k)\big) : x_k \in X_k\Big\}. \tag{3.101}$$

The probability density of the RFS $\Theta_k(X_k)$ is [44]

$$\pi_{\Theta,k}(Z_k|X_k) = K_o^{|Z_k|}(1-p_{D,k})^{X_k} \sum_{\tau\in\mathcal{T}(Z_k,X_k)} q_{D,k,\tau}^{Z_k}, \tag{3.102}$$

where

$$q_{D,k,\tau}(z) = \frac{p_{D,k}(\tau(z))\, g_k(z|\tau(z))}{1-p_{D,k}(\tau(z))}. \tag{3.103}$$

The set of false detections, which is assumed to be independent of targets, is modelled by RFS K_k and is often specified as a Poisson RFS in practical implementations. Finally, the multi-target observation at time k is given by

$$Z_k = \Theta_k(X_k) \cup K_k \tag{3.104}$$

and the multi-target likelihood is

$$g_k(Z_k|X_k) = \sum_{W\subseteq Z_k} \pi_{\Theta,k}(W|X_k)\, \pi_{K,k}(Z_k \setminus W). \tag{3.105}$$

MULTI TARGET BAYES RECURSION

In this section the Bayes filtering recursion introduced in section section 3.1 is generalised to the MTT scenario by using the RFS theory presented in section section 3.5. Suppose that

the all information about the multi-target system state is contained in a posteriori filtering density $\pi_{k-1}(X_{k-1}|Z_{1:k-1})$. Then the multi-target Bayes recursion is given by [44]

$$\pi_{k|k-1}(X_k|Z_{1:k-1}) = \int f_{k|k-1}(X_k|X)\, \pi_{k-1}(X|Z_{1:k-1})\, \mu(\mathrm{d}X), \tag{3.106}$$

$$\pi_k(X_k|Z_{1:k}) = \frac{g_k(Z_k|X_k)\, \pi_{k|k-1}(X_k|Z_{1:k-1})}{\int g_k(Z_k|X)\, \pi_{k|k-1}(X|Z_{1:k-1})\, \mu(\mathrm{d}X)}, \tag{3.107}$$

where $f_{k|k-1}(X_k|X_{k-1})$ and $g_k(Z_k|X_k)$ are previously defined multi-target transition density and likelihood.

The recursion given by eqs. (3.106) and (3.107) is intractable in real-time application due to the set integrals in addition to the combinatorial nature of the multi-target transition densities and likelihoods. To overcome the problem of a combinatorial explosion, many approximations to the full multi-target Bayes recursion were proposed. In [66] the first moment approximation of the multi-target filter, the so-called probability hypothesis density (PHD filter), was introduced and in [8] the closed-form solution to the PHD recursion was presented for the Gaussian mixture approximation, which will be studied more closely in section section 3.5.3. The PHD filter was further improved in [9, 85] in order to estimate the second moment of the cardinality distribution. Besides the moment approximations, in [10, 74] multi-Bernoulli approximation to the multi-target Bayes filter has been proposed.

State estimation

The multi-target Bayes filter outputs the multi-target probability density, hence to obtain the locations of targets it is necessary to extract the target states from the density $\pi_k(X_k|Z_{1:k})$. However, it is not easy to generalize the single-target EAP and MAP estimators to the multi-target case, as it is shown in [12, 44].

One possible approach to the state estimation problem is to make use of the first moment of the filtering density. Recall that the local maxima of the intensity function indicate locations with high expectancy of number of objects. Thus for the given cardinality n it is possible to obtain the target states as the n highest peaks of the intensity function, while the number of targets may be obtained using the MAP estimate of the cardinality distribution

$$\hat{n} = \arg\sup_n \rho_k(n|Z_{1:k}). \tag{3.108}$$

Besides the first moment visualization method described above, in [44] the marginal multi-object (MaM) and joint multi-object (JoM) estimators are proposed. The MaM estimate can be obtained in two steps. First the number of targets is MAP estimated by (3.108) and then the individual target states are estimated according to [44]

$$\hat{X}^{MaM} = \arg \sup_{X:|X|=\hat{n}} \pi_k(X|Z_{1:k}). \tag{3.109}$$

The JoM estimate can also be evaluated in two steps. First, determine the MAP estimate

$$\hat{X}_n = \arg \sup_{X:|X|=n} \pi_k(X|Z_{1:k}) \tag{3.110}$$

for each $n \geq 0$ and then obtain $\hat{n}$ as

$$\hat{n} = \arg\sup_{n} \pi_k(\hat{X}_n|Z_{1:k})\frac{c^n}{n!}, \tag{3.111}$$

where c is a dimensionless constant which determines accuracy and the convergence rate of the estimator [44]. The JoM estimate is then given by

$$\hat{X}_c^{JoM} = \hat{X}_{\hat{n}}. \tag{3.112}$$

3.5.3 *Probability Hypothesis Density Filter*

Although mathematically rigorous, the full multi-target Bayes recursion given by eqs. (3.106) and (3.107) it is computationally intractable in real time due to set integrals and combinatorial nature of multi-target probability densities. Thus, it is necessary to use some kind of approximation of the full multi-target recursion. In this section, the probability hypothesis density (PHD) filter is presented, which propagates only the first moment (intensity function) of the multi-target probability density to mitigate the computational complexity of a multi-target Bayes filter.

Suppose that the multi-target probability density $\pi_{k-1}(X_{k-1}\,|\,Z_{1:k-1})$ at time $k-1$ is a Poisson RFS with intensity function v_{k-1}. Suppose further that the multi-target system can be modelled in the following way [8]:

- At time k each target moves independently of other targets according to single-target Markov transition density $f_{k|k-1}(\cdot|\cdot)$ and continues to exist with probability $p_{S,k}(\cdot)$.
- RFS Γ_k of new targets is a Poisson RFS with intensity function $\gamma_k(\cdot)$ and is independent of existing targets.
- The RFS $B_{k|k-1}$ of targets spawned at time k by existing target is a Poisson RFS with intensity function $\beta_k(\cdot|\cdot)$.
- Each target is detected at time k with the probability $p_{D,k}(\cdot)$ and generates measurement independent of other targets according to likelihood $g_k(\cdot|\cdot)$.
- The RFS K_k of a false alarms at time k is a Poisson RFS with intensity function $\kappa_k(\cdot)$ and is independent of targets.

Mahler showed in [66] that under assumptions given above, the predicted multi-target density $\pi_{k|k-1}$ and updated posterior density π_k remain Poisson RFS densities. Therefore, it is possible to propagate only the intensity function instead of the full multi-target density. The Bayes recursion in terms of intensity functions is given by [8]

$$\begin{aligned} v_{k|k-1}(x) = &\int p_{S,k}(\xi) f_{k|k-1}(x\,|\,\xi)\, v_{k-1}(\xi)\, \mathrm{d}\xi \\ &+ \int \beta_{k|k-1}(x\,|\,\xi)\, v_{k-1}(\xi)\, \mathrm{d}\xi + \gamma_k(x), \end{aligned} \tag{3.113}$$

$$v_k(x) = \big(1 - p_{D,k}(x)\big) v_{k|k-1}(x) + \sum_{z \in Z_k} \frac{p_{D,k}(x) g_k(z \mid x)\, v_{k|k-1}(x)}{\kappa_k(z) + \int p_{D,k}(z) g_k(z \mid \xi)\, v_{k|k-1}(\xi)\, \mathrm{d}\xi}. \tag{3.114}$$

Note that both integrals in eqs. (3.113) and (3.114) are defined on $\mathcal{X} \subseteq \mathbb{R}^d$, while integrals in the full multi-target Bayes recursion are set integrals on $\mathcal{F}(\mathcal{X})$. Hence, propagating only the first moment of the multi-target density avoids a lot of the computational burden caused by data association problem.

Gaussian mixture implementation

Although recursion given by eqs. (3.113) and (3.114) is computationally feasible in real-time, the integrals cannot be solved analytically in general case. Therefore, sequential Monte Carlo (SMC) method is often used to approximate intensity functions [72]. However, under some additional assumptions on multi-target system model it is possible to obtain a closed-form solution of the PHD recursion [8], the Gaussian mixture probability hypothesis density (GM-PHD) filter. In addition to assumptions in the previous section, let's also assume the following [8]

- Each target follows a linear Gaussian dynamical model given by the Markov transition density

$$f_{k|k-1}\left(x_{k|k-1} \,\middle|\, x_{k-1}\right) = \mathcal{N}\left(x_{k|k-1} \mid F_{k-1} x_{k-1}, L_{k-1} Q_{k-1} L_{k-1}^T\right). \tag{3.115}$$

- Each target generates observations given by linear Gaussian measurement model

$$g_k(z_k \mid x_k) = \mathcal{N}\left(z_k \mid H_k x_k, M_k R_k M_k^T\right). \tag{3.116}$$

- The target survival and detection probabilities are state independent[3]

$$p_{S,k}(x) = p_{S,k}, \tag{3.117}$$

$$p_{D,k}(x) = p_{D,k}. \tag{3.118}$$

- The intensity function of the birth RFS Γ_k is a Gaussian mixture of the following form

$$\gamma_k(x) = \sum_{i=1}^{N_{\gamma,k}} w_{\gamma,k,i}\, \mathcal{N}\left(x \mid m_{\gamma,k,i}, P_{\gamma,k,i}\right), \tag{3.119}$$

  where $N_{\gamma,k}$ is a number of Gaussian components, $w_{\gamma,k,i} > 0$ is the weight of i-th component, $m_{\gamma,k,i}$ and $P_{\gamma,k,i}$ are mean and covariance of i-th Gaussian component.

[3] In [8] it was also shown that if the survival and detection probability are exponential mixtures, the closed form solution can still be obtained.

– The intensity of RFS $B_{k|k-1}$ spawned by target with previous state ξ is a Gaussian mixture of the form

$$\beta_{k|k-1}(x \mid \xi) = \sum_{i=1}^{N_{\beta,k}} w_{\beta,k,i} \mathcal{N}\left(x \mid F_{\beta,k-1,i}\xi + d_{\beta,k-1,i}, Q_{\beta,k-1,i}\right), \quad (3.120)$$

where $N_{\beta,k}$ is the number of Gaussian components, $w_{\beta,k,i} > 0$ is the weight of i-th component, $F_{\beta,k-1,i}$, $d_{\beta,k-1,i}$ and $Q_{\beta,k-1,i}$ are model parameters of i-th Gaussian component.

Prediction Step If the above assumptions hold and if the posterior intensity at time $k-1$ is a Gaussian mixture of the following form [8]

$$v_{k-1}(x) = \sum_{i=1}^{N_{k-1}} w_{k-1} \mathcal{N}\left(x \mid m_{k-1,i}, P_{k-1,i}\right), \quad (3.121)$$

then the predicted intensity at time k is also a Gaussian mixture of the form

$$v_{k|k-1}(x) = v_{S,k|k-1}(x) + v_{\beta,k|k-1}(x) + \gamma_k(x), \quad (3.122)$$

where $v_{S,k|k-1}$ is the intensity of the survived targets, $v_{\beta,k|k-1}$ is the intensity of targets spawned by targets existing at time $k-1$ and γ_k is the intensity of targets born at time k given by (3.119). The intensity of the target that survived is given by

$$v_{S,k|k-1}(x) = p_{S,k} \sum_{i=1}^{N_{k-1}} w_{k-1,i} \mathcal{N}\left(x \mid m_{S,k|k-1,i}, P_{S,k|k-1,i}\right), \quad (3.123)$$

$$m_{S,k|k-1,i} = F_{k-1} m_{k-1,i}, \quad (3.124)$$

$$P_{S,k|k-1,i} = F_{k-1} P_{k-1,i} F_{k-1}^T + L_{k-1} Q_{k-1} L_{k-1}^T. \quad (3.125)$$

The intensity of targets spawned by targets existing at time $k-1$ is given by

$$v_{\beta,k|k-1}(x) = \sum_{i=1}^{N_{k-1}} \sum_{j=1}^{N_{\beta,k}} w_{k-1,i} w_{\beta,k,j} \mathcal{N}\left(x \mid m_{\beta,k|k-1,i,j}, P_{\beta,k|k-1,i,j}\right), \quad (3.126)$$

$$m_{\beta,k|k-1,i,j} = F_{\beta,k-1,j} m_{k-1,i} + d_{\beta,k-1,j}, \quad (3.127)$$

$$P_{\beta,k|k-1,i,j} = F_{\beta,k-1,j} P_{k-1,i} F_{\beta,k-1,j}^T + Q_{\beta,k-1,j}. \quad (3.128)$$

Update Step If the above assumptions hold and if the predicted intensity at time k is a Gaussian mixture of the following form

$$v_{k|k-1}(x) = \sum_{i=1}^{N_{k|k-1}} w_{k|k-1,i} \mathcal{N}\left(x \mid m_{k|k-1,i}, P_{k|k-1,i}\right), \quad (3.129)$$

then the posterior intensity at time k is also a Gaussian mixture of the form [8]

$$v_k(x) = (1 - p_{D,k}) v_{k|k-1}(x) + \sum_{z \in Z_k} v_{D,k}(x \mid z), \quad (3.130)$$

where the first term is the intensity of the targets that are not detected at time k and the second term is the intensity of detected targets given by

$$v_{D,k}(x \mid z) = \sum_{i=1}^{N_{k|k-1}} w_{k,i}(z)\,\mathcal{N}(x \mid m_{k,i}(z), P_{k,i}), \tag{3.131}$$

$$w_{k,i}(z) = \frac{p_{D,k}\, w_{k|k-1,i}\, q_{k,i}(z)}{\kappa(z) + p_{D,k} \sum_{j=1}^{N_{k|k-1}} w_{k|k-1,j}\, q_{k,j}(z)}, \tag{3.132}$$

$$q_{k,i}(z) = \mathcal{N}\left(z \mid H_k m_{k|k-1,i}, S_{k,i}\right), \tag{3.133}$$

$$m_{k,i}(z) = m_{k|k-1,i} + K_{k,i}(z - H_k m_{k|k-1,i}), \tag{3.134}$$

$$P_{k,i} = (I - K_{k,i}H_k)P_{k|k-1,i}, \tag{3.135}$$

$$K_{k,i} = P_{k|k-1,i}H_k S_{k,i}^{-1}, \tag{3.136}$$

$$S_{k,i} = H_k P_{k|k-1,i}H_k^T + M_k R_k M_k^T. \tag{3.137}$$

Gaussian Mixture Reduction Although Gaussian mixture approximation of the PHD recursion yields a closed form solution, the number of mixture components needed to represent the intensity function grows exponentially with time. In a real-world application even after a few iterations of the GM-PHD recursion, the number of components would be to large for a limited memory, and the algorithm would become computationally intractable. However, most of the components would have negligibly small weights, and some of the components would be close enough, so that they could be well approximated by a single Gaussian. Thus, it is important to incorporate a heuristic for reduction of Gaussian mixture components in any real-world implementation of the GM-PHD filter.

Some of the algorithms for Gaussian mixture reduction can be found in [86–89]. The method proposed in [8] will be used in this report for simplicity. Firstly, all components whose weight is below certain threshold T are truncated and the weights of all remaining components are scaled so that the expected number of targets remains unchanged. Secondly, components that are close enough given a threshold U are merged, where the distance between two components is a Mahalanobis distance. Finally, if the number of components is still too large, then only a desired number with highest weights is kept (and weights of remaining components are appropriately scaled).

State Extraction After the posterior intensity function $v_k(x)$ is calculated, the state extraction procedure is needed in order to obtain individual targets. Since the peaks of intensity function represent locations with high concentration of objects and $\hat{N} = \int v(x)\,\mathrm{d}x$ is the expected number of objects, it is intuitive to pick $\hat{N}$ highest peaks of the intensity function as estimated targets. Moreover, since the intensity function in GM-PHD filter is a Gaussian mixture, and because Gaussian components are well separated due to merging procedure, the peaks of the intensity are approximately means of the individual Gaussian components. However, the height of peak depends not only on the weight of the Gaussian, but also on its covariance [8], and it is not desirable to pick a low-weight Gaussian as a target estimate.

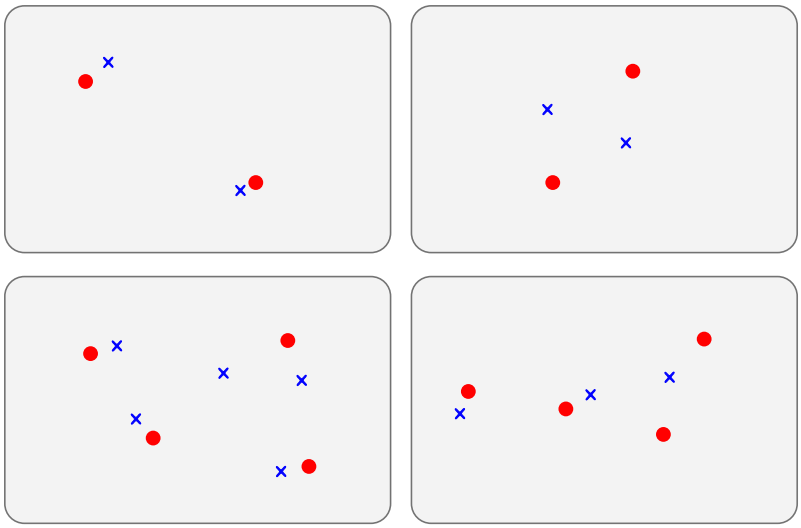


Figure 3.5: Illustration of four frames in MTT where red circles represent ground truth objects and blue crosses are estimated targets.

Hence, to obtain the target estimates of the GM-PHD filter, the $\hat{N}$ Gaussian components with highest weight, keeping in mind that $w > 1$ implies that more than one target exist near that peak.

3.6 Multiple Object Tracking Evaluation

Estimation error or miss-distance is an important concept in any estimation problem. In classical single-target filtering, the system state is a random variable on some Euclidean space, hence the estimation error can be defined as a distance from the estimated state to the true system state (ground truth). The main difference between multi-target and single-target filtering is that the multi-target state is a unordered set of individual target states. Thus, it is not easy to extend the single-target miss-distance concept to the multi-target concept. One could naively just stack target states in a single vector and try to use Euclidean distance. However, since RFSs are unordered, there are many different ways arrange target states into a vector resulting in a different distance for each arrangement. Furthermore, even if it was known how to associate estimated targets with ground truth objects, it is not clear how to penalize error in estimated target number.

3.6.1 *Optimal Sub-Pattern Assignemnt*

In [90] Schuhmacher et al. proposed the optimal sub-pattern assignment (OSPA) metric for evaluating the multi-target filtering performance. Let $d_c(x, y) \triangleq \min(c, d(x, y))$, where $d(x, y)$ is the Euclidean distance between $x, y \in \mathcal{X}$, and $c > 0$ is the cut-off distance. Then the OSPA distance between sets $X = \{x_1, \ldots, x_m\}$ and $Y = \{y_1, \ldots, y_n\}$ is defined as follows

$$\bar{d}_{p,c}(X, Y) \triangleq \left(\frac{1}{n} \left(\min_{\pi \in \Pi_n} \sum_{i=1}^{m} d_c(x_i, y_{\pi(i)})^p + c^p(n - m) \right) \right)^{\frac{1}{p}}, \tag{3.138}$$

if $m \leq n$ and $\bar{d}_{p,c}(X,Y) \triangleq \bar{d}_{p,c}(Y,X)$ otherwise, where Π_n is the set of permutations on $\{1,\ldots,n\}$. The proof that (3.138) is a well-defined metric can be found in [90]. The parameter c, as said above, is the cut-off distance, which means that estimated target and a true object can not be associated if their distance in greater than c. The parameter $p \in [1,\infty\rangle$ is the order of the metric which affects the sensitivity of the metric to outliers in the sense that as p increases the outliers are more penalized.

3.6.2 *Clear MOT Metrics*

The main disadvantage of the MTT evaluation method presented in section 3.6.1 is that it does not take into account the consistency over time of the tracker. To overcome this issue, Bernardin and Stiefelhagen proposed the clear MOT (CMOT) metrics in [91]. The CMOT provides two measures: multiple object tracking precision (MOTP), which measures the error in location of the tracker, and multiple object tracking accuracy (MOTA) which measures the trackers ability to assign correct labels to its object hypotheses.

The MOTA measures incorporates the number of false positives (FPs), missed objects or false negatives (FNs) and the number of mismatches or identity switches (IDS's). The number of FPs in a single frame is obtained by counting all of the tracker's hypotheses that cannot be matched to any of the ground truth (GT) objects. Similarly, the number of FNs in obtained by counting all GT objects that are not matched to any of the tracker's hypotheses at current frame. The IDS's number counts how many of the GT objects are matched to the hypothesis with label inconsistent with the one from the previous frame.

In order to calculate the MOTP and MOTA scores, it is necessary to map the track hypotheses to GT objects in each frame. Firstly, the threshold T is used to check if correspondences between GT objects and track hypotheses are valid [91]. If the distance between the true object and the track hypothesis is greater than the threshold, the correspondence between this object and track hypothesis is considered as not valid. Furthermore, all mappings from frame $k-1$ that are still valid in frame k are preserved. The rest of objects and tracks are paired in such a way that minimizes the total distance, which can be achieved with the Hungarian algorithm [51]. Now, denote with FP_k the count of all false positives in frame k, with FN_k the number of all false negatives in frame k and, finally, denote with IDs_k the number of identity switches in frame k.

$$\mathrm{MOTP} = \frac{\sum_{i,k} d_{i,k}}{\sum_k c_k}, \tag{3.139}$$

$$\mathrm{MOTA} = 1 - \frac{\sum_k (\mathrm{IDs}_k + \mathrm{FP}_k + \mathrm{FN}_k)}{\sum_k \mathrm{GT}_k}, \tag{3.140}$$

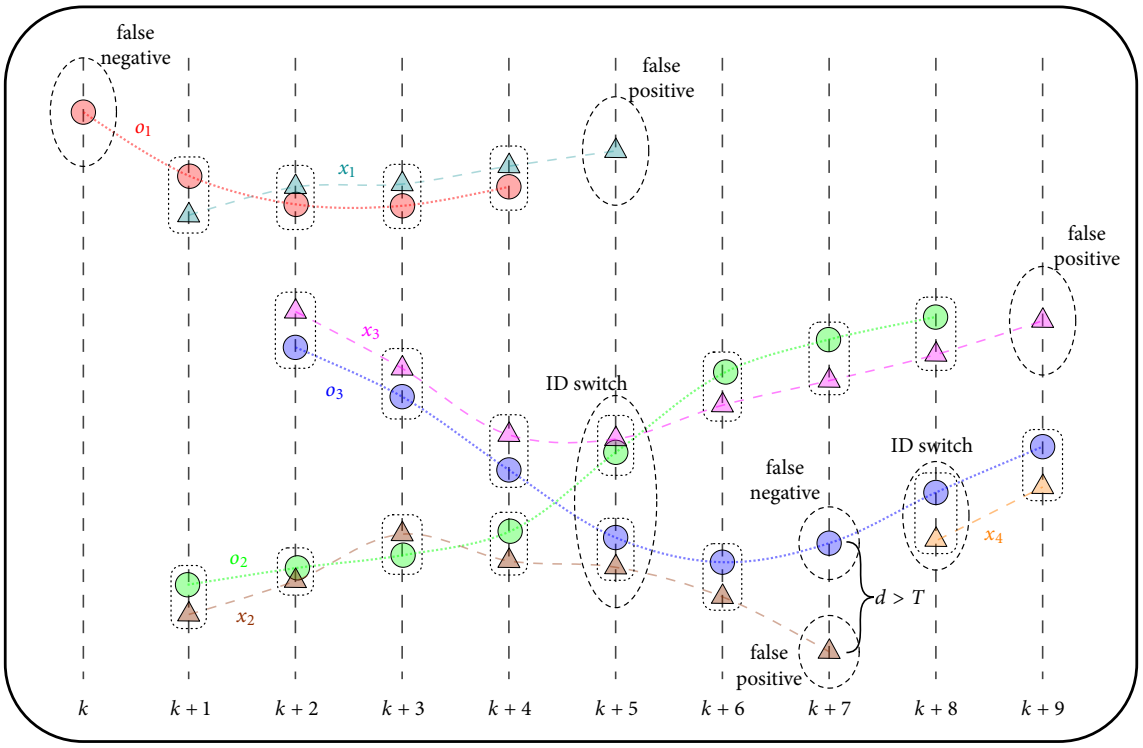


Figure 3.6: Illustration of the CMOT evaluation. Figure shows 10 consecutive time frames of tracking. True locations of objects o_1, o_2 and o_3 are marked with red, green and blue circles. Estimates of object positions are indicated by triangles, where the color of the triangle represents the unique label of the estimated object. Matching between true objects and tracker's hypotheses are shown with doted rectangles. There are a few examples of false positives, false negatives and identity switches on this illustration.

3.7 Summary

In this chapter we have presented the state-of-the-art state estimation and multi-target tracking methods. First, we introduced the Bayesian estimation paradigm. Then we considered some special cases of Bayesian estimators. Under linear Gaussian assumption, the Bayesian recursion admits a closed-form solution which is in the literature known as a Kalman filter. Although Kalman filter is an linear recursive estimator, there are several extensions for nonlinear systems. Extended Kalman filter uses first-order approximation of the state transition function to propagate the uncertainty of the filter, while the unscented Kalman filter utilizes the unscented transform to propagate both the state of an nonlinear system and its uncertainty. The second part of this chapter introduces the problem of multi-target tracking. Three different types of multi-target tracking methods are presented. Finally, optimal sub-pattern assignment and clear MOT metrics are introduced.

Chapter 4

Probabilistic Tracking over Deep Embeddings

Pedestrian multi target tracking methods require detections as inputs and deep convolutional models are particularly suitable for the task. Pre-training deep models on large datasets is shown to have a great regularization effect. Datasets like ImageNet [92] and COCO [93] hold great generalization potential which is available through pre-training on such large collections of annotated data. However, fine-tuning of a multi class object detector to detect pedestrians is not a straightforward task. Limitations in model vertical receptive field, noise in bounding box annotations, and annotation errors are common issues [94]. Fine-tuning on homogeneous video sequences incurs high overfitting risk, thus diversity in training data should be targeted to improve generalization. Given that, fine-tuning a multiclass object detector for the task of pedestrian detection is possible by training on a dataset like CityPersons [95]. This dataset contains diversity on multiple axes, such as person identity, clothing, pose, occlusion level etc.

Correspondence embeddings can be useful for a target association problem such as pedestrian tracking, since they are trained to measure similarity between images. The pioneer of approaches for deep metric learning used siamese networks [96], while triplet networks are considered as an improvement [97]. This is due to a slight, but impactful modification of the loss, which ensures better alignment between the similarity in the embedded space and the likeliness of correspondence. A body of work analyses and improves triplet loss functions. Convergence issues of metric learning using triplet loss are alleviated using N-pair loss, which compares a positive example to $N-1$ negatives [98]. Rather than focusing on pairwise distances in metric space, angular loss [99] minimizes the angle at the negative point of the triplet. This has a positive effect on quality of learning, since angles are insensitive to changes in scale. Our approach follows previous work which utilizes segmentation masks to make the appearance embedding less sensitive to occlusions and changes in the background [100].

A good overview of the current state-of-the-art of the MTT algorithms can be found in [1], where authors consider three different approaches to the MTT problem: (i) probabilistic data association, (ii) multiple hypothesis tracking, and (iii) random finite set approach. In probabilistic data association (PDA) [2–4] tracking methods, the measurement association uncertainty is untangled by soft assignment. The first such methods for single and multiple target tracking were the PDA filter [2] and the joint probabilistic data association filter (JPDAF) [3], respectively. However, both approaches assume known and constant target number and need some heuristics for track initialisation and termination. The integrated probabilistic data association filter (IPDAF) [52] and joint integrated probabilistic data association filter (JIPDAF) [4] alleviate this issue by estimating the targets existence probability

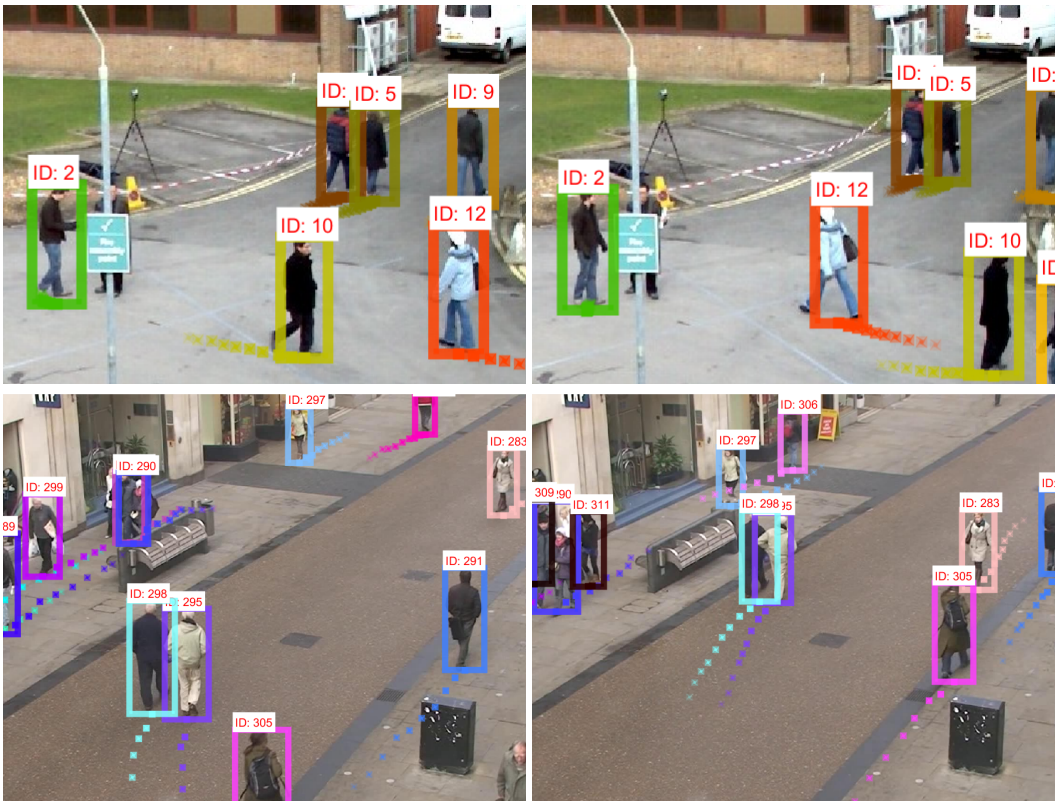


Figure 4.1: Pedestrian tracking on 3DMOT2015 sequences, PETS09-S2L1 (up) frame 125 (left) and frame 138 (right), AVG-TownCentre (down) frame 341 (left) and frame 351 (right).

together with its states, thus providing a natural method for automatic track initialisation and termination. Unlike probabilistic data association (PDA) methods, multiple hypothesis tracking (MHT) algorithms [6, 54] generate hypotheses for different associations and the decision about which of the hypotheses is correct is postponed until new data is collected. Somewhat more recent approaches are based on the random finite set (RFS) paradigm [101]. Based on the RFS theory, the closed-form first moment approximation of the RFS filter, the Gaussian mixture probability hypothesis density (GM-PHD) filter was presented in [8], and since then other novel approaches have been proposed [11, 12, 84].

In this chapter we present a pedestrian tracking-by-detection approach based on a deep learning detector combined with the JIPDA and an appearance-based tracker using deep correspondence embeddings. A convolutional neural network detector was pretrained on the COCO dataset for accurate pedestrian detection and serves as the input for the JIPDA based tracking algorithm where the state consists only of pedestrian kinematic cues (positions and velocities). The proposed pedestrian tracker with kinematic cues currently ranks first on the 3DMOT2015 online benchmark [102] that contains sequences with a static camera (cf. figure 4.1). In order to enable pedestrian tracking in sequences containing camera motion, under the assumption that camera ego-motion is not available, kinematic parameters in 3D need to be exchanged for appearance cues based on a deep correspondence metric in the image space. We therefore learn a correspondence embedding and leverage it for association across video frames using the global nearest neighbor approach (GNN). In the end, we compare GNN tracking of correspondence embeddings with the JIPDA tracker based on kinematic cues (position and velocity).

This chapter is organized as follows. Firstly, the section 4.1 gives a brief overview of the literature related to visual tracking. Then, the public MOT Challenge benchmark that was used to evaluate the proposed method is outlined in section 4.2. Section 4.3 describes models we used to detect pedestrians as well as calculate correspondence embeddings. The probabilistic data association method used for tracking pedestrians is presented in section 4.4. Results on the online MOT benchmark as well as validation experiments are presented in section 4.5. Finally, we give a summary of our accomplishments and findings in section 4.6.

4.1 Related Work

Visual pedestrian tracking can be implemented by consolidating an object detector, appearance based association metric, and a suitable MTT approach. In [103] authors track multiple pedestrians using a Rao-Blackwellized particle filter [104] with track management based on detection association likelihoods. Therein, the authors augment the state vector of tracked objects by an appearance based deep person re-identification vector [105] and compute data association probability by multiplying conditionally independent position and appearance association likelihoods. The appearance association likelihood computed as the *softmin* function of the norm of the appearance vector differences. The authors report that adding appearance information reduced the number of identity switches and increased slightly the overall tracking score; however, tracking using just the appearance, without position information, showed to perform quite poorly.

In [106] probabilistic models were incorporated into a track-by-detection approach using prior knowledge of a static scene, describing pedestrian state using position, height and width in world coordinates. Such approach lacks information on pedestrian appearance to correctly handle interactions between pedestrians in crowded scenes.

In [107] authors utilize the probability hypothesis density (PHD) filter to track multiple pedestrians. Since, PHD does not provide the track labels but only the locations of tracked objects, additional data association step must be performed on the outputs of the PHD filter. Hence, authors propose to use the min-cost flow network to solve the data association problem. Furthermore, they reformulate the PHD recursion in terms of single-target track hypotheses.

Song *et al.* propose a multi target tracking tracking method based on the Gaussian mixture implementation of PHD filter in [108]. They expanded the GM-PHD filter with the hierarchical data association module. The hierarchical data association uses the Hungarian algorithm [51] to solve both the detection-to-track and track-to-track assignment problems. Authors claim that due to hierarchical data association it is possible to recover tracks lost because of missed detections. Additionally, tracking merging and occlusion group energy minimization techniques are used together to tackle the occlusions.

Yang *et al.* proposed the multi-person localization and tracking framework in [109]. They estimate the 2D human poses for each person and then apply intrinsic camera parameters to obtain the 3D location of a person. 3D person locations makes it easier to utilize the Kalman filter (KF) to model human motion. Additionally, authors also utilize the appearance

similarity together with the trajectory to improve data association. In order to evaluate their method, authors annotated their own multi-person panoramic localization and tracking dataset and also use KITTI and MOT Challenge datasets.

The MOANA approach [110] introduces an adaptive model capable of encoding a long-term appearance change. Furthermore, they incorporate that appearance model together with the geometry information in the re-identification model to diminish the problem of occlusion and association ambiguity of nearby objects. The authors tested their method on the MOT Challenge benchmark [102] on both the 2D and 3D datasets of MOT15 challenge and achieved remarkable results.

In [111] Stadler and Beyerer tackles the problem of heavy occlusions which arise in pedestrian tracking in crowds. They propose to cluster tracks based on their overlap and adding cluster states to each cluster that depend on the number of target and detections in cluster. The past track data is used to correct the wrong detection-to-target associations after the targets are re-detected. Finally, authors combined the tracking-by-detection framework together with regression-based tracking in order to improve the tracking performance in crowded scenes even more. Their method achieved promising results on MOT16, MOT17 and MOT20 datasets of MOT Challenge benchmark [102].

Wang *et al.* propose a deep learning based approach to multiple object tracking [112]. Similar to many recent deep learning MTT methods, they use the joint MTT approach, where the object detection and data association modules are optimized simultaneously. Moreover, they apply graph neural network model to exploit its ability to learn the relations between variable sized sets of objects and detection in both the spatial and temporal domains. Authors compare their approach with other state-of-the-art pedestrian trackers on various MOT Challenge [102] datasets.

In [113] the graph convolutional neural network was utilized to achieve multi target tracking. With the combined the graph and convolutional neural network architecture, it is possible to extract the deep features as well as to model the interaction among the objects and among their features. In addition, authors applied the Sinkhorn algorithm to enforce the MTT constraints during the training of their model. Authors also claim that their method significantly improves performance compared to other graph neural network due to the use of geometric features. The performance of the approach is measured on the MOT15, MO16, MOT17 and MOT20 datasets of MOT Challenge benchmark.

4.2 MOT Challenge Benchmark

The MOT Challenge benchmark[1] [102] provides a set of different multi target tracking datasets. Datasets are split into two types of sequences. One group contain ground truth (GT) data and is suitable for training and parameter tuning, while the other group is intended for evaluation and comparison of different methods. To achieve that, the GT data is not publicly available, but the benchmark provides the centralized evaluation method to ensure the fair comparison.

One of the datasets available on MOT Challenge is the MOT15 dataset [102] which has

[1] https://motchallenge.net/

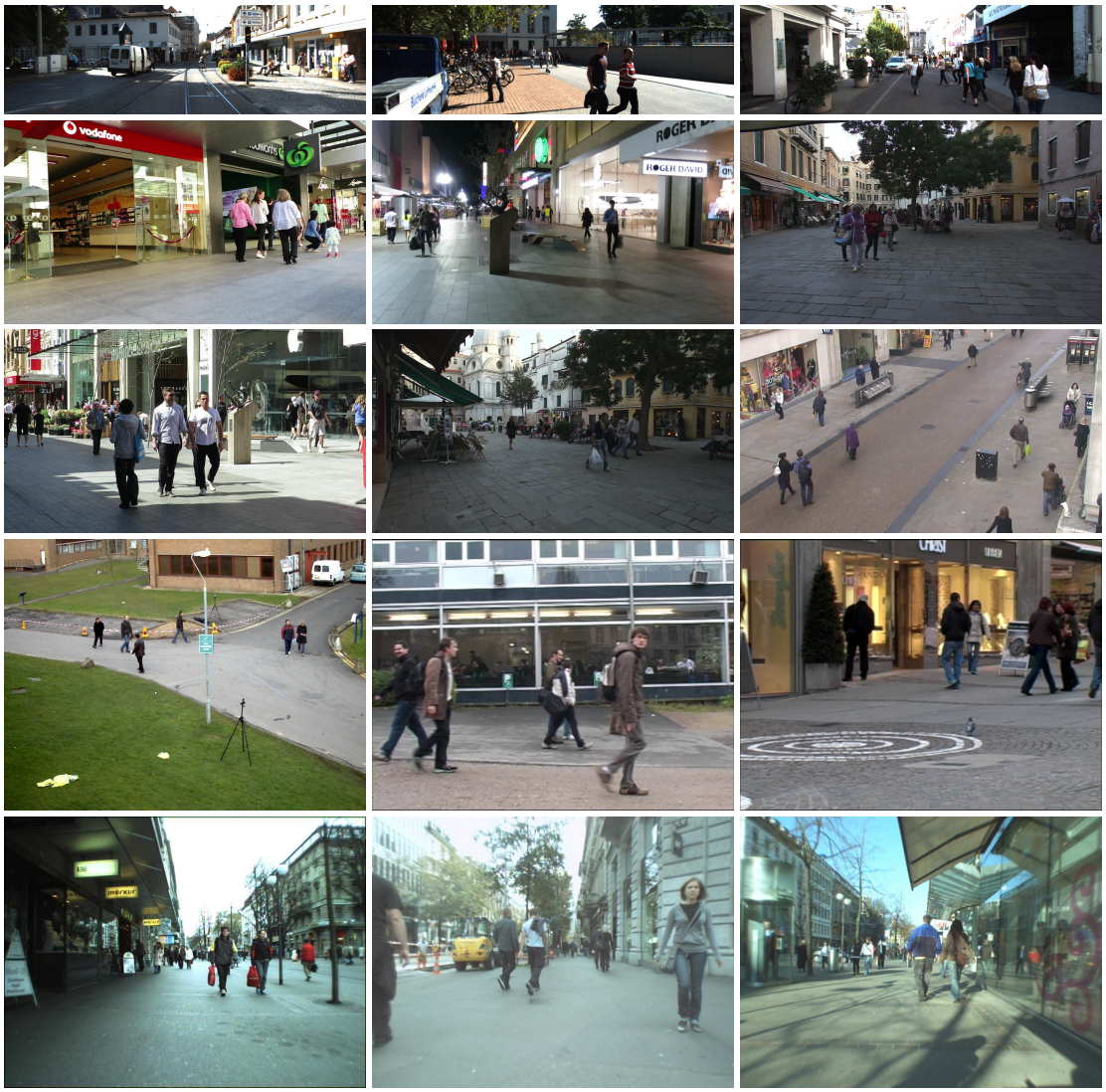


Figure 4.2: Some of the images from MOT15 sequences of the MOT Challenge multi target tracking benchmark.

both 2D and 3D versions. 2D-MOT15 dataset contains 11 training and 11 testing sequences. Some of the images from this dataset are shown on figure 4.2. The dataset contains sequences recorded with different types of cameras, from different angles and also contains some sequences with moving camera. The 3D-MOT15 dataset contains a few of the sequences of 2D-MOT15 where the camera is static, and furthermore, the intrinsic calibration parameters of the camera as well as the parameters for the transformation to the world coordinates are given. The MOT16 dataset [114] aims to improve some of the issues in MOT15 such as imbalance in the crowd density between training and test sequences and the quality of the detections provided with the dataset. Moreover, MOT16 introduces sequences with higher crowd density, different weather conditions and with different perspective to make the tracking more challenging.

4.3 Detection and Appearance Representation

In order to detect pedestrians, the Mask region based convolutional neural network (R-CNN) algorithm trained on a suitable blend of public datasets is applied. The obtained bounding boxes were cropped and scaled, masked with segmentation masks and then processed with the separate deep model trained with a metric loss. This resulted in correspondence embeddings which were used as descriptors in appearance-only tracking. The details are described in the rest of this section.

4.3.1 *Pedestrian Detection*

Mask R-CNN [115] is an extension of the Faster R-CNN [116] object detector. It consists of two stages: (i) finding regions of interest (RoIs) using region proposal network (RPN) and (ii) classification of the proposed RoIs and bounding box regression. Mask R-CNN enhances the second stage by predicting segmentation masks of RoIs provided by RPN. By utilizing the *RoIAlign* operation and attaining better representations through learning segmentation masks, Mask R-CNN surpasses Faster R-CNN on the task of object detection. We adapted the multi-class Mask R-CNN for pedestrian detection. The architecture of the Mask R-CNN is shown on figure 4.3.

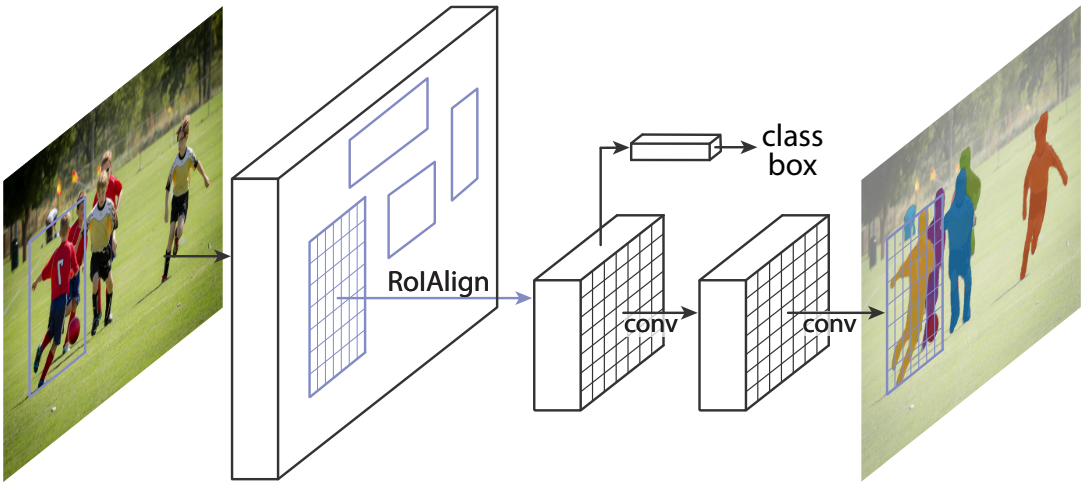


Figure 4.3: Architecture of mask R-CNN object detector. Figure taken from [115].

The most suitable transfer-learning strategy is chosen by performing validation experiments with a Mask R-CNN detector trained on different datasets. It turned out that the fine-tuning from COCO to CityPersons was the most appropriate course of action as shown in details in section 4.5. We believe this can be explained as follows. Firstly, CityPersons includes annotations with fixed aspect ratio (*BB-full*) which are suitable to train occlusion invariant bounding box regression. Secondly, we noticed that COCO people are much more diverse than MOT Challenge pedestrians due to numerous other contexts such as riding, driving, sitting down etc. Furthermore, CityPersons inherits ground truth pixel-level masks from the Cityscapes dataset [117]. Presence of ground truth pixel level masks is suitable for fine-tuning Mask R-CNNs mask head.

We adapted the pre-trained multi class Mask R-CNN [115] for pedestrian detection in two steps. Firstly, we adapted the Mask R-CNN classification, bounding box regression and mask prediction heads to have two possible outputs: background and pedestrian. We sliced the weights of the last layer of the classification head in order to leave only the logits for the *background* and *pedestrian* classes which we initialized with weights of the corresponding COCO classes. Secondly, we fine-tuned the resulting model with ground truth bounding boxes and segmentation masks from CityPersons.

4.3.2 *Deep Correspondence Embedding*

We represented pedestrian appearance with a metric embedding provided by a deep correspondence model. Appearance of each pedestrian is represented by high-dimensional embeddings in metric space. Selection of the correspondence model is not straightforward. We started form ResNet-18 [118] classification architecture which consists of four residual blocks, from RB1 to RB4. Features from RB4 are suitable for discriminating between different classes. However, we found that features from earlier blocks are more beneficial for differentiating between different person identities. Therefore, we calculated embeddings from features in the last convolutional layer of RB2. Validation experiments suggest that these features contain more information about person appearance than features in any other residual block. Furthermore, the last two blocks hold around 70 % of total ResNet parameters. By getting rid of them, we decreased the susceptibility to overfitting. At the same time, it is possible to initialize the first two blocks with pre-trained weights and profit from regularization induced by ImageNet. We demonstrate the effectiveness of this approach in more detail in section 4.5.

Furthermore, we investigated the possibility of using segmentation masks provided by Mask R-CNN to generate descriptors which are robust to changes in object background and occlusions. We experimented with two approaches for incorporating the segmentation mask M_S into the correspondence embedding. The first approach applies M_S to the input image. The second approach uses M_S to mask the convolutional features. The two approaches are not the same since the latter approach preserves some background influence due to receptive field of the convolutional features. Despite this, applying the mask to ResNet features performed better in experiments. We conjecture that this is due to low resolution of the Mask R-CNN mask resulting in poor accuracy when upsampled to RoI resolution. Note that a segmentation mask can be interpreted as a dense probability map that the corresponding pixel is foreground. Therefore, one can suppress the background by elementwise multiplication with the segmentation mask.

As mentioned before, we adapted the ImageNet pre-trained architecture by taking only the first two residual blocks. The features of the last residual block were passed to a 1×1 convolutional layer and masked using the output of Mask R-CNNs segmentation head. Finally, the correspondence embedding was produced by global average pooling.

The model was trained using angular loss [119]. We extended the angular loss by adding the margin term. For a given reference embedding r, a corresponding embedding of the same identity p and a negative embedding q, we calculated the angular loss (4.1), where m

is the margin hyperparameter and $c = \frac{r+p}{2}$:

$$L_{\text{ang}} = \max(0, m + \|r - p\| - 4\tan(\alpha)^2 \|q - c\|). \tag{4.1}$$

Gradients of the angular loss push the negative example away from the center of p and r examples in the $q - c$ direction. This also minimizes $r^T q$ and $p^T q$ (r, p and q are unit vectors).

4.3.3 *Details of Training Correspondence Embedding*

We trained the correspondence model on MOT2016 [114]. We refrain from training on 2D MOT 2015 since it does not include precise ground truth data regarding occlusion level. During training, we removed all training samples with occlusion level greater than 50 %. We incorporated the following method for generating positive and negative samples. We generated positive examples by taking random detections less than 5 frames away from the reference example frame. We generated negative examples by taking random identity from the same sequence. We sampled random easy negatives as bounding boxes which do not intersect any ground truth bounding boxes. This made the correspondence model more robust to pedestrian detector's false negative outputs. We chose the following sequences to serve as validation data: `MOT16-02`, `MOT16-04`, `MOT16-05`. The validation data was used for early stopping and tuning of hyperparameters. The output embedding vectors had 64 dimensions. We used the Adam [120] optimizer with fixed learning rate of 10^{-4}. Weight decay was set to 10^{-4} for all parameters and the model was trained for 10 epochs. During training and testing, we did not use whole images. Instead, we cropped the detection bounding boxes and resized them to the fixed resolution 224×96.

4.4 Pedestrian Tracking Method

Out of many available state-of-the-art MTT methods, we chose to utilize the JIPDAF described in section 3.3.3. The state of each pedestrian can be described by the coordinates of the centroid and the width and height of the bounding box. We use the constant velocity Kalman filter to model the motion of pedestrians. Denote with $x = \begin{bmatrix} p_x & \dot{p}_x & p_y & \dot{p}_y \end{bmatrix}^T$ the state of the pedestrian, where p_x and p_y denote the image coordinates of the centroid of the bounding box. The motion model is then given by

$$x_k = F\, x_{k-1} + w_{k-1}, \tag{4.2}$$

$$y_k = H x_k + v_k, \tag{4.3}$$

where w_{k-1} is the zero-mean Gaussian process noise with the covariance matrix $Q = \text{diag}[Q_1, Q_1]$ and v_k is the zero-mean Gaussian measurement noise with the covariance matrix R. $F = \text{diag}[F_1, F_1]$ is the state transition matrix. The motion model matrices are as follows

$$F_1 = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix} \qquad Q_1 = \sigma_q^2 \begin{bmatrix} \Delta T^3/3 & \Delta T^2/2 \\ \Delta T^2/2 & \Delta T \end{bmatrix}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad R = \sigma_r^2 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

The width and height of bounding box are not included in the Kalman filter because it is possible that they could take negative values after the update step of the filter. Instead, the updated heights and widths are calculated as weighted sums given prior bounding box, detected bounding boxes and obtained association probabilities. In case of the 3D-MOT Challenge, the ground plane coordinates are used in KF instead of the image coordinates of the bounding box centroid in order to improve performance. The strategy for initialisation is based on the probability that the detection does not originate from any of the existing tracks, which is given by $p_{\text{new}} = 1 - \sum_j \beta_{i,j}$. Hence, the new track is created for each detection for which the probability p_{new} exceeds certain threshold. The initial target's state is set to $\begin{bmatrix} p_x & 0 & p_y & 0 \end{bmatrix}^T$, where p_x and p_y are coordinates of the centroid of the detection wich is used to initiate this track.

We also consider the appearance-only tracking, where the deep embeddings of the deep neural network are used to describe the appearance of pedestrians. In this case, the simple global nearest neighbour (GNN) method is used to track the pedestrians. Similarity for all pairs of between tracked objects and detections are calculated based on deep embeddings, and then the Hungarian method [51] is utilized to find the optimal matching. The bounding boxes of the targets are then replaced with the bounding boxes of the matched detections without any filtering.

4.5 Experimental Results

The experimental evaluation of the proposed method is given in this section. Firstly, section 4.5.1 provides the validation of the detection and correspondence deep models introduced in section 4.3. Section 4.5.2 then shows the evaluation of the proposed pedestrian tracking method on the MOT Challenge benchmark.

4.5.1 *Validating the Detection and Correspondence Model*

For validation experiments we studied the impact of training Mask R-CNN on different combinations of training datasets and we carefully analyzed the design possibilities to find the most suitable correspondence embedding. Here, we describe several validation studies and comment on the results.

Fine Tuning Mask R-CNN After having little success in transfer learning from COCO to MOT in preliminary experiments, validation experiments were performed by training just on CityPersons, just on COCO, and on both datasets, achieving average precision of 45.1, 53.3, and 57.0, respectively table 4.1. Considering the size of dataset for pre-training, the results in the first two rows are not surprising. Fine-tuning on CityPersons is suitable to distinguish between pedestrians and other people. Also, bounding boxes generated by Mask R-CNN trained using *BB_full* annotations from CityPersons are a better fit for detection of MOT pedestrians. All detection experiments feature Mask R-CNN detector based on ResNet-50 FPN.

Table 4.1: Strategies for training pedestrian detectors. Checked columns denote that the model is trained on the given dataset. The *AP* column denotes detection average precision on MOT2016 train. All experiments use Mask R-CNN with a ResNet-50 FPN backbone.

COCO	CityPersons	AP
	✓	45.1
✓		53.3
✓	✓	57.0

Using segmentation maps The impact of using segmentation masks is shown in table 4.2, where *IDs* denote the number of identity switches, while *IDs†* shows evaluation on ground truth bounding boxes. There are more *IDs* when evaluating on ground truth because no fragmentations are present. The models were trained on the MOT2016 train dataset, while evaluation was performed using an appearance based GNN on 2DMOT2015 train. We showed that segmentation masks generated by Mask R-CNN benefit the correspondence model by alleviating impacts of background and occlusions. First, a baseline correspondence model which did not use segmentation masks was trained. Secondly, we trained two correspondence models improved by segmentation masks. The first model masks the input image. The second model masks the final feature map before the global average pooling operation. We witnessed an improvement in tracking with the latter approach.

Table 4.2: Applying segmentation masks at image vs feature level.

masked tensor	IDs†	IDs	MOTA
–	507	404	53.6
input image	420	337	53.8
final conv features	**328**	**291**	**53.9**

Residual Blocks Our final model uses only the first two residual blocks of an ImageNet pre-trained ResNet-18. This design choice is supported by experiments shown in table 4.3. In each experiment, we used one additional residual block. We trained the model on MOT2016 and evaluate tracking using a position-agnostic GNN approach. The results complement our initial hypothesis that for describing appearance, abstract features like ones in the output of a full ResNet model may not be beneficial. In figure 4.4 we can see how the appearance similarity is distributed throughout the frames by looking at the similarity score of appearance vectors of the same object separated in time. The results show that even for a separation of five time steps the similarity of most appearance vector is preserved, with clear separation from the other objects.

4.5.2 *Pedestrian Tracking Evaluation*

To track the states of individual targets we used the constant velocity motion model with the state vector of the targets given by eqs. (4.2) and (4.3). The process and measurement

Table 4.3: Validation of the model architecture. *RB** designates a resblock, while *#params* shows the total parameter count.

RB1	RB2	RB3	RB4	#params	IDs†	IDs
✓				161.7K	393	458
✓	✓			691.3K	**328**	**291**
✓	✓	✓		2.8M	416	398
✓	✓	✓	✓	11.2M	1271	687

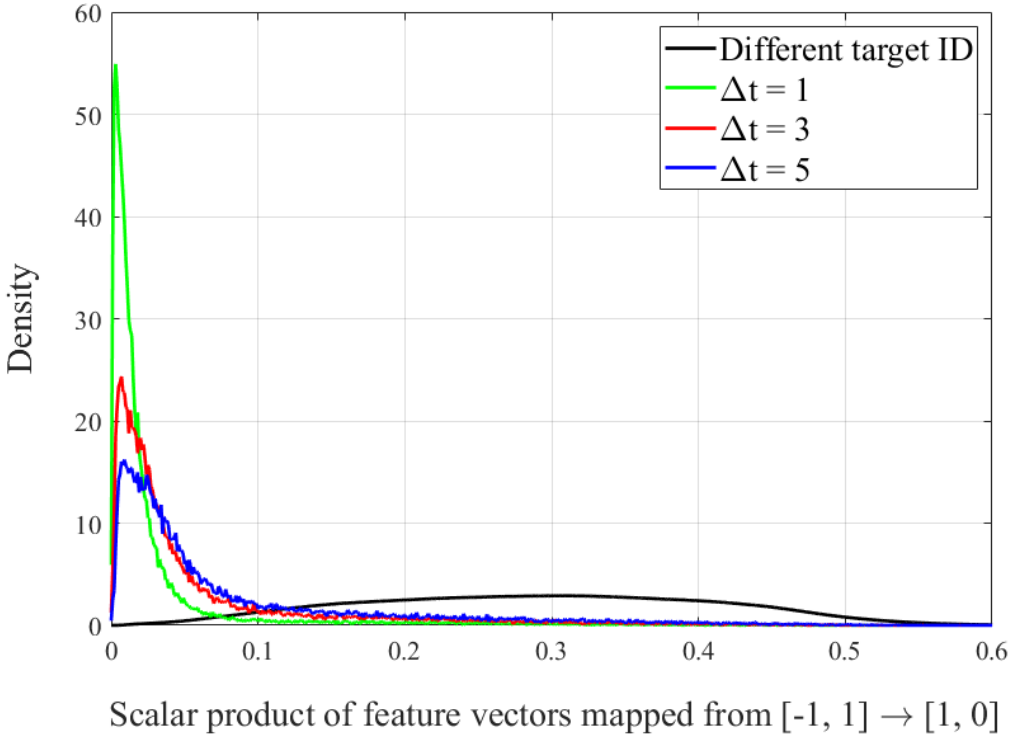


Figure 4.4: Distribution of scalar products of the deep embeddings mapped to interval $[0,1]$. Black line is the distribution of feature vector scalar products which do not belong to the same object. Red, green and blue lines show distributions of feature vector scalar products of the same object at consequent time steps. Feature vectors were evaluated on ground truth bounding boxes.

noise deviations were set to $\sigma_q = 0.836\,\mathrm{m\,s^{-2}}$ and $\sigma_r = 0.141\,\mathrm{m}$. Target survival and detection probabilities were $P_S = 0.999$ and $P_D = 0.990$. Measurement gating probability was $P_G = 0.990$. False alarm rate was set to $\lambda = 15V^{-1}$, where V is the surveillance area. [2]. New targets were initialized for the measurements whose a posteriori association probability of not being associated to any of the existing targets was

$$1 - \sum_j \beta_{i,j} > 0.7.$$

[2] In the 3DMOT2015 the surveillance area was estimated by projecting the image frame to the real-world coordinates and then calculating its area

Table 4.4: *MOTChallenge* benchmark results for *3DMOT2015* category, proposed method is *MCN_JIPDA*, ↑ denotes that higher is better and ↓ that lower is better, * denotes that the work is still unpublished. Best score for each metric is in boldface.

Tracker	MOTA↑	MOTP↑	FP↓	FN↓	IDs↓
MCN_JIPDA	**55.9**	**64.0**	2,910	**4,011**	486
MOANA [110]	52.7	56.3	2,226	5,551	**167**
DBN [106]	51.1	61.0	2,077	5,746	380
GPDBN[121]	49.8	62.2	**1,813**	6,300	311
GustavHX*	42.5	56.2	2,735	6,623	302

Initial existence probability for new targets was set to $w_{\text{init}} = 0.65$ and the target was confirmed when its existence probability exceeded threshold $w_{\text{confirm}} = 0.85$. Since nothing could be inferred about the new target's velocity from only one measurement, it was assumed to be zero, but the initial covariance matrix of the target was inflated so that the state of the target converges to the actual value when the new measurements arrived. Targets were terminated when their existence probability fell below the threshold $w_{\text{delete}} = 0.003$. To improve tracking performance we discarded all detections with confidence score below the threshold $t = 95\,\%$.

The tracking results are shown in table 4.4, where we can see that the proposed kinematic cues based JIPDAF with the Mask R-CNN detector ranked first on the 3DMOT2015 dataset that contains static camera sequences. The table shows results for the test sequence, while on the train sequences the tracker obtained MOTA 80.6 and MOTP 69.1. Our method did produce a higher number of identity switches compared to MOANA, since we did not use appearance cues and our detector has higher recall than public detections. The tracking performance could be further improved by using interacting multiple model [122] instead of a constant velocity Kalman filter and by taking unresolved measurements into account as proposed in [123].

In table 4.5, which compares the kinematic cues based JIPDAF with deep detections to the deep correspondence metric based GNN, we can see that both trackers show roughly the same performance for static camera sequences and tracking in the image space, while the kinematic based JIPDAF is not appropriate for moving camera with unknown motion. Augmenting the state space with deep correspondence embeddings directly within a soft data association approach such as JIPDAF did not result in increased tracking accuracy in our experiments. It remains an interesting venue of future work to investigate the correspondence embeddings space geometry and utilize the findings in soft data association approaches.

4.5.3 *Comparison With Public Detections*

In table 4.6 we compare tracking performance of the JIPDAF with our Mask R-CNN detections to the tracking with public benchmark detections. JIPDAF parameters used in the case of the Mask R-CNN detections are the same as in section section 4.5.2. In the case of benchmark detections we find that the detection confidence threshold $t = 47\,\%$ and the

Table 4.5: Comparison of a kinematic based JIPDAF and appearance based GNN on 2DMOT2015 train sequences.

Cam	Sequence	JIPDA	Appearance GNN
Static	ADL-Rundle-6	58.4	58.4
	KITTI-17	58.3	56,1
	PETS09-S2L1	79.8	78.8
	TUD-Campus	78.3	79.4
	TUD-Stadtmitte	81.0	81.6
	Venice-2	46.0	47.1
Moving	ADL-Rundle-8	–	49.5
	ETH-Bahnhof	–	29.4
	ETH-Pedcross2	–	58.0
	ETH-Sunnyday	–	62.8
	KITTI-13	–	40.8
	Total	–	53.8

clutter density $\lambda = 5\ V^{-1}$ are suitable. Other parameters are same as when Mask R-CNN detections were used.

Table 4.6: Comparison of the JIPDAF performance with public and with ours detections on 3DMOT2015 train sequences.

Detections	MOTA↑	MOTP↑	FP↓	FN↓	IDs↓
Mask R-CNN	**80.6**	69.1	**594**	**426**	**70**
Public	60.7	**74.7**	609	1,106	145

4.6 Summary

In this chapter an online pedestrian tracking method based on JIPDAF and deep models for pedestrian detection and correspondence embedding was proposed. It was demonstrated how a COCO pre-trained Mask R-CNN can be adapted for accurate pedestrian detection. Furthermore,segmentation masks were incorporated to improve the correspondence model embeddings. Correspondence embeddings use masked features from the second residual block of ResNet-18 in order to focus on low-level foreground appearance and reduce the parameter count. The features are pre-trained on ImageNet and fine-tuned with the angular loss. The proposed tracking method achieved best results on the 3DMOT2015 benchmark by combining Mask R-CNN detector and JIPDAF. Submitted tracking results achieved MOTA score of 55.9 and ranked #1. Suitable directions for future work include integrating correspondence embeddings within JIPDAF and investigating the geometry of such soft data association.

Chapter 5

State Estimation on Riemannian Manifolds

This chapter scrutinizes the problem of state estimation on smooth, possibly curved spaces called Riemannian manifolds (RMs) which were introduced in chapter 2. RMs often appear naturally in robotics, for example when dealing with rotations and poses, which can be captured with Lie groups. Due to the non-linearity of the underlying geometry, it is not convenient to use classical estimation methods such as the KF on RMs. The reason for this is that the KF uses the addition and subtraction of vectors to perform the prediction and update. However, adding or subtracting two points of a general RM rarely results in a valid point belonging to manifold. Although this issue could be solved by simply projecting resulting point back to the manifold after each prediction and update step, this would increase the estimation errors and possibly even result in divergence of the filter. A better approach is to use the filtering method that takes into account the geometry of the underlying space. Another possibility is to use particle filtering on Riemannian manifold [124]. In this case, the distribution on the manifold is approximated by a set of points lying on the manifold and corresponding set of weights.

The rest of this chapter is organized as follows. First, the review of the state estimation on manifolds is given in section 5.1. Then, the section 5.2 considers the motion models on Riemannian manifolds that will be used to implement state estimators. The extensions of extended Kalman filter and UKF to RM are given in the section 5.3. The section 5.4 introduces the constant velocity motion model for RM extensions of the KF. The experimental evaluation of proposed filtering methods is given in section 5.5.

5.1 Related Work

Bourmaud *et al.* studied filtering on Lie groups in series of papers. In recent years, Lie groups found immense use in various robotic applications since they can naturally represent orientations and poses. Authors generalized the EKF to processes evolving on Lie groups in [15]. Assuming that the distribution of the state can be represented by concentrated Gaussian distribution on Lie group, they obtained the closed-form expression of their discrete Lie group extended Kalman filter (LG-EKF). In addition, in the same paper it is also shown that the LG-EKF reduces to the ordinary EKF in case that the state space is Euclidean. In [16], authors extended discrete LG-EKF to continuous-discrete LG-EKF. Inspired by the iterative version of the EKF, Bourmaud *et al.* also developed the iterative EKF on Lie groups [125], which they used to solve the problem of global motion estimation.

Continuing on the work of Bourmaud *et al.* on LG-EKF, Ćesić *et al.* developed a

method for moving object tracking using the rigid body motion represented by Lie groups [18]. They generalized the constant velocity model, which is often used in object tracking in Euclidean space, to Lie groups. The state of the object was represented in two different ways. In one as a cross product of the special Euclidean group SE (2) modelling the pose of the rigid body and $\mathbb{R}^3$ modelling translation and angular velocities on $\mathbb{R}^2$. The other model was the product of two special Euclidean groups, the SE (2) × SE (2). Both of these models showed similar performance in simulation, while significantly outperforming basic EKFs with constant velocity and constant turn-rate and velocity models. This method was further extended by Joukov *et al.* in [126], where the LG-EKF was used to estimate the human motion. The complete pose of the human was represented as the kinematic chain, where each joint was modelled by the element of the special orthogonal group SO (2) or SO (3), while the SE (3) was used to model the pose of the base of the body. Moreover, this method can be applied to arbitrary kinematic chains.

In [127] authors present the Bayesian approach to filtering the state of the hidden Markov process evolving on the Stiefel manifold. In [128], the author considers the problem of state estimation on the RMs. In the same paper the author also obtains the conditions that the process on a RM must satisfy to be a Brownian motion. The results are provided in both local coordinate approach as well as in the extrinsic approach. Furthermore, the author obtains some new results concerning un-normalized nonlinear filter. In [124], Snoussi and Mohammad-Djafari proposed the particle filter for stochastic systems evolving on Riemannian manifolds. Zhang *et al.* expanded the feefback particle filter [129] to stochastic processes on RMs and matrix Lie groups in [130]. The main disadvantage of particle filters is the large number of particles needed to approximate the state distribution. Hauberg *et al.* approach this problem by generalizing the Kalman filter to RMs [32]. They develop the version of UKF for Riemannian manifolds. Following the work of Hauberg, Menegaz *et al.* provided the systematization of the UKF theory on RMs and introduced some extensions like augmented Riemannian manifold unscented Kalman filter (RM-UKF) in [34]. In the implementation, they used SMC method to solve the Bayesian recursion. Leonardos *et al.* were dealing with the problem of articulated motion estimation from a monocular image in [33]. Due to the physical constraints of the systems, differential geometry came into play naturally. They developed the generalization of the EKF to systems whose states are constrained to some RM. In addition, they used the generalized constant velocity model in order to obtain the estimation of angular velocities of the system as well. Since the velocity lies in the tangent space of a manifold, this required them to model the motion of a system on the tangent bundle of a RM.

5.2 System Models on Riemannian Manifold

As stated in [131], generalizing from discrete stochastic systems evolving in Euclidean space to RMs is not possible for arbitrary manifolds. In discrete stochastic models, the process noise acts on the system by addition, which is not generalizable to RMs because of the aforementioned reasons. However, in the continuous model, the process noise acts on the tangent space of the manifold, which is vector space, hence the generalization to manifold

is possible. Suppose that the system state x evolves on manifold $\mathcal{M}$. The time evolution of x can then be modelled by following equation

$$\dot{x}(t) = f\big(x(t), u(t), t\big) + w\big(x(t), t\big). \tag{5.1}$$

The function $f : \mathcal{M} \times \mathcal{U} \times \mathbb{R} \to T_x\mathcal{M}$ models the deterministic part of the process, while the function w models the stochastic part, the Brownian motion on manifold. The signal u in (5.1) is the known control input that takes values in the space $\mathcal{U}$. Now the discretized version of (5.1) can be obtained by Euler integration on RM [131], taking into account that $w\big(x(t), t\big)$ is Brownian motion, resulting in

$$x(t + \Delta T) = \mathrm{Exp}_{x(t)}\Big(\Delta T\, f\big(x(t), u(t), t\big) + \sqrt{\Delta T}\, w\left(x(t), t\right)\Big), \tag{5.2}$$

where ΔT is the sampling interval. Instead of (5.2), a simplified discrete model of the following form may be used

$$x_k = \mathrm{Exp}_{f_{k-1}(x_{k-1}, u_{k-1})}\left(w_{k-1}\right). \tag{5.3}$$

In this case, function $f_{k-1} : \mathcal{M} \times \mathcal{U} \to \mathcal{M}$ models the deterministic state transition from step $k-1$ to k and w_{k-1} is the process noise signal defined on the tangent space $T_{x_{k-1}}\mathcal{M}$.

The observation process can be described by equation

$$y_k = \mathrm{Exp}_{h_k(x_k)}\left(v_k\right), \tag{5.4}$$

where $h : \mathcal{M} \to \mathcal{N}$ is the mapping from manifold $\mathcal{M}$ to possibly different manifold $\mathcal{N}$, while v_k is the measurement noise that takes values at tangent space $T_{h_k(x_k)}\mathcal{N}$. Note that the exponential mapping in (5.2) differs from the one in (5.4) because the former is related to manifold $\mathcal{M}$, while the latter is related to manifold $\mathcal{N}$.

5.2.1 *Zero-Order Motion Model on Riemannian Manifold*

The simplest motion model is the zero-order (or constant) model $x_k = f(x_{k-1}) = x_{k-1}$. Motion of the state in this model is completely described by the process noise. The observation process in constant model is usually also given by identity function, i.e. $y_k = h(x_k) = x_k$. Figure 5.1 shows few examples of the zero-order Brownian motions on the sphere starting from the same point, but with different sampling times.

5.2.2 *First-Order Motion Model on Riemannian Manifold*

The object tracking methods often use higher-order motion models to improve the tracking performance. One of them is the first-order motion model or constant velocity model, where it is assumed that the zero-mean process noise acts as acceleration on the objects. This section generalizes the constant velocity motion model to manifolds. Hence, in this case the state of the object contains not only it's position, but also it's velocity. Denote the state with $x = (p, v)$, where $p \in \mathcal{M}$ is position and $v \in T_p\mathcal{M}$ is velocity. It is clear now that the state x is a point of the tangent bundle $T\mathcal{M}$ of the manifold $\mathcal{M}$. It is also supposed that the zero-mean concentrated Gaussian process noise w acts as acceleration on the object.

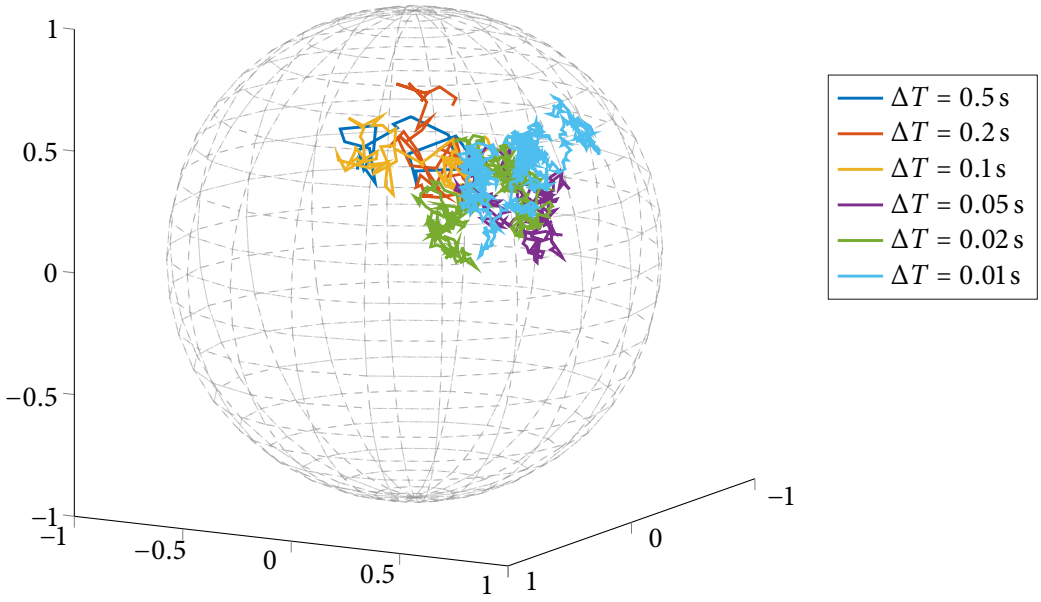


Figure 5.1: Simulation of Brownian motion on a sphere with different sampling times, starting from the same point.

Although the discrete constant velocity model could be derived by discretization of (5.1) where $x(t) = \big(p(t), v(t)\big)$, let's just for simplicity consider that the object moves at constant velocity between two sampling instants and the process noise only acts at sampling moments. We get following

$$p_k = \mathrm{Exp}_{p_{k-1}}\big(\Delta T\, v_{k-1}\big), \tag{5.5}$$

$$v_k = \mathrm{P}_{p_{k-1}\to p_k}\left(v_{k-1} + w_{k-1}\right). \tag{5.6}$$

A few examples of the discrete constant velocity motions on the sphere with different sampling times are shown on figure 5.2.

5.3 Kalman Filters on Manifolds

In this section the Riemannian manifold based KFs are presented. Due to the curvature of the manifolds, the uncertainty of the Kalman filter is described by the covariance matrix on the tangent space of the manifold. Consider a discrete stochastic system whose state takes values on some n-dimensional manifold $\mathcal{M}$ and suppose that the state transition of that system can be described by following equation

$$x_k = f_{k-1}\left(x_{k-1}, w_{k-1}\right), \tag{5.7}$$

where $w_{k-1} \in T_{x_{k-1}}\mathcal{M}$ is the zero-mean concentrated Gaussian process noise signal and $f_{k-1} : \mathcal{M} \times T\mathcal{M} \to \mathcal{M}$ is the non-linear mapping. The model (5.7) does not include the control input u but it can easily be extended to allow the control input. Suppose also that the state x_k of the system is partially observed by a measurement y_k which takes value on a (possibly) different manifold $\mathcal{N}$. The measurement model can then be expressed by

$$y_k = h_k\left(x_k, v_k\right), \tag{5.8}$$

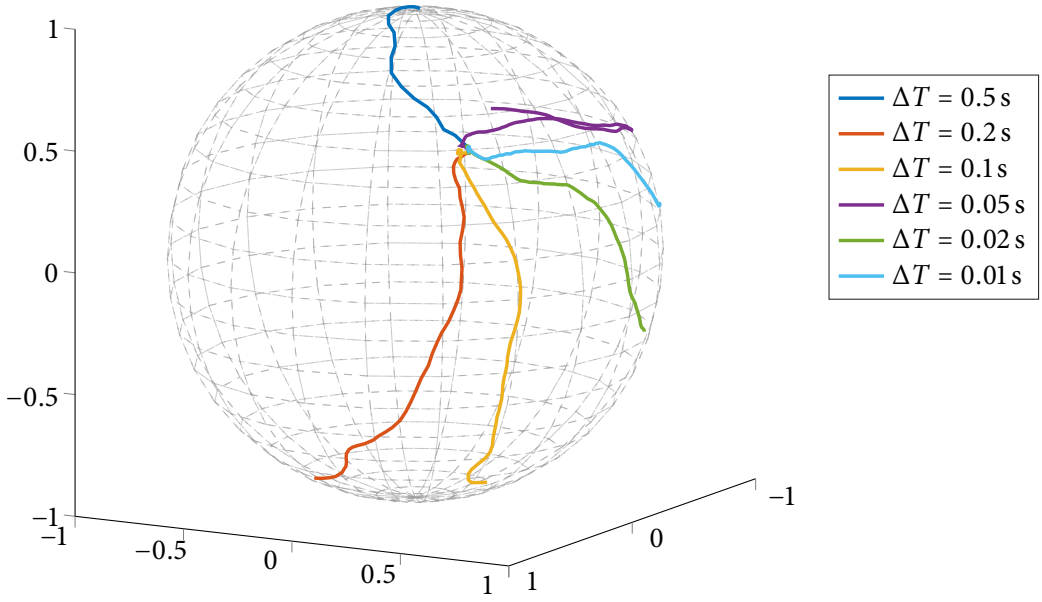


Figure 5.2: Simulation of a first-order Brownian motion on a sphere with different sampling times, starting from the same point.

where v_k is the zero-mean concentrated Gaussian measurement noise. In the rest of this section, the two implementations of the Riemannain manifold KFs are provided. The first one uses Jacobians of the motion and measurement models to propagate the covariance matrix of the state, while the second one utilizes the unscented transform.

5.3.1 *Extended Kalman Filtering on Manifolds*

PREDICTION STEP

Suppose that the prior distribution of the state x is concentrated Gaussian distribution (CGD) with mean $\hat{x}_{k-1}$ and covariance $\hat{P}_{k-1}$. Leonardos *et al.* propose the Riemannian manifold extended Kalman filter (RM-EKF) in [33] to propagate and update the state and its covariance. Although they model the system in a slightly different way, the filtering equations are similar. The prediction step of RM-EKF is given by [33]

$$\hat{x}_{k|k-1} = f_{k-1}(\hat{x}_{k-1}, 0), \tag{5.9}$$

$$\hat{P}_{k|k-1} = F_{k-1}\,\hat{P}_{k-1}\,F_{k-1}^T + L_{k-1}\,Q_{k-1}\,L_{k-1}^T, \tag{5.10}$$

where Q_{k-1} is the covariance matrix of the noise w_{k-1} at time $k-1$, while F_{k-1} and L_{k-1} are Jacobians of transition model given by

$$F_{k-1} = \left.\frac{\partial f_{k-1}}{\partial x_{k-1}}\right|_{\hat{x}_{k-1},\,0}, \qquad L_{k-1} = \left.\frac{\partial f_{k-1}}{\partial w_{k-1}}\right|_{\hat{x}_{k-1},\,0}. \tag{5.11}$$

Given the predicted density of the state parametrized by $\hat{x}_{k|k-1}$ and $\hat{P}_{k|k-1}$, the prediction of the measurement at time k can be obtained by [33]

$$\hat{y}_{k|k-1} = h_k\left(\hat{x}_{k|k-1}, 0\right), \tag{5.12}$$

while the covariance matrix of the predicted measurement is given by

$$\hat{S}_{k|k-1} = H_k \hat{P}_{k|k-1} H_k^T + M_k R_k M_k^T, \tag{5.13}$$

where R_k is the covariance matrix of the measurement noise v_k, while M_k and H_k are Jacobians of measurement model h calculated as follows

$$H_k = \left.\frac{\partial h_k}{\partial x_k}\right|_{\hat{x}_{k|k-1},0}, \qquad M_k = \left.\frac{\partial h_k}{\partial v_k}\right|_{\hat{x}_{k|k-1},0}. \tag{5.14}$$

UPDATE STEP

The Kalman gain of the RM-EKF can now be calculated similarly as in Euclidean version of EKF [33]

$$K_k = \hat{P}_{k|k-1} H_k^T \hat{S}_{k|k-1}^{-1}. \tag{5.15}$$

Given the observation y_k of the state x at time k, the innovation can be calculated as the vector on the tangent space around the predicted measurement as follows

$$\mu_k = \mathrm{Log}_{\hat{y}_{k|k-1}} y_k. \tag{5.16}$$

Finally, the update equation of the state is given by [33]

$$\hat{x}_k = \mathrm{Exp}_{\hat{x}_{k|k-1}} K_k \mu_k, \tag{5.17}$$

while the update of the covariance matrix is obtained by [33]

$$\bar{P}_k = (I - K_k H_k) \hat{P}_{k|k-1}, \tag{5.18}$$

or, written in the Joseph form which is numerically more robust, by

$$\bar{P}_k = (I - K_k H_k) \hat{P}_{k|k-1} (I - K_k H_k)^T + K_k M_k R_k M_k^T K_k^T. \tag{5.19}$$

The covariance matrix $\bar{P}_k$ calculated by (5.18) or (5.19) is in the tangent space $T_{\hat{x}_{k|k-1}}\mathcal{M}$ and must be transported to $T_{\hat{x}_k}\mathcal{M}$, resulting in

$$\hat{P}_k = \mathrm{P}_{\hat{x}_{k|k-1} \to \hat{x}_k} \left(\bar{P}_k\right). \tag{5.20}$$

IMPLEMENTATION ISSUES

When working with manifolds, there are three different approaches: intrinsic, local and extrinsic. Within the intrinsic approach in differential geometry, objects such as tensors, differential operators, and others are defined in an invariant way, without the reference to any coordinate frame. However, in filtering applications it is necessary to have some kind of coordinate representation. In extrinsic approach, the manifold is embedded in a higher dimensional Euclidean space and objects such as points, tangent vectors and tensors can be written in coordinates of the ambient Euclidean space. One example of estimation on RM entirely in extrinsic coordinates is [33]. When working in extrinsic coordinates, some relations like (5.15) are not valid. The reason for this is that covariance matrices are

constrained to the tangent space of the manifold, which has lover dimension than the ambient Euclidean space. Hence the covariance matrix is not of full rank and cannot be inverted when written in extrinsic coordinates. To overcome this issue, the set of linearly independent vectors that span the tangent space of the manifold must be chosen. Arranging those vectors in a matrix forms a linear map from higher dimensional ambient space, to the tangent space of the manifold. Using this matrix, covariance, as well as Jacobians can be transformed to lower dimensional space. Now, filtering equations can be applied and resulting covariance can be transformed back to extrinsic coordinates

Another problem with using extrinsic coordinates is that the dimension of the Euclidean space in which the manifold can be embedded might be much larger compared to the dimension of the manifold. Furthermore, additional care must be taken to ensure that the state of the filter and its covariance does not fall of the manifold due to the numerical errors.

The alternative is to use the representation in local coordinates. However, not all manifolds can be uniquely covered with only one coordinate chart (e.g. 2-sphere) and instead a set of multiple coordinate charts must be used. The issue with this is that when the system state approaches the end of domain of one coordinate chart, it is necessary to convert the state, as well as covariance matrix and Jacobians used in filtering to different chart. It might also be tedious to calculate Jacobians in local coordinate representation. However, this can be avoided by calculating the Jacobian in extrinsic coordinates and then converting it in the specific coordinate chart when needed. In this thesis, the representation in local coordinates is used.

5.3.2 *Unscented Kalman Filtering on Manifolds*

Like in the Euclidean case, it is also possible to implement the unscented version of KF for Riemannian manifolds. The RM-UKF was introduced by Hauberg *et al.* in [32], while some additional variants such as augmented state RM-UKF were proposed by Menegaz *et al.* in [34]. The main part of the unscented Kalman filter is the unscented transform, this section starts by presenting the extension of unscented transform to RMs.

RIEMANNIAN UNSCENTED TRANSFORM

Suppose that the function $f : \mathcal{M} \to \mathcal{N}$ is the mapping from m-dimensional manifold $\mathcal{M}$ onto n-dimensional manifold $\mathcal{N}$. Suppose as well that the concentrated Gaussian variable $x \sim \mathcal{N}_{\mathcal{M}}(\cdot \mid \mu, P)$ is given. The purpose of unscented transform is to find how x transforms under mapping f. The d-dimensional CGD can be approximated with $2d+1$ deterministically sampled points. Applying the Cholesky decomposition on the covariance matrix P yields the sigma vectors on the tangent space $T_\mu\mathcal{M}$

$$\bar{\sigma}_i = \sqrt{d+\kappa} \cdot L_i, \qquad i = 1, \ldots, d, \tag{5.21}$$

$$\bar{\sigma}_{i+d} = -\sqrt{d+\kappa} \cdot L_i, \qquad i = 1, \ldots, d, \tag{5.22}$$

where L_i represents the i-th column of the Cholesky factor of matrix P. The sigma points are obtained by projecting sigma vectors eqs. (5.21) and (5.22) to $\mathcal{M}$ by exponential map

$$\sigma_i = \mathrm{Exp}_\mu(\bar{\sigma}_i) \qquad i = 1, \ldots, 2d. \tag{5.23}$$

Keep in mind that one sigma point is the mean $\sigma_0 = \mu$. To obtain the unscented transform of distribution, sigma points are propagated by mapping f. The approximation of the mean of the propagated distribution is obtained by calculating the Kärcher mean of propagated sigma points [32]

$$\bar{\mu} = \arg\min_{x \in \mathcal{N}} \sum_{i=0}^{2d} w_i d^2_{\mathcal{N}}\left(x, f(\sigma_i)\right), \tag{5.24}$$

where w_i are weights calculated as in case of a regular UKF (see eqs. (3.33) – (3.37)). Once the mean (5.24) is calculated, it is possible to project the propagated sigma points $f(\sigma_i)$ to the tangent space at new mean $T_{\bar{\mu}}\mathcal{N}$ and use them to calculate the covariance matrix of the propagated distribution [32]

$$\bar{P} = \sum_{i=0}^{2d} w_i (\mathrm{Log}_{\bar{\mu}} f(\sigma_i))(\mathrm{Log}_{\bar{\mu}} f(\sigma_i))^T. \tag{5.25}$$

Thus, the transformed CGD is given by $f(x) \sim \mathcal{N}_{\mathcal{N}}(\cdot \mid \bar{\mu}, \bar{P})$.

PREDICTION STEP

Suppose that the estimation of state of the system at step $k-1$ is $\hat{x}_{k-1}$, while its covariance is $\hat{P}_{k-1}$ and let $f : \mathcal{M} \to \mathcal{M}$ be the state transition model of system. The prediction step of the RM-UKF is calculated as follows. Firstly, the set of $2d+1$ sigma points $\sigma_{i,k|k-1}$ is obtained as in eqs. (5.21) and (5.22). Then, the predicted state is obtained as the mean of sigma points transformed by the function f [32]

$$\hat{x}_{k|k-1} = \arg\min_{x \in \mathcal{M}} \sum_{i=0}^{2d} w_i d^2_{\mathcal{M}}\left(x, f(\sigma_{i,k-1})\right), \tag{5.26}$$

while the predicted covariance is given by

$$\hat{P}_{k|k-1} = \sum_{i=0}^{2d} w_i (\mathrm{Log}_{\hat{x}_{k-1}} f(\sigma_i))(\mathrm{Log}_{\hat{x}_{k-1}} f(\sigma_i))^T + Q_k, \tag{5.27}$$

where Q is the covariance of the process noise. Alternatively, the process noise covariance Q can be added at the beginning of the unscented transform, i.e. to the prior covariance $\hat{P}_{k-1}$.

Let now $h : \mathcal{M} \to \mathcal{N}$ be the measurement model of the system. The new set of sigma points $\sigma_{i,k|k-1}$ are calculated from $\hat{x}_{k|k-1}$ and $\hat{P}_{k|k-1}$. The prediction of the measurement is obtained by using the unscented transform on the new set of sigma points [32]

$$\hat{y}_{k|k-1} = \arg\min_{y \in \mathcal{N}} \sum_{i=0}^{2d} w_i d^2_{\mathcal{N}}\left(y, h(\sigma_{i,k|k-1})\right). \tag{5.28}$$

Now, the innovations covariance and the cross-covariance matrices are given by [32]

$$\hat{S}_{k|k-1} = \sum_{i=0}^{2d} w_i \left(\mathrm{Log}_{\hat{y}_{k|k-1}} h(\sigma_{i,k|k-1})\right)\left(\mathrm{Log}_{\hat{y}_{k|k-1}} h(\sigma_{i,k|k-1})\right)^T + R_k, \tag{5.29}$$

$$\hat{C}_{k|k-1} = \sum_{i=0}^{2d} w_i \left(\mathrm{Log}_{\hat{x}_{k|k-1}} \sigma_{i,k|k-1}\right)\left(\mathrm{Log}_{\hat{y}_{k|k-1}} h(\sigma_{i,k|k-1})\right)^T, \tag{5.30}$$

where R is the measurement noise covariance matrix.

update step

Given the innovation covariance matrix (5.29) and cross-covariance (5.30), the Kalman gain at step k is given by [32]

$$K_k = \hat{C}_{k|k-1}\hat{S}^{-1}_{k|k-1}. \tag{5.31}$$

Note that the Kalman gain $K_k : T_{\hat{y}_{k|k-1}}\mathcal{N} \to T_{\hat{x}_{k|k-1}}\mathcal{M}$ is the mapping from the tangent space of the measurement manifold to the tangent space of the manifold $\mathcal{M}$. Denote with $\mu_k = \mathrm{Log}_{\hat{y}_{k|k-1}} y_k$ the innovation of the measurement y_k at time k. Then the update of the state x is given by [32, 34]

$$\hat{x}_k = \mathrm{Exp}_{\hat{x}_{k|k-1}} K_k \mu_k. \tag{5.32}$$

The update equation for the covariance matrix is the same as in Euclidean case [32]

$$\bar{P}_k = \hat{P}_{k|k-1} - K_k \hat{S}_{k|k-1} K_k^T, \tag{5.33}$$

however, the resulting covariance is written in the tangent space at $\hat{x}_{k|k-1}$ and, hence, it must be parallel transported to $T_{\hat{x}_k}\mathcal{M}$

$$\hat{P}_k = \mathrm{P}_{\hat{x}_{k|k-1}\to\hat{x}_k}\left(\bar{P}_k\right). \tag{5.34}$$

5.4 Constant Velocity Kalman Filters on Riemannian Manifolds

This section considers the object tracking on Riemannian manifold with constant velocity motion model. In this case the state of the object contains not only its position, but also the velocity. Denote the state with $x = (p, v)$, where $p \in \mathcal{M}$ is position and $v \in T_p\mathcal{M}$ is velocity. It is clear now that the state x is a point of the tangent bundle $T\mathcal{M}$. The discrete constant velocity model $f : T\mathcal{M} \times T\mathcal{M} \to T\mathcal{M}$ is given by

$$\begin{bmatrix} p_k \\ v_k \end{bmatrix} = f_{k-1}\left(x_{k-1}, w_{k-1}\right) = \begin{bmatrix} \mathrm{Exp}_{p_{k-1}}(\Delta T\, v_{k-1}) \\ \mathrm{P}_{p_{k-1}\to p_k}\left(v_{k-1} + w_{k-1}\right) \end{bmatrix}, \tag{5.35}$$

where w_{k-1} is the zero-mean concentrated Gaussian process noise with covariance Q_{k-1} defined at tangent space $T_{p_{k-1}}\mathcal{M}$. Note that both $\mathrm{Exp}_{p_{k-1}}$ and $\mathrm{P}_{p_{k-1}\to p_k}(\cdot)$ in (5.35) are expressions of $\mathcal{M}$, not $T\mathcal{M}$. Suppose that the sensor measures only the position of the object. The observation model can then be written as

$$y_k = h_k(x_k, \omega_k) = \mathrm{Exp}_{p_k}(\omega_k), \tag{5.36}$$

where ω_k is the zero-mean concentrated Gaussian measurement noise[1].

[1] Measurement noise is usually denoted with v within this thesis, however ω is used here instead because v represents the velocity of the state x.

5.4.1 *Riemannian Manifold Extended Kalman Filter*

To implement the RM-EKF with the generalized constant velocity motion model, it is necessary to calculate the following jacobians of (5.35)

$$F_{k-1} = \left.\frac{\partial f_{k-1}}{\partial x_{k-1}}\right|_{(\hat{x}_{k-1},0)} = \begin{bmatrix} \left.\frac{\partial f_{1,k-1}}{\partial p_{k-1}}\right|_{(\hat{x}_{k-1},0)} & \left.\frac{\partial f_{1,k-1}}{\partial v_{k-1}}\right|_{(\hat{x}_{k-1},0)} \\ \left.\frac{\partial f_{2,k-1}}{\partial p_{k-1}}\right|_{(\hat{x}_{k-1},0)} & \left.\frac{\partial f_{2,k-1}}{\partial v_{k-1}}\right|_{(\hat{x}_{k-1},0)} \end{bmatrix}, \tag{5.37}$$

$$L_{k-1} = \left.\frac{\partial f_{k-1}}{\partial w_{k-1}}\right|_{(\hat{x}_{k-1},0)} = \begin{bmatrix} \left.\frac{\partial f_{1,k-1}}{\partial w_{k-1}}\right|_{(\hat{x}_{k-1},0)} \\ \left.\frac{\partial f_{2,k-1}}{\partial w_{k-1}}\right|_{(\hat{x}_{k-1},0)} \end{bmatrix} = \begin{bmatrix} 0 \\ \left.\frac{\partial f_{2,k-1}}{\partial w_{k-1}}\right|_{(\hat{x}_{k-1},0)} \end{bmatrix}. \tag{5.38}$$

and the Jacobians of observation model (5.36) with respect to x_k and ω_k

$$H_k = \left.\frac{\partial h_k}{\partial x_k}\right|_{(\hat{x}_{k|k-1},0)} = \begin{bmatrix} \left.\frac{\partial h_k}{\partial p_k}\right|_{(\hat{x}_{k|k-1},0)} & \left.\frac{\partial h_k}{\partial v_k}\right|_{(\hat{x}_{k|k-1},0)} \end{bmatrix} \tag{5.39}$$

$$= \begin{bmatrix} \left.\frac{\partial h_k}{\partial p_k}\right|_{(\hat{x}_{k|k-1},0)} & 0 \end{bmatrix}, \tag{5.40}$$

$$M_k = \left.\frac{\partial h_k}{\partial \omega_k}\right|_{(\hat{x}_{k|k-1},0)}. \tag{5.41}$$

Consider the case of constant velocity model on n-sphere $\mathcal{S}^n_\rho$. Using relations of exponential map and parallel transport of a sphere given by eqs. (2.35) and (2.37) yields following Jacobians

$$F = \begin{bmatrix} I \cdot \cos\theta & I \cdot \frac{\Delta T}{\theta} \sin\theta - pv^T \frac{\theta}{\|v\|^2} \sin\theta + vv^T \left(\frac{\Delta T}{\|v\|^2} \cos\theta - \frac{\rho}{\|v\|^3} \sin\theta \right) \\ -I \cdot \frac{\|v\|}{\rho} \sin\theta & I \cdot \cos\theta - vv^T \frac{\Delta T}{\rho\|v\|} \sin\theta - pv^T \left(\frac{1}{\rho\|v\|} \sin\theta + \frac{\Delta T}{\rho^2} \cos\theta \right) \end{bmatrix}, \tag{5.42}$$

$$L = \begin{bmatrix} 0 \\ I - \frac{pv^T}{\rho\|v\|} \sin\theta - \frac{vv^T}{\|v\|^2}(1 - \cos\theta) \end{bmatrix}, \qquad H = \begin{bmatrix} I & 0 \end{bmatrix}, \qquad M = I, \tag{5.43}$$

where $\theta = \frac{\Delta T \|v_{k-1}\|}{\rho}$. For the full derivation of these Jacobians see appendix A. Note that jacobians in eqs. (5.42) and (5.43) are written in the extrinsic coordinates of $\mathbb{R}^{n+1}$. To use them in the local coordinate implementation of RM-EKF, it is necessary to convert them in the corresponding coordinate system.

5.4.2 *Riemannian Manifold Unscented Kalman Filter*

The adaptation of RM-UKF from constant model to the constant velocity model is straightforward. Sigma points are created on the tangent space of the tangent bundle $T\mathcal{M}$ and then are projected to the $T\mathcal{M}$ via exponential mapping of the bundle. Then, sigma points are transformed according to the first-order model given by (5.35) and the new mean and covariance are then calculated by eqs. (5.26) and (5.27). To obtain the prediction of the measurement, the new sigma points are created from the state and are then projected to the measurement space which is the manifold $\mathcal{M}$.

Table 5.1: Root-mean-square error (RMSE) of four filtering methods on unit sphere over 100 Monte Carlo trials.

Method	RMSE [m]
RM-EKF	0.03811
RM-UKF	0.03812
vMF	0.03810
KF	0.03811

5.5 Experimental Results

This section provides the experimental evaluation of the Riemannian manifold EKF and UKF. The performance of filters is measured on the simulated data. In order to show that proposed methods can be applied to different Riemannian manifolds, they are tested on a few different manifolds, i.e. sphere and the space of positive definite matrices. Furthermore, both the zero-order and first order models were used in the simulations. In addition to the RM-EKF and RM-UKF, another two filters were used in simulations as baselines: the von Mises-Fisher filter and the discrete Kalman filter. The vMF was used in experiments on the sphere. In the case of the discrete KF, the state was projected to the manifold after every prediction and update step.

5.5.1 *Zero-Order Model*

filtering on the unit sphere

This section provides the evaluation of zero-order filtering methods on the unit sphere. Methods are tested on different dimensions of the system, ranging from 1 to 20.

Table 5.2: RMSE of RM-EKF, RM-UKF, vMF and KF on unit spheres of different dimensions.

	$\mathcal{S}^1$	$\mathcal{S}^2$	$\mathcal{S}^3$	$\mathcal{S}^4$	$\mathcal{S}^5$	$\mathcal{S}^{10}$	$\mathcal{S}^{15}$	$\mathcal{S}^{20}$
RM-EKF	0.02724	0.03811	0.04700	0.05415	0.06036	0.08560	0.1053	0.1220
RM-UKF	0.02726	0.03812	0.04700	0.05417	0.06039	0.08563	0.1054	0.1220
vMF	0.03125	0.03810	0.04700	0.05415	0.06036	0.08560	0.1053	0.1220
KF	0.02724	0.03811	0.04700	0.05415	0.06036	0.08560	0.1053	0.1220

In this experiment the trajectory was generated by simulating Brownian motion on the sphere. The Brownian motion was driven by the concentrated Gaussian noise with covariance matrix $Q = \sigma_q^2 \cdot I_2 = 0.02 \cdot I_2$, while the measurements were simulated by adding concentrated Gaussian noise with covariance matrix $R = \sigma_r^2 \cdot I_2 = 0.001 \cdot I_2$. The sampling time[2] in the simulation was set to $\Delta T = 0.1\,\text{s}$ and the duration of the simulation was 200 steps. The experiment consisted of 100 Monte Carlo runs with randomly generated starting point.

[2] This was the sampling rate at which the measurement were created, however, the simulation of a Brownian motion used sampling time $\Delta T = 0.005\,\text{s}$.

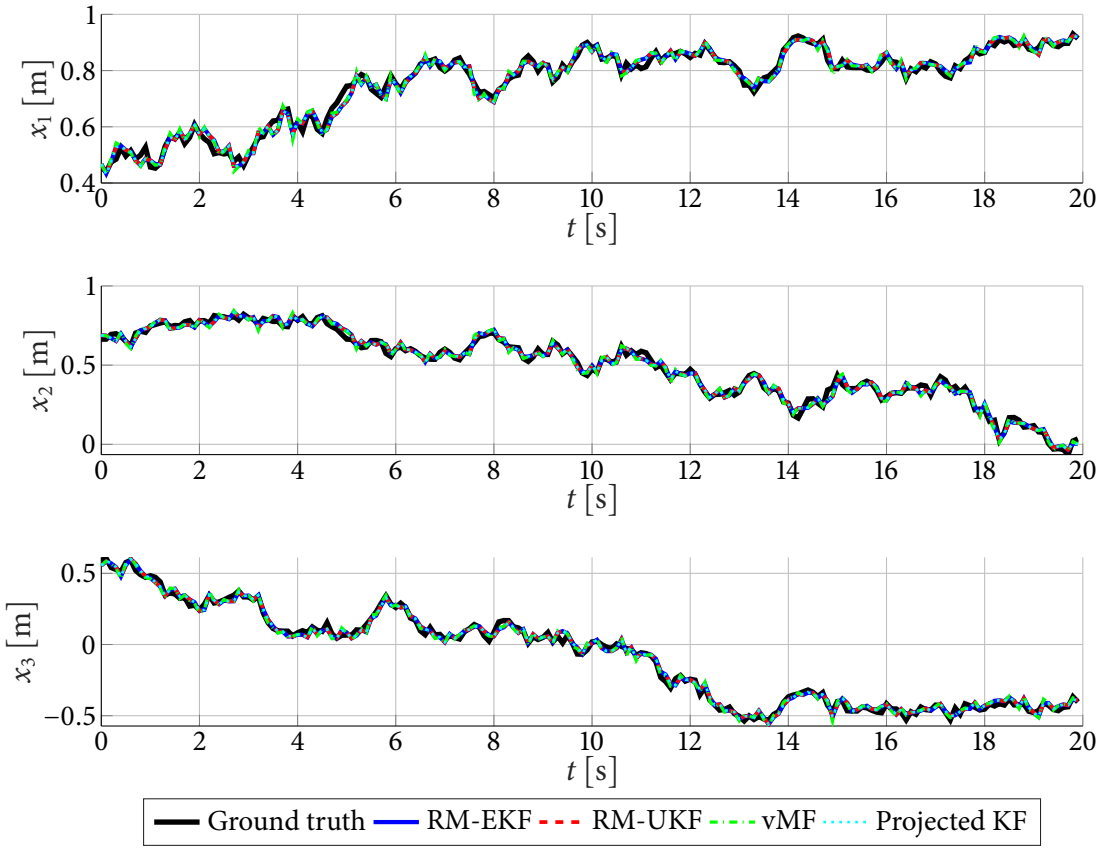


Figure 5.3: Simulation results of one trial of tracking on sphere with zero-order motion model.

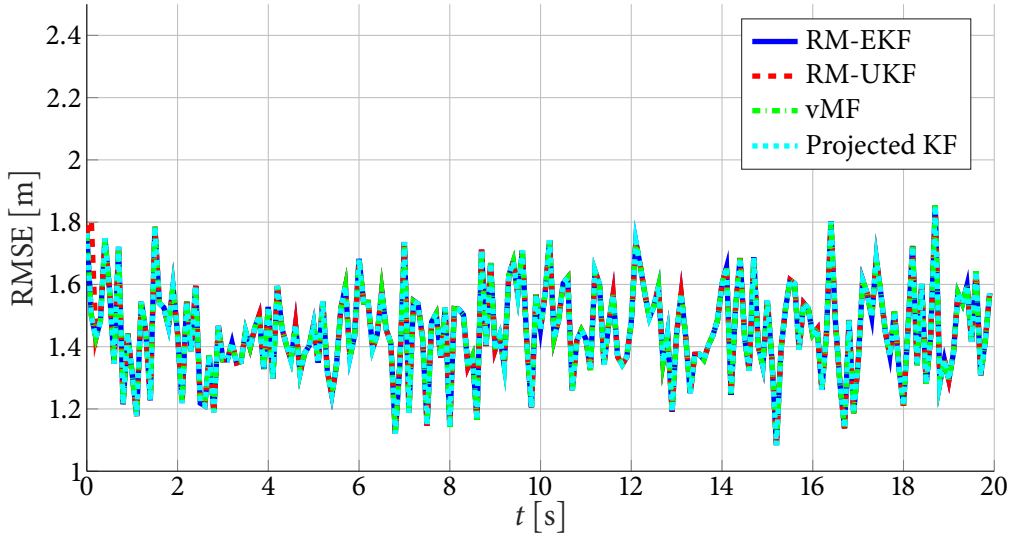


Figure 5.4: RMSE of filtering on unit sphere with with zero-order motion model.

Process and measurement noise covariance matrices of the RM-EKF were set to $Q = 0.02\Delta T \cdot I_2$, and $R = 0.001 \cdot I_2$. The same values were used by the RM-UKF. In addition, Jacobians of the RM-EKF were set to identity matrices, i.e. $F = H = L = M = I_n$. Additionally, the coefficients for calculating weights of sigma points of RM-UKF were $\alpha = \sqrt{0.5}$, $\beta = 2$ and $\kappa = 1$. The parameters of vMF were set to $\kappa_d = \frac{1}{\Delta T \sigma_q^2} = 500$ and $\kappa_o = \frac{1}{\sigma_r^2} = 1000$. Moreover, the parameters of the discrete KF were $Q = 0.02\Delta T \cdot I_3$ and

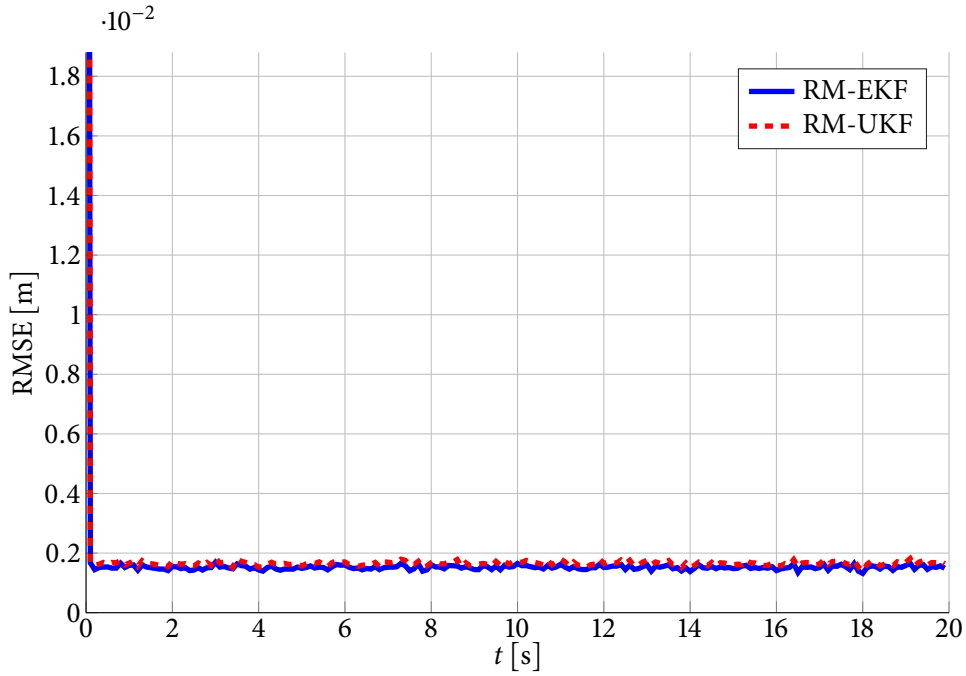


Figure 5.5: RMSE of filtering on space of SPD matrices with zero-order motion model.

$R = 0.001 \cdot I_3$. All of the filters were initialized with the first measurement x_0 and the covariances were set to $P_0 = 0.001 \cdot I_2$ for RM-EKF and RM-UKF and to $P_0 = 0.001 \cdot I_3$ for discrete KF. The initial concentration parameter of vMF filter was set to $\kappa_0 = 1000$.

Figure 5.4 shows the mean-square error (MSE) averaged over 100 Monte Carlo trials of these four filters. Furthermore, RMSE averaged over all runs and over time is given by table 5.1. The state estimates are shown on figure 5.3. Additionally, the methods were also tested on higher dimensional spheres and the results are shown in table 5.2. It can be concluded that for the systems that all four methods perform similar when the ground truth system obeys the zero-order Brownian motion.

Filtering on the space of SPD matrices

This section provides the evaluation of state estimation on the space of SPD matrices. The experiment consisted of 100 Monte Carlo trials. Each trial lasted 200 steps and the sampling time was $\Delta T = 0.1\,\mathrm{s}$. The ground truth trajectory was generated by simulating the Brownian motion on $\mathcal{P}_n$ driven by the concentrated Gaussian noise with covariance matrix $Q = \sigma_q^2 \Delta T \cdot I_n$, where $\sigma_q = 0.005\,\mathrm{m\,s^{-1}}$ is the standard deviation of the noise. Measurements were simulated by applying the concentrated Gaussian measurement noise with covariance $R = \sigma_r^2 \cdot I_n$, where $\sigma_r = 0.001\,\mathrm{m}$. Covariance matrices of the RM-EKF and RM-UKF were set to $Q = 0.005^2 \Delta T \cdot I_n$ and $R = 0.001^2 \cdot I_n$. In case of RM-EKF, Jacobian matrices were $F = H = L = M = I_n$. RM-UKF parameters used to calculate weights of sigma points were set to $\alpha = 1 \cdot 10^{-3}$, $\beta = 2$ and $\kappa = 1$. The RMSE of the RM-EKF and the RM-UKF filters in the case of the $\mathcal{P}_2$ manifold are shown on table 5.3, while table 5.4 how the error of filters depend on the dimension of the space.

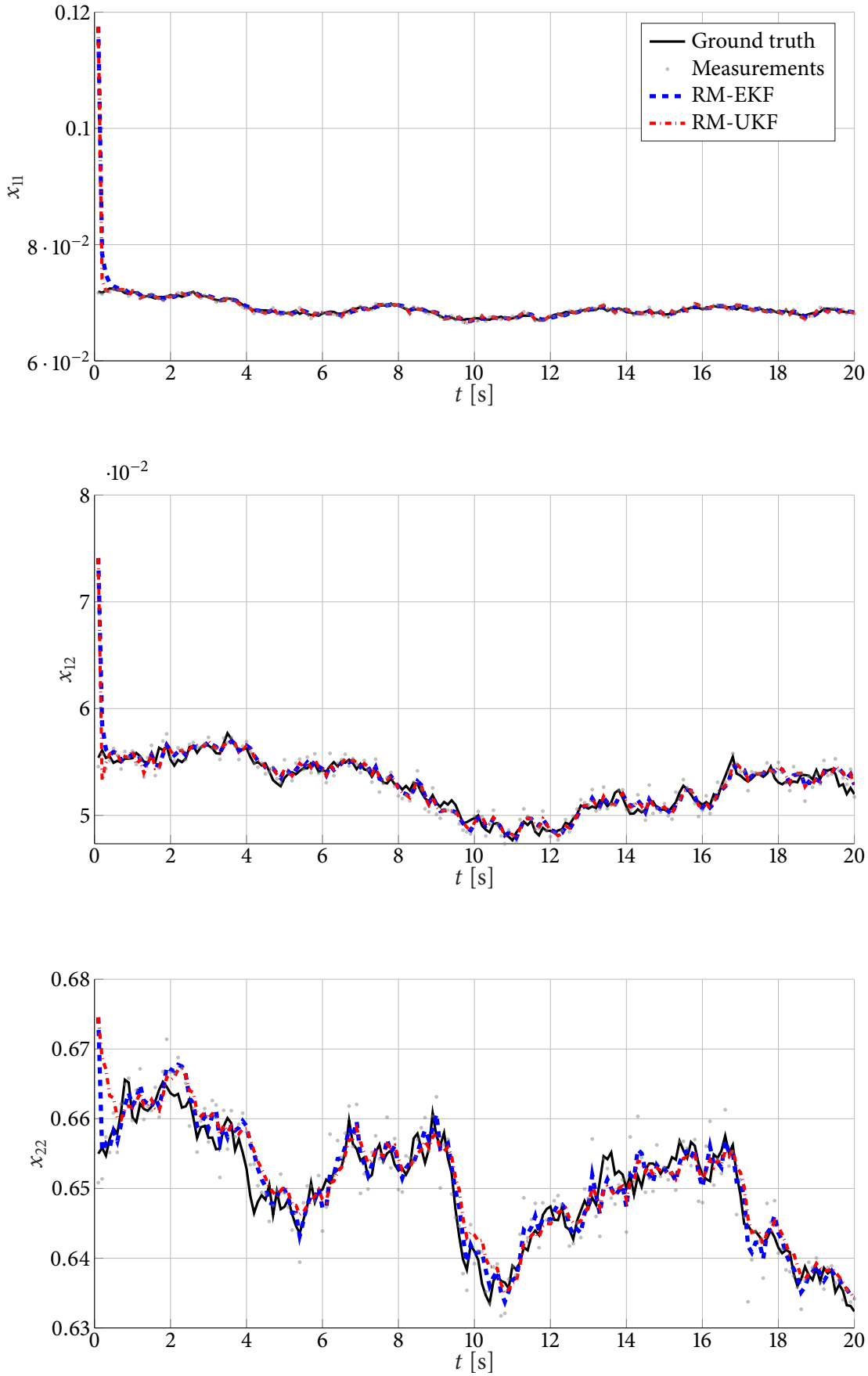


Figure 5.6: State estimation on $\mathcal{P}_2$. Black line is the ground truth, grey dots are measurements, blue line is the output of the RM-EKF and the red line is the output of the RM-UKF.

Table 5.3: RMSE of filtering methods on the space of 2×2 SPD matrices over 100 Monte Carlo trials.

Method	RMSE [m]
RM-EKF	0.006856
RM-UKF	0.006887

Table 5.4: RMSE of RM-EKF and RM-UKF on SPD manifolds of different dimension.

	$\mathcal{P}_2$	$\mathcal{P}_3$	$\mathcal{P}_4$	$\mathcal{P}_5$
RM-EKF	0.006856	0.007284	0.007914	0.008199
RM-UKF	0.006887	0.007362	0.008038	0.008246

5.5.2 *First-Order Model*

This section provides the experimental evaluation of the filtering methods on Riemannian manifolds for the systems that obey the constant velocity motion model.

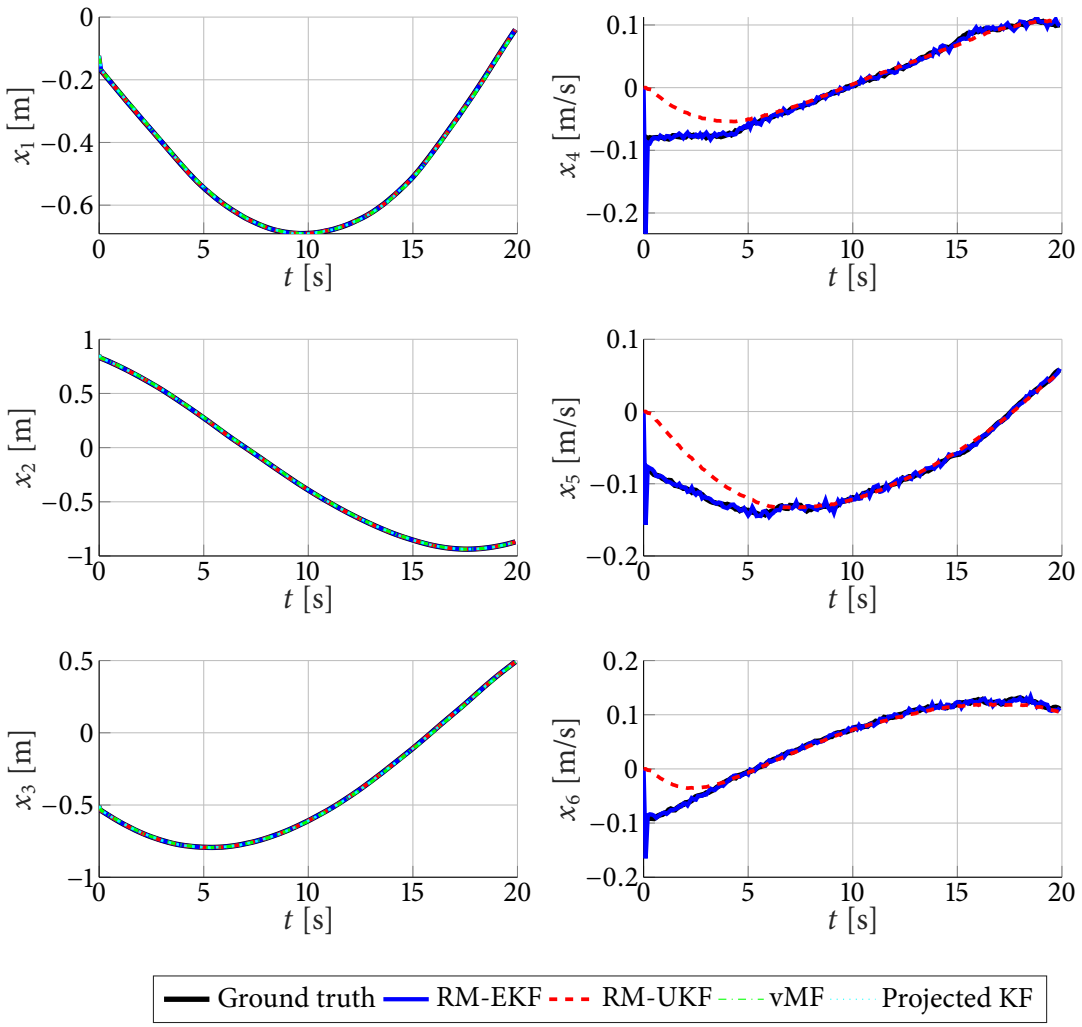


Figure 5.7: Ground truth and state estimates in the simulation whit the first-order motion model on the unit sphere.

Filtering on the unit sphere

In this experiment, the ground truth trajectory was created using discrete constant velocity model on the sphere. Parameters used to create the simulation scenario were as follows. The trajectory was generated by simulating the Brownian motion on the tangent bundle of the sphere given by eqs. (5.5) and (5.6). The process was driven by the concentrated Gaussian noise with covariance matrix $Q = \sigma_q^2 \Delta T \cdot I_n$, where the standard deviation was $\sigma_q = 0.005\,\mathrm{m\,s^{-1}}$. Measurements were simulated by projecting the state of the system to the sphere, keeping only the position, and then adding the concentrated Gaussian measurement noise with covariance matrix $R = \sigma_r^2 \cdot I_n$, where $\sigma_r = 0.001\,\mathrm{m}$.

Jacobians of the RM-EKF were set to eqs. (5.42) and (5.43). Covariance matrices of the process and measurement noise of the RM-EKF were set to $Q = 0.005^2 \cdot I_2$ and $R = 0.001^2 \cdot I_2$. The covariance of the state of the filter was initialised to $P_0 = \mathrm{diag}(0.01, 0.01, 0.1, 0.1)$.

The covariance matrix of the RM-UKF was set to

$$Q = 0.05^2 \cdot \begin{bmatrix} \frac{\Delta T^4}{4} & 0 & \frac{\Delta T^3}{2} & 0 \\ 0 & \frac{\Delta T^4}{4} & 0 & \frac{\Delta T^3}{2} \\ \frac{\Delta T^3}{2} & 0 & \Delta T^2 & 0 \\ 0 & \frac{\Delta T^3}{2} & 0 & \Delta T^2 \end{bmatrix},$$

and the covariance matrix of the measurement noise was $R = 0.001^2 \cdot I_2$. Additionally, parameters of RM-UKF that control the spread of the sigma points were $\alpha = 0.1$, $\beta = 2$ and $\kappa = 1$. The initial covariance of the state of the filter was set to $P_0 = \mathrm{diag}(0.01, 0.01, 0.1, 0.1)$.

Parameters of the vMF filter were $\kappa_d = 50$, $\kappa_o = 1000$ and the initial concentration of the filter was $\kappa_0 = 100$.

Jacobians of the baseline discrete KF were $F = H = I_3$, while covariance matrices of the process and measurement noise were $Q = 0.005^2 \cdot I_3$ and $R = 0.001^2 \cdot I_3$. Covariance of the state of the filter was initialised to $P_0 = 0.001 \cdot I_3$.

To assess filters, 10 Monte Carlo simulations were conducted. Initial states of filters were moved for 0.05 m in the random direction from the ground truth. The figure 5.7 shows the estimated states, while figure 5.8 shows the estimation error. It can be seen that the RM-EKF and RM-UKF have similar performance and both significantly outperform the vMF and baseline Kalman filter. The average RMSE over time and all Monte Carlo runs is 0.00368 for RM-EKF, 0.003799 for RM-UKF, 0.00902 for baseline KF and 0.01796 for vMF filter. As expected, the methods that use the constant velocity motion model outperformed the vMF and baseline KF.

Filtering on diagonal positive definite matrices

This section provides the evaluation of the first-order estimation on the space of the diagonal positive definite matrices $\mathcal{D}_3$, which is the Riemannian submanifold of SPD matrices. The ground truth was generated by simulating the discrete constant velocity model driven by zero-mean concentrated Gaussian noise with covariance $Q = 0.005^2 \cdot I_3$. Measurements were simulated by applying the concentrated zero-mean Gaussian measurement noise with covariance $R = 0.005^2 \cdot I_3$. The duration of the simulation was 200 steps with sampling period $\Delta T = 0.05\,\mathrm{s}$. Parameters of the RM-EKF and RM-UKF with constant velocity

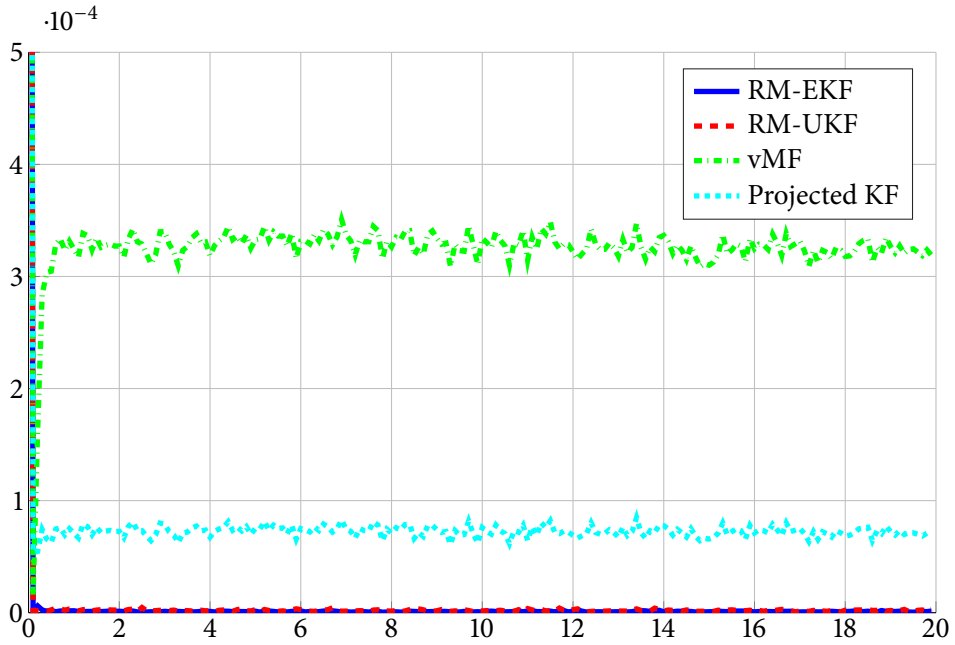


Figure 5.8: MSE of state estimation on the tangent bundle of the $\mathcal{S}^2$ averaged over 10 Monte Carlo runs.

motion model were as follows. Process and measurement noise covariances were set to same values used to simulate the ground truth. States of filters were initialised with the first measurement. Covariance matrices of filters were set to $P_0 = \text{blkdiag}(R, I_3)$. Jacobians of the RM-EKF with first-order motion model on $\mathcal{D}_3$ are calculated in appendix A.2. Additionally, a zero-order motion model RM-EKF was also used in this experiment in order to show the advantage of using the first-order motion model. Its parameters were as follows: process noise covariance was set to $Q = 0.01^2 \cdot I_3$, measurement noise covariance was set to $R = 0.005^2 \cdot I_3$ and the initial covariance of the filter was $P_0 = 0.005^2 \cdot I_n$. The results of 50 Monte Carlo runs are shown in table 5.5. Both filters with constant velocity model significantly outperformed the filter with constant model as expected. Furthermore, the RM-EKF with constant velocity model outperformed the RM-UKF with constant velocity model. This is most likely due to the numerical optimisation methods used to compute the Kärcher mean and logarithmic mapping on the tangent bundle of the manifold. Those optimisation methods were limited to only a few iterations so that it can be computed in reasonable time.

Table 5.5: RMSE of filtering on tangent bundle of $\mathcal{D}_3$ averaged over time and 50 Monte Carlo runs.

Method	RMSE
Zero-order RM-EKF	0.5232
First-order RM-UKF	0.09575
First-order RM-EKF	0.07027

5.6 Summary

In this chapter, we proposed a novel approach to state estimation problem on RMs. We extended a RM-UKF method by expanding the state of the filter with its velocity. Such expanded state is now an element of the tangent bundle of the Riemannian manifold. We then applied the tools of the geometry of the tangent bundles of Riemannian manifolds in order to develop the version of the unscented Kalman filter that can handle systems whose state evolve on the tangent bundle. Hence, the proposed filtering method can use the first-order motion model on Riemannian manifold. The proposed filtering method achieves slightly worse results than the RM-EKF with the first-order motion model. Nevertheless, the unscented Kalman filter with the first-order motion model uses the unscented transform in order to propagate the uncertainty of the filter. Hence, it does not need the Jacobians of the motion model and as such it provides a good alternative to the RM-EKF with first-order motion model in case it is tedious to obtain the closed-form Jacobians. In the end of the chapter, we conducted several experiments in order to evaluate proposed method. Experiments contain simulations with both zero and first-order motion models and are conducted on various manifolds such as unit sphere and space of symmetrical positive definite matrices. Nonetheless, proposed method is not limited only on those manifolds, but can by applied to any Riemannian manifold.

Chapter 6

Probabilistic Multi Object Tracking on Riemannian Manifolds

In the previous chapter we have studied the problem of the state estimation on RMs. The goal of this chapter is to extend developed methods to the case of tracking of multiple moving objects.

The rest of this chapter is structured as follows. We first start by giving a brief overview of the literature regarding the MTT on Riemannian manifolds in section 6.1. Then, in section 6.2 we propose the extension of JIPDAF to Riemannian manifolds. To achieve this goal, we utilize the Riemannian manifold extended and unscented KFs presented in the previous chapter. Finally, in section 6.4 we show the empirical results of proposed method.

6.1 Related work

Recently, many works regarding multiple target tracking on non-Euclidean spaces nave been proposed, however, they usually focus on a specific class of the manifold. In [21] Marković *et al.* consider the problem of tracking on the unit circle. They derive the PHD recursion on the von Mises mixtures. Authors tested the proposed filter on the simulated data and compared it with the Gaussian mixture PHD filter. The evaluation showed significant improvement with respect to OSPA metric. In [132] Markovic *et al.* consider the MTT with directional-only sensors. They develop the PDAF and JPDAF trackers based on the von Mises-Fisher Bayesian filter. To ensure that the posterior distribution remains in the von Mises-Fisher distribution family, authors apply the moment matching technique based on Kullback-Leibler divergence. Furthermore, their work on vMF-JPDAF is extended to all hyperspheres in [24]. The performance of the proposed methods is demonstrated on synthetic data.

One popular type of non-Euclidean spaces that is often used in various robotics-related applications are Lie groups. Ćesić *et al.* studied the problem of mixture reduction on Lie groups in [19]. As an example of application of their mixture reduction method, they presented the PHD filter on Matrix Lie groups. The performance of their Lie group PHD filter is presented in Monte Carlo simulations. In [133] Ćesić *et al.* tackle the problem of moving object detection and tracking in the context of adaptive driver assistance system (ADAS). They propose to model the sensor data in polar coordinates in order to model the measurement uncertainty more accurately. Hence, both the measurement model and motion model of the rigid body reside on the Lie group geometry. Authors decide to apply the Lie group EKF in order to perform the state estimation for that system. Furthermore, authors adapt the JIPDAF in order to achieve multi target tracking on the Lie groups.

In [134], Bak *et al.* deal with the problem of MTT in crowded scenes from a single camera. They extract the appearance models from the tracks obtained by a short-term MTT algorithm. These appearance models that are based on the covariance descriptors are then used for re-identification of targets, i.e. to achieve long-term tracking. Since covariance descriptors confine to the RM, authors propose to use the discriminative analysis on Riemannian manifold to link tracks. According to authors, the evaluation of their method showed that the discriminative analysis significantly reduced the number of false alarms and identity switches in both single camera tracking and in track matching between non-overlapping cameras in multi-camera tracking.

6.2 PDAF on Riemannian manifolds

Before tackling the full MTT problem on RMs, consider the simpler single target problem. In Euclidean case, the single target tracking problem in cluttered environment can be solved by applying the probabilistic data association filter or integrated probabilistic data association filter which also takes into account the disappearing and reappearing of the target. The PDA based tracking methods usually rely on some kind of Kalman filter that is responsible for estimation of the kinematic state of the target. Recall the expression for the association probabilities of PDAF given by eqs. (3.61) and (3.62)

$$\beta_k^0 = \frac{1 - p_d p_g}{1 - p_d p_g + \sum_{j=1}^{m_k} \lambda^{-1} p_d p_g g\left(y_{k,i} \,|\, \hat{x}_k\right)},$$

$$\beta_k^i = \frac{\lambda^{-1} p_d p_g g\left(y_{k,i} \,|\, \hat{x}_k\right)}{1 - p_d p_g + \sum_{j=1}^{m_k} \lambda^{-1} p_d p_g g\left(y_{k,i} \,|\, \hat{x}_k\right)}.$$

The likelihood $g\left(y_{k,i} \,|\, \hat{x}_k\right)$ in eqs. (3.61) and (3.62) is in Euclidean case given by normal distribution. Since we want to extend the PDAF to RMs by applying the Riemannian manifold EKF and UKF it is natural to use the concentrated Gaussian distribution to calculate the prior likelihood. Hence, the likelihood is given by

$$g\left(y_{k,i} \,|\, \hat{x}_k\right) = \mathcal{N}_{\mathcal{M}}(y_{k,i} \mid \hat{y}_k, \hat{S}_k), \tag{6.1}$$

where $\hat{y}_k$ is the predicted measurement at time k, while $\hat{S}_k$ is its covariance.

6.2.1 *Prediction step*

Prediction of the Riemannain manifold probabilistic data association filter (RM-PDAF) is simply given by the prediction step of the underlying estimator on Riemannian manifold. In the case of RM-UKF the Riemannian manifold unscented transform is applied

Furthermore, in case of IPDAF it is also necessary to calculate the existence probability of the target at new time step. This is done in the same way as in Euclidean case, by multiplying the existence probability from the previous step with the survival probability p_s which is assumed to be known and constant, i.e.

$$p_{k|k-1}\left(\mathcal{H} \,|\, Y_{1:k-1}\right) = p_S \, p_{k-1}\left(\mathcal{H} \,|\, Y_{1:k-1}\right), \tag{6.2}$$

where $\mathcal{H}$ denotes the hypothesis that the target exists and $Y_{1:k-1}$ denotes the measurement history (the set of all collected measurements from step 1 to $k-1$.

6.2.2 *Measurement gating*

To reduce the complexity of the data association step, measurements are gated using the generalized χ^2 test in the tangent space of the prediction $T_{\hat{y}_k^-}\mathcal{N}$

$$\mathrm{Log}_{\hat{y}_k^-} y_{k,i}{}^T \, \hat{S}_k^{-1} \, \mathrm{Log}_{\hat{y}_k^-} y_{k,i} \leq \chi_n^2(p_g), \tag{6.3}$$

where χ_n^2 is the quantile function of n-dimensional χ^2 distribution, and the p_g is the gating probability. Given that the detection $y_{k,i}$ is validated, the likelihood of $y_{k,i}$ given the predicted detection $\hat{y}_k$ is $g_i = p_g^{-1}\mathcal{N}_{\mathcal{M}}(y_{k,i} \mid \hat{y}_k, \hat{S}_k)$, where p_g is gating probability.

6.2.3 *Update step*

Given the predicted target states $\hat{x}_{k|k-1}$ and its existence probability $p_{k|k-1}(\mathcal{H} \mid Y_{1:k-1})$, the posterior association probabilities $\beta_{i,k}$ can be calculated by eqs. (3.61) and (3.62). The innovation of i-th measurement is given by

$$v_{i,k} = \mathrm{Log}_{\hat{y}_k} y_{i,k} \tag{6.4}$$

and the total weighted innovation is

$$v_k = \sum_{i=1}^{m_k} \beta_{i,k} v_{i,k}. \tag{6.5}$$

The update of the state of the target is than given by the update equation of the underlying Riemannian manifold Kalman filter

$$\hat{x}_k^+ = \mathrm{Exp}_{\hat{x}_k^-} K_k v_k, \tag{6.6}$$

where K_k is the Kalman gain.

Suppose that the updated covariance $\bar{P}_k$ of the underlying filter is given by (5.19) or (5.33), depending on whether the extended or unscented KF is used. $\bar{P}_k$ is exact when there is exactly one measurement, however, when there are multiple measurements, then the posterior covariance depends also on the number of measurements and their spread around the $\hat{y}_k$ [2]. This dependency can be expressed using the matrix

$$T_k = \sum_{i=1}^{m_k} \beta_{i,k} \, v_{i,k} \, v_{i,k}^T - v_k \, v_k^T. \tag{6.7}$$

Hence, final update equation is obtained by adding the term $K_k \, T_k \, K_k^T$ to $\bar{P}_k$, i.e.

$$\hat{P}_k^\dagger = \bar{P}_k + K_k \, T_k \, K_k^T. \tag{6.8}$$

Finally, all of the previous calculations are conducted in the tangent space of $\hat{x}_{j,k}^-$, and therefore, $\hat{P}_{j,k}^\dagger$ must be parallelly transported to the tangent space at the updated state $\hat{x}_k^+$ [32]. Hence, the updated covariance matrix of the RM-PDAF is

$$\hat{P}_k^+ = \mathrm{P}_{\hat{x}_k^- \to \hat{x}_k^+}\left(\hat{P}_k^\dagger\right). \tag{6.9}$$

6.3 JPDAF on Riemannian manifolds

6.3.1 *Prediction step*

The extension of the prediction step of the Riemannian manifold PDAF and IPDAF to the multi-target case is trivial. The state of each target is predicted independently of each other according to the motion model of the system. Furthermore, in case of the JIPDAF, the prediction of the existence probability of each target is also calculated independently for each target according to equation (6.2).

6.3.2 *Measurement gating*

Similarly as in single-target case, the computational burden of the JPDAF and JIPDAF can be significantly reduced by measurement gating. The gating is achieved by calculating the generalized Mahalanobis distance for each target-measurement pair. Thus, the validation matrix V is formed such that its (i, j)-th component is one if the relation

$$\mathrm{Log}_{\hat{y}^-_{k,j}} y_{k,i}{}^T \, \hat{S}_k^{-1} \, \mathrm{Log}_{\hat{y}^-_{k,j}} y_{k,i} \leq \chi^2_n(p_g) \tag{6.10}$$

is fulfilled and zero otherwise. The complexity of the RM-JPDAF can be further reduced by clustering the set of targets and measurements into independent clusters.

6.3.3 *Update step*

Given the predicted target states $\hat{x}_{k|k-1,j}$ and its existence probabilities $p_{k|k-1}\left(\mathcal{H}_j \,\middle|\, Y_{1:k-1}\right)$, where j denotes j-th target, the posterior association probabilities $\beta_{i,j,k}$ are calculated as described in the section 3.3.3. The innovation of i-th measurement to j-th target is given by

$$\nu_{i,j,k} = \mathrm{Log}_{\hat{y}_{j,k}} y_{i,k}. \tag{6.11}$$

Then, the total weighted innovation of all measurements to j-th target, given posterior association probabilities $\beta_{i,j,k}$, is

$$\nu_{j,k} = \sum_{i=1}^{m_k} \beta_{i,j,k} \, \nu_{i,j,k}. \tag{6.12}$$

Given the weighted innovation (6.12) and the Kalman gain $K_{j,k}$ of j-th target calculated by the Riemannian manifold extended or unscented Kalamn filter, j-th target is updated as follows

$$\hat{x}^+_{j,k} = \mathrm{Exp}_{\hat{x}^-_{j,k}} K_{j,k} \, \nu_{j,k}. \tag{6.13}$$

Similarly as in the update step of the RM-PDAF, the updated covariance matrix of the j-th target is obtained by adding the term which captures the spread of the detections to the covariance matrix $\bar{P}_{j,k}$ obtained by updating according to the underlying filter

$$\hat{P}^\dagger_{j,k} = \bar{P}_{j,k} + K_{j,k} \, T_{j,k} \, K^T_{j,k}, \tag{6.14}$$

where

$$T_{j,k} = \sum_{i=1}^{m_k} \beta_{i,j,k}\, \nu_{i,j,k}\, \nu_{i,j,k}^T - \nu_{j,k}\, \nu_{j,k}^T. \tag{6.15}$$

Finally, the updated covariance matrix $\hat{P}_{j,k}^{\dagger}$ of j-th target must be transported to the tangent space at the updated state $\hat{x}_{j,k}^{+}$

$$\hat{P}_{j,k}^{+} = \mathrm{P}_{\hat{x}_{j,k}^{-} \to \hat{x}_{j,k}^{+}} \left(\hat{P}_{j,k}^{\dagger} \right). \tag{6.16}$$

6.4 Results

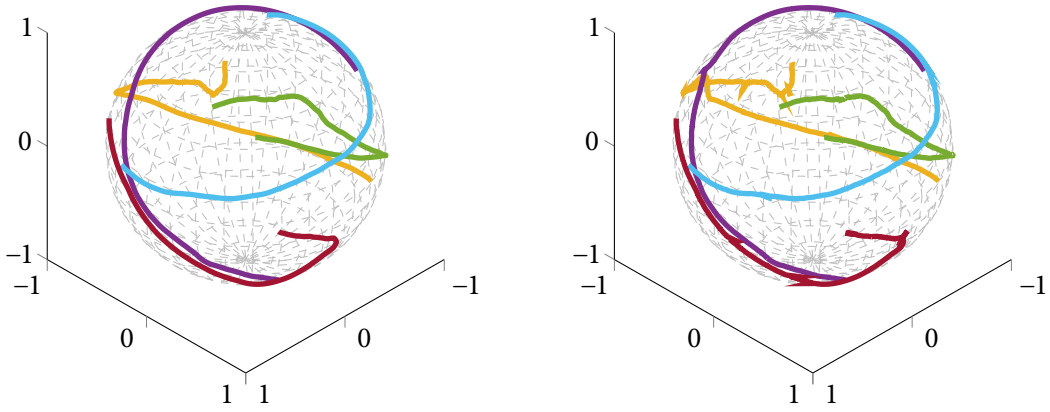


Figure 6.1: MOT on the sphere. Ground truth trajectories (left) and the Riemannian manifold JIPDA tracking result (right).

6.4.1 *Tracking on the unit sphere*

The experiment consisted of 100 Monte Carlo runs. The duration of each run was 200 steps with the sampling time $\Delta T = 0.05\,\mathrm{s}$. In each run, there were 5 randomly generated GT trajectories. Each trajectory was generated by simulating Brownian motion on the unit sphere. The Brownian motion was driven with the concentrated Gaussian noise with the standard deviation $\sigma_q = 0.025$. The measurements were simulated by adding the concentrated Gaussian noise with standard deviation $\sigma_r = 0.001$ to targets. Furthermore, to simulate the missed detections, generated measurements were randomly discarded with the probability 0.02. Then, the set of detections was cluttered with the set of spurious detections which was sampled from the Poisson distribution with the expected number of false alarms $\lambda = 5$. The spatial distribution of false alarms was uniform over unit sphere.

Three different MTT methods were compared in this experiment. First one uses the standard KF to model the motion of the targets. In order to ensure that estimates of this method are constrained on the unit sphere, the state of the filter is projected on the sphere after each prediction and update step. The second method used the von Mises-Fisher filter, and finally, the third method used the Riemannian manifold extended Kalman filter with the first-order motion model on the unit sphere. All three methods used JIPDAF to achieve multi-target tracking.

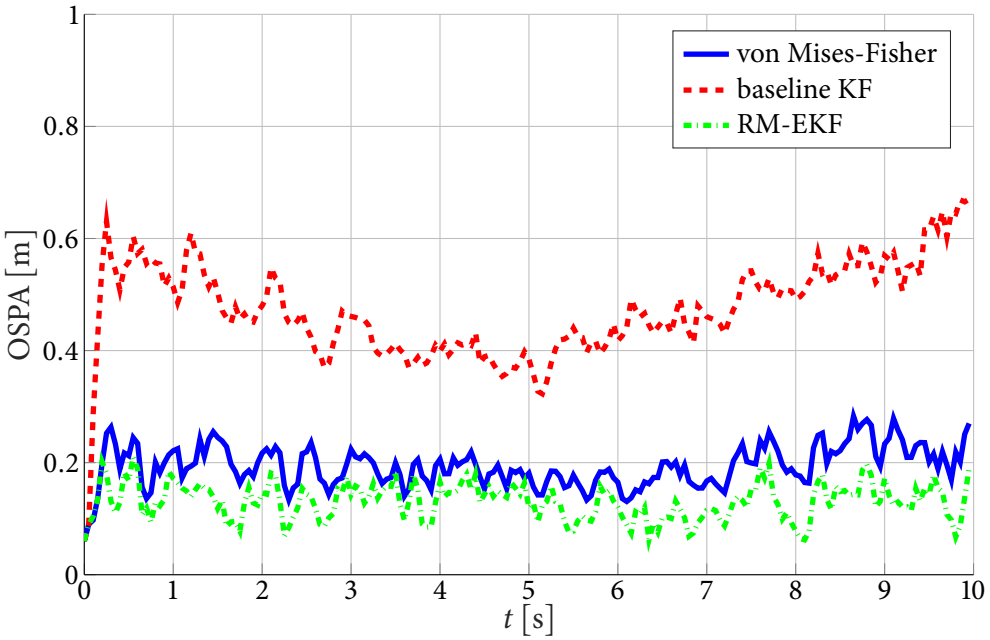


Figure 6.2: Evaluation of the multi-target tracking on the unit sphere. Figure shows OSPA error of three different multi-target tracking methods: JIPDAF with baseline KF model (red), with von Mises-Fisher filter (blue) and with Riemannian manifold EKF (green).

Parameters of the JIPDAF were as follows in all three cases. Survival, detection and gating probability were $p_s = 0.99$, $p_d = 0.98$ and $p_g = 0.95$. The expected number of false alarms was set to $\lambda = 5$. New targets were initialised from detections that satisfy the following criterion $\sum_{j=1}^{n_k} \beta_{i,j,k} \leq 0.5$ and their existence probability was initialised to 0.5. Targets were confirmed when their existence probability exceeded threshold 0.95 and terminated if it falls below 0.003.

Parameters of the baseline KF were as follows. The covariance matrix of the process noise was $Q = 0.01^2 \cdot \Delta T \cdot I_3$, while the covariance matrix of the measurement noise was $R = 0.001^2 \cdot I_3$. The initial covariance of each target was set to $0.01^2 \cdot I_3$. Parameters of the vMF filter were following. The diffusion coefficient was set to $\kappa_d = 50$, the concentration of the measurement distribution was $\kappa_o = 1000$ and the initial concentration of new targets was $\kappa_0 = 100$. The covariance matrices of the process and measurement noise of the RM-EKF were set to $Q = 0.025^2 \cdot \Delta T \cdot I_2$ and $R = 0.001^2 \cdot I_2$. The initial covariance matrix of the new targets was $P_0 = \mathrm{diag}(0.01, 0.01, 0.3, 0.3)$.

The metric used for the evaluation was the OSPA metric with parameters $c = 1$ and $p = 2$. Figure 6.2 shows the OSPA error averaged over all Monte Carlo runs. The JIPDAF with the RM-EKF with the first-order motion model achieved best result with average OSPA error of 0.1109, while the JIPDAF based on vMF filter achieved average OSPA error of 0.1775. The average error of the JIPDAF with the baseline KF was 0.4101. One example of the Monte Carlo simulation is shown on the figure 6.3, where we can see ground truth trajectories, simulated detections and estimates obtained by the JIPDAF with RM-EKF.

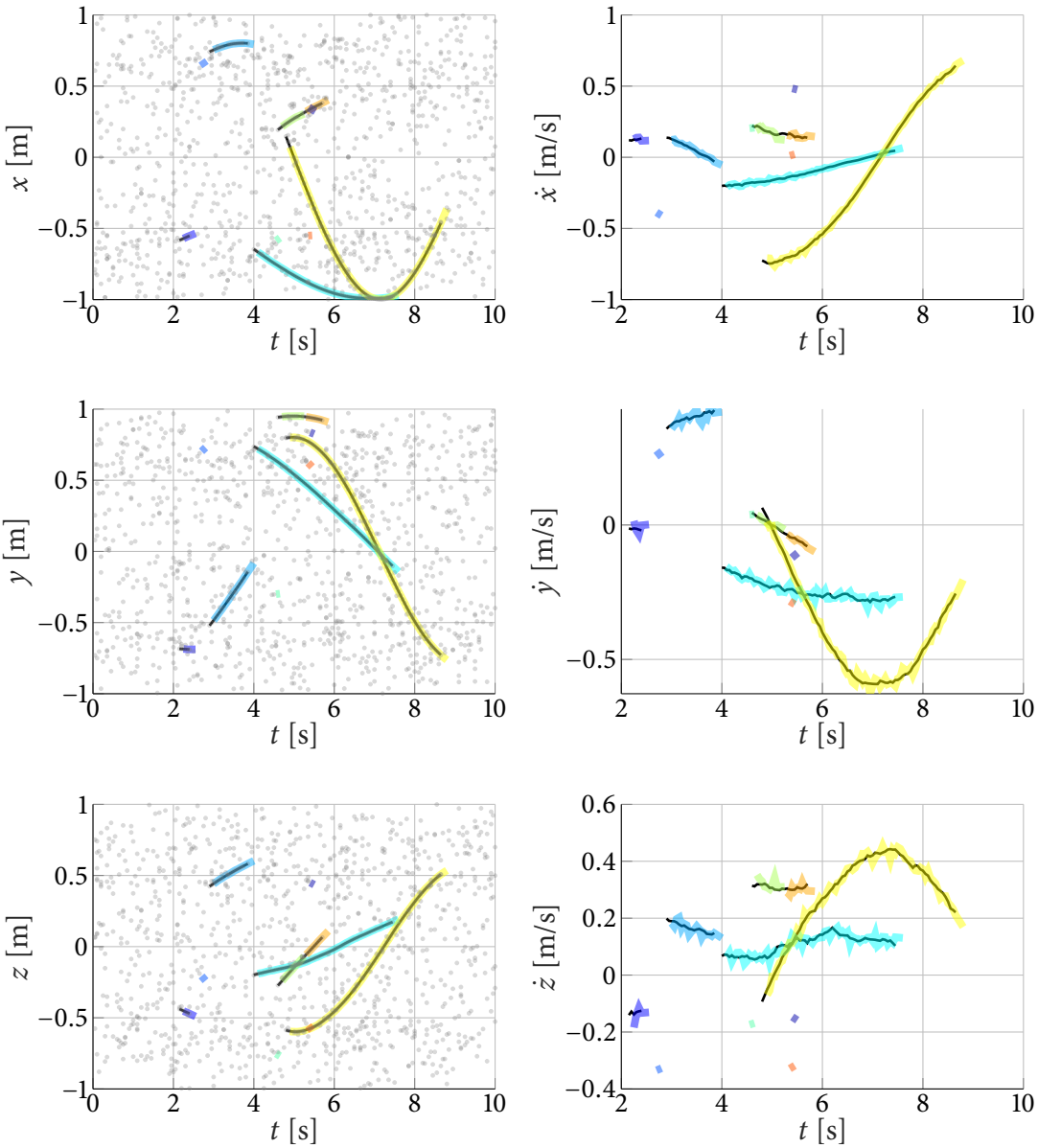


Figure 6.3: The results of the multi-target tracking on the unit sphere with the constant velocity model. Left column shows the position of targets, while the right column shows the velocity. Black lines show ground truth trajectories, gray dots show simulated measurements including clutter and colored lines represent estimates of the JIPDAF with the RM extended Kalman filter. Each color corresponds to the unique ID of the targets estimated by the filter.

6.5 Summary

This chapter propoes a novel multi-target tracking method for targets moving on the Riemannian manifold. The tracking method was developed starting from the extended and unscented KFs for systems constrained on Riemannian manifold. The update methods of those filters were adapted so that they can be used within the probabilistic data association framework. Additionally, the gating method of the classical probabilistic data association methods was adapted to use the generalized Mahalanobis distance on Riemannian manifold. The probabilistic data association filter on Riemannian manifolds were presented in case of single-target tracking in clutter and joint integrated probabilistic data association filter for multi-target tracking on manifolds. The Riemannain manifold JIPDAF was evaluated on simulated data. The conducted Monte Carlo simulations showed promising results.

Chapter 7

Conclusion and Outlook

"Done is better than perfect."

– Sheryl Sandberg

Due to the recent significant emergence of autonomous vehicles, there is a need for ever more precise and robust multiple object tracking methods to ensure the safety of other traffic participants. Autonomous vehicles are equipped with a plethora of sensors such as radars, lidars, cameras, etc. However, tracking of surrounding moving objects is not a trivial task due to the many problems such as spurious detections, missed detections due to occlusions, ego motion, detection-to-track assignment uncertainty and many others. In this thesis, we decided to tackle some of those problems.

The thesis started with an overview of the state-of-the-art state estimation and methods. Bayesian estimation paradigm is presented and some special cases of Bayesian estimators are further scrutinized. Under linear Gaussian approximation, Bayesian estimation recursion yields a closed-form solution, the Kalman filter. However, the Kalman filter is also very often used in nonlinear state estimation. Two common nonlinear extensions of the Kalman filter are extended Kalman filter and unscented Kalman filter. Former uses the linearisation of the nonlinear state transition model in order to propagate the uncertainty of the filter, while the later applies the unscented transform to propagate the distribution of the state through nonlinear state transition models. Another example of the Bayes filter emerges when dealing with direction-only data. Thus, assuming the von Mises-Fisher distribution yields the closed-form solution in this case, the von Mises-Fisher filter. Further, the few of the state-of-the-art multi-target tracking methods were studied within this thesis. It is customary to divide the tracking methods in three different groups: probabilistic data association approaches, multiple hypothesis tracking and random finite set approaches. In this thesis, the emphasis was on the probabilistic data association methods.

In the continuation of the thesis we tackle the problem of pedestrian detection and tracking. The Mask R-CNN was selected as the deep convolutional model appropriate for the task of pedestrian detection. The detector was pre-trained and then fine-tuned on various datasets in order to avoid over-fitting of the model. Additional deep model based on the ResNet18 architecture was used to extract the deep features. The deep features were extracted form the last layer of the second residual block of the ResNet18 model. The purpose of deep features was to use them in order to solve the detection-to-target assignment ambiguity. Furthermore, the joint integrated probabilistic data association filter was adapted to achieve the multi-target tracking of pedestrians from detections provided

by the Mask R-CNN model. By using only detections provided by Mask R-CNN detector, we achieved #1 result on the 3DMOT2015 dataset of the public MOT Challenge benchmark. However, on 2DMOT2015 dataset, which contains challenging scenes with moving camera, this approach did not yield good results. Hence, we implemented another multi-target tracker, the global nearest neighbour method which used only deep features to achieve data association. This appearance-based pedestrian tracking method showed better results than kinematic-only tracking on scenes with moving camera, however on the scenes with static camera it did not outperform kinematic-only tracker. This motivated us to try to incorporate the deep features together with the kinematic cues in the probabilistic data association filtering. However, it proved to be challenging due to highly nonlinear geometry of the underlying space of deep embeddings.

To this end, we concentrated on the state estimation on curved, but smooth spaces. Such spaces can be endowed with the metric which varies smoothly along the space. Such smooth and curved spaces together with the smooth metric are called Riemannian manifolds. Therefore, we focused on the estimation methods for the systems that evolve on Riemannian manifolds. There are a several Kalman filter extensions to the Riemannian manifolds, the extended and unscented Riemannian manifold Kalman filters. However, when the state of the system changes rapidly with time, as in tracking applications, simple motion models are not adequate enough. Higher-order motion models, such as constant velocity model, can be used instead. Although, it is in some cases possible to develop the Riemannian manifold extended Kalman filter with the constant velocity motion model, the calculation of Jacobians of the constant velocity model is tedious. Thus, we proposed the extension of the Riemannian manifold unscented Kalman filter with the constant velocity model in order to avoid the calculation of Jacobians. To extend the Riemannian manifold unscented Kalman filter, we extend its state together with its velocity. Such extended state forms a point on the tangent bundle of the Riemannian manifold, hence we use the geometry of the tangent bundle to implement the unscented Kalman filter for Riemannian manifolds with the constant velocity model. The proposed method showed promising results, but there still remain some challenges and there is still a room for further research.

Thereafter, we took on the problem of multi-target tracking on Riemannian manifolds. We started from the Riemannian manifold extended and unscented Kalman filters. The update equations of extended and unscented Riemannian manifold Kalman filter were adapted in order to be used within the probabilistic data association paradigm. Furthermore, we utilised the generalised Mahalanobis distance on the tangent space of the manifold to achieve the measurement gating. Therein, we presented both the single-target probabilistic data association filter and the multi-target joint integrated probabilistic data association filter for the targets that move on the Riemannian manifold. Several simulation experiments were conducted to evaluate proposed tracking method on a few examples of Riemannian manifolds. The experiments showed promising results, but there are still a lot of challenges interesting for further research. For example, this tracking method may be applied to the pedestrian tracking to fuse the appearance cues of the deep features together with the kinematic cues of the classical tracking methods. The target's state space in that case can be modelled as the product space of the embeddings and kinematic state-space of the targets. The model for extraction of deep features can be implemented in such a way that

those features lie on some Riemannian manifold, for example unit hypersphere, while the kinematic state of the target lies in the Euclidean space which is itself a Riemannian manifold, and finally, a product space of two Riemannian manifolds is a Riemannian manifold as well.

Bibliography

[1] B. ngu Vo, M. Mallick, Y. Bar-shalom, S. Coraluppi, R. Osborne, R. Mahler, and B. tuong Vo, "Multitarget tracking", *Wiley Encyclopedia of Electrical and Electronics Engineering*, pp. 1–15, Sep. 2015.

[2] Y. Bar-Shalom and E Tse, "Tracking in a cluttered environment with probabilistic data association", *Automatica*, vol. 11, pp. 451–460, 1975.

[3] T. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association", *IEEE Journal of Oceanic Engineering*, vol. 8, pp. 173–184, 3 Jul. 1983.

[4] D. Mušicki and R. Evans, "Joint integrated probabilistic data association - jipda", vol. 2, Int. Soc. Inf. Fusion, 2002, pp. 1120–1125.

[5] S. H. Rezatofighi, A. Milan, Z. Zhang, Q. Shi, A. Dick, and I. Reid, "Joint probabilistic data association revisited", IEEE, Dec. 2015, pp. 3047–3055.

[6] D. Reid, "An algorithm for tracking multiple targets", *IEEE Transactions on Automatic Control*, vol. 24, pp. 843–854, 6 1979.

[7] C.-Y. Chong, S. Mori, and D. B. Reid, "Forty years of multiple hypothesis tracking - a review of key developments", IEEE, Jul. 2018, pp. 452–459.

[8] B.-N. Vo and W.-K. Ma, "The gaussian mixture probability hypothesis density filter", *IEEE Transactions on Signal Processing*, vol. 54, pp. 4091–4104, 11 2006.

[9] B.-T. Vo, B. ngu Vo, and A. Cantoni, "Analytic implementations of the cardinalized probability hypothesis density filter", *IEEE Transactions on Signal Processing*, vol. 55, pp. 3553–3567, 7 Jul. 2007.

[10] B. tuong Ngu Vo, B. tuong Ngu Vo, and A. Cantoni, "On multi-bernoulli approximations to the bayes multi-target filter", *Fusion2007*, 2007.

[11] S. Reuter, B. T. Vo, B. N. Vo, and K. Dietmayer, "The labeled multi-bernoulli filter", *IEEE Transactions on Signal Processing*, vol. 62, pp. 3246–3260, 12 2014.

[12] R. P. Mahler, *Statistical Multisource-Multitarget Information Fusion*. Artech House, INC., 2007.

[13] T. D. Barfoot, *State Estimation for Robotics*. Cambridge University Press, 2017, pp. 1–368.

[14] S. Calinon and N. Jaquier, "Gaussians on riemannian manifolds for robot learning and adaptive control", Sep. 2019.

[15] G. Bourmaud, R. Megret, A. Giremus, and Y. Berthoumieu, "Discrete extended kalman filter on lie groups", 2013.

[16] G. Bourmaud, R. Mégret, M. Arnaudon, and A. Giremus, "Continuous-discrete extended kalman filter on matrix lie groups using concentrated gaussian distributions", *Journal of Mathematical Imaging and Vision*, vol. 51, pp. 209–228, 1 Jan. 2015.

[17] V. Joukov, I. Petrović, and J. Ćesić, "Full body human motion estimation on lie groups using 3d marker position measurements", *2016 IEEE-RAS International Conference on Humanoid Robots (Humanoids 2016)*, 2016.

[18] J. Cesic, I. Markovic, and I. Petrovic, "Moving object tracking employing rigid body motion on matrix lie groups", Aug. 2016.

[19] J. Cesic, I. Markovic, and I. Petrovic, "Mixture reduction on matrix lie groups", *IEEE Signal Processing Letters*, vol. 24, pp. 1719–1723, 11 Nov. 2017.

[20] K. V. Mardia and P. E. Jupp, *Directional Statistics*, K. V. Mardia and P. E. Jupp, Eds. John Wiley & Sons, Inc., Jan. 1999, pp. 1–432.

[21] I. Marković, J. Ćesić, and I. Petrović, "Von mises mixture phd filter", *IEEE Signal Processing Letters*, vol. 22, pp. 2229–2233, 12 2015.

[22] G. Kurz, M. Baum, and U. D. Hanebeck, "Real-time kernel-based multiple target tracking for robotic beating heart surgery", IEEE, Jun. 2014, pp. 201–204.

[23] I. Gilitschenski, G. Kurz, S. J. Julier, and U. D. Hanebeck, "Unscented orientation estimation based on the bingham distribution", *IEEE Transactions on Automatic Control*, vol. 61, pp. 172–177, 1 Jan. 2016.

[24] I. Marković, M. Bukal, J. Ćesić, and I. Petrović, "Multitarget tracking with the von mises-fisher filter and probabilistic data association", *Journal of Advances in Information Fusion*, pp. 1557–6418, 2016.

[25] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection", vol. 1, IEEE, 2005, pp. 886–893.

[26] O. Tuzel, F. Porikli, and P. Meer, "Human detection via classification on riemannian manifolds", IEEE, Jun. 2007, pp. 1–8.

[27] A. Romero, M. Gouiffés, and L. Lacassagne, "Covariance descriptor multiple object tracking and re-identification with colorspace evaluation", vol. 7729 LNCS, 2013, pp. 400–411.

[28] L. Chen, H. Ai, C. Shang, Z. Zhuang, and B. Bai, "Online multi-object tracking with convolutional neural networks", vol. 2017-Septe, IEEE, Sep. 2017, pp. 645–649.

[29] Q. Chu, W. Ouyang, H. Li, X. Wang, B. Liu, and N. Yu, "Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism", vol. 2017-Octob, IEEE, Oct. 2017, pp. 4846–4855.

[30] B. Bicanic, M. Orsic, I. Markovic, S. Segvic, and I. Petrovic, "Pedestrian tracking by probabilistic data association and correspondence embeddings", Jul. 2019.

[31] Y. Wu, B. Wu, J. Liu, and H. Lu, "Probabilistic tracking on riemannian manifolds", IEEE, Dec. 2008, pp. 1–4.

[32] S. Hauberg, F. Lauze, and K. S. Pedersen, "Unscented kalman filtering on riemannian manifolds", *Journal of Mathematical Imaging and Vision*, vol. 46, pp. 103–120, 1 May 2013.

[33] S. Leonardos, X. Zhou, and K. Daniilidis, "Articulated motion estimation from a monocular image sequence using spherical tangent bundles", vol. 2016-June, IEEE, May 2016, pp. 587–593.

[34] H. M. T. Menegaz, J. Y. Ishihara, and H. T. M. Kussaba, "Unscented kalman filters for riemannian state-space systems", *IEEE Transactions on Automatic Control*, vol. 64, pp. 1487–1502, 4 Apr. 2019.

[35] J. Jost, *Riemannian Geometry and Geometric Analysis*. Springer Berlin Heidelberg, 2011.

[36] S. Gudmundsson and E. Kappos, "On the geometry of tangent bundles", *Expositiones Mathematicae*, vol. 20, pp. 1–41, 1 2002.

[37] S. Sasaki, "On the differential geometry of tangent bundles of riemannian manifolds", *Tohoku Mathematical Journal*, vol. 10, pp. 338–354, 3 1958.

[38] P. Muralidharan and P. T. Fletcher, "Sasaki metrics for analysis of longitudinal data on manifolds", IEEE, Jun. 2012, pp. 1027–1034.

[39] S. Janson, "Riemannian geometry: some examples, including map projections", 2015, pp. 1–122.

[40] M. Moakher and M. Zéraï, "The riemannian geometry of the space of positive-definite matrices and its application to the regularization of positive-definite matrix-valued data", *Journal of Mathematical Imaging and Vision*, vol. 40, pp. 171–187, 2 Jun. 2011.

[41] A. Dolcetti and D. Pertici, "Some differential properties of $GL_n(R)$ with the trace metric", Dec. 2014.

[42] A. Dolcetti and D. Pertici, "Differential properties of spaces of symmetric real matrices", Jul. 2018.

[43] S. Reuter, "Multi-object tracking using random finite sets", 2014.

[44] B.-T. Vo, "Random finite sets in multi-object filtering", 2008, p. 254.

[45] A. J. Haug, *Bayesian Estimation and Tracking*. John Wiley & Sons, 2012.

[46] R. E. Kalman, "A new approach to linear filtering and prediction problems", *Journal of Basic Engineering*, vol. 82, p. 35, 1 1960.

[47] Y. Ho and R. Lee, "A bayesian approach to problems in stochastic estimation and control", *IEEE Transactions on Automatic Control*, vol. 9, pp. 333–339, 4 1964.

[48] S. J. Julier and J. K. Uhlmann, "New extension of the kalman filter to nonlinear systems", I. Kadar, Ed., vol. 3068, Jul. 1997, p. 182.

[49] H. M. T. Menegaz, J. Y. Ishihara, G. A. Borges, and A. N. Vargas, "A systematization of the unscented kalman filter theory", *IEEE Transactions on Automatic Control*, vol. 60, pp. 2583–2598, 10 Oct. 2015.

[50] S. Challa, M. R. Morelande, D. Mušicki, and R. J. Evans, *Fundamentals of Object Tracking - Knovel*. 2011, p. 375.

[51] H. W. Kuhn, "The hungarian method for the assignment problem", *Naval Research Logistics Quarterly*, vol. 2, pp. 83–97, 1-2 Mar. 1955.

[52] D. Mušicki, R. Evans, and S. Stankovic, "Integrated probabilistic data association", *Transaction on Automatic Control*, vol. 39, pp. 1237–1241, 6 1994.

[53] C. Otto, W. Gerber, F. P. Leon, and J. Wirnitzer, "A joint integrated probabilistic data association filter for pedestrian tracking across blind regions using monocular camera and radar", IEEE, Jun. 2012, pp. 636–641.

[54] S. S. Blackman, "Multiple hypothesis tracking for multiple target tracking", *IEEE Aerospace and Electronic Systems Magazine*, vol. 19, pp. 5–18, 1 2004.

[55] A. Frank, P. Smyth, and A. Ihler, "A graphical model representation of the track-oriented multiple hypothesis tracker", IEEE, Aug. 2012, pp. 768–771.

[56] S. S. Blackman, D. Tyler, and C. D. Pickle, "Application of mht to through the wall radar tracking", IEEE, Mar. 2013, pp. 1–9.

[57] T. Sathyan, T.-J. Chin, S. Arulampalam, and D. Suter, "A multiple hypothesis tracker for multitarget tracking with multiple simultaneous measurements", *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, pp. 448–460, 3 Jun. 2013.

[58] S. He, H.-S. Shin, and A. Tsourdos, "Track-oriented multiple hypothesis tracking based on tabu search and gibbs sampling", *IEEE Sensors Journal*, vol. 18, pp. 328–339, 1 Jan. 2018.

[59] S. Coraluppi and C. Carthel, "Track management in multiple-hypothesis tracking.pdf", IEEE, 2018, pp. 11–15.

[60] S. Coraluppi and C. Carthel, "Generalizations to the track-oriented mht recursion", IEEE, 2015, pp. 346–350.

[61] S. Coraluppi, C. Carthel, and M. Guerriero, "Hypothesis structure in enhanced multiple-hypothesis tracking", IEEE, Nov. 2013, pp. 1738–1742.

[62] R. L. Streit and T. E. Luginbuhl, "Probabilistic multi-hypothesis tracking.pdf", NAVAL UNDERWATER SYSTEMS CENTER NEWPORT, RI, 1995, p. 52.

[63] M. Krieg and D. Gray, "Multi-sensor, probabilistic multi-hypothesis tracking", IEEE, 2002, pp. 153–158.

[64] S. Schoenecker, P. Willett, and Y. Bar-Shalom, "The ml-pmht multistatic tracker for sharply maneuvering targets", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, pp. 2235–2249, 4 Oct. 2013.

[65] W. yong Choi and S. mog Hong, "Estimation of target states and signal strength in the probabilistic multiple hypothesis tracking framework", *IET Radar, Sonar & Navigation*, vol. 8, pp. 565–574, 6 Jul. 2014.

[66] R. Mahler, "Multitarget bayes filtering via first-order multitarget moments", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, pp. 1152–1178, 4 Oct. 2003.

[67] R. P. S. Mahler, B.-T. Vo, and B.-N. Vo, "Cphd filtering with unknown clutter rate and detection profile", *IEEE Transactions on Signal Processing*, vol. 59, pp. 3497–3513, 8 Aug. 2011.

[68] M. Ulmke, O. Erdinc, and P. Willett, "Gaussian mixture cardinalized phd filter for ground moving target tracking", IEEE, Jul. 2007, pp. 1–8.

[69] M. Lundgren, L. Svensson, and L. Hammarstrand, "A cphd filter for tracking with spawning models", *IEEE Journal on Selected Topics in Signal Processing*, vol. 7, pp. 496–507, 3 2013.

[70] M. Beard, B. T. Vo, B. N. Vo, and S. Arulampalam, "A partially uniform target birth model for gaussian mixture phd/cphd filtering", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 49, pp. 2835–2844, 4 2013.

[71] D. Macagnano and G. T. F. de Abreu, "Gating for multitarget tracking with the gaussian mixture phd and cphd filters", IEEE, Apr. 2011, pp. 149–154.

[72] B.-N. Vo, S. Singh, and A. Doucet, "Sequential monte carlo methods for bayesian multi-target filtering with random finite sets", *IEEE Trans. Aerospace and Electronic Systems*, vol. 41, pp. 1224–1245, 4 2005.

[73] B.-T. Vo, B.-N. Vo, and A. Cantoni, "The cardinality balanced multi-target multi-bernoulli filter and its implementations", *IEEE Transactions on Signal Processing*, vol. 57, pp. 409–423, 2 Feb. 2009.

[74] B.-T. Vo, B.-N. Vo, R. Hoseinnezhad, and R. P. Mahler, "Robust multi-bernoulli filtering", *IEEE Journal on Selected Topics in Signal Processing*, vol. 7, pp. 399–409, 3 2013.

[75] B.-T. Vo and B. ngu Vo, "Labeled random finite sets and multi-object conjugate priors", *IEEE Transactions on Signal Processing*, vol. 61, pp. 3460–3475, 13 Jul. 2013.

[76] B.-N. Vo, B.-T. Vo, and D. Phung, "Labeled random finite sets and the bayes multi-target tracking filter", *IEEE Transactions on Signal Processing*, vol. 62, pp. 6554–6567, 24 Dec. 2014.

[77] B. N. Vo, B. T. Vo, and H. G. Hoang, "An efficient implementation of the generalized labeled multi-bernoulli filter", *IEEE Transactions on Signal Processing*, vol. 65, pp. 1975–1987, 8 2017.

[78] J. Olofsson, C. Veiback, and G. Hendeby, "Sea ice tracking with a spatially indexed labeled multi-bernoulli filter", IEEE, Jul. 2017, pp. 1–8.

[79] B.-T. Vo, B. ngu Vo, and A. Cantoni, "Bayesian filtering with random finite set observations", *IEEE Transactions on Signal Processing*, vol. 56, pp. 1313–1326, 4 Apr. 2008.

[80] R. Mahler, "Phd filters for nonstandard targets, i: extended targets", IEEE, 2009, pp. 915–921.

[81] M. Beard, S. Reuter, K. Granstrom, B.-T. Vo, B.-N. Vo, and A. Scheel, "Multiple extended target tracking with labeled random finite sets", *IEEE Transactions on Signal Processing*, vol. 64, pp. 1638–1653, 7 Apr. 2016.

[82] A. Scheel, S. Reuter, and K. Dietmayer, "Using separable likelihoods for laser-based vehicle tracking with a labeled multi-bernoulli filter", *19th International Conference on Information Fusion (FUSION)*, 2016.

[83] M. Beard, B.-T. Vo, and B.-N. Vo, "Multi-target tracking with merged measurements using labelled random finite sets", IEEE, 2014.

[84] K. Krishanth, X. Chen, R. Tharmarasa, T. Kirubarajan, and M. McDonald, "The social force phd filter for tracking pedestrians", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, pp. 1–1, 4 2017.

[85] R. P. Mahler, "Phd filters of higher order in target number", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, pp. 1523–1543, 4 2007.

[86] A. R. Runnalls, "Kullback-leibler approach to gaussian mixture reduction", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, pp. 989–999, 3 2007.

[87] M. F. Huber and U. D. Hanebeck, "Progressive gaussian mixture reduction", 2008, pp. 9–16.

[88] D. Schieferdecker and M. F. Huber, "Gaussian mixture reduction via clustering", *Proceedings of the 12th International Conference on Information Fusion (Fusion)*, pp. 1536–1543, 2009.

[89] D. F. Crouse, P. Willett, K. Pattipati, and L. Svensson, "A look at gaussian mixture reduction algorithms", IEEE, 2011, pp. 1 –8.

[90] D. Schuhmacher, B.-T. Vo, and B. ngu Vo, "A consistent metric for performance evaluation of multi-object filters", *IEEE Transactions on Signal Processing*, vol. 56, pp. 3447–3457, 8 Aug. 2008.

[91] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: the clear mot metrics", *EURASIP Journal on Image and Video Processing*, vol. 2008, pp. 1–10, 2008.

[92] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge", *International Journal of Computer Vision*, vol. 115, pp. 211–252, 3 Dec. 2015.

[93] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: common objects in context", 2014, pp. 740–755.

[94] S. Zhang, R. Benenson, M. Omran, J. Hosang, and B. Schiele, "How far are we from solving pedestrian detection?", IEEE, Jun. 2016, pp. 1259–1267.

[95] S. Zhang, R. Benenson, and B. Schiele, "Citypersons: a diverse dataset for pedestrian detection", IEEE, Jul. 2017, pp. 4457–4465.

[96] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a siamese time delay neural network", vol. 6, 1993.

[97] E. Hoffer and N. Ailon, "Deep metric learning using triplet network", 2015.

[98] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective", 2016.

[99] J. Wang, F. Zhou, S. Wen, X. Liu, and Y. Lin, "Deep metric learning with angular loss", IEEE, Oct. 2017, pp. 2612–2620.

[100] C. Song, Y. Huang, W. Ouyang, and L. Wang, "Mask-guided contrastive attention model for person re-identification", IEEE, Jun. 2018, pp. 1179–1188.

[101] I. R. Goodman, R. P. S. Mahler, and H. T. Nguyen, *Mathematics of Data Fusion*. Springer Netherlands, 1997.

[102] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler, "Motchallenge 2015: towards a benchmark for multi-target tracking", Apr. 2015.

[103] B. H. Wang, Y. Wang, K. Q. Weinberger, and M. Campbell, "Deep person re-identification for probabilistic data association in multiple pedestrian tracking", 2018.

[104] S. Särkkä, A. Vehtari, and J. Lampinen, "Rao-blackwellized particle filter for multiple target tracking", *Information Fusion*, vol. 8, pp. 2–15, 1 Jan. 2007.

[105] Y. Wang, L. Wang, Y. You, X. Zou, V. Chen, S. Li, G. Huang, B. Hariharan, and K. Q. Weinberger, "Resource aware person re-identification across multiple resolutions", 2018.

[106] T. Klinger, F. Rottensteiner, and C. Heipke, "Probabilistic multi-person tracking using dynamic bayes networks", *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. II-3/W5, pp. 435–442, 3W5 Aug. 2015.

[107] N. Wojke and D. Paulus, "Global data association for the probability hypothesis density filter using network flows", IEEE, May 2016, pp. 567–572.

[108] Y. min Song, K. Yoon, Y. chul Yoon, K. choong Yow, and M. Jeon, "Online multi-object tracking framework with the gmphd filter and occlusion group management", pp. 1–15, Jul. 2019.

[109] F. Yang, F. Li, Y. Wu, S. Sakti, and S. Nakamura, "Using panoramic videos for multi-person localization and tracking in a 3d panoramic coordinate", IEEE, May 2020, pp. 1863–1867.

[110] Z. Tang and J.-N. Hwang, "Moana: an online learned adaptive appearance model for robust multiple object tracking in 3d", *IEEE Access*, vol. 7, pp. 31 934–31 945, 2019.

[111] D. Stadler and J. Beyerer, "Multi-pedestrian tracking with clusters", IEEE, Nov. 2021, pp. 1–10.

[112] Y. Wang, K. Kitani, and X. Weng, "Joint object detection and multi-object tracking with graph neural networks", IEEE, May 2021, pp. 13 708–13 715.

[113] I. Papakis, A. Sarkar, and A. Karpatne, "Gcnnmatch: graph convolutional neural networks for multi-object tracking via sinkhorn normalization", Sep. 2020.

[114] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "Mot16: a benchmark for multi-object tracking", Mar. 2016.

[115] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask r-cnn", IEEE, Oct. 2017, pp. 2980–2988.

[116] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 1137–1149, 6 Jun. 2017.

[117] M. Cordts, M. Omran, S. Ramos, T. Scharwächter, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset", 2015.

[118] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", IEEE, Jun. 2016, pp. 770–778.

[119] M. Wang, H. Ji, X. Hu, and Y. Zhang, "Gaussian mixture particle flow probability hypothesis density filter", IEEE, 2017.

[120] D. P. Kingma and J. L. Ba, "Adam: a method for stochastic optimization", 2015.

[121] T. Klinger, F. Rottensteiner, and C. Heipke, "Probabilistic multi-person localisation and tracking in image sequences", *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 127, pp. 73–88, May 2017.

[122] M. de Feo, A. Graziano, R. Miglioli, and A. Farina, "Immjpda versus mht and kalman filter with nn correlation: performance comparison", *IEE Proceedings - Radar, Sonar and Navigation*, vol. 144, p. 49, 2 1997.

[123] D. Svensson, M. Ulmke, and L. Hammarstrand, "Multitarget sensor resolution model and joint probabilistic data association", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, pp. 3418–3434, 4 2012.

[124] H. Snoussi and A. Mohammad-Djafari, "Particle filtering on riemannian manifolds", vol. 872, AIP, 2006, pp. 219–226.

[125] G. Bourmaud, R. Mégret, A. Giremus, Y. Berthoumieu, F-Talence, and F-Talence, "Global motion estimation from relative measurements using iterated extended kalman filter on matrix lie groups", IEEE, 2014, pp. 3362–3366.

[126] V. Joukov, J. Ćesić, K. Westermann, I. Marković, I. Petrović, and D. Kulić, "Estimation and observability analysis of human motion on lie groups", *IEEE transactions on cybernetics*, 2020.

[127] F. Tompkins and P. J. Wolfe, "Bayesian filtering on the stiefel manifold", IEEE, Dec. 2007, pp. 261–264.

[128] V. Solo, "On nonlinear state estimation in a riemannian manifold", IEEE, Dec. 2009, pp. 8500–8505.

[129] T. Yang, P. G. Mehta, and S. P. Meyn, "Feedback particle filter", *IEEE Transactions on Automatic Control*, vol. 58, pp. 2465–2480, 10 Oct. 2013.

[130] C. Zhang, A. Taghvaei, and P. G. Mehta, "Feedback particle filter on riemannian manifolds and matrix lie groups", *IEEE Transactions on Automatic Control*, vol. 63, pp. 2465–2480, 8 Aug. 2018.

[131] J. H. Manton, "A primer on stochastic differential geometry for signal processing", *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, pp. 681–699, 4 Aug. 2013.

[132] I. Marković, M. Bukal, J. Ćesić, and I. Petrović, "Direction-only tracking of moving objects on the unit sphere via probabilistic data association", IEEE, 2014.

[133] J. Ćesić, I. Marković, I. Cvišić, and I. Petrović, "Radar and stereo vision fusion for multitarget tracking on the special euclidean group", *Robotics and Autonomous Systems*, vol. 83, pp. 338–348, Sep. 2016.

[134] S. Bak, D.-P. Chau, J. Badie, E. Corvee, F. Bremond, and M. Thonnat, "Multi-target tracking by discriminative analysis on riemannian manifold", IEEE, Sep. 2012, pp. 1605–1608.

Appendix A

Derivation of Jacobians

This appendix provides the calculations of Jacobians used in Riemannian manifold EKF with constant velocity motion model. The discrete constant velocity model on the Riemannian manifolds is given by

$$p_k = \mathrm{Exp}_{p_{k-1}}\left(\Delta T\, v_{k-1}\right), \tag{A.1}$$

$$v_k = \mathrm{P}_{p_{k-1}\to p_k}\left(v_{k-1} + w_{k-1}\right), \tag{A.2}$$

where w_{k-1} is a concentrated Gaussian process noise on the tangent space $T_{p_{k-1}}\mathcal{M}$ and ΔT is the sampling period. We denote these two equations as $(p_k, v_k) = f_{k-1}\left(p_{k-1}, v_{k-1}, w_{k-1}\right)$.

A.1 Sphere

Recall from section 2.6.1 that the exponential map on $\mathcal{S}^n_\rho$ is given by

$$\mathrm{Exp}_p v = p\cos\frac{\|v\|}{\rho} + \frac{v}{\|v\|}\rho\sin\frac{\|v\|}{\rho}, \tag{A.3}$$

while parallel transport can be expressed by

$$\mathrm{P}_{x\to y}(v) = v - \frac{\mathrm{Log}_x(y)^T v}{d(x,y)^2}\left[\mathrm{Log}_x(y) + \mathrm{Log}_y(x)\right]. \tag{A.4}$$

The above expression can be simplified by substituting expressions for logarithmic mapping of sphere (2.36) and distance function of sphere (2.34):

$$\mathrm{P}_{x\to y}(v) = v - \frac{\mathrm{Log}_x y^T v}{d^2_{\mathcal{S}^n_\rho}(x,y)}\left[\mathrm{Log}_x y + \mathrm{Log}_y x\right] \tag{A.5}$$

$$= v - \frac{\left(y - x\frac{x^T y}{\rho^2}\right)^T v}{\left\|y - x\frac{x^T y}{\rho^2}\right\|^2}\cdot\left(y - x\frac{x^T y}{\rho^2} + x - y\frac{x^T y}{\rho^2}\right) \tag{A.6}$$

$$= v - \frac{\left(y - x\frac{x^T y}{\rho^2}\right)^T \cdot v\cdot(x+y)\cdot\left(1 - \frac{x^T y}{\rho^2}\right)}{\rho^2 - \frac{(x^T y)^2}{\rho^2}} \tag{A.7}$$

$$= v - \frac{\left(y - x\frac{x^T y}{\rho^2}\right)^T \cdot v\cdot(x+y)\cdot\left(1 - \frac{x^T y}{\rho^2}\right)}{\rho^2\cdot\left(1 - \frac{x^T y}{\rho^2}\right)\cdot\left(1 + \frac{x^T y}{\rho^2}\right)} \tag{A.8}$$

$$= v - \frac{\left(y - x\frac{x^T y}{\rho^2}\right)^T \cdot v}{\rho^2 + x^T y} \cdot (x + y), \tag{A.9}$$

where we used the fact that

$$\left\| y - x\frac{x^T y}{\rho^2} \right\|^2 = \rho^2 - \frac{(x^T y)^2}{\rho^2} \tag{A.10}$$

for $x, y \in \mathcal{S}_\rho^n$

Let's now write the state transition model f_{k-1} explicitly for $\mathcal{S}_\rho^n$. Substituting expressions for exponential map and parallel transport of the sphere in eqs. (A.1) and (A.2) results in

$$p_k = p_{k-1}\cos\theta + v_{k-1}\frac{\Delta T}{\theta}\sin\theta, \tag{A.11}$$

$$v_k = v_{k-1} + w_{k-1} - \frac{\left(p_k - p_{k-1}\frac{p_{k-1}^T p_k}{\rho^2}\right)^T (v_{k-1} + w_{k-1})}{\rho^2 + p_{k-1}^T p_k}(p_{k-1} + p_k), \tag{A.12}$$

where $\theta = \frac{\Delta T \|v_{k-1}\|}{\rho}$. Now let's substitute p_k in (A.12) with (A.11). Note that

$$\begin{aligned} p_{k-1}^T p_k &= p_{k-1}^T \left(p_{k-1}\cos\theta + \frac{\Delta T}{\theta} v_{k-1}\sin\theta \right) \\ &= p_{k-1}^T p_{k-1}\cos\theta + \frac{\Delta T}{\theta} p_{k-1}^T v_{k-1}\sin\theta \\ &= \rho^2\cos\theta \end{aligned}$$

since p_{k-1} is a point of a sphere of radius ρ and v_{k-1} is orthogonal to p_{k-1}. It now follows that

$$\begin{aligned} v_k &= v_{k-1} + w_{k-1} - \frac{(p_k - p_{k-1}\cos\theta)^T (v_{k-1} + w_{k-1})}{\rho^2(1+\cos\theta)} \left(p_{k-1} + p_{k-1}\cos\theta + \frac{\Delta T}{\theta} v_{k-1}\sin\theta \right) \\ &= v_{k-1} + w_{k-1} - \frac{(\frac{\Delta T}{\theta} v_{k-1}\sin\theta)^T (v_{k-1} + w_{k-1})}{\rho^2(1+\cos\theta)} \left(p_{k-1}(1+\cos\theta) + \frac{\Delta T}{\theta} v_{k-1}\sin\theta \right) \\ &= v_{k-1} + w_{k-1} - \frac{\Delta T}{\rho^2\theta}(v_{k-1}^T v_{k-1} + v_{k-1}^T w_{k-1}) \left(p_{k-1}\sin\theta + \frac{\Delta T}{\theta} v_{k-1}\frac{\sin^2\theta}{1+\cos\theta} \right) \\ &= v_{k-1} + w_{k-1} - \frac{\Delta T}{\rho^2\theta}(\|v_{k-1}\|^2 + v_{k-1}^T w_{k-1}) \left(p_{k-1}\sin\theta + \frac{\Delta T}{\theta} v_{k-1}\frac{\sin^2\theta}{1+\cos\theta} \right) \\ &= v_{k-1} + w_{k-1} - \frac{\|v_{k-1}\|}{\rho}\left(1 + \frac{v_{k-1}^T w_{k-1}}{\|v_{k-1}\|^2}\right) \left(p_{k-1}\sin\theta + \frac{\Delta T}{\theta} v_{k-1}\frac{\sin^2\theta}{1+\cos\theta} \right) \\ &= v_{k-1} + w_{k-1} - \left(1 + \frac{v_{k-1}^T w_{k-1}}{\|v_{k-1}\|^2}\right) \left(\frac{\|v_{k-1}\|}{\rho} p_{k-1}\sin\theta + v_{k-1}\frac{\sin^2\theta}{1+\cos\theta} \right) \\ &= v_{k-1}\cos\theta - \frac{\|v_{k-1}\|}{\rho} p_{k-1}\sin\theta + w_{k-1} - \frac{v_{k-1}^T w_{k-1}}{\rho\|v_{k-1}\|} p_{k-1}\sin\theta - \frac{v_{k-1}^T w_{k-1}}{\|v_{k-1}\|^2}(1-\cos\theta)v_{k-1}. \end{aligned}$$

Finally, the discrete constant velocity model on n-sphere is given by

$$f(x,w) = \begin{bmatrix} p\cos\theta + \frac{\Delta T}{\theta} v \sin\theta \\ v\cos\theta - \frac{\|v\|}{\rho} p\sin\theta + w - \frac{v^T w}{\rho\|v\|} p\sin\theta - \frac{v^T w}{\|v\|^2}(1-\cos\theta)v \end{bmatrix}, \tag{A.13}$$

where $x = (p, v)$.

The Jacobian of f with the respect to the system state $x = (p, v)$ is given by

$$F = \left.\frac{\partial f}{\partial x}\right|_{w=0} = \begin{bmatrix} \left.\frac{\partial f_1}{\partial p}\right|_{w=0} & \left.\frac{\partial f_1}{\partial v}\right|_{w=0} \\ \left.\frac{\partial f_2}{\partial p}\right|_{w=0} & \left.\frac{\partial f_2}{\partial v}\right|_{w=0}. \end{bmatrix} \tag{A.14}$$

The first submatrix is given by

$$\frac{\partial f_1}{\partial p} = \frac{\partial}{\partial p}\left(p\cos\theta + \frac{\Delta T}{\theta} v\sin\theta\right) = I\cdot\cos\theta,$$

since θ is not a function of p. Next,

$$\begin{aligned}
\frac{\partial f_1}{\partial v} &= \frac{\partial}{\partial v}\left(x\cos\theta + \frac{\Delta T}{\theta} v\sin\theta\right) \\
&= -p\sin\theta\frac{\partial\theta}{\partial v} + I\cdot\frac{\Delta T}{\theta}\sin\theta + v\sin\theta\,\frac{\partial}{\partial v}\left(\frac{\Delta T}{\theta}\right) + v\frac{\Delta T}{\theta}\cos\theta\,\frac{\partial\theta}{\partial v} \\
&= -pv^T\frac{\Delta T}{\rho\|v\|}\sin\theta + I\cdot\frac{\Delta T}{\theta}\sin\theta - v\frac{\Delta T}{\theta^2}\sin\theta\,\frac{\partial\theta}{\partial v} + vv^T\frac{\Delta T^2}{\theta\rho\|v\|}\cos\theta \\
&= I\cdot\frac{\Delta T}{\theta}\sin\theta - pv^T\frac{\theta}{\|v\|^2}\sin\theta - vv^T\frac{\Delta T^2}{\rho\theta^2\|v\|}\sin\theta + vv^T\frac{\Delta T^2}{\theta\rho\|v\|}\cos\theta \\
&= I\cdot\frac{\Delta T}{\theta}\sin\theta - pv^T\frac{\theta}{\|v\|^2}\sin\theta + vv^T\left(\frac{\Delta T}{\|v\|^2}\cos\theta - \frac{\rho}{\|v\|^3}\sin\theta\right).
\end{aligned}$$

To calculate derivatives of f_2, notice that the terms involving w will vanish after derivation and substituting $w = 0$. It follows that

$$\frac{\partial f_2}{\partial p} = -\frac{\partial}{\partial p}\frac{\|v\|}{\rho} p\sin\theta = -I\cdot\frac{\|v\|}{\rho}\sin\theta,$$

and finally

$$\begin{aligned}
\frac{\partial f_2}{\partial v} &= \frac{\partial}{\partial v}\left(v\cos\theta - \frac{\|v\|}{\rho} p\sin\theta\right) \\
&= I\cdot\cos\theta - v\sin\theta\,\frac{\partial\theta}{\partial v} - \frac{p}{\rho}\frac{\partial\|v\|}{\partial v}\sin\theta - p\frac{\|v\|}{\rho}\frac{\partial\theta}{\partial v}\cos\theta \\
&= I\cdot\cos\theta - vv^T\frac{\Delta T}{\rho\|v\|}\sin\theta - pv^T\frac{1}{\rho\|v\|}\sin\theta - pv^T\frac{\Delta T}{\rho^2}\cos\theta \\
&= I\cdot\cos\theta - vv^T\frac{\Delta T}{\rho\|v\|}\sin\theta - pv^T\left(\frac{1}{\rho\|v\|}\sin\theta + \frac{\Delta T}{\rho^2}\cos\theta\right).
\end{aligned}$$

The Jacobian of the state transition model is

$$F = \begin{bmatrix} I \cdot \cos\theta & I \cdot \frac{\Delta T}{\theta} \sin\theta - pv^T \frac{\theta}{\|v\|^2} \sin\theta + vv^T \left(\frac{\Delta T}{\|v\|^2} \cos\theta - \frac{\rho}{\|v\|^3} \sin\theta \right) \\ -I \cdot \frac{\|v\|}{\rho} \sin\theta & I \cdot \cos\theta - vv^T \frac{\Delta T}{\rho \|v\|} \sin\theta - pv^T \left(\frac{1}{\rho \|v\|} \sin\theta + \frac{\Delta T}{\rho^2} \cos\theta \right) \end{bmatrix}. \tag{A.15}$$

The Jacobian of f with respect to the process noise signal w is

$$L = \begin{bmatrix} \frac{\partial f_1}{\partial w}\Big|_{w=0} \\ \frac{\partial f_2}{\partial w}\Big|_{w=0} \end{bmatrix} \tag{A.16}$$

Since f_1 does not depend on w, it follows that $\frac{\partial f_1}{\partial w} = 0$ Furthermore,

$$\begin{aligned} \frac{\partial f_2}{\partial w} &= I - p \frac{\partial}{\partial w} \left(v^T w \right) \frac{1}{\rho \|v\|} \sin\theta - v \frac{\partial}{\partial w} \left(v^T w \right) \frac{1}{\|v\|^2} (1 - \cos\theta) \\ &= I - \frac{pv^T}{\rho \|v\|} \sin\theta - \frac{vv^T}{\|v\|^2} (1 - \cos\theta) \end{aligned}$$

Thus,

$$L = \begin{bmatrix} 0 \\ I - \frac{pv^T}{\rho \|v\|} \sin\theta - \frac{vv^T}{\|v\|^2} (1 - \cos\theta) \end{bmatrix}. \tag{A.17}$$

Now consider the observation process given by

$$y_k = h_k(x_k, \omega_k) = \mathrm{Exp}_{p_k} \omega_k, \tag{A.18}$$

where ω_k is a concentrated zero-mean Gaussian measurement noise, while $x_k = (p_k, v_k)$. Substituting the explicit expression for exponential map of a sphere in above equation yields

$$h_k(x_k, \omega_k) = p_k \cos \frac{\|\omega_k\|}{\rho} + \frac{\omega_k}{\|\omega_k\|} \rho \sin \frac{\|\omega_k\|}{\rho}. \tag{A.19}$$

The Jacobian of observation model (A.19) with respect to system state $x_k(p_k, v_k)$ is

$$H = \begin{bmatrix} \frac{\partial h}{\partial p}\Big|_{\omega=0} & \frac{\partial h}{\partial \omega}\Big|_{\omega=0} \end{bmatrix} \tag{A.20}$$

$$= \begin{bmatrix} I \cos \frac{\|\omega\|}{\rho}\Big|_{\omega=0} & 0 \end{bmatrix} \tag{A.21}$$

$$= \begin{bmatrix} I & 0 \end{bmatrix}. \tag{A.22}$$

And finally, the Jacobian of h with respect to measurement noise ω

$$\begin{aligned} M =& \frac{\partial h}{\partial \omega}\Big|_{\omega=0} \\ =& \Bigg(- p \frac{\partial}{\partial \omega} \left(\frac{\|\omega\|}{\rho} \right) \sin \frac{\|\omega\|}{\rho} + I \frac{\rho}{\|\omega\|} \sin \frac{\|\omega\|}{\rho} + \omega \frac{\partial}{\partial \omega} \left(\frac{\rho}{\|\omega\|} \right) \sin \frac{\|\omega\|}{\rho} \\ & + \omega \frac{\partial}{\partial \omega} \left(\frac{\|\omega\|}{\rho} \right) \frac{\rho}{\|\omega\|} \cos \frac{\|\omega\|}{\rho} \Bigg) \Bigg|_{\omega=0} \end{aligned}$$

$$
\begin{aligned}
=&\left(I\frac{\rho}{\|\omega\|}\sin\frac{\|\omega\|}{\rho} - p\omega^T\frac{1}{\rho\|\omega\|}\sin\frac{\|\omega\|}{\rho} - \omega\frac{\partial\|\omega\|}{\partial\omega}\frac{\rho}{\|\omega\|^2}\sin\frac{\|\omega\|}{\rho} \right.\\
&\left. + \omega\omega^T\frac{1}{\|\omega\|^2}\cos\frac{\|\omega\|}{\rho}\right)\Bigg|_{\omega=0}\\
=&\left(I\,\mathrm{sinc}\frac{\|\omega\|}{\rho} - \frac{p\omega^T}{\rho^2}\mathrm{sinc}\frac{\|\omega\|}{\rho} - \frac{\omega\omega^T}{\|\omega\|^2}\mathrm{sinc}\frac{\|\omega\|}{\rho} + \frac{\omega\omega^T}{\|\omega\|^2}\cos\frac{\|\omega\|}{\rho}\right)\Bigg|_{\omega=0}\\
=&\left(\left(I - \frac{p\omega^T}{\rho^2}\right)\mathrm{sinc}\frac{\|\omega\|}{\rho} + \frac{\omega\omega^T}{\|\omega\|^2}\left(\cos\frac{\|\omega\|}{\rho} - \mathrm{sinc}\frac{\|\omega\|}{\rho}\right)\right)\Bigg|_{\omega=0}\\
=&I + \lim_{\omega\to 0}\frac{\omega\omega^T}{\|\omega\|^2}\left(\cos\frac{\|\omega\|}{\rho} - \mathrm{sinc}\frac{\|\omega\|}{\rho}\right)\\
=&I
\end{aligned}
$$

A.2 Diagonal positive definite matrices

The first-order motion model on $\mathcal{D}_n$ is given by

$$(p_k, v_k) = f(p_{k-1}, v_{k-1}, w_{k-1}) = \left(\mathrm{Exp}_{P_{k-1}}(\Delta T\, v_{k-1}),\, \mathrm{P}_{p_{k-1}\to p_k}\left(v_{k-1} + \Delta T\cdot w_{k-1}\right)\right). \quad \text{(A.23)}$$

By utilizing expressions for exponential map and parallel transport of $\mathcal{D}_n$ we get

$$p_k^i = p_{k-1}^i \exp\frac{\Delta T\, v_{k-1}^i}{p_{k-1}^i} \qquad i = 1, \ldots, n \quad \text{(A.24)}$$

$$v_k = (v_{k-1}^i + \Delta T\cdot w_{k-1}^i)\exp\frac{\Delta T\, v_{k-1}^i}{p_{k-1}^i} \qquad i = 1, \ldots, n. \quad \text{(A.25)}$$

The Jacobian F is then given by

$$F = \begin{bmatrix} \frac{\partial p_k}{\partial p_{k-1}} & \frac{\partial p_k}{\partial v_{k-1}} \\ \frac{\partial v_k}{\partial p_{k-1}} & \frac{\partial v_k}{\partial v_{k-1}} \end{bmatrix} \quad \text{(A.26)}$$

Then, by partial derivation we obtain

$$\frac{\partial p_k^i}{\partial p_{k-1}^j} = \delta_{ij}\cdot\left(1 - \frac{\Delta T\, v_{k-1}^i}{p_{k-1}^i}\right)\exp\frac{\Delta T\, v_{k-1}^i}{p_{k-1}^i}, \quad \text{(A.27)}$$

$$\frac{\partial p_k^i}{\partial v_{k-1}^j} = \delta_{ij}\cdot\Delta T\cdot\exp\frac{\Delta T\, v_{k-1}^i}{p_{k-1}^i}, \quad \text{(A.28)}$$

$$\frac{\partial v_k^i}{\partial p_{k-1}^j} = -\delta_{ij}\cdot\Delta T\cdot\left(\frac{v_{k-1}^i}{p_{k-1}^i}\right)^2\cdot\exp\frac{\Delta T\, v_{k-1}^i}{p_{k-1}^i}, \quad \text{(A.29)}$$

$$\frac{\partial v_k^i}{\partial v_{k-1}^j} = \delta_{ij} \cdot \left(1 + \frac{\Delta T\, v_{k-1}^i}{p_{k-1}^i}\right) \exp \frac{\Delta T\, v_{k-1}^i}{p_{k-1}^i}. \tag{A.30}$$

The Jacobian L is given by

$$L = \begin{bmatrix} \frac{\partial p_k}{\partial w_{k-1}} \\ \frac{\partial v_k}{\partial w_{k-1}} \end{bmatrix}, \tag{A.31}$$

where

$$\frac{\partial p_k^i}{\partial w_{k-1}^j} = 0 \tag{A.32}$$

$$\frac{\partial v_k^i}{\partial w_{k-1}^j} = \delta_{ij} \cdot \Delta T \cdot \exp \frac{\Delta T\, v_{k-1}^i}{p_{k-1}^i}. \tag{A.33}$$

List of Figures

Figure 2.1 Local coordinate charts on smooth manifold 8
Figure 2.2 Parallel transport of vectors along a curve on a manifold 10
Figure 2.3 Exponential and logarithmic maps of Riemannian manifold 13
Figure 2.4 Coordinate change on Riemannian manifolds 14
Figure 3.1 Illustration of a Bayesian filtering 24
Figure 3.2 Multiple object tracking flowchart 31
Figure 3.3 Global hypotheses generation in the HOMHT 35
Figure 3.4 Target trees of an TOMHT 37
Figure 3.5 Challenges in multi target tracking evaluation 50
Figure 3.6 Multi target tracking evaluation with CLEAR MOT metrics 52
Figure 4.1 Pedestrian tracking on 3DMOT2015 56
Figure 4.2 MOT Challenge benchmark sequences 59
Figure 4.3 Architecture of mask R-CNN object detector. 60
Figure 4.4 Similarity of deep embeddings 65
Figure 5.1 Brownian motion simulation on sphere 73
Figure 5.2 First-order Brownian motion simulation on sphere 74
Figure 5.3 Filtering results on sphere with zero-order model 81
Figure 5.4 RMSE of filtering on sphere 81
Figure 5.5 RMSE of filtering on SPD 82
Figure 5.6 State estimation on $\mathcal{P}_2$ 83
Figure 5.7 Filtering on sphere with first-order motion model 84
Figure 5.8 Average MSE of state estimation on $T\mathcal{S}^2$ 86
Figure 6.1 Example of tracking on sphere 93
Figure 6.2 Evaluation of multi-target tracking on the unit sphere 94
Figure 6.3 Multi-target tracking on unit sphere 95

List of Tables

Table 4.1 Pedestrian detector training strategies 64
Table 4.2 Applying segmentation masks 64
Table 4.3 Validation of the model architecture 65
Table 4.4 3DMOT2015 benchmark results 66
Table 4.5 Kinematic vs. appearance based tracking 67
Table 4.6 JIPDAF with ours and public detections 67
Table 5.1 RMSE of zero-order filtering on sphere 80
Table 5.2 State estimation on unit spheres of different dimensions 80
Table 5.3 RMSE of zero-order filtering on SPD matrices 84
Table 5.4 RMSE of filtering on space of SPD matrices 84
Table 5.5 RMSE of filtering on $T\mathcal{D}_3$ 86

Acronyms

PDF	probability density function
LS	least squares
MAP	maximum a posteriori
EAP	expected a posteriori
SPD	symmetric positive definite
KF	Kalman filter
EKF	extended Kalman filter
UKF	unscented Kalman filter
MTT	multi target tracking
NN	nearest neighbour
GNN	global nearest neighbour
MHT	multiple hypothesis tracking
HOMHT	hypothesis oriented multiple hypothesis tracking
TOMHT	track oriented multiple hypothesis tracking
PMHT	probabilistic multiple hypothesis tracking
PDA	probabilistic data association
PDAF	probabilistic data association filter
IPDAF	integrated probabilistic data association filter
JPDAF	joint probabilistic data association filter
JIPDAF	joint integrated probabilistic data association filter
RFS	random finite set
PHD	probability hypothesis density
GM-PHD	Gaussian mixture probability hypothesis density
MSE	mean-square error
RMSE	root-mean-square error
vMF	von Mises-Fisher

RM	Riemannian manifold
RM-UKF	Riemannian manifold unscented Kalman filter
RM-EKF	Riemannian manifold extended Kalman filter
CGD	concentrated Gaussian distribution
LG-EKF	Lie group extended Kalman filter
SMC	sequential Monte Carlo
RM-PDAF	Riemannain manifold probabilistic data association filter
OSPA	optimal sub-pattern assignment
CMOT	clear MOT
MOTA	multiple object tracking accuracy
MOTP	multiple object tracking precision
FP	false positive
FN	false negative
IDS	identity switch
GT	ground truth
R-CNN	region based convolutional neural network
RoI	region of interest
RPN	region proposal network
ADAS	adaptive driver assistance system
OTHR	over-the-horizon radar

Curriculum vitae

Borna Bićanić was born in Našice, Croatia in 1993. He finished Gospić high school in 2012. During his high school education, he won 4th place in the national competition in physics in 2009 and 1st place in the national competition in mathematics in 2012. He received his BSc and MSc degrees in electrical engineering and information technology from the University of Zagreb, Faculty of Electrical Engineering and Computing (UNIZG-FER) in 2015 and 2017, respectively. During his studies, he received a scholarship for excellence from the University of Zagreb multiple times. Upon finishing his studies in 2017, he joined Laboratory for Autonomous Systems and Mobile Robotics (LAMoR) at the Department of Control and Computer Engineering (ZARI) on FER as a research assistant. He also started his PhD studies at FER in the same year. Furthermore, he worked on several international and domestic scientific projects including SafeTRAM – *Systems for Increased Driving Safety in Public Urban Rail*, L4MS – *Logistics for Manufacturing SMEs*, MAS – *Development of a Multi-Functional Anti-Terrorism Systems*, DATACROSS – *Advanced Methods and Technologies in Data Science and Cooperative Systems* and A-UNIT – *Research and Development of an Advanced Unit for Autonomous Control of Mobile Vehicles in Logistics*. In June 2022, he was a member of the organizing committee of the 17th International Conference on Intelligent Autonomous Systems (IAS-17) that took place in Zagreb, Croatia. Apart from his scientific work, he has also been a teaching assistant at several undergraduate and graduate courses at FER. He is also a graduate student member of the IEEE organization. His main research interests in robotics and autonomous systems revolve around state estimation and multi-target tracking. He is an author or a co-author of 1 paper published in a peer-reviewed journal and 2 papers presented at international conferences. The full list of papers published at the time of writing of the doctoral thesis is given below.

Full list of publications

Journal publications:

a) B. Bićanić, I. Marković and I. Petrović. Multi-Target Tracking on Riemannian Manifolds via Probabilistic Data Association. IEEE *Signal Processing Letters*, 28:1555–1559, 2021, IF: 3.201 (Q2).

Conference publications:

a) B. Bićanić, M. Oršić, I. Marković, S. Šegvić and I. Petrović. Pedestrian Tracking by Probabilistic Data Association and Correspondence Embeddings. *22th International Conference on Information Fusion (FUSION)* Canada, Ottawa, pages 1–6, 2019.

b) M. Seder, L. Petrović, J. Peršić, G. Popović, T. Petković, A. Šelek, B. Bićanić, I. Cvišić, D. Josić, I. Marković, I. Petrović and A. Muhammad. Open Platform Based Mobile Robot Control for Automation in Manufacturing Logistics. *IFAC Workshop on Robot Control* South Korea, Daejeon, 2019.

Životopis

Borna Bićanić rođen je u Našicama u Hrvatskoj 1993. godine. Završio je gimnaziju Gospić 2012. godine. Tijekom srednjoškolskog obrazovanja osvojio je 4. mjesto na državnom natjecanju iz fizike 2009. godine te prvo mjesto na državnom natjecanju iz matematike 2012. godine. Zvanje prvostupnika, odnosno magistra elektrotehnike i informacijske tehnologije stekao je 2015. i 2017. godine na Fakultetu Elektrotehnike i Računarstva Sveučilišta u Zagrebu (FER). Tijekom studija, više puta je nagrađen stipendijom za izvrsnost Sveučilišta u Zagrebu. Po završetku diplomskog studija 2017. godine pridružio se Laboratoriju za autonomne sustave i mobilnu robotiku (LAMoR) na Zavodu za automatiku i računalno inženjerstvo (ZARI) na FER-u kao znanstveni suradnik. Iste godine započeo je doktorski studij elektrotehnike na FER-u. Sudjelovao je kao istraživač na nekoliko domaćih i međunarodnih znanstvenih projekata među kojima su: SafeTram – *Sustav za povećanje sigurnosti vožnje javnog urbanog tračničkog prometa*, L4MS – *Logistics for Manufacturing SMEs*, MAS – *Development of a multi-functional anti-terrorism systems*, DATACROSS – *Napredne metode i tehnologije u znanostima o podacima i kooperativnim sustavima* te A-UNIT – *Istraživanje i razvoj napredne jedinice za autonomno upravljanje mobilnim vozilima u logistici*. U lipnju 2022. godine bio je član organizacijskog odbora 17. međunarodne konferencije o inteligentnim autonomnim sustavima (IAS-17). Osim znanstvenog rada, sudjelovao je i kao asistent u nastavi na nekoliko predmeta na preddiplomskom i diplomskom studiju na FER-u. Član je udruge IEEE. Njegovi glavni istraživački interesi u robotici i autonomnim sustavima nalaze se u području estimacije stanja i praćenja više objekata. Autor je ili suautor jednog znanstvenog rada u časopisu s međunarodnom recenzijom te 2 rada prezentiranih na međunarodnim konferencijama.

Colophon

This document was typeset and inspired by the typographical look-and-feel `classicthesis` developed by André Miede, which was based on Robert Bringhurst's book on typography *The Elements of Typographic Style*, and by the `FERElemental` developed by Ivan Marković whose design was based on `FERBook` developed by Jadranko Matuško.