

# **A Novel Plenoptic Camera based Measurement System for the Analysis of Flight and Combustion Properties of Refuse-derived Fuel Particles Utilizing Tracking-by-Detection**

Zur Erlangung des akademischen Grades einer  
DOKTORIN DER INGENIEURWISSENSCHAFTEN (Dr.-Ing.)

von der KIT-Fakultät für Maschinenbau des  
Karlsruher Instituts für Technologie (KIT)  
angenommene

DISSERTATION

von  
M. Sc. Miao Zhang  
geboren am 02. Okt. 1992 in Chifeng

Tag der mündlichen Prüfung: 25. Juli 2023

Hauptreferent: apl. Prof. Dr. -Ing. Jörg Matthes  
Korreferenten: Prof. Dr. Veit Hagenmeyer  
Prof. Dr. Stefan Hinz



# Declaration

Karlsruhe, August 3, 2023

I hereby declare that I completed this work independently without any improper help and without using any aids other than those cited.

Miao Zhang

# Acknowledgements

Time flies and nearly four years of research is coming to an end. At this moment, I would like to thank the following people, without whom I would not have been able to complete this research.

First of all, I would like to express my deepest gratitude to my supervisor, apl. Prof. Dr. Jörg Matthes for the dedicated support and guidance throughout the study. Jörg continuously encouraged and was always willing and enthusiastic to assist in any way timely. I would also like to extend my sincere thanks to Dr. Markus Vogelbacher for providing constructive and innovative advice and ideas.

I am also grateful to the Department of Combustion- and Particle Technology at Institute for Technical Chemistry from Karlsruhe Institute of Technology, especially to Dr. Hans-Joachim Gehrman and Dr. Krasimir Aleksandrov, whose support for preparing and conducting the measurement campaigns enabled this research to be possible.

Thanks should also go to the project partner team from the Department of Energy Plant Technology at Ruhr-University Bochum, particularly to Prof. Dr. Scherer, Dr. Wirtz, and M.Sc. Streier for their collaborative effort on project organization and coordination. Without them, the project would not have been completed so smoothly. Special thanks to M.Sc. Streier for completing the CFD models and conducting the comparison between CFD models and the experimental results obtained in the study. Additionally, this endeavor would not have been possible without the generous support from the Arbeitsgemeinschaft industrieller Forschungsvereinigungen (AIF), who financed my research primarily.

Further, I would like to thank Prof. Dr. Hagenmeyer and Prof. Dr. Hinz for the thoughtful comments and recommendations on this dissertation. Besides, I am also thankful to the defense committee for providing knowledge and expertise.

Words cannot express my appreciation to my parents for all the support through my research and study. I am grateful for their selfless dedication and support for so many years. I also appreciate my cohort Chang Li for providing help in various aspects. I would also like to thank my dogs for all the virtual entertainment and emotional support.

I would be remiss in not mentioning my office mates for their feedback and support. Finally, I would like to acknowledge all participants that took part in the study. Thank you for your efforts during the past four years of research.



# Abstract

In the past several decades, there has been a spate of interest in applying refuse-derived fuels (RDFs) in industrial combustion processes, for instance, cement production. As a consequence of the economic merit and the carbon-neutral characteristic, RDF owns a favorable application prospect. Nevertheless, utilizing RDF for controllable and secure combustion is challenging since RDF is composed of various waste fractions with complex shapes resulting in relatively complicated flight and combustion behaviors. The undertaken research presents a novel plenoptic camera based measurement system to determine the properties of RDF particles using image processing approaches. At first, the particles are captured by a plenoptic camera, which is able to provide information in 2D gray value images and 3D point clouds, *i.e.*, each spatially captured pixel contains both gray value as unsigned integer number stored with 16 bit and spatial coordinate in mm. Based on the information, the particles can be detected by 2D gray value based algorithms and 3D clustering approaches. Owing to the considerable fluctuation of the obtained point clouds, 3D clustering approaches perform inferiorly. Compared to the 3D clustering approaches, the 2D gray value based algorithms also exhibits particular deficiencies despite their better accuracy performance. Under the circumstance, the study proposes an innovative combined detection approach that combines the detection results of a 2D gray value based algorithm and a 3D clustering method aiming at making the most of the acquired 2D and 3D information to deal with the fluctuations of the measuring system. Subsequently, the particles are tracked by the linear Kalman filter, with the 2.5D global nearest neighbor (GNN) and the joint probabilistic data association (JPDA) approach, respectively. By introducing the depth information in gating, the tracking approach is extended to 2.5D. As a consequence of several inaccurate detection results caused by the measuring system, the initial tracking results contain faulty and incomplete tracklets that entail a post-processing process. Therefore, in the study, a post-processing framework based merely on particle motion similarity is developed, which benefits a precise tracking performance by eliminating faulty tracklets, deleting outliers, connecting tracklets, and fusing trajectories. The thereby obtained 2D particle trajectories are converted into spatial particle traces in accordance with the provided 3D information by the plenoptic camera. Subsequently, the trajectories are estimated by polynomials independently in three spatial directions for the sake of compensating fluctuations existing in the trajectories. To conduct a quantitative assessment of the proposed approaches, manually labeled ground truth datasets for particle detection and tracking are generated. The presented methods in the study are proven to deliver satisfactory performances with respect to several measurements, for instance, precision, recall, and  $F_1$ -score.

The thereby obtained complete spatial fuel trajectories enable the analysis of the flight and combustion behaviors of various fuel fractions. For the flight properties, the flight duration and space-sliced particle velocity in the depth direction are elaborated. The combustion behavior refers to the ignition time. As revealed by the study, wood chips and PE granules move more rapidly than the two paper materials. While paper shreds and confetti could be ignited during their flights, PE granules and wood chips land before ignition. The acquired statements demonstrate an excellent agreement with the computational simulations, which reversely prove the availability and applicability of the developed measurement system. This new measurement system and the provided experimental results can benefit a better understanding of the RDF's combustion for future research.



# Kurzfassung

In den letzten Jahrzehnten ist das Interesse an der Verwendung von Ersatzbrennstoffen (EBS) in industriellen Verbrennungsprozessen, z.B. bei der Zementherstellung, stark gestiegen. Infolge des wirtschaftlichen Nutzens und der Kohlenstoffneutralität besitzen Ersatzbrennstoffe eine günstige Anwendungsperspektive. Dennoch ist die Nutzung von Ersatzbrennstoffen für eine kontrollierte und sichere Verbrennung eine Herausforderung, da sich Ersatzbrennstoffe aus verschiedenen Abfallfraktionen mit komplexen Formen zusammensetzen, was zu relativ komplizierten Flug- und Verbrennungseigenschaften führt. In der vorliegenden Arbeit wird ein neuartiges plenoptisches, kamerabasiertes Messsystem zur Bestimmung der Eigenschaften von RDF-Partikeln mit Hilfe von Bildverarbeitungsansätzen vorgestellt. Zunächst werden die Partikel von einer plenoptischen Kamera erfasst, die in der Lage ist, Informationen in 2D-Grauwertbildern und 3D-Punktwolken zu liefern, d.h. jedes räumlich erfasste Pixel enthält sowohl den Grauwert als vorzeichenlose Ganzzahl, die mit 16 Bit gespeichert als auch die räumliche Koordinate in mm. Auf der Grundlage dieser Informationen können die Partikel durch 2D-Grauwertalgorithmen und 3D-Clustering-Ansätze erkannt werden. Aufgrund der erheblichen Fluktuation der erhaltenen Punktwolken schneiden 3D-Clustering-Ansätze schlechter ab. Im Vergleich zu den 3D-Clustering-Ansätzen weisen die 2D-Grauwert-basierten Algorithmen trotz ihrer besseren Genauigkeitsleistung ebenfalls besondere Defizite auf. Vor diesem Hintergrund wird in der Studie ein innovativer kombinierter Erkennungsansatz vorgeschlagen, der die Erkennungsergebnisse eines 2D-Grauwert-basierten Algorithmus und einer 3D-Clustering-Methode kombiniert und darauf abzielt, die gewonnenen 2D- und 3D-Informationen optimal zu nutzen, um die Schwankungen des Messsystems zu bewältigen. Anschließend werden die Partikel mit dem linearen Kalman-Filter, dem 2,5D Global Nearest Neighbour (GNN) bzw. dem Joint Probabilistic Data Association (JPDA) Ansatz verfolgt. Durch die Einführung der Tiefeninformation im Gating wird der Tracking-Ansatz auf 2,5D erweitert. Als Folge mehrerer ungenauer Erkennungsergebnisse, die durch das Messsystem verursacht werden, enthalten die anfänglichen Tracking-Ergebnisse fehlerhafte und unvollständige Tracklets, die einen Nachbearbeitungsprozess erforderlich machen. Daher wird in der Studie ein Nachbearbeitungsrahmen entwickelt, der lediglich auf der Ähnlichkeit der Partikelbewegungen basiert und eine präzise Verfolgungsleistung durch die Eliminierung fehlerhafter Tracklets, das Löschen von Ausreißern, das Verbinden von Tracklets und das Verschmelzen von Trajektorien ermöglicht. Die so gewonnenen 2D-Partikel-Trajektorien werden in Übereinstimmung mit den von der plenoptischen Kamera bereitgestellten 3D-Informationen in räumliche Partikelspuren umgewandelt. Anschließend werden die Trajektorien durch Polynome unabhängig in drei Raumrichtungen geschätzt, um die in den Trajektorien vorhandenen Schwankungen zu kompensieren. Zur quantitativen Bewertung der vorgeschlagenen Ansätze werden manuell Ground-Truth-Datensätze zur Partikeldetektion und -verfolgung erstellt. Die in der Studie vorgestellten Methoden liefern nachweislich zufriedenstellende Leistungen in Bezug auf verschiedene Messgrößen, wie z.B. Präzision, Recall und  $F_1$ -Score.

Die so gewonnenen vollständigen räumlichen Brennstofftrajektorien ermöglichen die Analyse des Flug- und Verbrennungsverhaltens verschiedener Brennstofffraktionen. Für die Flugeigenschaften werden die Flugdauer und die räumlich geschnittene Partikelgeschwindigkeit in Tiefenrichtung herausgearbeitet. Das Verbrennungsverhalten bezieht sich auf die Entzündungszeit. Wie die Studie zeigt, bewegen sich Holzhackschnitzel und PE-Granulat schneller als die beiden Papiermaterialien. Während

---

Papierschnipsel und Konfetti während ihres Fluges entzündet werden konnten, landen PE-Granulat und Holzhackschnitzel vor der Entzündung. Die gewonnenen Aussagen zeigen eine hervorragende Übereinstimmung mit den rechnerischen Simulationen, was umgekehrt die Verfügbarkeit und Anwendbarkeit des entwickelten Messsystems belegt. Dieses neue Messsystem und die bereitgestellten experimentellen Ergebnisse können zu einem besseren Verständnis der Verbrennung von Ersatzbrennstoffen für zukünftige Forschungen beitragen.

# Contents

<b>Acknowledgements</b>	<b>i</b>
<b>Abstract</b>	<b>iii</b>
<b>Kurzfassung</b>	<b>v</b>
<b>1. Introduction</b>	<b>1</b>
1.1. Motivation	1
1.2. Related work	3
1.3. Problem formulation	4
1.4. Objective and outline	5
<b>2. Experimental setup and theoretical foundations</b>	<b>7</b>
2.1. Experimental setup	7
2.1.1. BRENDA test facility	7
2.1.2. Refuse-derived fuel	10
2.1.3. Selection of camera system	11
2.1.4. Plenoptic camera system	13
2.1.5. Fluctuation and calibration of the depth coordinate	15
2.2. Particle detection	18
2.2.1. Background subtraction	20
2.2.2. 2D-OTSU threshold selection method	22
2.2.3. 2D-Scale invariant feature transform (SIFT)	23
2.2.4. DBSCAN clustering method	26
2.2.5. 3D clustering based on morphological operations	28
2.2.6. 3D clustering based on quasi-cluster centers	29
2.3. Multiple particle tracking and post-processing	30
2.3.1. Kalman filter	32
2.3.2. Nearest neighbor and global nearest neighbor approach	33
2.3.3. Probabilistic data association	34
2.3.4. Joint probabilistic data association	36
2.3.5. Post-processing	40
<b>3. Novel approaches to particle detection and classification</b>	<b>41</b>
3.1. A novel combined multiple particle detection approach	41
3.2. 3D Particle classification	47
3.3. Detection performance evaluation method	49
<b>4. Extended multiple particle tracking and post-processing</b>	<b>53</b>
4.1. 2.5D extended tracking algorithm	53
4.1.1. 2D linear Kalman filter estimator	53
4.1.2. 2.5D Mahalanobis distance gating	54



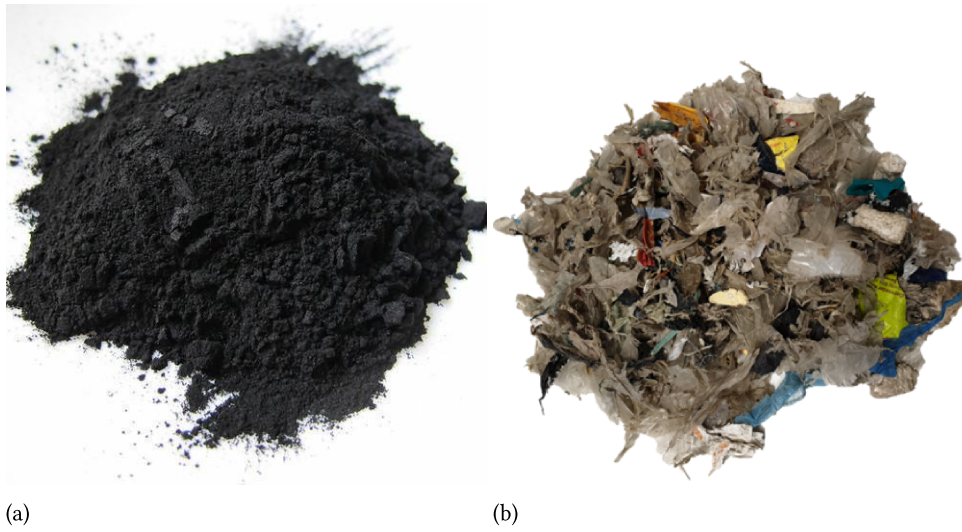
4.1.3.	Modified data association approach . . . . .	56
4.1.4.	Update of state vectors . . . . .	57
4.2.	Novel post-processing framework . . . . .	58
4.2.1.	False tracklets elimination . . . . .	59
4.2.2.	Deleting outliers in tracklets . . . . .	60
4.2.3.	Tracklets connection and merging . . . . .	62
4.2.4.	Trajectories fusion . . . . .	70
4.2.5.	Application examples . . . . .	71
4.3.	3D Regression of trajectories . . . . .	74
<b>5.</b>	<b>Results and discussion . . . . .</b>	<b>79</b>
5.1.	Results of multiple particle detection and classification . . . . .	79
5.1.1.	Ground truth dataset . . . . .	79
5.1.2.	Results of 2D background subtraction . . . . .	81
5.1.3.	Results of particle detection . . . . .	84
5.1.4.	Results of 3D particle classification . . . . .	98
5.2.	Results of multiple particle tracking . . . . .	99
5.2.1.	Ground truth of particle tracking . . . . .	99
5.2.2.	Tracking performance evaluation criteria . . . . .	101
5.2.3.	2.5D Multiple particle tracking . . . . .	103
5.2.4.	3D Regression . . . . .	106
5.3.	Analysis of RDF flight and combustion properties . . . . .	108
5.3.1.	Flight Behaviors of RDF Particles . . . . .	108
5.3.2.	Combustion Behaviors of RDF Particles . . . . .	116
<b>6.</b>	<b>Conclusion and outlook . . . . .</b>	<b>119</b>
6.1.	Conclusion and overview . . . . .	119
6.2.	Future work . . . . .	121
	<b>Bibliography . . . . .</b>	<b>123</b>
	<b>Nomenclature . . . . .</b>	<b>133</b>
	<b>List of Figures . . . . .</b>	<b>137</b>
	<b>List of Tables . . . . .</b>	<b>145</b>
<b>A.</b>	<b>Appendix . . . . .</b>	<b>147</b>
A.1.	2D particle Trajectories . . . . .	147
A.2.	Comparison of the experimental results with CFD simulations . . . . .	151

# 1. Introduction

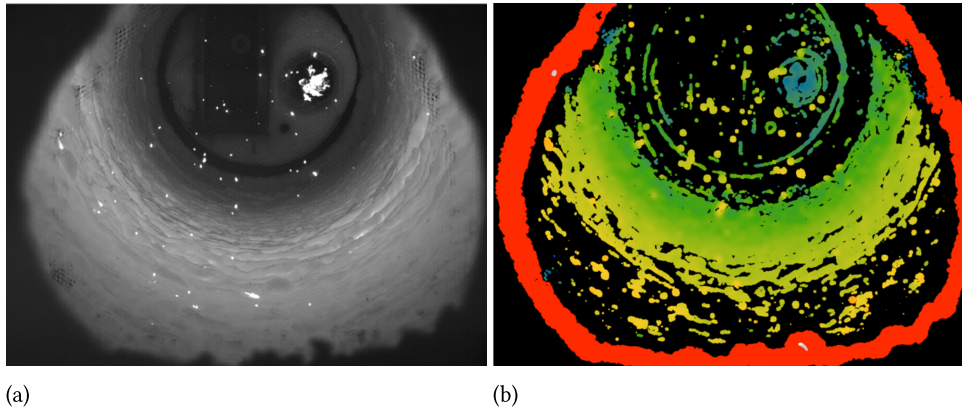
## 1.1. Motivation

In the last several decades, the application of refuse-derived fuel (RDF) has been favored in industrial combustion processes. In addition to economic advantages, a significant merit of utilizing RDF is its benefit to the CO<sub>2</sub> balance in the combustion process owing to the biogenic proportion of RDF. Weber *et al.*'s [121] study provides a detailed overview of all treatment facilities in Germany that converted waste into energy in 2015, including *e.g.* RDF power plants, municipal solid waste incineration plants, and cement plants (co-firing of waste). According to the study, roughly 320 PJ of end energy are produced in German waste treatment plants consisting of 225 PJ of heat and 90 PJ of electricity, contributing to around 3.7 % of the German end energy consumption. Faulstich *et al.* [28] studied the utilization of waste-based fuels in various plants in Germany, such as RDF heating power plants and cement plants. The research indicates that RDF has also been widely applied in the cement industry to replace fossil fuels in addition to industrial RDF power plants. The German Association for Substitute Fuels, Waste Wood, and Biogenic Waste estimates the amount of high-caloric waste appropriate for co-incineration at around 6 million tons per year [1]. In 2020, alternative fuels with energy production of 66.4 million GJ composed 69.2 % of the total fuel energy demand of the German cement industry and had been increasingly replacing fossil energy consumption [3]. Typical alternative fuels are waste oil, solid fuels, *e.g.*, wood and plastic waste, processed fractions from industrial, commercial, and municipal waste, animal meal, as well as lumpy fuels, especially used tires. In terms of mass, the largest proportion of high-calorific waste in cement works is processed, solid, pneumatically conveyable fuels, which are often denominated as Fluff.

Since the composition of Fluff is considerably complex with various fractions, for instance, paper, wood chips, and plastics, the flight and combustion behaviors can be significantly complicated and unsteady, which affects its widespread application. In addition, the particle size of Fluff (particle size in cm range) differs greatly from the standard fuel coal dust (particle size <200  $\mu\text{m}$ ), as depicted in Figure 1.1. Therefore, replacing fossil fuels with Fluff in the cement industry remains problematic. The changes in flame lengths, local heat flows, burnout times, and particle trajectories of Fluff result in several issues, such as malfunctions, decreases in energy efficiency, and negative impact on the product quality. Thus, the properties of Fluff ought to be investigated detailedly. Nevertheless, studies of replacing standard fuels with Fluff on the original scale are generally too expensive and complex. Moreover, the accessibility of measurement technology is also restricted. Under this circumstance, 3D Computational Fluid Dynamics (CFD) simulations of Fluff's combustion provide a cost-effective and efficient alternative. The simulation outcomes are of significant importance to instruct the industrial application of Fluff and entail validation of real technical data gathered in a semi-technical rotary kiln by a camera measuring system. Hence, the research project FLUFF, financed by AiF – German Federation of Industrial Research Associations, aims to conduct 3D CFD simulations of the entire flight and combustion processes of RDF particles and validate the outputs of the CFD model utilizing computed particles' properties based on captured images from a camera system.



**Figure 1.1.:** Example images of coal dust and Fluff particles. (a) Coal dust. (b) Fluff particles [111].



**Figure 1.2.:** Examples of captured images [129]. (a) Basic-focus image captured by the plenoptic camera. (b) Corresponding depth map in false color, where black indicates no 3D information available.

This work focuses on monitoring the flight and combustion processes of RDF particles in the test facility BRENDA using a plenoptic camera system to investigate several particles' properties based on the concept of tracking-by-detection. The applied camera enables simultaneous 2D gray value and 3D position measurements. As depicted in Figure 1.2, the camera provides gray value images as captured by conventional cameras, and meanwhile, the corresponding spatial positions of the captured pixels can also be measured. Based on the obtained information, RDF particles within video sequences ought to be detected and tracked to build spatial particle trajectories for further research into particles' properties. For the sake of studying the combustion properties of the particles, brightness changes of the particles represented as varying gray values on the captured images should be taken into consideration as well. The acquired flight and combustion properties of the particles, such as velocity, dwell time, and ignition time, are compared to the outcomes of the CFD model for the purpose of an objective and adequate validation of the model.

## 1.2. Related work

Since RDF is a vital alternative to fossil fuels, researchers have conducted a spate of investigations into its characteristics and behaviors in recent years. For instance, Marsh *et al.* [71] focused on the physical and thermal properties of extruded RDFs for further application in energy from waste technologies. The material charge was compressed and compacted to produce the desired extrusions, whose properties were then investigated by compressive and devolatilization testing. The research reports on the RDF pellet compressive strength by extrusion conditions and the devolatilizing properties at a high-temperature environment. Dou *et al.* [23] investigated the pyrolysis characteristics of RDF in a pilot-scale unit to determine the operating conditions of a melting incinerator. The research utilized gas chromatography for measuring the gaseous volatile and tar components, and moreover, the morphology of the char derived from RDF pyrolysis was observed by scanning electron microscopy.

In addition to the physical and chemical properties, the flight and combustion behaviors have also attracted a spate of interest since combustion builds an integral part of RDF's application. To determine RDF's flight characteristics and combustion behavior, Marzi *et al.* [72] researched trajectories of single RDF fractions in a cement kiln. Danz *et al.* [20] accumulated practical information concerning RDF's flight and combustion behaviors in a rotating tubular kiln by investigating trajectories of single RDF fractions and computing their combustion through air separation. Based on empirical methods and models, the research is able to determine several properties of RDF, for instance, the sink speed, the release speed, and the particle size. Liedmann *et al.*'s [62] paper demonstrates a simplified modeling approach for the combustion and flight behavior of RDF processed from municipal or industrial waste based on an advanced fuel characterization starting with a sorting analysis of various fractions. Beckmann *et al.* [11] applied various examples to elaborate on the influence of the RDF's fuel technical properties on its combustion behavior. In addition, the combustion behavior of RDF is also investigated by modeling the combustion of RDF particles under the consideration of particle shrinking, the release of volatile matter, and the gasification of fixed carbon. Nørskov [84] carried out experiments in a combustion reactor to investigate burnout behaviors of RDF particles, including ignition, devolatilization, and char oxidation times. Additionally, the impact of fuel properties on the combustion was also analyzed based on a developed mathematical model of the rotary kiln. Duan *et al.* [24] studied the combustion behavior and pollutant emission characteristics of RDF in a pilot-scale vortexing fluidized bed combustor. The authors employed flue gas recirculation (FGR) combustion mode to analyze the effects of several operating parameters, *e.g.*, oxygen ratio and temperature, on the combustion behavior and emission characteristics, such as CO and NO<sub>x</sub> emission. These studies use particular instruments, methods, or CFD simulations supplemented by empirical models to investigate various RDFs' flight and combustion behaviors. Therefore, they might suffer from a lack of generalizability under some circumstances, especially when researching flight properties. Imprecisely localization and tracking of particles could abate the accuracy of the results.

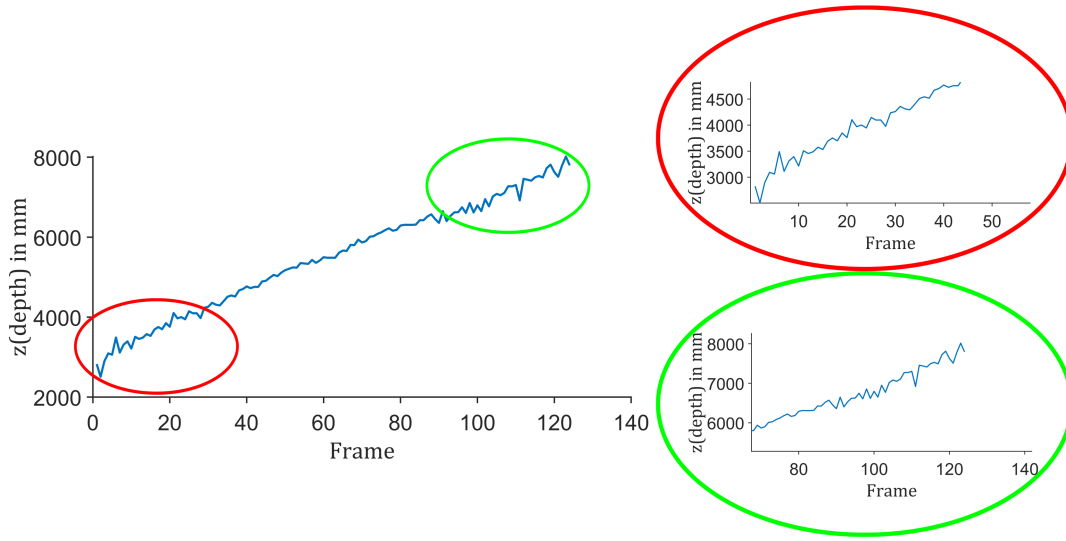
Recently, the investigation of various fuels by means of advanced imaging systems and approaches has attracted the attention of researchers. Different from previous studies, where physical or mathematical models are necessary for analyzing the fuel's properties to some extent, the image processing techniques can provide information without particular assumptions, which facilitates the accuracy and benefits a wide range of applications. Pedersen *et al.* [90] utilized an extraordinary camera setup for the sake of monitoring the combustion processes of various kinds and percentages of alternative fuels inside a rotary kiln. Nakhaei *et al.* [79] conducted studies on the physical and aerodynamic characteristics of RDF samples, where the particles were classified in a wind sieve and physically characterized by weight measurement and 2D photographing subsequently. The study put emphasis on CFD model relevant parameters, such as the size, shape, and mass of the RDF particles. Streier *et al.* [110] investigated the

aerodynamic properties of RDF in a drop shaft system using computer vision methods, including a method determining 3D particle models as well as an approach to obtain temporal particles' positions and velocities. An in-depth review of versatile image processing techniques applied in fuel science is given in [108]. In these studies, image processing was used as an auxiliary research method to provide uncomplicated fuel information, such as a single 2D trajectory. These researches are insufficient to simultaneously track the spatial motions of multiple particles independently without using any supplementary information, such as empirical models, computational models, *etc.* The literature survey reveals a research blank space of a straightforward application of image processing techniques to monitor industrial tanglesome fuel motions and combustions.

### **1.3. Problem formulation**

As delineated before, the primary task of the work is developing a measurement system for investigating the spatial flight and combustion properties of particular fuel particles using the basic idea of tracking-by-detection based on captured images from a plenoptic camera system in relatively hot measurement conditions (round 1200 °C). Nevertheless, current solutions to object detection and tracking fail to address the special issues listed below:

1. Flames, temperature fluctuations, and flickering from hot air strands of real combustion processes in the rotary kiln and combustion chamber challenge the performances of current background subtraction approaches, which is essential for a reliable 2D particle detection. Therefore, the accuracy and adequacy of the state-of-the-art background models under the mentioned conditions remain questionable.
2. Object detection algorithms based on gray value images fail to identify several fuel particles accurately because of errors from background subtraction .
3. 3D detection approaches fail to detect particles close to the inside wall of the rotary kiln since these particles are considered part of the wall.
4. Although the plenoptic camera can implement 3D measures with a single camera, the limitation of the camera is also apparent. The depth information captured and calculated by the camera fluctuates considerably, as schematically illustrated in Figure 1.3. Since the object tracking algorithms depend highly on the coordinates of the particles, a noticeable fluctuation of the particles' coordinates impacts the tracking performance considerably negatively.
5. The applied high-speed plenoptic camera records the scenes with a framerate of 330 fps, resulting in video sequences with several thousand frames. Plenty of particles are conveyed inside the scenes, and therefore, the requirements for object tracking algorithms are raised. The algorithms must deal with a large database with sufficient accuracy and efficiency.
6. The fuel particles rotate irregularly during their flights, which sometimes causes the invisibility of a few particles. Therefore, the tracking algorithm must overcome this difficulty.
7. Another challenging area in the work is determining realistic flight paths and velocities of the fuel particles from the achieved disturbed spatial trajectories.
8. Because of the limited amount of researches into the analysis of particles' combustion properties based exclusively on image processing approaches, the measures describing the properties and the approaches to computing the measures need defining and determining.



**Figure 1.3.:** An example of depth-frame of a particle trajectory that is acquired by detecting and tracking refuse-derived fuel particles inside a rotary kiln based on images captured by a plenoptic camera. The rotary kiln has a length of 8.4 m. Abscissa represents frame and the corresponding temporal depths of the trajectory are plotted on the ordinate.

## 1.4. Objective and outline

The research aims to develop a novel plenoptic camera based measurement system, which allows the investigation of several fuel particles' flight and combustion properties. In order to achieve the goal, the unresolved problems presented in the last section must be overcome. Hence, object detection and tracking algorithms ought to be adapted to the given work. Moreover, novel algorithms concerning object detection and tracking are also developed for the sake of sufficient performance accuracy. Figure 1.4 outlines the primary contribution of the dissertation.

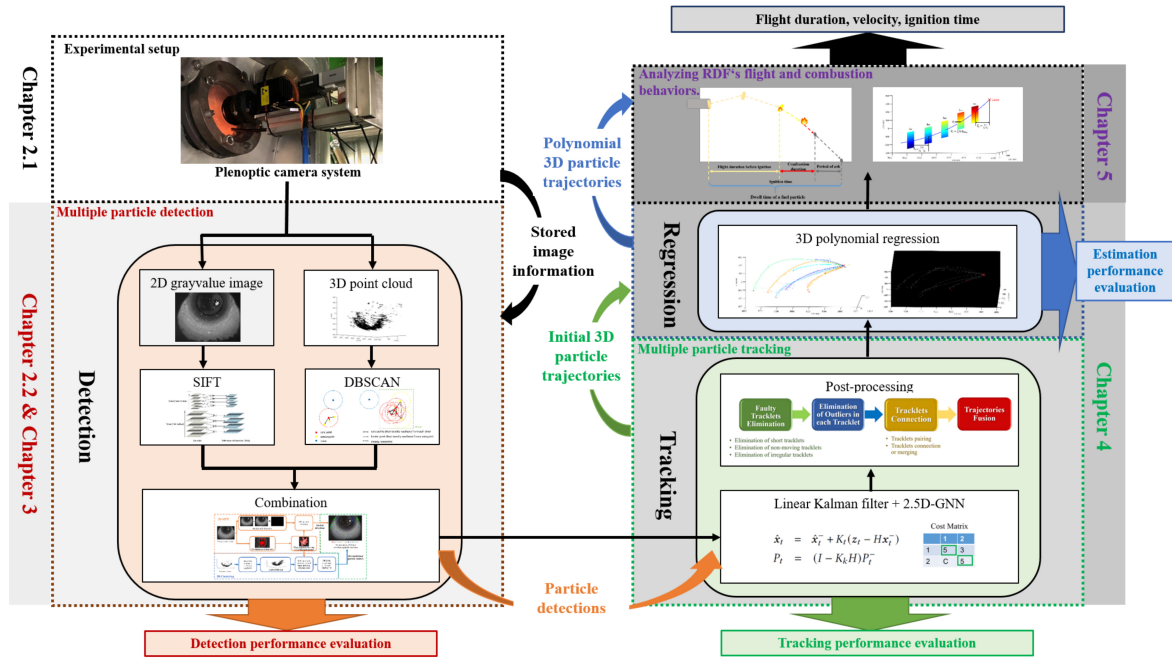
This dissertation is organized as follows:

In Chapter 2, the experiment setup, including the test facility BRENDA and the applied plenoptic camera system, is schematically described. Besides, foundational work focused on image processing, especially particle detection and multiple particle tracking, is also discussed.

The third chapter describes a novel methodology concerning particle detection, which combines a 2D gray value based detection approach and a 3D clustering algorithm to enhance the detection accuracy. Furthermore, a modification of background subtraction methods is also presented. The approach to spatially classifying the detected particles is proposed in this chapter as well.

Chapter 4 focuses on an extended 2.5D particle tracking algorithm and a novel post-processing approach for multiple particle tracking. With the proposed post-processing, false tracklets can be deleted and, meanwhile, incomplete tracklets can be connected into desired trajectories. Subsequently, the acquired trajectories are converted into 3D particle traces, which entail a regression procedure to compensate for the fluctuation of the spatial coordinates from the plenoptic camera.

In Chapter 5, the new developed measurement system is evaluated. The results of particle detection from various approaches are presented and compared. Consequently, the detected particles are temporally associated into tracklets that are connected into trajectories afterward. The obtained trajectories are then quantitatively compared with manually tracked ground truth for the purpose of a subjective statement of the tracking performance. Furthermore, the 3D particle trajectories after 3D regression



**Figure 1.4.:** Schematic of the primary contribution of the dissertation.

are also provided in the fifth chapter. Besides, the flight and combustion properties of the fuel particles based on the acquired trajectories are also analyzed.

The last chapter concludes the work and gives recommendations concerning future researches.

## 2. Experimental setup and theoretical foundations

This chapter focuses on the experimental setup and the theoretical foundations of the presented work. The first part describes the test facility BRENDA and discusses several 3D camera models briefly. In Chapter 2.2, the foundational approaches of particle detection and tracking are outlined and reviewed. Furthermore, several particle detection and tracking methods applied in the work are detailed as well.

### 2.1. Experimental setup

The test facility named BRENDA, located at the KIT campus north, is elaborated on in the section. The experiments conducted in a rotary kiln and a combustion chamber of the test facility are described in Chapter 2.1.1. The utilized RDF fractions are also presented in Chapter 2.1.2. Chapter 2.1.3 introduces and compares several widely applied 3D cameras. In Chapter 2.1.4, the applied plenoptic camera that enables a spatial measurement with only a single sensor is illustrated. Moreover, the plenoptic function is also briefly explained based on which the plenoptic camera is produced.

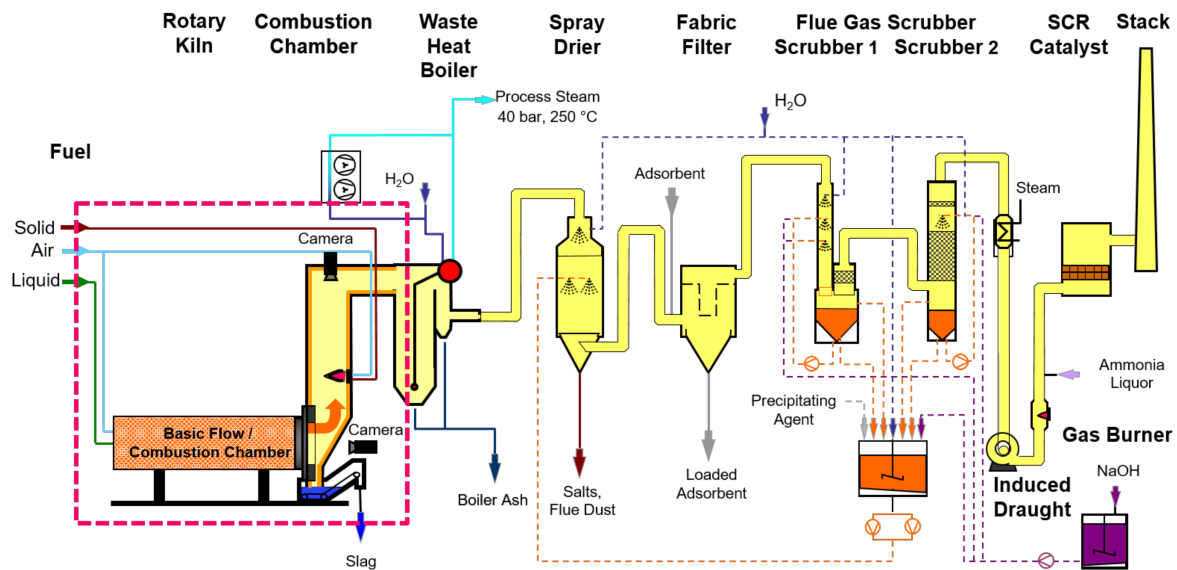
#### 2.1.1. BRENDA test facility

The test facility named BRENDA, which stands for (BRENnKammer mit DAmpfkessel, Engl. Combustion chamber with steam boiler) locates at the Institute for Technical Chemistry at KIT. As schematically illustrated in Figure 2.1, BRENDA is equipped with a boiler for heat recovery and a flue gas cleaning system [6]. The test facility enables a scalable investigation of the combustion process, for instance, the thermal properties of the fuels. Two main components of BRENDA are the rotary kiln and the combustion chamber, which are able to provide a thermal power of 1.5 MW and 1 MW, respectively. The experiments to analyze the particles' properties are carried out in these two components.

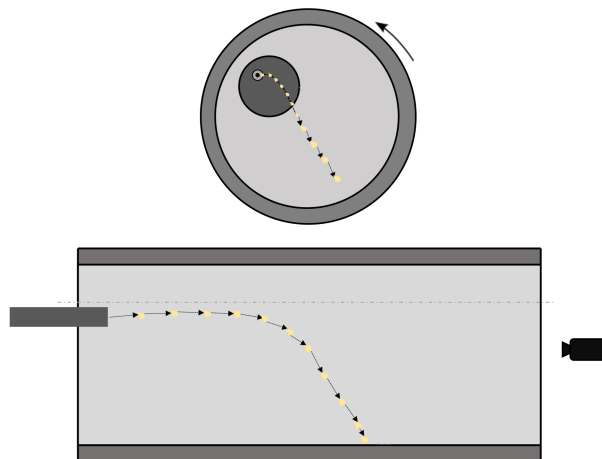
##### Rotary kiln

The rotary kiln has a length of 8.4 m and an inside diameter of 1.4 m. The refuse-derived fuel particles are blown into the rotary kiln via a lance at the inlet of the kiln, as illustrated in Figure 2.2. The rotary kiln rotates around its major axis with an angular velocity of 0.2 rpm. The rotary kiln rotates with the minimum speed to avoid sticking particles to the rotary kiln. Various RDF fractions are manually conveyed into the rotary kiln through the lance under the air feed pressure varying from 0.5 bar (outlet velocity  $\sim 4$  m/s, volume flow  $70.5 \text{ m}^3/\text{h}$ ) to 5 bar (outlet velocity  $\sim 16$  m/s, volume flow  $76.6 \text{ m}^3/\text{h}$ ). The conducted experiment primarily experiments with the air feed pressure of 4.5 bar and 5 bar since the particles gets stuck inside the lance with a pressure less than 4 bar. Simultaneously, a plenoptic camera is mounted outside the kiln to observe the particle movements through a quartz glass window at the outlet of the kiln, as depicted in Figure 2.3. Meanwhile, an infrared camera (IR, measuring at a wavelength of  $10.6 \mu\text{m}$ ) is also applied to measure the temperature distribution inside the kiln. Since the camera is mounted inside the kiln, which can reach a temperature of maximal  $1200^\circ\text{C}$ , the camera





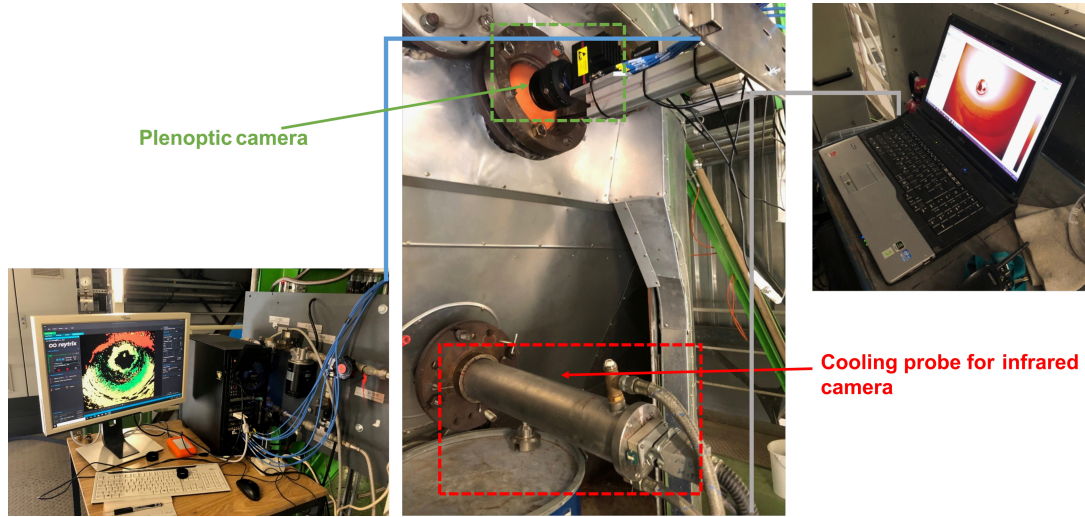
**Figure 2.1.:** Schematic of the BRENDA test facility [6]. Part of the test facility, where the experiments are conducted, is marked with a red dashed rectangle.



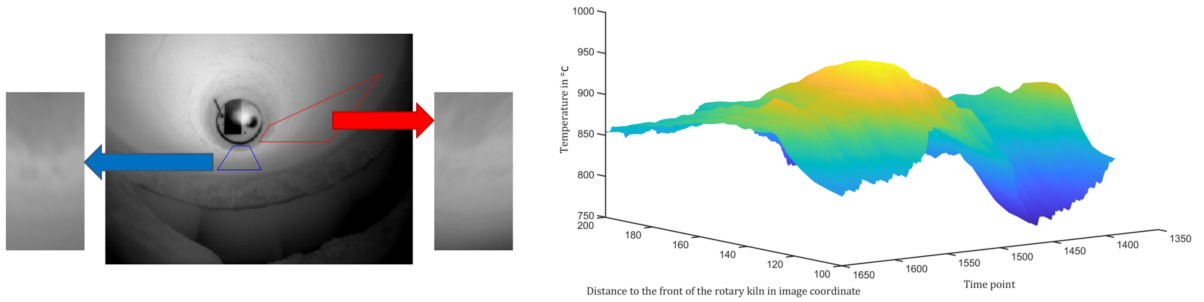
**Figure 2.2.:** Sketch map of fuel particle moving in the rotary kiln.

requires a cooling system to ensure a rational ambient temperature range. Thus, the camera is installed in a water-cooled cooling probe.

As mentioned, the infrared camera at the end of the rotary kiln enables the determination of the temperature profile at the inner wall of the rotary kiln. Trapezoidal image regions are defined from the camera perspective, which can be transformed into an equalized rectangle by a geometric transformation, as shown in Figure 2.4. This allows the extraction of inner wall temperatures along the rotary kiln axis by determining the mean temperature in each image line of the rectangle. Therefore, the temperature profile along the inner wall of the rotary kiln can be ascertained. An example of the temperature profile inside the rotary kiln is also illustrated in Figure 2.4. The acquired temperature information of the rotary kiln's inner wall is only essential for the later alignment with the CFD simulation. Thus, the



**Figure 2.3.:** Images of the experiments conducted in the rotary kiln. The plenoptic camera mounted outside the kiln marked with a green dashed rectangle is triggered by the computer in the left. An infrared camera cooled by a cooling probe is marked with a red dashed rectangle. The camera is controlled by the laptop on the right side.

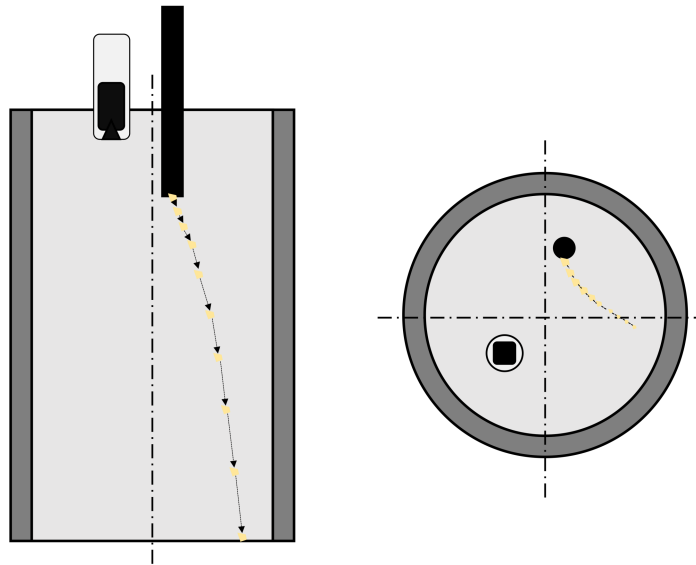


**Figure 2.4.:** Defined trapezoidal image area and an example of temperature profile at the inner wall of the rotary kiln. Left: Defined trapezoidal image area. Right: Example of temperature profile inside the rotary kiln.

delivered IR images are not considered further to analyze the RDF particles' properties within this work.

### Combustion chamber

The post-combustion chamber is 9 m high with an inner radius of 1.75 m. The fuel particles are manually blown into the chamber via a position-adjustable lance above, which has a diameter of 40 mm, as illustrated in Figure 2.5. Unlike the experiments in the rotary kiln, the plenoptic camera is installed directly at the top of the inside chamber, which can be heated above 900 °C. Hence, the camera is mounted in a water-cooled cooling probe, whose real-time temperature is measured by thermocouples, as illustrated in Figure 2.6. The height position of the lance is adjusted until it is visible by the camera.



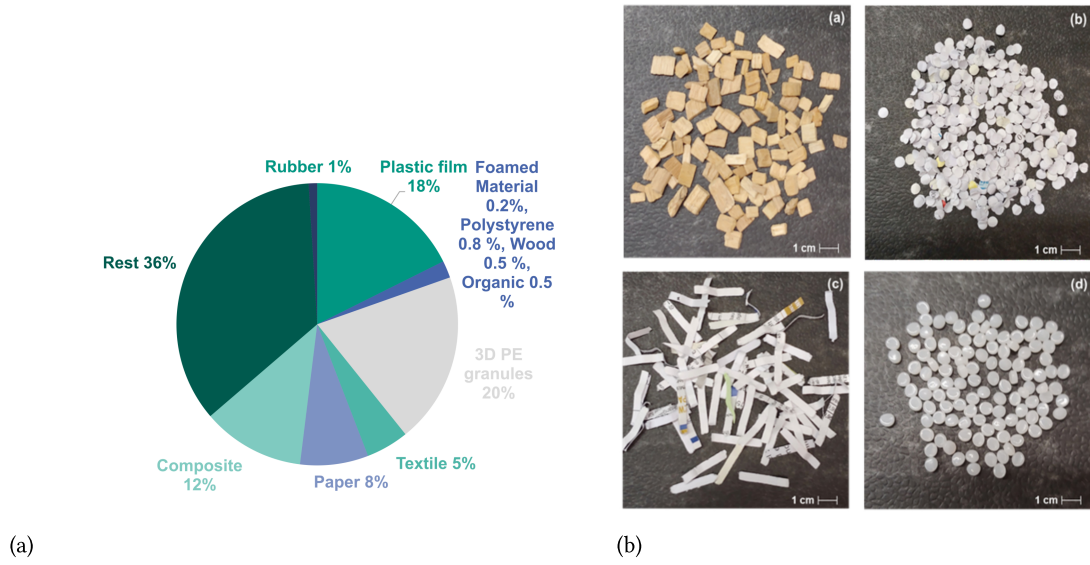
**Figure 2.5.:** Sketch map of fuel particle moving in the combustion chamber.



**Figure 2.6.:** Images of the experiments conducted in the combustion chamber. The plenoptic camera cooled by a cooling probe installed at the top of the combustion chamber is marked with a yellow dashed rectangle. The real-time temperature inside the cooling probe is measured and displayed by the thermocouple on the left side. A height-adjustable lance is marked with a red dashed rectangle.

### 2.1.2. Refuse-derived fuel

Refuse-derived fuel (RDF), produced from various types of waste, has a wide application in combustion processes, such as combustion in cement industry kilns for heat production. In addition to the cost efficiency, a significant advantage of utilizing RDF is the benefit for the  $\text{CO}_2$  balance in the combustion process owing to the biogenic proportion of RDF. The most widely applied RDFs are processed, solid, pneumatically conveyable fuels and often denominated as Fluff. Fluff is a mixture of various fractions,



**Figure 2.7.:** Refuse-derived fuel. (a) Composition of the RDF. (b) Various fractions of applied RDFs [130], (a) wood chips, (b) confetti, (c) paper shreds, (d) PE granules.

such as paper, wood, plastic film, and PE granules, as depicted in Figure 2.7. The particle sizes can be estimated according to the 1 cm scale in the bottom right of Figure 2.7.

### 2.1.3. Selection of camera system

To achieve the 3D localization and tracking of fuel particles, a camera system that is able to capture scenes spatially is of significant importance. Instances of typical 3D cameras are the stereo camera system, the time-of-flight camera, the structured light camera, and the plenoptic camera that are briefly illustrated in the following.

One general way to realize the spatial measurement is the stereo camera system consisting of more than one lens with a separate sensor for each lens. By utilizing stereo matching algorithms, pixels corresponding to the same 3D point from multiscopic views can be found [87]. This allows the stereo camera system to capture 3D images. This kind of camera system is widely applied for the acquisition of stereoviews and 3D information. For instance, Wu *et al.* [124] used a stereo camera system to detect and localize objects, and Raffel *et al.* [95] applied a stereo camera system for Particle Image Velocimetry to investigate fluid fields three-dimensionally. Notwithstanding, a stereo camera system entails a quite simultaneous trigger and a sophisticated calibration for both lenses. Moreover, for a stereo camera system with more than one camera, the test facility must contain more apertures and appropriate spaces for the cameras, which restricts the application of the stereo camera system. In addition, the calibration of the camera inside a combustion chamber could also be problematic, as in the present work.

Alternatively, time-of-flight (ToF) cameras, which employ time-of-flight techniques to acquire distance information of objects, have also found wide applications in various fields, for instance, robotics and navigations. Light pulses illuminate the scenes for a short time, which are reflected by objects and gathered by the time-of-flight camera afterward. By multiplying the time delay of the reflected light pulses and the light speed together, the objects' depth information can be obtained. In general, a time-of-flight system consists of an optical transmitter and an optical receiver [98]. This camera system is cost-efficient and able to provide a high real-time capability. Therefore, the ToF camera is often



**Table 2.1.:** Comparison of different 3D imaging technologies according to [59] and [2].

Type	Stereo camera system	ToF camera	Structured light camera	Plenoptic camera
Utilized property	Distance between cameras	Speed of light	Pattern of light	Positioning of microlenses
Effective range	Long	Medium	Short	Long
Compute load	High	Low	Medium	High
Cost	Low	Medium	Low	High
Resolution	High	Low	Medium	High
Depth accuracy	cm	cm	mm	mm
Applicability in the present work	Relatively inapplicable because of the experimental setup	Inapplicable due to small particle size	Inapplicable due to small particle size	Applicable

utilized in autonomous processes. For instance, Niskanen *et al.* [82] used an advanced ToF sensor to monitor urban transportation and traffic movements based on a novel approach, which plays a vital role in autonomous driving. Moreover, Ringbeck [99] applied ToF systems based on the Photonic Mixer Devices (PMC) principle to implement fast 3D measurements. However, the drawbacks or disadvantages of the ToF cameras are noticeable. Compared to other 3D systems, the present ToF technology offers a lower resolution, which leads to the missing capture of smaller particles. Moreover, the performance of the camera is relatively sensitive to external intrusions, such as strong ambient light or scattered light.

A structured light camera is also a viable alternative to conduct 3D measurements. This kind of camera pairs the camera sensor with an IR projector, which emits a particular pattern (typically lines or speckled dots) on the scene and determines the depth by analyzing the distortions of the pattern. Compared to other 3D cameras, a depth camera based on structured light attracts a spate of commercial interest owing to its cost-efficiency, *e.g.*, Microsoft Kinect. The availability of the camera also results in plenty of research around its application. Jafari *et al.*'s [42] study presents a real-time Kinect RGB-D based multi-person detection and tracking system appropriate for mobile robots. Liu *et al.* [64] designed a human detection and tracking system based on color data with depth information captured by a Kinect RGB-D camera. Structured light cameras are able to provide accurate depth information with relatively high resolution. Nevertheless, the majority of available structured light cameras fail to provide complete depth information for objects with thin surfaces, such as small particles [134]. Moreover, structured light cameras operate with a limited range of depth, and the depth error increases significantly with a distance greater than 3.5 m [128]. Besides, a structured light camera might fail to capture small particles when applied in a combustion chamber with additional light sources such as burning particles and flames.

Another option to realize 3D measurements is the plenoptic camera. Based on the plenoptic function introduced in 1997 [4], Ng [80] developed the first version of the plenoptic camera. With a microlens array in front of the sensor, the plenoptic camera is able to capture the light field of the scenes, in accordance with which the 3D information can be restored. In comparison to stereo camera systems, the plenoptic cameras involve only a single main lens and image sensor, which benefits its universal usability. Meanwhile, the plenoptic cameras precede the time-of-flight cameras in terms of resolution and robustness. In addition, Sandemann [100] proved the accuracy and adequacy of the focused plenoptic camera for long-distance measurements. Nevertheless, several problems of the plenoptic cameras remain to be solved, such as incomplete corresponding data processing techniques and required custom imaging hardware [35]. Considering comprehensive factors, *e.g.*, the construction of the test facility and the characteristics listed in Table 5.1, the plenoptic camera system is selected to capture the fuel particles.

### 2.1.4. Plenoptic camera system

The concept of a plenoptic camera, also called a light-field camera, was developed a century ago. Nevertheless, the type of camera has been introduced in several decades due to the ability limitation of the graphic processor units. In the work, a focused plenoptic camera with a microlens array consisting of microlens with different focal lengths is used, as elaborated in the following.

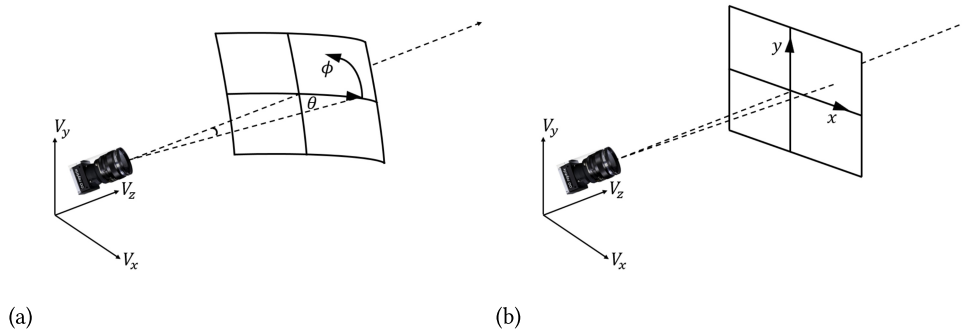
#### Plenoptic function

The plenoptic function, which was firstly introduced by E.H. Adelson and J. R. Bergen in 1997 [4], parameterize the intensity distribution  $P$  with seven parameters and takes the following form

$$P = P(\theta, \phi, \lambda, t, V_x, V_y, V_z). \quad (2.1)$$

where  $P$  is the intensity distribution,  $\theta, \phi$  stand for the angles that light rays passing through the center of the lens,  $\lambda$  is the wavelength of the light,  $t$  describes the time dimension, and  $(V_x, V_y, V_z)$  indicates the viewing position, as schematically illustrated in Figure 2.8. Alternatively, the angles in equation one can be replaced by  $x$  and  $y$ , which indicate a parameterization of the light rays in terms of spatial coordinates  $(x, y)$ . Thus, the plenoptic function becomes

$$P = P(x, y, \lambda, t, V_x, V_y, V_z). \quad (2.2)$$

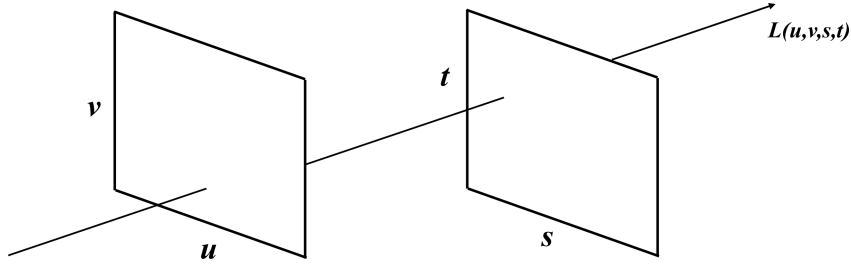


**Figure 2.8.:** Schematically depicted plenoptic function. (a) Function parametrized by angles. (b) Function parametrized by Cartesian values. [4]

In practice, the 7D plenoptic function can be simplified to 5D by neglecting the time dimension  $t$  and the wavelength  $\lambda$  since the image capture time is often of slight importance and the wavelength of a light beam changes scarcely. Marc Levoy and Pat Hanrahan parametrized a light beam with four dimensions in [58]. They described a particular light beam utilizing the intersection points of the beam via two arbitrarily positioned planes, as illustrated in Figure 2.9. The  $(s, t)$  plane is defined as the Focal Plane, and the  $(u, v)$  plane is called the Camera Plane.

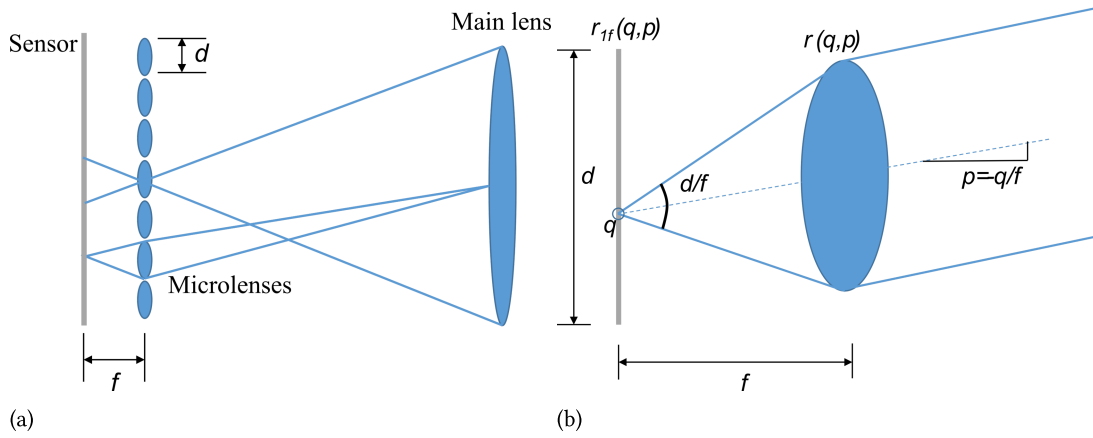
#### Plenoptic camera

In accordance with the plenoptic function, plenoptic cameras, also called light-field cameras, are developed. Compared to conventional cameras, the plenoptic camera has a microlens array in front of



**Figure 2.9.:** Parametrization of 4D light-field. [58]

the camera sensor. As illustrated in Figure 2.10, a plenoptic camera consists of a main lens, a sensor, and a microlens array that is placed at a distance of  $f$  in front of the sensor. This distance  $f$  corresponds to the focal length of the microlenses, and  $d$  denotes the aperture of the microlenses.



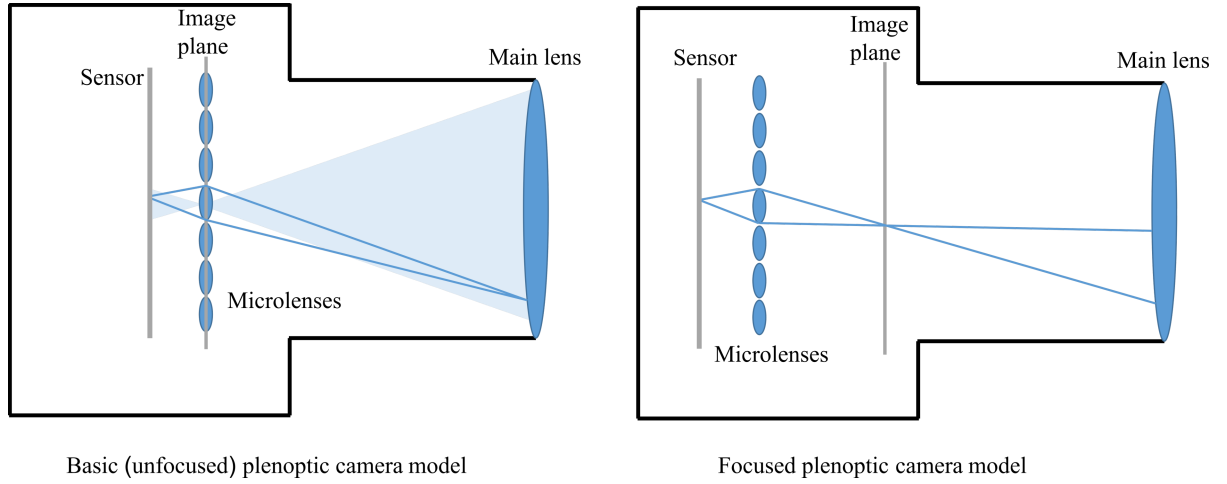
**Figure 2.10.:** Plenoptic camera model. (a) Traditional plenoptic camera. The main lens is focused at the microlens plane. (b) Single microlens model [68].

In order to explain the way the plenoptic camera functions, an arbitrary microlens is selected and denoted  $r(q, p)$  as the radiance at the microlens plane, where  $(q, p)$  is a ray factor. Hereby,  $q$  stands for the displacement away from the optical axis, and  $p$  is the slope [30]. Besides,  $r_{1f}(q, p)$  stands for the radiance at the sensor behind the microlens. According to [68], the image captured by the sensor is denoted as

$$I_f(q) = \int_p r_f(q, p) dp = \int_p r\left(q - fp, \frac{1}{f}q\right) dp. \quad (2.3)$$

With the plenoptic camera configuration, the radiance  $r(q, p)$  can be recovered from the sensor image  $I_f(q)$ .

Generally, the plenoptic cameras are based mainly on two concepts: unfocused plenoptic camera introduced by Ng [80] and focused plenoptic camera described in [67], which is also known as plenoptic camera 2.0. These two camera models are schematically depicted in Figure 2.11. In the unfocused plenoptic camera, the main lens is focused at the microlens plane and the microlenses focus at optical infinity (equivalently, the main lens) [68], while in the focused plenoptic camera, the microlenses are focused on the image created by the main lens. Thus, [67] suggests placing the microlenses at a distance of  $4/3f$  in front of the sensor instead of precisely  $f$ . The additional space is created by adding micro sheet glass. In comparison to the unfocused plenoptic camera, the focused plenoptic



**Figure 2.11.:** Plenoptic camera model. Left: Traditional plenoptic camera model. Right: Focused plenoptic camera model (Plempotic camera 2.0). [67]

camera provides a high resolution of the synthesized image, which benefits the estimation of the depth information [67] [127].

In our work, a focused plenoptic camera R12 produced by Raytrix based on the concepts demonstrated in [92] is utilized. The used plenoptic camera has a framerate of 330 frames per second and a lateral resolution of  $1536 \text{ pixel} \times 2048 \text{ pixel}$ . The microlens array of the camera consists of a microlens of three distinct focal lengths, which is defined as a multi-focus plenoptic camera and enables both a great depth of field and a high maximal lateral resolution [92]. The plenoptic camera is calibrated according to the algorithm presented in [38]. Since the accuracy of the obtained depth information from the camera is enhanced significantly with a great lateral resolution of the sensor, a big focal length of the main lens, and a short distance between sensor and object [100], a main lens with a focal length of 85 mm is selected for the measurements in the rotary kiln. Utilizing the identical lens for the experiments in the combustion chamber is not possible due to the camera's field of vision. Hence, an objective with a 35 mm focal length is chosen for the measurements in the chamber.

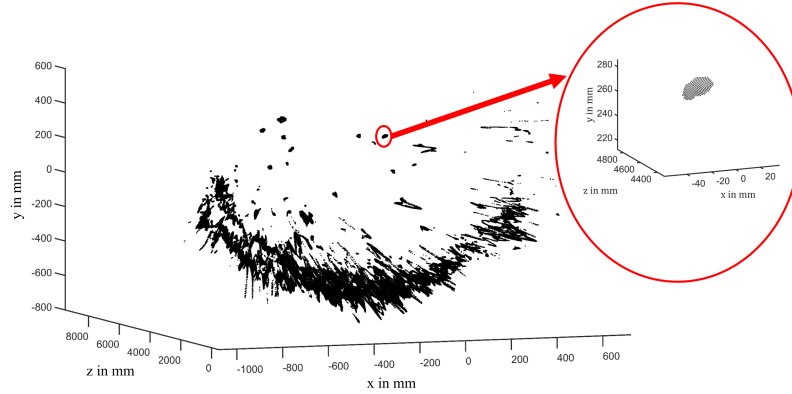
In addition, the camera software is able to generate 3D point clouds, whereby each point corresponds to a pixel in the original 2D gray value image, as illustrated in Figure 2.12. However, not all pixels can be captured spatially, as shown in Figure 2.8(b). The black color indicates no 3D coordinates available. Thus, the point cloud contains significantly less points than the resolution ( $1536 \text{ pixel} \times 2048 \text{ pixel}$ ).

### 2.1.5. Fluctuation and calibration of the depth coordinate

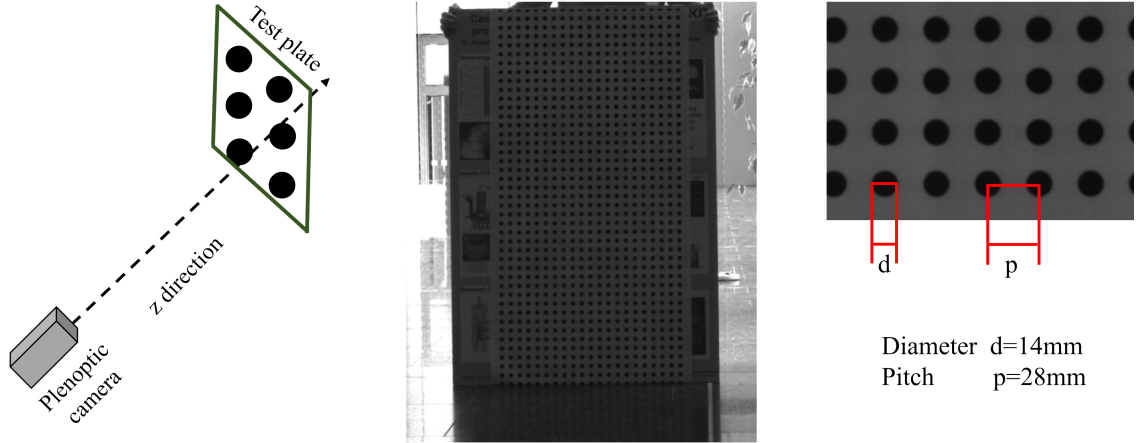
In order to acquire prior knowledge concerning the accuracy of the plenoptic camera in the depth direction before the experiment, a simple experiment using a test plate was performed. During the experiment, the plenoptic camera was fixed on a tripod and captured the pattern on the plate, as shown in Figure 2.13. The test plate was regularly arranged with solid black dots of 14 mm diameter, and the distance between adjacent dot centers is 28 mm. Initially, the test plate was placed 1 m away from the camera, then moved backward by one meter gradually until 11 m distance to the camera.

Figure 2.14 presents the 3D point clouds of the test plate obtained at different distances. Apparently, the depth values of each distance fluctuate with different intensities. In particular, the fluctuation at 4 m





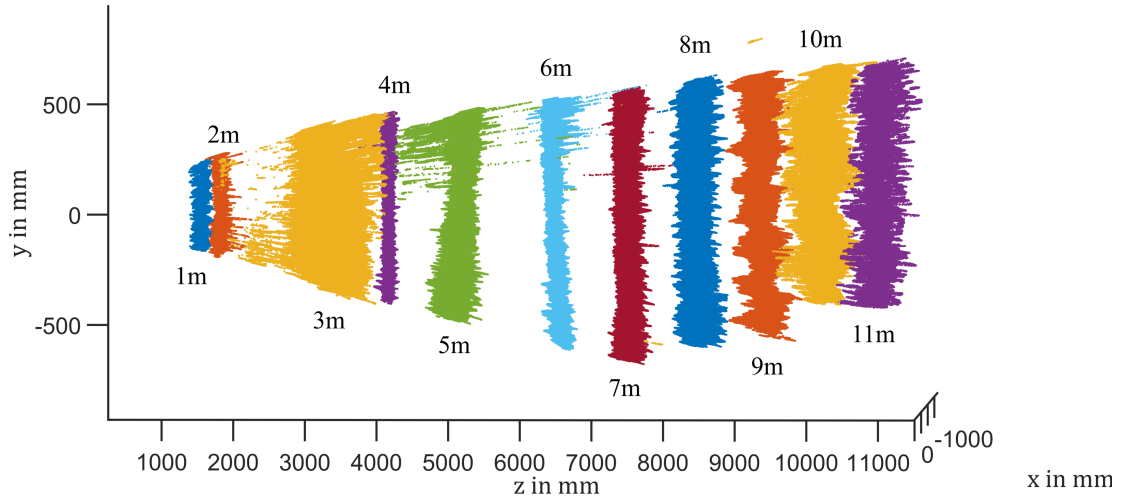
**Figure 2.12.:** An example of 3D point cloud of the rotary kiln and fuel particles captured by the plenoptic camera. The red circle on the right marks the point cloud of a single particle.



**Figure 2.13.:** Experiment setup to test the accuracy of the plenoptic camera in the depth direction. The left illustrates the test schematic; the center presents the test plate; the enlarged view of the pattern on the test plate is shown on the right side.

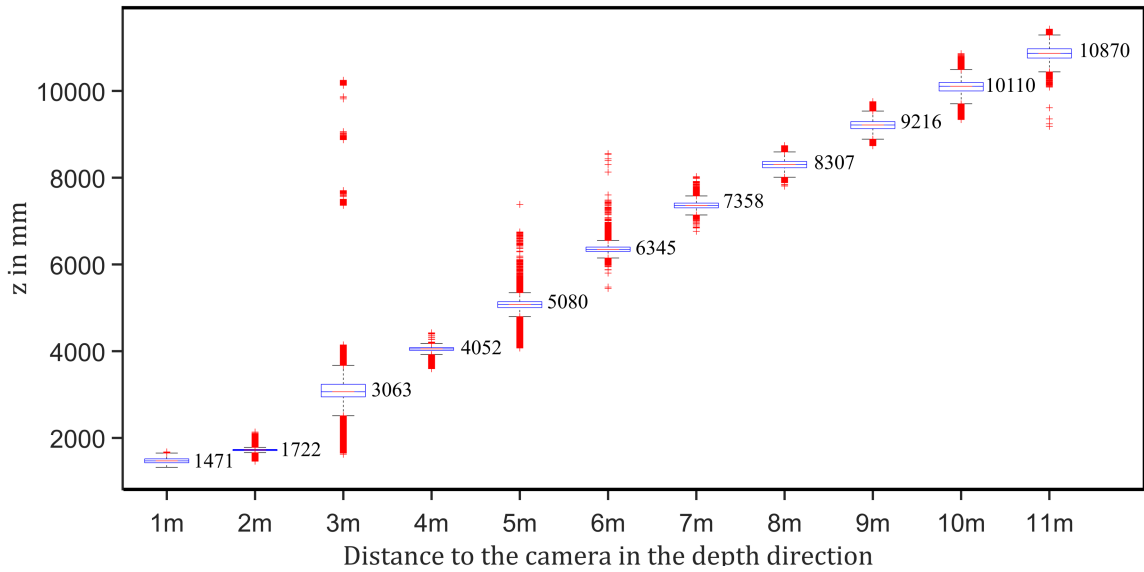
from the camera is only a few tens of centimeters, while the fluctuation at 3 m exceeds 1 m. Moreover, the average depth values obtained for each distance also deviate more or less from the actual values overall, as detailed by the boxplot in Figure 2.15.

As in the previous boxplots, the red line in the middle of each box indicates the median, and the outliers are plotted individually using the '+' symbol. As shown in Figure 2.15, the amount of outliers at 3 m is the most, and several outliers are significantly off the actual value. In addition, lots of outliers occur at 5 m and 6 m, whose degree of deviation is, however, lower than at 3 m. Besides, the depth values at 3 m are distributed dispersedly since the distance between the top and bottom lines of the box is the largest. Except for 3 m, the overall depth distribution becomes scattered as the distance increases. As for the matching degree between the obtained depth and the actual value, the median depth from 3 m to 5 m is very close to the actual value, with a difference of only a few centimeters. From 6 m to 8 m, the measured values deviate from the actual value by 0.3 m in general, but the absolute distance



**Figure 2.14.:** Acquired 3D point clouds of the test plate at each distance in the depth direction. Each time the test plate was spaced 1 m apart in the depth direction.

between the measured values remains roughly 1m. Furthermore, from 9m onward, the deviation value shrinks.

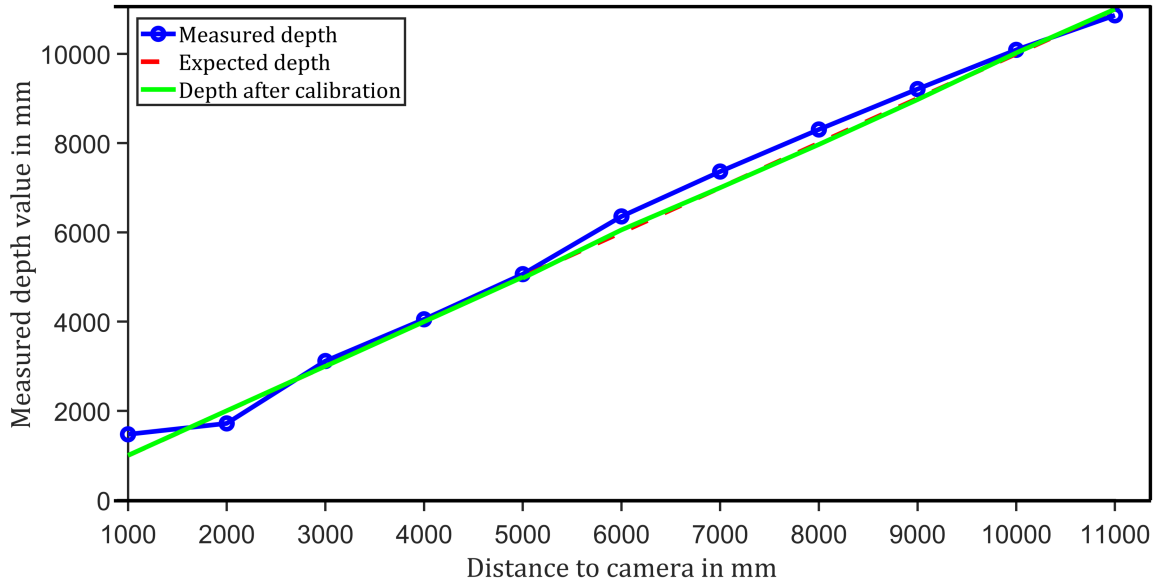


**Figure 2.15.:** Boxplot of the depth coordinates of the test plate at each distance.

Under this condition, the measured depth information is fitted by functions to approximate the actual values, facilitating subsequent tracking and analysis of the behaviors. Given that the segmented character of the measured values is pronounced, the estimation is also carried out in segments. The segmentation and the corresponding functions are referred to the following equations,

$$f(z) = \begin{cases} -0.002055z^2 + 10.65z - 10230 & \text{if } z < 3000\text{mm} \\ -0.00004158z^2 + 1.367z - 857.5 & \text{if } z \in [3000, 5000]\text{mm} , \\ 0.00004504z^2 + 0.3219z + 2189 & \text{if } z > 5000\text{mm} \end{cases} \quad (2.4)$$

where  $z$  is the captured depth coordinate by the camera. The measured depths are transformed by the polynomials, and the result of the transformation, together with its comparison with the pre-transformation, are presented in Figure 2.16. The blue line with circles represents the measured values, and the green line is the transformed depth. The red dashed line stands for the expectations, *i.e.*, the measured values equal the actual values. As indicated by Figure 2.16, the transformed depth exhibit high agreement with the expected values. This transformation is also regarded as a preliminary calibration of the measured depth coordinates and is applied to the experiments conducted in the study.



**Figure 2.16.:** Depth information graphic. The blue line is the average measured value, the red dashed line is the expected value, and the green line is the depth value after calibrating the measured value by the functions.

## 2.2. Particle detection

Object detection is increasingly becoming a vital task in computer vision and image processing. Many researchers contribute to developing algorithms concerning object detection from macro to micro based on 2D gray value images. Shantaiya *et al.* [104] reviewed and discusses various 2D detection approaches for recognizing relatively big single or multiple objects, such as pedestrians and vehicles, including, *e.g.*, feature based approaches, template based methods, classifier based, and motion based algorithms. Kulchandani *et al.* [52] provided a brief classification of approaches to identifying moving objects, including background subtraction, frame differencing, temporal differencing, and optical flow. Furthermore, the authors also reviewed the recent research trends to detect moving objects and compared them regarding positive aspects and limitations. Zou [138] compared and discussed various object detection approaches based on traditional machine learning techniques and deep learning methods, such as Harris corner features extraction and general Hough transform. Meanwhile, object detection utilizing deep learning methods is also summarized and introduced. Moreover, Zou *et al.* [139] extensively and comprehensively reviewed beyond 400 papers on object detection of a particular class (such as humans, animals, or vehicles) over the last two decades. The review covers a variety of topics, for instance, milestone detectors in history, detection datasets, speed-up techniques, and current state-of-the-art techniques.

Compared to detecting apparent objects, detecting multiple small objects, such as particles and cells, is more complex and challenging due to sophisticated gray value distributions. Therefore, existing approaches for multiple object detection might not definitely be appropriate for detecting small particles. Additionally, detecting small objects without specific properties usually entails a preprocessing of background estimation and subtraction (Chapter 2.2.1) so that the detection can be performed on the foreground image to facilitate accuracy. This work focuses on identifying multiple small fuel particles without specific texture, shape, or color features as a consequence of their small sizes. Detecting these particles entails sufficient illumination before combustion; meanwhile, overexposure should be avoided during combustion. For simple applications, threshold selection approaches, such as the OTSU selection method (Chapter 2.2.2), which was proposed in 1979 and performs automatic image thresholding by selecting a single gray value threshold [89], could achieve satisfactory accuracy and efficiency. Matthes *et al.* [73] detected burning fuel particles inside a combustion chamber by applying a simple threshold to estimated foreground models. Further practical usage is exemplified in [126], where the authors propose a detection and tracking method of firing particles from an image sequence based on gray level change. To deal with complex detection cases, the feature extraction method Scale Invariant Feature Transform (SIFT) [65] (Chapter 2.2.3) that extracts highly distinctive image features and matches them to features from particular objects in order to detect the objects is able to provide precise outcomes, as exemplified in [114]. Unlike several feature based approaches that are sensitive to the image geometric features and can be affected by image translation, scaling, and rotation [138], SIFT is invariant to image translation, scaling, and rotation [65]. Different from feature generation, Kass *et al.* proposed active contour models using energy minimization aiming at detecting edges, lines, and subjective contours in images [46]. Principally, the detection of microscopic objects, for instance, microscopic particles and cells, shares the analogous solutions as detecting fuel particles. Therefore, several presented methods in the literature could be applied to detect RDF particles as well. A comprehensive review of nuclei detection approaches is illustrated in [41], including diverse thresholding approaches, the above-mentioned active contour model, and several clustering methods, which is elaborated later in the section. Another detailed introduction of various methods for particle detection and classification in electron microscopy is given in [81], such as cross correlation based approaches [76], texture-based methods [55], and artificial neural networks [74]. With the development of modern computer techniques, the appliance of deep learning methods, especially convolutional neural networks (CNN), is commonly discussed and investigated in image processing. Girshick *et al.* [32] proposed an approach named Regions with CNN features (R-CNN) that combines region proposals with CNN for object detection. The R-CNN approach achieves significant performance enhancement proved by the inventors. Later, the approach is modified and optimized to Fast R-CNN [31] and Faster R-CNN [97] to improve computational efficiency. Further, Tong *et al.* [112] detailedly reviewed small object detection approaches based on deep learning methods with respect to five aspects, including multi-scale feature learning, data augmentation, training strategy, context-based detection, and GAN-based detection. The artificial neural network might provide precise performance compared to several conventional methods, such as the thresholding and cross-correlation approach that depends intensely on local gray value distribution and could vary significantly in actual industrial combustion processes. Nevertheless, it entails relatively high computational costs and learning datasets.

Unlike the above-mentioned gray value based detection algorithms, several researchers conducted investigations into object detection by clustering. In case input images are in the form of point clouds, the clustering algorithms are capable of classifying the points into corresponding objects based on the distribution of the points. One well-known clustering method is k-means clustering presented in [36]. The basic idea of the k-means clustering is to divide points into clusters by minimizing the within-cluster sum of squares. K-means clustering relies not only on pixel values but also on clusters' initialization, limiting its utilization. Hence, the approach is later generalized in [13]. This generalized

algorithm is defined as mean shift, which shifts points to the average of points in its neighborhood. To efficiently apply the clustering concept to a large spatial database, Ester *et al.* [26] developed a novel clustering algorithm DBSCAN (Chapter 2.2.4), which is the abbreviation for Density-Based Algorithm for Discovering Clusters in Databases with Noise. According to the distribution and position of the points, DBSCAN classifies them into core points, border points, and noise. Objects are identified by connecting core points and border points with respect to two pre-determined parameters. So far, DBSCAN has been slightly modified several times. For instance, the Ordering Points To Identify the Clustering Structure (OPTICS) is one of the DBSCAN variants that focus on finding hierarchical clustering results [7]. Practical applications of the clustering approaches to object detection are illustrated in [64], where a new idea using mean shift clustering candidate segmentation for people detection based on the point clouds gathered by an RGB-D camera is introduced. Besides, Chiang *et al.* [15] proposed a fast clustering method on the basis of the DBSCAN algorithm to realize traffic detection for self-driving technology. Differing from the density based clustering approaches, such as DBSCAN, clustering methods based on the adjacency of data points, for instance, clustering based on morphological operations [120], have been researched recently. According to the study, the method employs morphological dilation to connect adjacent points to form connected domains and is able to achieve enhanced accuracy in several usage cases. Another clustering idea is center-based clustering, for instance, the Quasi-Cluster Centers (QCC) algorithm [39] that builds clusters by defining quasi-cluster centers with a database and extending them by the corresponding sparse neighbors.

In addition to single information based (gray value point cloud) detection approaches, researchers attempted to fuse various 2D- and 3D information to realize more possibilities in image processing recently. For instance, Halima *et al.* [34] presents an approach to detecting and tracking human heads by fusing depth and 2D thermal information. Besides, an object detection approach based on a particle filter with integrated information, such as color, texture, and depth information, is introduced for the detection of everyday items [8]. Furthermore, Justen [44] conducted 3D-image processing, *e.g.*, 3D object reconstruction and classification, utilizing gray value images and individual depth information, which is considered a novel 2D/3D camera concept. Another application of information fusion for object detection is demonstrated in [86], where real-time object detection is realized by fusion of standard RGB data with depth information that can increase the performance of current detection networks.

Various detection approaches show sufficient adequacy for diverse applications, such as detections of single or multiple objects, macroscopic objects or microscopic particles, and objects with specific features or generally similar objects. For simple applications, such as single macroscopic object detection with specific features, the majority of the approaches is able to achieve reasonable accuracy and performance. Nevertheless, adaptations are usually entailed when detecting objects in complex situations, such as in our project, where plenty of small particles without particular features should be detected inside a rotary kiln with considerable illumination changes. Moreover, the particles could be ignited during their flights, enhancing further the difficulty of the detection procedure.

### **2.2.1. Background subtraction**

Background subtraction aims to model background images subtracted by each corresponding current frame to detect foreground objects. The background image can be modeled both statically and dynamically. Generally, static background models are simple, while dynamic models illustrate high accuracy. One typical example of the static models is the static mean background, which computes the pixel-wise arithmetic average of the gray values from a set of selected images. Dynamic models contain a wide range of approaches, for instance, the running Gaussian average [122] and the mixture of Gaussians [107]. Piccardi [93] reviewed seven common background subtraction approaches and

highlights their merits and demerits. Later, Sobral *et al.* [105] conducted a more comprehensive review of background subtraction algorithms and evaluates them with synthetic and real videos.

### Static frame difference

A straightforward way to create a static background model without moving objects is to select a set of captured images and compute their temporal mean or median value pixel-by-pixel. The thereby computed image remains unchanged in a certain time interval and is thus considered as a static background model, which will then be subtracted by each current frame for the purpose of foreground detection. The basic idea of the method is that the background dose not change or changes more slowly than the foreground, as in the present work, where the radial intensity of the rotary kiln walls varies significantly slightly compared to the rapid movements of the fuel particles. The selected time interval for computing the static background model ought to be sufficiently short so that the slight changes in the background can be captured. Meanwhile, the chosen interval should be so long that the fast dynamics of the foreground do not appear in the background model. This method requires a low computational cost. Nevertheless, the accuracy of the foreground detection might decrease significantly with an enormously changed illumination.

### Moving frame difference

Instead of a computed static background model, the previous frames can also be regarded as firsthand background models, defined as the frame difference. The frame difference is able to rectify the problem of applying static frame difference, whereby a high framerate could minimize the impact of illumination changes. As illustrated in [103], the 2-frame diffence at time  $t$  is defined as

$$D_t = |I_t - I_{t-i}|, \quad (2.5)$$

where  $I_t$  denotes the captured 2D gray value image at time  $t$ , and  $i$  stands for the selected frame interval. The frame difference could highlight the pixel locations, whose intensity change in the two frames. In practice, the frame difference might be modified defining a threshold to eliminate slight changes depending on the application. In the present work, where unburning particles with slight brightness persist, using a threshold value will adversely affect the detection performance.

In addition to 2-frame difference, a 3-frame difference model is also proposed in the work as follows,

$$D_t = \frac{1}{2} (|I_t - I_{t-i}| + |I_t - I_{t+i}|). \quad (2.6)$$

The 3-frame difference model takes both the previous frame and the following frame into consideration to achieve a precise localization of local intensity changes.

Generally, the frame interval  $i$  takes the value of 1. Whereas, it can also raise to 2 or 3 to possibly identify slow-moving objects. The frame difference  $D_t$  is simply the foreground image at time  $t$ . This approach is relatively efficient for the acquisition of the foreground image. However, the approach fails to segment non-moving and slow-moving objects accurately.

**Adaptive background learning**

To adapt the background model temporally, some authors recommend initializing and maintaining the background utilizing the arithmetic mean pixel-by-pixel [105]. The initial background model  $B_0$  is defined as

$$B_0 = \frac{1}{l} \sum_{t=1}^l I_t, \quad (2.7)$$

where  $l$  stands for the amount of the frames and  $I_t$  represents the current frame. It should be noted that the start time of the initial background model does not necessarily have to start from the first frame, as defined in the equation. Actually, the time period can be selected arbitrarily according to the use cases. Subsequently, the background model will then be adapted by the recursive equation

$$B_t = (1 - \alpha)B_{t-1} + \alpha I_t. \quad (2.8)$$

Here  $B_t$  is the background model at time  $t$ , and  $\alpha$  is the learning rate. Thus, this approach is defined as adaptive background learning, which performs background maintenance to adapt changes in the scene. Ultimately, the active background model corresponds to a discrete first order temporal low-pass filter (PT1) of the gray values in each pixel, where the time constant can be adjusted by the parameter  $\alpha$ .

**Running Gaussian average**

Wren *et al.* [122] proposed an approach to model background on the basis of fitting a Gaussian probability function on the gray values. To avoid fitting the function at each new frame, the authors suggest computing a running average instead as follows,

$$\mu_t = \alpha I_t + (1 - \alpha)\mu_{t-1}. \quad (2.9)$$

Here,  $I_t$  is the current frame,  $\mu_t$  is the previous average image, and  $\alpha$  represents an empirical factor. If the gray value of a pixel in the current frame  $t$  satisfies the following condition

$$|I_t - \mu_t| > k\sigma_t, \quad (2.10)$$

the pixel will be classified as foreground. Otherwise, the pixel is considered as background. The standard deviation  $\sigma_t$  can be computed similarly as  $\mu_t$  [93]. Compared to adaptive background learning, the running Gaussian average algorithm has the additional step of dynamic foreground selection introduced in the above equation. Therefore, the foreground image comprises a set of pixels that fulfill equation 2.10 instead of the difference between the current frame and the background model.

**2.2.2. 2D-OTSU threshold selection method**

OTSU [89] derived a nonparametric method to select a threshold automatically from a gray-level histogram for extracting foreground objects. In the method, only the gray-level histogram is relevant as input. Gray values of a given image are represented in several gray levels from low to high. The number of pixels at each level is counted and normalized by the total number of pixels afterwards. Then the pixels are dichotomized into background and foreground objects by a threshold at a certain level. To elaborate the method, the following assumptions are made: The pixels from an image are divided into  $L$  gray levels and  $P_i$  denotes the normalized number of pixels at the level  $i$ . Then a threshold

level  $k$  is selected and thus, the corresponding normalized number of pixels of the background and the foreground are

$$\omega_0 = \sum_{i=1}^k P_i \quad (2.11)$$

$$\omega_1 = \sum_{i=k+1}^L P_i. \quad (2.12)$$

$\omega_0$  is regarded as the zeroth-order cumulative moment of the histogram up to the  $k$ th level. If  $P_i$  is weighted by its gray level  $i$  and the weighted  $P_i$  are totaled up from  $i = 1$  to  $i = k$ , the first-order cumulative moment of the histogram is obtained as

$$\mu(k) = \sum_{i=1}^k iP_i. \quad (2.13)$$

Then variances of the foreground and the background are given by

$$\sigma_0^2 = \sum_{i=1}^k \frac{(i - \sum_{i=1}^k \frac{iP_i}{\omega_0})^2 P_i}{\omega_0} \quad (2.14)$$

$$\sigma_1^2 = \sum_{i=k+1}^L \frac{(i - \sum_{i=k+1}^L \frac{iP_i}{\omega_1})^2 P_i}{\omega_1}. \quad (2.15)$$

Following criterion measures are introduced and thereby evaluate the selected threshold  $k$  as

$$\lambda = \frac{\omega_0 \omega_1 (\sum_{i=k+1}^L \frac{iP_i}{\omega_1} - \sum_{i=1}^k \frac{iP_i}{\omega_0})^2}{\omega_0 \sigma_0^2 + \omega_1 \sigma_1^2} \quad (2.16)$$

$$\kappa = \frac{\sum_{i=1}^L (i - \mu(L))^2 P_i}{\omega_0 \sigma_0^2 + \omega_1 \sigma_1^2} \quad (2.17)$$

$$\eta = \frac{\omega_0 \omega_1 (\sum_{i=k+1}^L \frac{iP_i}{\omega_1} - \sum_{i=1}^k \frac{iP_i}{\omega_0})^2}{\sum_{i=1}^L (i - \mu(L))^2 P_i}. \quad (2.18)$$

Hence, the problem is converted to the calculation of an appropriate threshold that maximizes one of the criterion measures.  $\eta$  is the simplest among the three measures in terms of  $k$ , since  $\eta$  is based on first-order statistics. Therefore, an optimal threshold  $k^*$  is achieved by maximizing  $\eta$  as

$$\eta(k) = \frac{[\mu(L)\omega_0 - \mu(k)]^2}{\sum_{i=1}^L (i - \mu(L))^2 P_i \omega_0 (1 - \omega_0)} \quad (2.19)$$

and the optimal threshold  $k^*$  equals

$$\eta(k^*) = \max \eta(k), 1 \leq k < L. \quad (2.20)$$

### 2.2.3. 2D-Scale invariant feature transform (SIFT)

Scale invariant feature transform (SIFT) is an approach for image feature generation and was introduced by Lowe in 1999 [65]. Features, which are extracted by SIFT, are invariant to image translation, scaling, and rotation, and are as stable as possible against illumination and local affine distortions [65] [66]. The



features are highly distinctive and thus, can be correctly matched to a large database of features from certain objects, which offers the opportunity to detect objects. Exploiting the fact that the difference-of-Gaussian filter used in the SIFT algorithm is quite similar to the gray scale of a particle in the image, the method has the ability to detect fuel particles in the work [114]. As demonstrated in [66], the following four steps are essential for the generation of the features: scale-space extrema detection, keypoint localization, orientation assignment, keypoint descriptor.

### Scale-Space Extrema Detection

Lindeberg [63] has proved that under some reasonable assumptions, the Gaussian kernel is the only possible kernel for scale space analysis. Hence, the convolution of an image  $I(x, y)$  with the Gaussian function  $G(x, y, \sigma)$  will be calculated as

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y), \quad (2.21)$$

with

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \quad (2.22)$$

The difference of two Gaussian functions of nearby scales is defined as the difference-of-Gaussian function (DoG) as follow

$$DoG = G(x, y, k\sigma) - G(x, y, \sigma), \quad (2.23)$$

where  $k$  is a constant multiplicative factor. In the light of [66], the DoG can be approximated by

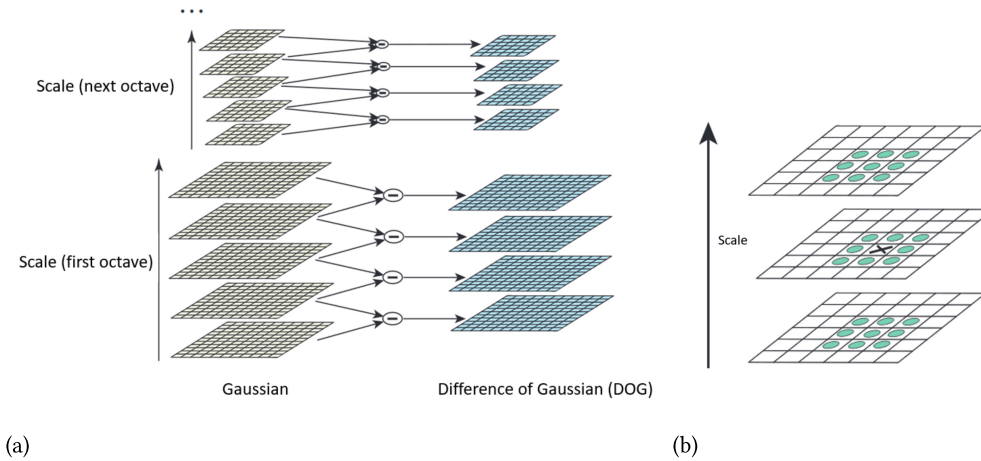
$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k - 1)\sigma^2 \nabla^2 G \quad (2.24)$$

and the approximation has shown almost no impact on the extrema detection or localization in the application. Figure 2.17(a) illustrates the main concept of the approach SIFT. A set of Gaussian functions, which are separated by the constant factor  $k$ , are convolved with the input image and the convolutions of two adjacent scales are subtracted to obtain the difference-of-Gaussian images afterward. The process will be repeated for the next octave by utilizing a resampled image that takes every second pixel in each row and column from the original image. In order to detect keypoint locations, the local maxima and minima of the difference-of-Gaussian images should be sought out at first. For this purpose, each sample point from the image is compared to its eight neighbors and nine neighbors in both scales above and below, as illustrated in Figure 2.17(b). Only if the sample point is smaller or larger than all the 26 neighbors, it can be considered as local extrema.

### Accurate keypoint localization

After detecting the local extrema, a further step for accurate localization of the keypoints by fitting it to the nearby data is implemented. A simple approach is to locate the keypoints at the location and scale of the central sample point, as delineated in [65]. However, this approach is not adequate for lack of matching performance and stability. Therefore, Brown and Lowe [12] developed a method for fitting a 3D quadratic function to the local sample points to locate the local extrema. The method utilizes the Taylor expansion of  $D(x, y, \sigma)$ , which denotes the convolution of DoG and the input image  $I(x, y)$ :

$$T_2 D(\mathbf{x}; \mathbf{x}_s) := D(\mathbf{x}_s) + \frac{\partial D(\mathbf{x}_s)}{\partial \mathbf{x}} (\mathbf{x} - \mathbf{x}_s) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_s)^T \frac{\partial^2 D(\mathbf{x}_s)}{\partial \mathbf{x}^2} (\mathbf{x} - \mathbf{x}_s) \quad (2.25)$$



**Figure 2.17.:** Overview of SIFT feature extraction. (a) The scale space of SIFT. (b) Comparing a pixel to its neighbors to detect maxima and minima of the difference-of-Gaussian images [66].

where  $\mathbf{x} = (x, y, \sigma)^T$  is the scale-space coordinate,  $\mathbf{x}_s = (x_s, y_s, \sigma_s)^T$  is the sample point, and  $(\mathbf{x} - \mathbf{x}_s)$  is the offset from the sample point. By taking the derivative of this function with respect to  $\mathbf{x}$  and setting it to zero the localization of the local extremum can be obtained as

$$\hat{\mathbf{x}} = -\frac{\partial^2 D(\mathbf{x}_s)^{-1}}{\partial \mathbf{x}^2} \frac{\partial D(\mathbf{x}_s)}{\partial \mathbf{x}} + \mathbf{x}_s. \quad (2.26)$$

The function value at the maximum  $\hat{\mathbf{x}}$  can be calculated by replacing  $\mathbf{x}$  by  $\hat{\mathbf{x}}$  in Equation (2.22) that results in

$$D(\hat{\mathbf{x}}) = D(\mathbf{x}_s) + \frac{1}{2} \frac{\partial D(\mathbf{x}_s)^T}{\partial \mathbf{x}} (\hat{\mathbf{x}} - \mathbf{x}_s), \quad (2.27)$$

which helps to discard the extrema with low contrast. Not only affect the keypoints with low contrast the stability of the SIFT approach, but the strong edge response of the DoG function also shows an impact on the stability. Therefore, Lowe [66] proposes to eliminate the edge response by exploiting a  $2 \times 2$  Hessian matrix,  $\mathbf{H}$ , computed at the location and scale of the keypoint. The Keypoint will be regarded as a valid keypoint only if the relation between the trace and the determinant of the Hessian matrix satisfies a certain condition that

$$\frac{\text{tr}(\mathbf{H})^2}{\det(\mathbf{H})} < \frac{(r+1)^2}{r}, \quad (2.28)$$

where  $r$  represents a user-defined threshold.

### Orientation Assignment

Once the location of the keypoint has been accurately determined, the next step is to assign a consistent orientation to the keypoint. Lowe [66] recommended an approach for orientation assignment that provides the most stable outcomes among several approaches in experimentation. The keypoint localization, especially the scale of the keypoint, contribute to the selection of  $L(x, y, \sigma)$  with the closest scale and thereby,  $L(x, y, \sigma)$  is converted into  $L(x, y)$ , which is scale-invariant. The gradient magnitude  $m(x, y)$  and orientation  $\theta(x, y)$  of each image sample  $L(x, y)$  at this scale is computed as:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (2.29)$$

$$\theta(x, y) = \tan^{-1}((L(x, y+1) - L(x, y-1)) / (L(x+1, y) - L(x-1, y))). \quad (2.30)$$

Image samples within a certain region around the keypoint are weighted by its gradient magnitude and a Gaussian-weighted circular window. Based on the weighted sample points an orientation histogram is formed afterward and Peaks in the histogram correspond to directions of local gradients. By fitting a parabola to the 3 histogram values closest to the peak, the accurate interpolated peak position can be obtained.

### **The local image descriptor**

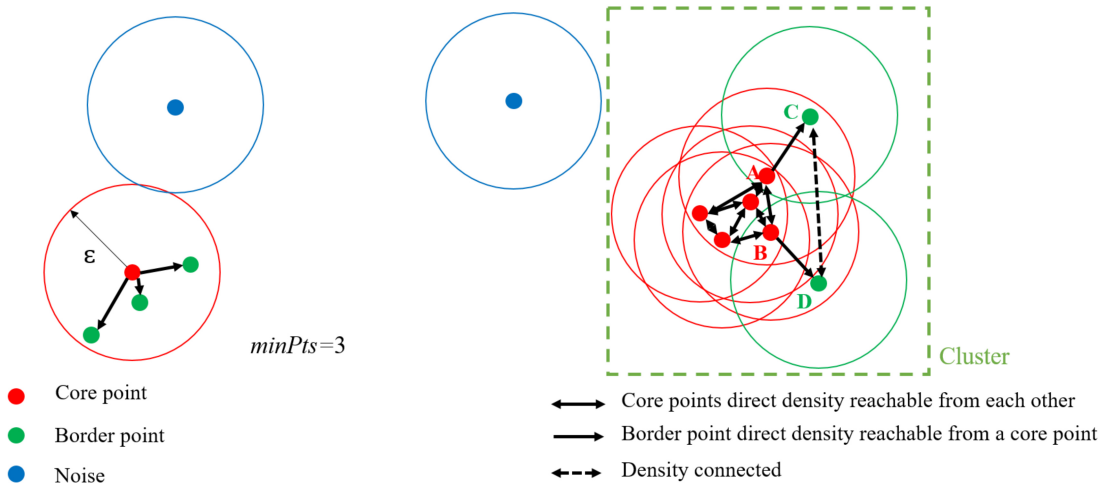
After the three previous steps, each keypoint possesses its location, scale, and orientation. Therefore, the next operation is to compute a local image descriptor that is not only distinctive but sufficiently robust against some changes as well, for instance, change in illumination. Based on the idea from Edelman *et al.* [25], Lowe [66] describes an approach to creating a keypoint descriptor. At first, the gradient magnitude and orientation at each sample point within a region around the keypoint are computed and weighted by a Gaussian window. The samples are then added and accumulated into an orientation histogram over several sample regions, which results in a set of arrows with certain lengths and directions. The length of each arrow corresponds to the sum of the gradient magnitude with a certain direction in the defined region.

#### **2.2.4. DBSCAN clustering method**

The DBSCAN algorithm, which was firstly presented in 1996, is a density-based algorithm for discovering clusters in a large spatial database with noise [26]. The algorithm is designed to discover clusters with arbitrary shapes utilizing only two user-defined parameters, which contribute to reducing the requirement of a priori domain knowledge. Moreover, the DBSCAN algorithm also provides efficiency benefits on large databases. The present work applies the DBSCAN algorithm to detect fuel particles based on 3D point clouds.

#### **DBSCAN cluster model**

With the help of two defined parameters: radius  $\epsilon$  and  $minPts$ , points in a given point cloud will be classified into three types: core point, border point, and noise. Figure 2.18 illustrates schematically the concepts. Points with more than  $minPts$  neighbors within the radius  $\epsilon$  (including the query point) are deemed to be core points, for instance, point A and B. All points within the radius of a core point are considered direct density reachable from the core point. Non-core points, *e.g.* points C and D, which are direct density reachable from core points, are defined as border points. Border point C is not direct density reachable from core point B but is direct density reachable from core point A, which is again direct density reachable from core point B. In this case, border point C is defined to be density reachable from core point B. Density-reachability is an extension of direct density-reachability with which more points can be connected. For instance, border points C and D are both (direct) density reachable from core point A and are therefore considered density-connected. The notions of density-reachability and density-connectivity enable the definition of a cluster. A cluster is a set of density connected points, *i.e.* all points within the same cluster are density-connected. Points that are not density reachable from any core point are regarded as noise and hence do not belong to any cluster.



**Figure 2.18.:** Illustration of the DBSCAN Cluster Model.

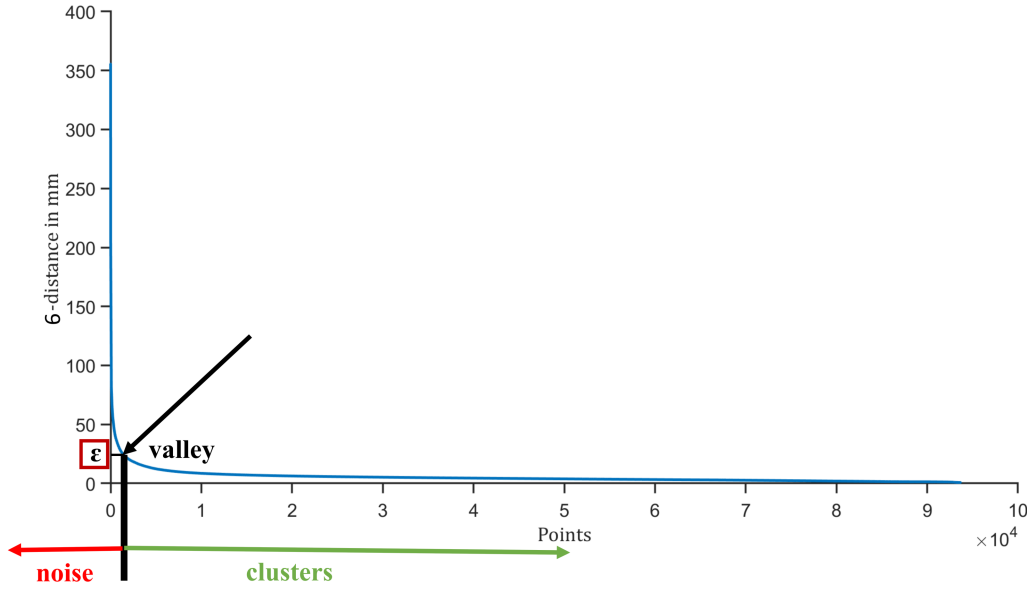
### DBSCAN parameters

The performance of this density-based clustering algorithm depends highly on the selection of the presented parameters: radius  $\epsilon$  and  $minPts$ . Compared to radius  $\epsilon$ ,  $minPts$  is an easier-to-set parameter [102]. Sander *et al.* [101] recommended setting  $minPts$  as twice as the dataset dimensionality, i.e.,  $minPts = 2 \times dim$ . For a 3D dataset, the default value of  $minPts$  is 6. In case that the datasets involve a lot of noise and duplicates or the datasets are very large, an increasing  $minPts$  is necessary.

The value of the radius  $\epsilon$  depends on the distance function. Thus, Ester *et al.* [26] suggested choosing the radius based on the distance to the fourth nearest neighbor for 2D datasets. Sander *et al.* [101] provided a general heuristic for choosing the radius and recommend utilizing the  $2 \cdot dim$  nearest neighbor (including the query point). The distance of each point within the database to its  $2 \cdot dim$ -th nearest neighbor is computed and sorted in descending order as schematically illustrated in Figure 2.19. As defined by the DBSCAN algorithm, the points with a  $2 \cdot dim$ -distance equal or smaller than  $\epsilon$  are considered core points, whereas the points with a higher  $2 \cdot dim$ -distance are probably noise. Therefore,  $\epsilon$  is selected as a threshold that distinguishes points. As recommended, this threshold value is the first “valley” of the sorted  $2 \cdot dim$ -distance graph that is relatively simple to find in the graphical representation. Nevertheless, the radius selection should also be in accordance with specific conditions in reality. For instance, if the clustering method is used in discovering small particles with a high density distribution, a small  $\epsilon$  and a high  $minPts$  should be set accordingly. Conversely, to identify large objects with relatively sparse distribution, a sizeable  $\epsilon$ , and a small  $minPts$  might be suitable.

### DBSCAN algorithm

In order to compute clusters according to the cluster model and the parameters explained above, the algorithm starts with scanning an arbitrary point in a given dataset. If a core point A is discovered, all points that are density connected with A will be visited and added to the cluster. Non-core points or points already assigned to a cluster will be skipped if they are visited by the scan. A border point can



**Figure 2.19.:** Sorted 6-distance graph of points in descending order according to [26].

belong to more than one cluster, but it could nevertheless be assigned to only the first one of them. Therefore a cluster might contain less than  $minPts$  points.

In principle, the result of the DBSCAN algorithm is invariant to the scanning sequence. However, it can vary slightly if the database is permuted. For instance, the label of a cluster depends on its discovered order, and as described in the first paragraph, a border point is assigned to the first cluster, from which it is reachable. However, the cluster labels show almost no influence on the outcomes in most situations, and the assignment of a few joint border points is only of little interest. Therefore, trying different permutations of the dataset is unnecessary.

### 2.2.5. 3D clustering based on morphological operations

Unlike DBSCAN, which builds clusters in accordance with the local point density, Wang [120] proposed a novel clustering method based on the adjacency of the points. Considering that points of the same class are closer to each other than those of different classes, the approach of 3D clustering based on morphological operations (MO-based clustering) is developed to provide robust and generic clustering performance. For a given 3D point cloud  $p$  with the length of  $N$ , the point cloud is denoted as

$$p = (p_x(i), p_y(i), p_z(i)), i = 1, 2, \dots, N. \quad (2.31)$$

As the first step, the points of  $p$  are shifted and scaled to a positive range by the following equation

$$P = R \left( \frac{p - \min(p)}{\max(p)} \right), \quad (2.32)$$

where  $R$  stands for the selected range of the scaled points and  $P$  is the shifted and scaled point cloud. Subsequently, the points of the positive point cloud  $P$  are rounded to be integers and then transformed into a grid as

$$G(x, y, z) = \begin{cases} 0 & \forall x \in X, y \in Y, z \in Z \\ 1 & \forall (x, y, z) \in [P] \end{cases} \quad (2.33)$$

with  $X = 1, 2, \dots, R, Y = 1, 2, \dots, R, Z = 1, 2, \dots, R$  denoting the indexes in the three dimensions.

With the completion of the steps, the location in the grid is binary assigned. The locations with the value of one are then expanded by setting all its adjacent locations with the value of one. The process is repeated until points in the same class merge into one connected domain. The expansion of the “1” locations corresponds to the morphological dilation, and therefore, the algorithm is named a morphological operations based clustering method. The iteration of the morphological process stops when the number of connected domains equals the number of clusters. Afterward, the connected domains in the grid are labeled with sequential numbers to generate a labeled grid  $G_L$  as

$$G_L(x, y, z) = j, \quad (2.34)$$

and  $j$  denotes the  $j$ th connected domain. To distinguish the domains according to their sizes, a size threshold is defined as

$$T_s = \frac{1}{M} \sum_{j=1}^M |X_j| = \frac{1}{M} \sum_{j=1}^M |Y_j| = \frac{1}{M} \sum_{j=1}^M |Z_j|, \quad (2.35)$$

where  $|X_j| = |Y_j| = |Z_j|$  denote the total number of elements in the  $j$ th labeled domain, and  $M$  stands for the amount of the labeled domains. If one domain contains less than  $T_s$  elements, the domain is regarded invalid and deleted. After removing the invalid domains, the remaining domains are labeled again. The iterative dilation is repeated until the number of connected domains equals the number of clusters. Finally, the points in the remaining labeled domains can be denoted as

$$C(l) = (C_x(l, k), C_y(l, k), C_z(l, k)), k = 1, 2, \dots, K_l, l = 1, 2, \dots, L. \quad (2.36)$$

$L$  represents the amount of labeled clusters and  $K_l$  stands for the number of points in the  $l$ th cluster. For each remaining positive point, its nearest Euclidean distance to each labeled cluster is computed. The point is then assigned to the cluster that yields the smallest Euclidean distance to the point. After all the positive data points are labeled with a number, they are transformed back into the original range using the inverse function of Equation 2.32.

### 2.2.6. 3D clustering based on quasi-cluster centers

Another spatial clustering option is introduced by Huang *et al.* [39], which classifies points following the idea that the density of a cluster center is the maximum among its  $K$  neighbors or reverse neighbors. For an input point cloud  $p$ , the  $K$ -distance of the  $i$ th point in  $p$  is denoted as  $Dist_K(p_i)$ , which stands for the distance between  $P_i$  and its  $K$ th nearest neighbor. Based on  $Dist_K(p_i)$  the density of  $p_i$  is defined as

$$Den(p_i) = \frac{1}{Dist_K(p_i)}. \quad (2.37)$$

The points with a distance to  $p_i$  not exceeding  $Dist_K(p_i)$  compose the  $K$  nearest neighbor of  $p_i$  and are denoted  $KNN(p_i)$ . Correspondingly, the concept of reverse  $K$  nearest neighbor of  $p_i$  is also defined. The point with a distance to  $p_i$  not more than its  $K$ -distance counts towards the  $RKNN(p_i)$ . The two concepts are mathematically described as

$$KNN(p_i) = \{p_j | d(p_i, p_j) \leq Dist_K(p_i)\} \quad (2.38)$$

and

$$RKNN(p_i) = \{p_j | d(p_j, p_i) \leq Dist_K(p_j)\}, \quad (2.39)$$

where  $p_j$  stands for a point within  $p$  and  $d(p_i, p_j)$  denotes the Euclidean distance between  $p_i$  and  $p_j$ .

Subsequently, the neighbors of  $p_i$  are classified concerning their densities. If the density of one  $K$  nearest neighbor of  $p_i$  is greater than the density of  $p_i$ , the neighbor is then defined as the dense neighbor of  $p_i$  ( $DN(p_i)$ ). Otherwise, the neighbor is called the sparse neighbor ( $SN(p_i)$ ).

Moreover, a definition of exemplar  $Q$  is also determined as follows,

$$Q = \{p_j | Den(p_j) = \max\{Den(KNN(p_i))\}\}. \quad (2.40)$$

If the density of  $p_i$  is greater than the density of all  $KNN(p_i)$  or  $RKNN(p_i)$ ,  $p_i$  is then the exemplar of itself, which is also called a quasi-cluster center (QCC). With the completion of searching all quasi-cluster centers inside the point cloud  $p$ , the proposed algorithm starts from an arbitrary center and classifies the center and all its sparse neighbors to the same cluster. Afterward, the algorithm scans each point within the cluster and includes the sparse neighbor of the point in the cluster until all cluster points are visited, *i.e.*, the algorithm builds one cluster by constantly spreading out from the quasi-cluster center. Finally, the process is repeated for entire quasi-cluster centers.

Under this circumstance, the clusters spread from dense areas to sparse areas. Nevertheless, the algorithm could classify one point into various clusters simultaneously. Therefore, the algorithm recommends further a similarity parameter  $Sim(C_i, C_j)$  with

$$Sim(C_i, C_j) = \frac{|C_i \cap C_j|}{K}. \quad (2.41)$$

$C_i$  and  $C_j$  are two clusters, and  $|C_i \cap C_j|$  denotes the number of mutual points. If the similarity between two clusters exceeds a defined threshold  $\alpha$ , the clusters are merged into one cluster. Otherwise, the mutual points are classified into the same cluster as its exemplar.

Finally, the algorithm regards the clusters with a number of points less than  $K$  as outliers, whose points are considered clutter. Hence,  $K$  ought to be chosen more minor than the estimated smallest cluster to avoid missing detections. At last, the algorithm labels the remaining clusters and outputs the results.

### 2.3. Multiple particle tracking and post-processing

Multiple object tracking (MOT) is of crucial importance in various applications, such as target tracking, surveillance technology, particle-tracking velocimetry (PTV), or for the purpose of investigation into the movement characteristics of individual particles. Generally, three main ideas of MOT have been presented over the last several decades: PTV, tracking filters with data association algorithms, and deep learning methods.

Some researchers recommend particle tracking based on the local flow field, for instance, the relaxation algorithm improved by Ohmi and Li [85], the Delaunay tessellation particle tracking algorithm described in [136], and the PTV algorithm based on Voronoi Diagramm [135]. Typically, the particles being tracked are tracer particles suspended in a particular flow field. Because the motion of the fluid is relatively continuous over a relatively short period of time, the particles can be matched across two consecutive frames corresponding to a specific flow pattern. The pattern can be a geometric structure formed by particles of the same frame or a mathematical combination of all possible particle matches [136]. These PTV approaches usually start with the inter-frame pattern matches with respect to the implicit and explicit geometric pattern similarity criteria. The particle matches will then be derived from the pattern matching results. PTV requires a restriction on the inter-frame displacements of the particles and is mainly utilized to track particles in the flow field consequently.

To address the tracking problem of relatively free-moving objects, several authors propose tracking approaches based on tracking filters together with association algorithms. One commonly used tracking filter is the Kalman Filter [45] (Chapter 2.3.1), which estimates the current state for the assignment procedure and then updates the state in accordance with the assigned detection from the data association algorithms. The nearest neighbor approach is a simple and efficient data association algorithm that assigns the local nearest detection to the corresponding track, as used in [73]. Nevertheless, the method fails to provide sufficient accuracy when tracking a large number of objects. Hence, the method is modified to the global nearest neighbor algorithm (Chapter 2.3.2), which associates the detections with tracks by taking the global cost instead of the individual nearest neighbor into account, as illustrated in [50]. The global nearest neighbor approach achieves better tracking performance compared to the nearest neighbor algorithm. However, the method is inadequate for tracking tasks in a cluttered environment since one track might be associated with the nearest noise. To avoid the issue caused by clutter, some researchers advise using probabilistic data association (Chapter 2.3.3), which weights all detections within the gating region instead of assigning simply one detection for data association. The probabilistic data association is firstly introduced in [10] and shows its adequacy and accuracy in following tracks in a cluttered environment. This probabilistic approach has been directly extended to integrated probabilistic data association, which regards the existence of a track as an event with an associated probability [78]. Parallely, Fortmann *et al.* [29] improved the probabilistic data association algorithm by computing the probabilities jointly across the set of tracks and clutter instead of separately for individual tracks (Chapter 2.3.4). Musicki *et al.* [77] introduced a further extension of integrated probabilistic data association named joint integrated data association that is recursive and integrates seamlessly with the integrated data association algorithm. Besides probabilistic data association and its deviation, Reid developed the multiple hypotheses tracking algorithm in 1979 [96]. This algorithm is able to track objects in a cluttered environment by building a tree of potential track hypotheses for each candidate track and, therefore, providing a systematic solution to the data association problem. The algorithm is capable of acquiring tracking results with high accuracy. However, the disadvantage is also apparent. The computational cost increases significantly with an increasing number of objects.

The third tracking idea is based on neural networks and is generally particular for tracking objects with explicit features, such as human beings. The unique features of the objects are extracted and utilized in the association procedure subsequently, as illustrated in [17, 27]. Further, Milan *et al.* [75] introduced an approach to online multi-target tracking based on recurrent neural networks (RNNs) and applies the approach to a benchmark of people tracking, afterward. Later, Ahmad *et al.* [5] investigated the performance of Faster region convolutional neural network (Faster-RCNN) in combination with Generic Object Tracking Using Regression Networks (GOTURN) architecture for human-being tracking. As revealed by the authors, deep learning methods could achieve sufficient accuracy. Nevertheless, the computational cost increases significantly, and in addition, extensive training data and specific features of the tracked objects are necessary, which leads to the inapplicability of the methods in the



present work. The above-mentioned tracking approaches are feasible in both 2D and 3D under the circumstance that the recorded 3D information can deliver satisfactorily steady depth information within a video sequence. In the following, the approaches used in work are outlined.

### 2.3.1. Kalman filter

Kalman filter, which is proposed by R. E. Kalman in 1960 [45], is a state estimator for dynamic systems and has been widely applied in various fields, *e.g.*, target tracking and navigation, due to its efficiency and robustness. The three mainly utilized Kalman filters are Linear Kalman filter, Extended Kalman filter, and Unscented Kalman filter [60]. Linear Kalman filter estimates state for linear systems by presenting them with state equations and measurement equations under the consideration of corresponding noises and updating the states together with the noises in accordance with the measurements afterward. The state equation propagates the state  $\mathbf{x}$  from time  $t - 1$  to time  $t$  as

$$\mathbf{x}_t = F\mathbf{x}_{t-1} + B\mathbf{u}_{t-1} + \mathbf{w}_{t-1}, \quad (2.42)$$

where  $F$  stands for the state transition matrix,  $B$  is the control-input matrix,  $\mathbf{u}$  denotes the control vector at time  $t - 1$ , and  $\mathbf{w}$  represents the process noise vector that is supposed to be zero-mean Gaussian with the covariance  $Q$ .

The measurement equation describing the relationship between the state and the measurement at the current time step  $t$  is defined as

$$\mathbf{z}_t = H\mathbf{x}_t + \mathbf{v}_t. \quad (2.43)$$

Here,  $\mathbf{z}$  is the measurement vector,  $H$  denotes the measurement matrix, and  $\mathbf{v}$  stands for the measurement noise vector that obeys normal distribution with the covariance  $R$ .

Let  $\hat{\mathbf{x}}_t^-$  denotes the prior estimated state at time  $t$  derived from the state at  $t - 1$  and  $\hat{\mathbf{x}}_t$  denotes the posterior state estimation combining the measurements at time  $t$ ,  $\hat{\mathbf{x}}_t^-$  is then defined as

$$\hat{\mathbf{x}}_t^- = F\hat{\mathbf{x}}_{t-1} + B\mathbf{u}_{t-1}, \quad (2.44)$$

The corresponding estimated error covariance is defined as

$$P_t^- = FP_{t-1}^-F^T + Q, \quad (2.45)$$

where  $P_t^-$  and  $P_{t-1}^-$  stand for apriori estimation of error covariance at time  $t$  and aposterior estimation of error covariance at time  $t - 1$ , respectively.

Combined with the measurements, the aprior estimation  $\hat{\mathbf{x}}_t^-$  can be updated as

$$\hat{\mathbf{x}}_t = \hat{\mathbf{x}}_t^- + K_t(\mathbf{z}_t - H\hat{\mathbf{x}}_t^-). \quad (2.46)$$

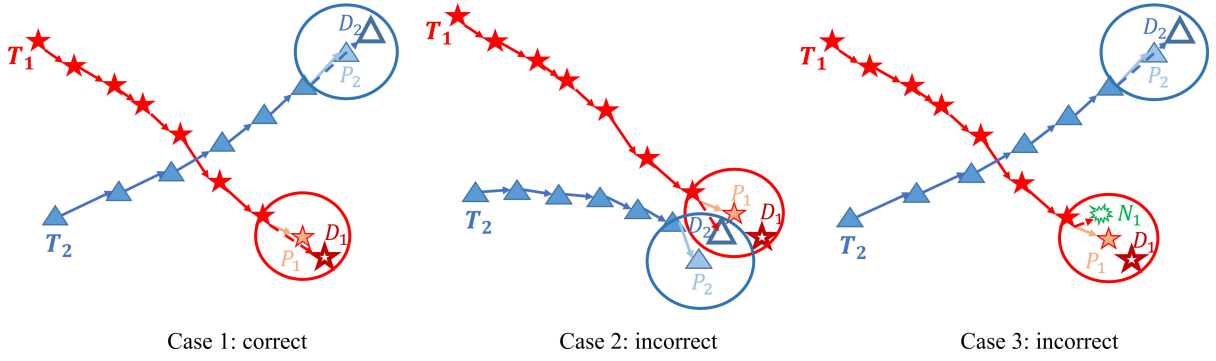
Here  $K_t$  is the Kalman gain, which is defined by

$$K_t = P_t^-H^T(R + HP_t^-H^T)^{-1}. \quad (2.47)$$

The error variance at time  $t$  is then updated as

$$P_t = (I - K_tH)P_t^-. \quad (2.48)$$

As mentioned above, the linear Kalman filter is merely adequate for linear systems. Nevertheless, most of the practical applications are nonlinear, which entails corresponding nonlinear filters for the



**Figure 2.20.:** Overview of the nearest neighbor approach dealing with various situations. Left: correct association of tracks and detections. Middle: Nearest neighbor approach fails to handle assignment problems with high particle density. Right: Track 1 is incorrectly assigned with the nearest noise. Note that only a schematic representation of the assignment process is shown here; subsequent update processes will be based on this assignment result, *i.e.*, the association does not represent the final track. It is also valid for Figure 2.21 and 2.22.

investigation. Thus, the extended Kalman filter (EKF) is developed, transforming the nonlinear system into a linear system equation based on the first-order nonlinear Taylor expansion around the estimated states [60]. Due to the utilization of the linear process Taylor expansion, EKF provides results with sufficient accuracy only if the practical system is approximately linear.

The unscented Kalman filter (UKF) proposed by Julier and Uhlmann [43] is also applied to estimate states for nonlinear systems. Since the state distribution in the EKF is approximated by a Gaussian random distribution and predicted through the first-order linearization of the nonlinear system afterward, the cumulative error in the posterior estimation can be significant. Thus, the UKF uses a deterministic sampling approach instead of random sampling and represents the state distribution by a set of chosen sample points, which facilitates the posterior mean and covariance accuracy. Moreover, the computational efficiency of the UKF is approximately equivalent to the efficiency of the EKF [117].

Since the used camera has a framerate of 330 fps, the motion of a particle between two consecutive frames can be approximated as uniform and modeled as a linear dynamic model consequently. Hence, the linear Kalman filter is selected as the state estimator, and the EKF and UKF are only crudely described.

### 2.3.2. Nearest neighbor and global nearest neighbor approach

One simple solution for data association is the Nearest Neighbor (NN) approach, which assigns the detection closest to the predicted state vector for state update. This approach achieves adequate accuracy and efficiency in sparse scenarios with fewer false detections and duplications. In the presence of many detections and much clutter, NN probably tends to associate the predictions with detections or clutter incorrectly. As depicted in Figure 2.20 middle, when a detection  $D_2$  locates nearer to  $T_1$  than  $T_2$ , it will be assigned to  $T_1$ . Meanwhile,  $D_1$  is discarded as noise, and  $T_2$  fails to be associated with any detections leading to incorrect update outcomes. If noise  $N_1$  coincidentally appears nearest to the predicted position  $P_1$  of track  $T_1$ ,  $N_1$  will be wrongly associated with  $T_1$ , as shown in Figure 2.20 right.

Since NN is not capable of acquiring satisfied results in case of complex association problems, some researchers modified NN by searching the global nearest neighbor instead of local nearest neighbor. The approach Global Nearest Neighbor (GNN) generates at first a cost matrix with  $n$  rows and  $m$  columns,

where  $n$  represents the number of tracks and  $m$  stands for the number of detections, as in the following equation,

$$[C_{ij}] = \begin{bmatrix} c_{11} & c_{12} & c_{13} & \cdots & c_{1m} \\ c_{21} & c_{22} & c_{23} & \cdots & c_{2m} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{n1} & c_{n2} & c_{n3} & \cdots & c_{nm} \end{bmatrix}. \quad (2.49)$$

The elements in the cost matrix are defined as

$$c_{ij} = \begin{cases} C & \text{if measurement } j \text{ is not in the gate of track } i \\ d_{ij}^2 + \ln |S_i| & \text{if measurement } j \text{ is in the gate of track } i \end{cases} \quad (2.50)$$

where  $C$  denotes a predefined cost for non-assignment, which is much larger than the maximum of  $C_{ij}$ ;  $d_{ij}^2$  represents the 2D Mahalanobis distance between track  $i$  and measurement  $j$ ;  $S_i$  is the residual covariance matrix for track  $i$  at the current time dimension.  $\ln |S_i|$  is a penalizing term for tracks with highly uncertain predictions, such as the tracks without assigned measurements for a period of time. For those tracks,  $S_i$  could be very large, which lead to smaller Mahalanobis distance and further to incorrect data association. Under this circumstance, the penalizing term is conducive to reducing the possibility of wrong assignments [50].

The assignment problem is addressed by searching for the sole solution of minimal global cost (the sum of the individual assignment costs) using the Kuhn–Munkres algorithm [51]. Since the detections and tracks are associated under the consideration of the minimal global cost, wrong assignments caused by NN can be modified, as illustrated in Figure 2.21(b). While  $D_2$  remains to be the nearest detection of  $T_1$ ,  $D_2$  is assigned to  $T_2$  for the sake of a minimal global cost. Nevertheless, the problem illustrated in Figure 2.20 caused by clutter can not be solved by GNN, as indicated in Figure 2.21(c).

### 2.3.3. Probabilistic data association

Instead of assigning only one detected measurement to a particular track and discarding the others, Y. Bar-Shalom and E. Tse [10] used probabilistic data association (PDA) that weights all detections within the gating region of the track to follow targets in a cluttered environment. The detections in the gate are weighted by their probabilities calculated on the basis of the Mahalanobis distances to the current predictions. For a track  $T_i$  at time  $t$  with  $m$  detections in the gate, the set of detections can be denoted as

$$Z_i(t) = z_j(t)_{j=1}^{m(t)}, \quad (2.51)$$

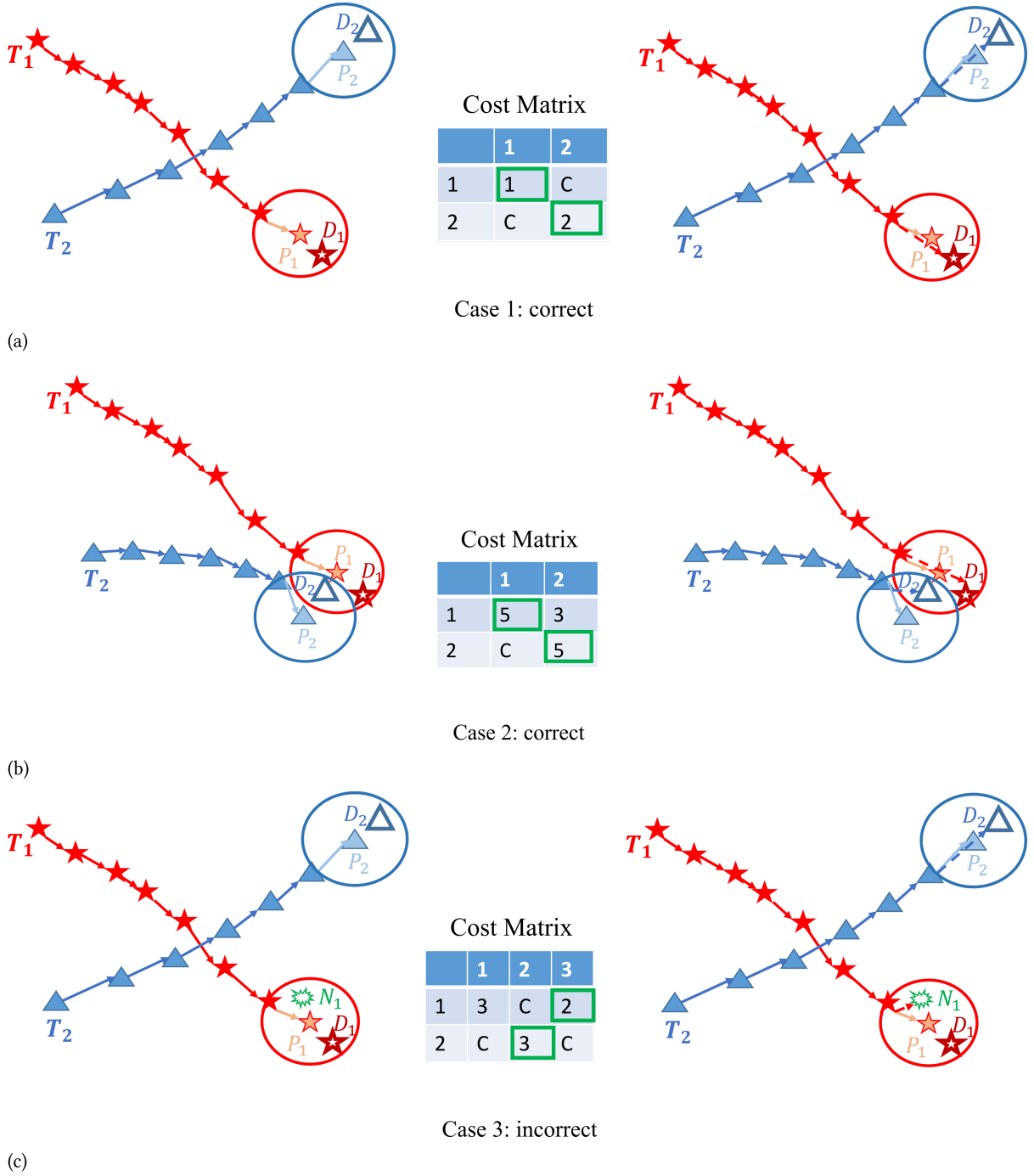
where  $z_j(t)$  is the  $j$ th detection in the gate and  $m(t)$  stands for the total number of detections in the gate at time  $t$ .

Taking the entire detections in the gate into account, the state is updated as

$$\hat{\mathbf{x}}_t = \hat{\mathbf{x}}_t^- + K_t \mu(t), \quad (2.52)$$

where  $\mu(t)$  represents the combination of the detections and is defined as

$$\mu(t) = \sum_{j=1}^{m(t)} \beta_j(t) \mu_j(t). \quad (2.53)$$



**Figure 2.21.:** Overview of the global nearest approach dealing with various situations. (a) correct association of tracks and detections. (b) Following the cost matrix in the middle, GNN is able to correctly associate the detections with the corresponding tracks. (c) Track 1 remains incorrectly assigned with the nearest noise.

Here,  $\beta_j(t)$  is the conditional association probability of the  $j$ th detection, and  $\mu_j(t)$  is the difference between the  $j$ th detection and the estimated state.

The corresponding error covariance is updated as

$$P_t = \beta_0 P_t^- + [1 - \beta_0] P_t^c + \tilde{P}(t) \quad (2.54)$$

with

$$P_t^c = P_t^- - K_t(R + HP_t^-H^T)K_t^T, \quad (2.55)$$

and

$$\tilde{P}(t) = K_t \left[ \sum_{j=1}^{m(t)} \beta_j(t) \mu_j(t) \mu_j(t)^T - \mu(t) \mu(t)^T \right] K_t^T \quad (2.56)$$

is an additional term according to the mixture probability density function [10].

The prediction of the state and detection to time step  $t + 1$  is implemented as in the standard linear kalman filter as mentioned in Subsection 2.2.1. PDA could handle cases with a high density of detections, such as in Figure 2.22 (b),  $T_1$  with  $D_1$  and  $D_2$  in the gate. Although  $D_2$  is closer to  $T_1$  than  $D_1$ ,  $T_1$  is updated in accordance with both  $D_1$  and  $D_2$  ( $W_1$ ). Therefore,  $T_1$  will not be entirely wrong due to a faulty assignment and can be modified to correct trajectory later by accurate detection performance. Notwithstanding, the correctness of the association can only be determined later according to the subsequent trend of the two trajectories. Since PDA considers all possible detections in the gate, the negative influence caused by clutter or high density of detections leading to wrong tracking results can be reduced, as depicted in Figure 2.22(c). While clutter  $N_1$  still locates closest to  $T_1$ ,  $T_1$  is updated by the combination of  $N_1$  and  $D_1$  ( $W_1$ ) and remains not far from  $D_1$ . Because clutters appear temporally irregular compared with detections,  $T_1$  is supposed to be correct after several assignments despite a few clutters.

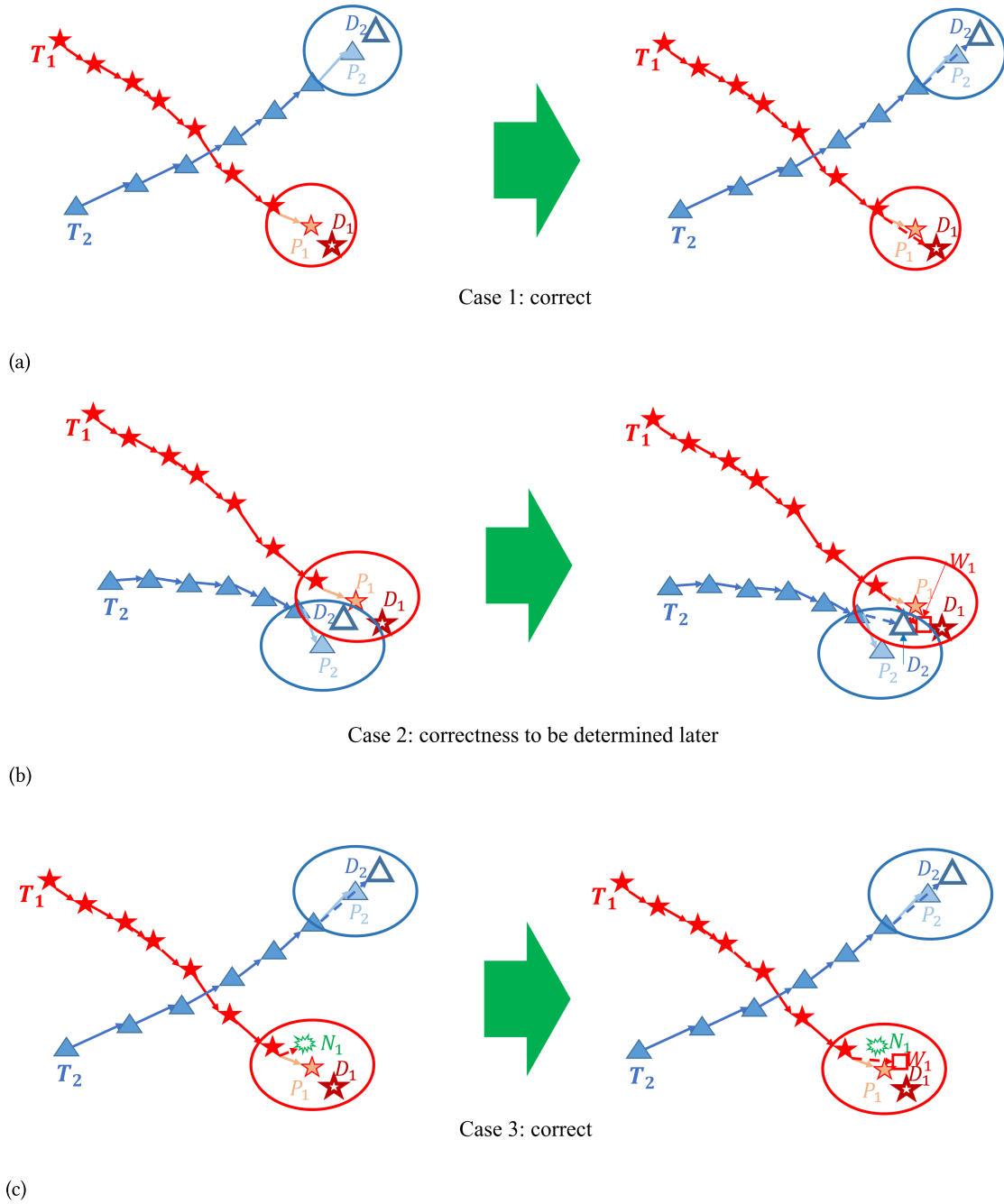
#### 2.3.4. Joint probabilistic data association

Since PDA assumes that tracks are isolated from each other, i.e., the situation that different tracks share the same detections has not been paid much attention, T. E. Fortmann *et al.* [29] proposed the joint probabilistic data association (JPDA) algorithm that computes joint posterior association probabilities for multiple object tracking. Principally, JPDA and PDA utilize identical prediction and update equations. The only difference is the way the association probabilities are computed. Instead of computing the probabilities separately for each track, as in PDA, JPDA computes the probabilities jointly across the set of tracks and clutter.

The JPDA algorithm contains four essential steps:

- Formation of the validation matrix
- Generation of feasible event matrices
- Computing the temporal association probabilities for each track
- Updating the state estimate vector and the corresponding error covariance

Since the last step is the same as PDA in Subsection 2.3.3, it is not detailed here. The other three steps are discussed in the following.



**Figure 2.22.:** Overview of the probabilistic data association dealing with various situations. (a) Correct association of tracks and detections. (b) Track 1 is associated with a weighted position of detection 1 and detection 2. However, the correctness can only be determined later. If the two trajectories are close together subsequently with mostly overlapped gating areas, the two trajectories will probably become increasingly similar later on. Conversely, if the two trajectories tend to be separated, the PDA will most likely arrive at the correct trajectories. (c) The probabilistic data association approach is able to reduce the impact of noise.

### Validation matrix

A validation matrix denoted as  $\Omega$  in the following equation, which consists of binary elements  $\omega_{ji}$ , indicates if the  $j$ th detection lies in the gate of the  $i$ th track, as

$$\Omega = [\omega_{ji}], \quad j = 1, \dots, m, \quad i = 0, 1, \dots, n. \quad (2.57)$$

Here, index  $i = 0$  means non-matching of a detection. Let  $\chi$  denotes a joint event, the validation matrix of event  $\chi$  is then represented as

$$\hat{\Omega}(\chi) = [\hat{\omega}_{ji}(\chi)] \quad (2.58)$$

with

$$\hat{\omega}_{ji}(\chi) = \begin{cases} 1 & \text{if } \chi_{ji} \text{ occurs} \\ 0 & \text{otherwise} \end{cases}. \quad (2.59)$$

Based on the validation matrix of a particular event  $\chi$ , the detection association indicator  $\tau_j(\chi)$  and the target detection indicator  $\delta_i(\chi)$  can be defined as

$$\tau_j(\chi) = \sum_{i=1}^n \hat{\omega}_{ji}(\chi) \quad (2.60)$$

and

$$\delta_i(\chi) = \sum_{j=1}^m \hat{\omega}_{ji}(\chi), \quad (2.61)$$

respectively.

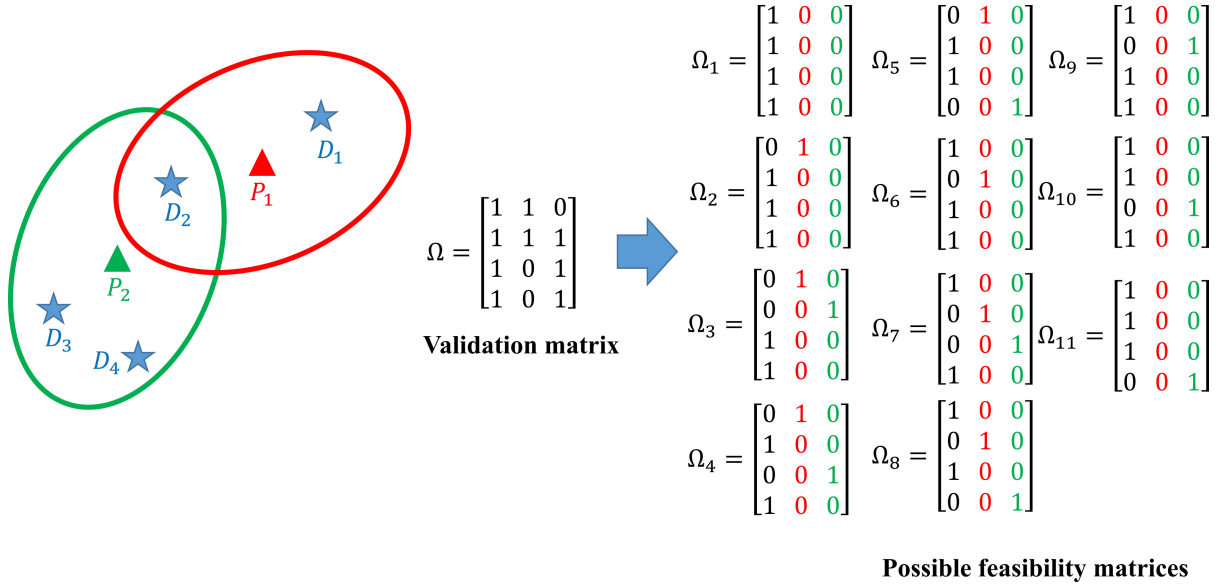
### Feasibility matrices

Feasibility matrices of a validation matrix are acquired by selecting one 1 per row and at most one 1 per column, except for the first column with  $i = 0$ , so that a detection is associated with at most one track and a track is responsible for at most only one detection. Figure 2.23 gives an example of a validation matrix and the corresponding feasibility matrices. Track 1 with current prediction  $P_1$  has two detections ( $D_1, D_2$ ) in the gate, and track 2 has three detections ( $D_2, D_3, D_4$ ) in the gate. For this case, there are totally eleven possible feasibility matrices, as listed in Figure 2.23 right. For instance,  $\Omega_1$  denotes the feasibility event that all detections are non-matching detections, since all 1 appear at  $i = 0$  (first column). Meanwhile,  $\Omega_3$  represents the event that  $D_1$  is assigned to  $P_1$  and  $D_2$  is associated with  $P_2$ , while  $P_3$  and  $P_4$  fail to find corresponding tracks.

The number of feasibility matrices of a unique validation matrix depends on the structure of the validation matrix. For a validation matrix with  $n$  tracks and  $m$  detections, the number of feasibility matrices is not exceeding the following number

$$\text{number of feasibility matrices} \leq 1 + \sum_{i=0}^{n-1} \frac{n!m!}{i!(n-i)!(m-n+i)!}, \quad (2.62)$$

which represents the number of possible feasibility matrices of a validation matrix that is all 1's.



**Figure 2.23.:** Detection validation example with the corresponding validation matrix and all possible feasibility matrices of the validation matrix.

**Association probabilities** Let  $\theta_{ji}(t)$  denotes the particular feasibility event that detection  $j$  from time  $k$  belongs to track  $i$ , the probability of this event conditioned on all detections up to time  $t$  is defined by Bayes' rule as

$$\beta_{ji}(t) = P\{\theta_{ji}(t)|Z_k\}, \quad (2.63)$$

where  $Z_k$  stands for the total amount of detections up to the present time. The probability  $\beta_{ji}(t)$  can be obtained by summing over products of feasibility events  $\chi$  and their corresponding factors  $\hat{\omega}_{ji}(\chi)$  as

$$\beta_{ji}(t) = \sum_{\chi} \hat{\omega}_{ji}(\chi) P\{\chi|Z_k\}. \quad (2.64)$$

With the ability to compute the association probability of the joint events, the state vector can be updated subsequently. As mentioned before, the joint probabilistic data association utilizes the estimation and update equations formatted in equation 2.37-2.42. Regarding the implication  $\beta_{ji}(t)$  is consistent with the previous  $\beta_j(t)$  for a particular track  $T_i$ . In PDA,  $\beta_j(t)$  is not specifically labeled for an individual track because the computing of the probability algorithm is identical for different tracks. In JPDA, however, the probabilities of tracks are specially marked since they are calculated separately according to their respective cases.

As described in the text, the JPDA approach is relatively sophisticated, which entails a high computational load due to the considerable number of matrices to be computed. In order to reduce the computational cost of JPDA and further achieve a wide range of applications, several researchers present fast JPDA algorithms to facilitate computational performance. Zhou *et al.* [137] introduced three fast algorithms to compute the joint probability, including a direct computation approach, an approximate computation algorithm, and a depth-first search method. As proved by the authors, the fast algorithms are able to reduce the computational cost significantly. Whereas the first two methods are generally appropriate for tracking problems with few tracking targets, the computational time of the last approach increases drastically with increasing targets.



### 2.3.5. Post-processing

In order to obtain accurate object trajectories and eliminate unsatisfactory tracking outcomes, post-processing is vital to deal with initial tracking results. In some unsophisticated cases, the post-processing only needs to remove trajectories that are not able to fulfill several pre-defined requirements, such as spatial and temporal length. Whereas for complex tracking problems with frequent appearances of clutter, miss detections, and duplications, the post-processing should realize more possibilities. Since false detections or duplications show a considerably negative impact on the tracking accuracy and remain inevitable due to characteristics of utilized measurement equipment, more precise and robust tracking filters [83, 119, 123] or more sophisticated data association methods [106] could merely achieve limited enhancement of tracking performance. Therefore, the post-processing process ought to implement a thoroughgoing verification and processing of the original tracklets. Thereby, the direct tracking results is preliminarily defined as tracklets regardless of their actual lengths. Several researchers contribute to developing post-processing frameworks to optimize tracking performance. For instance, Bae *et al.* [9] proposed the tracklet confidence using the detectability and continuity of the tracklets and associate the tracklets based on the computed confidence values. Wang *et al.*'s [118] study demonstrates an efficient method to extract discriminative appearance based tracklet affinity models by training convolutional neural networks (CNNs) for tracklet association. Similarly, tracklets reconnection approaches based on the appearance affinity of the representative objects and the motion similarity of the tracklets are introduced in [69, 91]. Some studies regard the tracklet association and reconnection as part of tracklet association based tracking methods. In the present thesis, this process is classified as part of the post-processing procedure since it is based on the already acquired tracklets from a certain tracking algorithm. The majority of tracklet association approaches work with CNNs and depend highly on the representative features of the targets and the similarity between tracklets. Therefore, modified post-processing to handle tracklets of small particles without specific appearance properties is necessary.

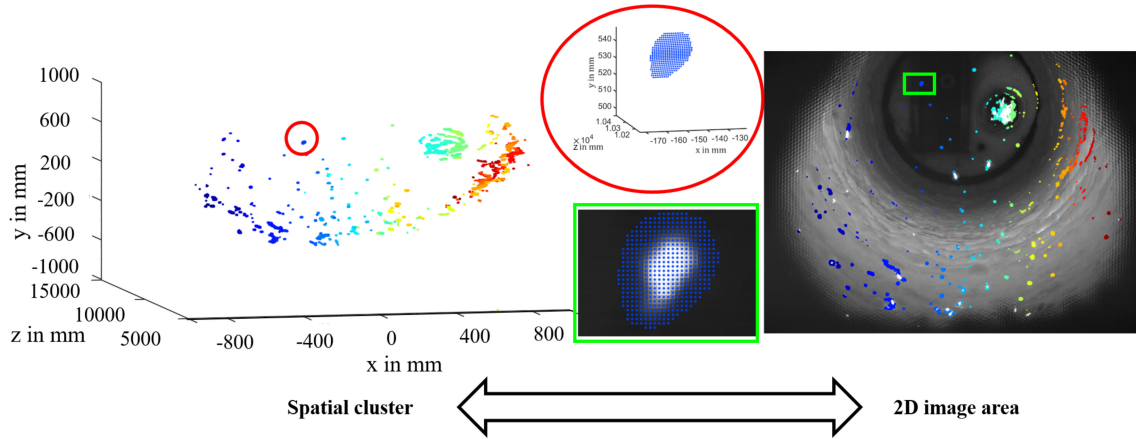
### **3. Novel approaches to particle detection and classification**

In this chapter, a novel multiple particle detection approach is comprehensively detailed. The presented detection approach synthesizes two distinct detection methods and validates their results to tackle the deficiencies of each individual method in the applications. In addition, a 3D particle classification procedure to classify the current states of particles by computing their euclidean distances to a simulated cylindrical rotary kiln model is also illustrated.

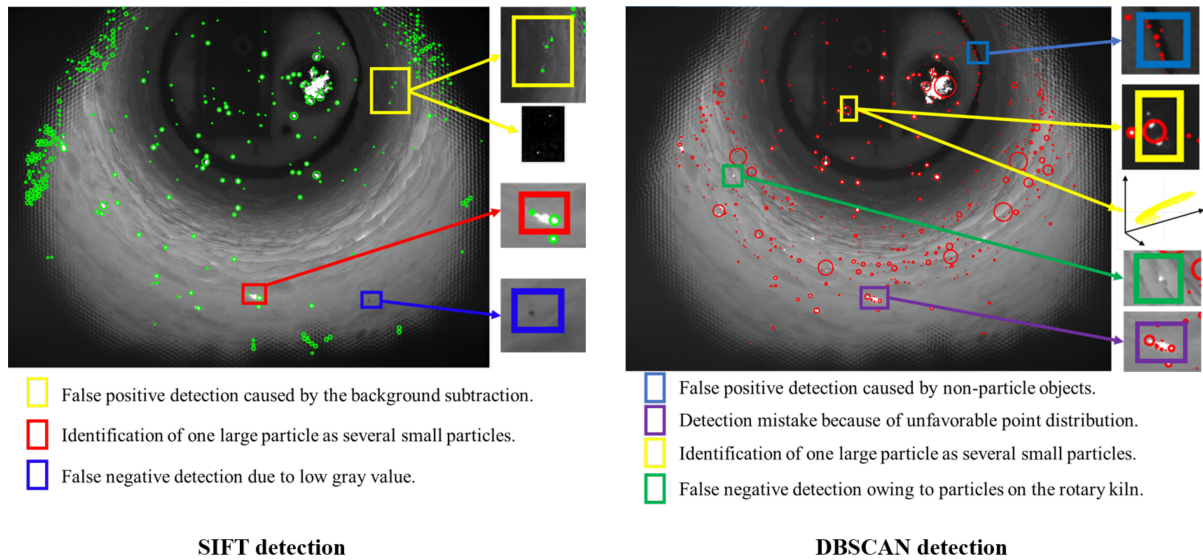
#### **3.1. A novel combined multiple particle detection approach**

As reviewed in the previous state-of-the-art detection approaches, two major object detection concepts are existing: object detection based on 2D gray value images and object detection via 3D clustering. Notwithstanding, these two concepts have specifically inherent merits and deficiencies. Their accuracies suffer from dependency on the characteristics of the images. One common error source for object detection utilizing 2D gray values is the local gray value difference caused by background subtraction. Since the computed background model could vary slightly from the actual background of each image, new gray value gradients could arise after background subtraction and subsequently impact the object detection. Moreover, a large object might be identified as several small separate objects owing to the unfavorable gray value distribution of the object. This could lead to wrong object localizations and adversely affect the subsequent object tracking. In addition, the gray value based methods fail to detect objects that differ minorly from the background. On the contrary, if objects are detected by forming clusters in 3D point clouds, the above-mentioned problems caused by gray values are avoided. In comparison, the performances of the clustering methods are conditional on the characteristics of point clouds. Thus, the clustering approaches are only capable of clustering several particular types of data robustly. Since the camera also captures several non-particle objects, *e.g.*, edges, lance, and the wall of the rotary kiln, the acquired point clouds contain points from these objects, which could incorporate into existing clusters or even form new clusters. These false clusters constitute the first error source of the clustering methods. In addition, several clusters might be formed inside one big particle due to the unfavorable point distribution. Furthermore, the clustering method is not able to divide the points from neighboring objects correctly if the objects are too close to each other. Analogously, particles lying on the inner wall of the rotary kiln can be incorporated into the cluster of the rotary kiln leading to missing detections. In order to detail the mentioned issues, the SIFT algorithm and DBSCAN approach are implemented separately. For a visualized comparison of the detection performances, the 3D detection outcome of 3D-DBSCAN is converted into the 2D image coordinate, where a 2D image area represents each spatial cluster, as depicted in Figure 3.1. Figure 3.2 schematically illustrates the detection results of SIFT and DBSCAN with examples of respective detection inaccuracy.

To rectify the problems of the individual concepts, a combined detection approach is presented, which applies not merely the 2D gray value images but also the 3D point clouds to extract objects accurately. This approach was firstly introduced in [131] and subsequently extended in [129]. The schematic

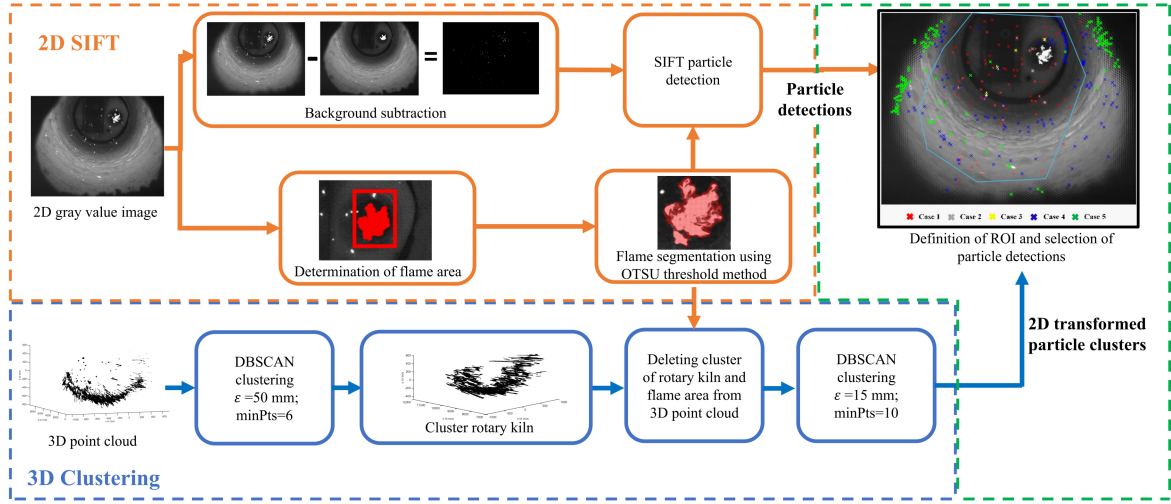


**Figure 3.1.:** Spatially formed clusters and their corresponding 2D image areas. Each color stands for an individual cluster. Since each captured pixel is assigned with an image coordinate, a spatial coordinate, and a gray value, the 3D clusters in the cartesian coordinate and the 2D clusters in the image coordinate can be transformed into each other. The red ellipse and the green rectangular enlarge a 3D cluster and a 2D transformed image area, respectively.



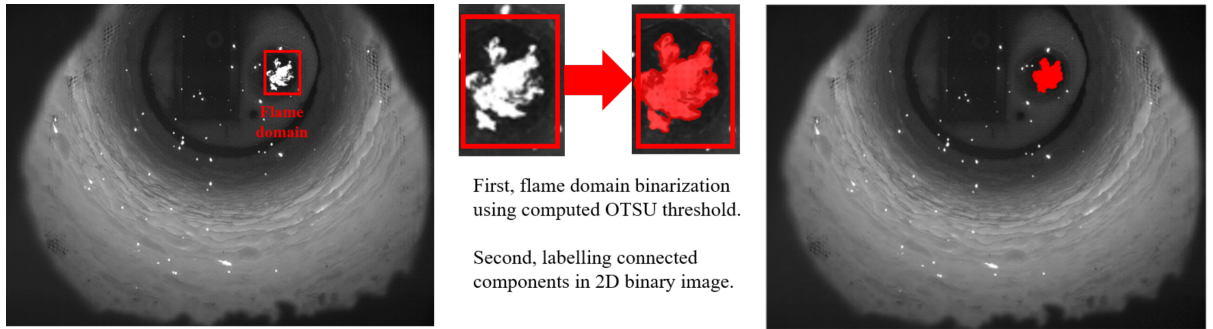
**Figure 3.2.:** Detection results of 2D-SIFT and 3D-DBSCAN. Left: Detection result of 2D-SIFT. The center of each green circle corresponds to the position of the detected particle, whose sizes are pointed by the circles' radius. The yellow box is zoomed in to display the detection on the initial frame and the frame after background subtraction. Right: Detection result of 3D-DBSCAN. The center of each red circle corresponds to the center of the cluster. Radius of the circle indicates the cluster size. The yellow box shows the particle detection both on the 2D frame and in the 3D point cloud. [133]

presented in Figure 3.3 illustrates the process of the novel detection approach. In the work, SIFT algorithm is used as an instance of the gray value based object detection method. In principle, the method can be replaced by other detection approaches with respect to the applications, such as OTSU-threshold or artificial neural networks. Because the objects detected in the work are RDF particles, the term particle detection in Figure 3.3 is consistent with object detection.



**Figure 3.3.:** Schematic of the novel combined detection approach.

As mentioned in the chapter on the experimental setup, an oil burner that is partly inside the rotary kiln heats the test facility and thus, brings flame into the kiln. The flame is exceptionally bright and can be identified by the gray value based detection approach resulting in false positive detection. Hence, a flame domain is defined, and the flame within the domain is segmented utilizing the OTSU-threshold method before the detection, as schematically depicted in Figure 3.4. In addition, the two-dimensionally segmented flame of a certain frame is converted into 3D points pixel-by-pixel, which ought to be deleted in the 3D clustering input point cloud afterward. With the completion of background subtraction and flame segmentation, the images are processed by the SIFT approach.



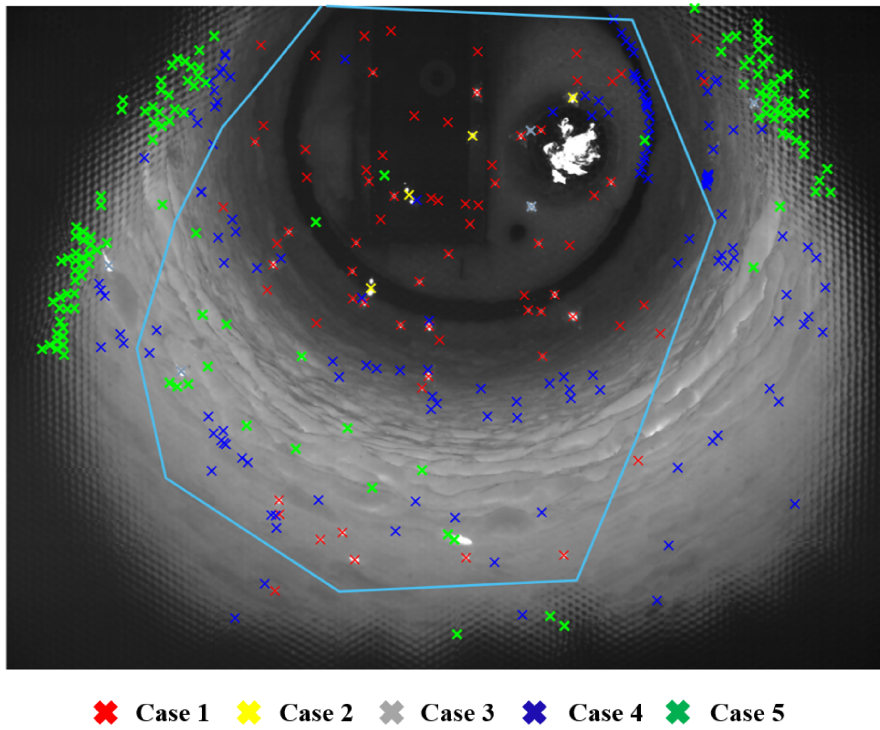
**Figure 3.4.:** Flame segmentation within a defined rectangular ROI around the burner, as marked in red rectangular. The red-marked flame on the right is the segmented flame.

In parallel, particles are identified by forming clusters in the input point clouds utilizing the DBSCAN algorithm. At first, a relatively big radius  $\epsilon$  and a small  $minPts$  are selected, which allows us to gather more points within one cluster, aiming at discovering the biggest cluster: the inner wall of the rotary kiln. Subsequently, the points belonging to the flame and the clustered rotary kiln are eliminated from the input point cloud to reduce false positive detections. Finally, clusters in the rest of the point cloud are formed with a small radius and a big  $minPts$  in order to separate the particles adequately.

After the independent implementation of the two object detection approaches, their detection results can be compared and combined to tackle the deficiencies of the individual methods. To reduce the

issues caused by artifacts around the images, a region of interest is defined to exclude the artifacts and concurrently contain as many particles as possible. One ROI example is depicted in Figure 3.5, where the ROI is marked by a light blue polygon. Apparently, a few particles at the side are also excluded from the scope leading to false negative detections. Notwithstanding, the majority of these particles are on the kiln's wall and are consequently not of research interest for further analysis of particle flight properties. Therefore, the benefits of defining the ROI outweigh the disadvantages.

After introducing the ROI and excluding the particles outside the polygon, the two acquired detection results are validated with each other, which results in five specific cases, as highlighted in Figure 3.5. The SIFT algorithm ascertains not only the pixel-exact localization of a particular particle (keypoint) but also its corresponding size (radius), viz., a keypoint bounds to a corresponding radius. A keypoint together with the radius is considered as a detected particle. Therefore, the 5th case, that a potential particle corresponds to precisely one detected keypoint and more than one discovered cluster is rare but possible. In the following, solutions to deal with the specific cases are elaborated separately.



**Figure 3.5.:** Five cases of the comparison of particle detections. The blue polygon is an instance of the defined ROI. [133]

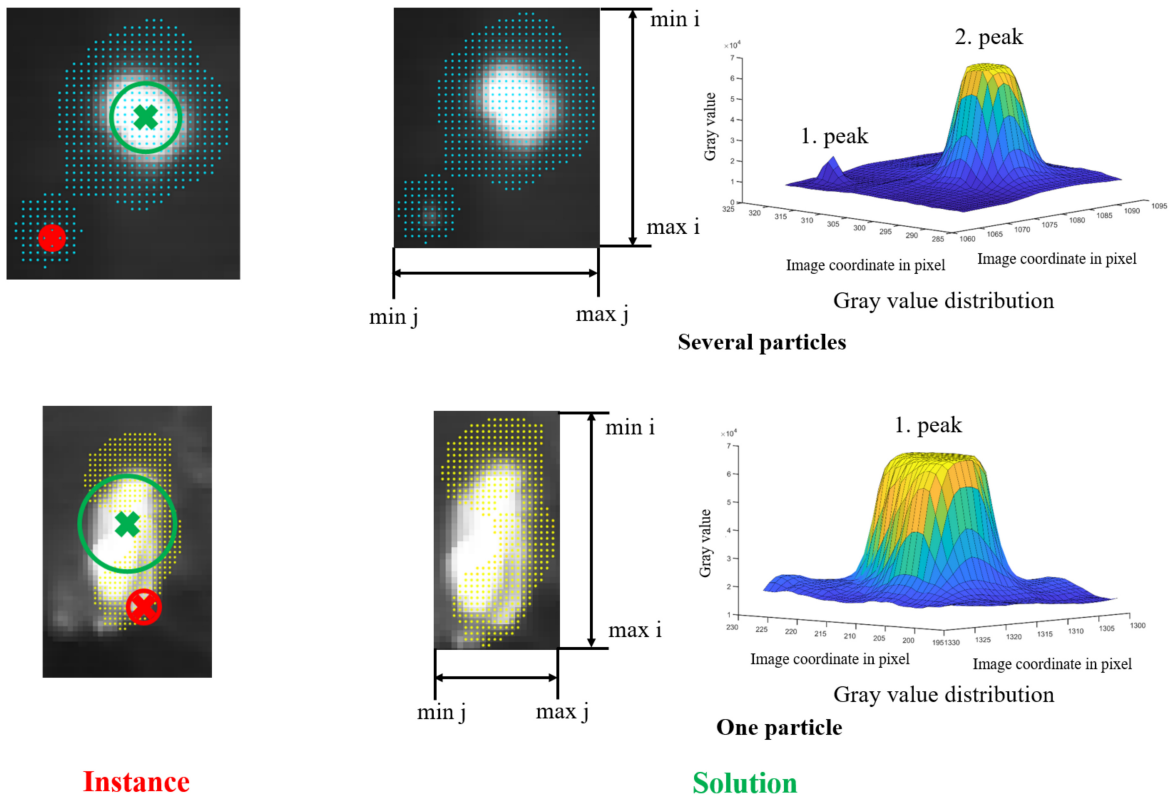
**Case 1: A potential particle corresponds to exactly one detected keypoint and one discovered cluster.**

Case 1 is relatively common and indicates identical detection results of the utilized approaches. Under this circumstance, the potential particle is identified by both methods, which confirms the correctness of the outcomes. Therefore, the detection is straightforwardly regarded as a particle without any further analysis.



**Case 2: A potential particle corresponds to exactly one discovered cluster but more than one detected keypoint.**

Case 2 frequently occurs under the circumstance of a large particle or several contiguous particles. For the former, the large particle might contain complicated gray value distribution after background subtraction that could lead to a SIFT detection result with a few keypoints. On the condition that several particles are too close to each other, the clustering algorithm might fail to divide the points appropriately. To deal with this case, the cluster region is determined, and the gray value distribution within the area is investigated, as depicted in Figure 3.6. The cluster area is a rectangle enclosed by the cluster's maximum and minimum line ( $[\min i, \max i]$ ) and column ( $[\min j, \max j]$ ) coordinates. By counting the amount of the local gray value peaks within the cluster area, the precise number of particles is able to be determined.



**Figure 3.6.:** Instance and solution for case 2. Transformed 2D clusters are marked by small dots, where each dot corresponds to a pixel. Circles with cross centers highlight the detections of the SIFT algorithm. The cross center locates at the detected keypoints, and the circle's radius indicates the particle's size. The identical notation is also used in the following three figures.

**Case 3: A potential particle corresponds to exactly one detected keypoint and more than one discovered cluster.**

This case occurs only rarely and only if one massive burning particle was not captured entirely by the plenoptic camera, as shown in Figure 3.7. As a consequence of these particles' overexposure, the plenoptic camera fails to provide complete 3D point cloud information for such particles. The solution for this special case is similar to the solution for case 2. The gray value within the particle area ought to be investigated to decide the accurate number and location of particles. The particle area is determined

by overlapping the SIFT detection and the discovered clusters. The area remains a rectangle, which is enclosed by the maximum and minimum line and column coordinates of the overlapping. As depicted in Figure 3.7, both burning particles are considered as multiple particles by the clustering method because of the provided dispersed three-dimensional information. The final determination of the particle number can be accomplished by analyzing the distribution of gray values in the defined area and determining the number of peaks within the region.

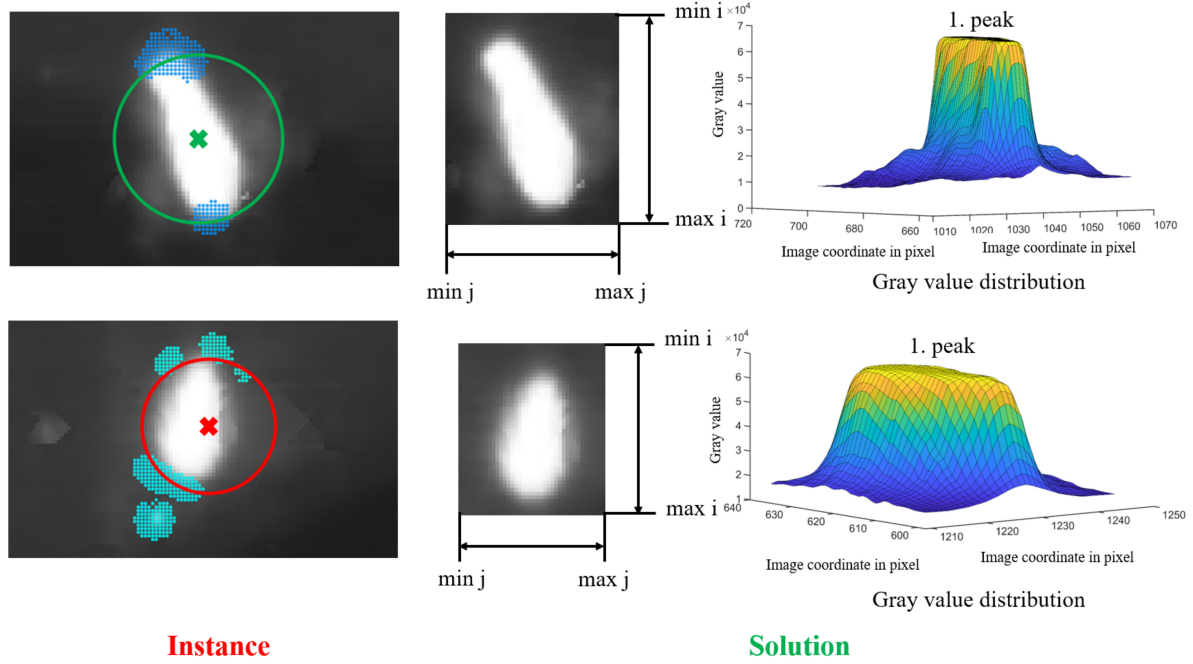


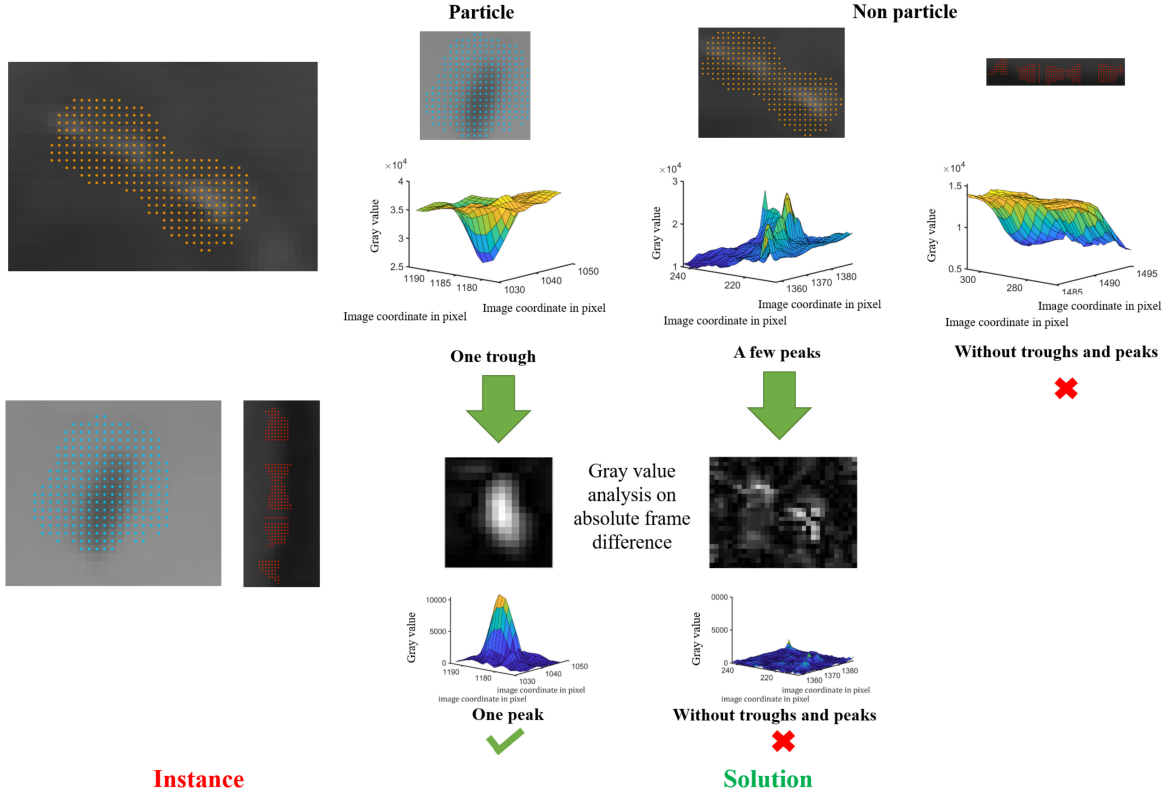
Figure 3.7.: Instance and solution for case 3.

#### Case 4: A potential particle is only discovered by the clustering approach.

This case can result from dark particles and non-particle objects inside the rotary kiln, for instance, edges. The dark particles might be deleted automatically after the background subtraction, depending on the applied background model. Therefore, the SIFT algorithm fails to detect such particles. In comparison, the clustering method is able to discover clusters of both particle and non-particle objects, which could also give rise to case 4. Figure 3.8 illustrates the instances and the solution for the case schematically. In order to ascertain whether a discovered cluster corresponds to a real particle, the gray value distribution within the 2D cluster area on the current frame is analyzed at first. On the condition that troughs or peaks can be detected in the area, the analysis of the gray value distribution proceeds on the absolute frame difference, which takes the absolute value of the difference between the current frame and the adjacent previous frame. Therefore, non-particle detections can be removed since they hardly move between two consecutive frames.

#### Case 5: A potential particle is only detected by the SIFT algorithm.

If a potential particle can only be detected by the SIFT algorithm, the reason could be relatively diversified, as exemplified in Figure 3.9. For instance, if one particle is on the rotary kiln, its corresponding point cloud is assimilated into the kiln's cluster resulting in a missing cluster. Additionally, one tiny



**Figure 3.8.:** Instance and solution for case 4. Three instances containing one particle and two non-particles and their corresponding solutions are presented in the figure. Each column of the solution illustrates an example.

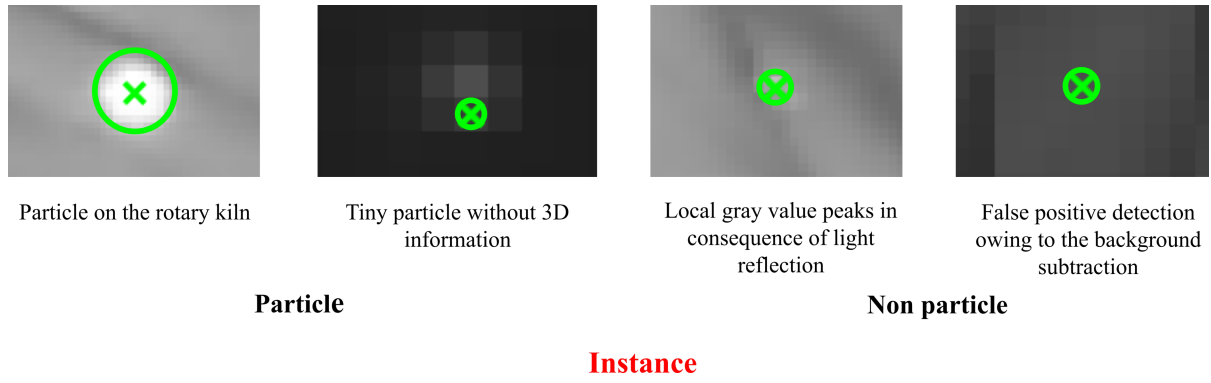
particle with several pixels in size could also lead to a false negative cluster since the plenoptic camera might not successfully capture the 3D information of these tiny particles. Besides particles, non-particle objects could also cause case 5, such as local gray value peaks in consequence of light reflection and gray value peaks owing to the background subtraction. The approach to dealing with this case is the same as in case 4. The false positive detections caused by the background subtraction can be discovered by analyzing the gray value distribution on the current frame, whereas the local gray value peaks due to the light reflection disappear on the absolute frame difference.

The proposed method is able to address the mentioned problems of the detection approaches and, thus, reduce false detections and correct inaccurate detection results. In Chapter 5, the results of different detection methods are compared that can quantitatively confirm the accuracy and adequacy of this novel combined detection method. The detection results of this method provide a reliable basis for the subsequent particle tracking.

### 3.2. 3D Particle classification

Since only the particles in the air are of research interest for further analysis of particles' flight and combustion behaviors, particles are classified regarding their spatial positions relative to the rotary kiln. The particles near the rotary kiln are considered to be landed particles, while those not yet close





**Figure 3.9.:** Instance for case 5.

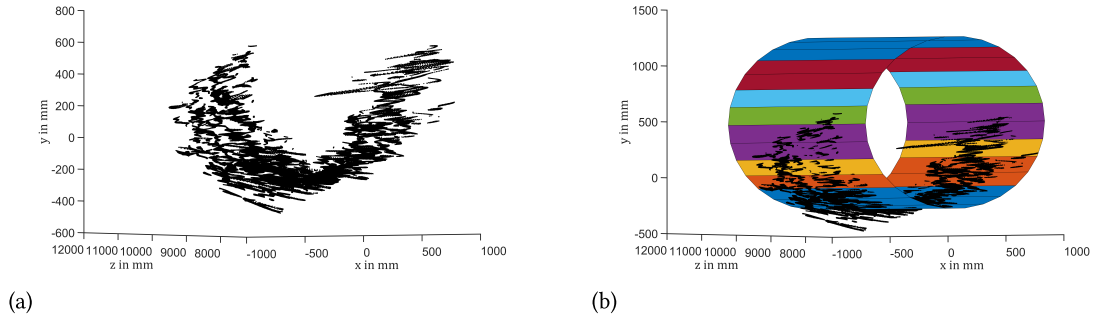
to the rotary kiln are regarded as in-air particles. In the following, the simulation of the rotary kiln and the classification of the particles are demonstrated.

### Estimation of the rotary kiln geometry from the point cloud

To achieve the classification, the 3D position and geometry of the cylindrical rotary kiln are estimated by employing the estimator maximum likelihood estimation sample consensus (MLESA) [113] at first. The estimator MLESAC is a robust method to estimate image geometry. It utilizes a particular sampling strategy to generate putative solutions and determines the sole solution by maximizing a derived likelihood. By limiting a maximum point-to-cylinder distance and an orientation cylinder axis, the rotary kiln model can be estimated by the MLESAC estimator based on the temporally obtained clusters of the rotary kiln, as mentioned in Figure 3.3. In the present work, the input is point clouds representing the inner wall of the rotary kiln. Additionally, a maximum allowable distance from an inlier point to the cylinder model ought to be defined. Moreover, the approximated axis of the cylinder (the depth direction, *i.e.*, the *z*-axis) could also be given for the sake of accurate estimation. Consequently, the MLESAC algorithm is able to estimate the geometry of the rotary kiln as a specific cylinder with certain information, such as position, center, radius, and length. Alternatively, the prior knowledge concerning the size and the position of the rotary kiln can also be applied to estimate the kiln's model. Nevertheless, the exact position of the kiln relative to the camera is hardly measurable. Withal, the calibration might also result in deviation of the point cloud from the actual size and position. Therefore, the rotary kiln geometry is simulated by the MLESAC based on the captured point cloud that is precise and universally applicable. Figure 3.10 gives an example of a rotary kiln cluster and an estimated cylindrical kiln model.

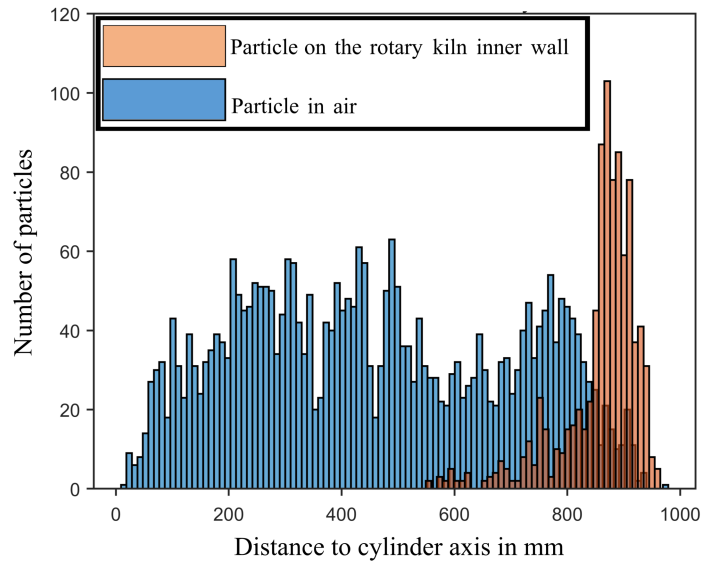
### Particle classification

Based on the estimated cylinder model, the particles are able to be classified according to their Euclidean distance to the cylinder axis. In order to determine an appropriate criterion for particle classification, the particles of the detection ground truth were manually followed to ascertain their positions relative to the rotary kiln. Figure 3.11 depicts the distances between particles and the cylinder axis. As shown in the figure, the majority of the in-air particles have a distance to the axis shorter than the radius of the rotary kiln (700 mm). Notwithstanding, owing to the measurement uncertainty of the plenoptic camera, several in-air particles are beyond (800 mm) away from the cylinder axis. Approximately, all



**Figure 3.10.:** Cluster of the rotary kiln and a corresponding estimation of the kiln's geometry. (a) A cluster of the rotary kiln. (b) Estimated cylindrical kiln model.

the particles on the inner wall of the kiln have a distance to the axis over the kiln's radius. In the light of the distance distribution illustrated in Figure 3.11, a criterion to classify the particles can be selected preliminarily. In the present work, the intersection point of the two Gaussian distributions is chosen as the criterion to determine the relative positions of the particles. These two Gaussian distribution curves are obtained by computing the expectation and standard deviation of the respective distance to the cylinder axis, *i.e.*, particle in air and particle on the rotary kiln.



**Figure 3.11.:** Distribution of the distances between particles and the rotary kiln axis. The horizontal coordinate represents the distance to the axis, and the vertical coordinate illustrates the number of particles from the individual distance interval.

### 3.3. Detection performance evaluation method

In order to investigate the relevance between the detections and the manually labeled ground truth (Chapter 5.1.1), a confusion matrix is constructed and the precision, recall, and F1-score are computed accordingly [94]. The confusion matrix consists of four elements: true positive (TP), false positive (FP), false negative (FN), and true negative (TN). TP stands for correct detection, while FP indicates false

detection and FN denotes the missing-detected particle in the ground truth. TN refers to the missing detection of a non-particle object, which is not available in our detection work. Based on the three obtainable elements of the confusion matrix, the measures to assess the detection performance can be computed by the following equations

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (3.1)$$

$$\text{Recall} = \frac{TP}{TP + FN}, \quad (3.2)$$

$$F_1 \text{ score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (3.3)$$

The precision, also called the positive predictive value (PPV), is the proportion of positive detections that are true positive. A high precision value can be interpreted as a precise detection performance. The recall, which is also defined as the sensitivity or the true positive rate (TPR), refers to the probability of positive detections conditioned on truly being positive. A high recall value indicates the ability to identify true positive detections. The  $F_1$  score is a combination of precision and recall that synthesizes the individually provided information and estimates the detection performance more comprehensively. The further evaluation of the detection performance is in accordance with these three measures.

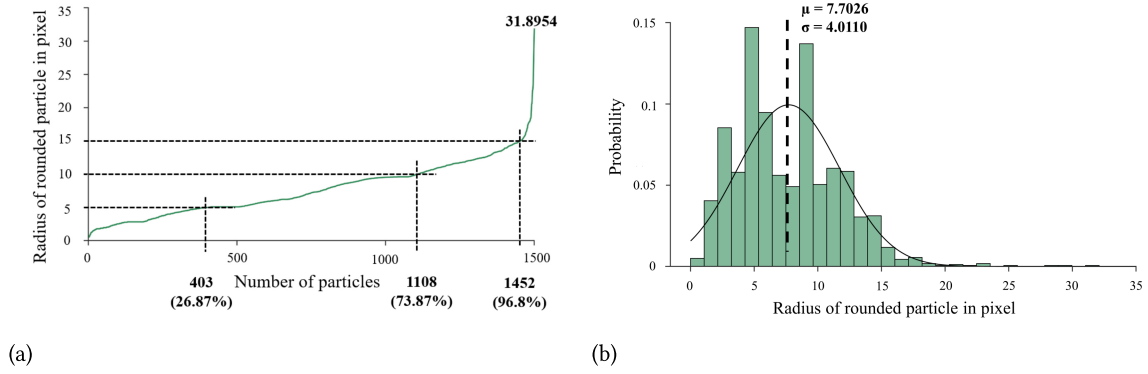
Apparently, the completion of the confusion matrix entails a correct assignment between the detections and the particles from the ground truth. Therefore, a tool is developed that assigns detections to particles based on the Hungarian method [51]. One significant benefit of the developed tool is to get rid of the dependence on the detection scan order that affects the matching performance noticeably. Moreover, by computing the costs of possible assignments, various factors can be taken into consideration, such as the Euclidean distance and the gray value.

In our research, the cost of each assignment is defined as

$$\text{Cost} = \begin{cases} d(d, p) & \text{if } d(d, p) \leq d_\epsilon \\ C_{\max} & \text{if } d(d, p) > d_\epsilon \end{cases}, \quad (3.4)$$

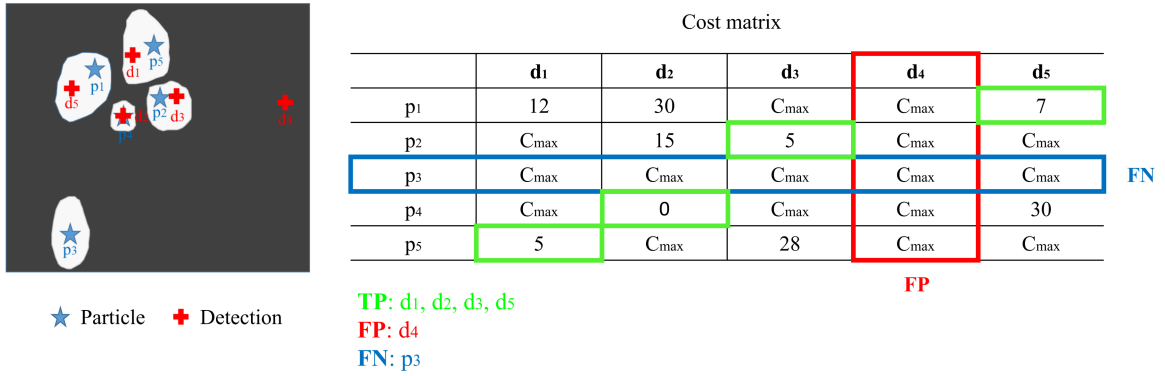
where  $d(d, p)$  denotes the Euclidean distance between one detection and one particle in image coordinate,  $d_\epsilon$  stands for a gating threshold for the distance, and  $C_{\max}$  is a preset maximum cost that is significantly greater than the Euclidean distances. To determine the appropriate gating threshold  $d_\epsilon$ , the sizes of 1500 ground truth particles have been investigated. The particles were assumed as circles, whose radii were acquired by regarding the clusters' sizes as the circular areas since each cluster point is exactly 1 pixel  $\times$  1 pixel in size. Figure 3.12 illustrates the rounded particle radius schematically. Approximately all particles are with radii smaller than 15 pixel, and the mathematic expectation of the radii is around 7.7 pixel. Considering the shape of the actual particles, the diameter of the particles is enlarged by a factor of 1.5. Therefore, the gating threshold (diameter) is selected as 45 pixel.

Figure 3.13 details the assignment process using a matching instance with corresponding cost matrix. Detection  $d_4$  can not be assigned to any ground truth particles under the condition of the gating threshold and is therefore regarded as FP. Analogously, the ground truth particle  $p_3$  is identified as FN in consequence of non-assignment. In addition to computing merely the Euclidean distance as the assignment cost, experiments using the distance together with the gray value changes between particles and detections as cost were also performed. For instance, the standard deviation of the gray values at the detection position, the ground truth particle position, and the middle of these two positions was computed as an additional term of the assignment cost. Because the gray value between unmatched particles and detections varies considerably, more accurate assignment outcomes is expected using this



**Figure 3.12.:** Rounded particle radii of 1500 particles. (a) The radii are sorted ascendingly. The horizontal coordinate indicates the index of a particle, and the vertical coordinate reports the rounded radius of the particle. (b) Distribution of the particles' radii and the corresponding normal distribution. The horizontal axis shows the radii, and the vertical axis stands for the probability of each radius interval.

complicated cost function. Nevertheless, the thereby obtained results were similar to those obtained utilizing only the distance since the matched detection and particle with the closest Euclidean distance is usually the matching pair with the slightest gray value change. Thus, the most straightforward way is chosen to define the cost function.



**Figure 3.13.:** Assignment instance with five detections and five particles. Blue stars represent particles, and red crosses stand for detections. The corresponding cost matrix is illustrated on the left. The red color marks the FP, while the blue color marks the FN. The green color is the global lowest cost, *i.e.*, the final assignment outcome.



## 4. Extended multiple particle tracking and post-processing

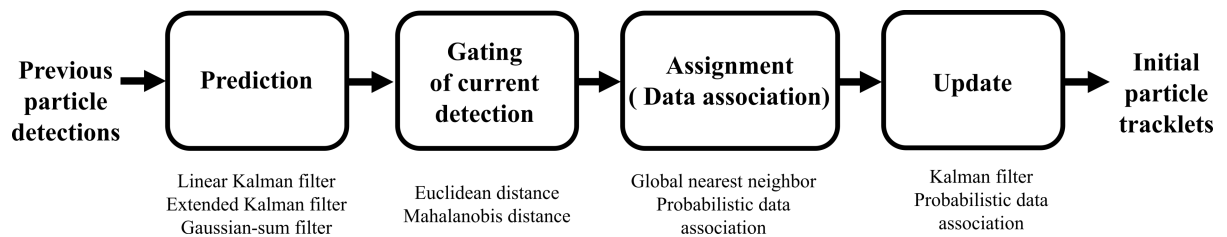
As mentioned in the first chapter, the concept of tracking-by-detection consists of object detection and object tracking. With the completion of detecting particles, the detections ought to be associated subsequently to build particle trajectories. Chapter 4 proposes the applied tracking approaches, including an extended 2.5D tracking method aiming at yielding initial particle tracklets and a novel post-processing framework to alleviate issues in the initial tracking results for the purpose of obtaining final particle trajectories. Withal, a polynomial regression of the straightforwardly extracted spatial trajectories to compensate for the fluctuations of the trajectories is also presented. The exact values of the parameters in this chapter are discussed detailedly in the next chapter.

### 4.1. 2.5D extended tracking algorithm

Principally, classic multiple particle tracking comprises four primary steps, viz., prediction, gating, assignment, and update, as schematically illustrated in Figure 4.1. The tracking process follows the presented four steps and adapts the selection of the approaches in accordance with the usage.

#### 4.1.1. 2D linear Kalman filter estimator

Since the plenoptic camera captured the image with a framerate of 330 fps corresponding to around one image every 3 ms, the motion of one particular particle within two adjacent images can be considered simply uniform, whose state vector contains only position and velocity when employing the efficient linear Kalman filter to estimate the particle position and the corresponding covariance matrix for the current frame. The tracking process can proceed in either 2D or 3D. Nevertheless, the camera fails to provide 3D information with sufficient stability, as revealed in the first chapter. Because tracking filters are generally based on regular motions (linear or nonlinear), the noticeable fluctuations of the spatial information, especially in the depth direction, significantly limit the performance of direct 3D tracking. Hence, the particles are tracked two-dimensionally on the image coordinate. Each particle detection of



**Figure 4.1.:** Schematic of the multiple particle tracking procedure. Instances of available approaches for individual steps are listed below.

the first frame is considered an initial track, with which a linear Kalman filter begins to predict the particle's position. Let  $\vec{x}_{i,t}$  denotes the state vector of a special track  $i$  at time  $t$ , then  $\vec{x}_{i,t}$  equals

$$\vec{x}_{i,t} = \begin{bmatrix} \vec{p}_{i,t} \\ \vec{v}_{i,t} \end{bmatrix}. \quad (4.1)$$

Both  $\vec{p}_{i,t}$  and  $\vec{v}_{i,t}$  are column vectors. For a uniform motion,  $\vec{x}_{i,t}$  can be propagated as

$$\vec{p}_{i,t} = \vec{p}_{i,t-1} + \Delta t \vec{v}_{i,t-1} \quad (4.2)$$

$$\vec{v}_{i,t} = \vec{v}_{i,t-1}, \quad (4.3)$$

where  $\vec{p}_{i,t}$  stands for the position of track  $i$  at time  $t$  with two elements corresponding to two image coordinates, and  $\vec{v}_{i,t}$  is the relevant velocity vector. Thus, the state transition matrix  $F$  can be determined as

$$F = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.4)$$

In order to describe the temporally vague deviation of the state vector  $\vec{x}_{i,t}$ , an additive process noise vector  $\vec{w}_{i,t}$  that is white and Gaussian with zero mean is defined [19]. The covariance of  $\vec{w}_{i,t}$  is denoted as  $Q_{i,t}$ . As the consequence of the additional process noise vector, the state vector is modified as

$$\vec{x}_{i,t} = F\vec{x}_{i,t-1} + \vec{w}_{i,t-1}. \quad (4.5)$$

The corresponding error covariance matrix is predicted as

$$P_{i,t} = FP_{i,t-1}F^T + Q. \quad (4.6)$$

#### 4.1.2. 2.5D Mahalanobis distance gating

In the gating process, a specific region around one prediction is determined to restrict the range of candidate detections for subsequent particle association. Generally, the region is defined by a certain distance to the prediction, such as the Euclidean distance or the Mahalanobis distance. Unlike the Euclidean distance, which considers only pure distance, the Mahalanobis distance takes the Euclidean norm and a probability distribution into account. Thus, the Mahalanobis distance corresponds to the standard Euclidean distance after a whitening transformation. For a given probability distribution  $A$  and a covariance matrix  $P$ , which reflects different measurement accuracies in row and column directions in the image (different measurement accuracies in lateral and transverse directions of the particle), the Mahalanobis distance between a temporal prediction  $\vec{p}_{i,t}$  and an arbitrary detection of the same frame  $\vec{m}$  with respect to  $A$  equals [70]

$$d_M(\vec{x}_{i,t}, \vec{m}) = \sqrt{(\vec{x}_{i,t} - \vec{m})^T P^{-1} (\vec{x}_{i,t} - \vec{m})}. \quad (4.7)$$

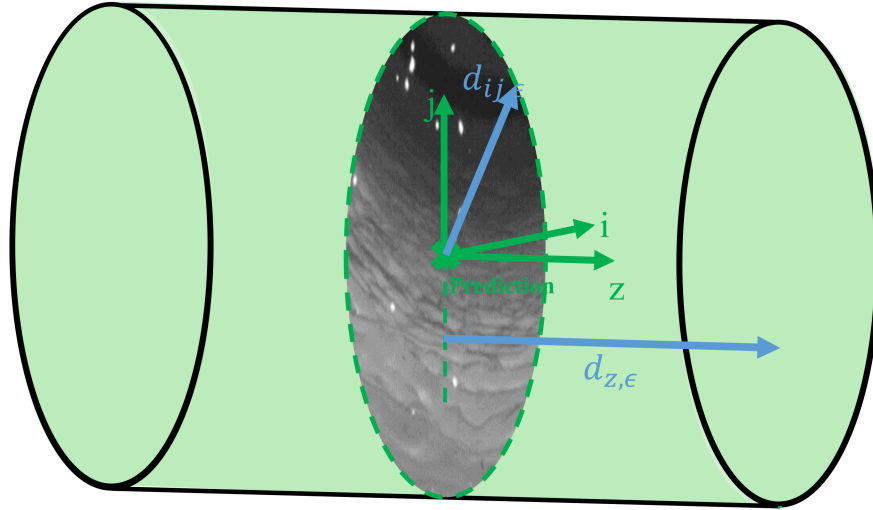
The gating area excludes the detections with a Mahalanobis distance to the specific prediction exceeding a predefined threshold  $d_e$ . As introduced in the second chapter, the plenoptic camera is able to provide spatial information on the detections. Notwithstanding, the acquired 3D knowledge lacks sufficient stability and precision in the depth direction. This could lead to the consequence that the variance in the depth direction is predominant and significantly outweighs the other two directions. Suppose a

detection is very close to a prediction in the x-y-plane and extremely far away in the depth direction. In that case, it might also be incorrectly included in the gating area even with the Mahalanobis distance gating, causing problems for the subsequent process. Additionally, the fluctuated depth information challenges the selection of  $d_\epsilon$  considerably. A large  $d_\epsilon$  will lead to a gating result with many candidate detections that locate far away from the prediction in the xy-plane, whereas a small  $d_\epsilon$  might ignore the correct detection with a long depth displacement.

On the other hand, a complete abandonment of applying the depth information in gating could also give rise to incorrect results due to the loss of the depth information. Therefore, a 2.5D gating is proposed in the present work to tackle the abovementioned issues. As illustrated in Figure 4.2, the gating area of an arbitrary tracklet at a certain frame is defined as an elliptical cylinder, where the ellipse corresponds to the 2D Mahalanobis distance gating in image coordinate  $d_{ij,\epsilon}$ , and the length of the cylinder is determined by a maximal depth motion within two adjacent frames  $d_{z,\epsilon}$ . A detection will only be regarded as a potential candidate if it fulfills

$$d_{ij} < d_{ij,\epsilon} \quad \& \quad d_z < d_{z,\epsilon}, \quad (4.8)$$

where the  $ij$  refers to the image coordinates instead of a specific tracklet or detection.



**Figure 4.2.:** Gating elliptical cylinder. The center of the elliptical cylinder locates at the prediction.

The initial tracking process occurs two-dimensionally in image coordinates. Therefore, each prediction owns a position with two image coordinates. Meanwhile, the prediction takes the depth coordinate of the last assigned detection as its apriori depth, which is considered the origin of the gating in depth afterward. Principally, the x-y coordinates system delivered by the plenoptic camera is an alternative substitute for the image coordinates system. Nevertheless, the x and y coordinates computed by the camera internally in accordance with the depth coordinate could also reflect the fluctuation to some extent. Since fluctuation is an unfavorable condition in the tracking process, especially when using a linear Kalman filter, the particles are thus followed in image coordinates. Moreover, utilizing the image coordinates instead of the x-y coordinates benefits an objective and accurate assessment of the tracking performance, since the ground truth is generated optically on the basis of 2D captured images.



### 4.1.3. Modified data association approach

With the completion of the gating process, the predictions are then associated with the candidate detections within the corresponding gating elliptical cylinders to extend the tracks to current frames. In the present work, both GNN and JPDA are utilized to track the fuel particles. As demonstrated in Chapter 2.3, the two algorithms begin to diverge significantly at the assignment step. The GNN algorithm attempts to find the sole global minimal cost of the assignments, whereas the JPDA approach considers the probability of each joint event and weights all candidate detections within the gate. In the following, the algorithms specialized for the work are illustrated.

#### GNN

The GNN algorithm assigns detections to predictions using a generated cost matrix, each of whose elements stands for the cost of the association of a particular detection to a particular tracklet prediction. The universal cost is defined in Equation 2.50. This work modifies the cost by adding an ancillary cost regarding particles' sizes represented by the number of points in a cluster. If a detection  $j$  is inside the gate of a prediction  $i$ , the cost of assigning  $j$  to  $i$  is thereby defined as

$$c_{ij} = d_{ij}^2 + \ln |P_{i,t}| + a_0 |N_d - N_p|, \quad (4.9)$$

where the first two terms are directly adopted from Equation 2.50 in Chapter 2.3 [50]. The last term is the supplemented term weighted by a factor  $a_0$  to compensate for the magnitude difference between the terms. The term denotes the absolute difference in the point numbers between the prediction and detection., where  $N_d$  stands for the point number of the candidate detection, and  $N_p$  equals the amount of point within the last assigned detection of the tracklet. The point number of a cluster is able to show the size of a particle since each spatial point is commensurate with a pixel in the image coordinate system. The point number is also a simple but efficient indicator of particle appearance similarity. Apparently, the similarity could also be computed alternatively, for instance, by extracting certain features of the particles. However, the fuel particles of the same fraction differ hardly from each other, leading to an extremely challenging difficulty in searching for appropriate features. Moreover, even if suitable features are found, the entailed computational load increases dramatically. Therefore, the cost element contains only the point number of a cluster as the particles' appearance similarity. As presented in Chapter 3.1, the camera might capture the particles incorrectly due to the overexposure caused by combustion. This problem can be solved by decreasing  $a_0$ , which can consequently reduce the weight of the point number in the cost function. The cost for the assignment of outside gate detections  $C$ , as introduced in Chapter 2.3.2, ought to be chosen much larger than the cost elements to avoid false assignments.

For the purpose of a precise and conveniently controllable assignment, another parameter  $C_{NA}$  concerning cost threshold is introduced and defined as the cost of non-assignment. If the cost of an assignment exceeds  $C_{NA}$ , this assignment will be regarded as impossible with an updated cost equaling  $C_{NA}$ . Introducing  $C_{NA}$  aims to impose restrictions on candidate detections within the gate. Suppose an extreme case where only one false positive detection with a sizeable assignment cost is inside the gate of a prediction. For instance, the detection is far from the prediction, and (or) their numbers of points are highly distinct. Without the parameter, the detection would be wrongly associated with the prediction. Meanwhile, the parameter  $C_{NA}$  can conveniently control the assignment by selecting different values. For example, if  $C_{NA}$  and  $C$  are equal in value, the impact of  $C_{NA}$  can hardly be noticed. On the contrary, a small  $C_{NA}$  can substantially affect the association process by excluding numerous candidate detections within the gate.

As an alternative to using depth information only for gating, the depth information can also be integrated into the cost function of Equation 4.9. Nevertheless, the depth information fluctuates severely and irregularly, leading to almost unpredictable cost values. Thus, the depth information is only utilized in the previous gating process.

### JPDA

As mentioned in Chapter 2.3.3 and 2.3.4, the JPDA approach weights all candidate detections within the gate following the respective probability  $\beta$ . For PDA,  $\beta$  is generally computed by a predefined rule, while for JPDA,  $\beta$  ought to be calculated separately in accordance with each joint event afterward. The initial  $\beta_j$  for an arbitrary prediction  $i$  with totally  $m(i)$  detections in gate is defined as [47]

$$\beta_j(i) = \begin{cases} \frac{e_j}{b + \sum_{j=1}^{m(i)} e_j} & j = 1, \dots, m(i) \\ \frac{b}{b + \sum_{j=1}^{m(i)} e_j} & i = 0 \end{cases}, \quad (4.10)$$

where

$$e_j = e^{-\frac{1}{2}c_{ij}} \quad (4.11)$$

denotes the unnormalized probability of assigning detection  $j$  to the prediction with  $c_{ij}$  representing the cost function of Equation 4.9. By introducing this newly defined cost, as in the above GNN approach, JPDA also takes the changes in points numbers into account. In Equation 4.10,  $b$  is termed as the probability of non-assignment and equals

$$b = \lambda |2\pi P_i|^{\frac{1}{2}} \frac{1 - P_D P_G}{P_D}. \quad (4.12)$$

In the expression for the non-assignment probability,  $\lambda$  stands for a certain spatial density, which can be replaced by the sample spatial density of the measurements [47], as

$$\lambda = \frac{m(i)}{V(i)} \quad (4.13)$$

where  $V(i)$  is the volume of the ellipical validation region. In Equation 4.12,  $P_i$  is the corresponding covariance matrix mentioned before,  $P_D$  denotes the detection probability that  $i$  evokes a detection, and  $P_G$  stands for the gating probability considering a detection correctly falling into the gating ellipse [33]. Grinberg [33] illustrated computing  $P_G$  based on the Gaussian distribution assumption. Apparently, the larger the  $P_D$  and  $P_G$ , the lower the probability of non-assignment. In the study, a possibly large gating threshold could increase  $P_G$ . Furthermore, the value of  $P_D$  can also be appropriately large since the trajectories are relatively continuous. In addition, since the detections on the inner wall of the rotary kiln have been previously removed and the detection method has been improved, reducing the probability of non-assignment in the case of having detections within the gate is conducive to facilitating tracking accuracy. Based on the probability  $\beta$ , the probability regarding the joint events are computed following definitions in Chapter 2.3.4. Similar to GNN, the depth information is considered in the gating process instead of explicitly being involved in the data association process.

#### 4.1.4. Update of state vectors

The last step of multiple particle tracking is to update the state vector and its corresponding covariance matrix according to the assignment outcomes. For GNN, the state vector and the covariance matrix

of 2D image coordinates are updated by Equation 2.46-2.48, respectively. The update of the JPDA algorithm refers to Equation 2.52-2.56. The measurement matrix required in the update functions to map measurements to state vectors is detailed as

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

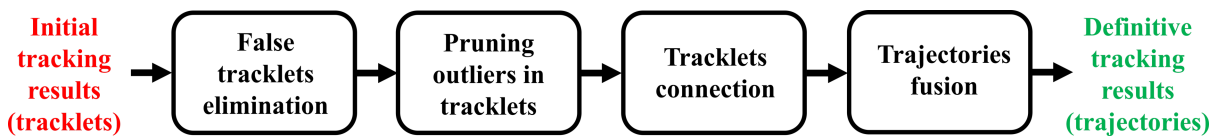
since only the x and y coordinates of the detections are available, *i.e.*, particle velocities were not measured. Later, the depth information simply takes the depth value of the assigned detections. Unassigned tracklets will not be updated. If a tracklet remains unassigned for a certain period, the tracklet will be deleted from the current tracklist. During the prediction process, the error covariance matrix of a track is enlarged, *i.e.*, the prediction can fluctuate in a broad range, while the update step shrinks the covariance matrix and thereby reduces the spread of the predictions. If a tracklet remains unassigned for a while, its covariance matrix will become extremely large. In spite of a penalty term for large covariance in the cost function defined in Equation 4.9, a tracklet with a sizeable gating caused by the large covariance matrix will adversely impact the accuracy of the data association. Hence, the following conditions are defined to remove tracks that have been unassigned for a long time,

$$v = \frac{V}{\text{Age}} < v_{\epsilon} \quad \text{or} \quad CIC > CIC_{\epsilon}. \quad (4.14)$$

In the equation,  $v$  is the visibility defined as the ratio of total successfully assigned times  $V$  and the age of the track. The age of a track stands for the overall appearance frames of the track, regardless of the assignment. Once the visibility of a specific track is under a threshold  $v_{\epsilon}$ , *viz.*, a relatively small percentage of assigned frames in total, the track ought to be deleted from the list. Additionally, if the consecutive invisible count  $CIC$  of a track is beyond the limited value  $CIC_{\epsilon}$ , the track will not be considered in the following frames as well. As for unassigned detections, based on each unassigned detection, a new track is established.

## 4.2. Novel post-processing framework

Notwithstanding the efforts invested in multiple particle detection and tracking approaches, the straightforwardly obtained tracking-by-detection results fail to deliver satisfactory accuracy and reliability for further analysis of fuel particles' aerodynamic and combustion behaviors. On account of detection inaccuracy, such as false positive detections and missing (FN) detections, tracking errors, *e.g.*, incomplete tracklets and faulty tracklets, persist in the preliminary tracking outcomes. In order to address these issues and thereby optimize the tracking performance, a post-processing framework for multiple particle tracking-by-detection is proposed and demonstrated in the present work, which is initially introduced in [130] and modified by [133] afterward. A concise and schematic overview is shown in Figure 4.3. All acquired particle traces before post-processing are recognized as tracklets regardless of their actual lengths and durations. The traces can be ultimately identified as trajectories after the post-processing. Withal, the notations with subscript  $\epsilon$  indicate threshold values whose exact values are elaborated in Chapter 5. In the following, the presented steps in Figure 4.3 are described gradually.



**Figure 4.3.:** Schematic overview of the novel post-processing framework.

### 4.2.1. False tracklets elimination

Firstly, false tracklets ought to be excluded from the tracklist to avoid their adverse influence on further steps. The tracklets that are extremely short, moving noticeably slowly, or too irregularly are regarded as false tracklets. The following equation gives quantitative constraint conditions of remaining tracklets. A tracklet  $T_i$  can only persist if it fulfills the requirements

$$\mathcal{D}(T_i) > \mathcal{D}_\epsilon \quad \& \quad |\bar{V}_i| > \bar{V}_\epsilon \quad \& \quad \mathcal{R}_{T_i \tilde{T}_i} > \mathcal{R}_\epsilon. \quad (4.15)$$

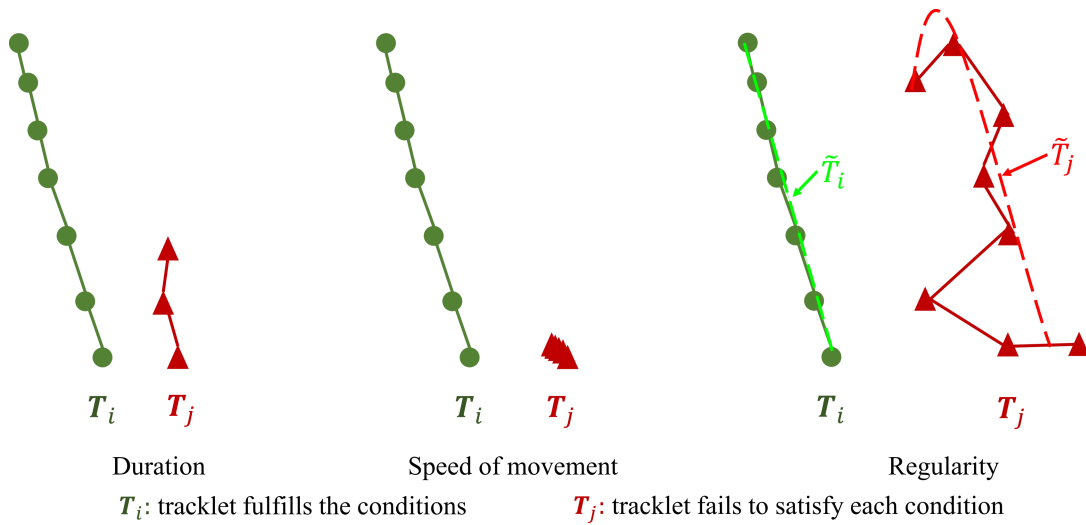
The three conditions restrict the duration, motion, and regularity of a tracklet, respectively. The first term requests a minimum duration of  $T_i$ , viz., the duration of  $T_i$  ( $\mathcal{D}(T_i)$ ) must be beyond a predefined limitation  $\mathcal{D}_\epsilon$ . Subsequently, if the  $i$ th particle moves considerably slowly or even non-moving, such as particles landing on the inner wall of the kiln, its motion is outside the research interest and hence should be pruned. The movement of the particle is represented by the arithmetic average magnitude of the temporary velocity vector  $|\bar{V}_i|$  that is formulated as,

$$|\bar{V}_i| = \frac{\sum_{t=t_1}^{t=t_{end}-1} |V_{i,t}|}{\mathcal{D}(T_i) - 1} \quad (4.16)$$

with

$$V_{i,t} = \frac{\mathbf{T}_{i,t+\Delta t} - \mathbf{T}_{i,t}}{\Delta t} \quad (4.17)$$

denoting the temporary velocity of  $T_i$  at time  $t$ . Here should be noted that the bolded  $\mathbf{T}_{i,t}$  represents the position vector of a tracklet  $T_i$  at a frame  $t$ , while the unbolded  $T_i$  simply refers to the  $i$ th tracklet. The temporary velocity  $V_{i,t}$  is computed as the displacement within two consecutive positions  $\mathbf{T}_{i,t}$  and  $\mathbf{T}_{i,t+\Delta t}$  divided by the corresponding time difference  $\Delta t$ . The second condition deletes the tracklets with an average arithmetic magnitude below the threshold value  $\bar{V}_\epsilon$ . The third term of the Equation 4.15 places restrictions on the regularity of the tracklet. In the light of the manually labeled ground truth trajectories, the 2D particle traces roughly approximate second-degree polynomials. Therefore, the cross-correlation coefficient of tracklet  $T_i$  and  $\tilde{T}_i$ , which stands for a second-degree polynomial approximation of  $T_i$ , is computed and compared to the limitation  $\mathcal{R}_\epsilon$ . With a coefficient  $\mathcal{R}_{T_i \tilde{T}_i}$  over  $\mathcal{R}_\epsilon$ ,  $T_i$  is able to prove its regularity preliminarily. Figure 4.4 schematically illustrates these three



**Figure 4.4.:** Examples of eligible and ineligible tracklets. Each condition is accompanied by two instances. Tracklets in green are eligible tracklets, while red tracklets are false tracklets.

conditions visually, and for each condition, instances of tracklets meeting and not meeting the criteria are presented. Only the tracklets with the abovementioned three measures exceeding the defined threshold values remain for further processing.

#### 4.2.2. Deleting outliers in tracklets

On account of false positive detections persisting in retained tracklets, the outliers in each tracklet will be subsequently identified and pruned. The majority of the outliers are a consequence of clutter and could result in irregular local peaks in tracklets. Deleting outliers is conducive to computing potential tracklet matching pairs for the next step and, thus, is regarded as a preparation for further tracklet connection. In the present work, the change of the velocity vector, both in magnitude and angle, of two consecutively assigned frames is defined as the criterion for judging outliers. A detection belonging to tracklet  $i$  at time  $t$  is identified as an outlier, if

$$\theta_{i,t-1} > \theta_\epsilon \quad \& \quad |\Delta \mathbf{V}_{i,t-1}| > \Delta \mathbf{V}_\epsilon, \quad (4.18)$$

with

$$|\Delta \mathbf{V}_{i,t}| = |\mathbf{V}_{i,t} - \mathbf{V}_{i,t-1}| \quad (4.19)$$

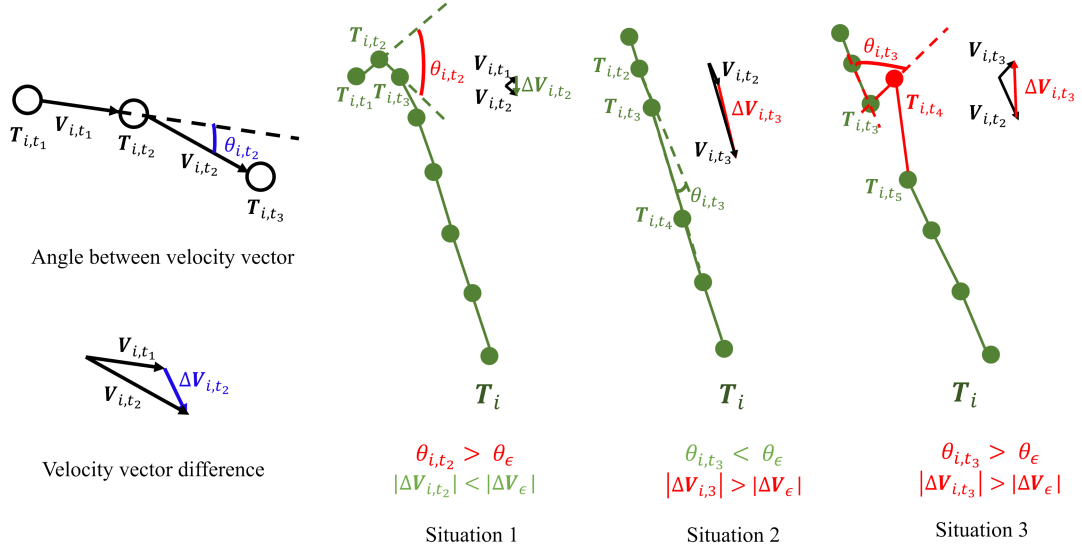
denoting the magnitude of the successive velocity change and  $\theta_{i,t}$  standing for the corresponding angle, as depicted in Figure 4.5. Additionally, three situations concerning the criteria in the Equation 4.18 are also illustrated. As enlightened by the figure, the fulfillment of one condition (angle or magnitude) is insufficient to identify outliers. The outliers ought to satisfy both requirements simultaneously. Once an outlier is detected, it will be provisionally removed from the tracklet, and then the entire scanning process is repeated until no outlier is found. Finally, the marked outliers are pruned. In practice, latent outliers are found by extrapolating forwards, starting from the first position. Notwithstanding, the solution could give rise to erroneous identifications for the second assigned detections, as illustrated in Figure 4.6. Following the definition of Equation 4.18,  $T_{i,t_3}$  could be inaccurately defined as an outlier. To tackle the issue, the second assigned detection is examined backwards from the last one with the modified condition,

$$\theta_{i,t+1} > \theta_\epsilon \quad \& \quad |\Delta \mathbf{V}_{i,t+1}| > \Delta \mathbf{V}_\epsilon. \quad (4.20)$$

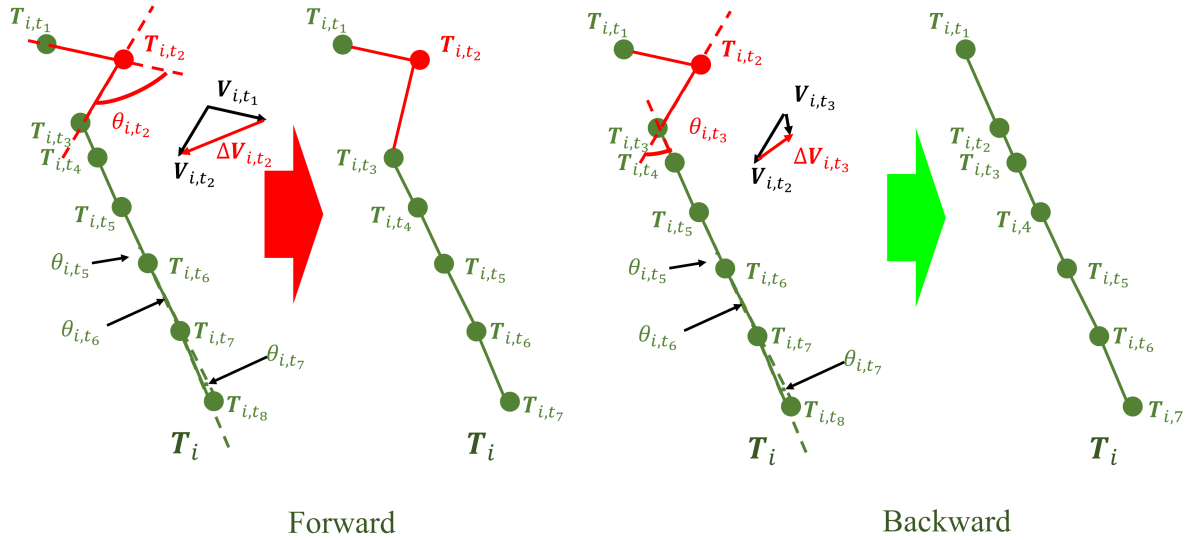
As revealed by Figure 4.6, the second assigned detection can be identified and pruned. Subsequently, the identification of outlier is conducted according to the previous description.

Apparently, the above-elaborated approach can only screen for outliers except for the initial position. The initial position lack a velocity vector. For the first point, the determination is made according to the average of the following three velocities, as depicted in Figure 4.7. The average vector is thereby assumed to be the following velocity. The outlier detection is performed according to Equation 4.18. Checking whether the first point is an outlier should be performed before the backward scanning to avoid impact on the judgment of the following points.

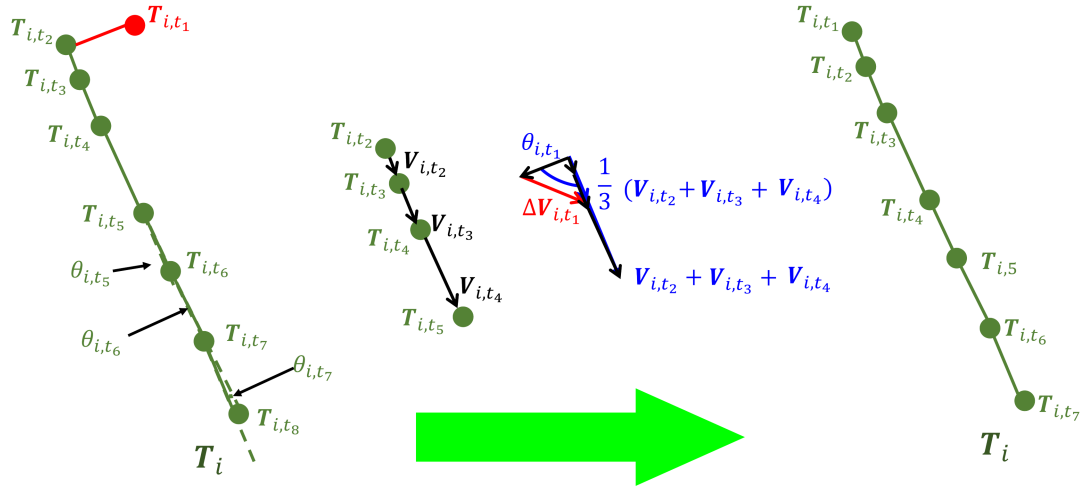
Trajectory outlier detection is a vital data processing research topic that increasingly attracts researchers' interest. For instance, Knorr *et al.* [48] studied the notion of distance-based outliers for the sake of detecting exceptions in large, multidimensional datasets. The trajectory is transformed into an object consisting of three independent attributes: position, direction, and velocity. Then the traditional distance function is applied to calculate the distance between trajectories to detect outliers. Further, Li *et al.* [61] proposed a novel framework ROAM (Rule- and Motif-based Anomaly Detection in Moving Objects) for efficiently and effectively identifying anomalies. ROAM expresses trajectories with discrete pattern fragments (motifs) and extracts associated features to build feature space, which is then explored by a rule-based classifier. Besides, a trajectory outlier detection algorithm TRAOD, which



**Figure 4.5.:** Schematics of the notations for outlier detection and three situations concerning the outlier criteria. Green circles stand for inliers, while red circle represent outlier. Situation 1: the angle between  $V_{i,t_1}$  and  $V_{i,t_2}$  satisfies the first condition, whereas the vector difference magnitude ( $|\Delta V_{i,t_2}|$ ) does not exceed the limit. This situation may be a consequence of several particles falling after rising slightly due to their inertia, which is not able to judge outliers. Situation 2: the velocity difference magnitude  $|\Delta V_{i,t_3}|$  satisfies the second condition, but the angle  $\theta_{i,t_3}$  is relatively small. The main reason is the cross-frame assignment. The two positions  $T_{i,t_3}$  and  $T_{i,t_4}$  are from two adjacently associated time points instead of two consecutive frames. Consequently,  $|\Delta V_{i,t_3}|$  is beyond the threshold, while  $\theta_{i,t_3}$  is small. Situation 3:  $T_{i,t_4}$  fulfills the both conditions and should be pruned.



**Figure 4.6.:** Forward and backward computing of outliers in Situation 3 of Figure 4.5. For forward scanning,  $T_{i,t_3}$  could be wrongly identified as an outlier because of the unfavorable influence of the real outlier  $T_{i,t_3}$ . Reversely, computing both forward and backward can address the issue since the problem does not persist in computing backward.



**Figure 4.7.:** Checking the last point. The average of the last three velocity vectors  $V_{i,t_3}$ ,  $V_{i,t_4}$ ,  $V_{i,t_5}$  is assumed the next velocity vector for computing of Equation 4.18.

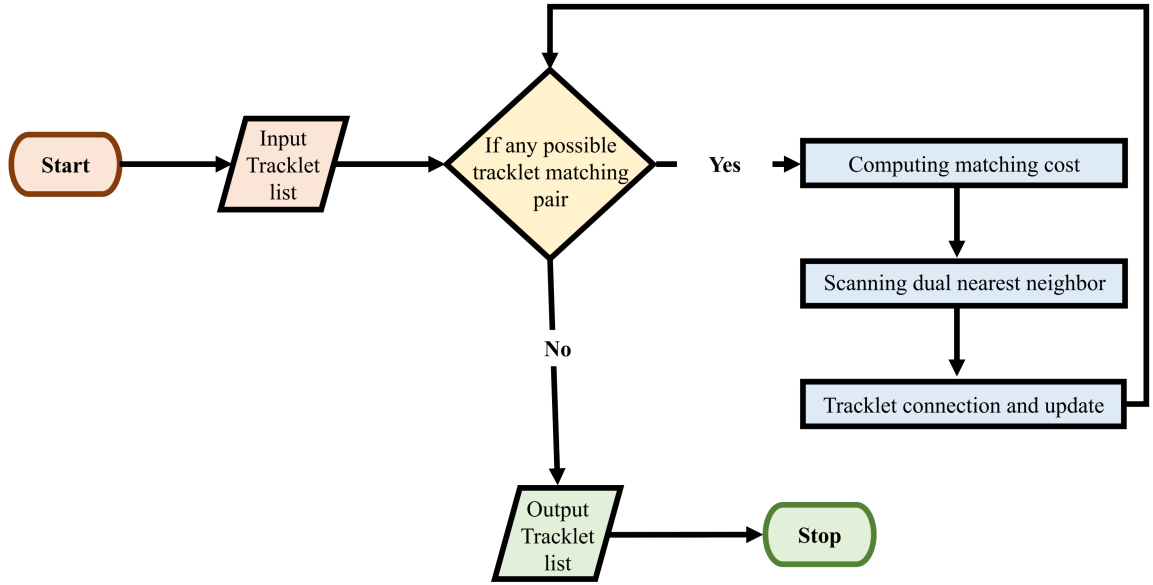
partitions a trajectory into several line segments and detects outlying line segments for trajectory outliers subsequently, is developed by Lee *et al.* [57]. As proved by the authors, TRAOD is able to detect outlying parts of trajectories. Enlightened by TRAOD, Liu *et al.* [125] recommended a Trajectory Outlier Detection Based on Local Outlier Degree (TraLOD) that uses a degree to express outlying features. The approach introduces the relative distance of segments and shows a satisfactory performance of detecting trajectory outliers. These mentioned algorithms can accurately target anomalous segments in an extensive trajectory database and thus are adequate for analyzing suspicious motion. However, these algorithms also require a high computational cost. Since the particle trajectories in the present study are relatively regular, the algorithm described above (Equation 4.18) can deliver outcomes with sufficient accuracy. Therefore these complex algorithms are not further discussed in the work.

#### 4.2.3. Tracklets connection and merging

After accomplishing the previous two preparation steps, the remaining tracklets can be connected or merged into trajectories. Figure 4.8 schematically illustrates the connection and merging approach, which is developed as an iterative process. The process begins with scanning for potential tracklet matching pairs among the entire tracklet list. Subsequently, the corresponding pairing cost of each matching pair is computed, which constructs the cost matrix afterward. In accordance with the cost matrix, the bilateral nearest neighbor matching pairs are allowed for connection and merging. Afterward, the tracklet list is updated accordingly. The entire procedure will be repeated until there is no possible tracklet matching pair within the current tracklet list. In the following, the individual steps are elaborated.

##### Criteria for possible tracklet matching pair

At first, two cases with respect to time overlap ought to be highlighted and pinpointed by means of Figure 4.9. If a potential tracklet matching pair  $T_i$  and  $T_k$  with  $T_i$  appears earlier than  $T_k$ , two situations



**Figure 4.8.:** Flowchart of tracklet connection.

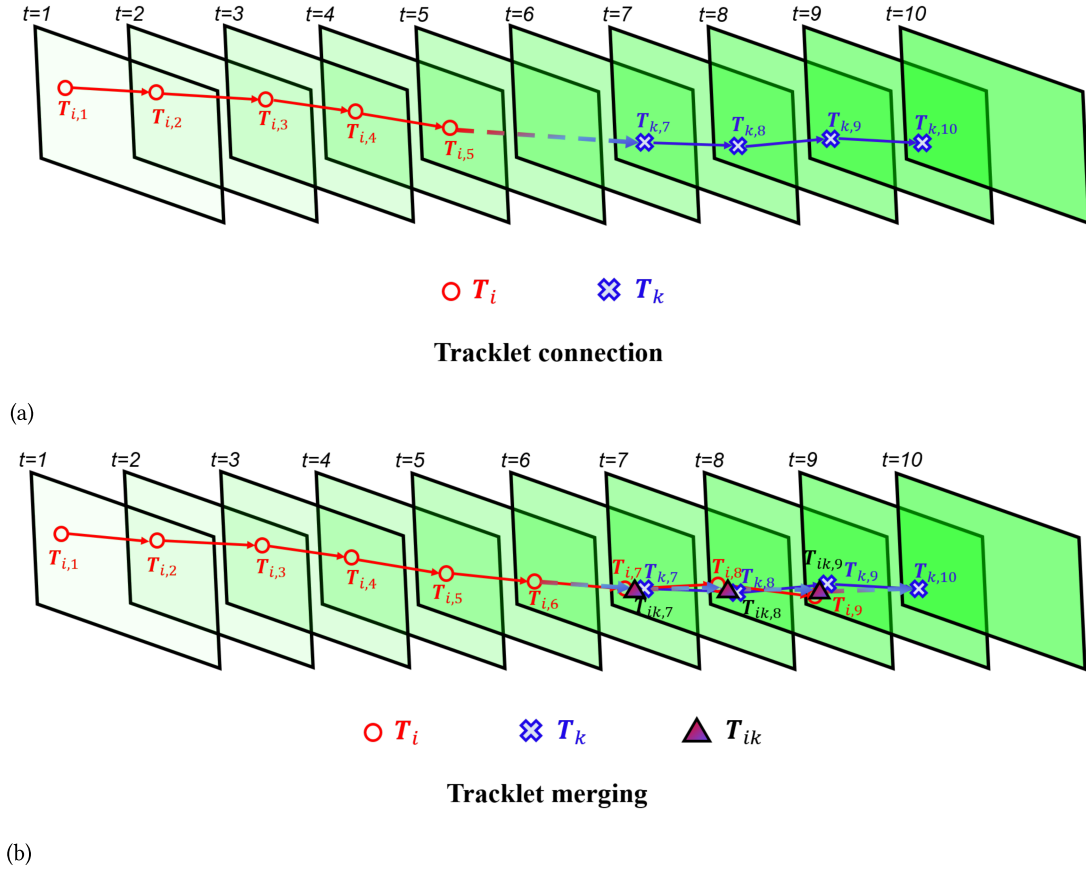
can be expected regarding the temporal overlap, as presented in Figure 4.9. If  $T_i$  and  $T_k$  do not overlap in the timeline, their matching is specified as the tracklet connection. Conversely, if the matching pair  $T_i$  and  $T_k$  appear partially simultaneously, their matching corresponds to the tracklet merging. In summary, temporal discontinuity is a sign of tracklet connection, while temporal overlap is a prerequisite for tracklet merging. The ideal situation is a complete trajectory, *i.e.*, neither gaps nor overlaps. Therefore, two tracklets should have as few temporal gaps or overlaps as possible in order to be regarded as potential matching pairs. Only if the time gap or time overlap of  $T_i$  and  $T_k$  varies within a reasonable range can they preliminarily be considered as a potential matching pair, *viz.*,

$$t_{T_k,1} - t_{T_i,end} < t_{\epsilon,connect} \quad \text{or} \quad KN_{T_i} \cap KN_{T_k} < kn_{\epsilon,merge}. \quad (4.21)$$

where  $t_{T_k,1}$  stands for the initial time of  $T_k$ , while  $t_{T_i,end}$  denotes the ending time of  $T_i$ . Based on the fact that a tracklet probably might not be successfully assigned for each frame, the parameter  $N$  is introduced, which denotes a set of discrete appearance times of a particular tracklet. For instance,  $N_{T_i}$  in Figure 4.10 right equals  $[1,3,4,5,7,9,10,11]$ , while  $KN_{T_k}$  is  $[8,9,11,13,14,15,16]$ . Under this circumstance,  $KN_{T_i} \cap KN_{T_k}$  stands for the cardinality overlap, *viz.*,  $[9,11]$  with a size of 2. Additionally,  $t_{\epsilon,connect}$  and  $kn_{\epsilon,merge}$  are predefined thresholds with subscripts indicating distinct values for tracklet connection and merging, respectively. While  $t_{\epsilon,connect}$  restricts the time gap of the matching candidate,  $kn_{\epsilon,merge}$  constraints the simultaneous time cardinality overlap to a rational range.

In addition to the temporal requirements demonstrated in Equation 4.21, further criteria to impose restrictions on potential tracklet matching pairs with respect to the connected part are also defined. Since the locations and tendencies of the tracklets can vary significantly, criteria apart from temporal limitations are pretty vital. These newly introduced restrictions basically constrain the flatness and smoothness of the connected part of the two tracklets, which is presented in Figure 4.10. Thereby, the notations in Figure 4.9 are followed, where  $T_i$  denotes the antecedent tracklet, and  $T_k$  stands for the subsequent tracklet. The connected part  $T_{ik}$  is underlined by black dashed rectangles. For tracklet connection, the connected part consists of three last temporary detections of the antecedent tracklet





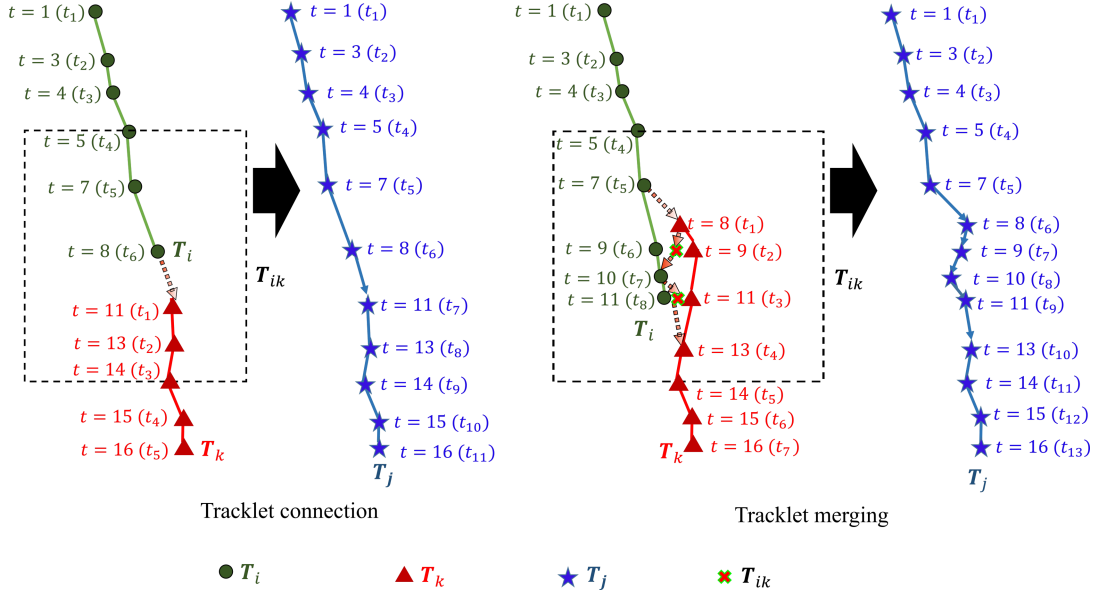
**Figure 4.9.:** Tracklet connection and merging [130]. (a) Tracklet connection:  $T_i$  and  $T_k$  do not overlap temporally and are simply connected. (b) Tracklet merging:  $T_i$  and  $T_k$  overlap temporally. For the overlapping time, the tracklet is represented by the average arithmetic positions of  $T_i$  and  $T_k$ .

(e.g., Figure 4.10,  $T_i$ :  $t=5, 7$ , and  $8$  ( $t_4, t_5, t_6$ ))) together with three first detections of the subsequent tracklet (e.g., Figure 4.10,  $T_k$ :  $t=11, 13$ , and  $14$  ( $t_1, t_2, t_3$ ))). For tracklet merging, the first temporally coincident detection (e.g., Figure 4.10,  $T_i$ :  $t=9$  ( $t_6$ )) and its two previous detections of the antecedent tracklet (e.g., Figure 4.10,  $T_i$ :  $t=5$  and  $7$  ( $t_4, t_5$ )), as well as the last temporally coincident detection (e.g., Figure 4.10,  $T_i$ :  $t=11$  ( $t_8$ )) and its following two detections from the subsequent tracklet (e.g., Figure 4.10,  $T_j$ :  $t=13$  and  $14$  ( $t_4, t_5$ )), belong to the connected part.

In order to be potential matching pair, two tracklets  $T_i$  and  $T_k$  must fulfill the following requirements,

- No outliers exists in the connected part  $T_{ik}$  according to Equation 4.18. As the outliers in the original tracklets are already pruned, any outlier in  $T_{ik}$  can forecast an inadequate connection of  $T_i$  and  $T_k$ . Moreover, this requirement also avoids connecting two trajectories far apart in the 2D image coordinates since, by definition, an outlier exists in this case.
- The connected part is not a false tracklet in line with Equation 4.15 but without the duration limitation.
- An adequate tracklet connection ought to avoid significant fluctuation of the connected part, viz.,

$$\Delta z_{ik,t} < \Delta z_\epsilon \quad (4.22)$$



**Figure 4.10.:** Tracklet connection part.  $T_i$  and  $T_k$  is a possible matching pair that is connected or merged to build a new tracklet  $T_j$ . Left illustrates the connection of tracklet  $T_i$  and  $T_k$ , while the tracklet merging is shown on the right. Dashed rectangles mark the respective connected parts  $T_{ik}$ . In the figure, the time cardinality of each tracklet is also indicated in parentheses to distinguish it from the actual time (frame). Obviously, unlike the discontinuous actual time of each tracklet, the time cardinality is always continuous.

with

$$\Delta z_{ik,t} = \left| \frac{z_{ik,t+\Delta t} - z_{ik,t}}{\Delta t} \right| \quad (4.23)$$

denoting the temporally averaged depth difference between two successively assigned frames.

### Matching cost

With the possible tracklet pairings identified, their pairing costs are computed on the basis of the following expression

$$C(T_i, T_k) = a\bar{\theta}_{ik} + b|\bar{\Delta V}_{ik}| + c\bar{\Delta z}_{ik} + e|KN_{T_i} \cap KN_{T_k}| + d \cdot \max((t_{T_k,1} - t_{T_i,end}), 0) + \frac{f}{\max(\bar{F}_{ik}, 1)}. \quad (4.24)$$

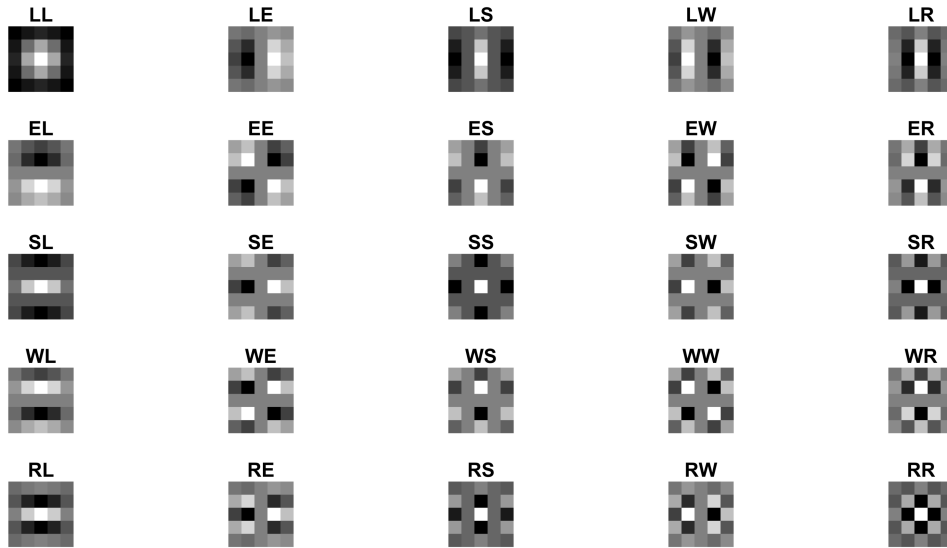
These six terms represent the angle, velocity, depth, time overlap, time gap, and connection feature of the tracklet connection, respectively. The first term  $\bar{\theta}_{ik}$  denotes the arithmetic average of  $\theta_{ik,t}$  with the unit degree;  $|\bar{\Delta V}_{ik}|$  stands for the arithmetic average of  $V_{ik,t}$  with the unit pixel per frame;  $\bar{\Delta z}_{ik}$  is the arithmetic depth difference between frames with the unit millimeter per frame. These three terms are conducive to selecting smooth connections. If two tracklets are not adequate for matching, such as being far away from each other, a high cost can be expected. The fourth and fifth terms penalize the connection with long time gaps or temporal overlaps. For tracklet connection, the fourth term equals zero, as the matching pair do not overlap. Correspondingly, the fifth term matters only for tracklet merging since the time gap does not persist in tracklet merging. Thereby,  $|KN_{T_i} \cap KN_{T_k}|$  denotes the number of elements of the intersection of the sets of appearance time.

The last term is derived from the laws' properties (also known as texture energy measures) [56], that extracts texture features by virtue of filtering images with various masks. In the research, Laws [56]

recommended five one-dimensional convolution masks, Level, Edge, Spot, Wave, and Ripple, that are generated by convolving two 1x3 vectors, as presented by the following expressions,

$$\begin{bmatrix} L5 \\ E5 \\ S5 \\ W5 \\ R5 \end{bmatrix} = \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ -1 & -2 & 0 & 2 & 1 \\ -1 & 0 & 2 & 0 & -1 \\ -1 & 2 & 0 & -2 & 1 \\ 1 & -4 & 6 & -4 & 1 \end{bmatrix}. \quad (4.25)$$

The vectors are ordered by sequency and are weighted toward the center. Moreover, the vectors are independent but not orthogonal. The five masks are designed to identify different features. *L5* aims to compute the symmetric weighted local average; *E5* is able to detect borders and edges; *S5* is applied to recognize spots. As suggested by the name, *W5* can detect potential wave shapes, while *R5* is utilized to identify image as rippled [54]. Additionally, the author suggested combining these masks to build two-dimensional masks that could facilitate feature extraction performance. Based on the listed five masks, 25 combinations can be produced, as presented in Figure 4.11. Convolving these combinations with a texture image will accentuate the microstructure of the texture [21].

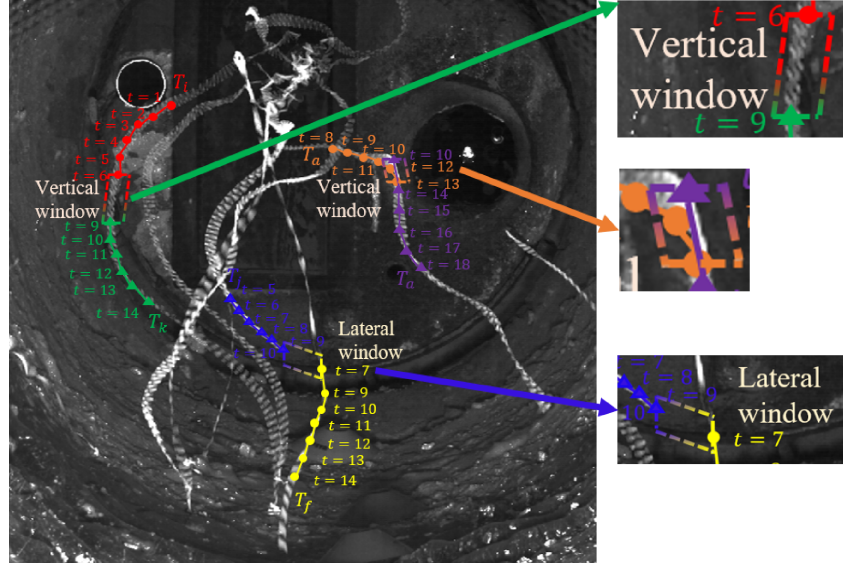


**Figure 4.11.:** 25 combinations of Law's masks for feature identification.

The mask is subsequently applied on the sum of frame differences acquired by adding a set of consecutive frame difference images within the time period of the tracklets' appearance, as figured in Figure 4.12. The computing window is schematized on the figure as well, which is defined as a parallelogram whose vertexes depend on the dominant connection of the tracklets. For a vertically dominant connection, the vertexes have the following localization,

$$\left(T_{i,t_{T_i,end}}^c - w, T_{i,t_{T_i,end}}^r\right), \quad \left(T_{i,t_{T_i,end}}^c + w, T_{i,t_{T_i,end}}^r\right), \quad \left(T_{k,t_{T_k,1}}^c - w, T_{k,t_{T_k,1}}^r\right), \quad \left(T_{k,t_{T_k,1}}^c + w, T_{k,t_{T_k,1}}^r\right).$$

Accordingly, a lateral connection entails a parallelogram with the vertexes

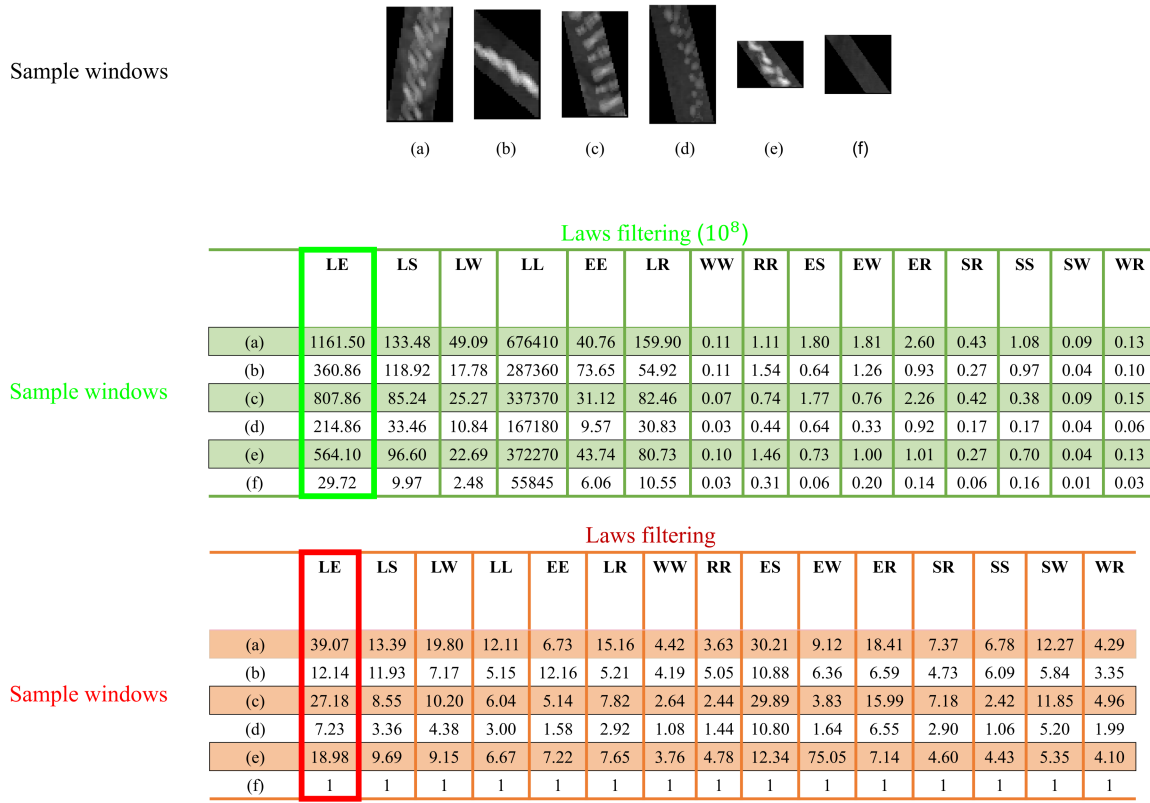


**Figure 4.12.:** Example of windows for laws' mask on the sum of difference images acquired by background subtraction [133]. The left window (red/green) is a vertical window for tracklet connection, where the vertical connection dominates. Correspondingly, the right window (blue/yellow) is a lateral window for tracklet connection. Moreover, the window (orange/purple) indicates a vertical window for tracklet merging.

$$\left(T_{j,t_{T_j,end}}^c, T_{j,t_{T_j,end}}^r - w\right), \quad \left(T_{j,t_{T_j,end}}^c, T_{j,t_{T_j,end}}^r + w\right), \quad \left(T_{f,t_{T_f,1}}^c, T_{f,t_{T_f,1}}^r - w\right), \quad \left(T_{f,t_{T_f,1}}^c, T_{f,t_{T_f,1}}^r + w\right).$$

Thereby,  $T_{i,t}^c$  stands for the column position of the tracklet  $T_i$  at time  $t$ , and  $T_{i,t}^r$  denotes the row position of the tracklet at time  $t$ , e.g.,  $T_{i,t_{T_i,end}}^c$  is the column position of tracklet  $T_i$  at the end time of the tracklet  $T_i$ . And  $w$  is the half of the window's length. The window's direction is related to the dominant connection distance of the pairing tracklets. The window size of a particular matching case depends on the locations of the corresponding tracklets and the window length  $w$ , which is selected according to the average particle size in the image coordinate. As exemplified in Figure 4.12, the red/green and orange/purple are vertical connection windows since the connection is primarily vertical. Correspondingly, the blue/yellow window indicates a lateral window for tracklet connection. Using laws' properties to identify tracklet-like features reduces the impacts of accidental illumination changes and enhances the robustness afterward. By convolving diverse mask combinations with extracted subimages with and without tracklet-like features, the combination of Level-Edge (LE) mask is selected, which can deliver the most considerable difference in values, as presented in Figure 4.13. Initially, 25 combinations could give rise to 25 filtered windows. Nevertheless, for transposed mask combinations, their corresponding filtered images are fused by taking the average value of each element resulting in 15 different varieties. Moreover, by computing the average of the transposed filtered image, vertical and lateral features are more likely to be detected simultaneously. Subsequently, the filtered image is summed over and divided by the number of non-zero pixels. The result is then listed in Figure 4.13. As revealed by the tables, the LE mask combination can best highlight the tracklet-like feature, with a maximum difference of up to 39 times and a minimum of more than 7.

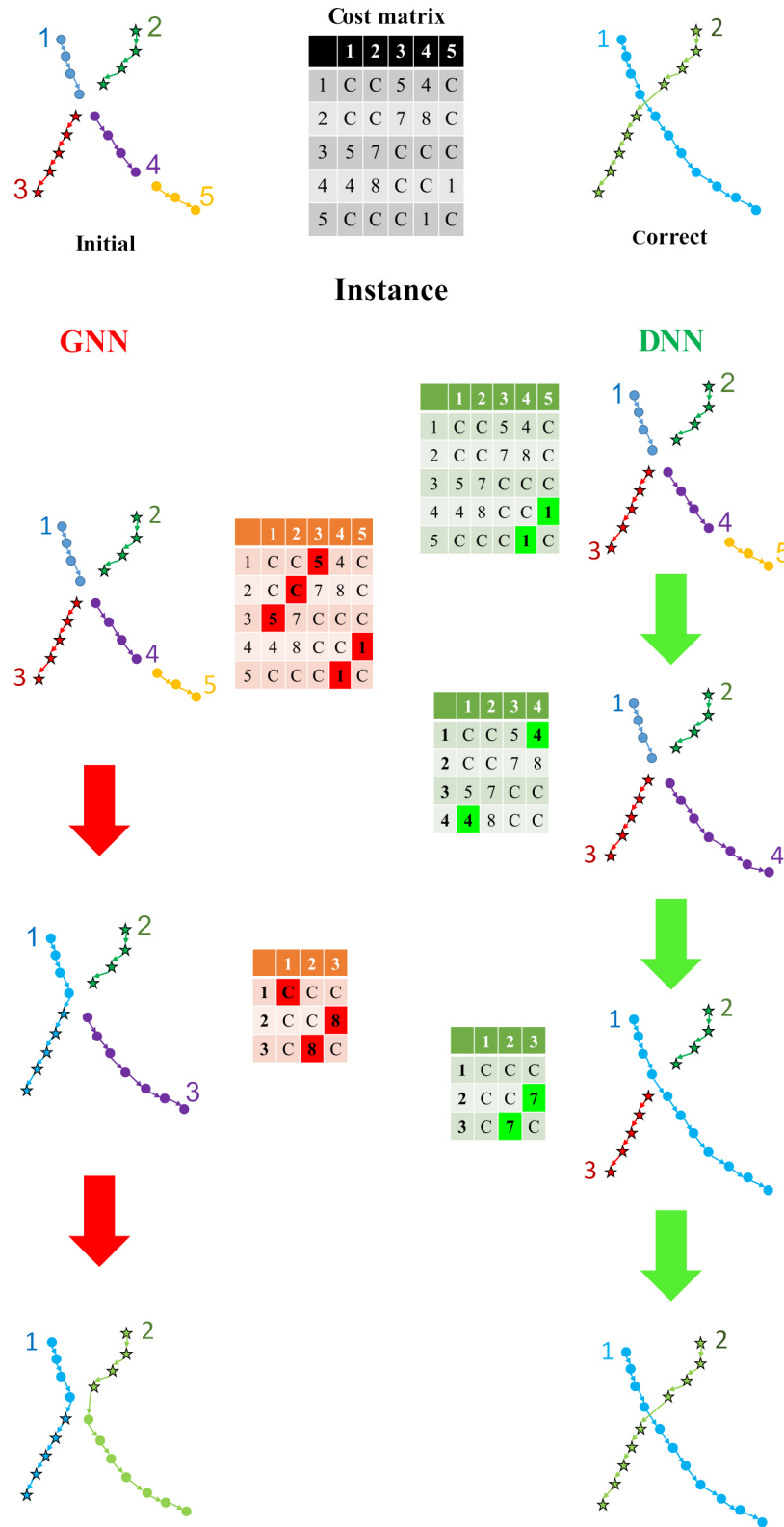
Additionally,  $a$ ,  $b$ ,  $c$ ,  $d$ ,  $e$ , and  $f$  in the cost function represent the corresponding weighting factors of each term, whose values are detailed in the next chapter.



**Figure 4.13.:** Examples of windows and their respective convolutions with diverse combinations of Laws mask. Window (f) corresponds to the window without tracklet-like features, while the remaining five are with features. The upper table shows the convolved values of order  $10^8$ , while the lower table is normalized by the corresponding bottom values, thus highlighting the contrast of values with and without tracklet-like features.

### Dual Nearest neighbor (DNN)

While the previous two steps of tracklets connection and merging tackle the problem of scanning potential matching pairs and computing their matching costs, the issue concerning processing the connection in accordance with the calculated pairing cost remains unsolved. In principle, the connection should follow the minimum cost rule, *viz.*, the lower the cost, the higher the probability of the connection. Theoretically, the Kuhn–Munkres algorithm utilized in the GNN is introduced to deal with the problem of searching global minimum cost. However, as demonstrated in Figure 4.14, the overall minimum cost is often accompanied by a compromise of local matching options leading to unfavorable outcomes. The figure illustrates the solution of the Kuhn–Munkres algorithm (GNN) and a dual nearest neighbor (DNN) approach for the exact matching problem, respectively. As mentioned before, the tracklet connection/merging is an iterative process. Once one iteration is accomplished, the tracklet list is updated. The GNN’s solution of each iteration is based on the cost matrix in orange, and the corresponding lowest cost is darkened in red. Apparently, in the first iteration, the Kuhn–Munkres algorithm mistakenly connects tracklets 1 and 3 to achieve the global minimum cost, consequently resulting in an undesired outcome. Under this circumstance, the dual nearest neighbor approach is recommended, which precedes the GNN algorithm from the performance point of view. Two tracklets are allowed to be connected only if they are the bilateral nearest neighbor, *viz.*, one tracklet is the

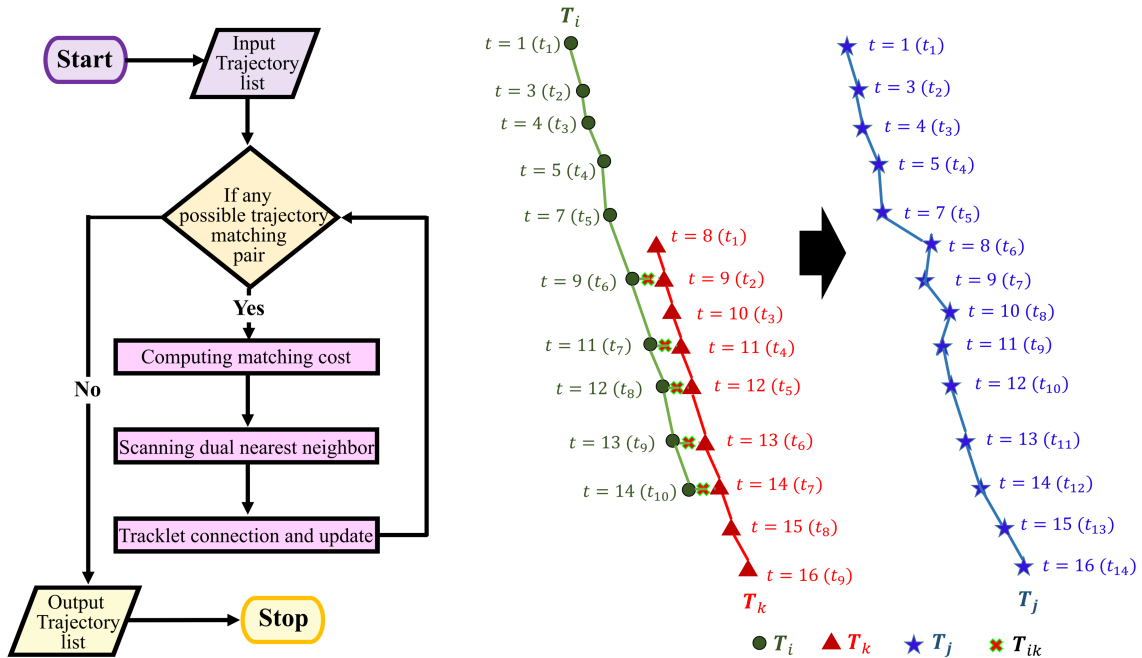


**Figure 4.14.:** Instances of tracklet connection using Kuhn-Munkres algorithm (GNN) and DNN, respectively. As described previously, the connection process is iterative. The cost matrix is updated immediately after each connection, based on which the remaining tracklets are reconnected until no potential matching pairs. Each arrow in the figure points to an iteration.  $C$  in the cost matrices indicates the impossibility of a tracklet matching pair with a tremendous cost value. Tracklets with the same marker belong to the same tracklet. [133]

nearest neighbor of the other, and the other is also the nearest neighbor of the first one. As depicted in Figure 4.14, the DNN algorithm solves the connection problem according to the cost matrix in pale green, and the dual nearest neighbors are underlined in green. Because the DNN algorithm merely connects the dual nearest neighbor pairs in each iteration, local connections are not supposed to make concessions for the sole global minimum cost. In the first iteration, only the fourth and fifth tracklets can be connected. Subsequently, the first and the fourth tracklets are matched. Finally, the second and third tracklets are coupled. In comparison to the Kuhn-Munkres algorithm that endeavors for a globally optimal solution, DNN considers the local minimum cost and succeeds in obtaining the expected result.

#### 4.2.4. Trajectories fusion

With the completion of the tracklets connection/merging, the tracklets become trajectories. For the majority of applications, the tracking process is accomplished, and the trajectories represent the ultimate outcomes. Nevertheless, one particle could trigger more than one trajectory within a certain period in case of dealing with detections containing substantial duplications and noise. Hence, a further step of fusing trajectories of one identical particle is essential. As schematically presented in Figure 4.15 left, the trajectories fusion differs only slightly from tracklets connection/merging. Generally, tracklets connection/merging and trajectories fusion follow the same iterative procedure. Potential tracklet matching pairs do not allow a long temporal overlap, but for potential fusion pairs, this constraint is unnecessary. Besides, the terms in Equation 4.24 concerning the time gap and overlap are omitted. Except for these distinctions, the rest of the procedure remains unchanged. An example of trajectory fusion is shown in Figure 4.15 right, where trajectory  $T_i$  and  $T_k$  are fused to one trajectory  $T_j$ .

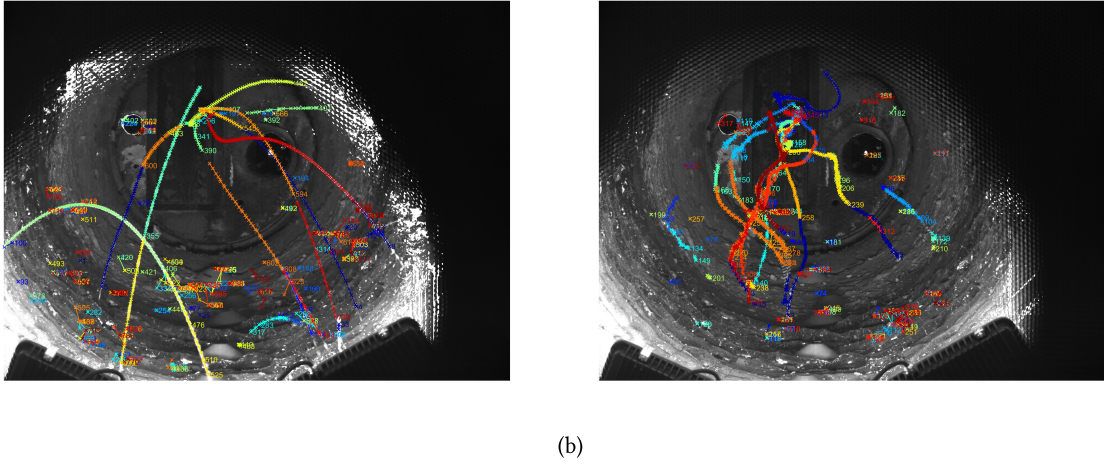


**Figure 4.15.:** Flowchart of trajectories fusion (left) and a corresponding instance of trajectories fusion (right).



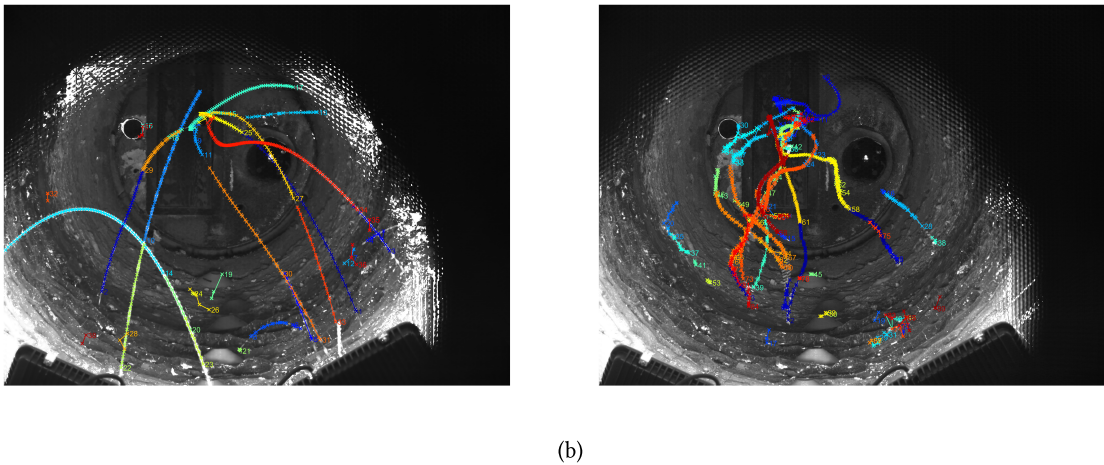
#### 4.2.5. Application examples

In order to enhance the visualization of various steps of post-processing and consequently improve the understanding of their effects, this chapter gives application examples with respect to each step of the proposed post-processing framework. Because the trajectories of different fuel particles vary significantly, two representative particle trajectories, wood chip and paper shred, are exemplified here. Figure 4.16 shows the initial trajectories of the two fractions obtained by 2.5D GNN before post-processing. As described beforehand, with the presence of tracking deficiencies, such as incomplete trajectories and outliers, the initial tracking results fail to provide satisfactory tracking performance. These trajectories are processed by the proposed novel post-processing framework subsequently.



**Figure 4.16.:** Initial tracking results of wood chips (a) and paper shreds (b).

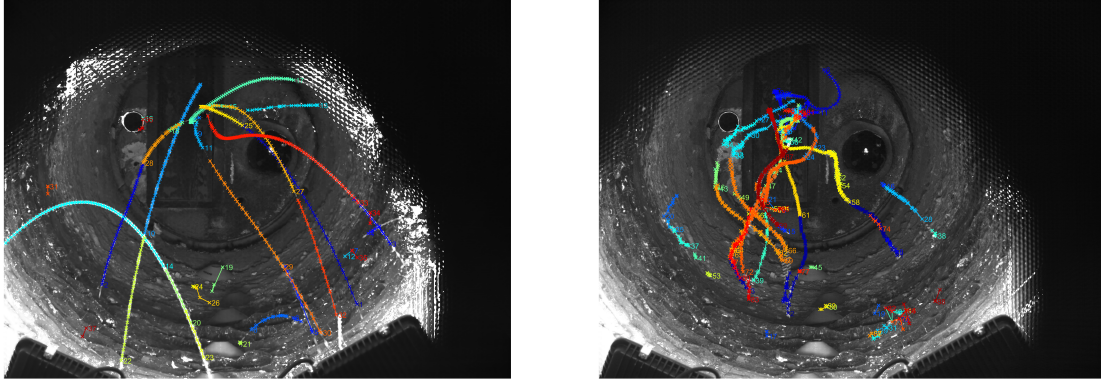
Figure 4.17 presents the temporary tracking results after the first step of the post-processing, viz., false tracklets elimination. The first step aims to delete tracklets that are too short, moving too slowly, or too irregular. As shown in the figure, the vast majority of the error tracklets meet the pre-defined conditions in Chapter 4.2.1 and are consequently removed.



**Figure 4.17.:** Temporary tracking results of wood chips (a) and paper shreds (b) after false tracklets elimination.



Figure 4.18 gives the tracking outcomes after performing the second step, deleting outliers in each tracklet. With the completion of this step, the outliers defined in Chapter 4.2.2 within the tracklets are pruned. As proved by the figure, the tracklets become smoother despite incompleteness. This step is regarded as a preparation for tracklet connection and is conducive to scanning potential tracklet connection pairs.

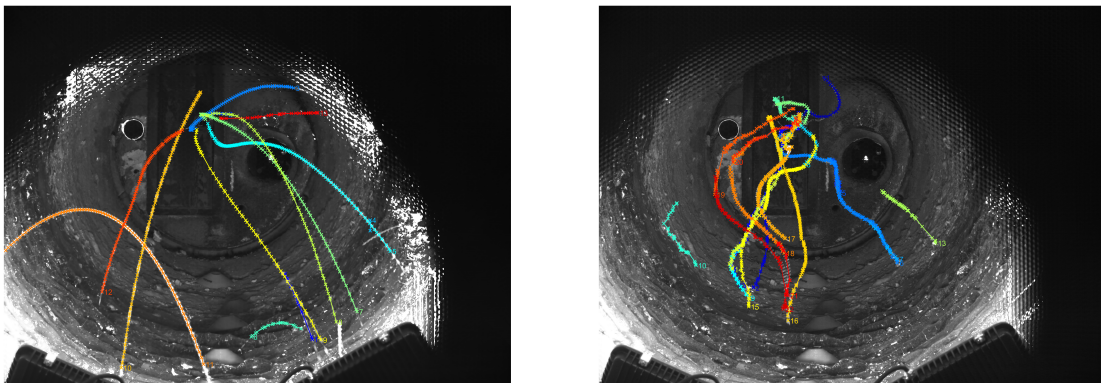


(a)

(b)

**Figure 4.18.:** Temporary tracking results of wood chips (a) and paper shreds (b) after deleting outliers in each tracklet.

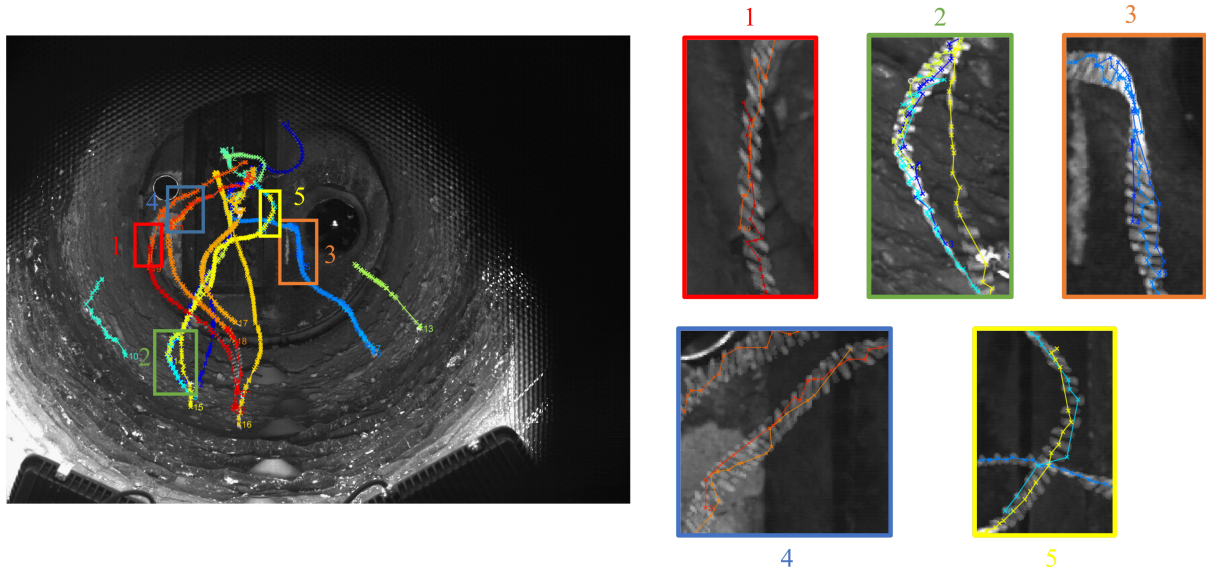
Subsequently, the post-processing framework begins to search potential tracklets matching pairs in accordance with Equation 4.21 and connect or merge them by means of the DNN approach. Figure 4.19 shows the obtained trajectories after accomplishing the third post-processing step. For fractions with regular motions, for instance, wood chips, the results could provide sufficient adequacy and accuracy to allow for further analysis of the particles' properties. Notwithstanding, for fractions with complicated motions, such as paper shreds, the acquired trajectories entail further processing. As highlighted in the five enlarged rectangles in Figure 4.20, several particle traces correspond to more than one obtained trajectory. Due to the sizeable length of paper shreds, detection duplications occur more frequently, which gives rise to more simultaneous tracklets of one particle. Therefore, the trajectories are then processed by the supplementary step of the post-processing framework.



(a)

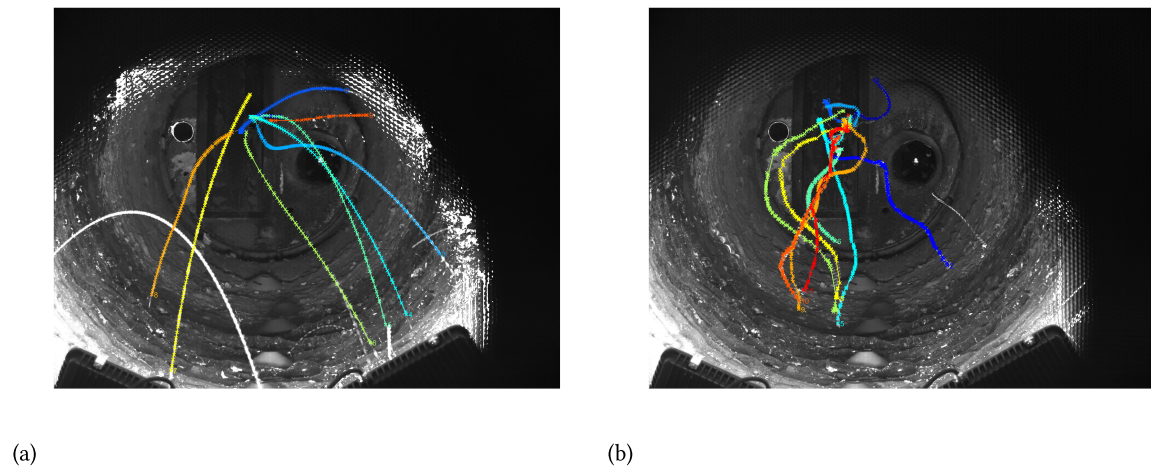
(b)

**Figure 4.19.:** Temporary tracking results of wood chips (a) and paper shreds (b) after tracklets connection and merging.



**Figure 4.20.:** Temporary tracking results of paper shreds after tracklets connection/merging and several enlarged examples of the trajectories highlighted in the rectangles.

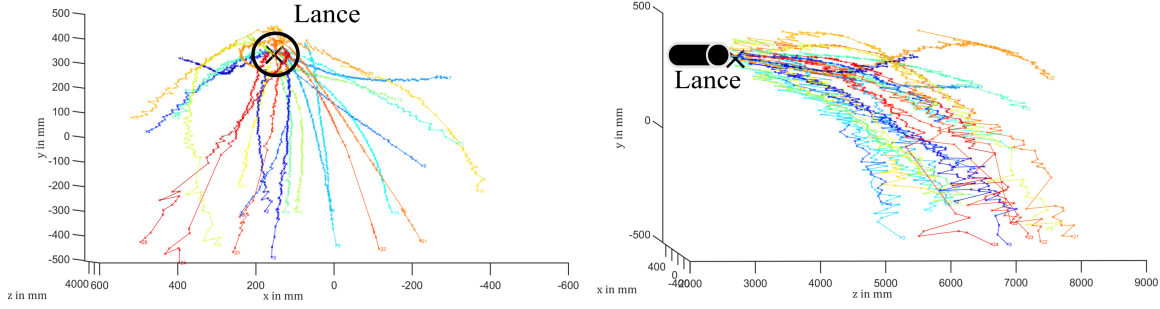
The last step of the post-processing framework is to fuse trajectories corresponding to one particle. This step serves as an additional step that is performed when necessary. For instance, particles with regular motions, like wood chips, do not entail trajectories fusion since, generally, one particle triggers only one trajectory. For paper shreds with a number of detection duplications, the last step benefits the optimization of tracking performance and lays the foundation for the subsequent analysis. Figure 4.21 provides the final tracking result of wood chips and paper shreds. Only the trajectories starting from the lance remain for further process. Compared to the initial tracking results, the trajectories after post-processing provides significant improvement.



**Figure 4.21.:** Final tracking results of wood chips (left) and paper shreds (right) after post-processing.

### 4.3. 3D Regression of trajectories

Since the particles were with faint brightness as conveyed through the lance, they were probably not captured by the plenoptic camera until several frames later. Therefore, the resulting trajectories also do not start at the lance, as shown in Figure 4.22. Moreover, because of the fluctuating information delivered by the camera, the obtained 3D trajectories are not smooth enough to provide a reliable basis for further statistical analysis of particle properties.



**Figure 4.22.:** Examples of spatial particle trajectories without regression from two views.

Thus, the resulting 3D trajectories should be regressed at first. On the one hand, the trajectories can be smoothed by the regression to facilitate the subsequent analysis. On the other hand, the particles' fly-out time can be computed, benefitting the determination of their combustion time. Given that the motion of different particles varies significantly and the movements of the same particle in different directions are also distinct, the polynomial order must be determined separately for each particle and each coordinate. For each direction, the corresponding polynomial is a function of time  $t$ . Meanwhile, since the three polynomials are independent, they are also not necessarily of the same degree. For example, in the  $x$ -direction, the displacement of the particle is smaller and more regular, and therefore the approximate polynomial could be with a lower degree. The vertical motion is more consistent with free fall. Thus, the polynomial corresponding to the  $y$ -direction is probably quadratic. In contrast, the depth displacement fluctuates significantly, leading to regression with higher order.

However, considering the precondition that all particles pass through a spatial point representing the lance's exit, *i.e.*, that each estimated polynomial is supposed to own a fixed coordinate at the exact moment, these three regressions are not entirely independent. Let the following simple instance illustrates the regression process with three quadratic polynomials,

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} p_{2x}t^2 + p_{1x}t + p_{0x} \\ p_{2y}t^2 + p_{1y}t + p_{0y} \\ p_{2z}t^2 + p_{1z}t + p_{0z} \end{bmatrix}, \quad (4.26)$$

with time variable  $t$  and coefficient vector  $[p_{2x}, p_{1x}, p_{0x}, p_{2y}, p_{1y}, p_{0y}, p_{2z}, p_{1z}, p_{0z}]$ . The expression for a trajectory passing through a particular spatial point  $[x_0, y_0, z_0]$  denoting the lance's exit at time  $t_0$  is

$$\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} = \begin{bmatrix} p_{2x}t_0^2 + p_{1x}t_0 + p_{0x} \\ p_{2y}t_0^2 + p_{1y}t_0 + p_{0y} \\ p_{2z}t_0^2 + p_{1z}t_0 + p_{0z} \end{bmatrix}. \quad (4.27)$$

By reformulating the equations,  $[p_{0x}, p_{0y}, p_{0z}]$  can be expressed as

$$\begin{bmatrix} p_{0x} \\ p_{0y} \\ p_{0z} \end{bmatrix} = \begin{bmatrix} x_0 - (p_{2x}t_0^2 + p_{1x}t_0) \\ y_0 - (p_{2y}t_0^2 + p_{1y}t_0) \\ z_0 - (p_{2z}t_0^2 + p_{1z}t_0) \end{bmatrix}. \quad (4.28)$$

Subsequently, Equation 4.28 is substituted into Equation 4.26 to acquire the following equations

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} p_{2x}t^2 + p_{1x}t + x_0 - (p_{2x}t_0^2 + p_{1x}t_0) \\ p_{2y}t^2 + p_{1y}t + y_0 - (p_{2y}t_0^2 + p_{1y}t_0) \\ p_{2z}t^2 + p_{1z}t + z_0 - (p_{2z}t_0^2 + p_{1z}t_0) \end{bmatrix}. \quad (4.29)$$

Thus, the polynomials representing the trajectory are transformed from the previous nine parameters (Equation 4.26) to seven parameters (Equation 4.29), *i.e.*,  $[p_{2x}, p_{1x}, p_{2y}, p_{1y}, p_{2z}, p_{1z}, t_0]$ . The constraint, *viz.*, that each of the three polynomials take the values  $x_0$ ,  $y_0$  and  $z_0$  respectively at the same time  $t_0$ , reduces the polynomial model by two degrees of freedom (This also applies to higher-order polynomial regressions). Moreover, by virtue of estimating polynomials,  $t_0$  as a parameter can also be computed, from which the exact time when the corresponding particles are conveyed out can be determined.

The polynomials are approximated by solving the nonlinear least squares problem utilizing the trust region approach [18] based on temporal trajectory coordinates. In order to reduce the difficulty of the regression and increase the performance accuracy, the ranges of parameters are preset beforehand. The ranges are determined by solving Equation 4.29 for a few randomly selected points on the trajectory. In addition,  $t_0$  itself has its own restrictions, *e.g.*, it must take a value less than the time when the first detection occurs. Introducing such constraints benefits reducing the computational cost and meanwhile improving the accuracy.

In computing, the different orders in each direction are combined with each other. If the highest order is set to  $n$ , the total number of combinations is  $n^3$ . That is to say, the number of combinations increases significantly for each additional order. Moreover, a higher order also retains more fluctuations in the trajectory, which is not definitively conducive to further analysis. Besides, the higher the regression order, the longer the computation time required. On the contrary, polynomials with lower orders might not be able to represent the motions appropriately. To determine the adequate polynomial order, a set of trajectories with different regression orders is investigated according to the measures described in [88], *viz.* root mean squared error (RMSE) [40], Mean absolute percentage error (MAPE) [22], R-squared  $R^2$  [16], and adjusted R-squared  $R^{*2}$ . Assuming a trajectory with  $n$  data points is fitted by  $k$  order polynomials, then the RMSE in x-direction is defined as

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (x_i - \hat{x}_i)^2}{n - (k + 1)}}, \quad (4.30)$$

where  $x_i$  denotes the detected x coordinate, and  $\hat{x}_i$  stands for the corresponding fitted value. RMSE is an average squared error and can estimate the performance of the regression. A small RMSE indicates an accurate fitting outcome.

MAPE is another estimation criterion that compares the relative performance of forecasts. With a value of less than 10%, an estimation is interpreted as excellent. Basically, a regression with MAPE smaller than 50% would be acceptable. MAPE is defined as follows,

$$\text{MAPE} = \frac{100}{n} \sum_{i=1}^n \left| \frac{x_i - \hat{x}_i}{x_i} \right|. \quad (4.31)$$

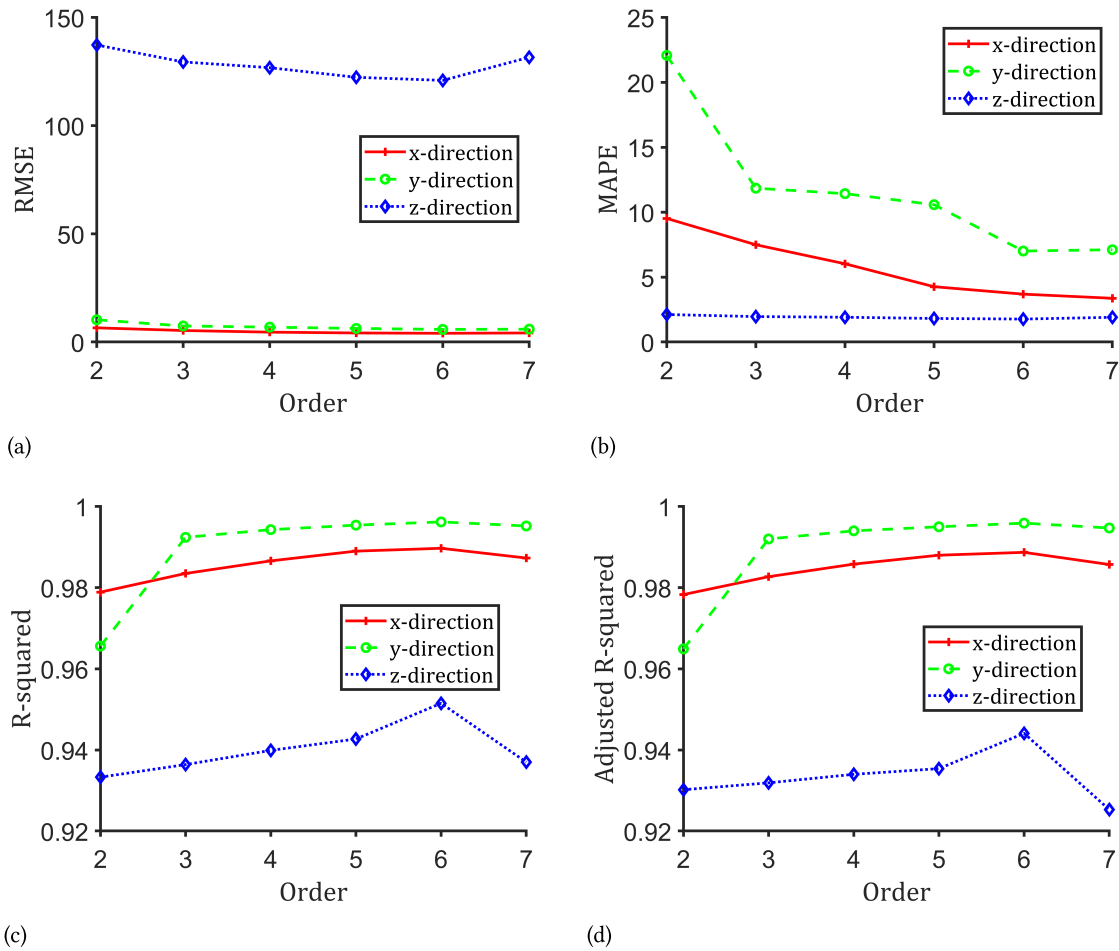
The R-squared is defined as

$$R^2 = 1 - \frac{\sum_{i=1}^n (x_i - \hat{x}_i)^2}{\sum_{i=1}^n (x_i - \bar{x})^2}, \quad (4.32)$$

where  $\bar{x}$  is the arithmetic average coordinate. Theoretically, the R-squared is a normalized measurement with a value from the range [0,1]. The value of  $R^2$  provides statements concerning regression accuracy. The larger the value, the more accurately the estimation fits the actual. Additionally, the adjusted R-squared  $R^{*2}$  is computed by

$$R^{*2} = R^2 - \frac{(1 - R^2)k}{n - (k + 1)}. \quad (4.33)$$

Obviously,  $R^{*2}$  adjusts for the number of variables included in the regression equation and is always more minor as  $R^2$ . If  $R^{*2}$  is much lower than  $R^2$ , the sample trajectory might be over-fitted and of limited generalization [88].



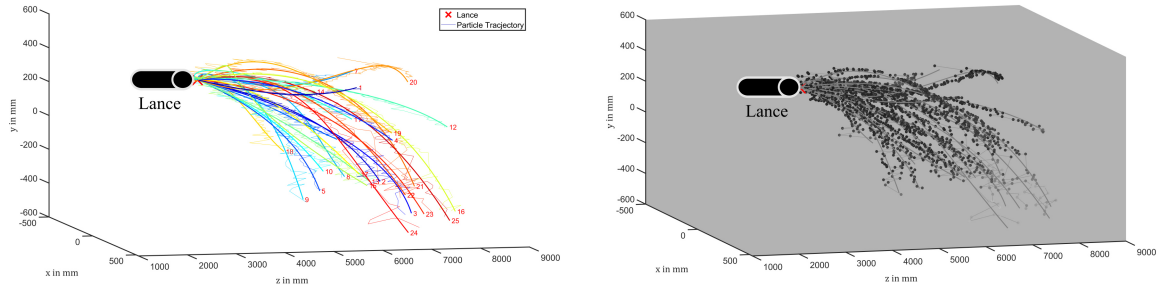
**Figure 4.23.:** Measures of the polynomial estimated trajectories with different orders (a) RMSE (b) MAPE. (c) R-squared. (d) Adjusted R-squared.

Figure 4.23 figuratively shows the values of the measures for the selected trajectories fitted with polynomials of different orders. Generally, the accuracy of the regression improves as the order increases. However, this improvement becomes ignorable as the order increases and even slightly decreases with the order higher than 6. Since the fluctuation in the depth direction is intense, RMSE in the z-direction is significantly larger than in the other two directions. Similarly, the  $R^2$  and  $R^{*2}$  values



in the z-direction are smaller than those in the other two directions. Still, they are all higher than 0.9, indicating the satisfied adequacy of the regression model. On the contrary, MAPE in the z-direction is the lowest as a consequence of the relatively large depth values. Based on the information delivered by the figure, the computational range of the directional regression order is determined from 2 to 6 to ensure a certain adequacy and efficiency. The regression process is stopped if the optimization of the measures is less than 1% for each order of enhancement.

An example of estimated spatial trajectories is given in Figure 4.24 left. In addition, the estimated polynomials with corresponding average particle gray values are shown in Figure 4.24 right, which enables a statistical investigation into the particles' combustion properties. The estimated polynomials are evaluated according to the measures mentioned above.



**Figure 4.24.:** Spatial particle trajectories with polynomial regression. Left: Estimated polynomials. Red Cross stands for the lance, and thick solid lines are the estimated polynomials. Right: spatial trajectories with corresponding particle gray values. Each point represent the corresponding gray values of the detections.



## 5. Results and discussion

This chapter summarizes the findings and contributions made utilizing the beforehand presented approaches. At first, the results of multiple particle detection and classification are illustrated and evaluated in accordance with the confusion matrix (precision, recall, and  $F_1$ -score) [53]. In addition, a ground truth dataset of particle detection is introduced, which can also function as a benchmark dataset to assess detection and clustering algorithms, as demonstrated in [115]. Subsequently, the obtained 2D particle trajectories with and without the developed novel post-processing are shown. The trajectories are further evaluated with respect to several measures recommended in [14]. Similarly, a set of 2D ground truth trajectories is also shown in the chapter. The trajectories are then converted into spatial trajectories and estimated by polynomials afterward. Finally, the flight and combustion behaviors of RDF particles, primarily consisting of dwell time, velocity, and ignition time, are analyzed and discussed. The measurements were conducted both in the rotary kiln and the combustion chamber. Notwithstanding, the information accumulated from the measurements in the combustion chamber fails to provide satisfactory outcomes since the fuel particles only fall a short distance before attaching to the inner wall, which is not able to deliver statistical statements concerning the particles' properties. Therefore, only the results obtained in the rotary kiln are presented in this chapter.

### 5.1. Results of multiple particle detection and classification

In the following, the results of multiple particle detection and classification are presented. Above all, the benchmark dataset serving as the ground truth for the assessment is introduced. Based on the ground truth, the performance of the background subtraction algorithms presented in Chapter 2 together with particle detection and clustering approaches can be quantitatively evaluated. The benchmark also provides statements concerning particle positions relative to the kiln's inner wall. Therefore, the performance of 3D particle classification using approximated cylinder model can also be assessed objectively.

#### 5.1.1. Ground truth dataset

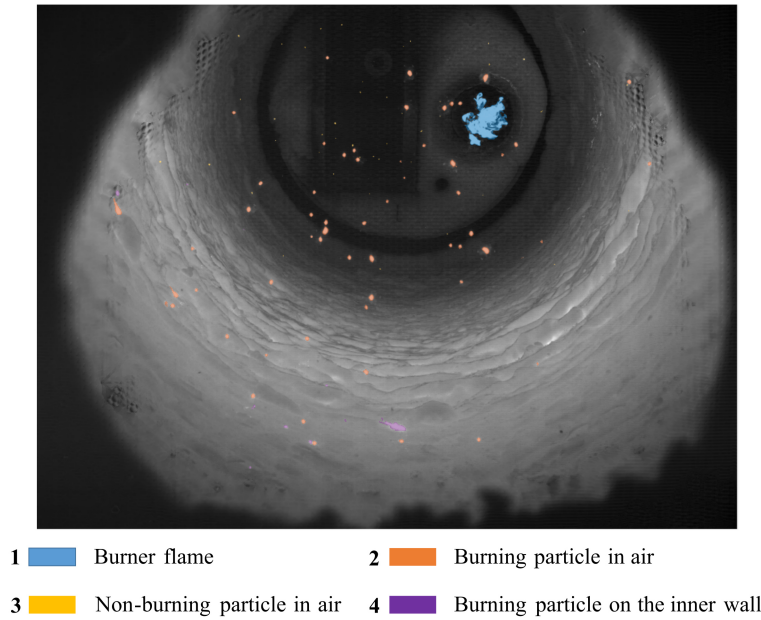
For the acquisition of a ground truth dataset to quantitatively evaluate the detection performance, particles from 50 frames were manually identified and labeled. The 50 images were selected from five different sequences with ten images each. As mentioned before, all RDF fractions are conveyed together into the lance to ensure an objective statement of the detection performance. The burning and non-burning particles were not evenly distributed across the five sequences, with several sequences containing more burned particles, some dominated by non-burning particles, and others with burning and non-burning particles roughly equal in total number. The particles were classified into three categories denoted by different digits 2,3, and 4, as presented in Table 5.1. The digits 0 and 1 indicate the rotary kiln's inner wall and the burner flame, respectively. The class representing non-burning particle on the inner wall does not exist since the high inner wall temperature (approx. 1200 °C) ignite all landing particles. The particles were labeled pixel-by-pixel, *i.e.*, each pixel of a particle was labeled,



to ensure an accurate assessment. As long as one detection falls within the pixel range of a particle, the detection can be associated with the particle. Figure 5.1 gives an example of labeled ground truth with the mentioned classes.

**Table 5.1.:** Classes with their corresponding labels in ground truth.

Class	Label
Rotary kiln inner wall	0
Burner flame	1
Burning particle in air	2
Non-burning particle in air	3
Burning particle on the inner wall	4



**Figure 5.1.:** Example of the labeled ground truth image. The particles and flame are labeled pixel-by-pixel.

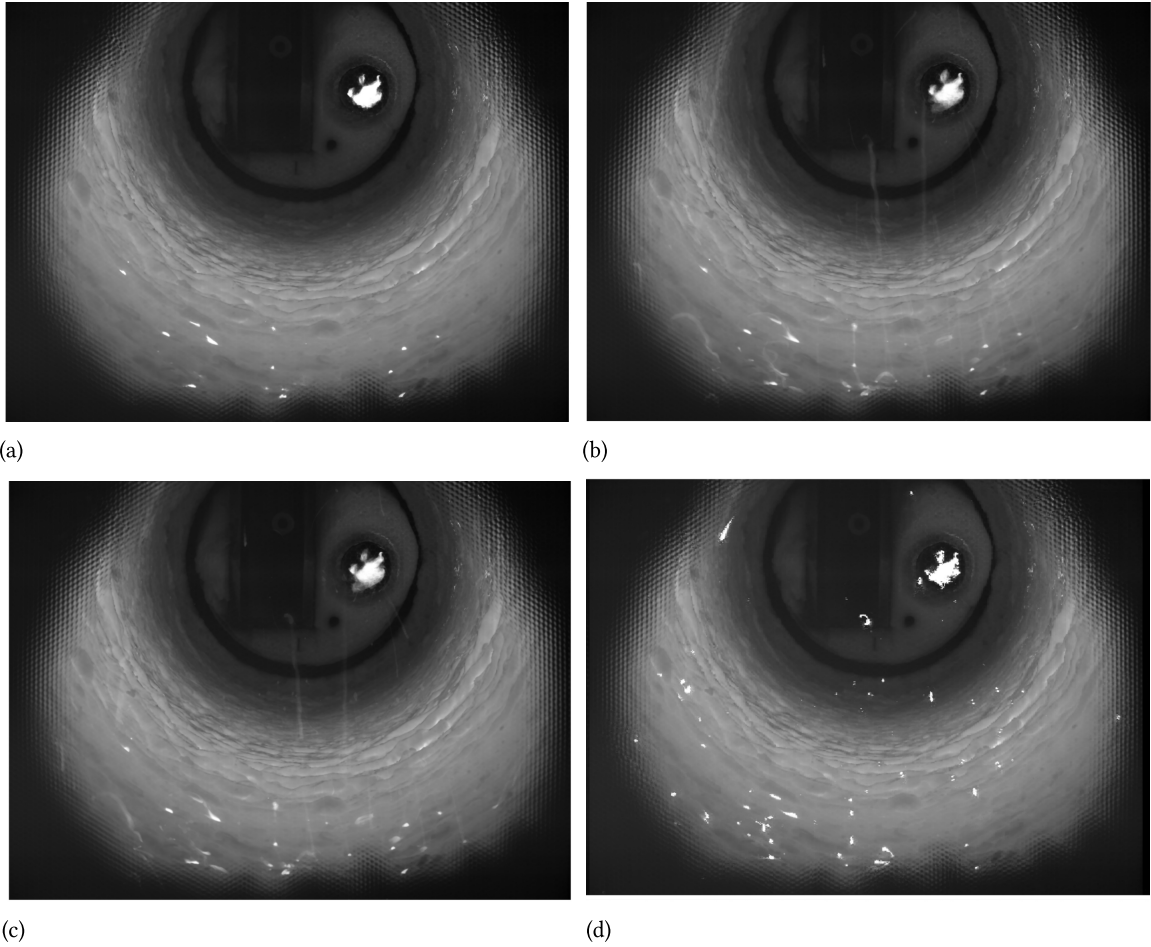
Additionally, another label concerning the particle's 3D information is also listed. If the 3D information of a ground truth particle is available, the list is filled with 1; otherwise, 0. Overall, 5701 ground truth particles are identified, with 3920 particles in air and 1781 on the inner wall, as illustrated in Table 5.2. Because the burning particles are larger and brighter, which benefits the capturing of the plenoptic camera, almost all burning particles own complete or partial 3D information. In contrast, non-burning particles are relatively smaller and darker. Therefore, only less than 80% of such particles can be captured by the camera with 3D information. In total, the plenoptic camera is able to capture 92% of the particles spatially. Table 5.2 also details the distribution of the various classes of RDF particles in the five subsets. V1, V2, and V4 contain significantly more burning particles in air, while in V3, the number of burning particles in air is approximately equal to the number of non-burning particles. In V5, non-burning particles dominate. Since the differences in these five subsets lead to distinct detection performances, the detection results will also be discussed separately for the five subsets. The ground truth dataset is available in [116], based on which the detection and classification performance can be evaluated.

**Table 5.2.:** Distribution of various classes of RDF particles in the five ground truth subsets (V1-V5).

Class	V1	V2	V3	V4	V5	Amount	With 3D info	Percentage with 3D info
Burning particle in air	600	427	161	1150	116	2454	2439	99.39%
non-burning particle in air	335	143	145	167	676	1466	1155	78.79%
Burning particle on the inner wall	248	469	313	631	120	1781	1658	93.09%
Particle in air	935	570	306	1317	792	3920	3594	91.68%
Total	1183	1039	619	1948	912	5701	5252	92.12%

### 5.1.2. Results of 2D background subtraction

As described in Chapter 2, background subtraction is often used as a pre-procedure step for object detection to highlight the foreground. Removing the background by computing a background model can effectively reduce the negative impact of background brightness on the detection performance. In the following, the six background models listed in Chapter 2.2.1, *i.e.*, static frame difference (mean, median), moving frame difference (2-frame, 3-frame), adaptive background learning, running Gaussian average, are detailed and compared. Firstly, the resulting background images of four background models are presented in Figure 5.2. Since the moving frame difference approaches consider simply the preceding and following images as backgrounds, they are not shown here.



**Figure 5.2.:** Various backgroundmodels. (a) Temporal median background model. (b) Temporal mean background model. (c) Adaptive background model with  $\alpha = 0.05$  (d) Running Gaussian average with  $\alpha = 0.99, k = 0.5$ .

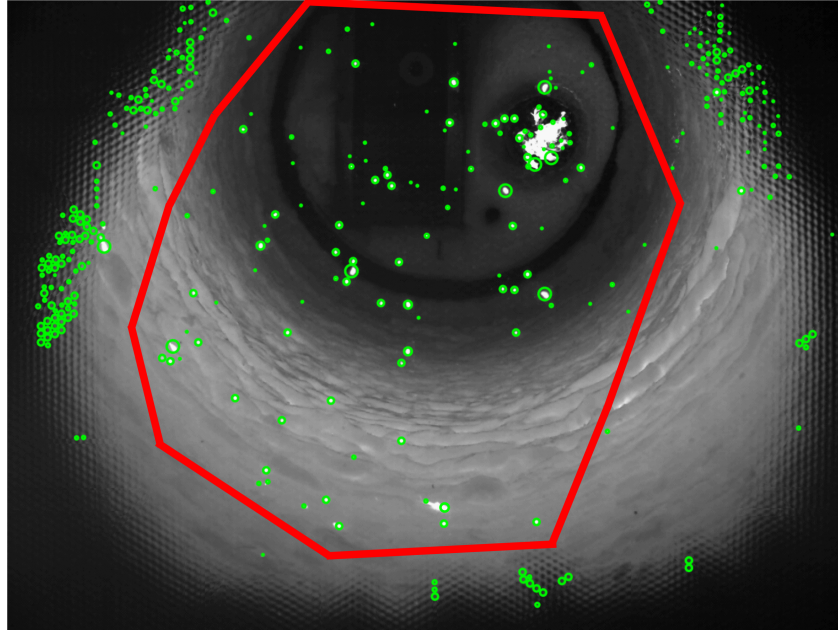
The methods of computing the temporal median and mean gray values as background models belong to static frame difference because the resulting background models remain constant over a specific time period. Figures 5.2 (a) and (b) show examples of background models derived from the temporal median and mean methods, respectively. As highlighted by the figure, the temporal mean background model contains several particle trajectories as a consequence of local high gray values enhancing the average significantly. Whereas the median background model considers the temporal median instead of the mean gray value, the transitory local extrema exert barely influence on the obtained model. The adaptive background learning dynamically updates the background model by weighting the current image and the previous background model. Since the initial background model corresponds to a mean background, the acquired temporal model also contains several slight particle traces. Additionally, the learning rate  $\alpha$  determines the importance of the current image. A high learning rate will aggravate the impact of local gray value extrema. Compared to the other background models, the background model computed by the running Gaussian average algorithm retains the majority of the particles adhering to the wall, which could benefit the further detection process. Nevertheless, the background fails to completely eliminate relatively large particles, as shown in Figure 5.2 (d). Because the first image considerably influences the background, a slow-moving particle appearing in the first frame is regarded as part of the background according to the Equation 2.10.

In order to achieve a quantitative and accurate comparison of various background models' effects on particle detection, these backgrounds are subtracted from the 50 images in the ground truth dataset. Based on the resulting foregrounds, the SIFT algorithm conducts the detection process, whose outcomes are subsequently assessed by the evaluation method introduced in Chapter 3.3. Figure 5.3 presents an example of the detection result using SIFT algorithm and median background subtraction. The center of each green circle corresponds to the position of a detected particle, whose sizes are pointed by the circles' radius. Apparently, substantial numbers of FP detections occur at the edge due to artifacts caused by the camera. Hence, a region of interest (ROI) that includes as many particles as possible and meanwhile excludes the artifacts region ought to be introduced to address the issue. The ROI is a red polygon, which is shown in Figure 5.3 as well. Table 5.3 gives the distribution of the various classes of particles within the ROI in each subset. The last column of the table presents the percentage of corresponding particles remaining in the ROI. Overall, almost 90% of the in air particles (burning or non-burning) persist in the region. By contrast, around half of the particles attached to the inner wall are excluded from the ROI. Since particles on walls are of less significance for subsequent particle tracking, excluding a large number of such particles does not actually impact the entire work negatively. This ROI also applies to all subsequent particle detections.

**Table 5.3.:** Distribution of various classes of RDF particles within the defined ROI in the five ground truth subsets.

Class	V1	V2	V3	V4	V5	Amount	Percentage
Burning particle in air	554	307	119	1066	109	2155	87.82%
Non-burning particle in air	287	118	140	158	629	1332	90.86%
Burning particle on the inner wall	179	188	123	362	48	900	50.53%
Particle in air	841	425	259	1224	738	3487	88.95%
Total	1020	613	382	1586	786	4387	76.95%

As described in the previous chapter, each background subtraction method has particular adjustable parameters, which somewhat affect the results. Table 5.4 lists the influencing parameters for the various approaches. Moreover, the optimal values of these parameters offering the highest  $F_1$ -scores of the methods are also presented at the end of each corresponding line. The only adjustable parameter



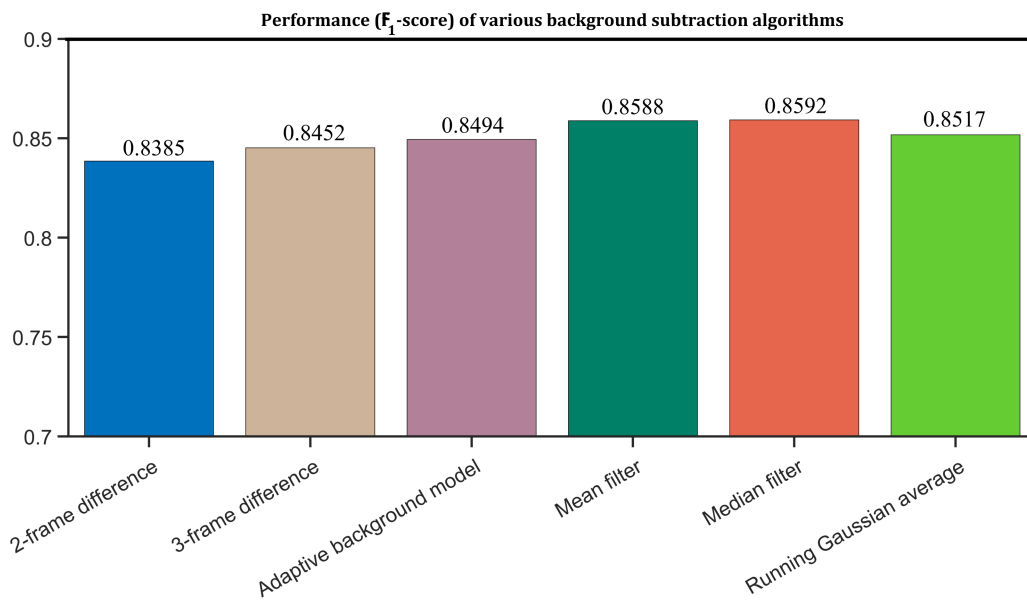
**Figure 5.3.:** Detection result of 2D-SIFT after background subtraction and the defined ROI. The center of each green circle corresponds to the position of the detected particle, whose sizes are pointed by the circles' radius. The red polygon is the defined ROI.

**Table 5.4.:** The adjustable parameters of different background methods and their optimal values for the current work.

Method	Adjustable parameters	Optimal value
2-frame difference	Computing interval	3
3-frame difference	Computing interval	3
Adaptive background learning	Number of frames	50
	Sequence	25 previous frames and 25 subsequent frames
	Learning rate $\alpha$	0.05
Mean filter	Number of frames	50
	Sequence	25 previous frames and 25 subsequent frames
Median filter	Number of frames	150
	Sequence	75 previous frames and 75 subsequent frames
Running Gaussian average	Number of frames	10
	Learning rate $\alpha$	0.99
	Coefficient $k$	0.4

for the moving frame difference is the computing interval, *viz.*,  $i$  in Equation 2.5 and 2.6. The static background models require attention to the amounts of computing images and their sequences, while for the adaptive background learning and the running Gaussian average algorithms, other parameters should also be taken into consideration, such as  $\alpha$  and  $k$ . The values of these parameters were studied in detail in a supervised master's thesis [37]. And the values listed in the table are the ones that provide the best results within the study.

In addition, the performance of the SIFT algorithm utilized for particle detection relies considerably on an essential parameter, *i.e.*, the local threshold of DoG, which can significantly affect the detection of feature points and therefore entails particular consideration. In general, the parameter ought to be selected by statistical techniques, for instance, cross-validation, which was also researched in the master thesis. The study reveals a slight performance difference between applying cross-validation to determine the SIFT parameter and using a fixed empirical SIFT parameter for various background subtraction approaches (with a  $F_1$ -score difference of the order of approx. 0.001). Since background subtraction is only a pre-processing for 2D particle detection and the cross-correlation entails substantial computational cost, the SIFT parameter used in the research to compare different background algorithms is chosen in accordance with the empirical value of 500. Figure 5.4 shows the detection results ( $F_1$ -score) achieved using various background subtraction methods combined with SIFT detection. The  $F_1$ -scores presented in Figure 5.4 refer to all particles (classes 2, 3, and 4) together.



**Figure 5.4.:** Performance ( $F_1$ -score) of various background subtraction algorithms with the corresponding parameter values listed in Table 5.4. The  $F_1$ -scores are computed by 2D-SIFT algorithm with local maximum of DoG=500.

In general, the six tested background subtraction methods provide comparable  $F_1$ -scores, with the highest value delivered by the median filter at 0.8592 and the lowest value falling on the 2-frame difference method at 0.8385. The mean filter performs approximately as well as the median filter in spite of remaining particle traces on the background model. The adaptive background learning algorithm, which also has trajectory residuals, is inferior. However, the score is only less than 0.01 below the highest score. In summary, the static background methods outperform the other approaches in this research, while the frame difference methods perform the worst. Therefore, the best-performing median filter will be selected as the background subtraction algorithm in the subsequent 2D particle detection.

### 5.1.3. Results of particle detection

The particle detection approaches mentioned beforehand, including 2D detection methods, 3D clustering algorithms, and their corresponding combination, are applied to the ground truth data to identify potential particles. The obtained detections are quantitatively evaluated following the criteria described

in Chapter 3.3. Thereby, the mean filter background model is applied to all 2D particle detection approaches. Besides, the evaluation proceeds only within the 2D ROI presented in Figure 5.3, *i.e.*, particles outside the predefined region in ground truth datasets, and particle detections from 2D and 3D methods are ignored.

## 2D particle detection

The performance of two-dimensional particle detection algorithms is highly dependent on the local gray values. A number of influencing factors, such as illumination changes, object rotation, and burning flames, can greatly affect detection performances. Generally, 2D particle detection algorithms contain parameters to adapt to different brightness situations, except for several methods that can automatically adjust parameters, *e.g.*, the OTSU threshold selection method. These parameters essentially dominate the detection effect and must therefore be chosen discreetly. In this study, the parameter for 2D particle detection (the local maximum of DoG in SIFT) is determined based on cross-validation [49] by maximizing the corresponding  $F_1$ -score. Cross-validation is a model validation technique to assess the effect of generalizing statistical analysis results to an independent data set. By using different parts of the data to test and train a model, the accuracy of a predictive model performed in practice can be estimated. In the applications, a model is usually given a known dataset to train (training dataset) and an unknown dataset to test the model (test set). As mentioned in Chapter 5.1.1, the ground truth dataset consists of images equally from five separate sequences (subsets). Each time, one subset is selected as the test set, and the other four are considered the training set. The parameter values that deliver the best results in the training set are directly utilized in the test set for objective evaluation of the performed algorithm.

### 2D-OTSU particle detection

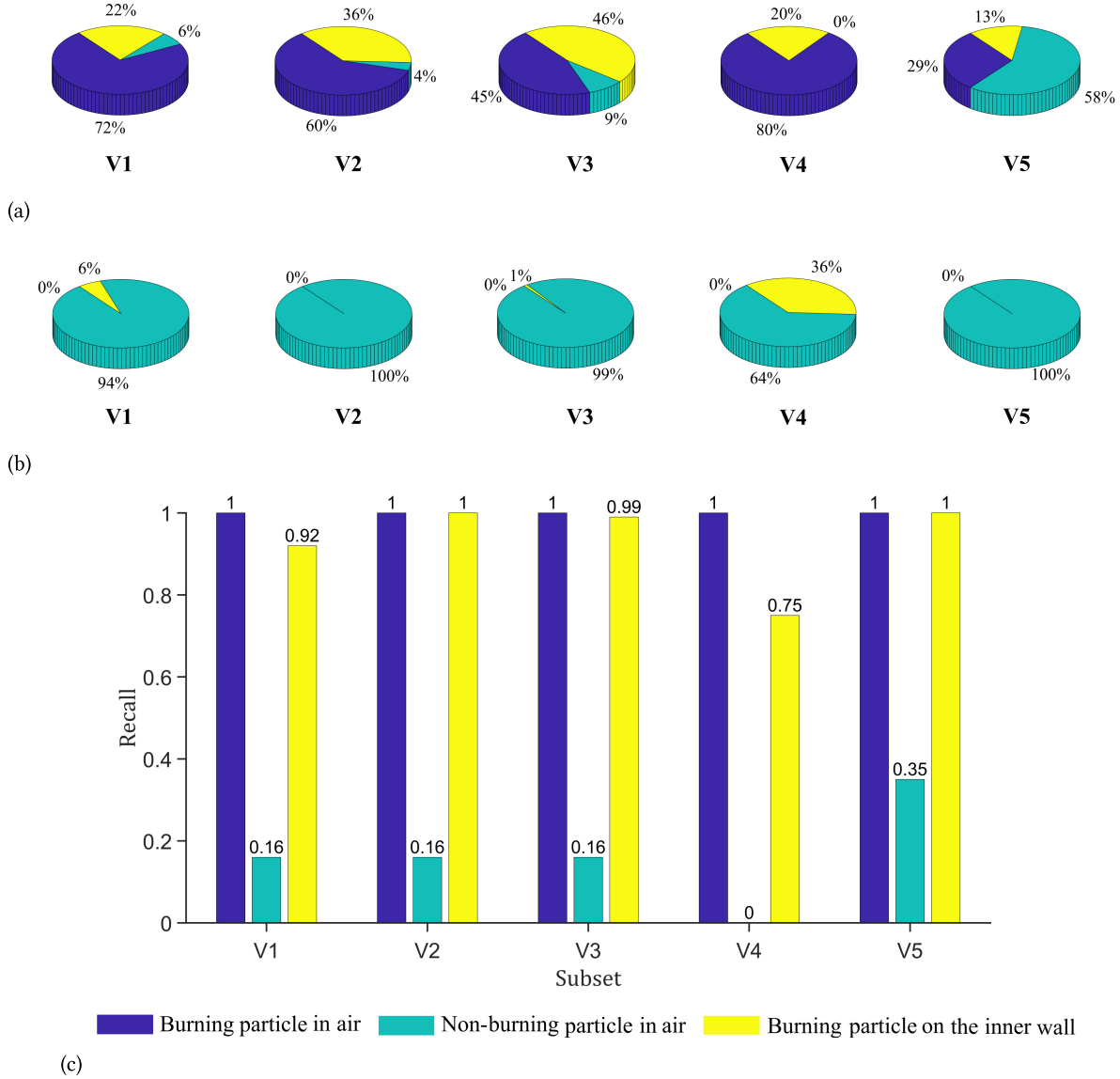
The OTSU threshold selection method is autonomous in selecting thresholds for particle identification, and, therefore, no parameters require to be defined. Table 5.5 presents the results of particle detection by means of the OTSU algorithm for the five ground truth subsets. For V1, V2, and V4, where burning particles are in the majority, the OTSU algorithm has the ability to reach  $F_1$ -scores beyond 0.85. Whereas for V3 and V5, which contain a lot of non-burning particles, the  $F_1$ -scores are relatively low. In particular, for V5, the score is only 0.6. Overall, the OTSU approach provides high precision scores above 0.9, and except for V2, the others are even close to 1. Thus, the  $F_1$ -scores are basically dominated by the recall.

**Table 5.5.:** Performance of 2D particle detection using the OTSU approach for the five ground truth subsets.

Subset	TP	FP	FN	Precision	Recall	$F_1$ -score
V1	766	9	254	0.9884	0.7510	0.8535
V2	514	44	99	0.9211	0.8385	0.8779
V3	264	5	118	0.9814	0.6911	0.8111
V4	1338	15	248	0.9889	0.8436	0.9105
V5	374	3	412	0.9920	0.4758	0.6432
Total	3256	76	1131	0.9772	0.7422	0.8436

Figure 5.5 details the distribution of various classes of particles detected in each subset, including TP, FN, and recall values. Since only the ground truth particles are classified, only the assigned ground truth particles (TP) and the unassigned ground truth particles (FN) can be counted. The unassigned

detections (FP) can not be accurately classified. Consequently, for each particle class, only the recall values, *i.e.*, how many ground truth particles are correctly detected, are computed. As illustrated in the figure, the vast majority of burning particles can be detected, and, therefore, they also constitute the majority of TP detections in several cases depending on the particle distribution of the ground truth. On the contrary, non-burning particles are less likely to be detected due to their smaller size as well as



**Figure 5.5.:** Detection performance of various classes of RDF particles by means of the OTSU threshold selection approach. Blue stands for burning particle in air; cyan represents non-burning particle in air; yellow denotes burning particle on the inner wall. The color notation also applies to all following similar image. (a) Distribution of TP particle detections of the ground truth subsets. (b) Distribution of FN particle detections of the ground truth subsets. (c) Recall values of various particle classes in the ground truth subsets.

their lower brightness. The undetected ground truth particles in V2, V3, and V5 are almost all non-burning particles. V4 is unique since the majority of the particles (over 90%) are burning particles, which leads to high threshold values and consequently missing the whole non-burning particles. Apparently,



no burning particles in air exist in unassigned ground truth particles (FN), *i.e.*, all burning particles in air are detected. This is also reflected in Figure 5.5. For the entire five subsets, the recall values of burning particles in air reach equal to 1. Meanwhile, the recall value of the burning particles on the inner wall is maintained at roughly around 0.9. In contrast, the detection performance of non-burning particles is inferior. V1, V2, and V3 have a recall of 0.16, while V4 owns a recall of 0 as a consequence of the overly bright particles raising the threshold. In V5, the predominance of non-burning particles results in a lower threshold value that benefits the detection of dark non-burning particles. Nevertheless, V5 reaches a recall value of 0.35, much lower than the burning particles.

To conclude, the OTSU threshold selection method can accurately detect the burning particles in the current study and provide satisfactory results. In contrast, it fails to prove sufficient adequacy in detecting non-burning particles, especially when burning particles and non-burning particles are mixed. In addition, given the simplicity of this algorithm, its computation time is also short, about 1.77 s for one image with MATLAB.

#### 2D-SIFT particle detection

Unlike the OTSU threshold selection method, the SIFT algorithm ought to choose a crucial parameter, which is determined according to the cross-validation approach described before. The determined parameters, as well as the detection results of the subsets, are separately given in Table 5.6. Essentially, both SIFT and OTSU algorithms detect particles based on their brightness compared to local gray values (OTSU uses global threshold, while SIFT considers local gray value differences), so the differences existing in the detection results are not excessive. For subsets containing more burning particles, such as V1 and V4, the detection performance of SIFT is superior, with  $F_1$ -scores reaching above 0.9. For V5, which is dominated by non-burning particles, the  $F_1$ -score provided by the SIFT algorithm remains the lowest at 0.78. However, it is still 0.13 higher than that delivered by the OTSU method. Overall, the performance of the SIFT algorithm is better compared to the OTSU method. The recall values notably reflect the accuracy and adequacy of the SIFT algorithm in comparison to the OTSU method, *viz.*, the SIFT algorithm is capable of detecting more particles in the ground truth. Moreover, the precision provided by the SIFT algorithm is also comparable to that of OTSU. The only exception is V2, whose detection performance is slightly inferior to the OTSU approach due to the poorer precision caused by the selected parameters by cross-validation. The lower parameter leads to more FP detections of the SIFT algorithm, which is disadvantageous to precision and  $F_1$ -score.

**Table 5.6.:** Performance of 2D particle detection using the SIFT approach for the five ground truth subsets.

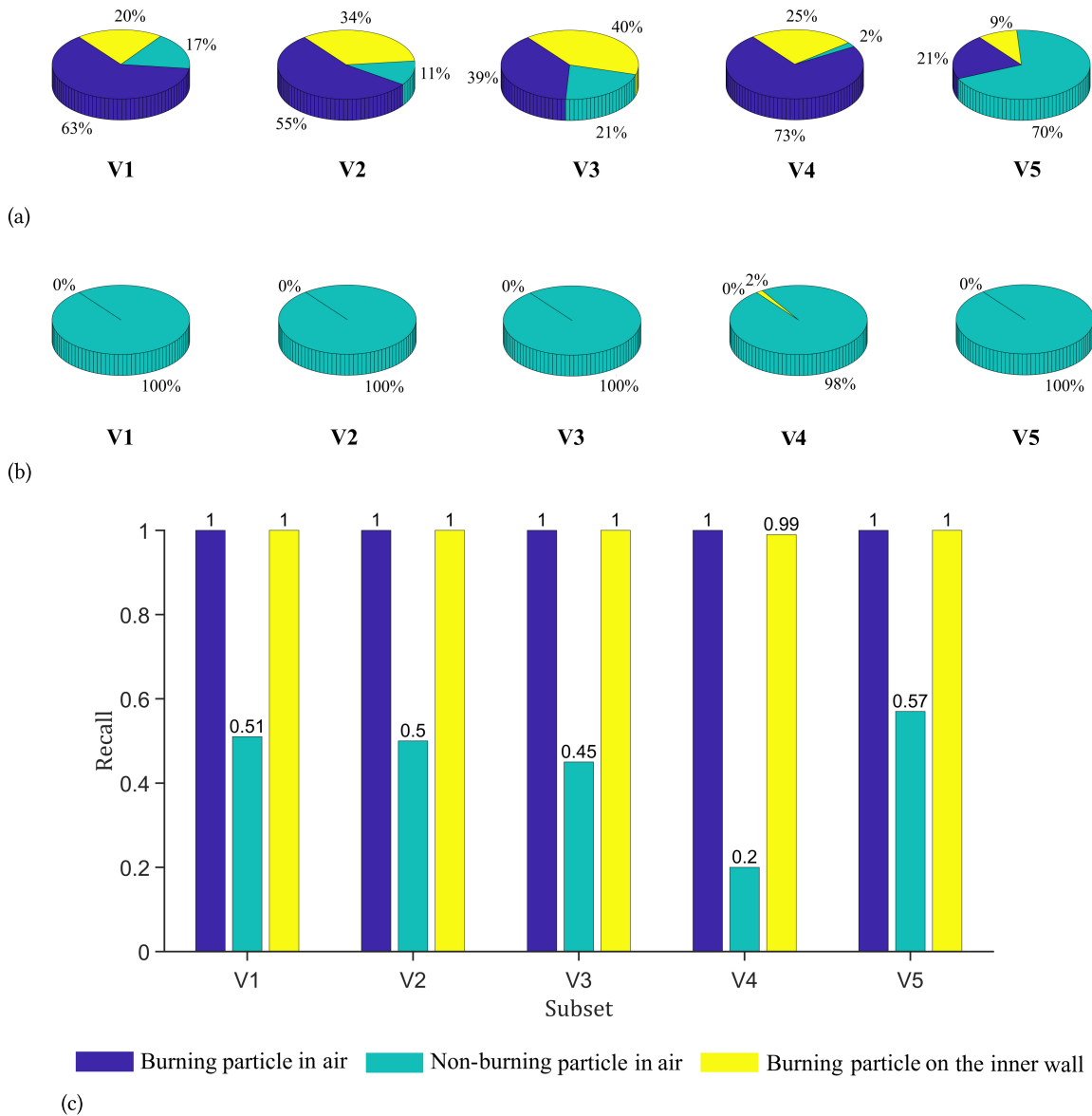
Test set	The local maximum of DoG	TP	FP	FN	Precision	Recall	$F_1$ -score
V1	550	874	21	146	0.9765	0.8569	0.9128
V2	450	548	179	65	0.7538	0.8940	0.8179
V3	550	301	28	81	0.9149	0.7880	0.8467
V4	450	1449	111	137	0.9288	0.9136	0.9212
V5	550	515	15	271	0.9717	0.6552	0.7827
Total	3687	354	700	271	0.9124	0.8404	0.8749

Figure 5.6 illustrates the detection performance of the different classes of RDF particles. Similar as the OTSU method, burning particles are the backbone of the TP detections, while the undetected ground truth particles FN are basically composed of non-burning particles. Figure 5.6 (c) presents the recall values for the different classes of particles quantitatively. In total, SIFT can detect almost all burning particles, *i.e.*, the recall values corresponding to burning particles (in air or on the inner wall) are equal



to or approximately equal to 1. Although the detection performance for non-burning particles is the most inferior, around 0.5 on average, it is still much more optimized than the OTSU method. V4 still owns the lowest recall value of non-burning particles, but it is much improved compared to the previous 0 by OTSU.

In summary, the SIFT approach performs better in detecting non-burning particles in the current study compared to the OTSU threshold selection method. Furthermore, the SIFT algorithm also presents a slight advantage in detecting burning particles on the inner wall. With respect to burning particles in air, the SIFT algorithm provides identical performance as the previous OTSU method, *viz.*, all burning particles in air can be detected. On the condition that burning particles and non-burning particles are mixed, the performance of detecting non-burning particles is relatively inferior.



**Figure 5.6.:** Detection performance of various classes of RDF particles by means of the SIFT approach. (a) Distribution of TP particle detections of the ground truth subsets. (b) Distribution of FN particle detections of the ground truth subsets. (c) Recall values of various particle classes in the ground truth subsets.

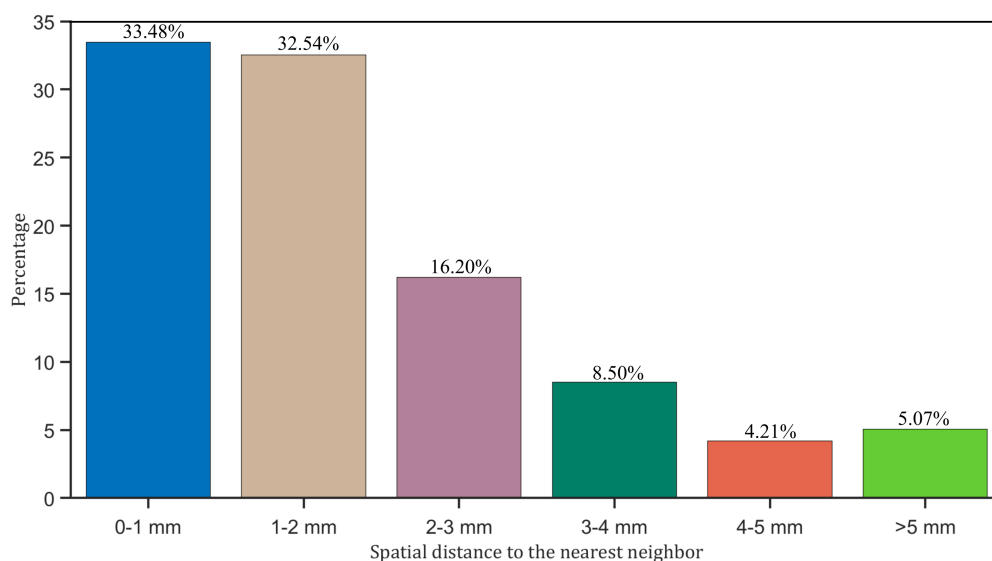
Table 5.7 shows the computation time of one image utilizing 2D-SIFT programmed with MATLAB for different parameter values (the local maximum of DoG). As illustrated by the table, the computation time for one image in this study is about 5-7 s. As the local maximum of DoG increases, the conditions of the detected particles become higher *viz.*, particles entail higher local gray value differences compared to the background in order to enhance the detection probability, leading to a reduced number of particle detections. Therefore, the computation time decreases.

**Table 5.7.:** Computation time of SIFT with various local maximum of DoG values using MATLAB.

the local maximum of DoG	400	500	600	700	800	900	1000
Computation time (s)	6.609	6.593	6.518	5.470	5.412	5.336	5.403

### 3D clustering

Different from the previous 2D grayscale value based detection method, where the gray value varies significantly from particle to particle, and even the brightness of the identical particle may change considerably temporally, the 3D clustering approaches, on the other hand, use the distribution of 3D points and their distance for building different clusters to achieve particle detection. Since the distance between two adjacent points representing the same particle is fixed at 1 pixel in image coordinates, the spatial distance varies slightly after the camera transformation, as schematically illustrated in Figure 5.7. Five images from V1 to V5 respectively are transformed into 3D point clouds, and the Euclidean distances of the points from their nearest neighbors are presented in the figure. The total number of points for V1 to V5 is about one million, and about one-third of all points have a neighbor point within 1 mm. Another one-third of the points are between 1 and 2 mm from the nearest neighbor. More than 90 percent of the points are less than 4 mm away from their nearest neighbors. Compared to the grayscale values, which can vary by thousands (uint16), the distance variation of the 3D point cloud is minimal. Under this circumstance, the clustering method does not require specific consideration for different data. Therefore, the relevant parameters of each 3D clustering method are only selected in a range according to the recommended or empirical values.



**Figure 5.7.:** Distance between points from five distinct point clouds and their nearest neighbors.

### 3D-DBSCAN clustering

When using the 3D-DBSCAN clustering approach for constituting clusters and thus for the purpose of particle detection, two parameters need to be determined, *viz.*, the minimum number of neighboring points required to become a core point *minPts* and the corresponding radius  $\epsilon$  (Chapter 2.2.4). Obviously, these two parameters are not entirely independent. As a rule, the more neighbors required, the larger the radius will follow accordingly. Sander *et al.* [101] recommended setting *minPts* to twice the dimensional size, *i.e.*, for the 3-dimensional data in this study, *minPts* should be set to  $2 \cdot 3 = 6$ . The radius can then be adjusted according to the sorted n-distance graph, as exemplified in Figure 2.19. Table 5.8 concludes the detection results of all images from the ground truth with *minPts* = 6 corresponding to three different radii  $\epsilon = 10, 30$ , and 50 mm. Moreover, the *minPts* is also grown to 10 and 15, corresponding to the different radii.

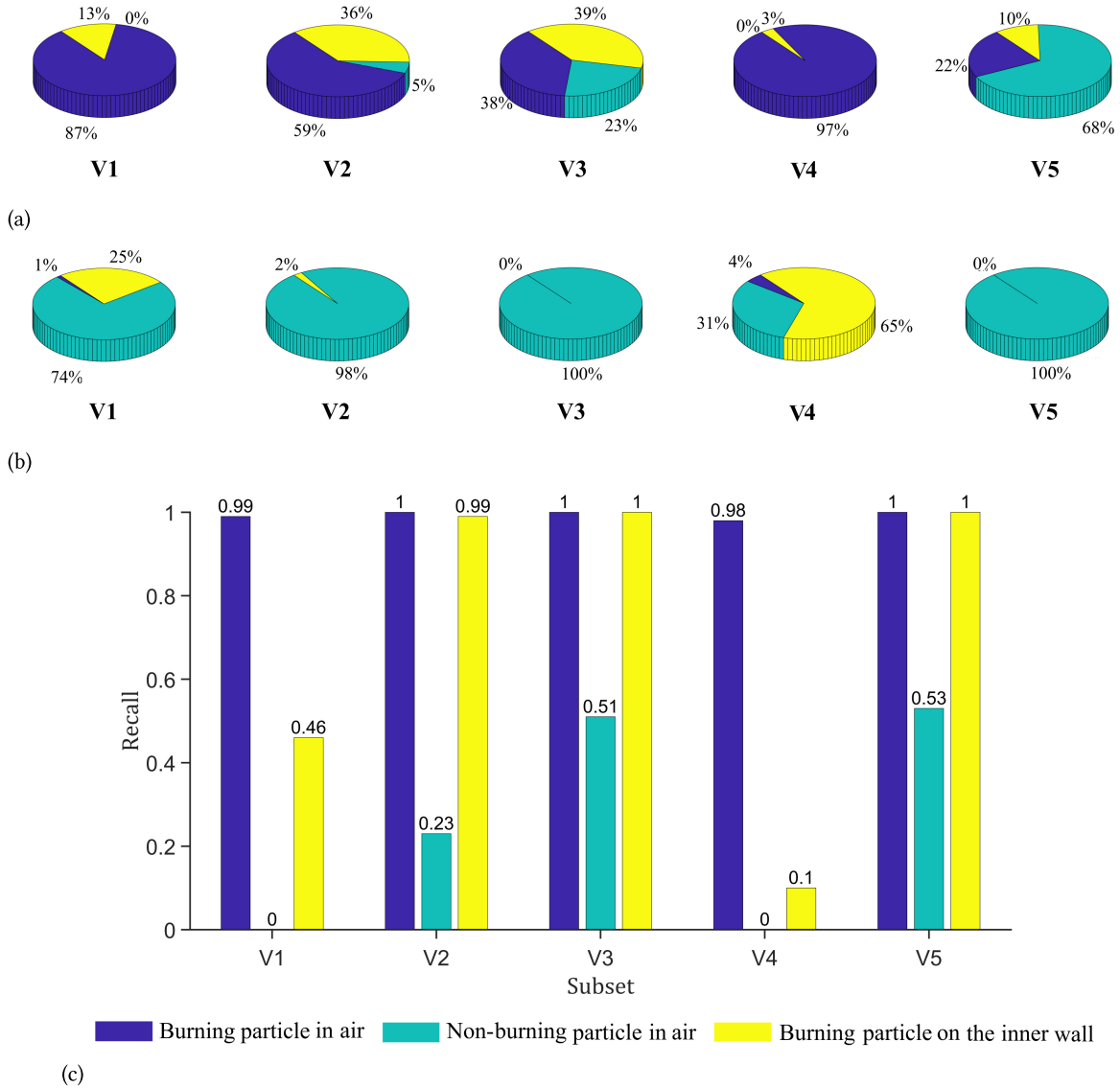
**Table 5.8.:** Performance and computation time of 3D-DBSCAN clustering approach with different parameter values using Python.

Parameter value	TP	FP	FN	Precision	Recall	F <sub>1</sub> -score	Computation time per image
<i>minPts</i> =6, $\epsilon$ =10	3965	33986	422	0.1045	0.9038	0.1873	4.846 s
<i>minPts</i> =6, $\epsilon$ =30	3561	6237	826	0.3634	0.8117	0.5020	5.385 s
<i>minPts</i> =6, $\epsilon$ =50	3073	2732	1314	0.5294	0.7005	0.6030	7.189 s
<i>minPts</i> =10, $\epsilon$ =15	3865	13251	522	0.2258	0.8810	0.3595	4.590 s
<i>minPts</i> =10, $\epsilon$ =30	3513	5296	874	0.3988	0.8008	0.5324	5.354 s
<i>minPts</i> =10, $\epsilon$ =50	3034	2436	1353	0.5547	0.6916	0.6156	7.149 s
<i>minPts</i> =10, $\epsilon$ =70	2586	1455	1801	0.6399	0.5895	0.6137	9.104 s
<i>minPts</i> =15, $\epsilon$ =30	3505	5002	882	0.4120	0.7990	0.5437	5.394 s
<i>minPts</i> =15, $\epsilon$ =50	2990	2341	1397	0.5609	0.6816	0.6154	7.394 s
<i>minPts</i> =15, $\epsilon$ =70	2570	1395	1817	0.6482	0.5858	0.6154	9.245 s

Small *minPts* and the large  $\epsilon$  means that points with the presence of fewer neighbor points within a large area can be identified as core points, which in turn determine the integration between the clusters. With more core points, the clusters tend to be relatively combined. Therefore, this combination of parameters leads to more points in a cluster and, thus, facilitates the building of oversized objects. Meanwhile, the disadvantage is that distinguishing small objects in close proximity becomes tough. Therefore, when the  $\epsilon$  gradually increases, the false detection (FP) is considerably reduced, thus enhancing precision. Simultaneously, small particles may be missed resulting in decreasing FN, so the corresponding recall is affected. Conversely, when *minPts* increases and  $\epsilon$  becomes smaller, *i.e.*, a point must have substantial neighboring points inside a small area in order to be recognized as a core point. Hence, the conditions for constructing core points are relatively harsh, *i.e.*, the clusters do not integrate easily, which leads to more detected particles and naturally more erroneous detection results (FP), thus seriously affecting the precision value. But, also, because of more detected particles, the number of undetected particles (FN) is reduced, so this parameter combination is often accompanied by a higher recall value. Generally, the impact of  $\epsilon$  is much more profound than *minPts*. With the identical *minPts*, changing the radius results in significantly different F<sub>1</sub>-scores. A changing *minPts* can also be reflected in the differences in the F<sub>1</sub>-scores, but the difference is less. As Table 5.8 shows, the best result comes from *minPts* = 10,  $\epsilon$  = 50. The thereby obtained result is also analyzed separately for each subset, as shown in the Figure 5.8.

The 3D clustering method performs as well as the previous 2D gray value based detection methods when detecting burning particles in air. Almost all burning particles in air were detected, which is

reflected in Figure 5.8 (c). The recall values of burning particles in air are all equal to or extremely close to 1. The DBSCAN clustering algorithm performs better than the 2D-OTSU method and slightly worse than the 2D-SIFT algorithm in detecting non-burning particles, which is caused by the smaller sizes of non-burning particles compared to the burning particles instead of their slighter brightness. When the size of a non-burning particle is more petite than  $minPts$ , it is mistakenly considered as noise by the algorithm and thus ignored. If  $minPts$  is reduced, more small-sized particles will be detected, but also FP will significantly increase. The major difference compared to the 2D detection methods occurs when detecting burning particles on the wall. In particular, for V1 and V4, where many burning particles on the wall exist, the recall values corresponding to burning particles on the inner wall are even below 0.5. As described in Chapter 3, because the particles attached to the wall may probably be integrated into the cluster representing the rotary kiln's inner wall, some of these particles are neglected. Notwithstanding, the flame of several particles of the rotary inner wall can be detected, leading to the detection of these particles.



**Figure 5.8.:** Detection performance of various classes of RDF particles by means of the 3D-DBSCAN clustering method. (a) Distribution of TP particle detections of the ground truth subsets. (b) Distribution of FN particle detections of the ground truth subsets. (c) Recall values of various particle classes in the ground truth subsets.

In addition, Table 5.8 also presents the computation time per image using Python in the last column. Overall, the computation time is within a few seconds for the selected parameter values. As shown in the table, the smaller the value of  $\varepsilon$  and  $minPts$ , the lower the computation time. Compared to  $minPts$ ,  $\varepsilon$  has a more significant impact on the computation time. And when  $\varepsilon$  is constant,  $minPts$  has a minimal effect on the computation time (of the order of 0.1 s).

### 3D clustering based on morphological operations

Unlike the DBSCAN clustering method, which builds clusters relying on the density of the input data (point cloud), the clustering method based on the morphological operations (MO-based clustering) utilizes the morphological dilation to connect points concerning their adjacency and forms different connected domains afterward [120]. The MO-based clustering first shifts and scales the spatial coordinates of input data points respectively into a range of  $[0,1]$ , which is then expanded by an adjustable parameter  $R$ . The transformed coordinates are then rounded to an integer, based on which a 3D grid can be initialized. If there is a data point corresponding to a spatial coordinate on the grid, the location is assigned a value of 1, otherwise 0. For each location with the value 1, its adjacent locations are also assigned with the value 1. The morphological dilation is applied to label the connected domains. With respect to each individual point, its Euclidean distances to points within particular labeled domains are computed, and the point is assigned the same label as the nearest domain afterward. The morphological dilation is an iterative process that stops when the number of connected domains remains unchanged.

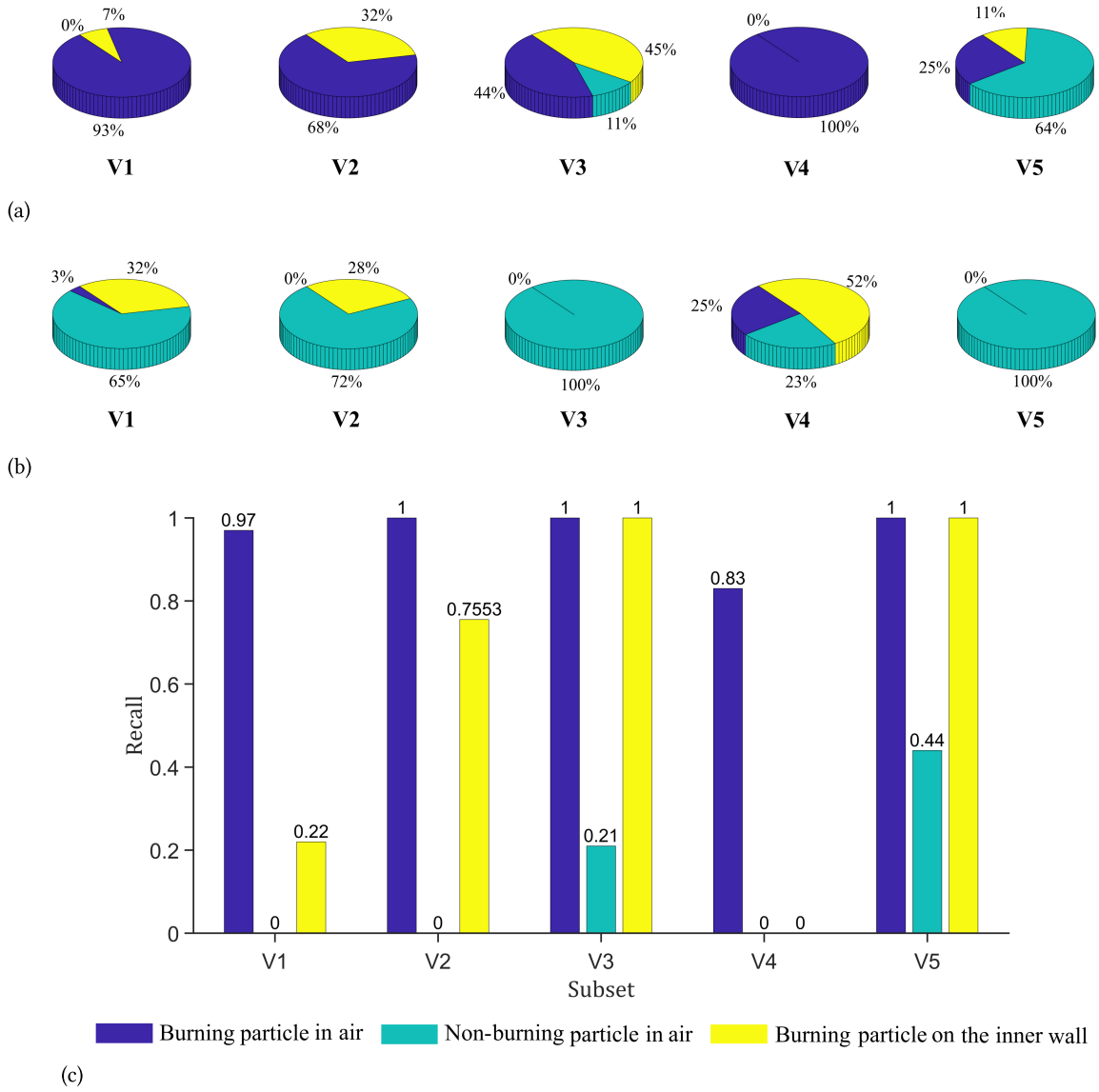
Table 5.9 summarizes the performance of the MO-based clustering method for all subsets with different values of  $R$ .  $R$  represents the range of the grid, viz., the 3D coordinates of the points are within the range of  $[0, R]$ . When  $R$  is small, all points are distributed inside a small range of the grid, which is beneficial to fuse clusters and thereby detect large objects. Under the circumstance, the error detection rate is low, i.e., the precision value performs accurately. On the other hand, due to the integration of clusters, several small particles may be incorrectly combined with their adjacent particles, which reduces the overall detection rate of particles and leads to an unsatisfactory recall value. On the contrary, when  $R$  takes a larger value, the point distribution is then relatively sparse, which is conducive to distinguishing individual particles. Therefore, the recall value increases in this case. Meanwhile, the number of false detections increases, and thus the performance in terms of precision is poor. Overall, this MO-based clustering method performs similarly to the previous DBSCAN clustering algorithm, and both are inferior to the 2D gray value based detection approaches.

**Table 5.9.:** Performance and computation time of 3D-MO-based clustering approach with different parameter values by means of MATLAB.

Parameter value	TP	FP	FN	Precision	Recall	F <sub>1</sub> -score	Computation time per image
$R=100$	1860	750	2527	0.7126	0.4240	0.5316	9.438 s
$R=150$	2619	1715	1768	0.6043	0.5970	0.6006	12.347 s
$R=200$	2869	4389	1518	0.3953	0.6540	0.4928	15.381 s
$R=250$	3325	4434	1062	0.4285	0.7576	0.5474	18.483 s
$R=300$	3468	5429	919	0.3898	0.7905	0.5221	21.207 s
$R=400$	3657	7987	730	0.3141	0.8336	0.4563	26.402 s
$R=500$	3732	10638	655	0.2597	0.8507	0.3979	31.367 s

According to Table 5.9, the best F<sub>1</sub>-score occurs when  $R$  equals 150. This result is then discussed detailedly concerning various particle classes, as illustrated in Figure 5.9. As with the previous detection

methods, the MO-based clustering approach performs best in detecting burning particles in air, with all recall values approximating or equal to 1, except for V4. The MO-based clustering approach is by far the worst algorithm for detecting non-burning particles. In V1, V2, and V4, where burning particles predominate, the MO-based method missed all non-burning particles because small-sized non-burning particles are probably incorporated into larger particles after the scaling and shifting. In V3 and V5, where non-burning particles dominate, the MO-based method also provides worse results than the 3D-



**Figure 5.9.:** Detection performance of various classes of RDF particles by means of the 3D clustering based on morphological operations. (a) Distribution of TP particle detections of the ground truth subsets. (b) Distribution of FN particle detections of the ground truth subsets. (c) Recall values of various particle classes in the ground truth subsets.

DBSCAN clustering algorithm. Moreover, the performance of the MO-based approach is also unsatisfactory when detecting burning particles on the wall, especially for V1 and V4, which contain more burning particles on the wall. The recall value corresponding to burning particles on the wall in V1 and V4 are extremely low. On the contrary, for V3 and V5, where non-burning particles are in the

majority, the recall value corresponding to wall particles could reach 1. The possible reason is that most non-burning particles are far from the wall, so the particles on the wall could be clearly distinguished.

The last column of Table 5.9 lists the computation time of the MO-based clustering approach for one image programmed by MATLAB for each  $R$  value. Overall, the computation time of the approach is longer than that of the 2D gray value based detection methods. As  $R$  increases, the range after scale transformation expands, *i.e.*, the scaled point range extends, which leads to a significant increase in computation time.

### 3D clustering based on quasi-cluster centers

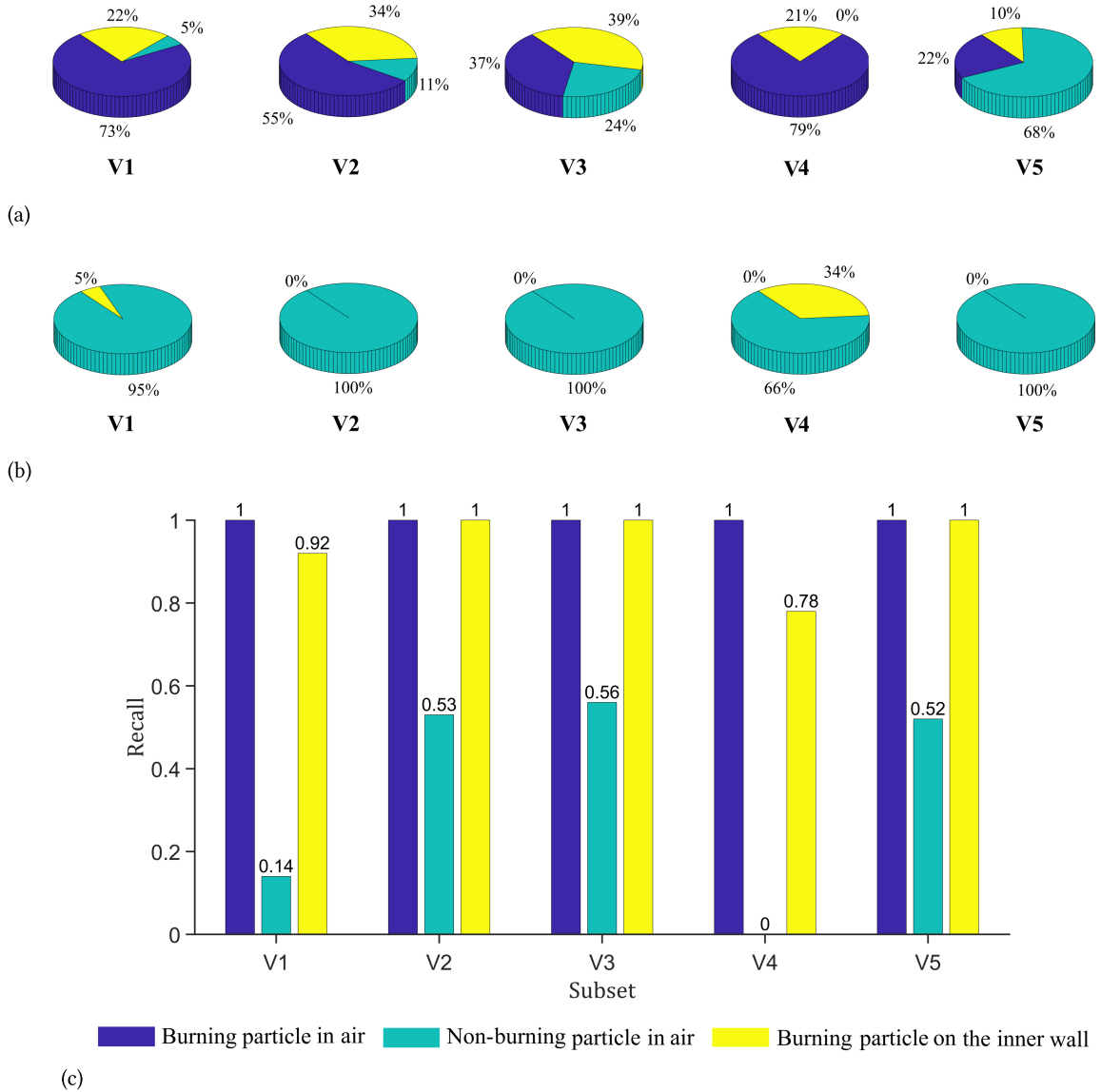
The clustering algorithm based on quasi-cluster centers (QCC) is another option to form clusters according to given point clouds. Principally, the QCC method divides clusters in accordance with defined sparse and dense regions [39]. At first, the density of each point with respect to a parameter  $K$  is computed, which is defined as the multiplicative inverse of its distance to the  $K$ th nearest neighbor. One point is considered a cluster center if its density is the maximum among its neighbors or reverse neighbors. Starting with an arbitrary cluster center, the QCC algorithm classifies the center and its sparse neighbor to the same cluster. Subsequently, all cluster points are scanned, and their sparse neighbors are completely classified into the cluster. This process will be repeated until all cluster centers are visited. Finally, the similarity of the obtained clusters is computed. The clusters are merged if the similarity is greater than a threshold value  $\alpha$ .

Table 5.10 concentrates on summarizing the performance of the 3D-QCC clustering algorithm in detecting particles of the ground truth with different parameter values. Similar to the performance of the previous DBSCAN and MO-based methods, the best  $F_1$ -score achieved by the QCC clustering algorithm is around 0.6, which is inferior to the 2D detection approaches. The accuracy of QCC is mainly affected by two parameters, namely  $K$  and  $\alpha$ , which represent the  $K$ th neighbors according to which the initialized cluster centers are selected and the threshold  $\alpha$  that determines whether the two clusters are similar enough to be fused, respectively. When the value of  $K$  is larger, the number of initial cluster centers for the same point is lower, so FP detections can be reduced, leading to improved precision. However, a larger  $K$  can also result in missing the relatively small particles next to the large particles, which increases FN and reduces recall as a consequence. A large  $\alpha$  represents the harsh conditions for the fusion of two clusters. Therefore, the probability of incorrectly fusing two clusters representing different particles decreases, which is reflected in the higher recall value. Meanwhile, a large  $\alpha$  also enhances the conditions for the fusion of the same clusters, which is to the disadvantage of reducing FP detections and enhancing precision. Overall, the influence of  $K$  is more significant than that of  $\alpha$ , and the impact of  $\alpha$  increases as  $K$  rises.

**Table 5.10.:** Performance and computation time of 3D-QCC clustering approach with different parameter values utilizing MATLAB.

Parameter value	TP	FP	FN	Precision	Recall	$F_1$ -score	Computation time per image
$K=20, \alpha=0.1$	3695	6461	692	0.3638	0.8423	0.5081	23.825 s
$K=25, \alpha=0.1$	3564	4945	823	0.4189	0.8124	0.5528	26.293 s
$K=30, \alpha=0.1$	3469	4090	918	0.4589	0.7907	0.6210	26.475 s
$K=35, \alpha=0.1$	3342	3411	1045	0.4949	0.7618	0.6000	26.859 s
$K=20, \alpha=0.2$	3712	7003	675	0.3464	0.8461	0.4916	24.633 s
$K=25, \alpha=0.2$	3598	5423	789	0.3988	0.8201	0.5366	24.759 s
$K=30, \alpha=0.2$	3496	4417	891	0.4418	0.7969	0.5685	25.023 s

The best  $F_1$ -score obtained by the QCC method is around 0.62, occurring when  $K = 30, \alpha = 0.1$ . Figure 5.10 shows the percentage of various particle classes in TP and FN for each data subset and the corresponding recall values achieved by these parameter values. The QCC method performs as well as the 2D method in detecting burning particles in air, with the ability to detect all airborne combustion particles. With respect to detecting burning particles on the inner wall, the QCC method is also superior to the previous DBSCAN and MO-based approaches. While the first two approaches can only detect less than half of the particles on the wall for V1 and V4, the QCC algorithm is capable of detecting the majority of the particles on the wall, and its performance in this respect is even slightly better than that of the OTSU method. As for the detection of non-burning particles, the QCC method also outperforms the other two clustering methods.



**Figure 5.10.:** Detection performance of various classes of RDF particles by means of the 3D clustering based on quasi-cluster centers. (a) Distribution of TP particle detections of the ground truth subsets. (b) Distribution of FN particle detections of the ground truth subsets. (c) Recall values of various particle classes in the ground truth subsets.



Finally, Table 5.10 also shows the MATLAB computation times for each image with the QCC clustering method for different parameter values. Obviously, this method also requires a longer computation time (basically more than 20 s). The effect of varying parameter selection values on the computation time is less pronounced. As presented by the table, the larger the value of  $K$  and the smaller the selected  $\alpha$ , the longer the computation time. When the value of  $K$  increases, the range of the selected nearest neighbors becomes wider, *i.e.*, more points will be merged into one cluster so that the computation time will be slightly longer. A small  $\alpha$ , on the other hand, represents a lenient similarity condition between two clusters, which leads to the need for multiple clusters to fuse again and so prolongs the computation time.

### Comparison of the detection performances

Since the results of the individual methods have been presented beforehand, the performance of the methods is summarized and compared with the introduced combination approaches in the following. In total, two 2D detection methods, as well as three 3D clustering algorithms, were tested in the study, which gives rise to a total of six available combinations. All results are listed in detail in Table 5.11, where the results are counted uniformly for all data sets, *i.e.*, TP, FP, and FN are counted in total, and then the absolute precision, recall, and  $F_1$ -score are computed.

**Table 5.11.:** Performance evaluation of particle detection employing 2D gray value based algorithms with median background subtraction, 3D clustering methods, and the corresponding combined approaches.

Method	TP	FP	FN	Precision	Recall	$F_1$ -score
2D-OTSU	3256	76	1131	0.9772	0.7422	0.8436
2D-SIFT	3687	354	700	0.9124	0.8404	0.8749
3D-DBSCAN Clustering	3034	2436	1353	0.5547	0.6916	0.6156
3D-MO-based Clustering	2619	1715	1768	0.6043	0.4970	0.6006
3D-QCC Clustering	3469	4090	918	0.4589	0.7907	0.6210
Combi OTSU+DBSCAN	3653	158	734	0.9585	0.8327	0.8912
Combi SIFT+DBSCAN	3784	222	603	0.9446	0.8625	0.9017
Combi OTSU+MO	3518	158	869	0.9570	0.8019	0.8726
Combi SIFT+MO	3723	219	664	0.9444	0.8486	0.8940
Combi OTSU+QCC	3647	220	740	0.9431	0.8313	0.8837
Combi SIFT+QCC	3779	287	608	0.9294	0.8614	0.8941

The results of the 2D gray value based detection approaches are presented first, followed by the 3D clustering algorithms, and finally, the outcomes of the six combinations. Overall, the  $F_1$ -scores achieved by the 2D detection approaches are around 0.85, which is much better than the  $F_1$ -scores obtained by the 3D clustering methods. Compared to 3D clustering methods, 2D detection approaches could reach a higher detection accuracy rate, and meanwhile, they are more capable of detecting real particles. The OTSU threshold selection method provides the highest precision values, while its recall value is relatively inferior. The 2D-SIFT algorithm, on the other hand, is more capable of detecting particles and, therefore, outperforms the OTSU method in terms of score, although it detects more error particles than the 2D-OTSU.

The 3D clustering results presented here are obtained by the set of selected parameters with the highest  $F_1$ -scores. Compared to the 2D gray value based approaches, the 3D clustering methods perform unsatisfactorily and inadequately. The performances of the clustering methods are mainly dragged by

precision. Because the 3D point clouds generated by the camera, on which the clustering method is based, fluctuate with a substantial amount of noise, more FP detections remain in the results resulting in inaccurate outcomes. Overall, the best  $F_1$ -score acquired by the 3D clustering approaches is around 0.6. The difference between the best  $F_1$ -score obtained by the QCC clustering method and the worst  $F_1$ -score provided by the MO-based clustering method is only 0.02. The QCC algorithm can detect the most particles at the expense of detection accuracy. Conversely, the MO-based clustering method achieves the highest precision value with an inferior ability to detect particles. The performance of the DBSCAN algorithm is relatively compromised. As mentioned before, the performances of the clustering methods are related to selected parameter values. Generally, the chosen parameters favor only one of precision and recall values, and the best  $F_1$ -score is the consequence of the coordination of these two indices.

The novel combined method follows the procedure described in Chapter 3. For the 3D clustering method, the set of parameter values providing the highest precision value is used to extract the cluster of the rotary kiln inner wall, which corresponds to DBSCAN  $minPts = 10, \epsilon = 50$ ; MO  $R = 100$ ; QCC  $K = 35, \alpha = 0.1$ . In general, the result with the highest precision corresponds to the one with the smallest value of FP detections. The reason for the small amount of FP is that the selected parameters extend clusters to an extensive range by including many points within each cluster, which is beneficial for detecting large objects. Since the inner wall of the rotary kiln is the largest object compared to the particles, choosing the parameters providing the highest precision facilitates the complete removal of the points belonging to the wall and further benefits the subsequent detection of particles. The points corresponding to the inner wall cluster are deleted from the point cloud. Subsequently, each corresponding clustering method is performed again based on the remaining points. Theoretically, this time the parameters ought to be chosen as the ones with the highest recall values in order to detect as many particles as possible. For the DBSCAN algorithm, the optimal recall value is achieved with  $minPts=6$  and  $\epsilon=10$ . However, this set of parameter values also causes a large amount of FP detections, which leads to more computational cost for the combination process and may result in more FPs in the final result. When  $minPts = 10$  and  $\epsilon=15$ , the resulting recall value is only 0.02 less than the previous  $minPts = 6$  and  $\epsilon = 10$ , and the FP particle detection is over 20,000 less. Hence, for the DBSCAN algorithm the parameter values with the second highest recall are chosen. Similarly, for the MO-based clustering approach the grid range  $R$  is selected as 400. Since the precision value of the QCC method under the circumstance of various parameter values differ slightly, the QCC algorithm utilizes the set of parameter values with the highest recall, *i.e.*,  $K = 20, \alpha = 0.2$ .

The experimental results on the ground truth datasets prove the superiority of the proposed new combination approach over the individual methods. On average, the  $F_1$ -scores of the combined methods are 3 to 5 percentage points higher than those of the corresponding 2D detection methods. When the 3D clustering methods are combined with 2D-OTSU, the recall value increases substantially. Meanwhile, integrating the 3D clustering outcomes introduces more FP detections giving rise to a slight decrease in the precision value. If the 3D clustering methods are combined with 2D-SIFT, both recall and precision are enhanced. Overall, combining with 2D-OTSU could lead to higher precision values, while combining with 2D-SIFT results in better detection ability. The best  $F_1$ -score occurs when 2D-SIFT is combined with 3D-DBSCAN with a  $F_1$ -score exceeding 0.9. This combination is also selected as the final detection method to provide input for subsequent studies.

The experimental results on the ground truth datasets prove the superiority of the proposed novel combination approach over the individual methods.

#### 5.1.4. Results of 3D particle classification

As mentioned in Chapter 3, the particles with 3D information are classified in accordance with their relative positions to the rotary inner wall to reduce the impacts of particles on the wall. A cylindrical inner wall model can be obtained by fitting the input point clouds by means of the MLESAC algorithm [113] introduced in Chapter 3.2, represented by the parameters in Table 5.12. The center represents the x, y, and z coordinates of the cylinder center, and the orientation stands for the reference orientation vector of the cylinder. Additionally, the radius is the radius size of the cylinder. All data are given in millimeters. Obviously, the estimated cylinder model differs from the theoretical cylinder of the rotary kiln inner wall, which has a reference vector of  $[0, 0, 1]$  with a radius of 700 mm. The difference could be the consequence of the test facility's construction error and the deviation of the plenoptic camera system. Since the primary purpose of the cylinder model is to classify the particles by comparing their distances to the cylinder wall (cylinder reference vector) with the radius, the length (height) of the cylinder is irrelevant in this process.

**Table 5.12.:** Parameters of the estimated cylindrical rotary kiln inner wall in mm.

Center	Orientation	Radius
[74.2750, 495.0013, 9257.7]	[21.6677, -41.4764, 1533.6]	772.1827

According to the determined cylindrical model, particles can be classified by presetting a threshold value of the distance from the cylinder axis. When a particle's distance to the cylinder axis exceeds a certain threshold, it can be considered a particle on the wall. Conversely, if a particle locates within a certain distance from the cylinder axis, it is identified as an particle in air (burning or non-burning). The 3D particle classification could benefit the determination of the particles' flight durations illustrated later. As schematically depicted in Figure 3.11, two Gaussian distributions respectively representing the distance of the particles in air and particles on the wall to the cylinder axis intersect at about 760 mm. However, some airborne particles are also distributed between 760-800 mm from the axis. In order to preserve as much as possible the airborne particles, the threshold value is increased to 800 mm. This threshold and the cylindrical model are initially applied to classify the particles in the ground truth to evaluate the accuracy and adequacy. The result is given in Table 5.13, where only particles with 3D information are classified. For classification performance evaluation, TP refers to correctly classified particles in air, while FP denotes particles that are wrongly classified as particles in air, which may be due to two reasons, *i.e.*, false particle detections or particles on the wall being classified as particles in air. For Table 5.13, where only particles from the ground truth dataset are classified, FP refers only to the second case, *viz.*, particles on the rotary kiln inner wall are misclassified. FN indicates missing unclassified particles in air, including particles in air that are not detected and particles in air that are incorrectly classified as particles on the wall. Similarly, FN in Table 5.13 refers only to the second case, *i.e.*, particles in air that are inaccurately classified.

As shown in Table 5.13, 263 particles on the wall are incorrectly classified as particles in air, while 242 are wrongly excluded after 3D classification. The vast majority of FPs, as well as FNs, are owing to camera measurement deviation. Since only less than 8% of the particles in air are incorrectly classified, the impact on the subsequent long-term multiple particle tracking is relatively slight.

**Table 5.13.:** Performance of 3D classification on ground truth dataset.

Subset	TP	FP	FN	Precision	Recall	F <sub>1</sub> -score
V1	757	63	45	0.9232	0.9439	0.9334
V2	365	57	32	0.8649	0.9194	0.8913
V3	202	33	16	0.8596	0.9266	0.8918
V4	1057	100	148	0.9136	0.8772	0.8950
V5	558	10	1	0.9824	0.9982	0.9902
Total	2939	263	242	0.9179	0.9239	0.9209

Table 5.14 presents the classification performance of the particle detection result from the combination of 2D-SIFT and 3D-DBSCAN. Likewise, only particle detections with available 3D information are taken into consideration. In terms of precision, the 3D classification performs well when applied to the particle detection result, and only 16 more detections are incorrectly classified as in air compared to the classification of the ground truth particles. Nevertheless, the performance is slightly inferior concerning the recall, with approximately 200 more FNs than the classification of the ground truth particles. Regarding the results provided in the previous Table 5.11, the recall achieved with respect to all particles is 0.86. Therefore, the majority of these 200 FNs should be the consequence of undetected particles. Another part of FNs is caused by the incorrect exclusion of the detections due to inappropriate 3D information. Overall, the 3D classification could reach an F<sub>1</sub>-score of about 0.88, which is sufficient to deliver a reliable basis for the later multiple particle tracking.

**Table 5.14.:** Performance of 3D classification on the detection result from the combination of 2D-SIFT and 3D-DBSCAN.

Subset	TP	FP	FN	Precision	Recall	F <sub>1</sub> -score
V1	709	61	93	0.9208	0.8840	0.9020
V2	350	74	47	0.8255	0.8816	0.8526
V3	183	40	35	0.8206	0.8394	0.8299
V4	969	82	236	0.9220	0.8041	0.8590
V5	496	22	63	0.9575	0.8873	0.9211
Total	2707	279	474	0.9066	0.8510	0.8779

## 5.2. Results of multiple particle tracking

This section focuses on presenting and evaluating the obtained 2D and 3D particle trajectories. The 2D trajectories in image coordinates are acquired by the novel proposed 2.5D GNN and JPDA with Kalman filter, respectively. The 2D trajectories are subsequently converted into 3D and then estimated by polynomials jointly in three spatial directions. Based on the fitted 3D trajectories, the particle flight and combustion behaviors can be analyzed. Additionally, the 2D trajectories are evaluated quantitatively in this section according to the criteria introduced in the following. Moreover, the ground truth used to evaluate the tracking performance is also described at the beginning.

### 5.2.1. Ground truth of particle tracking

In order to evaluate the multiple particle tracking results objectively and quantitatively, the trajectories of various particles over a specific period of time are manually tracked, and then the positions in image

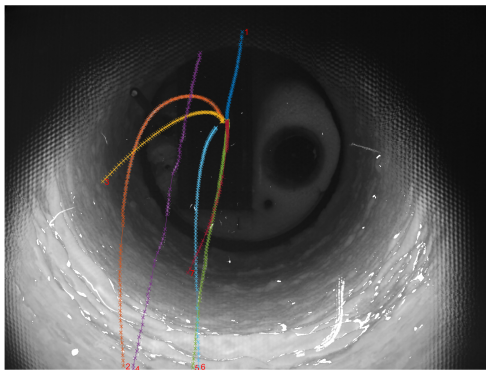
coordinates are noted. Unlike the evaluation of particle detection, where various fractions are mixed to achieve a more accurate assessment, in particle tracking, different fractions are followed separately to obtain specific trajectories for subsequent analysis of flight characteristics. As shown in Figure 2.7, a total of four fractions are tracked, whose trajectories and properties are also presented and discussed separately later.

A fragment of the ground truth is given in Figure 5.11. Each trajectory is separately stored in a .txt file, where the first column represents the number of frames, and the second and third columns stand for the rows and columns in image coordinates, respectively. Columns with not a number (NaN) mean one particle does not appear in this frame, or it is simply invisible, which is probably caused by the inappropriate angle of the particle relative to the plenoptic camera.

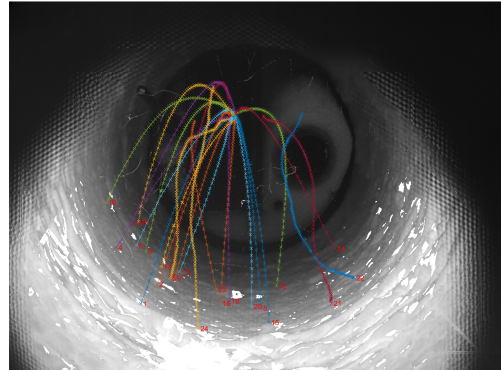
62	NaN	NaN	
63	NaN	NaN	
64	NaN	NaN	
65	483.785714000000		936.285714000000
66	477.842105000000		937.473684000000
67	471.840000000000		937.440000000000
68	465.645161000000		938.354839000000

**Figure 5.11.:** Extract from a .txt file of the ground truth trajectories. First column indicates the frame number, while the coordinates of a particle are presented in the last two columns.

Figure 5.12 shows two trajectories, wood chips, and PE-granules. Because only complete trajectories are valuable for subsequent studies, trajectories that originate from the lance and continue for some time are marked. Strictly speaking, the purple trajectory of the wood chip does not start exactly from the lance. As a consequence of its length and the small number of trajectories overall, it is also present in the ground truth to allow for a more objective evaluation of the tracking performance. In comparison, the trajectory of wood is more regular. Table 5.15 summarizes the trajectories of all particle fractions, including the amount and their corresponding number of frames. The selected frames vary because the particles were thrown in the rotary kiln manually, resulting in a considerable variation in the number of different particle fractions within a certain period.



(a)



(b)

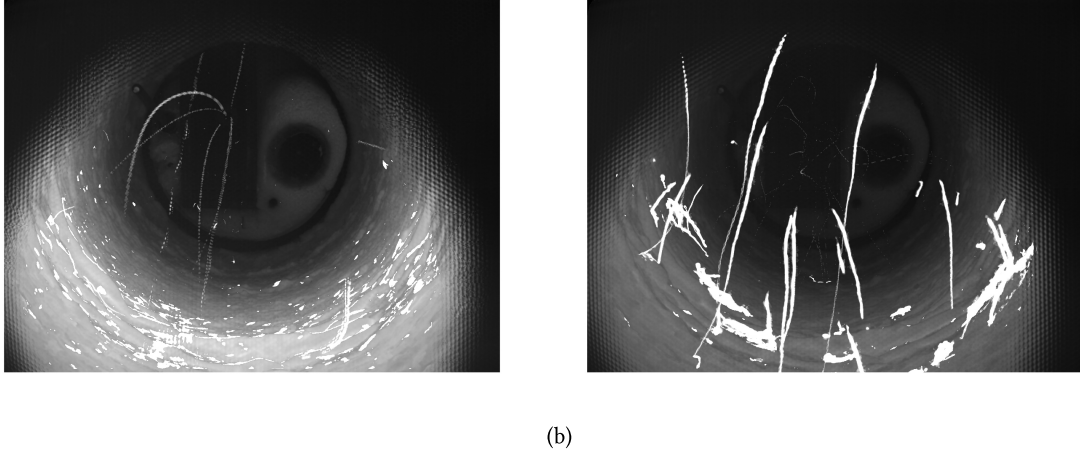
**Figure 5.12.:** Ground truth trajectories of wood chips (a) and PE granules (b). Crosses stand for particle coordinates and each color indicates one individual trajectory.

**Table 5.15.:** Overview of ground truth trajectories of the RDF fractions with their corresponding frame intervals and amount of trajectories.

Fraction	Frame	Amount of trajectories
Paper shred	550-650	32
Confetti	200-500	37
PE granule	200-500	25
Wood chip	100-400	7

### 5.2.2. Tracking performance evaluation criteria

As a preliminarily qualitative and visual indicator of the tracking performance, optical ground truths are generated by adding a single background image to a set of adjacent difference images, which represents the difference between an image and a calculated statistical background image, as exemplified in Figure 5.13. These images enable a visual comparison of the computed and actual trajectories and, thus, are able to provide qualitative information concerning tracking performance.

**Figure 5.13.:** Adding a set of adjacent difference images and a background image of each fraction as the optical ground truth of wood chips (a) and confetti (b).

Besides the optical ground truth, the particle positions are also manually tracked and documented, enabling a quantitative evaluation of the tracking performance. In order to assess the tracking performance quantitatively, Chenouard *et al.* proposed a set of evaluation indices [14]. A particular trajectory  $\theta$  is considered as a temporal series of subsequent positions and hence, can be denoted as  $\theta = \theta(t) = ((x(t), y(t), z(t)))$  with  $t = t_1 \dots t_{int}$ . The gated Euclidian distance between two simultaneous positions from two distinct trajectories is defined as

$$\| \theta_1(t) - \theta_2(t) \|_{2,\epsilon} \triangleq \min(\| \theta_1(t) - \theta_2(t) \|_2, \epsilon), \quad (5.1)$$

where  $\epsilon$  stands for a predefined gate. If two positions are more than  $\epsilon$  apart, their distance is regarded as  $\epsilon$ . For time dimensions, when only one trajectory has a corresponding position, *e.g.*, only  $\theta_1(t)$  exist, the distance at time  $t$  is denoted as  $\epsilon$ . If none of the trajectories exist at time  $t$ , the distance is considered as 0. Accordingly, the distance between two trajectories is formulated as

$$d(\theta_1, \theta_2) \triangleq \sum_{t=0}^{T-1} (\| \theta_1(t) - \theta_2(t) \|_{2,\epsilon}), \quad (5.2)$$

where  $T$  represents the length of the image sequence.

Let  $\mathcal{X}$  denote a set of ground truth trajectories with  $|\mathcal{X}|$  individual trajectories, and  $\mathcal{Y}$  a set of candidate trajectories. The distance between  $\mathcal{X}$  and  $\mathcal{Y}$  is defined as

$$d(\mathcal{X}, \mathcal{Y}) \triangleq \min \sum_{k=1}^{|\mathcal{X}|} d(\theta_{\mathcal{X}}^k, \theta_{\mathcal{Z}}^k) \text{ with } \mathcal{Z} \in \Omega_{\mathcal{Y}}. \quad (5.3)$$

Here,  $\Omega_{\mathcal{Y}}$  stands for the acquired ensemble sets of trajectories by selecting  $|\mathcal{X}|$  trajectories from the extended candidate tracks  $\tilde{\mathcal{Y}}$ , which extends  $\mathcal{Y}$  with  $|\mathcal{X}|$  dummy empty tracks.  $\mathcal{Z}$  represents an arbitrary element in  $\Omega_{\mathcal{Y}}$ . The assignment problem between  $\mathcal{X}$  and  $\tilde{\mathcal{Y}}$  is addressed by applying the Munkres algorithm.

Based on the optimal pairing  $(\mathcal{X}, \mathcal{Z}^*)$  with minimum distance, the following indices to assess the tracking performance are computed:

- Normalized pairing score

$$\alpha(\mathcal{X}, \mathcal{Y}) \triangleq 1 - \frac{d(\mathcal{X}, \mathcal{Y})}{d(\mathcal{X}, \emptyset)}. \quad (5.4)$$

$\emptyset$  stands for the set of  $|\mathcal{X}|$  dummy empty trajectories. If the candidate set of trajectories  $\mathcal{Y}$  is consistent with the ground truth set  $\mathcal{X}$ , the distance  $d(\mathcal{X}, \mathcal{Y})$  equals 0. In contrast,  $d(\mathcal{X}, \emptyset)$  is an upper bound of the distance. Hence, the defined normalized pairing score ranges in the interval  $[0, 1]$ . If  $\mathcal{X}$  and  $\mathcal{Y}$  match exactly,  $\alpha(\mathcal{X}, \mathcal{Y}) = 1$ , and if  $\mathcal{X}$  and  $\mathcal{Y}$  are entirely distinct,  $\alpha(\mathcal{X}, \mathcal{Y}) = 0$ .

- Full normalized pairing score

$$\beta(\mathcal{X}, \mathcal{Y}) \triangleq \frac{d(\mathcal{X}, \emptyset) - d(\mathcal{X}, \mathcal{Y})}{d(\mathcal{X}, \emptyset) + d(\tilde{\mathcal{Y}}, \emptyset_{\tilde{\mathcal{Y}}})}, \quad (5.5)$$

where  $\tilde{\mathcal{Y}}$  denotes the trajectories in  $\mathcal{Y}$  but not in  $\mathcal{Z}^*$  and  $\emptyset_{\tilde{\mathcal{Y}}}$  stands for a set of dummy trajectories with  $|\tilde{\mathcal{Y}}|$  elements. The trajectories in  $\tilde{\mathcal{Y}}$  are considered spurious trajectories and typically consist of trajectories associating erroneous detections and trajectories combining detections corresponding to various ground truth trajectories. If no spurious trajectory exists,  $\beta(\mathcal{X}, \mathcal{Y})$  is equivalent to  $\alpha(\mathcal{X}, \mathcal{Y})$ . Otherwise,  $\beta(\mathcal{X}, \mathcal{Y})$  ranges from 0 to  $\alpha(\mathcal{X}, \mathcal{Y})$ .

- The Jaccard similarity index for positions

$$JSC \triangleq \frac{TP}{TP + FN + FP}. \quad (5.6)$$

If the distance between two temporal positions is smaller than  $\epsilon$ , they are considered as a matching pair. The number of matching pairs of positions in  $(\mathcal{X}, \mathcal{Z}^*)$  is denoted as TP. FN stands for the number of non-matching pairs. FP represents the total amount of positions in  $\tilde{\mathcal{Y}}$ .

- The Jaccard similarity index for tracks

$$JSC_{\theta} \triangleq \frac{TP_{\theta}}{TP_{\theta} + FN_{\theta} + FP_{\theta}}. \quad (5.7)$$

$TP_{\theta}$  denotes the number of non-dummy tracks in  $\mathcal{Z}^*$ . On the contrary,  $FN_{\theta}$  denotes the number of dummy tracks in  $\mathcal{Z}^*$ .  $FP_{\theta}$  counts the number of tracks in  $\tilde{\mathcal{Y}}$ .

### 5.2.3. 2.5D Multiple particle tracking

The particle detections obtained by the combined 2D-SIFT and 3D-DBSCAN approach are associated by employing the 2.5 D multiple particle tracking algorithms demonstrated in Chapter 4. This section presents the 2D trajectories of various fuel fractions with and without the novel post-processing framework, together with their corresponding evaluations in accordance with the indices introduced before. Moreover, all positions here refer to the detected coordinates of the particles. Since the ground truth is generated optically on the basis of captured images, utilizing the detected positions instead of the updated coordinates by the Kalman filter benefits an objective and accurate assessment of the tracking performance.

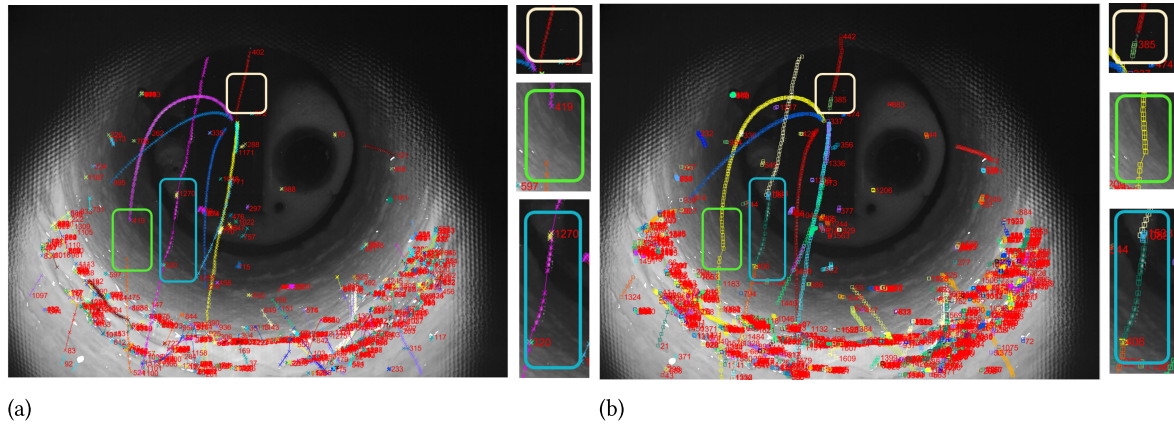
At first, the particle tracking is processed by GNN and JPDA. An instance of the tracking result utilizing GNN and JPDA without postprocessing is shown in Figure 5.14 (a) and (b), whose manual ground truth and optical ground truth are presented in Figure 5.12 (a) and Figure 5.13 (a), respectively. The identical color indicates the positions of one particular tracklet. In comparison with GNN, JPDA updates the state vectors with larger variances as the consequence of an additional term of the covariance update function in JPDA (Equation 2.56) concerning the mixture probability density function [10], which results in broad elliptical gating areas determined by JPDA. Because the computational cost and entailed storage capacity of JPDA increase enormously with the rising gate and the number of tracks, the 2D Mahalanobis distance gating of JPDA is selected to be smaller than GNN.

Apparently, the direct outcomes of both approaches are with the presence of discrepancies, as highlighted by the boxes in Figure 5.14, which are primarily caused by detection inaccuracies, such as missing detections, false detections, and duplications. Missing detections could lead to interruptions in the tracklets, as illustrated in the green rectangle of Figure 5.14 (a) and the yellow rectangle the Figure 5.14 (b). And meanwhile, the detection duplications can trigger another tracklet, thus also causing the incompleteness of the trajectories, as presented in the blue box in Figure 5.14 (b). Besides, the two detection algorithms yield different results for the same input detections, primarily due to the different values of the 2D Mahalanobis distance gating threshold and distinct functions for updating the covariance matrices. As mentioned, the 2D gate selected by JPDA at the beginning is small, resulting in missing potential associations at the beginning of each tracklet, as shown in the yellow area of Figure 5.14 (b). While GNN has a relatively large gate selection initially, its updated covariance for computing the Mahalanobis distance is smaller than JPDA. Hence, in the later part of a tracklet, JPDA can successfully associate the detections with the corresponding tracklets, but GNN might fail to search for the candidate detection in the gate, as shown in the green area in Figure 5.14 (a).

In general, the direct outcomes of both approaches are unable to provide sufficient accuracy for further analysis of spatial particles' flight trajectories and properties. In addition, choosing the appropriate initial 2D gate for each method is also challenging since subsequent updates of the states' covariances are difficult to estimate quantitatively. The initial gate, however, is critical to the tracking results. Therefore, the presented novel post-processing framework is applied to the tracking results with the parameter values listed in Table 5.16.

As depicted in Figure 5.15, the obtained trajectories of wood chips by GNN and JPDA differ only slightly from each other after the proposed post-processing procedure. Compared to tracking results without post-processing in Figure 5.14, the tracking outcomes with post-processing prove better visual accuracy. Referring to the ground truth in Figure 5.13, the results are also close to the actual particle trajectories. Withal, the trajectories of the PE granules derived by the two algorithms are also shown in the figure. Compared with the ground truth trajectories in Figure 5.12, the post-processed trajectories indicate a high degree of similarity to the actual traces.





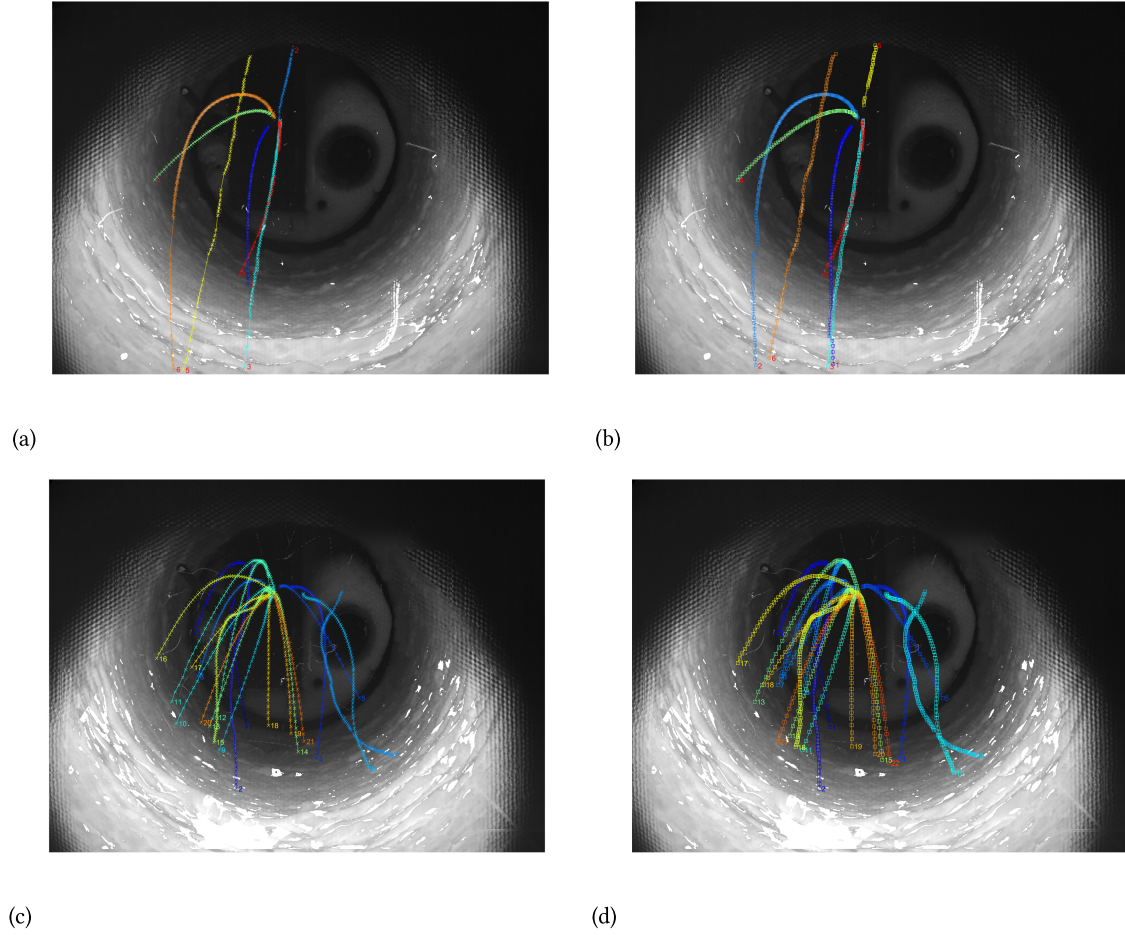
**Figure 5.14.:** Tracking results of wood chips. (a) Tracking result of GNN. Crosses stand for detections. Detections of same tracklets are linked with solid lines. (b) Tracking result of JPDA. Squares stand for detections. Detections of same tracklets are linked with solid lines.

**Table 5.16.:** Parameter values of the post-processing framework.

Parameter	Value	Unit	Parameter	Value	Unit
$\mathcal{D}_\epsilon$	2	frame	$t_{\epsilon,CONNECT}$	15	frame
$V_\epsilon$	1	pixel/frame	$kn_{\epsilon,merge}$	5	frame
$\mathcal{R}_\epsilon$	0.9	-	$z_\epsilon$	1500	mm
$\Delta V_\epsilon$	18	pixel/frame	$\epsilon$	10	pixel
$\theta_\epsilon$	30	°	$a$	2	-
$b$	2	-	$c$	0.1	-
$d$	1	-	$e$	2	-
$f$	$10^{11}$	-	$w$	15	pixel

For the purpose of a quantitative assessment and comparison of the tracking performance, the tracking outcomes are evaluated utilizing the measurements mentioned in Chapter 5.2.2, *viz.*,  $\alpha$ ,  $\beta$ ,  $JSC$ ,  $JSC_\theta$ . Table 5.17 exhibits the performance of multiple particle tracking with and without post-processing. As visually presented in Figure 5.14 and quantitatively shown in Table 5.17, the direct tracking outcomes of GNN and JPDA are relatively inaccurate with lower measurement values owing to the incompleteness and inexactitude of the tracklets. In particular, tracking performances of confetti and paper shreds are relatively worse. Since small papers are, in comparison to wood chips and PE granules, light in weight and therefore fall slowly down with rotations in the air, their trajectories are tangled and intertwined, which exerts adverse impacts on particle tracking.

As indicated in Table 5.17, the proposed novel post-processing approach benefits a precise tracking performance and significantly facilitates the tracking performance with respect to the four measures. Since a lot of faulty tracklets and outliers in the tracklets are deleted, the acquired trajectories show more similarity to the original trajectories. Thus, the measurements achieve considerable enhancements. While  $\alpha$  and  $\beta$  of the direct tracking outcomes from GNN and JPDA are relatively low, with values of roughly 0.2–0.3 for the majority of the fractions, the tracking results with post-processing provide considerably higher  $\alpha$  and  $\beta$  values with values around 0.5–0.6. Meanwhile,  $JSC$  and  $JSC_\theta$  are also significantly meliorated with up to about 40 %. In spite of the noticeable improvement of the  $JSC$  by the post-processing for confetti (0.06→0.47 and 0.08→0.45, respectively), the  $JSC$  values for confetti



**Figure 5.15.:** Tracking results of wood chips and PE granuels with post-processing. (a) Tracking result of wood chips using GNN. Crosses stand for detections. Detections of same tracklets are linked with solid lines. (b) Tracking result of wood chips by means of JPDA. Squares stand for detections. Detections of same tracklets are linked with solid lines. (c) Tracking result of PE granuels using GNN. (d) Tracking result of PE granuels by means of JPDA.

**Table 5.17.:** Evaluation of tracking performance with and without post-processing utilizing GNN and JPDA. Upper values illustrate performance of GNN and JPDA without post-processing, and lower values displays performance of GNN and JPDA with post-processing.

Measure Fraction	$\alpha$		$\beta$		$JSC$		$JSC_{\theta}$	
	GNN	JPDA	GNN	JPDA	GNN	JPDA	GNN	JPDA
Wood chips	0.61	0.61	0.60	0.59	0.68	0.69	0.47	0.44
with post-processing	0.67	0.68	0.67	0.68	0.76	0.79	1.0	1.0
Confetti	0.05	0.06	0.05	0.06	0.06	0.08	0.22	0.20
with post-processing	0.45	0.43	0.44	0.41	0.47	0.45	0.76	0.74
Paper shreds	0.15	0.19	0.14	0.18	0.20	0.29	0.43	0.42
with post-processing	0.60	0.62	0.59	0.60	0.61	0.65	0.84	0.77
PE granules	0.40	0.43	0.35	0.38	0.40	0.44	0.40	0.28
with post-processing	0.62	0.64	0.59	0.64	0.64	0.71	0.84	0.88

remain below those for the other fuel fractions, which can be attributed to their weak visibility and complicated and irregular traces.

Overall, the two employed data association algorithms, GNN and JPDA, provide comparable outcomes for tracking various RDF fuel particles. Regarding measurements relating to position matching, *e.g.*,  $\alpha$ ,  $\beta$ , and  $JSC$ , JPDA presents a slight advantage of a few percent since GNN might assign several noises to tracks, leading to deviations of several particle positions, whereas JPDA updates the state vector based on all detections within the gate, leading to compensation of deviations caused by noise. With reference to track matching ( $JSC_\theta$ ), the difference is even slighter than the previous position matching. The most apparent disparity occurs when tracking paper shreds, whose relatively large size and intricate gray value distribution result in more FP detections leading to several FP trajectories employing JPDA.

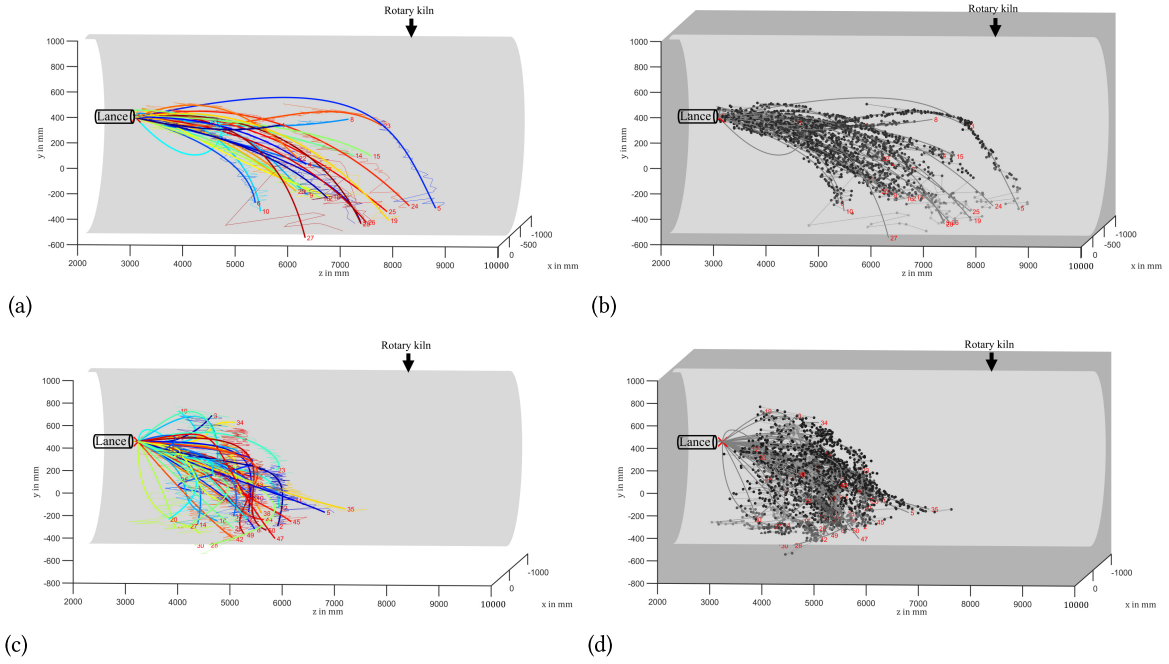
Concerning tracking accuracy, GNN and JPDA are able to provide comparable performance. Nevertheless, as for the aspect of the computational cost and essential storage capacity, JPDA is clearly at a disadvantage. JPDA entails significantly more computing cost and storage capacity for weighting candidates within the gate jointly, which disables its application for long-time tracking with a substantial amount of detections. Nevertheless, in order to discuss the flight and combustion properties of various fuel particles, a long-time observation and tracking to accumulate a sufficient amount of complete trajectories is regarded as the prerequisite. In addition, JPDA computes the average coordinates of all detections within the gate as the temporal track locations, which might enhance the fluctuation of the converted 3D trajectories. Taking these factors into consideration, *i.e.*, the suitability of long-time tracking, computational cost, and quality of the obtained trajectories, the particles are followed by means of the GNN algorithm. The thereby acquired 2D trajectories are subsequently converted into 3D.

Only the result fragments corresponding to the ground truth period are shown in the section. In fact, the multiple particle tracking process is continuous with respect to a sequence. Since each sequence lasts about 3000 frames with a framerate of 330 fps, each tracking result corresponds to trajectories within about 10 s. While wood chips, paper shreds, and confetti were captured with the maximal frame rate of 330 fps, the framerate for capturing PE granules was reduced to 153.85 fps. As a consequence of the small size and transparency of PE granules, they require necessarily sufficient illumination to be successfully captured by the plenoptic camera. Therefore, the entailed shutter time is increased, and the framerate is reduced accordingly, which results in a tracking time for PE granules more than twice as long as the other fractions with the same frames. The full trace results of the experimented fuel fractions under the condition of two different air feed pressures (4.5 bar and 5 bar) are available in the appendix.

#### 5.2.4. 3D Regression

With the completion of 2.5D multiple particle tracking, the corresponding 3D trajectories can be extracted based on the 3D coordinates provided by the plenoptic camera. As illustrated in Chapter 4.3, the directly converted spatial trajectories fluctuate considerably by reason of measurement inaccuracy in the depth direction. Thus, the trajectories are estimated jointly with three polynomials over time in three directions to fulfill the condition that each trajectory ought to originate from a fixed point (the lance) simultaneously. For each direction, the polynomial representing the correspondingly directional location of one specific trajectory is a function of time. Instances of directly converted 3D trajectories together with their corresponding estimated polynomials are given in Figure 5.16. Moreover, the trajectories are also plotted in accordance with the corresponding temporal average particle gray values. The figure indicates that the trajectories of wood particles are clearly farther in the depth direction than

the confetti. In contrast, the motion of confetti in the x-direction is significantly dispersed. Concerning the brightness of the trajectory, the trajectories of wood chips are relatively dark, whose brightness only slightly increases when close to the inner wall of the rotary kiln, while the brightness of the confetti trajectory is a mixture of light and dark. Therefore, it can be inferred that the wood chips are basically non-combustible during their flights, while some confetti can be ignited during flight. The trend of the trajectory brightness serves as the basis for the subsequent analysis of particles' combustion behaviors. Furthermore, by virtue of the joint regression, the exact time when a particle passes the lance can be figured out, which is conducive to analyzing the flight duration and ignition time of the fuel particles. The detailed flight and combustion properties are elaborated on in the next section.



**Figure 5.16.:** Estimated spatial trajectories of wood chips and confetti with corresponding particle gray values. (a) Estimated polynomials of wood chips. Red cross stands for the lance, thick solid lines are the estimated polynomials, and thin solid lines represent the direct converted 3D trajectories. (b) Estimated polynomials of confetti. (c) Estimated polynomials of wood chips with corresponding particle gray values. The red cross stands for the lance, thin solid lines are the estimated polynomials, and points' brightness represent the corresponding gray values of the detections. (d) Estimated polynomials of confetti with corresponding particle gray values.

The estimated polynomials are quantitatively assessed following the measures illustrated in Chapter 4.3. Table 5.18 documents the evaluation of various particle trajectories. The top, middle, and bottom sub-rows of the table represent the median of each measure corresponding to the x, y, and z directions, respectively. In terms of RMSE, the trajectories of various fractions perform essentially similarly in the x and y direction, with the level of single digits. The paper shred is an exception, whose sizeable windward area and significant wind resistance give rise to irregular motion and further larger RMSE of the fitted trajectories. The RMSE in the depth direction is much higher than the other two directions for all fractions, which is the consequence of the camera's intense fluctuations and the particles' rapid motion in the depth direction. On the contrary, MAPE in the z-direction is the lowest caused by the relatively large depth coordinates of the particle. By definition, MAPE is proportional to the difference between each coordinate and the mean divided by the coordinate, *i.e.*, the larger the coordinate value, the smaller the MAPE tends to be. In practice, the coordinates in the x and y directions are in the range of a few hundred millimeters, whereas the coordinates in the depth direction are in the order of several thousand millimeters. Hence, although the fluctuation in the depth direction is the strongest,

**Table 5.18.:** Evaluation of 3D regression. Upper, middle, and lower values represent the respective measure values in the x, y, and z directions.

Measure	RMSE	MAPE	$R^2$	$R^{*2}$
	x	x	x	x
	y	y	y	y
	z	z	z	z
Wood chips	7.86	6.70	0.97	0.97
	7.22	5.13	0.99	0.99
	143.56	2.28	0.98	0.98
Confetti	5.46	2.11	0.98	0.98
	5.61	3.02	0.99	0.99
	140.15	2.09	0.72	0.67
Paper shreds	18.77	6.94	0.77	0.75
	12.40	9.06	0.98	0.98
	218.99	3.97	0.55	0.52
PE granules	6.01	4.84	0.95	0.94
	6.77	6.04	0.99	0.99
	170.24	2.41	0.88	0.84

its converted MAPE is the smallest. In general,  $R^2$  and  $R^{*2}$  in the x- and y- directions reach roughly 0.9, indicating the satisfied adequacy of the regression model. Due to the complex motion of the paper shred,  $R^2$  and  $R^{*2}$  of the regressed model is inferior. Overall,  $R^2$  and  $R^{*2}$  in the depth direction are small due to the fluctuations. Based on the rubric demonstrated in [88], these values can indicate a qualified regression model as the basis for the subsequent study.

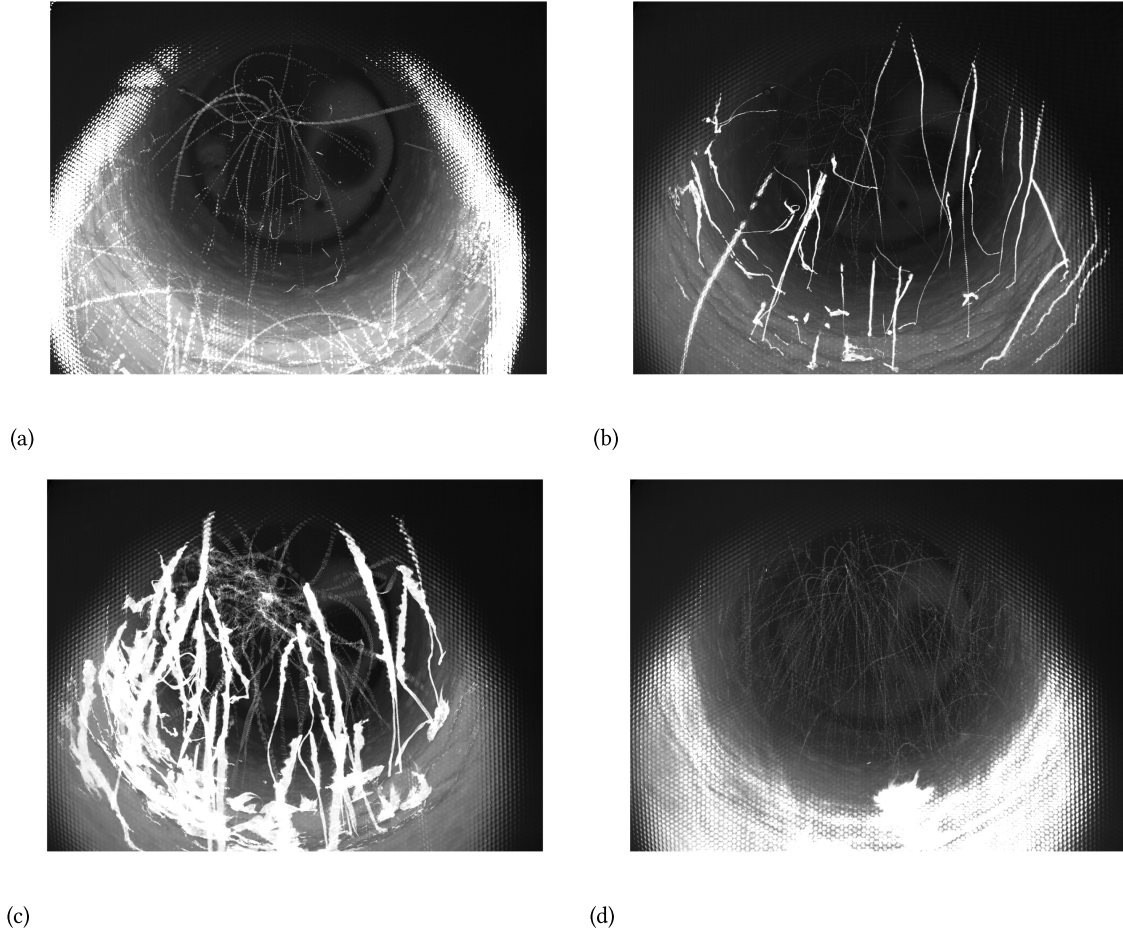
### 5.3. Analysis of RDF flight and combustion properties

The spatial particle trajectories with their corresponding gray value progressions enable the investigation into several particles' flight and combustion behaviors. In the following, the key findings concerning the properties of the particles, including flight duration, velocity in the depth direction, and ignition time, are elaborated. In order to create an initial impression of the particles' motion and combustion properties, the optical ground truth of the particle fractions is given in Figure 5.17. Unlike Figure 5.13, which presents trajectories only within one time period, Figure 5.17 adds up the adjacent difference images of each fraction over the entire sequence. As depicted in the figure, wood chips and PE granules move more regularly compared to paper, which can also give rise to the superior tracking performance of wood chips and PE granules. In addition, the paper materials can be ignited during the flight, whereas airborne wood chips and PE granules do not combust basically.

#### 5.3.1. Flight Behaviors of RDF Particles

This section discusses investigations into two flight behaviors of RDF fractions. At first, the flight duration of the particles, which is also deemed the particles' dwell time, is described. Subsequently, the RDF particles' velocities in the depth direction under the condition of two different air feed pressures (4.5 bar and 5 bar) are extensively explained. In principle, the speeds corresponding to three spatial directions can be computed and analyzed. Notwithstanding, the horizontal and vertical velocities are



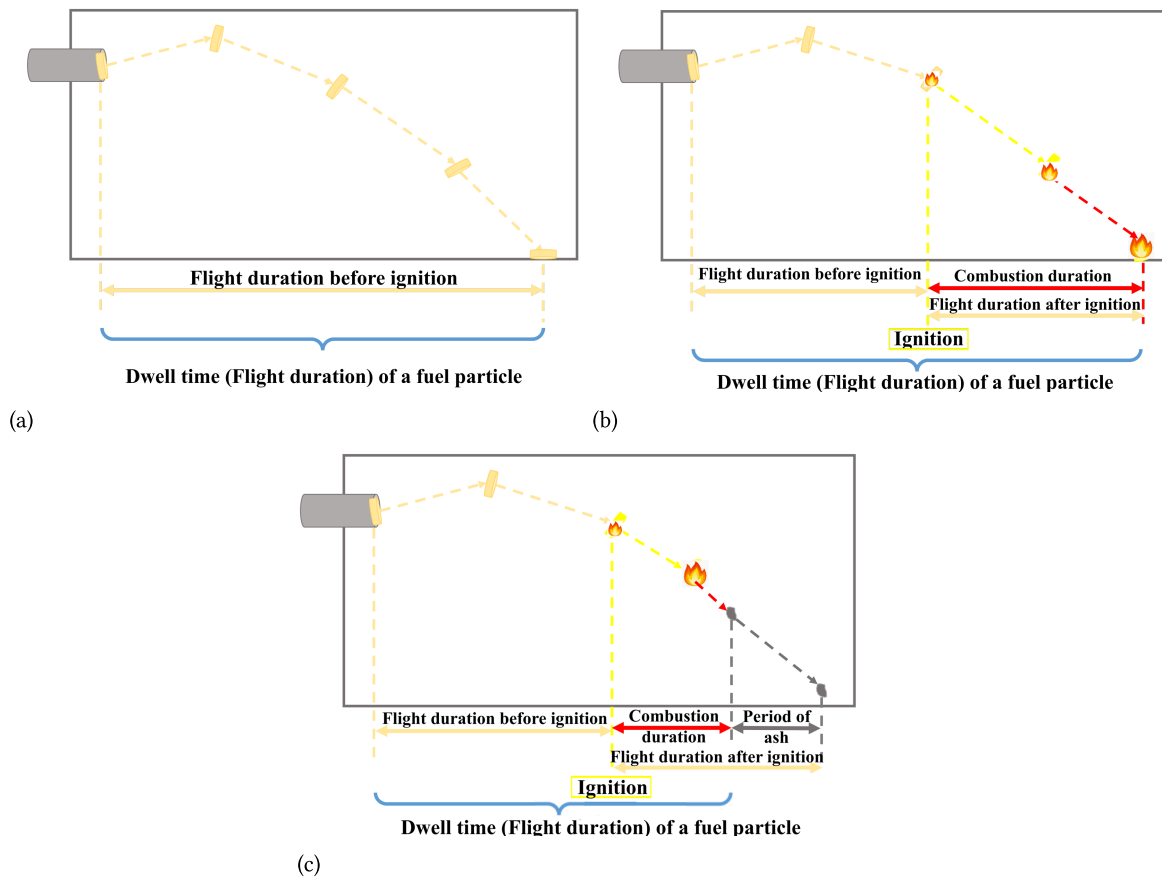


**Figure 5.17.:** Examples of optical ground truth obtained by adding a set of adjacent difference images and a background image of each fraction for the entire sequence [132]. (a) Wood chips. (b) Confetti. (c) Paper shreds. (d) PE granules.

of less interest because the horizontal velocity can be relatively random due to the initial speed, and the vertical motion approximates free fall despite the wind resistance. Therefore, only flight duration (dwell time) and velocity in the depth direction are illustrated.

#### Flight duration (dwell time)

In the beginning, the precise definition of the flight duration (resp. dwell time) in the present study should be highlighted. The flight duration of a specific particle refers to the time interval starting from passing the lance and ending by landing or complete burning out. Given that various particle fractions burn differently, the definition of flight duration ought to be discussed separately for different cases. Figure 5.18 lists three general cases for the flight duration. Figure 5.18 (a) defines the flight duration of a non-burning particle. For non-burning particles, the flight duration represents simply the flight time until landing, which is determined by means of the 3D particle classification demonstrated in Chapter 3.2. With respect to a burning particle, the dwell time terminates at the time of landing or burning out. If the particle lands before burns out, the flight duration ends by landing, as delineated in Figure 5.18 (b). Conversely, if the particle ignites rapidly and vanishes already before landing, the flight duration counts the time from passing the lance to complete burning out (flight duration before ignition + combustion duration), as shown in Figure 5.18 (c).



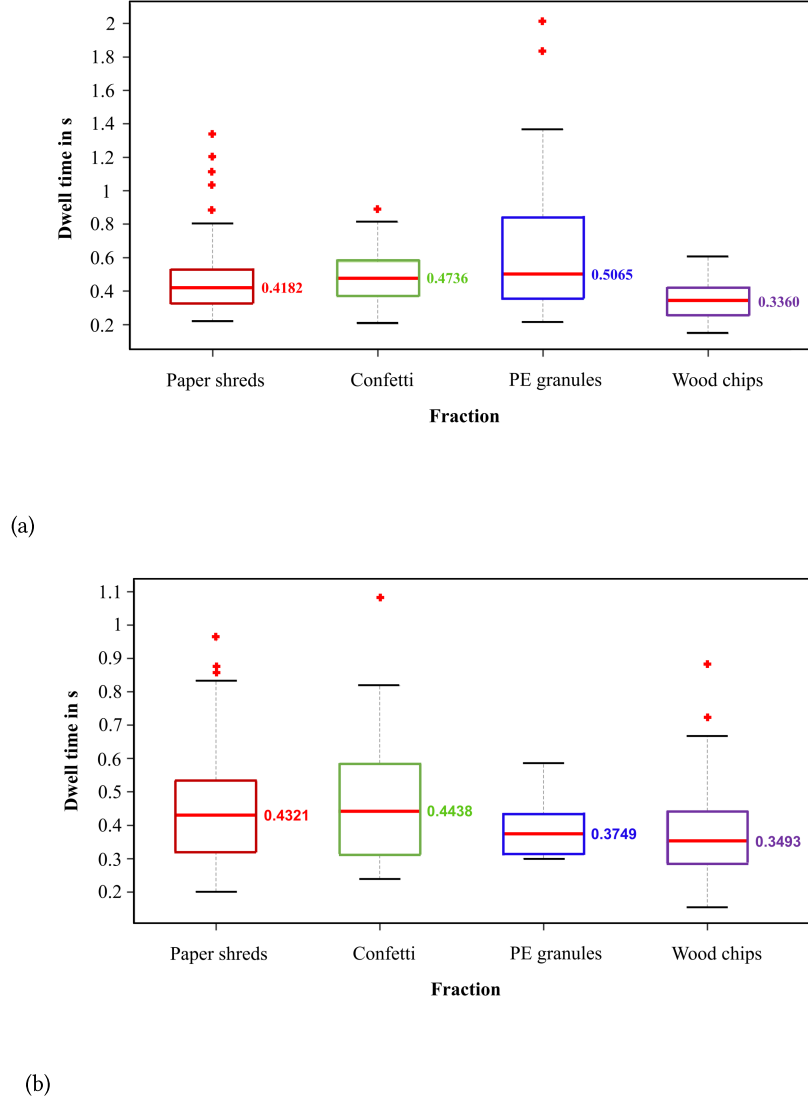
**Figure 5.18.:** Definitions of the dwell time (flight duration) in different situations. (a) Dwell time of non-burning particles. (b) Dwell time of burning particles that do not burn out during the flights. (c) Dwell time of burning particles that burn out during the flights. [132]

Figure 5.19 present the flight durations of the RDF fractions under two air feed pressures by means of box plots. The red line inside each box represents the median flight duration of the corresponding fraction, and the bottom and top of the box indicate the 25th and 75th percentiles, respectively. The whiskers extend to the extreme values that are not considered outliers, while the outliers are shown with the red symbol "+".

Figure 5.19 (a) schematically depicts the dwell time under the air feed pressure of 4.5 bar. Overall, the median particle dwell time is within the range of 0.3 s-0.5 s. Among them, wood chips land the most rapidly on the inner wall. By reason of heavy and concentrated mass, wood chips have the shortest flight duration whose distribution is the most centralized. The flight durations of paper shreds and confetti are similar in terms of median and distribution since they are of the same material and can be ignited during the flight. The combustion causes the carbonization of the paper material and shortens the flight time considerably. Therefore, although PE granules have a more concentrated mass compared to paper, their flight durations are the longest flight time and the most dispersed.

The flight duration of each RDF fraction at 5 bar feed pressure is shown in Figure 5.19 (b). In general, the flight duration of the individual fraction at 5 bar bar differs slightly from the dwell time at 4.5 bar, except for PE granules, whose median flight duration is much shorter with a more concentrated distribution than previous. Therefore, the PE granule is no longer the one with the longest flight duration. The confetti with the second longest dwell time before becomes the particle with the longest flight duration.

The wood chip remains the fattest landing fraction on account of its rapid movement with more spread distribution. Notwithstanding, the flight durations of paper shred and confetti distribute more dispersedly as the result of the irregular motion. Moreover, Figure 5.19 also indicates several outliers with flight durations of more than 1 s, which can be attributed to occasionally occurring complex trajectories of the particles.



**Figure 5.19.:** Box plot of the flight durations of various RDF fractions. (a) Flight durations of the fractions at 4.5 bar air feed pressure. (b) Flight durations of the fractions at 5 bar air feed pressure.

### Space-sliced velocity in the depth direction (z-direction)

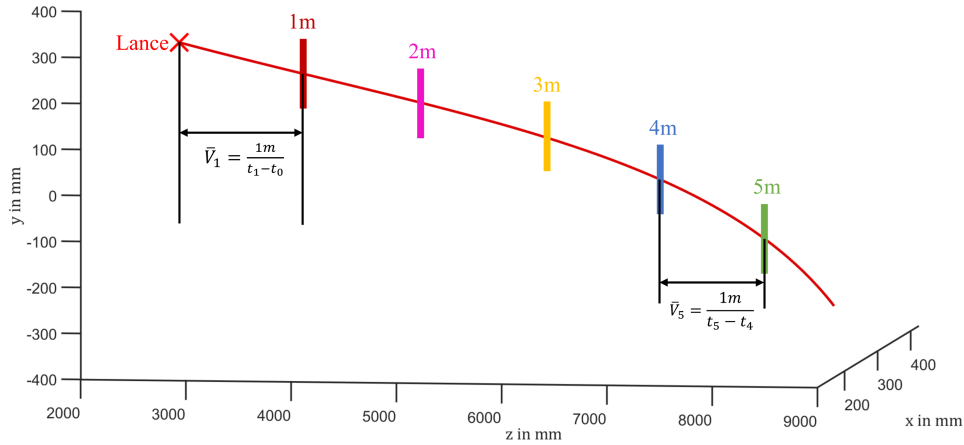
As previously described, the trajectory of each particle is jointly estimated by polynomials in three spatial directions greater than or equal to the second order. Hence, the velocity of the particles in the depth direction is related to the flight distance. In order to deliver practically instructive results, the statistical space-sliced velocity in the depth direction with a distance interval of one meter starting



from the lance is reported. Figure 5.20 schematically depicts the definition of space-sliced velocity, which can be mathematically interpreted as:

$$\bar{V}_n = \frac{1 \text{ m}}{t_n - t_{n-1}}, \quad (5.8)$$

where  $V_n$  denotes the space-sliced velocity that representing the average velocity from  $n - 1$  to  $n$  meter.



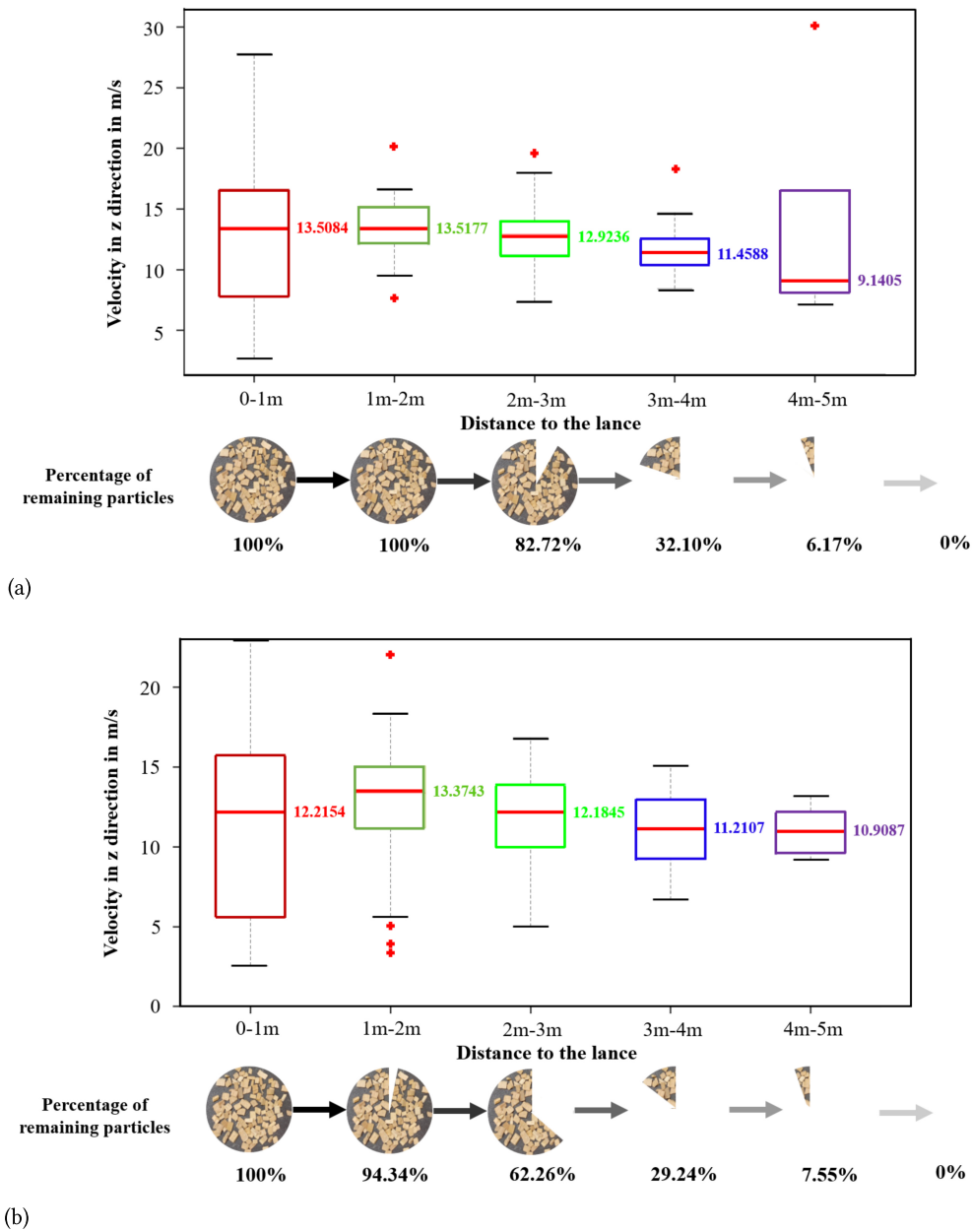
**Figure 5.20.:** Schematically illustration of the space-sliced velocity in the depth direction. The lance is marked by a red cross as previous. [132]

In the equation,  $t_n$  and  $t_{n-1}$  stand for the time when the estimated trajectory in the depth direction reaches  $n$  meter and  $n - 1$  meter away from the lance, respectively. Under the circumstance, the speed is split into the average speed corresponding to each distance interval. In the following, the space-sliced velocities of the RDF fractions are presented by box plots individually. Additionally, the percentage of remaining particles at each distance interval is also presented at the bottom to provide an overview concerning the velocity distribution.

### Wood chip

The space-sliced depth velocity of wood chips at 4.5 bar and 5 bar air feed pressure is presented in Figure 5.21. Under 4.5 bar feed pressure, wood chips can reach a median initial velocity of 13.5 m/s. The median speed subsequently decreases to 9 m/s as the distance to the lance increases. In the beginning, the uncertainty of the initial velocity leads to the dispersedness of the velocity distribution within the first 1 m. Subsequently, the velocity distribution tends to be concentrated. In the end, the velocity starts to distribute scatteredly again because of the small number of remaining particles. With respect to the percentage of wood chips remaining at each distance, all wood chip particles can move beyond 2 m in depth, and the vast majority can reach more than 3 m. More than 30% of wood chips can move up to a distance of 4 m from the lance, and only very few can reach a distance of more than 5 m.

When the feed pressure is increased to 5 bar, the initial median depth velocity of the particles decreases, which then ascends slightly and descends again afterward. Nevertheless, the particles do not really speed up. Since 6% of particles with slow motions have already landed within the first 1 m, the median depth velocity is enhanced. In general, the velocity at 5 bar feed pressure differs insignificantly from 4.5 bar. The speed at 5 bar air pressure is only slightly reduced. Concerning the percentage of remaining particles, all wood chip particles can move more than 1 m in depth and two-thirds of the particles are



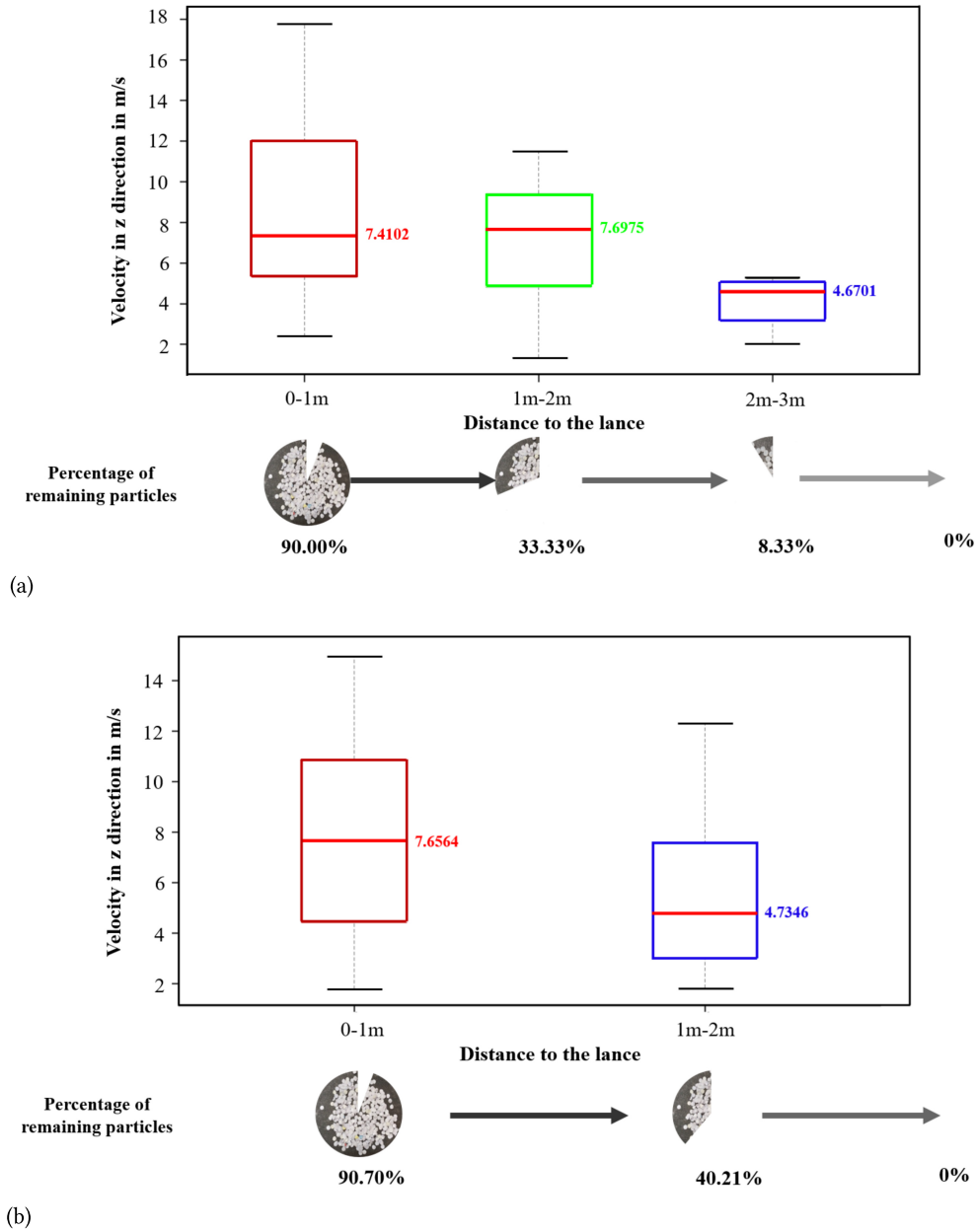
**Figure 5.21.:** Box plot of space-sliced velocity in the depth direction of wood chips. The corresponding percentage of remaining particles at each distance interval is shown on the bottom. (a) Space-sliced velocity at 4.5 bar air feed pressure. (b) Space-sliced velocity at 5 bar air feed pressure.

able to reach a 3 m distance to the lance. The percentage of remaining particles is much lower at a distance of more than 5 m.

### Confetti

Figure 5.22 schematically depicts the space-sliced depth velocity of confetti at 4.5 bar and 5 bar air feed pressure. As presented by the figure, the flight speed and distance of the confetti in the depth direction are much lower than those of the wood chips. At 4.5 bar, the median initial velocity of the confetti is 7.4 m/s, and then the median velocity increases as the slow particles land on the inner wall, while after 2 m from the lance, the velocity decreases significantly to about 4.7 m/s. As for the percentage of

remaining particles, 90% of the confetti can reach up to 1 m in depth, while only around one-third of the particles can continue to move up to 2 m, and less than 10% can reach 3 m in depth. As a consequence of the early ignition of the confetti, the particles do not penetrate far in the depth direction. Therefore, hardly any confetti particles are discovered beyond a distance of 3 m.



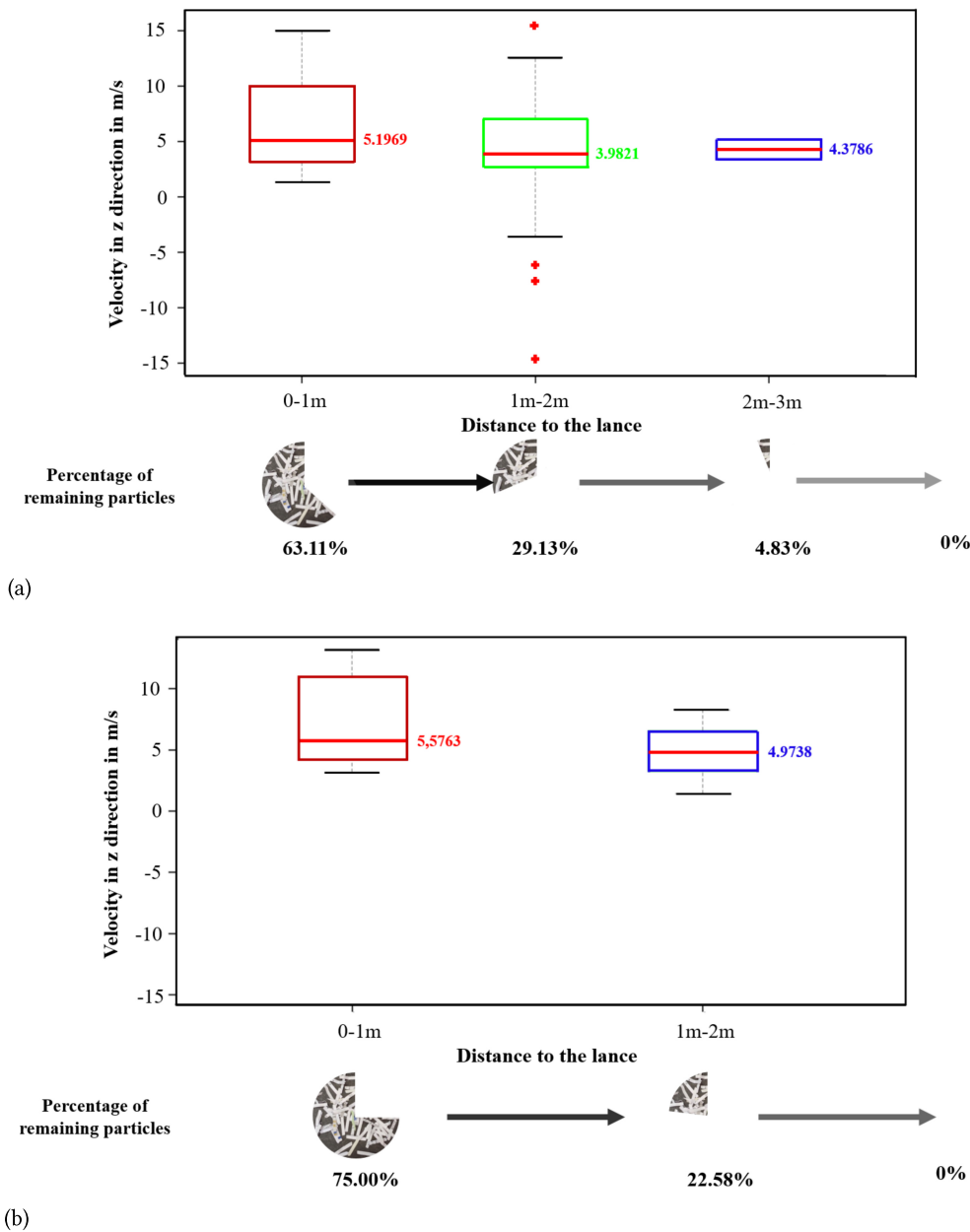
**Figure 5.22.:** Box plot of space-sliced velocity in the depth direction of confetti. The corresponding percentage of remaining particles at each distance interval is shown on the bottom. (a) Space-sliced velocity at 4.5 bar air feed pressure. (b) Space-sliced velocity at 5 bar air feed pressure.

When the feed pressure is raised to 5 bar, the initial depth velocity of the confetti differs slightly concerning median speed and distribution. As illustrated in Figure 5.22 (b), the median velocity can reach approximately 7.5 m/s within a distance of 1 m from the lance. Notwithstanding, the particle velocity decreases considerably with increasing distance. From a distance from 1 m to 2 m to the lance,

the median speed descends to 4.7 m/s, approaching the median velocity between 2 m and 3 m at 4.5 bar air feed pressure. Moreover, as for the percentage of confetti remaining at each distance, beyond 90% of the confetti can move to 1 m from the lance, whereas less than half of the particles are not able to reach a distance of 2 m. No confetti can arrive at 3 m under the air feed pressure of 5 bar.

### Paper shred

Figure 5.23 provides information concerning the space-sliced depth velocity of paper shreds. As revealed by the figure, the paper shred exhibits a slower movement than the confetti despite the identical material.



**Figure 5.23.:** Box plot of space-sliced velocity in the depth direction of paper shreds. The corresponding percentage of remaining particles at each distance interval is shown on the bottom. (a) Space-sliced velocity at 4.5 bar air feed pressure. (b) Space-sliced velocity at 5 bar air feed pressure.

Likewise, paper shreds are also not able to penetrate far in the depth direction. Under the air feed pressure of 4.5 bar, the initial median depth velocity of paper shreds is 5.2 m/s. Subsequently, the median velocity descends to around 4 m/s with a distance of 2 m. In the end, the speed shows a slight increase, but since only less than five percent of the particles remain, the conclusion fails to provide a valid statement with reference to the kinematic properties. The velocity distribution tends to be concentrated as the movement distance increases. Unlike the other fractions, several velocity outliers of the paper shreds are negative, *viz.*, reverse motions towards the lance. The small negative velocities can probably be caused by vortex in the rotary kiln, while relatively large negative velocities might be due to camera inaccuracy and faulty tracking results. With respect to the percentage of remaining particles, less than two-thirds of the particles can move 1 m long in depth direction, and the amount of remaining particles at a distance of more than 2 m decreases sharply.

At 5 bar air feed pressure, the median initial velocity is 5.6 m/s, and then the speed decreases slightly with increasing distance. No paper shred particles are detected at distances greater than 3 m. As a result of the rapid transverse motion, lots of paper shreds touch the side walls before moving far in the depth direction. With respect to the percentage of paper shreds remaining at each distance, exactly 75% paper shreds reach a depth distance of 1 m, and less than 25% could move forward to 2 m. Paper shreds with more than 2 m motions from the lance are not discovered.

### PE granule

Figure 5.24 schematically illustrates the space-sliced depth velocity of PE granules at 4.5 bar and 5 bar air feed pressure. Compared to the previous two paper fractions, PE granules move significantly more rapidly and further in the depth direction. At 4.5 bar air feed pressure, the initial median velocity reaches over 8 m/s. Because particles with extraordinarily low speed fail to penetrate further than 2 m in the depth direction, the speed increases with a distance beyond 2 m. This can also be reflected by the percentage of remaining particles. More than 95% of particles are capable of penetrating 2 m in the depth direction, whereas less than half of PE granules can move to 3 m. At last, barely 4% PE granules could arrive at 5 m.

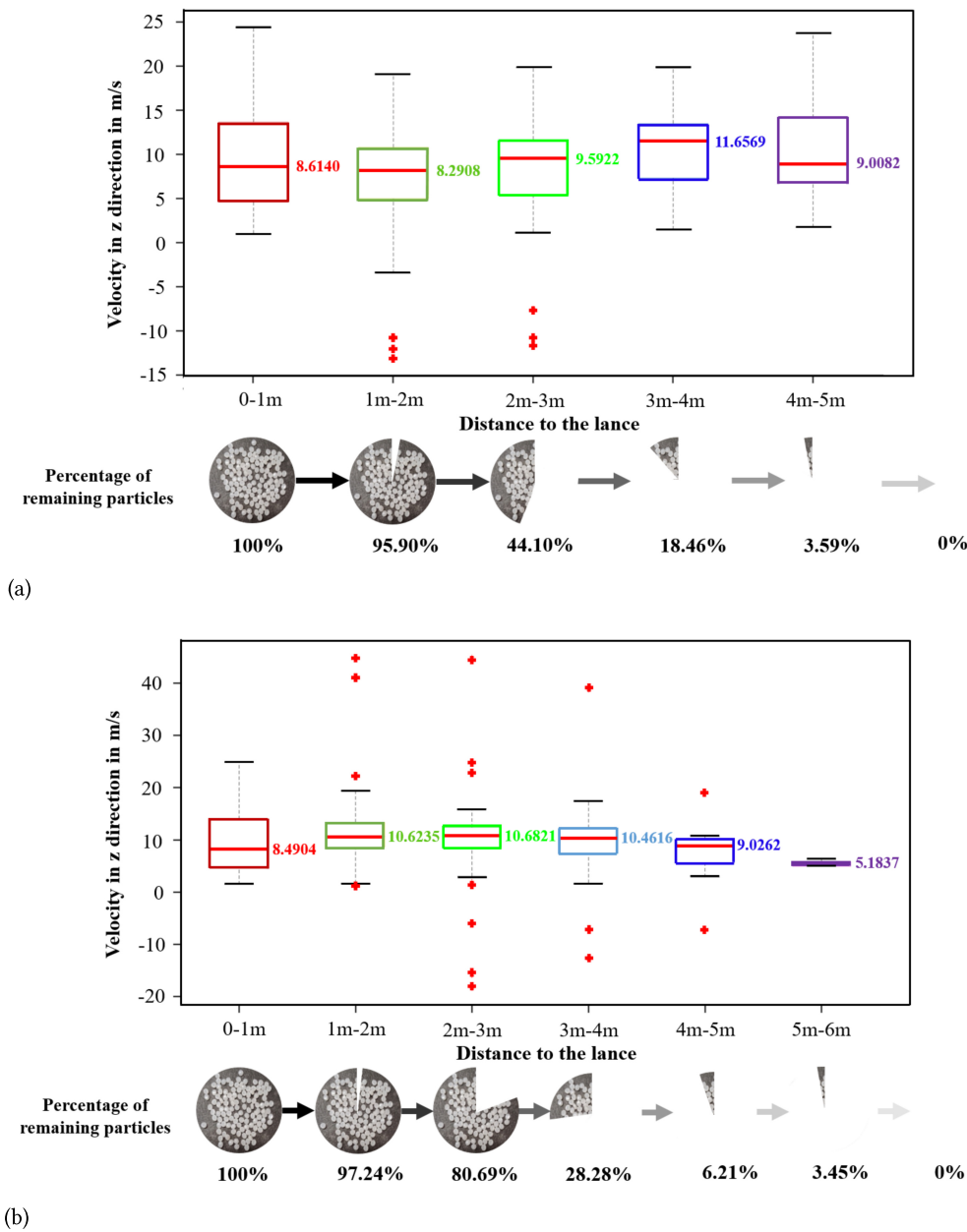
When the feed pressure is increased to 5 bar, the initial velocity of PE granules decreases slightly in comparison to the speed at 4.5 bar. From 1 m, the median speed gradually increases to around 10 m/s. Over the 4 m distance, the median velocity descends again. Regarding the percentage of remaining particles, almost all PE granules can reach 2 m, and the vast majority can move more than 3 m in the depth direction. Few PE granule particles can also be detected at a distance exceeding 5 m with extremely low speed.

### 5.3.2. Combustion Behaviors of RDF Particles

This section summarizes the combustion behaviors of various RDF fractions, *i.e.*, the ignition time. As Figure 5.17 reveals, wood chips and PE granules are not able to be ignited by the heat inside the rotary kiln during their flight. Therefore, the study only delves into the ignition time of the confetti and paper shreds hereafter. The ignition time can be inferred from the change in the gray value of each estimated 3D trajectory. If the gray value increases abruptly, the particles can be assumed ignited. In the following, the ignition times under the 4.5 bar and 5 bar air feed pressures are shown in 0.1 s intervals in bar charts.

#### Confetti

Figure 5.25 presents the ignition time of confetti at 4.5 bar and 5 bar air feed pressure. Under the air feed pressure of 4.5 bar, only 26% of the confetti particles combust. Among them, 3% of the confetti

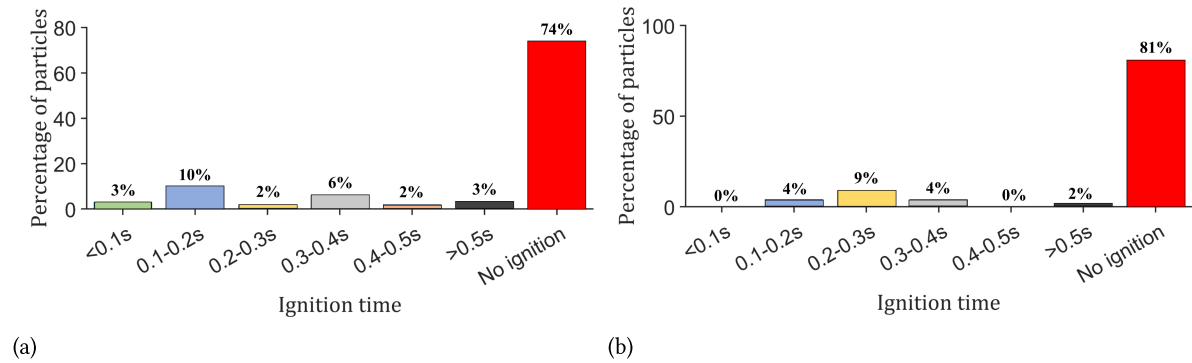


**Figure 5.24.:** Box plot of space-sliced velocity in the depth direction of PE granules. The corresponding percentage of remaining particles at each distance interval is shown on the bottom. (a) Space-sliced velocity at 4.5 bar air feed pressure. (b) Space-sliced velocity at 5 bar air feed pressure.

is ignited within 0.1 s. Half of the burning confetti combusts in the first 0.2 s. As the feed pressure increases to 5 bar, the percentage of burning particles descends. Only less than 20% of confetti can be ignited during their flights, and the ignition time is also delayed. No confetti ignites within 0.1 s after being conveyed from the lance. The majority of burning confetti combusts between 0.2 s and 0.3 s.

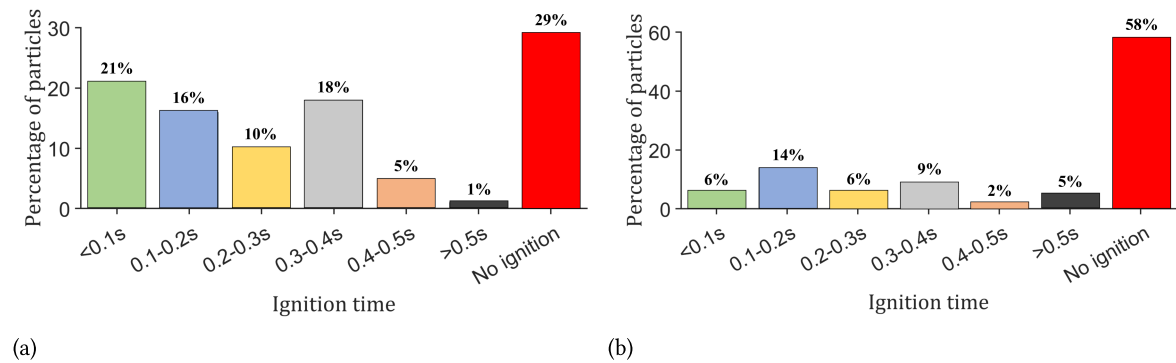
### Paper shred

Figure 5.26 gives an overview of the ignition time of paper shreds at 4.5 bar and 5 bar air feed pressure. In comparison to the confetti, paper shreds are more easily ignited with shorter ignition time and higher combustion rate. At 4.5 bar air pressure, 71% paper shreds combust during their flight, and 21%



**Figure 5.25.:** Bar plot of ignition time of confetti. (a) Ignition time at 4.5 bar air feed pressure. (b) Ignition time at 5 bar air feed pressure.

are ignited within 0.1 s. More than half of the paper shreds that can be ignited by the heat combust within 0.2 s. The vast majority of paper shreds combust within 0.4 s. Similar to confetti, paper shreds are harder to ignite when the feed pressure is raised to 5 bar, which can be reflected by the reduced combustion rate and the extended ignition time. At 5 bar feed pressure, only around 40% of the paper shreds can be ignited. Moreover, only 6% of the paper shreds are ignited within 0.1 s. Overall, paper shreds have a higher ignition rate and a shorter ignition time than confetti, which might be caused by the sizeable surface area of paper shreds resulting in more heat absorption.



**Figure 5.26.:** Bar plot of ignition time of paper shreds. (a) Ignition time at 4.5 bar air feed pressure. (b) Ignition time at 5 bar air feed pressure.

## 6. Conclusion and outlook

RDFs are increasingly favored by modern industry as a consequence of their economical as well as carbon-neutral properties. However, the complexity of their components gives rise to difficulty in predicting their properties, which further negatively impacts controllable combustion. In the present dissertation, a new measurement system is developed and evaluated that uses a plenoptic camera to observe the flight and combustion processes of RDF particles in a rotary kiln. Therefore, computer vision technology and algorithms were developed and adapted in this work. This chapter summarizes the vital findings and contributions.

### 6.1. Conclusion and overview

In the present dissertation, a new measurement system is developed. The study elaborates on a novel plenoptic camera based measurement system to investigate the flight and combustion behaviors of various RDF fractions. The plenoptic camera mounted outside the rotary kiln provides information concerning spatial particle positions and the gray value distribution inside the kiln. Under real conditions in a combustion chamber or a rotary kiln, the plenoptic camera provides only strongly fluctuating depth information due to unfavorable exposure conditions. The captured information is processed by the introduced approaches for multiple particle tracking-by-detection afterward, primarily including a novel combined detection approach, an extended 2.5D tracking algorithm, and a post-processing framework for multiple particle tracking.

Chapter 1 introduces the motivation and outline of the presented work. In Chapter 2, the experimental setup, including the test facility BRENDA and the applied plenoptic camera system, and state of the art concerning image processing techniques are described. Chapter 3 elaborates on the proposed novel combined multiple particle detection approach, while Chapter 4 presents the extended 2.5D particle tracking algorithm and a novel post-processing framework. In Chapter 5, the results of particle tracking-by-detection are shown. In addition, the flight and combustion behaviors of the experimented RDF fractions are discussed.

The applied plenoptic camera is able to provide 2D gray value and 3D point cloud information. Based on the 2D gray value images, several detection approaches, for instance, the OTSU threshold selection method and the SIFT algorithm, could identify objects in accordance with local gray value distribution. Meanwhile, particles can also be detected by forming clusters within 3D point clouds using clustering approaches, such as the DBSCAN and the QCC algorithms. Given that both types of methods apply only a specific part of acquired information, resulting in several particular deficiencies in their detection performances, *e.g.*, 2D gray value based methods are unable to discriminate positive detections caused by artifacts, whereas 3D-clustering approaches fail to distinguish particles close to each other. Under this circumstance, a novel combined detection method that combines the detection results of a 2D gray value-based detection approach and a 3D-clustering method providing superior performance is proposed. By virtue of the combined approach, the detection issues of each method are addressed. In this work, various methods, namely 2D-OTSU, 2D-SIFT, 3D-DBSCAN, 3D-MO, and 3D-QCC, together



with their combinations, are tested on the same set of ground truth. Among them, the 3D clustering method performs at the bottom in terms of the  $F_1$ -score due to the excessive false positive detection rate. Compared to the 3D clustering approaches, 2D gray value based methods could improve the  $F_1$ -score (value range [0,1]) by 0.2 as a result of the excellent precision. The proposed combined method is the champion concerning the detection performance with both satisfactory precision and recall. In particular, the combination of 2D-SIFT and 3D-DBSCAN performs the best. Overall, the combined approach outperformed the corresponding individual methods. Compared to the 3D clustering methods, the  $F_1$ -score of the combined method is improved by about 0.3, while compared to the 2D approaches, the  $F_1$ -score of the combined method is enhanced by 0.03-0.05.

Subsequently, the novel 2.5D linear Kalman filter, together with GNN and JPDA, are utilized respectively to associate temporary particle detections for the sake of particle tracking, where the depth information imposes constraints on the potential association candidates within the 2D elliptical gating area. Despite the optimized tracking algorithm, tracking inaccuracies occur in the preliminary tracking results caused by detection errors. To alleviate the issues, a post-processing framework is developed, which takes full advantage of the motion similarity between tracklets to connect and merge tracklets to trajectories and fuse trajectories afterward. The presented post-processing framework eliminates at first false tracklets, and the outliers inside the remaining tracklets are pruned afterward. Subsequently, the post-processing framework connects the tracklets by means of the introduced dual nearest neighbor approach. For complex cases, such as with the presence of lots of duplications, a last step of trajectories fusion is essential to avoid redundant trajectories. The proposed post-processing framework for multiple-particle tracking is applied to GNN and JPDA tracking results, which significantly outperform the standard tracking methods, as proved by the quantitative assessment. As demonstrated before, JPDA is not appropriate to follow a substantial number of detections for an extended period. The particles from the whole sequence with roughly 3000 frames are thereby tracked barely by GNN. The acquired 2D trajectories are converted into 3D in accordance with the particle spatial coordinates provided by the plenoptic camera afterward. For the purpose of compensating for the fluctuations of the straightforwardly converted trajectories, the spatial trajectories are then estimated by polynomials under the condition that all particles must be conveyed through the lance. Each particle trajectory is estimated by three polynomials over time in the x,y, and z-direction, with all three polynomials simultaneously intersecting at the lance position (a particular spatial point). Thus the exact time when the particle leaves the lance can be determined, even if the particle is not visible right after leaving the lance.

To evaluate the newly developed measurement system, the estimated polynomials are used for a statistical analysis of the flight and combustion behaviors of various RDF particles, *viz.*, wood chip, confetti, paper shred, and PE granule, where the flight behaviors contain the flight duration (dwell time) and the velocity in the depth direction, and the combustion behavior refers to the ignition time. The flight duration stands for the time interval starting from passing through the lance and ending by landing or complete burning out. The speed is presented as space-sliced velocity with a 1 m interval beginning from the lance. The flight properties can be inferred by the polynomials, which are functions of time in three spatial directions. By observing the brightness of the trajectories, the combustion behavior can also be assessed by the proposed novel measurement system.

Finally, these obtained properties were compared with the CFD model (Chapter A.2), which shows a sufficient coherency of the two outcomes [109].

## 6.2. Future work

The experiments with various fuel particles indicate the adequacy of the proposed novel plenoptic camera based measurement system for the preliminary investigation of RDFs' flight and combustion properties. Future research could aim at investigating the adaptability and robustness of the presented approaches for the detection and tracking of other macroscopic and microscopic objects and developing an integrated tool with optimized visualization and user interaction for multiple object tracking-by-detection. The tracking-by-detection tool ought to integrate different 2D gray value based approaches and 3D clustering algorithms to adapt to various use cases. In addition, other measurement systems, such as the stereo camera system and ToF camera, could also be applied to identical situations for a comprehensive comparison of various camera technologies to instruct further experiments and investigations.

The study elaborates on the investigation into various RDF fractions' dwell time, space-sliced velocity in the depth direction, and ignition time. Further study could explore the fuel behaviors in conjunction with their physical and chemical characteristics, for instance, the heat value and distillation temperature, to extendedly and interdisciplinarily explain the reasons behind the behaviors.



## Bibliography

- [1] Abfallprognosen und Mengenstromszenarien. Technical report, bvse-Fachverband für Ersatzbrennstoffe Altholz und biogene Abfälle, Bonn, Germany, 2017.
- [2] Depth camerras and RGB-D camera SLAM. Technical report, Kudan Inc., Bristol, UK, 2020.
- [3] Environmental Data of the German Cement Industry 2020. Technical report, Verein der Deutschen Zementindustrie, Düsseldorf, Germany, 2021.
- [4] E. Adelson and J. Bergen. The plenoptic function and the elements of early vision. In *Computational Models of Visual Processing*, pages 3–20. MIT Press, Cambridge, MA, USA, 1997.
- [5] M. Ahmad, I. Ahmed, F. A. Khan, F. Qayum, and H. Aljuaid. Convolutional neural network-based person tracking using overhead views. *International Journal of Distributed Sensor Networks*, 16(6), 2020.
- [6] K. Aleksandrov, H.-J. Gehrmann, M. Hauser, H. Mätzing, D. Pigeon, D. Stapf, and M. Wexler. Waste incineration of Polytetrafluoroethylene (PTFE) to evaluate potential formation of per- and poly-fluorinated alkyl substances (PFAS) in flue gas. *Chemosphere*, 226, Apr. 2019.
- [7] M. Ankerst, M. Breunig, H.-P. Kriegel, and J. Sander. OPTICS: Ordering points to identify the clustering structure. In *Proc. ACM Sigmod Record International Conference on Management of Data*, pages 49–60, Philadelphia PA, USA, June 1999.
- [8] M. Attamimi, T. Nagai, and D. Purwanto. Object detection based on particle filter and integration of multiple features. *Procedia Computer Science*, 144:214–218, Jan. 2018.
- [9] S.-H. Bae and K.-J. Yoon. Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning. In *Proc. the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, June 2014.
- [10] Y. Bar-Shalom and E. Tse. Tracking in a cluttered environment with probabilistic data association. *Automatica*, 11(5):451–460, Sept. 1975.
- [11] M. Beckmann and S. Ncube. Characterisation of refuse derived fuels (RDF) in reference to the fuel technical properties. In *Proc. the International Conference on Incineration and Thermal Treatment Technologies*, Phoenix, USA, 2007.
- [12] M. Brown and D. Lowe. Invariant features from interest point groups. In *Proc. BMVC 2002: 13th British Machine Vision Conference*, pages 253–262, Cardiff, Wales, Sept. 2002.
- [13] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(8):790–799, 1995.
- [14] N. Chenouard, I. Smal, F. Chaumont, M. Maška, I. Sbalzarini, Y. Gong, J. Cardinale, C. Carthel, S. Coraluppi, M. Winter, A. Cohen, W. Godinez, K. Rohr, Y. Kalaidzidis, L. Liang, J. Duncan, H. Shen, Y. Xu, K. Magnusson, and E. Meijering. Objective comparison of particle tracking methods. *Nature methods*, 11:281–289, Jan. 2014.

- [15] Y. H. Chiang, C. M. Hsu, and A. Tsai. Fast multi-resolution spatial clustering for 3D point cloud data. In *Proc. 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 1678–1683, Bari, Italy, Oct. 2019.
- [16] D. Chicco, M. J. Warrens, and G. Jurman. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Computer Science*, 7:e623, July 2021.
- [17] 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. In *Proc. 2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4846–4855, Venice, Italy, Oct. 2017.
- [18] T. F. Coleman and Y. Li. An interior trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, 6(2):418–445, 1996.
- [19] E. V. Cuevas, D. Zaldivar, and R. Rojas. Kalman filter for vision tracking. Technical report, Freie Universität Berlin, Institut für Informatik, Berlin, Germany, Aug. 2005.
- [20] P. Danz, T. Marzi, A. Mrotzek, L. D. Matteo, and G. Marotz. Flight and burnout behaviour of refuse derived fuels in cement rotary kilns. In *Proc. Abfallforschungstage 2008*, Hannover, Germany, June 2008.
- [21] S. Dash and U. R. Jena. Multi-resolution laws’ masks based texture classification. *Journal of Applied Research and Technology*, 15(6):571–582, 2017.
- [22] A. de Myttenaere, B. Golden, B. Le Grand, and F. Rossi. Mean absolute percentage error for regression models. *Neurocomputing*, 192:38–48, 2016.
- [23] B. Dou, S. Park, S. Lim, T.-U. Yu, and J. Hwang. Pyrolysis characteristics of refuse derived fuel in a pilot-scale unit. *Energy & Fuels*, 21(6):3730–3734, Nov. 2007.
- [24] F. Duan, J. Liu, C.-S. Chyang, C.-H. Hu, and J. Tso. Combustion behavior and pollutant emission characteristics of RDF (refuse derived fuel) and sawdust in a vortexing fluidized bed combustor. *Energy*, 57(C):421–426, 2013.
- [25] S. Edelman, N. Intrator, and T. Poggio. Complex cells and object recognition. In *Proc. 1997 conference on Advances in neural information processing systems*, Denver Colorado, USA, June 1997.
- [26] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. 2. International Conference on Knowledge Discovery and Data Mining*, pages 226–231, Portland, OR, USA, Aug. 1996.
- [27] J. Fan, W. Xu, Y. Wu, and Y. Gong. Human tracking using convolutional neural networks. *IEEE Transactions on Neural Networks*, 21(10):1610–1623, 2010.
- [28] M. Faulstich, S. Vodegel, E. Fediana, M. Franke, P. Degener, J. Aigner, and K. Reh. *Umweltschutzgerechte Verwertung nicht etablierter Stoffströme in Abfallverbrennungsanlagen*. Weltbundesamt, Dessau-Roßlau, Germany, 2016.
- [29] T. Fortmann, Y. Bar-Shalom, and M. Scheffe. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE Journal of Oceanic Engineering*, 8(3):173–184, 1983.
- [30] T. Georgiev, A. Lumsdaine, and S. Goma. Plenoptic principal planes. In *Computational Optical Sensing and Imaging, Proc. Imaging and Applied Optics*, Toronto, Canada, July 2011.

- 
- [31] R. Girshick. Fast R-CNN. In *Proc. 2015 IEEE International Conference on Computer Vision (ICCV)*, NW Washington, DC, USA, Dec. 2015.
  - [32] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proc. 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Columbus, OH, USA, June 2014.
  - [33] M. Grinberg. *Feature-Based Probabilistic Data Association for Video-Based Multi-Object Tracking*. PhD thesis, Karlsruhe Institute of Technology, Karlsruhe, Germany, 2018.
  - [34] I. Halima, J.-M. Laferté, G. Cormier, A.-J. Fougères, and J.-L. Dillenseger. Depth and thermal information fusion for head tracking using particle filter in a fall detection context. *Integrated Computer-Aided Engineering*, 27:1–14, Jan. 2020.
  - [35] E. M. Hall, B. S. Thurow, and D. R. Guildenbecher. Comparison of three-dimensional particle tracking and sizing using plenoptic imaging and digital in-line holography. *Appl. Opt.*, 55(23):6410–6420, Aug. 2016.
  - [36] J. A. Hartigan and M. A. Wong. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.
  - [37] X. He. *Entwicklung und Analyse von Verfahren zur Hintergrundbildberechnung für die Partikeldetektion*. Master’s thesis, Karlsruhe Institute of Technology, Karlsruhe, Germany, 2022.
  - [38] C. Heinze, S. Spyropoulos, S. Hussmann, and C. Perwass. Automated robust metric calibration algorithm for multifocus plenoptic cameras. *IEEE Transactions on Instrumentation and Measurement*, 65(5):1197–1205, 2016.
  - [39] J. Huang, Q. Zhu, L. Yang, D. Cheng, and Q. Wu. QCC: a novel clustering algorithm based on quasi-cluster centers. *Machine Learning*, 106(3):337–357, Jan. 2017.
  - [40] R. J. Hyndman and A. B. Koehler. Another look at measures of forecast accuracy. *International Journal of Forecasting*, 22(4):679–688, 2006.
  - [41] H. Irshad, A. Veillard, L. Roux, and D. Racoceanu. Methods for nucleidetection, segmentation, and classification in digital histopathology: Abreview-current status and future potential. *IEEE Reviews in Biomedical Engineering*, 7:97–114, 2014.
  - [42] O. H. Jafari, D. Mitzel, and B. Leibe. Real-time RGB-D based people detection and tracking for mobile robots and head-worn cameras. In *Proc. 2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5636–5643, Hong Kong, China, June 2014.
  - [43] S. Julier and J. Uhlmann. A new extension of the kalman filter to nonlinear systems. In *Proc. Signal Processing, Sensor Fusion, and Target Recognition VI, 1997*, pages 54–65, Orlando, Florida, USA, July 1997.
  - [44] D. Justen. *Untersuchung eines neuartigen 2D- gestützten 3D-PMD Bildverarbeitungssystems*. PhD thesis, University Siegen, Siegen, Germany, 2001.
  - [45] R. Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82:35–45, Jan. 1960.
  - [46] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, pages 321–331, 1988.
  - [47] T. Kirubarajan and Y. Bar-Shalom. Probabilistic data association techniques for target tracking in clutter. *Proceedings of the IEEE*, 92(3):536–557, 2004.

- [48] E. M. Knorr, R. T. Ng, and V. Tucakov. Distance-based outliers: algorithms and applications. *The International Journal on Very Large Data Bases*, 8(3-4):237–253, Feb. 2000.
- [49] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proc. the Fourteenth International Joint Conference on Artificial Intelligence*, volume 2, pages 1137–1143, Montreal, Quebec, Canada, Aug. 1995.
- [50] P. Konstantinova, A. Udvarrev, and T. Semerdjiev. A study of a target tracking algorithm using global nearest neighbor approach. In *Proc. 4th International Conference on Computer Systems and Technologies - CompSysTech'2003*, pages 290–295, Rousse, Bulgaria, June 2003.
- [51] H. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics*, pages 83–97, Mar. 1955.
- [52] J. Kulchandani and K. Dangarwala. Moving object detection: Review of recent research trends. In *Proc. 2015 International Conference on Pervasive Computing (ICPC)*, pages 843–847, Pune, India, Jan. 2015.
- [53] A. Kulkarni, D. Chong, and F. A. Batarseh. 5 - foundations of data imbalance and solutions for a data democracy. In F. A. Batarseh and R. Yang, editors, *Data Democracy*, pages 83–106. Academic Press, Cambridge, MA, USA, 2020.
- [54] R. Kvyetnyy, S. Olga, A. Olesenko, P. Komada, J. Sikora, A. Kalizhanova, and S. Smailova. Method of image texture segmentation using Laws' energy measures. In R. S. Romaniuk and M. Linczuk, editors, *Proc. Photonics Applications in Astronomy, Communications, Industry, and High-Energy Physics Experiments 2017*, volume 10445, Wilga, Poland, June 2017. SPIE.
- [55] K. R. Lata, P. Penczek, and J. Frank. Automatic particle picking from electron micrographs. *Ultramicroscopy*, 58(3):381–391, June 1995.
- [56] K. I. Laws. Texture energy measures. In *Proc. Image Understanding Workshop*, pages 47–51, Los Angeles, California, USA, Nov. 1979.
- [57] J.-G. Lee, J. Han, and X. Li. Trajectory outlier detection: A partition-and-detect framework. In *Proc. 2008 IEEE 24th International Conference on Data Engineering*, pages 140–149, Cancun, Mexico, Apr. 2008.
- [58] M. Levoy and P. Hanrahan. Light field rendering. In *Proc. SIGGRAPH96: 23rd International Conference on Computer Graphics and Interactive Techniques*, New Orleans, Louisiana, USA, Aug. 2002.
- [59] L. Li. Time-of-Flight Camera - An Introduction. Technical report, Texas Instruments, Dallas, Texas, USA, 2014.
- [60] Q. Li, R. Li, K. Ji, and D. Wei. Kalman filter and its application. In *Proc. 8th International Conference on Intelligent Networks and Intelligent Systems*, Tianjin, China, Nov. 2015.
- [61] X. Li, J. Han, S. Kim, and H. Gonzalez. ROAM: Rule- and motif-based anomaly detection in massive moving object data sets. In *Proc. 2007 SIAM International Conference on Data Mining*, pages 273–284, Minneapolis, MN, USA, Apr. 2007. Society for Industrial and Applied Mathematics.
- [62] B. Liedmann, W. Arnold, B. Krüger, A. Becker, S. Krusch, S. Wirtz, and V. Scherer. An approach to model the thermal conversion and flight behaviour of refuse derived fuel. *Fuel*, 200:252–271, July 2017.

- 
- [63] T. Lindeberg. Scale-space theory: A basic tool for analysing structures at different scales. *Journal of Applied Statistics*, 21:224–270, Sept. 1994.
- [64] H. Liu, J. Luo, P. Wu, S. Xie, and H. Li. People detection and tracking using RGB-D cameras for mobile robots. *International Journal of Advanced Robotic Systems*, 13(5), 2016.
- [65] D. Lowe. Object recognition from local scale-invariant features. In *Proc. 7th IEEE International Conference on Computer Vision*, volume 2, Kerkyra, Greece, Sept. 1999.
- [66] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, Feb. 2004.
- [67] A. Lumsdaine and T. Georgiev. Full resolution lightfield rendering. Technical report, Adobe Systems, San Jose, CA, USA, Jan. 2008.
- [68] A. Lumsdaine and T. Georgiev. The focused plenoptic camera. In *Proc. 2009 IEEE International Conference on Computational Photography (ICCP)*, pages 1–8, San Francisco, California, USA, Apr. 2009.
- [69] C. Ma, C. Yang, F. Yang, Y. Zhuang, Z. Zhang, H. Jia, and X. Xie. Trajectory factory: Tracklet cleaving and re-connection by deep siamese Bi-GRU for multiple object tracking. In *Proc. 2018 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, San Diego, CA, USA, July 2018.
- [70] R. D. Maesschalck, D. Jouan-Rimbaud, and D. Massart. The Mahalanobis distance. *Chemometrics and Intelligent Laboratory Systems*, 50(1):1–18, Jan. 2000.
- [71] R. Marsh, A. Griffiths, K. Williams, and S. Wilcox. Physical and thermal properties of extruded refuse derived fuel. *Fuel Processing Technology*, 88:701–706, July 2007.
- [72] T. Marzi, P. Danz, A. Mrotzek, L. di Matteo, and G. Marotz. Flight characteristics and combustion behaviour of refused derived fuels (RDF) in a cement kiln. In *Proc. Second International Symposium on Energy from Biomass and Waste*, Venice, Italy, Nov. 2008.
- [73] J. Matthes, J. Hock, P. Waibel, A. Scherrmann, H.-J. Gehrman, and H. Keller. A high-speed camera based approach for the on-line analysis of particles in multi-fuel burner flames. *Experimental Thermal and Fluid Science*, 73, Aug. 2015.
- [74] J. J. Merelo, A. Prieto, F. Moran, R. Marabini, and J. M. Carazo. Automatic classification of biological particles from electron-microscopy images using conventional and genetic-algorithm optimized learning vector quantization. *Neural Process. Lett.*, page 55–65, 1998.
- [75] A. Milan, S. H. Rezatofighi, A. Dick, I. Reid, and K. Schindler. Online multi-target tracking using recurrent neural networks. In *Proc. the Thirty-First AAAI Conference on Artificial Intelligence (AAAI-17)*, San Francisco, California USA, 2017.
- [76] P. C. Miller and R. S. Caprari. Demonstration of improved automatic target-recognition performance by moment analysis of correlation peaks. *Appl. Opt.*, 38(8):1325–1331, 1999.
- [77] D. Musicki and R. Evans. Joint integrated probabilistic data association: JIPDA. *IEEE Transactions on Aerospace and Electronic Systems*, 40(3):1093–1099, 2004.
- [78] D. Musicki, R. Evans, and S. Stankovic. Integrated probabilistic data association. *IEEE Transactions on Automatic Control*, 39(6):1237–1241, 1994.



- [79] M. Nakhaei, M. N. Pedersen, H. Wu, L. S. Jensen, P. Glarborg, P. A. Jensen, D. Grévain, and K. Dam-Johansen. Aerodynamic and physical characterization of refuse derived fuel. *Energy & Fuels*, 32(7):7685–7700, May 2018.
- [80] R. Ng. *Digital Light Field Photography*. Phd thesis, Stanford University, Stanford, California, USA, 2006.
- [81] W. V. Nicholson and R. M. Glaeser. Review: Automatic particle detection in electron microscopy. *Journal of Structural Biology*, 133(2):90–101, 2001.
- [82] I. Niskanen, M. Immonen, L. Hallman, G. Yamamuchi, M. Mikkonen, T. Hashimoto, Y. Nitta, P. Keränen, J. Kostamovaara, and R. Heikkilä. Time-of-flight sensor for getting shape model of automobiles toward digital 3D imaging approach of autonomous driving. *Automation in Construction*, 121, Jan. 2021.
- [83] A. Nordsjo. Target tracking based on Kalman-type filters combined with recursive estimation of model disturbances. In *IEEE International Radar Conference, 2005.*, pages 115–120, Arlington, Virginia, May 2005.
- [84] L. K. Nørskov. *Combustion of solid alternative fuels in the cement kiln burner*. PhD thesis, Technical University of Denmark, Kongens Lyngby, Denmark, 2012.
- [85] K. Ohmi and H.-Y. Li. Particle-tracking velocimetry with new algorithms. *Measurement Science and Technology*, 11(6):603–616, May 2000.
- [86] T. Ophoff, K. Van Beeck, and T. Goedemé. Exploring RGB+Depth fusion for real-time object detection. *Sensors*, 19:866, Feb. 2019.
- [87] R. Orozco, C. Loscos, I. Martin, and A. Artusi. Chapter 4 - multiview HDR video sequence generation. In F. Dufaux, P. Le Callet, R. K. Mantiuk, and M. Mrazek, editors, *High Dynamic Range Video*, pages 121–138. Academic Press, Cambridge, MA, USA, 2016.
- [88] E. Ostertagová. Modelling using polynomial regression. *Procedia Engineering*, 48:500–506, 2012.
- [89] N. Otsu. A threshold selection method from grey level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, pages 62–66, Sept. 1979.
- [90] M. N. Pedersen, M. Nielsen, S. Clausen, P. Jensen, L. Jensen, and K. Dam-Johansen. Imaging of flames in cement kilns to study the influence of different fuel types. *Energy & Fuels*, (31):11424–11439, 2017.
- [91] J. Peng, T. Wang, W. Lin, J. Wang, J. See, S. Wen, and E. Ding. TPM: Multiple object tracking with tracklet-plane matching. *Pattern Recognition*, 107:252–271, Nov. 2020.
- [92] C. Perwass and L. Wietzke. Single lens 3D-camera with extended depth-of-field. In *Proc. SPIE 8291, Human Vision and Electronic Imaging XVII*, volume 8291, Burlingame, CA, USA, Feb. 2012. SPIE.
- [93] M. Piccardi. Background subtraction techniques: A review. *IEEE Trans Syst Man Cybern*, 4:3099 – 3104 vol.4, Nov. 2004.
- [94] D. M. W. Powers. Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. *Journal of Machine Learning Technologies* 2 (1), pages 37–63, 2011.
- [95] M. Raffel, C. E. Willert, F. Scarano, C. J. Kähler, S. T. Wereley, and J. Kompenhans. *Stereoscopic PIV*, pages 285–307. Springer International Publishing, Cham, Germany, 2018.

- 
- [96] D. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, 1979.
- [97] 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*, 39(6):1137–1149, June 2017.
- [98] T. Ringbeck, M. Albrecht, J. Frey, M. Grothof, H. Heß, H. Kraft, T. Möller, J. Mosen, and B. Schink. Time-of-Flight 3D-camera for autonomous navigation and industrial automation. In R. Lerch, editor, *Proceedings : 11th international conference ; an event of the Association for Sensor Technology, AMA*, Exhibition Centre Nuremberg, Germany, May 2003.
- [99] T. Ringbeck and B. Hagebecker. A 3D time of flight camera for object detection. In *Proc. Optical 3-D Measurement Techniques*, Zürich, Switzerland, July 2007.
- [100] H. Sandemann. Untersuchungen zur Genauigkeit von Streckenmessungen mit einer fokussierten plenoptischen Kamera im Fernbereich. In *Proc. Photogrammetrie - Laserscanning - Optische 3D-Messtechnik: Beiträge der Oldenburger 3D-Tage 2016 Taschenbuch – 22. April 2016*, Oldenburg, Germany, Apr. 2016.
- [101] J. Sander, M. Ester, H.-P. Kriegel, and X. Xu. Density-based clustering in spatial databases: The algorithm GDBSCAN and its applications. *Data Mining and Knowledge Discovery*, 2:169–194, June 1998.
- [102] E. Schubert, J. Sander, M. Ester, H. Kriegel, and X. Xu. DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems*, 42:1–21, July 2017.
- [103] S. H. Shaikh, K. Saeed, and N. Chaki. *Springer Briefs in Computer Science*, chapter Moving Object Detection Using Background Subtraction, pages 15–23. Springer International Publishing, Cham, Germany, 2014.
- [104] S. Shantaiya, K. Verma, and K. Mehta. A survey on approaches of object detection. *International Journal of Computer Applications*, 65(18):14–20, Mar. 2013.
- [105] A. Sobral and A. Vacavant. A comprehensive review of background subtraction algorithms evaluated with synthetic and real videos. *Computer Vision and Image Understanding*, 122:4–21, May 2014.
- [106] T. L. Song, H. W. Kim, and D. Musicki. Iterative joint integrated probabilistic data association for multitarget tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 51(1):642–653, 2015.
- [107] Y.-m. Song, S. Noh, J. Yu, C.-w. Park, and B.-g. Lee. Background subtraction based on gaussian mixture models using color and depth information. In *Proc. 2014 International Conference on Control, Automation and Information Sciences (ICCAIS 2014)*, pages 132–135, Gwangju, South Korea, Dec. 2014.
- [108] M. Soosai, C. Joshya, S. K. Rajaram, G. M. Innasi Muthu, S. Karthikumar, N. Chi, and A. Pugazhendhi. Versatile image processing technique for fuel science: A review. *Science of The Total Environment*, 780:146469, Mar. 2021.
- [109] R. Streier, S. Wirtz, K. Aleksandrov, H. Gehrmann, D. Stapf, M. Zhang, M. Vogelbacher, J. Matthes, and V. Scherer. Bereitstellung von experimentell abgesicherten CFD-Modellen für die thermische Umsetzung von Ersatzbrennstoffen (FLUFF). Technical report, Industrielle Gemeinschaftsforschung (IGF), Köln, Germany, 2022.

- [110] R. Streier, S. Wirtz, K. Aleksandrov, H.-J. Gehrmann, D. Stapf, M. Zhang, M. Vogelbacher, J. Matthes, and V. Scherer. Determination of the aerodynamic properties of refuse derived fuel by computer vision. In *Proc. 13th European Conference on Industrial Furnaces and Boilers*, Algarve, Portugal, Apr. 2022.
- [111] R. Streier, S. Wirtz, V. Scherer, K. Aleksandrov, H.-J. Gehrmann, D. Stapf, M. Zhang, M. Vogelbacher, and J. Matthes. Experimentelle Absicherung von CFD-Modellen für die thermische Umsetzung von Ersatzbrennstoffen (FLUFF). In *Proc. 30. Deutscher Flammentag*, Hanover, Germany (Online), Sept. 2021.
- [112] K. Tong, Y. Wu, and F. Zhou. Recent advances in small object detection based on deep learning: A review. *Image and Vision Computing*, 97:103910, 2020.
- [113] P. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78:138–156, Aug. 2000.
- [114] M. Vogelbacher. *Eine neue Methode zur kamerabasierten Analyse von Mehrstoffbrennern in industriellen Verbrennungsprozessen*. PhD thesis, Karlsruhe Institute of Technology, Karlsruhe, Germany, 2018.
- [115] M. Vogelbacher, M. Zhang, K. Aleksandrov, H. Gehrmann, and J. Matthes. Two- and three-dimensional benchmarks for particle detection from an industrial rotary kiln combustion chamber based on light-field-camera recording. *data*, 7(179), 2022.
- [116] M. Vogelbacher, M. Zhang, K. Aleksandrov, H.-J. Gehrmann, and J. Matthes. *Benchmark Dataset (2D/3D) of an Industrial Rotary Kiln Combustion Chamber with Refuse-Derived Fuel Particles from a Light-Field-Camera*, 2022.
- [117] E. Wan and R. Van Der Merwe. The unscented kalman filter for nonlinear estimation. In *Proc. IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, pages 153–158, Lake Louise, AB, Canada, Oct. 2000.
- [118] B. Wang, L. Wang, B. Shuai, Z. Zuo, T. Liu, K. L. Chan, and G. Wang. Joint learning of convolutional neural networks and temporally constrained metrics for tracklet association. In *Proc. 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 386–393, Las Vegas, Nevada, USA, June 2016.
- [119] J. Wang, T. Zhang, X. Xu, and Y. Li. A variational bayesian based strong tracking interpolatory cubature kalman filter for maneuvering target tracking. *IEEE Access*, 6:52544–52560, 2018.
- [120] Z. Wang. A new clustering method based on morphological operations. *Expert Systems with Applications*, 145:113102, Nov. 2019.
- [121] K. Weber, P. Quicker, J. Hanewinkel, and S. Flamme. Status of waste-to-energy in Germany, Part I – waste treatment facilities. *Waste Management & Research*, 38:23–44, Jan. 2020.
- [122] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfnder: real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.
- [123] C. Wu and C. Han. Strong tracking finite-difference extended kalman filtering for ballistic target tracking. In *Proc. 2007 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1540–1544, Sanya, China, Dec. 2007.
- [124] H. Wu, H. Su, Y. Liu, and H. Gao. Object detection and localization using stereo cameras. In *Proc. 2020 5th International Conference on Advanced Robotics and Mechatronics (ICARM)*, pages 628–633, Shenzhen, China, Dec. 2020.

- [125] L. L. X., L. J. J., Q. S. J., and S. J. T. Trajectory outliers detection based on local outlying degree. *Chinese Journal of Computers*, 10:1966–1975, 2011.
- [126] Y. Xuan, J. Pei, and Y. Wanhai. Firing particle flow detection and tracking in sequence images. In *Proc. the 3rd World Congress on Intelligent Control and Automation*, page 2666–2670, Hefei, China, June 2000.
- [127] N. Zeller, F. Quint, and U. Stilla. Calibration and accuracy analysis of a focused plenoptic camera. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3:205–212, Sept. 2014.
- [128] C. Zhang, T. Huang, and Q. Zhao. A New Model of RGB-D Camera Calibration Based on 3D Control Field. *Sensors*, 19(23), 2019.
- [129] M. Zhang, J. Matthes, K. Aleksandrov, H.-J. Gehrmann, and M. Vogelbacher. Detection and tracking of refuse-derived fuel particles based on a light field camera system. *Technisches Messen*, 88(6):396–407, 2021.
- [130] M. Zhang, J. Matthes, K. Aleksandrov, H.-J. Gehrmann, and M. Vogelbacher. Light-field camera based spatial multiple small particle tracking with post-processing. In M. A. Laribi, G. Carbone, and Z. Jiang, editors, *Advances in Automation, Mechanical and Design Engineering*, pages 95–123, Cham, Germany, 2022. Springer International Publishing.
- [131] M. Zhang, M. Vogelbacher, K. Aleksandrov, H.-J. Gehrmann, and J. Matthes. Analyse des Flug- und Abbrandverhaltens von Ersatzbrennstoffen auf Basis eines Lichtfeldkamarasystems. In *Proc. Forum Bildverarbeitung 2020*, pages 105–117, Karlsruhe, Germany (Online), Nov. 2020.
- [132] M. Zhang, M. Vogelbacher, K. Aleksandrov, H.-J. Gehrmann, D. Stapf, R. Streier, S. Wirtz, V. Scherer, and J. Matthes. A novel plenoptic camera-based measurement system for the investigation into flight and combustion behavior of refuse-derived fuel particles. *ACS Omega*, 8(19):16700–16712, May 2023.
- [133] M. Zhang, M. Vogelbacher, V. Hagenmeyer, K. Aleksandrov, H.-J. Gehrmann, and J. Matthes. 3-D refuse-derived fuel particle tracking-by-detection using a plenoptic camera system. *IEEE Transactions on Instrumentation and Measurement*, 71:1–15, 2022.
- [134] Y. Zhang and T. Funkhouser. Deep depth completion of a single RGB-D image. In *Proc. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 175–185, Salt Lake City, Utah, USA, June 2018.
- [135] Y. Zhang, Y. Wang, W. He, and B. Yang. Application of a novel particle tracking algorithm in the flow visualization of an artificial abdominal aortic aneurysm. *Bio-Medical Materials and Engineering*, 24(6):2585–2591, 2014.
- [136] Y. Zhang, Y. Wang, and P. Jia. Improving the delaunay tessellation particle tracking algorithm in the three-dimensional field. *Measurement*, 49:1–14, 03 2014.
- [137] B. Zhou and N. Bose. Multitarget tracking in clutter: fast algorithms for data association. *IEEE Transactions on Aerospace and Electronic Systems*, 29(2):352–363, 1993.
- [138] X. Zou. A review of object detection techniques. In *Proc. 2019 International Conference on Smart Grid and Electrical Automation (ICSGEA)*, pages 251–254, Xiangtan, China, Aug. 2019.
- [139] Z. Zou, Z. Shi, Y. Guo, and J. Ye. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3):257–276, 2023.



# Nomenclature

## Latin letters

### upper case

SYMBOL	DESCRIPTION
$I$	intensity
$B$	Background model
$F$	state transition matrix
$B$	control-input matrix
$H$	measurement matrix
$K_t$	Kalman gain
$D$	detection
$T$	tracklet
$C$	cost matrix
$S$	residual covariance
$N$	point number with one cluster
$V(i)$	volume of the elliptical validation region
$P$	covariance matrix
$CIC$	consecutive invisible count
$\mathcal{D}$	duration of a tracklet
$ \bar{V}_i $	arithmetic average magnitude of the temporary velocity vector
$\mathcal{R}$	cross-correlation coefficient
$KN$	cardinality of assigned detections
$T$	temporary position vector of a tracklet
$R^2$	measurement R-squared
$R^{*2}$	measurement adjusted R-squared
$R$	range of the grid
$JSC$	Jaccard similarity index for positions
$JSC_\theta$	Jaccard similarity index for tracks

### lower case

SYMBOL	DESCRIPTION
$t$	time
$f$	focal length
$d$	aperture of the microlense
$k$	threshold/factor
$m$	gradient magnitude

$minPts$	minimum number of points
$\mathbf{w}$	process noise vector
$\mathbf{u}$	control vector
$\mathbf{v}$	measurementnoise vector
$\mathbf{z}$	measurement vector
$d_{ij}$	2D Mahalanobis distance between track and measurement
$a_0$	weighting factor
$b$	probability of non-assignment
$z$	z (depth) coordinate
$w$	window size
$d(\mathcal{X}, \mathcal{Y})$	distance between a set of ground truth trajectories $\mathcal{X}$ and a set of candidate trajectories $\mathcal{Y}$

## Greek letters

### upper case

SYMBOL	DESCRIPTION
$\Omega$	validation matrix

### lower case

SYMBOL	DESCRIPTION
$\lambda$	wave length of the light
$\alpha$	learning rate
$\sigma$	standard deviation
$\varepsilon$	radius
$\beta$	conditional association probablitiy
$\chi$	joint event
$\tau$	detection association indicator
$\delta$	target detection indicator
$\epsilon$	subscript for threshold
$\theta$	angle of two velocity vectors
$\alpha(\mathcal{X}, \mathcal{Y})$	normalized pairing score of a set of ground truth trajectories $\mathcal{X}$ and a set of candidate trajectories $\mathcal{Y}$
$\beta(\mathcal{X}, \mathcal{Y})$	full normalized pairing score of a set of ground truth trajectories $\mathcal{X}$ and a set of candidate trajectories $\mathcal{Y}$

## Mathematical symbols

SYMBOL	DESCRIPTION
$*$	convolution operator
$\nabla = \left( \frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)^T$	partial derivative operator
$\text{var}(\dots)$	variance
$\text{cov}(\dots)$	covariance

## Abbreviations

SYMBOL	DESCRIPTION
RDF	refuse-derived fuel
FGR	flue gas recirculation
GNN	global nearest neighbor
JPDA	joint probabilistic data association
PE	polyethylene
CFD	computational fluid dynamics
BREND	combustion chamber with steam boiler
ToF	time-of-flight
PMC	photonic mixer devices
IR	infrared
RGB-D	red, green, blue - depth
SIFT	scale invariant feature transform
DoG	difference of Gaussian
CNN	convolutional neural network
DBSCAN	density-based spatial clustering of applications with noise
OPTICS	ordering points to identify the clustering structure
MOT	multiple object tracking
QCC	quasi-cluster centers
PTV	particle-tracking velocimetry
GOTURN	generic object tracking using regression networks
EKF	extended Kalman filter
UKF	unscented Kalman filter
NN	nearest neighbor
PDA	probabilistic data association
ROI	region of interest
MLESAC	maximum likelihood estimation sample consensus
TP	true positive
FP	false positive
FN	false negative
TN	true negative
PPV	positive predictive value
TPR	true positive rate
ROAM	rule- and motif-based anomaly detection in moving objects
TraLOD	trajectory outlier detection based on local outlier degree
DNN	dual nearest neighbor
RMSE	root mean squared error
MAPE	mean absolute percentage error
MO	morphological operation





# List of Figures

1.1. Example images of coal dust and Fluff particles. (a) Coal dust. (b) Fluff particles [111]. . . . .	2
1.2. Examples of captured images [129]. (a) Basic-focus image captured by the plenoptic camera. (b) Corresponding depth map in false color, where black indicates no 3D information available. . . . .	2
1.3. An example of depth-frame of a particle trajectory that is acquired by detecting and tracking refuse-derived fuel particles inside a rotary kiln based on images captured by a plenoptic camera. The rotary kiln has a length of 8.4 m. Abscissa represents frame and the corresponding temporal depths of the trajectory are plotted on the ordinate. . . . .	5
1.4. Schematic of the primary contribution of the dissertation. . . . .	6
2.1. Schematic of the BRENDA test facility [6]. Part of the test facility, where the experiments are conducted, is marked with a red dashed rectangle. . . . .	8
2.2. Sketch map of fuel particle moving in the rotary kiln. . . . .	8
2.3. Images of the experiments conducted in the rotary kiln. The plenoptic camera mounted outside the kiln marked with a green dashed rectangle is triggered by the computer in the left. An infrared camera cooled by a cooling probe is marked with a red dashed rectangle. The camera is controlled by the laptop on the right side. . . . .	9
2.4. Defined trapezoidal image area and an example of temperature profile at the inner wall of the rotary kiln. Left: Defined trapezoidal image area. Right: Example of temperature profile inside the rotary kiln. . . . .	9
2.5. Sketch map of fuel particle moving in the combustion chamber. . . . .	10
2.6. Images of the experiments conducted in the combustion chamber. The plenoptic camera cooled by a cooling probe installed at the top of the combustion chamber is marked with a yellow dashed rectangle. The real-time temperature inside the cooling probe is measured and displayed by the thermocouple on the left side. A height-adjustable lance is marked with a red dashed rectangle. . . . .	10
2.7. Refuse-derived fuel. (a) Composition of the RDF. (b) Various fractions of applied RDFs [130], (a) wood chips, (b) confetti, (c) paper shreds, (d) PE granules. . . . .	11
2.8. Schematically depicted plenoptic function. (a) Function parametrized by angles. (b) Function parametrized by Cartesian values. [4] . . . . .	13
2.9. Parametrization of 4D light-field. [58] . . . . .	14
2.10. Plenoptic camera model. (a) Traditional plenoptic camera. The main lens is focused at the microlens plane. (b) Single microlens model [68]. . . . .	14
2.11. Plenoptic camera model. Left: Traditional plenoptic camera model. Right: Focused plenoptic camera model (Plemoptic camera 2.0). [67] . . . . .	15
2.12. An example of 3D point cloud of the rotary kiln and fuel particles captured by the plenoptic camera. The red circle on the right marks the point cloud of a single particle. . . . .	16
2.13. Experiment setup to test the accuracy of the plenoptic camera in the depth direction. The left illustrates the test schematic; the center presents the test plate; the enlarged view of the pattern on the test plate is shown on the right side. . . . .	16

2.14	Acquired 3D point clouds of the test plate at each distance in the depth direction. Each time the test plate was spaced 1 m apart in the depth direction. . . . .	17
2.15	Boxplot of the depth coordinates of the test plate at each distance. . . . .	17
2.16	Depth information graphic. The blue line is the average measured value, the red dashed line is the expected value, and the green line is the depth value after calibrating the measured value by the functions. . . . .	18
2.17	Overview of SIFT feature extraction. (a) The scale space of SIFT. (b) Comparing a pixel to its neighbors to detect maxima and minima of the difference-of-Gaussian images [66]. . . . .	25
2.18	Illustration of the DBSCAN Cluster Model. . . . .	27
2.19	Sorted 6-distance graph of points in descending order according to [26]. . . . .	28
2.20	Overview of the nearest neighbor approach dealing with various situations. Left: correct association of tracks and detections. Middle: Nearest neighbor approach fails to handle assignment problems with high particle density. Right: Track 1 is incorrectly assigned with the nearest noise. Note that only a schematic representation of the assignment process is shown here; subsequent update processes will be based on this assignment result, <i>i.e.</i> , the association does not represent the final track. It is also valid for Figure 2.21 and 2.22. . . . .	33
2.21	Overview of the global nearest approach dealing with various situations. (a) correct association of tracks and detections. (b) Following the cost matrix in the middle, GNN is able to correctly associate the detections with the corresponding tracks. (c) Track 1 remains incorrectly assigned with the nearest noise. . . . .	35
2.22	Overview of the probabilistic data association dealing with various situations. (a) Correct association of tracks and detections. (b) Track 1 is associated with a weighted position of detection 1 and detection 2. However, the correctness can only be determined later. If the two trajectories are close together subsequently with mostly overlapped gating areas, the two trajectories will probably become increasingly similar later on. Conversely, if the two trajectories tend to be separated, the PDA will most likely arrive at the correct trajectories. (c) The probabilistic data association approach is able to reduce the impact of noise. . . . .	37
2.23	Detection validation example with the corresponding validation matrix and all possible feasibility matrices of the validation matrix. . . . .	39
3.1.	Spatially formed clusters and their corresponding 2D image areas. Each color stands for an individual cluster. Since each captured pixel is assigned with an image coordinate, a spatial coordinate, and a gray value, the 3D clusters in the cartesian coordinate and the 2D clusters in the image coordinate can be transformed into each other. The red ellipse and the green rectangular enlarge a 3D cluster and a 2D transformed image area, respectively. . . . .	42
3.2.	Detection results of 2D-SIFT and 3D-DBSCAN. Left: Detection result of 2D-SIFT. The center of each green circle corresponds to the position of the detected particle, whose sizes are pointed by the circles' radius. The yellow box is zoomed in to display the detection on the initial frame and the frame after background subtraction. Right: Detection result of 3D-DBSCAN. The center of each red circle corresponds to the center of the cluster. Radius of the circle indicates the cluster size. The yellow box shows the particle detection both on the 2D frame and in the 3D point cloud. [133] . . . . .	42
3.3.	Schematic of the novel combined detection approach. . . . .	43
3.4.	Flame segmentation within a defined rectangular ROI around the burner, as marked in red rectangular. The red-marked flame on the right is the segmented flame. . . . .	43
3.5.	Five cases of the comparison of particle detections. The blue polygon is an instance of the defined ROI. [133] . . . . .	44

3.6. Instance and solution for case 2. Transformed 2D clusters are marked by small dots, where each dot corresponds to a pixel. Circles with cross centers highlight the detections of the SIFT algorithm. The cross center locates at the detected keypoints, and the circle's radius indicates the particle's size. The identical notation is also used in the following three figures.	45
3.7. Instance and solution for case 3.	46
3.8. Instance and solution for case 4. Three instances containing one particle and two non-particles and their corresponding solutions are presented in the figure. Each column of the solution illustrates an example.	47
3.9. Instance for case 5.	48
3.10 Cluster of the rotary kiln and a corresponding estimation of the kiln's geometry. (a) A cluster of the rotary kiln. (b) Estimated cylindrical kiln model.	49
3.11 Distribution of the distances between particles and the rotary kiln axis. The horizontal coordinate represents the distance to the axis, and the vertical coordinate illustrates the number of particles from the individual distance interval.	49
3.12 Rounded particle radii of 1500 particles. (a) The radii are sorted ascendingly. The horizontal coordinate indicates the index of a particle, and the vertical coordinate reports the rounded radius of the particle. (b) Distribution of the particles' radii and the corresponding normal distribution. The horizontal axis shows the radii, and the vertical axis stands for the probability of each radius interval.	51
3.13 Assignment instance with five detections and five particles. Blue stars represent particles, and red crosses stand for detections. The corresponding cost matrix is illustrated on the left. The red color marks the FP, while the blue color marks the FN. The green color is the global lowest cost, <i>i.e.</i> , the final assignment outcome.	51
4.1. Schematic of the multiple particle tracking procedure. Instances of available approaches for individual steps are listed below.	53
4.2. Gating elliptical cylinder. The center of the elliptical cylinder locates at the prediction.	55
4.3. Schematic overview of the novel post-processing framework.	58
4.4. Examples of eligible and ineligible tracklets. Each condition is accompanied by two instances. Tracklets in green are eligible tracklets, while red tracklets are false tracklets.	59
4.5. Schematics of the notations for outlier detection and three situations concerning the outlier criteria. Green circles stand for inliers, while red circle represent outlier. Situation 1: the angle between $V_{i,t_1}$ and $V_{i,t_2}$ satisfies the first condition, whereas the vector difference magnitude ( $ \Delta V_{i,t_2} $ ) does not exceed the limit. This situation may be a consequence of several particles falling after rising slightly due to their inertia, which is not able to judge outliers. Situation 2: the velocity difference magnitude $ \Delta V_{i,t_3} $ satisfies the second condition, but the angle $\theta_{i,t_3}$ is relatively small. The main reason is the cross-frame assignment. The two positions $T_{i,t_3}$ and $T_{i,t_4}$ are from two adjacently associated time points instead of two consecutive frames. Consequently, $ \Delta V_{i,t_3} $ is beyond the threshold, while $\theta_{i,t_3}$ is small. Situation 3: $T_{i,t_4}$ fulfills the both conditions and should be pruned.	61
4.6. Forward and backward computing of outliers in Situation 3 of Figure 4.5. For forward scanning, $T_{i,t_3}$ could be wrongly identified as an outlier because of the unfavorable influence of the real outlier $T_{i,t_3}$ . Reversely, computing both forward and backward can address the issue since the problem does not persist in computing backward.	61
4.7. Checking the last point. The average of the last three velocity vectors $V_{i,t_3}$ , $V_{i,t_4}$ , $V_{i,t_5}$ is assumed the next velocity vector for computing of Equation 4.18.	62
4.8. Flowchart of tracklet connection.	63

4.9. Tracklet connection and merging [130]. (a) Tracklet connection: $T_i$ and $T_k$ do not overlap temporally and are simply connected. (b) Tracklet merging: $T_i$ and $T_k$ overlap temporally. For the overlapping time, the tracklet is represented by the average arithmetic positions of $T_i$ and $T_k$ . . . . .	64
4.10 Tracklet connection part. $T_i$ and $T_k$ is a possible matching pair that is connected or merged to build a new tracklet $T_j$ . Left illustrates the connection of tracklet $T_i$ and $T_k$ , while the tracklet merging is shown on the right. Dashed rectangles mark the respective connected parts $T_{ik}$ . In the figure, the time cardinality of each tracklet is also indicated in parentheses to distinguish it from the actual time (frame). Obviously, unlike the discontinuous actual time of each tracklet, the time cardinality is always continuous. . . . .	65
4.11 25 combinations of Law's masks for feature identification. . . . .	66
4.12 Example of windows for laws' mask on the sum of difference images acquired by background subtraction [133]. The left window (red/green) is a vertical window for tracklet connection, where the vertical connection dominates. Correspondingly, the right window (blue/yellow) is a lateral window for tracklet connection. Moreover, the window (orange/purple) indicates a vertical window for tracklet merging. . . . .	67
4.13 Examples of windows and their respective convolutions with diverse combinations of Laws mask. Window (f) corresponds to the window without tracklet-like features, while the remaining five are with features. The upper table shows the convolved values of order $10^8$ , while the lower table is normalized by the corresponding bottom values, thus highlighting the contrast of values with and without tracklet-like features. . . . .	68
4.14 Instances of tracklet connection using Kuhn-Munkres algorithm (GNN) and DNN, respectively. As described previously, the connection process is iterative. The cost matrix is updated immediately after each connection, based on which the remaining tracklets are reconnected until no potential matching pairs. Each arrow in the figure points to an iteration. $C$ in the cost matrices indicates the impossibility of a tracklet matching pair with a tremendous cost value. Tracklets with the same marker belong to the same tracklet. [133] . . . . .	69
4.15 Flowchart of trajectories fusion (left) and a corresponding instance of trajectories fusion (right). . . . .	70
4.16 Initial tracking results of wood chips (a) and paper shreds (b). . . . .	71
4.17 Temporary tracking results of wood chips (a) and paper shreds (b) after false tracklets elimination. . . . .	71
4.18 Temporary tracking results of wood chips (a) and paper shreds (b) after deleting outliers in each tracklet. . . . .	72
4.19 Temporary tracking results of wood chips (a) and paper shreds (b) after tracklets connection and merging. . . . .	72
4.20 Temporary tracking results of paper shreds after tracklets connection/merging and several enlarged examples of the trajectories highlighted in the rectangles. . . . .	73
4.21 Final tracking results of wood chips (left) and paper shreds (right) after post-processing. . . . .	73
4.22 Examples of spatial particle trajectories without regression from two views. . . . .	74
4.23 Measures of the polynomial estimated trajectories with different orders (a) RMSE (b) MAPE. (c) R-squared. (d) Adjusted R-squared. . . . .	76
4.24 Spatial particle trajectories with polynomial regression. Left: Estimated polynomials. Red Cross stands for the lance, and thick solid lines are the estimated polynomials. Right: spatial trajectories with corresponding particle gray values. Each point represent the corresponding gray values of the detections. . . . .	77
5.1. Example of the labeled ground truth image. The particles and flame are labeled pixel-by-pixel. . . . .	80

5.2. Various background models. (a) Temporal median background model. (b) Temporal mean background model. (c) Adaptive background model with $\alpha = 0.05$ (d) Running Gaussian average with $\alpha = 0.99, k = 0.5$ . . . . .	81
5.3. Detection result of 2D-SIFT after background subtraction and the defined ROI. The center of each green circle corresponds to the position of the detected particle, whose sizes are pointed by the circles' radius. The red polygon is the defined ROI. . . . .	83
5.4. Performance ( $F_1$ -score) of various background subtraction algorithms with the corresponding parameter values listed in Table 5.4. The $F_1$ -scores are computed by 2D-SIFT algorithm with local maximum of DoG=500. . . . .	84
5.5. Detection performance of various classes of RDF particles by means of the OTSU threshold selection approach. Blue stands for burning particle in air; cyan represents non-burning particle in air; yellow denotes burning particle on the inner wall. The color notation also applies to all following similar image. (a) Distribution of TP particle detections of the ground truth subsets. (b) Distribution of FN particle detections of the ground truth subsets. (c) Recall values of various particle classes in the ground truth subsets. . . . .	86
5.6. Detection performance of various classes of RDF particles by means of the SIFT approach. (a) Distribution of TP particle detections of the ground truth subsets. (b) Distribution of FN particle detections of the ground truth subsets. (c) Recall values of various particle classes in the ground truth subsets. . . . .	88
5.7. Distance between points from five distinct point clouds and their nearest neighbors. . . . .	89
5.8. Detection performance of various classes of RDF particles by means of the 3D-DBSCAN clustering method. (a) Distribution of TP particle detections of the ground truth subsets. (b) Distribution of FN particle detections of the ground truth subsets. (c) Recall values of various particle classes in the ground truth subsets. . . . .	91
5.9. Detection performance of various classes of RDF particles by means of the 3D clustering based on morphological operations. (a) Distribution of TP particle detections of the ground truth subsets. (b) Distribution of FN particle detections of the ground truth subsets. (c) Recall values of various particle classes in the ground truth subsets. . . . .	93
5.10. Detection performance of various classes of RDF particles by means of the 3D clustering based on quasi-cluster centers. (a) Distribution of TP particle detections of the ground truth subsets. (b) Distribution of FN particle detections of the ground truth subsets. (c) Recall values of various particle classes in the ground truth subsets. . . . .	95
5.11. Extract from a .txt file of the ground truth trajectories. First column indicates the frame number, while the coordinates of a particle are presented in the last two columns. . . . .	100
5.12. Ground truth trajectories of wood chips (a) and PE granules (b). Crosses stand for particle coordinates and each color indicates one individual trajectory. . . . .	100
5.13. Adding a set of adjacent difference images and a background image of each fraction as the optical ground truth of wood chips (a) and confetti (b). . . . .	101
5.14. Tracking results of wood chips. (a) Tracking result of GNN. Crosses stand for detections. Detections of same tracklets are linked with solid lines. (b) Tracking result of JPDA. Squares stand for detections. Detections of same tracklets are linked with solid lines. . . . .	104
5.15. Tracking results of wood chips and PE granules with post-processing. (a) Tracking result of wood chips using GNN. Crosses stand for detections. Detections of same tracklets are linked with solid lines. (b) Tracking result of wood chips by means of JPDA. Squares stand for detections. Detections of same tracklets are linked with solid lines. (c) Tracking result of PE granules using GNN. (d) Tracking result of PE granules by means of JPDA. . . . .	105

5.16	Estimated spatial trajectories of wood chips and confetti with corresponding particle gray values. (a) Estimated polynomials of wood chips. Red cross stands for the lance, thick solid lines are the estimated polynomials, and thin solid lines represent the direct converted 3D trajectories. (b) Estimated polynomials of confetti. (c) Estimated polynomials of wood chips with corresponding particle gray values. The red cross stands for the lance, thin solid lines are the estimated polynomials, and points' brightness represent the corresponding gray values of the detections. (d) Estimated polynomials of confetti with corresponding particle gray values.	107
5.17	Examples of optical ground truth obtained by adding a set of adjacent difference images and a background image of each fraction for the entire sequence [132]. (a) Wood chips. (b) Confetti. (c) Paper shreds. (d) PE granules.	109
5.18	Definitions of the dwell time (flight duration) in different situations. (a) Dwell time of non-burning particles. (b) Dwell time of burning particles that do not burn out during the flights. (c) Dwell time of burning particles that burn out during the flights. [132]	110
5.19	Box plot of the flight durations of various RDF fractions. (a) Flight durations of the fractions at 4.5 bar air feed pressure. (b) Flight durations of the fractions at 5 bar air feed pressure.	111
5.20	Schematically illustration of the space-sliced velocity in the depth direction. The lance is marked by a red cross as previous. [132]	112
5.21	Box plot of space-sliced velocity in the depth direction of wood chips. The corresponding percentage of remaining particles at each distance interval is shown on the bottom. (a) Space-sliced velocity at 4.5 bar air feed pressure. (b) Space-sliced velocity at 5 bar air feed pressure.	113
5.22	Box plot of space-sliced velocity in the depth direction of confetti. The corresponding percentage of remaining particles at each distance interval is shown on the bottom. (a) Space-sliced velocity at 4.5 bar air feed pressure. (b) Space-sliced velocity at 5 bar air feed pressure.	114
5.23	Box plot of space-sliced velocity in the depth direction of paper shreds. The corresponding percentage of remaining particles at each distance interval is shown on the bottom. (a) Space-sliced velocity at 4.5 bar air feed pressure. (b) Space-sliced velocity at 5 bar air feed pressure.	115
5.24	Box plot of space-sliced velocity in the depth direction of PE granules. The corresponding percentage of remaining particles at each distance interval is shown on the bottom. (a) Space-sliced velocity at 4.5 bar air feed pressure. (b) Space-sliced velocity at 5 bar air feed pressure.	117
5.25	Bar plot of ignition time of confetti. (a) Ignition time at 4.5 bar air feed pressure. (b) Ignition time at 5 bar air feed pressure.	118
5.26	Bar plot of ignition time of paper shreds. (a) Ignition time at 4.5 bar air feed pressure. (b) Ignition time at 5 bar air feed pressure.	118
A.1	Optical ground truth and 2D trajectories of wood chips under 4.5 bar and 5 bar air feed pressures. (a) Optical ground truth at 4.5 bar pressure over 3102 frames. (b) Optical ground truth at 5 bar pressure over 3234 frames. (c) 2D trajectories of wood chips at 4.5 bar pressure. (d) 2D trajectories of wood chips at 5 bar pressure.	147
A.2	Optical ground truth and 2D trajectories of confetti under 4.5 bar and 5 bar air feed pressures. (a) Optical ground truth at 4.5 bar pressure over 3126 frames. (b) Optical ground truth at 5 bar pressure over 3286 frames. (c) 2D trajectories of confetti at 4.5 bar pressure. (d) 2D trajectories of confetti at 5 bar pressure.	148

---

A.3.Optical ground truth and 2D trajectories of paper shreds under 4.5 bar and 5 bar air feed pressures. (a) Optical ground truth at 4.5 bar pressure over 500 frames. (b) Optical ground truth at 5 bar pressure over 3002 frames. (c) 2D trajectories of paper shreds at 4.5 bar pressure. (d) 2D trajectories of paper shreds at 5 bar pressure. . . . .	149
A.4.Optical ground truth and 2D trajectories of PE granules under 4.5 bar and 5 bar air feed pressures. (a) Optical ground truth at 4.5 bar pressure over 3029 frames. (b) Optical ground truth at 5 bar pressure over 3113 frames. (c) 2D trajectories of PE granules at 4.5 bar pressure. (d) 2D trajectories of PE granules at 5 bar pressure. . . . .	150
A.5.Simulation of the BRENDA rotary kiln and trajectories of paper shreds and wood chips. Each circle is spaced 1 m apart in the depth direction. . . . .	151
A.6.Comparison of simulated flight durations with experimentally determined values at 5 bar air feed pressure. (a) Wood chips. (b) Confetti. (c) Paper schreds. (d) PE granules. . . . .	152
A.7.Comparison of simulated flight velocities in the depth direction with experimental outcomes at 5 bar feed pressure. (a) Wood chips. (b) Confetti. (c) Paper schreds. (d) PE granules. . . . .	153





## List of Tables

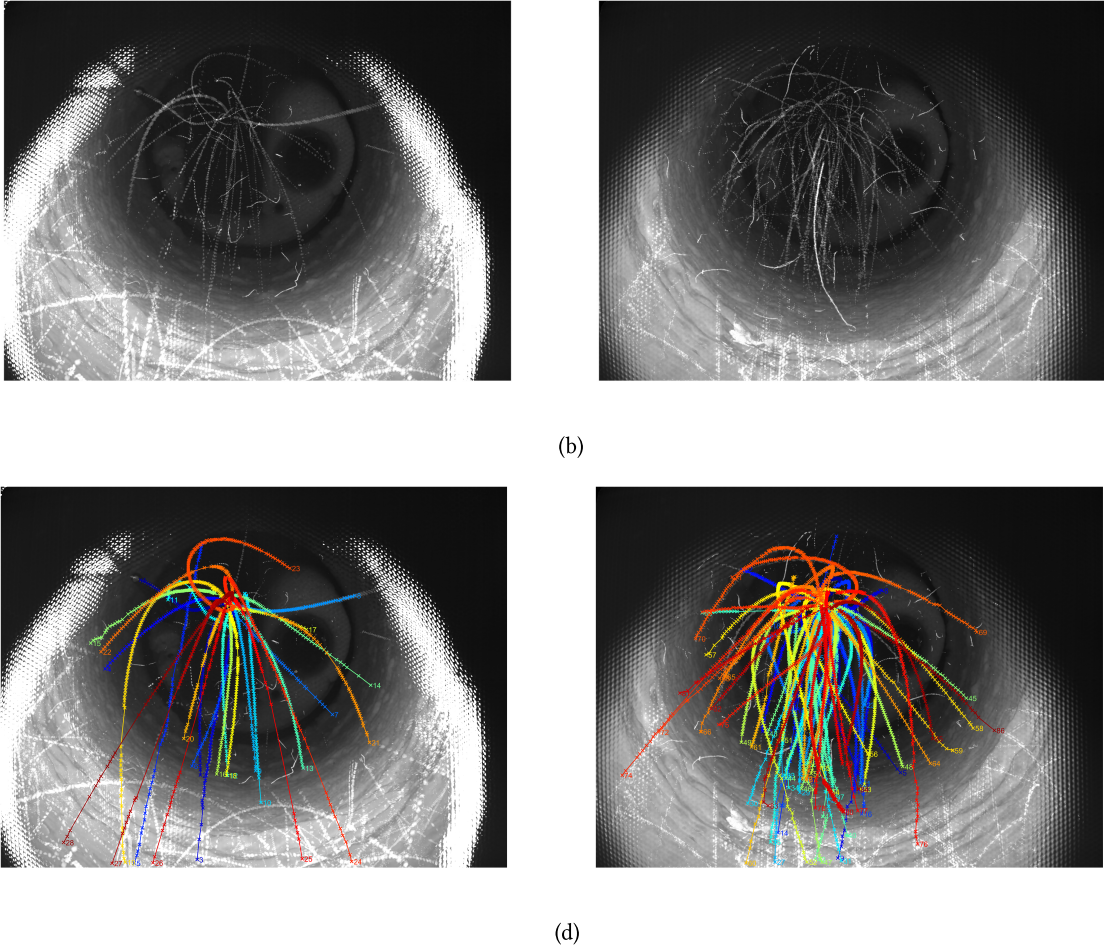
2.1. Comparison of different 3D imaging technologies according to [59] and [2]. . . . .	12
5.1. Classes with their corresponding labels in ground truth. . . . .	80
5.2. Distribution of various classes of RDF particles in the five ground truth subsets (V1-V5). . . .	81
5.3. Distribution of various classes of RDF particles within the defined ROI in the five ground truth subsets. . . . .	82
5.4. The adjustable parameters of different background methods and their optimal values for the current work. . . . .	83
5.5. Performance of 2D particle detection using the OTSU approach for the five ground truth subsets. . . . .	85
5.6. Performance of 2D particle detection using the SIFT approach for the five ground truth subsets. . . .	87
5.7. Computation time of SIFT with various local maximum of DoG values using MATLAB. . . .	89
5.8. Performance and computation time of 3D-DBSCAN clustering approach with different parameter values using Python. . . . .	90
5.9. Performance and computation time of 3D-MO-based clustering approach with different parameter values by means of MATLAB. . . . .	92
5.10 Performance and computation time of 3D-QCC clustering approach with different parameter values utilizing MATLAB. . . . .	94
5.11 Performance evaluation of particle detection employing 2D gray value based algorithms with median background subtraction, 3D clustering methods, and the corresponding combined approaches. . . . .	96
5.12 Parameters of the estimated cylindrical rotary kiln inner wall in mm. . . . .	98
5.13 Performance of 3D classification on ground truth dataset. . . . .	99
5.14 Performance of 3D classification on the detection result from the combination of 2D-SIFT and 3D-DBSCAN. . . . .	99
5.15 Overview of ground truth trajectories of the RDF fractions with their corresponding frame intervals and amount of trajectories. . . . .	101
5.16 Parameter values of the post-processing framework. . . . .	104
5.17 Evaluation of tracking performance with and without post-processing utilizing GNN and JPDA. Upper values illustrate performance of GNN and JPDA without post-processing, and lower values displays performance of GNN and JPDA with post-processing. . . . .	105
5.18 Evaluation of 3D regression. Upper, middle, and lower values represent the respective measure values in the x, y, and z directions. . . . .	108



## A. Appendix

### A.1. 2D particle Trajectories

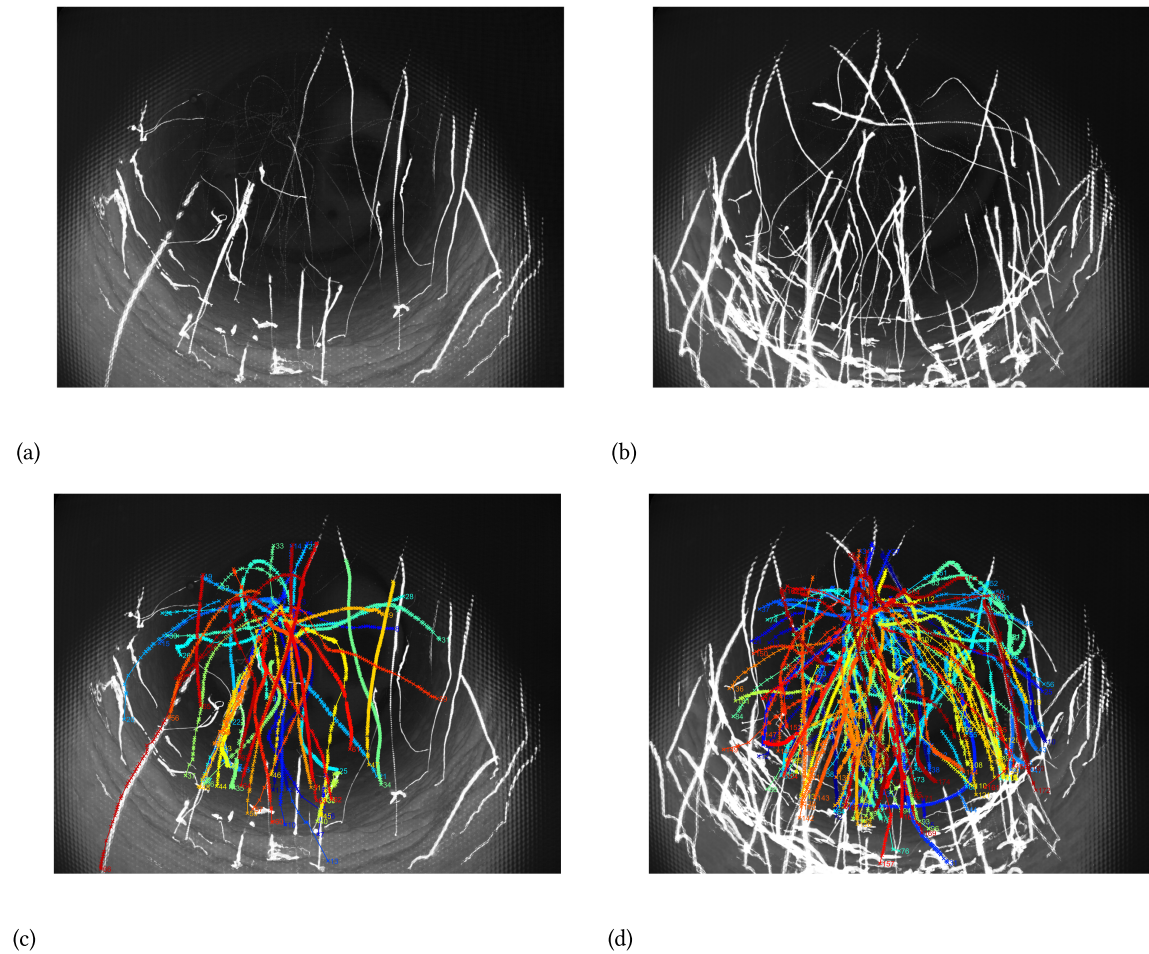
Particle trajectories of the four RDF fractions under 4.5 bar and 5 bar air feed pressures are presented in the chapter. As mentioned in Chapter 5.2.3, the fuel particles starting from the lance are tracked by the linear Kalman filter and 2.5D GNN over the entire sequence. Figure A.1 shows the tracking results of wood chips at 4.5 bar and 5 bar feed pressure with sequence lengths of 3102 and 3234 frames, corresponding to a time interval of 9.4 s and 9.8 s, respectively. The trajectories of the wood chips are more uniform than those of paper chips due to their relatively concentrated masses and sizes.



**Figure A.1.:** Optical ground truth and 2D trajectories of wood chips under 4.5 bar and 5 bar air feed pressures. (a) Optical ground truth at 4.5 bar pressure over 3102 frames. (b) Optical ground truth at 5 bar pressure over 3234 frames. (c) 2D trajectories of wood chips at 4.5 bar pressure. (d) 2D trajectories of wood chips at 5 bar pressure.

Moreover, the wood chips possess favorable visibility, facilitating accurate particle detection. Besides, the movements of the wood chips are essentially affected by the initial velocity and gravity, so their trajectories are more approximate parabolas. The optical ground truth indicates that wood chips can not be ignited during the flight. Since several wood chips do not land inside the camera's field of view, their actual flight time is longer than computed.

Figure A.2 shows the tracking results of confetti over 3126 (9.47 s) and 3286 (9.94 s) frames under the pressure of 4.5 bar and 5 bar. As presented by the figure, confetti triggers more complex and tangled trajectories compared to wood chips, significantly challenging the tracking performance. Notwithstanding, confetti particles are smaller with a shorter burning time, which does not affect the detection of other confetti particles due to their combustion flames giving rise to slight effects on particle tracking-by-detection. The optical ground truths prove that confetti is ignited during the flight.

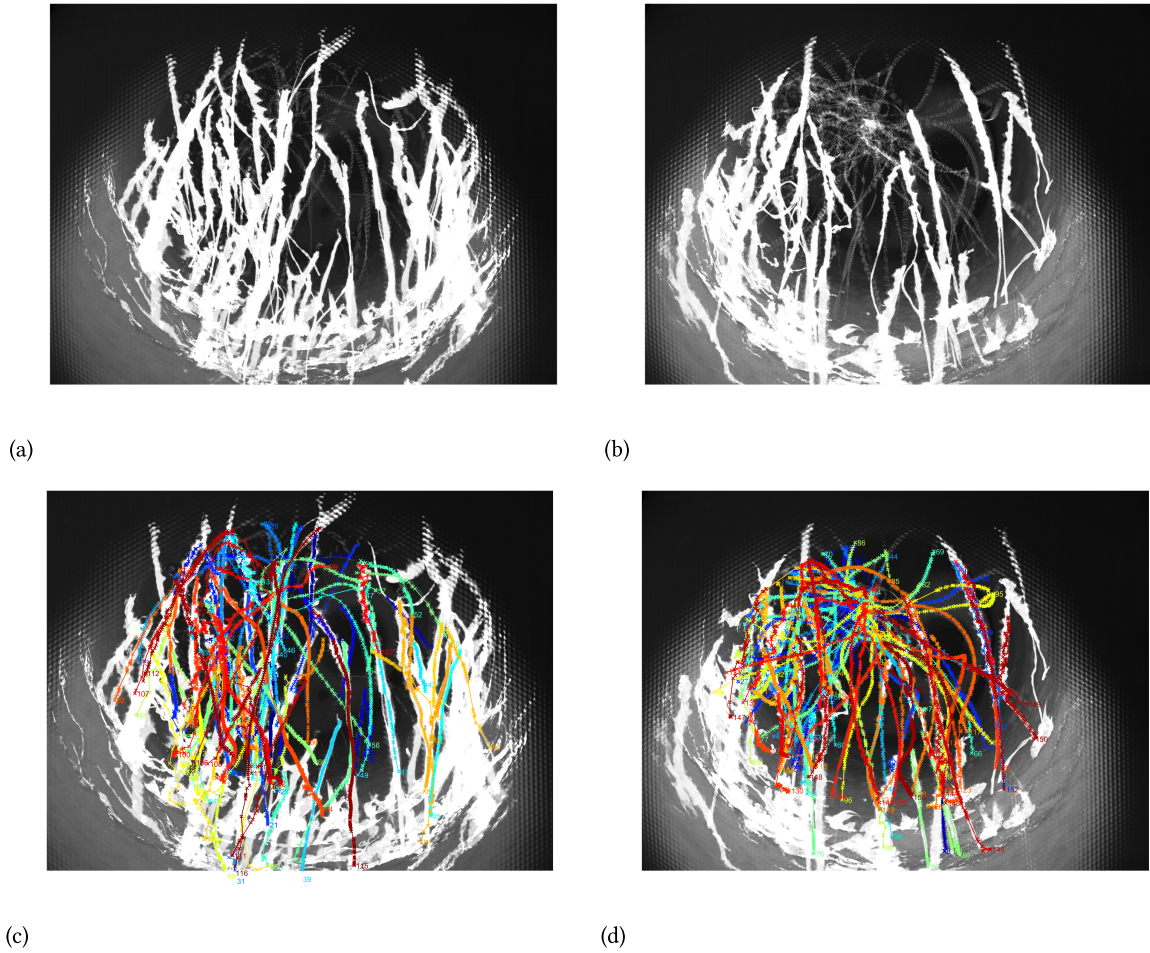


**Figure A.2.:** Optical ground truth and 2D trajectories of confetti under 4.5 bar and 5 bar air feed pressures. (a) Optical ground truth at 4.5 bar pressure over 3126 frames. (b) Optical ground truth at 5 bar pressure over 3286 frames. (c) 2D trajectories of confetti at 4.5 bar pressure. (d) 2D trajectories of confetti at 5 bar pressure.

Figure A.3 exhibits the 2D trajectories of paper shreds over 500 (1.5 s) and 3002 (9.10 s) frames. Because the particles were manually and, therefore, randomly dosed, too many particles were thrown into the rotating kiln at 4.5 bar resulting in an entirely bright optical ground truth. Thus, the corresponding optical ground truth is shown over 500 frames. As a consequence of a large inflow area and uniform



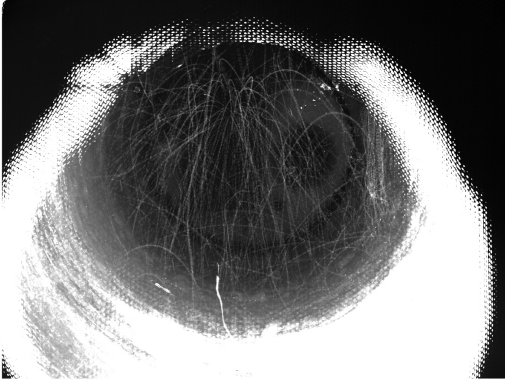
mass distribution, the motions of paper shreds are considerably influenced by flow drag. In addition, the flight of a paper shred is also accompanied by a pronounced tumbling motion due to its larger dimensions compared to the other three fractions. If several paper shreds are conveyed simultaneously from the lance, their traces probably cross. As indicated by the figure, paper shreds can be ignited by the heat inside the rotary kiln. After ignition, the detected area of one paper shred increases in the captured image, which also affects the detection of the other paper shreds. Therefore, the trajectories of the paper shreds appear disorderly and irregular. However, because of the large number of particles, a few complete trajectories are still available for subsequent analysis of flight behavior. It should be noted that the captured image owns artifacts in the edge regions, where the applied plenoptic camera is not able to provide usable 3D information. Therefore, some vertically ascending burning particles cannot be fully tracked, so their flight and burning behavior cannot be analyzed.



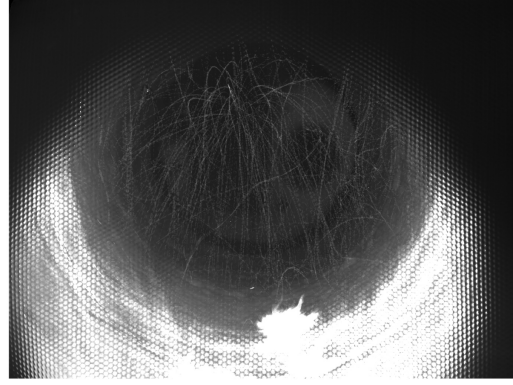
**Figure A.3.:** Optical ground truth and 2D trajectories of paper shreds under 4.5 bar and 5 bar air feed pressures. (a) Optical ground truth at 4.5 bar pressure over 500 frames. (b) Optical ground truth at 5 bar pressure over 3002 frames. (c) 2D trajectories of paper shreds at 4.5 bar pressure. (d) 2D trajectories of paper shreds at 5 bar pressure.

Figure A.4 presents the tracking results of PE granules over 3029 (19.69 s) and 3113 (20.23 s) frames. As mentioned beforehand, the PE granules require a longer exposure time due to their transparency and smaller sizes, which also results in a halving of the frame rate. For the same sequence length of around 3000 frames, the total duration of the sequence is about twice the other fractions, giving rise to significantly more trajectories, which is advantageous for subsequent statistical analysis of the behavior. Compared to paper materials, PE granules have a higher mass and lower flow resistance, so

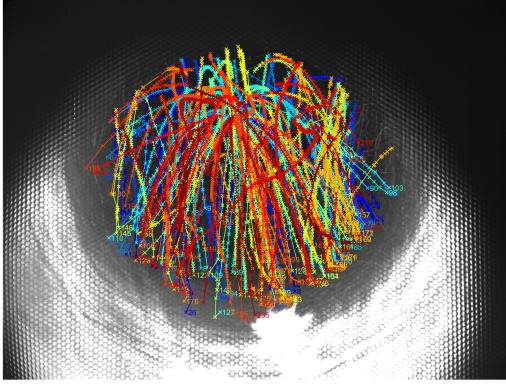
their flight speeds are mainly influenced by initial velocity and gravity. Consequently, the trajectories of PE granules are more regularly benefiting the tracking performance. Notwithstanding, due to the small size of the PE granule, the 3D coordinates of the particles provided by the camera vary considerably. As indicated by the optical ground truth, PE granules are unlikely to ignite in flight. Several PE granules ignite on impact with the hot inner wall of the rotary kiln.



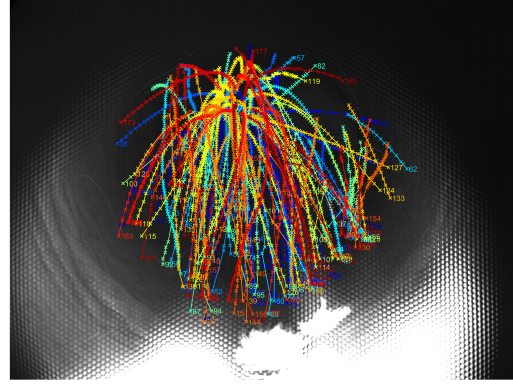
(a)



(b)



(c)



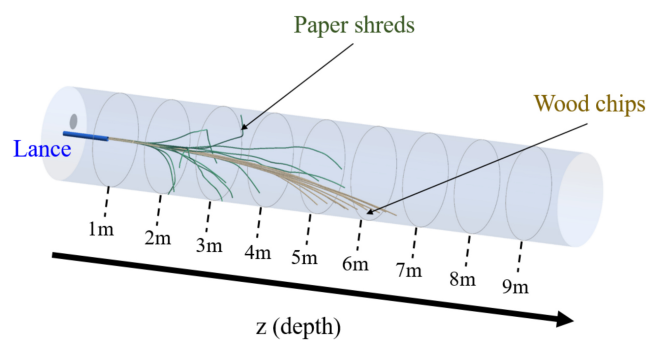
(d)

**Figure A.4.:** Optical ground truth and 2D trajectories of PE granules under 4.5 bar and 5 bar air feed pressures. (a) Optical ground truth at 4.5 bar pressure over 3029 frames. (b) Optical ground truth at 5 bar pressure over 3113 frames. (c) 2D trajectories of PE granules at 4.5 bar pressure. (d) 2D trajectories of PE granules at 5 bar pressure.

## A.2. Comparison of the experimental results with CFD simulations

The obtained flight behaviors of RDF fuel particles by the research were compared with the CFD simulations performed at the Department of Energy Plant Technology, Ruhr University Bochum, which indicates a satisfactory agreement. This section presents a general overview of the comparison outcomes. For the detailed and specific conditions concerning the CFD models, for instance, boundary conditions and drag coefficients, please refer to [109].

Figure A.5 illustrates the simulation of the BRENDA rotary kiln and example trajectories of paper shreds and wood chips. As shown in the figure, paper shreds move irregularly compared to wood chips. Moreover, wood chips could move farther in the depth direction. The observed conclusion agrees with the experimental results demonstrated in Chapter 5.3.1.



**Figure A.5.:** Simulation of the BRENDA rotary kiln and trajectories of paper shreds and wood chips. Each circle is spaced 1 m apart in the depth direction.

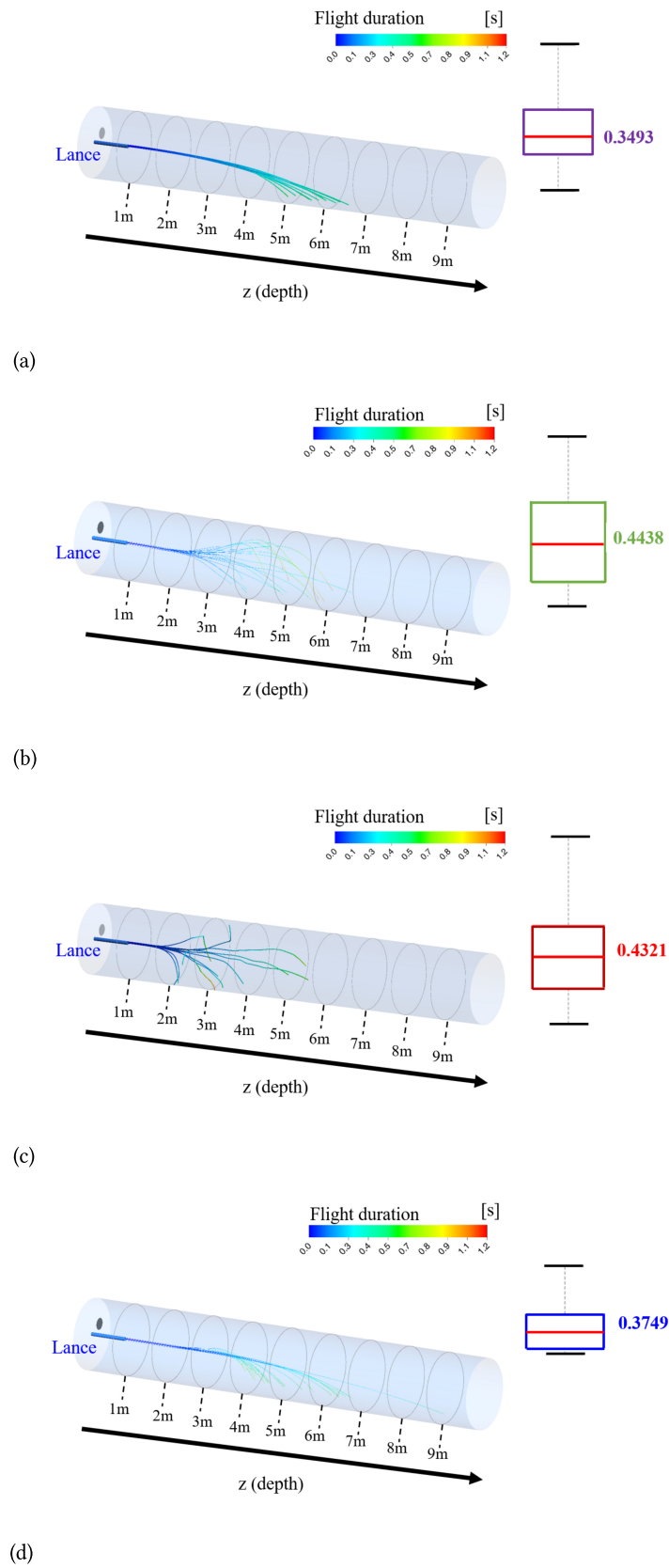
### Flight durations

The flight durations of the RDF fuel fractions under the 5 bar air feed pressure are shown in Figure A.6. The simulated trajectories are presented on the left, whose colors are plotted in accordance with the corresponding dwell times as illustrated by the color bar. The boxplot on the right side, standing for the computed flight durations by computer vision technology, is extracted from Figure 5.19 (b). As proved by Figure A.6, the simulated models agree with the findings of the study. Wood chips and PE granules land more rapidly than paper shreds and confetti, and the majority of the wood chips and PE granules land within the range of 0.3 s-0.4 s, while particles of paper materials exhibit more strongly dispersing trajectories with most flight durations beyond 0.4 s.

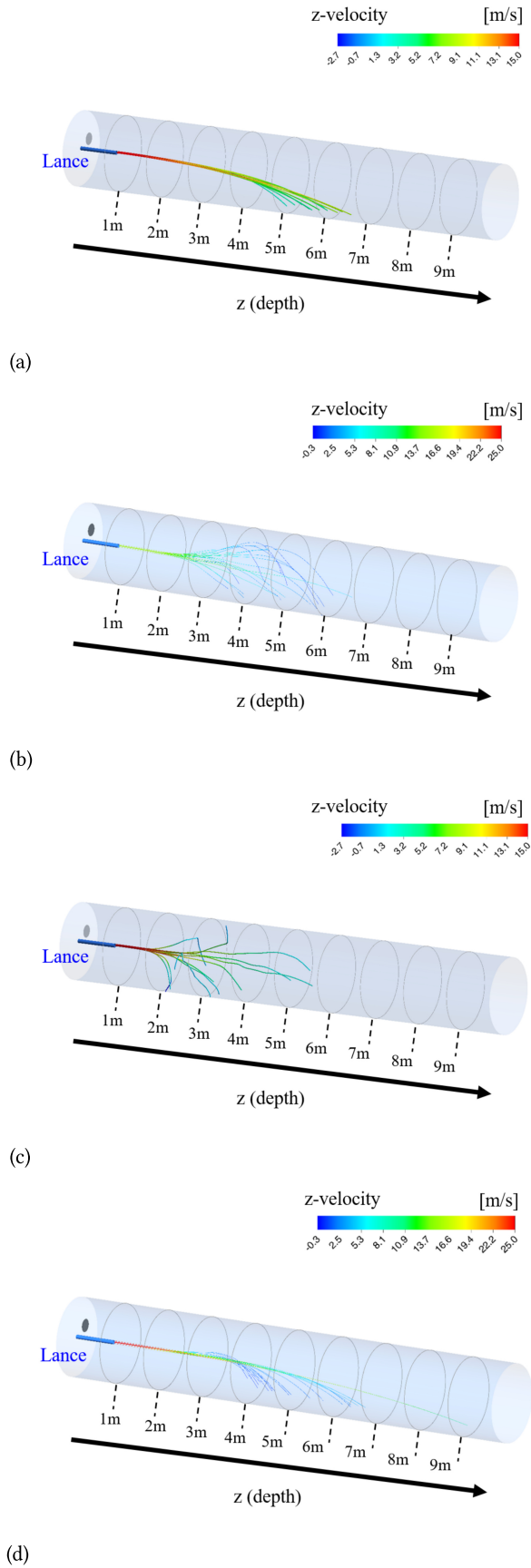
### Velocities in the depth direction

The simulated velocities of various fuel fractions in the depth direction are presented in Figure A.7. As schematically illustrated by the figure, wood chips trigger more homogeneous flight motions compared to paper materials. Despite the presence of slight irregularity in the trajectories of PE granules, they also move inerratic in the depth direction. Compared to wood chips and PE granules, confetti and paper shreds exhibit complex flight behaviors with significantly dispersing trajectories. In addition, the majority of wood chips and PE granules could penetrate more than 4 m in depth, whereas most of the paper shreds land at around 3 m. In general, the CFD models agree with the experimentally determined depth velocities in the research.





**Figure A.6.:** Comparison of simulated flight durations with experimentally determined values at 5 bar air feed pressure. (a) Wood chips. (b) Confetti. (c) Paper schreds. (d) PE granules.



**Figure A.7.:** Comparison of simulated flight velocities in the depth direction with experimental outcomes at 5 bar feed pressure. (a) Wood chips. (b) Confetti. (c) Paper schreds. (d) PE granules.



## List of Publications

- [1] M. Zhang, M. Vogelbacher, K. Aleksandrov, H.-J. Gehrman, D. Stapf, R. Streier, S. Wirtz, V. Scherer, and J. Matthes. A Novel Plenoptic Camera-Based Measurement System for the Investigation into Flight and Combustion Behavior of Refuse-Derived Fuel Particles. *ACS Omega* 8(19):16700-16712, 2023. DOI: 10.1021/acsomega.2c08004
- [2] M. Zhang, J. Matthes, K. Aleksandrov, H.-J. Gehrman, and M. Vogelbacher. Light-field camera based spatial multiple small particle tracking with post-processing. In M. A. Laribi, G. Carbone, and Z. Jiang, editors, *Advances in Automation, Mechanical and Design Engineering*, pages 95–123, Cham, Germany 2022. Springer International Publishing. 2023. DOI: 10.1007/978-3-031-09909-0\_8
- [3] M. Vogelbacher, M. Zhang, K. Aleksandrov, G. H., and M. J. Two- and three-dimensional benchmarks for particle detection from an industrial rotary kiln combustion chamber based on light-field-camera recording. *data*, 7(179), 2022. DOI: 10.3390/data7120179
- [4] M. Zhang, M. Vogelbacher, V. Hagenmeyer, K. Aleksandrov, H.-J. Gehrman, and J. Matthes. 3-D refuse-derived fuel particle tracking-by-detection using a plenoptic camera system. *IEEE Transactions on Instrumentation and Measurement*, 71:1–15, 2022. DOI: 10.1109/TIM.2022.3217858
- [5] R. Streier, S. Wirtz, K. Aleksandrov, H. Gehrman, D. Stapf, M. Zhang, M. Vogelbacher, J. Matthes, and V. Scherer. Bereitstellung von experimentell abgesicherten CFD-Modellen für die thermische Umsetzung von Ersatzbrennstoffen (FLUFF). Technical report, Industrielle Gemeinschaftsforschung (IGF), Köln, Germany, 2022.
- [6] M. Zhang, M. Vogelbacher, K. Aleksandrov, H.-J. Gehrman, D. Stapf, R. Streier, S. Wirtz, V. Scherer and J. Matthes. Determination of the aerodynamic properties of refuse derived fuel by computer vision. In *Proc. 13th European Conference on Industrial Furnaces and Boilers*, Algarve, Portugal, Apr. 2022.
- [7] R. Streier, S. Wirtz, K. Aleksandrov, H.-J. Gehrman, D. Stapf, M. Zhang, M. Vogelbacher, J. Matthes, and V. Scherer. Determination of the aerodynamic properties of refuse derived fuel by computer vision. In *Proc. 13th European Conference on Industrial Furnaces and Boilers*, Algarve, Portugal, Apr. 2022.
- [8] R. Streier, S. Wirtz, V. Scherer, K. Aleksandrov, H.-J. Gehrman, D. Stapf, M. Zhang, M. Vogelbacher, and J. Matthes. Experimentelle Absicherung von CFD-Modellen für die thermische Umsetzung von Ersatzbrennstoffen (FLUFF). In *Proc. 30. Deutscher Flammentag*, Hannover, Germany, 2021.
- [9] M. Zhang, J. Matthes, K. Aleksandrov, H.-J. Gehrman, and M. Vogelbacher. Detection and tracking of refuse-derived fuel particles based on a light field camera system. *Technisches Messen*, 88(6):396–407, 2021. DOI:10.1515/teme-2021-0016
- [10] M. Zhang, M. Vogelbacher, K. Aleksandrov, H.-J. Gehrman, and J. Matthes. Analyse des Flugund Abbrandverhaltens von Ersatzbrennstoffen auf Basis eines Lichtfeldkamarasystems. In *Proc. Forum Bildverarbeitung 2020*, Karlsruhe, Germany (Online) pages 105–117, 2020. DOI: 10.5445/IR/1000127054

